SoapUI/SoapUIPro – It is a tool to test webservices.

Initially created to test Soap based webservices but later they added features to test Rest APIs.

Automating webservices:

SoapUI – Tool

HttpClient, RestAssured, Chakram – Frameworks where we write code to test RestAPIs

What do you need in order to test webservices?

Soap- WSDl or wsdl url

Specification documents/wiki links -explain functionalty and details of mandatory/optional parameters in request and response, what operations are supported.

Authentication details.

How to test web services or what kind of testcases you come across?

Functionality Testing:

status codes- 200(get/put/delete), 404(resource not found), 201(post), 400(bad request),

202, 204,500(internal server error)

header information- content type, content length

actual response- element presence/value in response, data count, data order

valid/invalid inputs/ boundary values

authentication test

EndPoint Testing: given endpoint where the webservice is being hosted ,could have multiple endpoints depending upon the the environment ,so all that should be tested

Security Testing

LoadTesting

End-End Testing:

FunctionaltyTesting, security testing, load testing, database test

Post followed - post followed by DB query (record is created with correct data)

or

Post followed by get request.

Payload – Actual body of the response or request

request payload

response payload

generally payload - response payload

WebService request or response consist – Header, authentication, actual body

CRUD operations :

TestSuite- one or more TestCases

TestCase- Different TestSteps

Datadriven testing- DataSource , DataSourceLoop

TestSteps:

SoapRequest/RestRequest

JDBC Request-

oracle – ojdbc jar should be copied to soap ui installation folder/ lib

Properties – to load properties from external file - generally used to load project specific properties from a file and use it in multiple test steps wherever required.

PropertyTransfer- Transfer data from one test step to other test step.

Directly set property values to request: ${#TestCase#token}

Generally when we are testing dependency in webservices

eg: response of one service acts as request to other service

response of first create service request- orderid/session id

orderid/sessionid is request data to get service

Groovy:

https://www.soapui.org/scripting---properties/scripting-and-the-script-library.html#2-Groovy-Script-Library

<https://www.soapui.org/functional-testing/working-with-scripts.html>

https://www.soapui.org/scripting---properties/scripting-and-the-script-library.html

https://github.com/SmartBear/soapui-groovy-examples

<https://learnsoapui.wordpress.com/2011/07/17/10-groovy-scripts-on-your-finger-tips-soapui/>

Mock response:

When to use Mock service:

Web service is not completely developed, only few operations are implemented.

Data driven approach-

login – username, password(100)

Web service is down, and u want to perform all smoke test

Client based testing

What you will do on mock response:

Test cases, add test steps, add assertions.

Load/Performance Testing –

Threads – 5 (5 Users are using my Webservice)

Limit – total time my load test has to run

Strategy –

Simple – with given delay threads will be running

Burst – with very minimum delay threads will be running

tps, max, min, avg, err – load test success depends on these things

Security Test :

Boundry scan

Xml bomb

sql Injection

Rest API:

Http – xml

soap - xml

rest – json, xml, plaintext, csv

Soap :

soap is a standard(WSDL). w3c

XML – Heavy weight

http, smtp, ftp

Rest : - Representational state transfer

Architecture Style

http protocol

json, xml, plaintext, csv

ordermanagement

creating order – adding cart, response – order id

updating order -

deleteorder-

login

logout

<http://www.macys.com/orderManagement/creatingorder>

<http://www.macys.com/orderManagement/updatingorder>

<http://www.macys.com/orderManagement/deletingorder>

Everything it will be treated as resource – item

on this item I am doing some operation

create

delete

put

post

get – search

EndPoint – http://www.macys.com/orderManagement

Resource – createorder

Assertions :

Json path count -$[0].id – gives the count of id

Json path match - $[0].id – gives value of the id

$[0].common\_category.name

**Whiteboxqa.com – jan batch – soapUI3 – last 15min (Data driven testing in soapUI pro)**

**For step by step implementation of data driven testing in SoapUI pro**

[**https://www.soapui.org/data-driven-tests/functional-tests.html**](https://www.soapui.org/data-driven-tests/functional-tests.html)