**Assignment-5**

**Testing SOAP Web Service with SoapUI**

* Schema compliance — ensuring the response adheres to the WSDL schema.
* Correct responses — verifying the service returns expected data for valid requests.
* Error scenarios — checking how the service behaves with invalid inputs.

**Tools Used**

* SoapUI (Open Source) — a tool for testing SOAP and REST web services.

**Sample SOAP Request XML**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns="http://example.com/namespace">

<soapenv:Header/>

<soapenv:Body>

<ns:YourOperation>

<ns:parameter1>validValue</ns:parameter1>

<ns:parameter2>123</ns:parameter2>

</ns:YourOperation>

</soapenv:Body>

</soapenv:Envelope>

**Groovy Script for Assertion**

// Get response from test step named "Request 1"

def response = context.testCase.testSteps["Request 1"].getProperty("response").value

// Assert response contains expected success status

assert response.contains("<status>success</status>"), "Response does not contain success status"

**Groovy Script to Test Error Scenario**

def requestStep = testRunner.testCase.testSteps["Request 1"]

def invalidRequest = '''<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns="http://example.com/namespace">

<soapenv:Header/>

<soapenv:Body>

<ns:YourOperation>

<ns:parameter1></ns:parameter1> <!-- Missing required value -->

<ns:parameter2>abc</ns:parameter2> <!-- Invalid data type -->

</ns:YourOperation>

</soapenv:Body>

</soapenv:Envelope>'''

requestStep.testRequest.requestContent = invalidRequest

requestStep.run(testRunner, context)

def response = requestStep.testRequest.response.contentAsString

assert response.contains("Fault") || response.contains("error"), "Expected error response but got success"