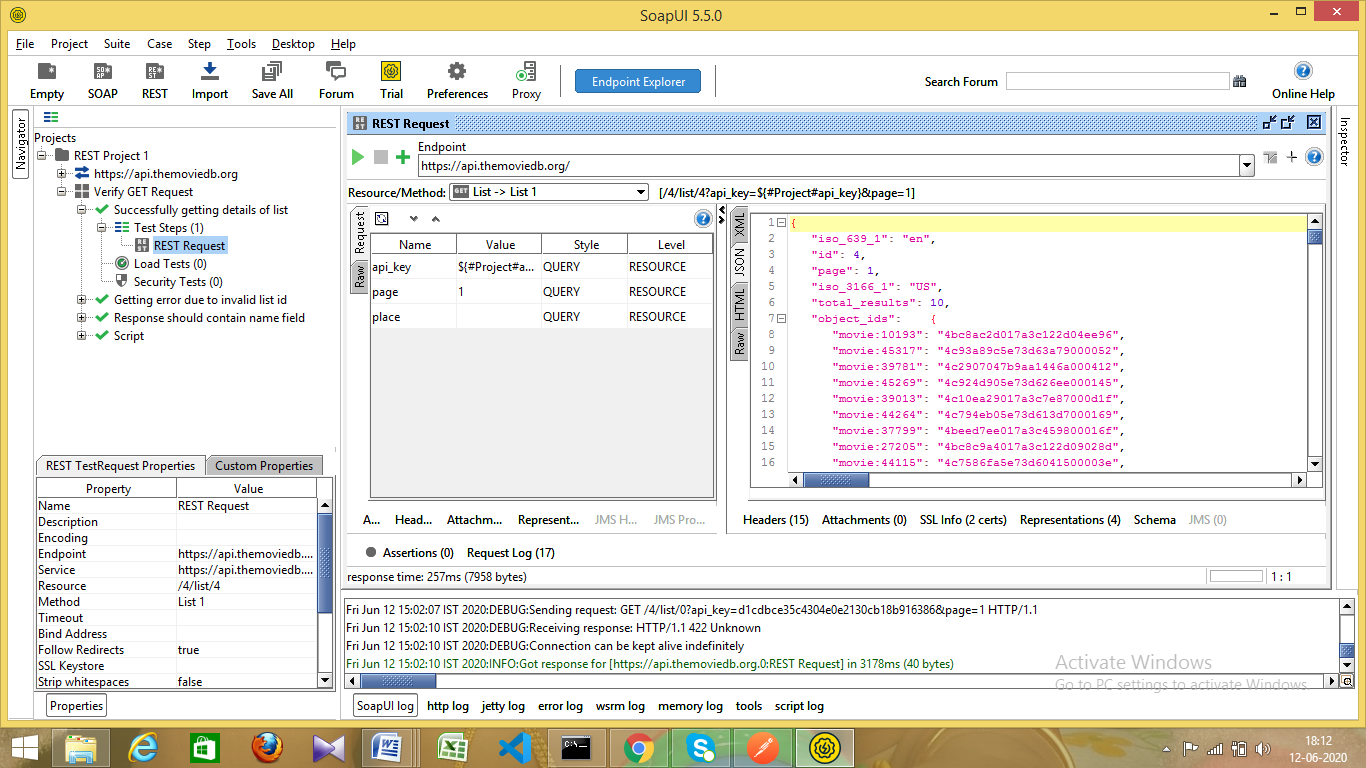
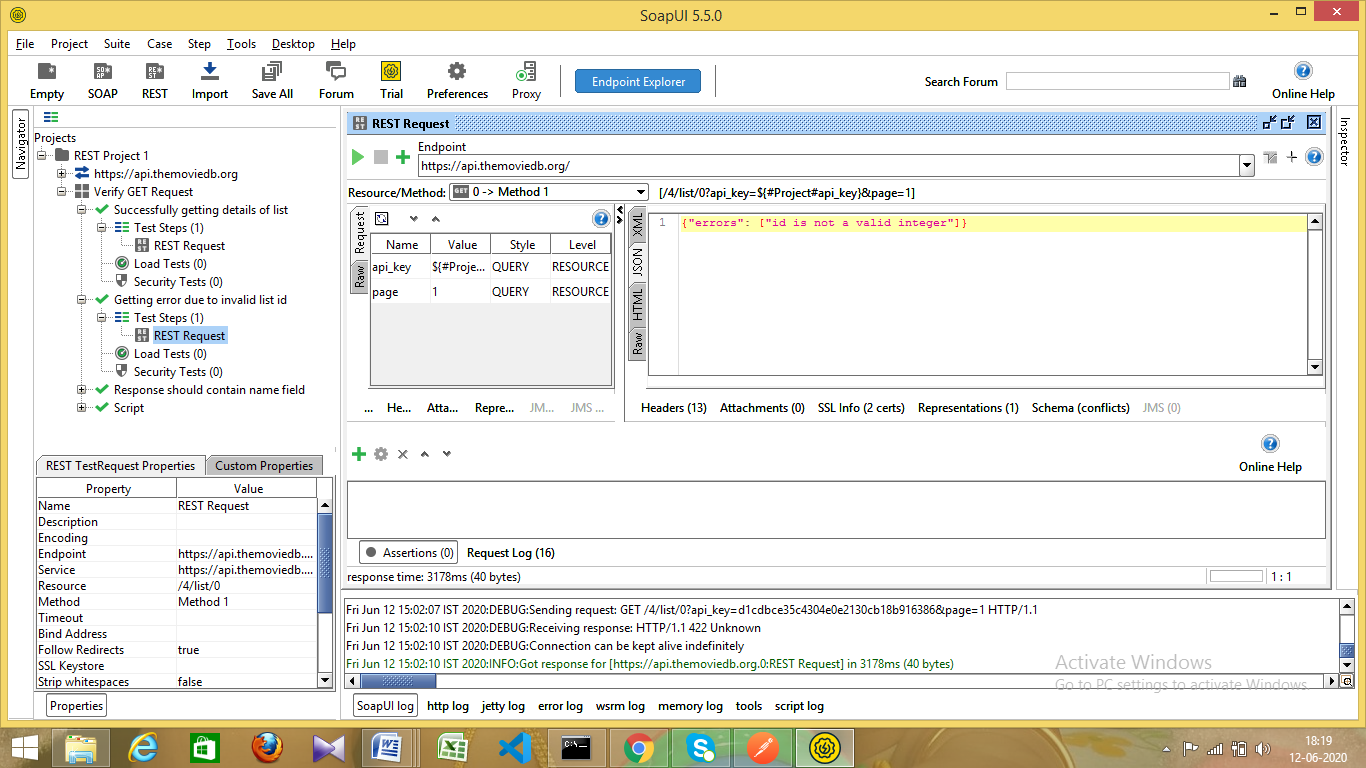


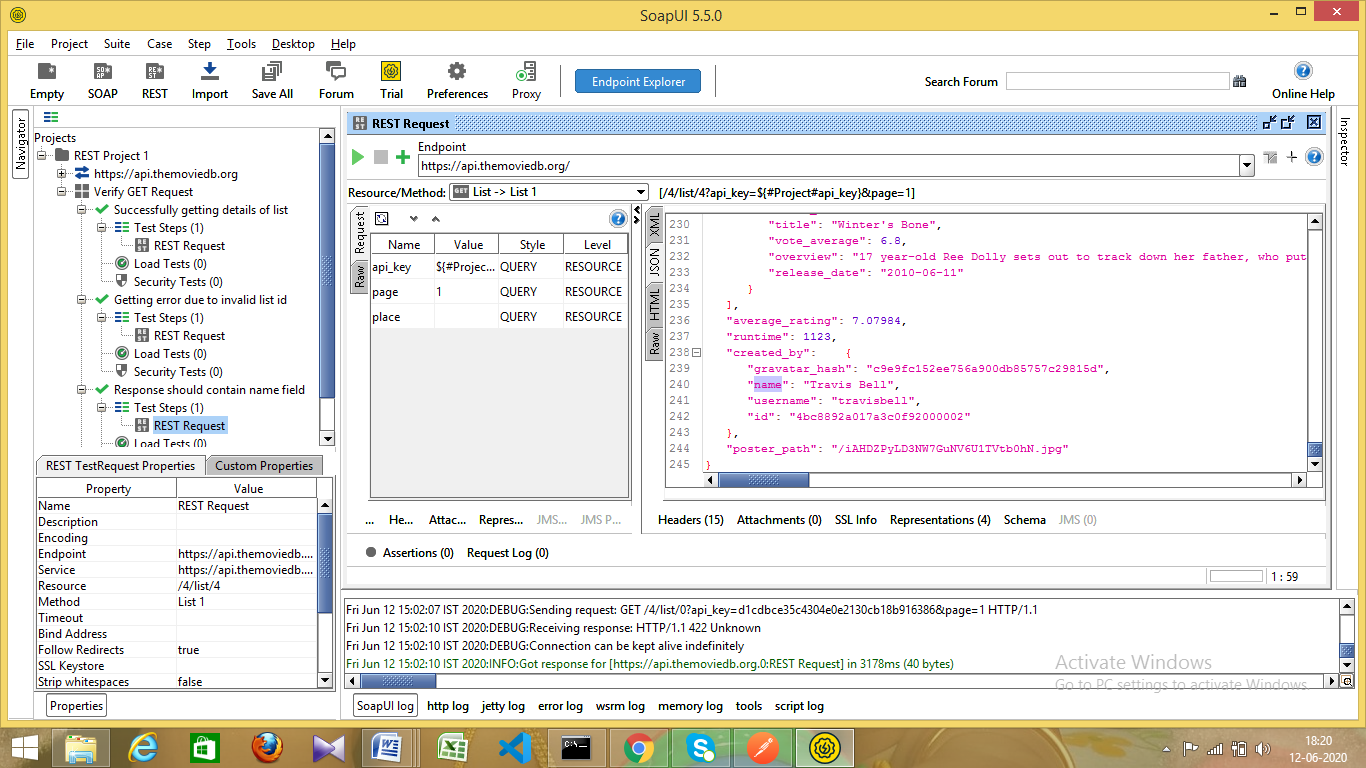
**Test Case 1: Getting successful response**



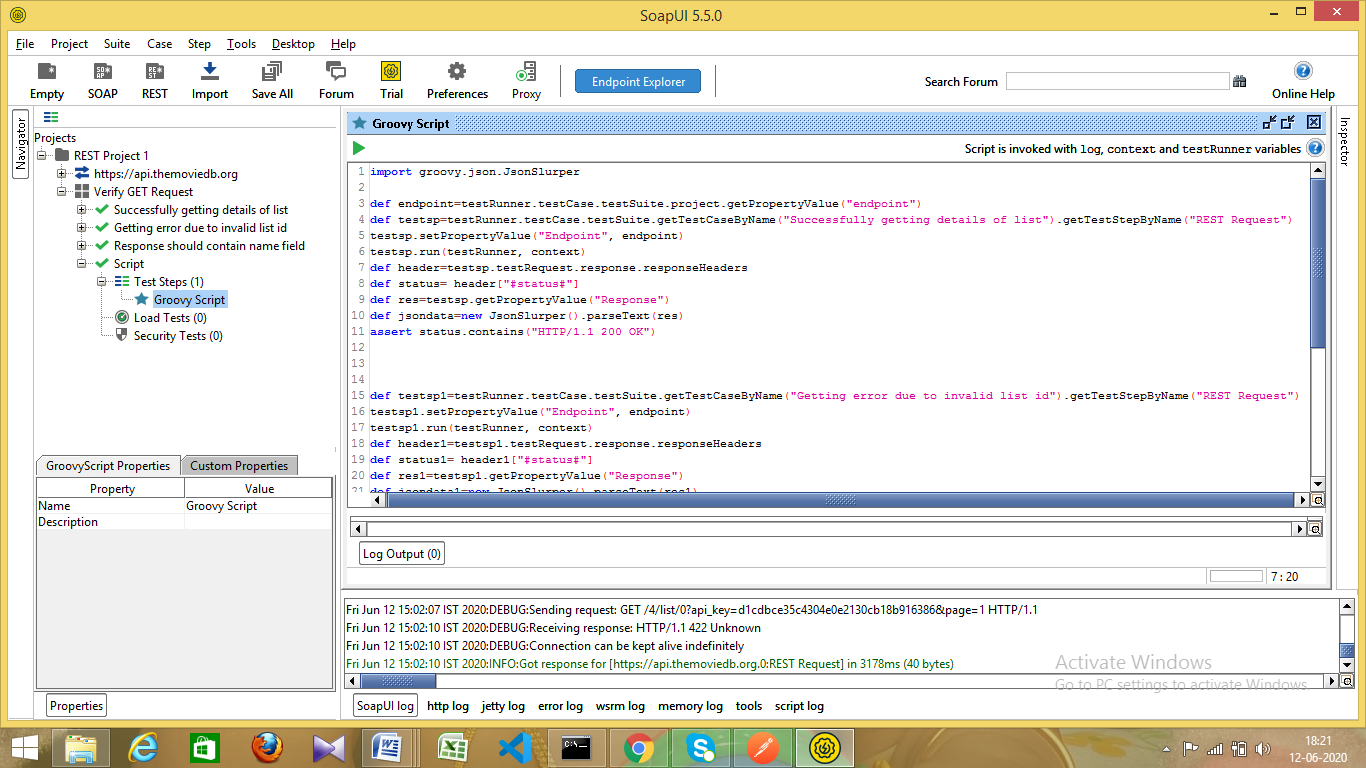
**Test Case 2: Getting error due to invalid list id**



**Test Case 3: Response should contain name field**



**Groovy Script to run all these test cases:**



**Code:**

import groovy.json.JsonSlurper

def endpoint=testRunner.testCase.testSuite.project.getPropertyValue("endpoint")

def testsp=testRunner.testCase.testSuite.getTestCaseByName("Successfully getting details of list").getTestStepByName("REST Request")

testsp.setPropertyValue("Endpoint", endpoint)

testsp.run(testRunner, context)

def header=testsp.testRequest.response.responseHeaders

def status= header["#status#"]

def res=testsp.getPropertyValue("Response")

def jsondata=new JsonSlurper().parseText(res)

assert status.contains("HTTP/1.1 200 OK")

def testsp1=testRunner.testCase.testSuite.getTestCaseByName("Getting error due to invalid list id").getTestStepByName("REST Request")

testsp1.setPropertyValue("Endpoint", endpoint)

testsp1.run(testRunner, context)

def header1=testsp1.testRequest.response.responseHeaders

def status1= header1["#status#"]

def res1=testsp1.getPropertyValue("Response")

def jsondata1=new JsonSlurper().parseText(res1)

assert status1.contains("HTTP/1.1 422 Unknown")

assert jsondata1.errors != null

def testsp3=testRunner.testCase.testSuite.getTestCaseByName("Response should contain name field").getTestStepByName("REST Request")

testsp3.setPropertyValue("Endpoint", endpoint)

testsp3.run(testRunner, context)

def header3=testsp3.testRequest.response.responseHeaders

def status3= header3["#status#"]

def res3=testsp3.getPropertyValue("Response")

def jsondata3=new JsonSlurper().parseText(res3)

assert jsondata.name != null