**UI applications, Webservices testing,Mobile App testers…**

**WebService** : S/W available over the intenet which receives request from the user and sends back the response(xml, json) .

**Rest Vs Soap:**

**Rest**: light weight ws, http protocol, ues xml, json, html , csv etc, rest uses only http

**Soap** :WSDL – defintion file assiciated with soap, exclusively uses only xml, have its defined security called ws- security, built in error handling,

Language, platform and transport independent

**Tools To test webservices**: SoapUI,browser plugins like postman, adv rest client etc, itco lisa, jmeter, programming languages like java – http client API, apche axis

**SoapUI –** tool to test Soap and rest webservices

<https://www.soapui.org/downloads/soapui.html>

SoapUI pro : added benefits compared to your open source tool.

To test soap web service – we need wsdl file or wsdl url

**What to test in a webservice:**

1. Status code :

(Post – 201 – resource created succesfully

400 – bad request

Trying to create same resource again / duplicate – 409

Delete – 204 – no content

After you delete – if you do a get – 404 – resource not found)

1. Response correctness –

* data fields
* existence of certain elements, associated value of certain elements(xpaths to locate elements)
* count of the some e lements
* data order
* data types

1. Postive test cases, negative test cases (incorrect input i.e., incorrect zipcode)
2. Content type, content length, headers
3. Functional testin g : test different operations, end point testing(server at which webservice is running)
4. Load Testing: how many concurrent requests can be sent.
5. Security Testing – it should not accept malicious requests

**Type of testing involved with webservices:**

BlackBox

Whitebox – access to code, acces to logs

Grey box testing

SmokeTesting and Functionaltesting:

**Create a soap project using wsdl url or wsdl file**

**Understamd the webservice using wsdl**

**Rightclick on the request – add to test case- give names for suite and testcases and cklick ok**

**testSuite – testcase – test step**

**Assertions-(what to test) on webservices**

**PropertyTransfer/ Data driven testing Using SoapUI:**

**Pass the respon se of one webservice as input to other webservice**

**DataDriven testing – Singlesetofdata**

**DataSource and DataSourceloop:**

**Iterate Multiple set of data – u can do in soap UI pro**

**SoapUi- has support to create and run groovy script**

**Ws1 -Creates a Loan – generates LoanID – post(create), update**

**Ws2-UsingLoanID- It updates the DB records**

**Ws3- send email to customer that your loan is approved**

**Loan is closed - Delete the loan – delete**

**Get details of the loan – get(read)**

**Db Validation – u can do that using DBREquest**

**Driver, COnnectionString, download ojdbc jar and place it in soapui/lib, restart soapui**

**oracle.jdbc.OracleDriver**

**jdbc:oracle:thin:atgprod/atgprod@192.168.29.148:1521:XE**

**Execute queries**

**//Row[@rowNumber=’2’]/LOAN\_ID**

**Groovy - SuperSet of Java and it runs in jvm**

**Why do we need scripting – to make executions based on logical decisions, conditional executions**

**Something logical – u need programming assistance along with ur tool.**

**Variables- datatypes (String, Integer,decimal ,Boolean etc)**

**Operators- + - % \***

**Conditional statements- if(sal>1000){ then log.info(“he have basic salary”)}, case**

**Looping - for, while**

**For(i=1; i<=10;i++){**

**Log.info(emp[i].name)**

**}**

**Conditional goto**

*AIzaSyDAsITAvPHosKVIOO-Z4oZf9NolyiINCPU*

content\_copy

REST project

LoadTesting – to check the performance of website when you send some concurrent users.

Threads – no of users

Security testing

**Db to test**

**Delay = testDelay- randomNumberBetween(0, testdelay\*random)**

**Testdelay-600, random 0.5- randomnum (0-300)**

**Malformed xml**

**Sql injection**

**Boundary scan**

**MockService : wsdl or have rest api url to test – but its not readily available- may be issue with server, coding is not done….**

**Testdriven development – where webservice functionalty is not fully implemented**

**Client testing**

**Rapid webservices testing**