

SMARTINDONASIA_API_SmokeTest

Execution Environment

| | |
|-----------------|---|
| Host name | Rajesh.Veeranna - VSPLBLRLT0060.visionetsystems.com |
| OS | Windows 7 64bit |
| Katalon version | 6.3.3.11 |
| Browser | |

Summary

| | | | |
|-------------|--|------------|---------------------|
| ID | Test Suites/SMARTINDONASIA_API_SmokeTest | | |
| Description | | | |
| Total | 32 | | |
| Passed | 31 | Failed | 1 |
| Error | 0 | Incomplete | 0 |
| Start | 2019-09-26 15:18:39 | End | 2019-09-26 15:19:43 |
| Elapsed | 1m - 3.724s | | |

| # | ID | Description | Status |
|----|---|-------------|--------|
| 1 | Test Cases/a.ValidateMobileNumber_Test | | PASSED |
| 2 | Test Cases/a/Validate_OTPTest | | PASSED |
| 3 | Test Cases/b/InsertUser_CheckInTest | | PASSED |
| 4 | Test Cases/b/UpdateUser_CheckInTest | | PASSED |
| 5 | Test Cases/c/InsertTripDetails_Test | | PASSED |
| 6 | Test Cases/c/UpdateTripDetails_Test | | PASSED |
| 7 | Test Cases/d.VersionMaster_Test | | PASSED |
| 8 | Test Cases/d/LeaveType_Test | | PASSED |
| 9 | Test Cases/e/LeaveApplication_Test | | PASSED |
| 10 | Test Cases/e/LeaveCancelApplicationTest | | PASSED |
| 11 | Test Cases/g/CropMasterTest | | PASSED |
| 12 | Test Cases/h/ProductMaster_Test | | PASSED |
| 13 | Test Cases/i/UserCropProduct_Test | | PASSED |
| 14 | Test Cases/j/UserGeographyDetails_Test | | PASSED |
| 15 | Test Cases/k/UserTimeline_Test | | PASSED |
| 16 | Test Cases/l/SeverityMaster_Test | | PASSED |

SMARTINDONASIA_API_SmokeTest

| # | ID | Description | Status |
|----|--|-------------|--------|
| 17 | Test Cases/mQuestionAnswerMaster_Test | | PASSED |
| 18 | Test Cases/nHolidayMaster_Test | | PASSED |
| 19 | Test Cases/oRefrenceMaster_Test | | FAILED |
| 20 | Test Cases/pLobByUser_Test | | PASSED |
| 21 | Test Cases/qActivityHistory_Test | | PASSED |
| 22 | Test Cases/RoleAccessUser_Test | | PASSED |
| 23 | Test Cases/sIncidentHistory_Test | | PASSED |
| 24 | Test Cases/tMergeContactDetail_Test | | PASSED |
| 25 | Test Cases/vInsertIncident_Test | | PASSED |
| 26 | Test Cases/xDashboard_Test | | PASSED |
| 27 | Test Cases/wDiseaseMaster_Test | | PASSED |
| 28 | Test Cases/zMobileAppVersion_Test | | PASSED |
| 29 | Test Cases/zActivityCategory_Test | | PASSED |
| 30 | Test Cases/z1SupervisorTeamStatus_Test | | PASSED |
| 31 | Test Cases/z2Supervisor Users | | PASSED |
| 32 | Test Cases/x1TargetBreakDown_Test | | PASSED |

a.ValidateMobileNumber_Test

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/a.ValidateMobileNumber_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:40 | End | 2019-09-26 15:18:44 |
| Elapsed | 4.024s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequest(findTestObject("aValidate Mobile Number", ["protocal":Endpoint])) Send request successfully | 2.093s | PASSED |
| 2 | response = sendRequestAndVerify(findTestObject("aValidate Mobile Number", ["protocal":Endpoint])) Send request and verify the response successfully | 1.404s | PASSED |
| 3 | slurper = new groovy.json.JsonSlurper() | 0.010s | PASSED |
| 4 | result = slurper.parseText(response.getResponseText()) | 0.022s | PASSED |
| 5 | println("*****") | 0.010s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM VALIDATEMOBILENUMBER TEST CASE = " + result) | 0.000s | PASSED |

a.ValidateMobileNumber_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 7 | println("*****") | 0.000s | PASSED |
| 8 | value = uniqueTokenCode | 0.005s | PASSED |
| 9 | println(value) | 0.003s | PASSED |
| 10 | uniquetoken = value | 0.002s | PASSED |
| 11 | println("*****") | 0.007s | PASSED |
| 12 | println("UNIQ TOKEN ID IS GENARATED = " + uniquetoken) | 0.004s | PASSED |
| 13 | println("*****") | 0.003s | PASSED |

aValidate_OTPTest

Information

| | | | |
|-------------|------------------------------|-----|---------------------|
| ID | Test Cases/aValidate_OTPTest | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:44 | End | 2019-09-26 15:18:45 |
| Elapsed | 1.262s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequest(findTestObject("aValidate OTP", ["uniqueToken":uniquetoken, "userId":userid])) Send request successfully | 0.393s | PASSED |
| 2 | response1 = sendRequest(findTestObject("aValidate Mobile Number")) Send request successfully | 0.354s | PASSED |
| 3 | slurper1 = new groovy.json.JsonSlurper() | 0.001s | PASSED |
| 4 | result1 = slurper1.parseText(responseBodyContent) | 0.001s | PASSED |
| 5 | println("*****") | 0.001s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM VALIDATEOTP TEST CASE = " + result1) | 0.000s | PASSED |
| 7 | println("*****") | 0.000s | PASSED |

aValidate_OTPTest

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 8 | value1 = userId | 0.001s | PASSED |
| 9 | userid = value1 | 0.000s | PASSED |
| 10 | println("*****") | 0.000s | PASSED |
| 11 | println("THE USER ID IS GENARATED = " + userid) | 0.001s | PASSED |
| 12 | println("*****") | 0.001s | PASSED |

bInsertUser_CheckInTest

Information

| | | | |
|-------------|------------------------------------|-----|---------------------|
| ID | Test Cases/bInsertUser_CheckInTest | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:45 | End | 2019-09-26 15:18:46 |
| Elapsed | 0.868s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | chkInID = toString().toUpperCase() | 0.025s | PASSED |
| 2 | println(chkInID) | 0.000s | PASSED |
| 3 | f = new java.io.File(D:\SmartDoxApiResponse\CtID.txt) | 0.000s | PASSED |
| 4 | f.write(chkInID) | 0.000s | PASSED |
| 5 | now = new java.util.Date() | 0.010s | PASSED |
| 6 | chkInTime = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.010s | PASSED |
| 7 | println(chkInTime) | 0.000s | PASSED |

bInsertUser_CheckInTest

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 8 | <pre>resp3 = sendRequest(findTestObject("bInsert User CheckIn", ["uniqueToken":uniquetoken, "userId":userid, "checkInTime":chkInTime, "checkInID":chkInID, "checkInLocality":checkInLocality]))</pre> <p>Send request successfully</p> | 0.308s | PASSED |
| 9 | <pre>slurper3 = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 10 | <pre>result3 = slurper3.parseText(resp3.getResponseText())</pre> | 0.000s | PASSED |
| 11 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 12 | <pre>println("THIS IS THE RESPONSE FROM USER_CHECKIN TEST CASE = " + result3)</pre> | 0.010s | PASSED |
| 13 | <pre>println("*****")</pre> | 0.004s | PASSED |
| 14 | <pre>toString().split(chkInID, 0)</pre> | 0.015s | PASSED |
| 15 | <pre>value3 = toString().substring(11, 47)</pre> | 0.001s | PASSED |
| 16 | <pre>println(value3)</pre> | 0.010s | PASSED |

bInsertUser_CheckInTest

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 17 | println("THE CHECKIN iD IS " + value3) | 0.003s | PASSED |
| 18 | f = new java.io.File(D:\Checkin_ID.txt) | 0.003s | PASSED |
| 19 | f.write(value3) | 0.007s | PASSED |

bUpdateUser_CheckInTest

Information

| | | | |
|-------------|------------------------------------|-----|---------------------|
| ID | Test Cases/bUpdateUser_CheckInTest | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:46 | End | 2019-09-26 15:18:47 |
| Elapsed | 0.703s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | file3 = new java.io.File(D:\Checkin_ID.txt) | 0.010s | PASSED |
| 2 | checkinID = file3.readLine() | 0.000s | PASSED |
| 3 | println("----->" + checkinID) | 0.000s | PASSED |
| 4 | now1 = new java.util.Date() | 0.001s | PASSED |
| 5 | chkOutTime = now1.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 6 | println(chkOutTime) | 0.000s | PASSED |
| 7 | resp4 = sendRequest(findTestObject("bUpdate User CheckIn", ["uniqueToken":uniquetoken, "userId":userid, "checkinID":checkinID, "checkedOutTime":chkOutTime])) | 0.213s | PASSED |

Send request successfully

cInsertTripDetails_Test

Information

| | | | |
|-------------|------------------------------------|-----|---------------------|
| ID | Test Cases/cInsertTripDetails_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:47 | End | 2019-09-26 15:18:48 |
| Elapsed | 1.852s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("aValidate OTP", [{"uniqueToken":uniquetoken, "userId":userid}])) Send request and verify the response successfully | 0.784s | PASSED |
| 2 | triPID = toString().toUpperCase() | 0.002s | PASSED |
| 3 | println(triPID) | 0.001s | PASSED |
| 4 | now = new java.util.Date() | 0.001s | PASSED |
| 5 | tiMe = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.001s | PASSED |
| 6 | println(tiMe) | 0.001s | PASSED |

cInsertTripDetails_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 7 | resp2 = sendRequest(findTestObject("cInsert Trip Detail", ["uniqueToken":uniquetoken, "userId":userid, "tripid":triPID, "startriptime":tiMe, "starttripLocality":starttripLocality])) Send request successfully | 0.303s | PASSED |
| 8 | slurper2 = new groovy.json.JsonSlurper() | 0.001s | PASSED |
| 9 | result2 = slurper2.parseText(resp2.getResponseText()) | 0.003s | PASSED |
| 10 | println("*****") | 0.001s | PASSED |
| 11 | println("THIS IS THE RESPONSE FROM INSERT TRIP DETAILS TEST CASE = " + result2) | 0.010s | PASSED |
| 12 | println("*****") | 0.010s | PASSED |
| 13 | toString().split(triPID, 0) | 0.000s | PASSED |
| 14 | value2 = toString().substring(25, 61) | 0.020s | PASSED |
| 15 | println(value2) | 0.000s | PASSED |

cInsertTripDetails_Test

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 16 | println("THE TRIP iD IS " + value2) | 0.000s | PASSED |
| 17 | f = new java.io.File(D:\InsertTripDetails_ID.txt) | 0.245s | PASSED |
| 18 | f.write(value2) | 0.003s | PASSED |

cUpdateTripDetails_Test

Information

| | | | |
|-------------|------------------------------------|-----|---------------------|
| ID | Test Cases/cUpdateTripDetails_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:48 | End | 2019-09-26 15:18:49 |
| Elapsed | 0.677s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | file1 = new java.io.File(D:\InsertTripDetails_ID.txt) | 0.000s | PASSED |
| 2 | triPID = file1.readlines() | 0.000s | PASSED |
| 3 | println("----->" + triPID) | 0.000s | PASSED |
| 4 | sendRequest(findTestObject("aValidate OTP", ["uniqueToken":uniquetoken, "userId":userid])) Send request successfully | 0.229s | PASSED |
| 5 | sendRequest(findTestObject("cUpdate Trip Detail", ["uptripID":triPID, "userId":userid])) Send request successfully | 0.209s | PASSED |

d.VersionMaster_Test

Information

| | | | |
|-------------|---------------------------------|-----|---------------------|
| ID | Test Cases/d.VersionMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:49 | End | 2019-09-26 15:18:50 |
| Elapsed | 0.945s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("d.Version Master", ["uniqueToken":uniquetoken])) | 0.638s | PASSED |
| | Send request and verify the response successfully | | |

dLeaveType_Test

Information

| | | | |
|-------------|----------------------------|-----|---------------------|
| ID | Test Cases/dLeaveType_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:50 | End | 2019-09-26 15:18:52 |
| Elapsed | 1.461s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("eLeave Type", [{"versiontableId":versiontableId, "userId":userid, "uniqueToken":uniquetoken}])) Send request and verify the response successfully | 1.221s | PASSED |

eLeaveApplication_Test

Information

| | | | |
|-------------|-----------------------------------|-----|---------------------|
| ID | Test Cases/eLeaveApplication_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:52 | End | 2019-09-26 15:18:53 |
| Elapsed | 0.896s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("eLeave Application", [{"leavetypeId":leavetypeId, "fromdate":fromdate, "todate":todate, "comments":comments, "userId":userid, "uniqueToken":uniquetoken}])) | 0.630s | PASSED |
| Send request and verify the response successfully | | | |

eLeaveCancelApplicationTest

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/eLeaveCancelApplicationTest | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:53 | End | 2019-09-26 15:18:53 |
| Elapsed | 0.963s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("eLeave Cancel Application", [{"userId":userid, "uniqueToken":uniquetoken, "fromdate":fromdate, "todate":todate}])) Send request and verify the response successfully | 0.673s | PASSED |

gCropMasterTest

Information

| | | | |
|-------------|----------------------------|-----|---------------------|
| ID | Test Cases/gCropMasterTest | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:53 | End | 2019-09-26 15:18:55 |
| Elapsed | 1.435s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("gCrop-Master", ["versiontableId":versiontableId, "userId":userid, "uniqueToken":uniquetoken])) Send request and verify the response successfully | 1.187s | PASSED |

hProductMaster_Test

Information

| | | | |
|-------------|--------------------------------|-----|---------------------|
| ID | Test Cases/hProductMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:55 | End | 2019-09-26 15:18:56 |
| Elapsed | 0.888s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("iUser_Crop_Product", [{"versiontableId": "", "userId": "userid", "uniqueToken": "uniquetoken"}])) | 0.667s | PASSED |
| Send request and verify the response successfully | | | |

iUserCropProduct_Test

Information

| | | | |
|-------------|----------------------------------|-----|---------------------|
| ID | Test Cases/iUserCropProduct_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:56 | End | 2019-09-26 15:18:57 |
| Elapsed | 1.189s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("iUser_Crop_Product", [{"versiontableId": "", "userId": "userid", "uniqueToken": "uniquetoken"}])) Send request and verify the response successfully | 0.937s | PASSED |

jUserGeographyDetails_Test

Information

| | | | |
|-------------|---------------------------------------|-----|---------------------|
| ID | Test Cases/jUserGeographyDetails_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:57 | End | 2019-09-26 15:18:58 |
| Elapsed | 1.318s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("jUser Geography Details", ["versiontableId":versiontableId, "userId":userid, "uniqueToken":uniquetoken])) Send request and verify the response successfully | 1.049s | PASSED |

kUserTimeline_Test

Information

| | | | |
|-------------|-------------------------------|-----|---------------------|
| ID | Test Cases/kUserTimeline_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:58 | End | 2019-09-26 15:18:59 |
| Elapsed | 1.105s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("kUserTimeline", [{"fromdaTe":fromdaTe, "todaTe":todaTe, "userId":userid, "uniqueToken":uniquetoken}])) | 0.794s | PASSED |
| Send request and verify the response successfully | | | |

ISeverityMaster_Test

Information

| | | | |
|-------------|---------------------------------|-----|---------------------|
| ID | Test Cases/ISeverityMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:18:59 | End | 2019-09-26 15:19:00 |
| Elapsed | 1.076s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("ISeverity Master", ["versiontableId":versiontableId, "userId":userid, "uniqueToken":uniquetoken])) Send request and verify the response successfully | 0.818s | PASSED |

mQuestionAnswerMaster_Test

Information

| | | | |
|-------------|---------------------------------------|-----|---------------------|
| ID | Test Cases/mQuestionAnswerMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:00 | End | 2019-09-26 15:19:01 |
| Elapsed | 0.924s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("mQuestion Answer Master", [{"lobID":lobID, "userId":userid, "uniqueToken":uniquetoken}])) Send request and verify the response successfully | 0.680s | PASSED |

nHolidayMaster_Test

Information

| | | | |
|-------------|--------------------------------|-----|---------------------|
| ID | Test Cases/nHolidayMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:01 | End | 2019-09-26 15:19:03 |
| Elapsed | 1.177s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("nHoliday Master", ["versiontableId":versiontableId, "uniqueToken":uniquetoken, "userId":userid])) Send request and verify the response successfully | 0.794s | PASSED |

oReferenceMaster_Test

Information

| | | | |
|-------------|----------------------------------|-----|---------------------|
| ID | Test Cases/oReferenceMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:03 | End | 2019-09-26 15:19:05 |
| Elapsed | 2.552s | | |
| Status | FAILED | | |

Details

| # | Description | Elapsed | Status |
|---|-------------|---------|--------|
|---|-------------|---------|--------|

oReferenceMaster_Test

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | <p>sendRequestAndVerify(findTestObject("oReference Master", ["uniqueToken":uniquetoken, "userId":userid]))</p> <p>Unable to send request and verify the response (Root cause: com.kms.katalon.core.exception.StepFailedException: Verification failed at com.kms.katalon.core.webservice.keyword.builtin. SendRequestAndVerifyKeyword\$_sendRequestAndVerify_closure1.doCall (SendRequestAndVerifyKeyword.groovy:59) at com.kms.katalon.core.webservice.keyword.builtin. SendRequestAndVerifyKeyword\$_sendRequestAndVerify_closure1.call (SendRequestAndVerifyKeyword.groovy) at com.kms.katalon.core.keyword.internal.KeywordMain.runKeyword (KeywordMain.groovy:68) at com.kms.katalon.core.webservice.keyword.builtin. SendRequestAndVerifyKeyword.sendRequestAndVerify(SendRequestAndVerifyKeyword.groovy:43) at com.kms.katalon.core.webservice.keyword.builtin. SendRequestAndVerifyKeyword.execute(SendRequestAndVerifyKeyword.groovy:38) at com.kms.katalon.core.keyword.internal.KeywordExecutor. executeKeywordForPlatform(KeywordExecutor.groovy:56) at com.kms.katalon.core.webservice.keyword.WSBuiltInKeywords. sendRequestAndVerify(WSBuiltInKeywords.groovy:65) at com.kms.katalon.core.webservice.keyword. WSBuiltInKeywords\$sendRequestAndVerify\$0.call(Unknown Source) at oReferenceMaster_Test.run(oReferenceMaster_Test:16) at com.kms.katalon.core.main.ScriptEngine.run(ScriptEngine.java:194) at com.kms.katalon.core.main.ScriptEngine.runScriptAsRawText(ScriptEngine. java:119) at com.kms.katalon.core.main.TestCaseExecutor.runScript(TestCaseExecutor. java:337) at com.kms.katalon.core.main.TestCaseExecutor.doExecute(TestCaseExecutor. java:328) at com.kms.katalon.core.main.TestCaseExecutor.processExecutionPhase (TestCaseExecutor.java:307) at com.kms.katalon.core.main.TestCaseExecutor.accessMainPhase (TestCaseExecutor.java:299) at com.kms.katalon.core.main.TestCaseExecutor.execute(TestCaseExecutor. java:233) at com.kms.katalon.core.main.TestSuiteExecutor.accessTestCaseMainPhase (TestSuiteExecutor.java:169) at com.kms.katalon.core.main.TestSuiteExecutor.accessTestSuiteMainPhase (TestSuiteExecutor.java:142) at com.kms.katalon.core.main.TestSuiteExecutor.execute(TestSuiteExecutor.java: 91) at com.kms.katalon.core.main.TestCaseMain.startTestSuite(TestCaseMain.java: 157) at com.kms.katalon.core.main.TestCaseMain\$startTestSuite\$0.call(Unknown Source) at TempTestSuite1569491316428.run(TempTestSuite1569491316428.groovy:35))</p> | 2.316s | FAILED |

pLobByUser_Test

Information

| | | | |
|-------------|----------------------------|-----|---------------------|
| ID | Test Cases/pLobByUser_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:05 | End | 2019-09-26 15:19:06 |
| Elapsed | 0.799s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("pLob By User", [{"uniqueToken":uniquetoken, "userId":userid}])) | 0.536s | PASSED |
| Send request and verify the response successfully | | | |

qActivityHistory_Test

Information

| | | | |
|-------------|----------------------------------|-----|---------------------|
| ID | Test Cases/qActivityHistory_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:06 | End | 2019-09-26 15:19:22 |
| Elapsed | 16.445s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("qActivity History", [{"userId":userid, "uniqueToken":uniquetoken}])) Send request and verify the response successfully | 16.154s | PASSED |

RoleAccessUser_Test

Information

| | | | |
|-------------|--------------------------------|-----|---------------------|
| ID | Test Cases/RoleAccessUser_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:22 | End | 2019-09-26 15:19:24 |
| Elapsed | 1.655s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("Role Access By User", ["uniqueToken":uniquetoken, "userId":userid])) Send request and verify the response successfully | 0.537s | PASSED |
| 2 | responseR = sendRequestAndVerify(findTestObject("Role Access By User")) Send request and verify the response successfully | 0.759s | PASSED |
| 3 | slurperR = new groovy.json.JsonSlurper() | 0.000s | PASSED |
| 4 | resultR = slurperR.parseText(responseR.getResponseText()) | 0.000s | PASSED |
| 5 | println("*****") | 0.000s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM VALIDATEOTP TEST CASE = " + resultR) | 0.000s | PASSED |
| 7 | println("*****") | 0.000s | PASSED |

RoleAccessUser_Test

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 8 | valueR = roleId | 0.000s | PASSED |
| 9 | loGID = valueR | 0.000s | PASSED |
| 10 | println("*****") | 0.000s | PASSED |
| 11 | println("THE USER LOG ID IS " + loGID) | 0.000s | PASSED |
| 12 | println("*****") | 0.072s | PASSED |

sIncidentHistory_Test

Information

| | | | |
|-------------|----------------------------------|-----|---------------------|
| ID | Test Cases/sIncidentHistory_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:24 | End | 2019-09-26 15:19:26 |
| Elapsed | 1.565s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequestAndVerify(findTestObject("sIncident History", ["uniqueToken":uniquetoken, "userId":userid])) Send request and verify the response successfully | 1.358s | PASSED |

tMergeContactDetail_Test

Information

| | | | |
|-------------|-------------------------------------|-----|---------------------|
| ID | Test Cases/tMergeContactDetail_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:26 | End | 2019-09-26 15:19:27 |
| Elapsed | 1.200s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequest(findTestObject("tMerge Contact Detail", ["countryID":countryID, "uniqueToken":uniquetoken, "userId":userid])) Send request successfully | 0.219s | PASSED |
| 2 | responset = sendRequestAndVerify(findTestObject("tMerge Contact Detail", ["countryID":countryID, "uniqueToken":uniquetoken, "userId":userid])) Send request and verify the response successfully | 0.665s | PASSED |
| 3 | slurpert = new groovy.json.JsonSlurper() | 0.000s | PASSED |
| 4 | resultt = slurpert.parseText(responset.getResponseText()) | 0.000s | PASSED |
| 5 | println("*****") | 0.000s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM MERGE CONTACT DETAILS TEST CASE = " + resultt) | 0.000s | PASSED |

tMergeContactDetail_Test

| # | Description | Elapsed | Status |
|---|------------------|---------|--------|
| 7 | println("*****") | 0.000s | PASSED |

vInsertIncident_Test

Information

| | | | |
|-------------|---------------------------------|-----|---------------------|
| ID | Test Cases/vInsertIncident_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:27 | End | 2019-09-26 15:19:28 |
| Elapsed | 1.205s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | incidentTID = toUpperCase().replace("-", "") | 0.010s | PASSED |
| 2 | println(incidentTID) | 0.000s | PASSED |
| 3 | now = new java.util.Date() | 0.000s | PASSED |
| 4 | incidentTDate = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 5 | println(incidentTDate) | 0.000s | PASSED |
| 6 | sendRequest(findTestObject("uInsert Incident", [{"userId":userid, "uniqueToken":uniquetoken, "incidentID":incidentTID, "incidentDate":incidentTDate, "checkInLocality":checkInLocality, ... }])) Send request successfully | 0.182s | PASSED |

vInsertIncident_Test

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 7 | <pre>responsev = sendRequestAndVerify(findTestObject("uInsert Incident", ["userId":userid, "uniqueToken":uniquetoken, "incidentID":incidentTID, "incidentDate":incidentTDate, "checkInLocality":checkInLocality, ...]))</pre> Send request and verify the response successfully | 0.599s | PASSED |
| 8 | <pre>slurperv = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 9 | <pre>resultv = slurperv.parseText(responsev.getResponseText())</pre> | 0.000s | PASSED |
| 10 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 11 | <pre>println("THIS IS THE RESPONSE FROM ISERT INCIDENT TEST CASE = " + resultv)</pre> | 0.000s | PASSED |
| 12 | <pre>println("*****")</pre> | 0.000s | PASSED |

xDashboard_Test

Information

| | | | |
|-------------|----------------------------|-----|---------------------|
| ID | Test Cases/xDashboard_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:28 | End | 2019-09-26 15:19:30 |
| Elapsed | 1.788s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | now = new java.util.Date() | 0.669s | PASSED |
| 2 | fromDatE = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 3 | println(fromDatE) | 0.000s | PASSED |
| 4 | toDatE = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 5 | println(toDatE) | 0.000s | PASSED |
| 6 | sendRequest(findTestObject("xDashboard", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE])) Send request successfully | 0.278s | PASSED |

xDashboard_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 7 | <pre>responsex = sendRequestAndVerify(findTestObject("xDashboard", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE]))</pre> Send request and verify the response successfully | 0.587s | PASSED |
| 8 | <pre>slurperx = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 9 | <pre>resultx = slurperx.parseText(responsex.getResponseText())</pre> | 0.000s | PASSED |
| 10 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 11 | <pre>println("THIS IS THE RESPONSE FROM DASBOARD TEST CASE = " + resultx)</pre> | 0.010s | PASSED |
| 12 | <pre>println("*****")</pre> | 0.000s | PASSED |

wDiseaseMaster_Test

Information

| | | | |
|-------------|--------------------------------|-----|---------------------|
| ID | Test Cases/wDiseaseMaster_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:30 | End | 2019-09-26 15:19:32 |
| Elapsed | 1.772s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | sendRequest(findTestObject("wDisease Master", ["uniqueToken":uniquetoken, "userId":userid, "versionTableId":versionTableId])) Send request successfully | 0.356s | PASSED |
| 2 | responsew = sendRequestAndVerify(findTestObject("wDisease Master", ["uniqueToken":uniquetoken, "userId":userid, "versionTableId":versionTableId])) Send request and verify the response successfully | 0.648s | PASSED |
| 3 | slurperw = new groovy.json.JsonSlurper() | 0.000s | PASSED |
| 4 | resultw = slurperw.parseText(responsew.getResponseText()) | 0.001s | PASSED |
| 5 | println("*****") | 0.000s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM DISEASE MASTER TEST CASE = " + resultw) | 0.000s | PASSED |

wDiseaseMaster_Test

| # | Description | Elapsed | Status |
|---|------------------|---------|--------|
| 7 | println("*****") | 0.001s | PASSED |

zMobileAppVersion_Test

Information

| | | | |
|-------------|-----------------------------------|-----|---------------------|
| ID | Test Cases/zMobileAppVersion_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:32 | End | 2019-09-26 15:19:34 |
| Elapsed | 2.505s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequest(findTestObject("y1Mobile App Version", ["uniqueToken":uniquetoken, "oparatingSystem":oparatingSystem, "versionNumber":versionNumber])) Send request successfully | 0.177s | PASSED |
| 2 | responsez = sendRequestAndVerify(findTestObject("y1Mobile App Version", ["uniqueToken":uniquetoken, "oparatingSystem":oparatingSystem, "versionNumber":versionNumber])) Send request and verify the response successfully | 0.556s | PASSED |
| 3 | slurperz = new groovy.json.JsonSlurper() | 0.000s | PASSED |
| 4 | resultz = slurperz.parseText(responsez.getResponseText()) | 0.000s | PASSED |
| 5 | println("*****") | 0.000s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM MOBILE APP VERSION TEST CASE = " + resultz) | 0.000s | PASSED |

zMobileAppVersion_Test

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 7 | <code>println("*****")</code> | 0.000s | PASSED |
| 8 | <code>CtlID = toString().toUpperCase()</code> | 0.000s | PASSED |
| 9 | <code>f = new java.io.File(D:\SmartDoxApiResponse\CtlID.txt)</code> | 0.765s | PASSED |
| 10 | <code>f.write(CtlID)</code> | 0.020s | PASSED |
| 11 | <code>now = new java.util.Date()</code> | 0.000s | PASSED |
| 12 | <code>CreatedDate = now.format("dd-MM-YYYY HH:mm:ss +5:30")</code> | 0.000s | PASSED |
| 13 | <code>f = new java.io.File(D:\SmartDoxApiResponse\CreatedDate.txt)</code> | 0.000s | PASSED |
| 14 | <code>f.write(CreatedDate)</code> | 0.000s | PASSED |
| 15 | <code>now1 = new java.util.Date()</code> | 0.000s | PASSED |
| 16 | <code>SubmittedDate = now1.format("dd-MM-YYYY HH:mm:ss +5:30")</code> | 0.000s | PASSED |

zMobileAppVersion_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 17 | f = new java.io.File(D:\SmartDoxApiResponse\SubmittedDate.txt) | 0.000s | PASSED |
| 18 | f.write(SubmittedDate) | 0.010s | PASSED |

zActivityCategory_Test

Information

| | | | |
|-------------|-----------------------------------|-----|---------------------|
| ID | Test Cases/zActivityCategory_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:34 | End | 2019-09-26 15:19:35 |
| Elapsed | 1.215s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | sendRequest(findTestObject("y2Activity Category", ["versiontableID":versiontableID, "userId":userid, "uniqueToken":uniquetoken])) Send request successfully | 0.274s | PASSED |
| 2 | responsez1 = sendRequestAndVerify(findTestObject("y2Activity Category", ["versiontableID":versiontableID, "userId":userid, "uniqueToken":uniquetoken])) Send request and verify the response successfully | 0.661s | PASSED |
| 3 | slurperz1 = new groovy.json.JsonSlurper() | 0.000s | PASSED |
| 4 | resultz1 = slurperz1.parseText(responsez1.getResponseText()) | 0.000s | PASSED |
| 5 | println("*****") | 0.000s | PASSED |
| 6 | println("THIS IS THE RESPONSE FROM ACTIVITY CATEGORY TEST CASE = " + resultz1) | 0.011s | PASSED |

zActivityCategory_Test

| # | Description | Elapsed | Status |
|---|------------------|---------|--------|
| 7 | println("*****") | 0.000s | PASSED |

z1SupervisorTeamStatus_Test

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/z1SupervisorTeamStatus_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:35 | End | 2019-09-26 15:19:38 |
| Elapsed | 2.594s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | dte = new java.util.Date() | 0.335s | PASSED |
| 2 | fromDatE = dte.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 3 | println(fromDatE) | 0.000s | PASSED |
| 4 | bte = new java.util.Date() | 0.000s | PASSED |
| 5 | toDatE = bte.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 6 | println(toDatE) | 0.000s | PASSED |
| 7 | sendRequest(findTestObject("ySupervisor Team Status", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE])) | 0.217s | PASSED |

Send request successfully

z1SupervisorTeamStatus_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 8 | <pre>responsez1 = sendRequestAndVerify(findTestObject("ySupervisor Team Status", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE]))</pre> Send request and verify the response successfully | 0.997s | PASSED |
| 9 | <pre>slurperz1 = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 10 | <pre>resultz1 = slurperz1.parseText(responsez1.getResponseText())</pre> | 0.000s | PASSED |
| 11 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 12 | <pre>println("THIS IS THE RESPONSE FROM SUPERVISOR TEAM STATUS TEST CASE = " + resultz1)</pre> | 0.000s | PASSED |
| 13 | <pre>println("*****")</pre> | 0.000s | PASSED |

z2Supervisor Users

Information

| | | | |
|-------------|-------------------------------|-----|---------------------|
| ID | Test Cases/z2Supervisor Users | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:38 | End | 2019-09-26 15:19:40 |
| Elapsed | 2.437s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | ddt = new java.util.Date() | 0.000s | PASSED |
| 2 | fromDatE = ddt.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.001s | PASSED |
| 3 | println(fromDatE) | 0.001s | PASSED |
| 4 | btd = new java.util.Date() | 0.000s | PASSED |
| 5 | toDatE = btd.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.002s | PASSED |
| 6 | println(toDatE) | 0.000s | PASSED |
| 7 | sendRequest(findTestObject("y3Supervisor Users", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE])) | 0.874s | PASSED |

Send request successfully

z2Supervisor Users

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 8 | <pre>responsez2 = sendRequestAndVerify(findTestObject("y3Supervisor Users", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE]))</pre> Send request and verify the response successfully | 0.628s | PASSED |
| 9 | <pre>slurperz2 = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 10 | <pre>resultz2 = slurperz2.parseText(responsez2.getResponseText())</pre> | 0.000s | PASSED |
| 11 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 12 | <pre>println("THIS IS THE RESPONSE FROM SUPERVISOR USERS TEST CASE = " + resultz2)</pre> | 0.000s | PASSED |
| 13 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 14 | <pre>ctlID = toString().toUpperCase()</pre> | 0.000s | PASSED |
| 15 | <pre>println(ctlID)</pre> | 0.000s | PASSED |
| 16 | <pre>f = new java.io.File(D:\SmartDoxApiResponse\ctlID.txt)</pre> | 0.000s | PASSED |

z2Supervisor Users

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 17 | f.write(ctID) | 0.000s | PASSED |
| 18 | now = new java.util.Date() | 0.568s | PASSED |
| 19 | createdDaTE = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 20 | f1 = new java.io.File(D:\SmartDoxApiResponse\CreatedDate.txt) | 0.000s | PASSED |
| 21 | f1.write(createdDaTE) | 0.000s | PASSED |
| 22 | now1 = new java.util.Date() | 0.000s | PASSED |
| 23 | submittedDaTE = now1.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 24 | f2 = new java.io.File(D:\SmartDoxApiResponse\SubmittedDate.txt) | 0.000s | PASSED |
| 25 | f2.write(submittedDaTE) | 0.021s | PASSED |

x1TargetBreakDown_Test

Information

| | | | |
|-------------|-----------------------------------|-----|---------------------|
| ID | Test Cases/x1TargetBreakDown_Test | | |
| Description | | | |
| Tag | | | |
| Start | 2019-09-26 15:19:40 | End | 2019-09-26 15:19:43 |
| Elapsed | 2.551s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | now = new java.util.Date() | 0.662s | PASSED |
| 2 | fromDatE = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 3 | println(fromDatE) | 0.000s | PASSED |
| 4 | toDatE = now.format("dd-MM-YYYY HH:mm:ss +5:30") | 0.000s | PASSED |
| 5 | println(toDatE) | 0.000s | PASSED |
| 6 | sendRequest(findTestObject("x1Target Break Down", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE])) Send request successfully | 0.280s | PASSED |

x1TargetBreakDown_Test

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 7 | <pre>responsex1 = sendRequestAndVerify(findTestObject("x1Target Break Down", ["uniqueToken":uniquetoken, "userId":userid, "fromDatE":fromDatE, "toDatE":toDatE]))</pre> <p>Send request and verify the response successfully</p> | 0.657s | PASSED |
| 8 | <pre>slurperx1 = new groovy.json.JsonSlurper()</pre> | 0.000s | PASSED |
| 9 | <pre>resultx1 = slurperx1.parseText(responsex1.getResponseText())</pre> | 0.000s | PASSED |
| 10 | <pre>println("*****")</pre> | 0.000s | PASSED |
| 11 | <pre>println("THIS IS THE RESPONSE FROM TARGET BREAKDOWN TEST CASE = " + resultx1)</pre> | 0.000s | PASSED |
| 12 | <pre>println("*****")</pre> | 0.738s | PASSED |