

TS_Order_Service

Execution Environment

| | |
|-----------------|----------------------|
| Host name | PC - LAPTOP-UBH8UTE5 |
| Local OS | Windows 11 64bit |
| Katalon version | 10.4.2.0 |
| Browser | |
| Device name | |

Summary

| | | | |
|-------------|-------------------------------------|------------|---------------------|
| ID | Test Suites/TS_API/TS_Order_Service | | |
| Description | | | |
| Total | 5 | | |
| Passed | 5 | Failed | 0 |
| Error | 0 | Incomplete | 0 |
| Skipped | 0 | | |
| Start | 30-11-2025 22:25:32 | End | 30-11-2025 22:25:36 |
| Elapsed | 3.496s | | |

| # | ID | Description | Status |
|---|---|-------------|--------|
| 1 | Test Cases/API/Order-Service/TC01_Get_All_Orders | | PASSED |
| 2 | Test Cases/API/Order-Service/TC02_Create_Order | | PASSED |
| 3 | Test Cases/API/Order-Service/TC03_Get_Order_By_ID | | PASSED |
| 4 | Test Cases/API/Order-Service/TC04_Update_Order_Status | | PASSED |
| 5 | Test Cases/API/Order-Service/TC05_Delete_Order | | PASSED |

TC01_Get_All_Orders

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/API/Order-Service/TC01_Get_All_Orders | | |
| Description | | | |
| Tag | | | |
| Start | 30-11-2025 22:25:33 | End | 30-11-2025 22:25:34 |
| Elapsed | 1.489s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|---|---------|--------|
| 1 | println("=" * 60) | 0.013s | PASSED |
| 2 | println("TEST: Get All Orders") | 0.003s | PASSED |
| 3 | println("URL: http://localhost:3001/orders") | 0.003s | PASSED |
| 4 | println("=" * 60) | 0.003s | PASSED |
| 5 | response = sendRequest(findTestObject("API/Order-Service/getAllOrders")) Send request successfully | 0.818s | PASSED |
| 6 | verifyResponseStatusCode(response, 200) Verify response status code successfully | 0.020s | PASSED |
| 7 | println(" Status Code: 200 OK") | 0.004s | PASSED |

TC01_Get_All_Orders

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 8 | responseTime = response.getElapsedTime() | 0.011s | PASSED |
| 9 | assert responseTime < 3000 | 0.002s | PASSED |
| 10 | println(Response Time: \$responseTimems) | 0.027s | PASSED |
| 11 | contentType = getHeaderFields().get("Content-Type")[0] | 0.008s | PASSED |
| 12 | assert contentType.contains("application/json") | 0.012s | PASSED |
| 13 | println(" Content-Type: application/json") | 0.003s | PASSED |
| 14 | jsonResponse = JsonSlurper().parseText(response.getResponseText()) | 0.027s | PASSED |
| 15 | assert jsonResponse instanceof java.util.Map : "Response không phi là object JSON" | 0.002s | PASSED |
| 16 | println(" Response is an object") | 0.003s | PASSED |
| 17 | assert data instanceof java.util.List : "Data is not an array" | 0.009s | PASSED |

TC01_Get_All_Orders

| # | Description | Elapsed | Status |
|----|------------------------------|---------|--------|
| 18 | println(" data is an array") | 0.003s | PASSED |
| 19 | orders = data | 0.004s | PASSED |
| 20 | if (orders.size() > 0) | 0.024s | PASSED |
| 21 | println("=" * 60) | 0.003s | PASSED |
| 22 | println("TEST PASSED") | 0.003s | PASSED |
| 23 | println("=" * 60) | 0.001s | PASSED |

TC02_Create_Order

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/API/Order-Service/TC02_Create_Order | | |
| Description | | | |
| Tag | | | |
| Start | 30-11-2025 22:25:35 | End | 30-11-2025 22:25:35 |
| Elapsed | 0.664s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|-----------------------------------|---------|--------|
| 1 | println("=" * 60) | 0.001s | PASSED |
| 2 | println("TEST: Create New Order") | 0.001s | PASSED |
| 3 | println("=" * 60) | 0.002s | PASSED |
| 4 | UserID = 999 | 0.001s | PASSED |
| 5 | UserName = "TestKatalon" | 0.002s | PASSED |
| 6 | contactNumber = "0000000000" | 0.001s | PASSED |
| 7 | tableNumber = 99 | 0.001s | PASSED |

TC02_Create_Order

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 8 | cart = [{"id":8, "name":"Miso", "Quantity":2, "price":30000}] | 0.003s | PASSED |
| 9 | println("Request Data:") | 0.002s | PASSED |
| 10 | println(- UserID: \$UserID) | 0.002s | PASSED |
| 11 | println(- UserName: \$UserName) | 0.002s | PASSED |
| 12 | println(- TableNumber: \$tableNumber) | 0.000s | PASSED |
| 13 | println(- Cart: \$cart) | 0.001s | PASSED |
| 14 | response = sendRequest(findTestObject("API/Order-Service/postOrder", [{"UserID":UserID, "UserName":UserName, "ContactNumber":contactNumber, "TableNumber":tableNumber, "Cart":cart}])) Send request successfully | 0.304s | PASSED |
| 15 | responseText = response.getResponseText() | 0.001s | PASSED |
| 16 | println("RESPONSE:") | 0.001s | PASSED |
| 17 | println(responseText) | 0.001s | PASSED |

TC02_Create_Order

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 18 | assert responseText != null && responseText.length() > 0 : "Response body is empty" | 0.001s | PASSED |
| 19 | parsed = JsonSlurper().parseText(responseText) | 0.001s | PASSED |
| 20 | assert parsed instanceof java.util.Map : " Response is not a JSON Object" | 0.001s | PASSED |
| 21 | assert success == true : " API returned success = false" | 0.001s | PASSED |
| 22 | println(" Success = true") | 0.002s | PASSED |
| 23 | orderData = data | 0.001s | PASSED |
| 24 | assert orderData != null : " Data field is missing" | 0.001s | PASSED |
| 25 | assert OrderID != null : " OrderID is missing" | 0.000s | PASSED |
| 26 | assert UserID.toInteger() == UserID : " UserID mismatch" | 0.004s | PASSED |
| 27 | assert UserName == UserName : " UserName mismatch" | 0.001s | PASSED |

TC02_Create_Order

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 28 | assert ContactNumber == contactNumber : " ContactNumber mismatch" | 0.002s | PASSED |
| 29 | assert TableNumber.toInteger() == tableNumber : " TableNumber mismatch" | 0.001s | PASSED |
| 30 | assert OrderStatus != null : " OrderStatus missing" | 0.000s | PASSED |
| 31 | assert TotalPrice != null : " TotalPrice missing" | 0.001s | PASSED |
| 32 | println(" Order Info Valid") | 0.002s | PASSED |
| 33 | println(- OrderID: \$orderData.OrderID) | 0.000s | PASSED |
| 34 | println(- Status: \$orderData.OrderStatus) | 0.002s | PASSED |
| 35 | println(- Total Price: \$orderData.TotalPrice) | 0.001s | PASSED |
| 36 | assert Items != null : "Items is missing" | 0.002s | PASSED |
| 37 | assert Items.size() == cart.size() : "Items size mismatch" | 0.001s | PASSED |

TC02_Create_Order

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 38 | Items.forEachWithIndex({ java.lang.Object item, java.lang.Object index -> ... }) | 0.016s | PASSED |
| 39 | println("=" * 60) | 0.003s | PASSED |
| 40 | println("TEST PASSED") | 0.001s | PASSED |
| 41 | println("=" * 60) | 0.000s | PASSED |

TC03_Get_Order_By_ID

Information

| | | | |
|-------------|---|-----|---------------------|
| ID | Test Cases/API/Order-Service/TC03_Get_Order_By_ID | | |
| Description | | | |
| Tag | | | |
| Start | 30-11-2025 22:25:35 | End | 30-11-2025 22:25:35 |
| Elapsed | 0.208s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | println("=" * 60) | 0.002s | PASSED |
| 2 | println("TEST: Get Order By ID") | 0.001s | PASSED |
| 3 | println("=" * 60) | 0.002s | PASSED |
| 4 | response = sendRequest(findTestObject("API/Order-Service/getOrderByOrderID")) Send request successfully | 0.060s | PASSED |
| 5 | verifyResponseStatusCode(response, 200) Verify response status code successfully | 0.006s | PASSED |
| 6 | println(" Status Code: 200 OK") | 0.002s | PASSED |
| 7 | jsonResponse = JsonSlurper().parseText(response.getResponseText()) | 0.000s | PASSED |

TC03_Get_Order_By_ID

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 8 | println("Order Details:") | 0.000s | PASSED |
| 9 | println(- ID: \$jsonResponse.id) | 0.026s | PASSED |
| 10 | println(- User ID: \$jsonResponse.UserId) | 0.002s | PASSED |
| 11 | println(- Status: \$jsonResponse.status) | 0.002s | PASSED |
| 12 | println(- Total Amount: \$jsonResponse.totalAmount) | 0.000s | PASSED |
| 13 | println(- Table Number: \$jsonResponse.tableNumber) | 0.003s | PASSED |
| 14 | println(- Order Type: \$jsonResponse.orderType) | 0.001s | PASSED |
| 15 | if (items) | 0.003s | PASSED |
| 16 | println(" " + "=" * 60) | 0.004s | PASSED |
| 17 | println("TEST PASSED") | 0.001s | PASSED |

TC03_Get_Order_By_ID

| # | Description | Elapsed | Status |
|----|-------------------|---------|--------|
| 18 | println("=" * 60) | 0.001s | PASSED |

TC04_Update_Order_Status

Information

| | | | |
|-------------|---|-----|---------------------|
| ID | Test Cases/API/Order-Service/TC04_Update_Order_Status | | |
| Description | | | |
| Tag | | | |
| Start | 30-11-2025 22:25:35 | End | 30-11-2025 22:25:36 |
| Elapsed | 0.160s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | println("=" * 60) | 0.002s | PASSED |
| 2 | println("TEST: Update Order Status") | 0.001s | PASSED |
| 3 | println("=" * 60) | 0.000s | PASSED |
| 4 | orderId = 1 | 0.000s | PASSED |
| 5 | newStatus = "preparing" | 0.000s | PASSED |
| 6 | statusFlow = ["pending", "preparing", "ready", "delivered", "cancelled"] | 0.002s | PASSED |
| 7 | assert statusFlow.contains(newStatus) | 0.001s | PASSED |

TC04_Update_Order_Status

| # | Description | Elapsed | Status |
|----|---|---------|--------|
| 8 | <pre>response = sendRequest(findTestObject("API/Order-Service/updateStatus", ["orderId":orderId, "status":newStatus]))</pre> Send request successfully | 0.044s | PASSED |
| 9 | <pre>verifyResponseStatusCode(response, 200)</pre> Verify response status code successfully | 0.004s | PASSED |
| 10 | <pre>responseText = response.getResponseText()</pre> | 0.001s | PASSED |
| 11 | <pre>println("RESPONSE BODY: " + responseText)</pre> | 0.002s | PASSED |
| 12 | <pre>jsonResponse = JsonSlurper().parseText(response.getResponseText())</pre> | 0.001s | PASSED |
| 13 | <pre>assert success == true</pre> | 0.002s | PASSED |
| 14 | <pre>data = data</pre> | 0.001s | PASSED |
| 15 | <pre>assert statusFlow.contains(OrderStatus)</pre> | 0.000s | PASSED |
| 16 | <pre>assert OrderID == orderId.toString()</pre> | 0.000s | PASSED |
| 17 | <pre>assert OrderStatus == newStatus</pre> | 0.000s | PASSED |

TC04_Update_Order_Status

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 18 | println(Order \$orderId updated successfully to status: \$data.OrderStatus) | 0.002s | PASSED |
| 19 | println(" + "=" * 60) | 0.001s | PASSED |
| 20 | println("TEST PASSED - All status transitions successful") | 0.001s | PASSED |
| 21 | println("=" * 60) | 0.001s | PASSED |

TC05_Delete_Order

Information

| | | | |
|-------------|--|-----|---------------------|
| ID | Test Cases/API/Order-Service/TC05_Delete_Order | | |
| Description | | | |
| Tag | | | |
| Start | 30-11-2025 22:25:36 | End | 30-11-2025 22:25:36 |
| Elapsed | 0.326s | | |
| Status | PASSED | | |

Details

| # | Description | Elapsed | Status |
|---|--|---------|--------|
| 1 | println("=" * 60) | 0.001s | PASSED |
| 2 | println("TEST: Delete Order") | 0.002s | PASSED |
| 3 | println("=" * 60) | 0.001s | PASSED |
| 4 | println("[STEP 1] Creating test order for deletion...") | 0.001s | PASSED |
| 5 | createResponse = sendRequest(findTestObject("API/Order-Service/postOrder")) Send request successfully | 0.031s | PASSED |
| 6 | createJson = JsonSlurper().parseText(createResponse.getResponseText()) | 0.000s | PASSED |
| 7 | assert success == true : "Failed to create test order" | 0.002s | PASSED |

TC05_Delete_Order

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 8 | testOrderId = OrderID | 0.000s | PASSED |
| 9 | println(Test order created - ID: \$testOrderId) | 0.003s | PASSED |
| 10 | println(" [STEP 2] Verifying order exists before deletion...") | 0.001s | PASSED |
| 11 | getBeforeResponse = sendRequest(findTestObject("API/Order-Service/getOrderByOrderID", ["orderId":testOrderId])) Send request successfully | 0.021s | PASSED |
| 12 | verifyResponseStatusCode(getBeforeResponse, 200) Verify response status code successfully | 0.002s | PASSED |
| 13 | println(" Order exists and can be retrieved") | 0.002s | PASSED |
| 14 | println([STEP 3] Deleting order \$testOrderId...) | 0.001s | PASSED |
| 15 | deleteResponse = sendRequest(findTestObject("API/Order-Service/deleteOrder", ["orderId":testOrderId])) Send request successfully | 0.029s | PASSED |
| 16 | deleteStatusCode = deleteResponse.getStatusCode() | 0.001s | PASSED |
| 17 | assert deleteStatusCode == 200 deleteStatusCode == 204 | 0.001s | PASSED |

TC05_Delete_Order

| # | Description | Elapsed | Status |
|----|--|---------|--------|
| 18 | println(Status Code: \$deleteStatusCode) | 0.002s | PASSED |
| 19 | if (deleteStatusCode == 200 && getResponseText().trim()) | 0.010s | PASSED |
| 20 | println(" Order deleted successfully") | 0.001s | PASSED |
| 21 | println(" [STEP 4] Verifying order no longer exists...") | 0.000s | PASSED |
| 22 | getAfterResponse = sendRequest(findTestObject("API/Order-Service/getOrderByOrderID", ["orderId":testOrderId])) Send request successfully | 0.026s | PASSED |
| 23 | afterStatusCode = getAfterResponse.getStatusCode() | 0.000s | PASSED |
| 24 | if (afterStatusCode == 404) | 0.000s | PASSED |
| 25 | println(" [NEGATIVE TEST] Attempting to delete already deleted order...") | 0.001s | PASSED |
| 26 | deleteAgainResponse = sendRequest(findTestObject("API/Order-Service/deleteOrder", ["orderId":testOrderId])) Send request successfully | 0.027s | PASSED |
| 27 | deleteAgainStatus = deleteAgainResponse.getStatusCode() | 0.001s | PASSED |

TC05_Delete_Order

| # | Description | Elapsed | Status |
|----|---------------------------------------|---------|--------|
| 28 | if (deleteAgainStatus == 404) | 0.001s | PASSED |
| 29 | println(" + " * 60) | 0.002s | PASSED |
| 30 | println("TEST PASSED - DELETE ORDER") | 0.001s | PASSED |
| 31 | println("=" * 60) | 0.001s | PASSED |