# Cucumber Report

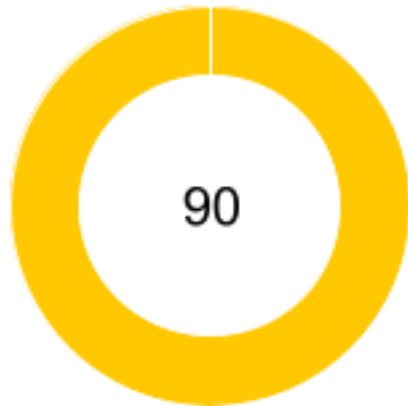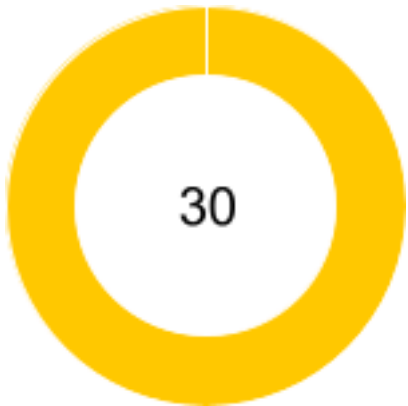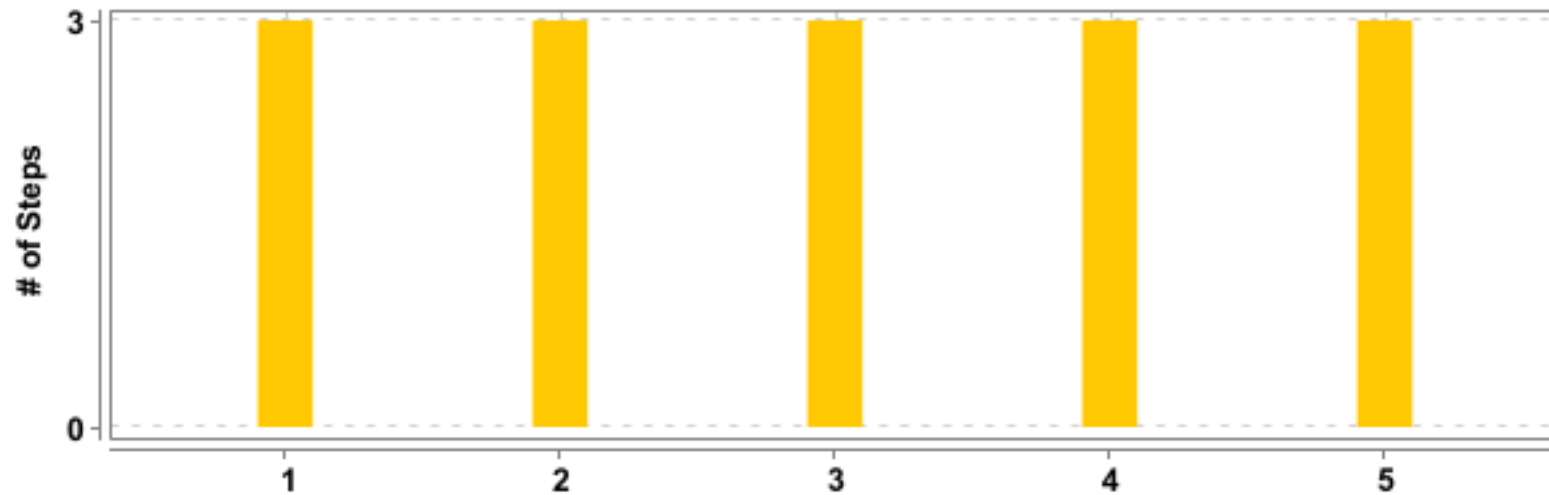| Start : Jul 26, 3:02:26.134 AM | End : Jul 26, 3:02:27.228 AM | Duration : 1.094 s |
|:---:|:---:|:---:|
| Features | Scenarios | Steps |
| PASSED - 0<br>FAILED - 0<br>SKIPPED - 1 | PASSED - 0<br>FAILED - 0<br>SKIPPED - 30 | PASSED - 0<br>FAILED - 0<br>SKIPPED - 90 |
| 1 | 30 | 90 |

| Feature | | Scenario | | | | Step | | | |
|---|---|---|---|---|---|---|---|---|---|
| Name | Duration | T | P | F | S | T | P | F | S |
| User Module | 1.094 s | 30 | 0 | 0 | 30 | 90 | 0 | 0 | 90 |

| Tag | Scenario | | | | Feature | | | |
|---|---|---|---|---|---|---|---|---|
| Name | T | P | F | S | T | P | F | S |
| @GETRequest_AllUsers | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @GETRequest_Userby_Role | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @GETRequest_GetAllStaff | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @POSTRequest_UserWith_MissingMandatoryFields | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @POSTRequest_UserWith_AdminRole | 2 | 0 | 0 | 2 | 1 | 0 | 0 | 1 |
| @POSTRequest_UserWith_StaffRole | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @POSTRequest_UserWith_StudentRole | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @GETRequest_Userby_UserID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserByMissingFields | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserById | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @07PUTRequest_UpdateUserRoleStatusById_MissingFields | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserRoleStatusById | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @POSTRequest_UserWith_ExistingPhoneNumber | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @DeleteRequest_DeleteAdminUser_ByID | 2 | 0 | 0 | 2 | 1 | 0 | 0 | 1 |
| @GETRequest_Userby_InvalidUserID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserByInvalidId | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserRoleStatusById_invalidid | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

| Tag | Scenario | | | | Feature | | | |
|---|---|---|---|---|---|---|---|---|
| **Name** | **T** | **P** | **F** | **S** | **T** | **P** | **F** | **S** |
| @DeleteRequest_DeleteUser_ByInvalidID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @DeleteRequest_DeleteStaffUser_ByID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @DeleteRequest_DeleteStudentUser_ByID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PostProgram | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PostBatch | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_UpdateUserRoleStatusByDiffRoleID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_AssignUserToProgBatchByMissingfield | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_AssignUserToProgBatchById | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @PUTRequest_AssignUserToProgBatchByInvalidID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @DeleteRequest_DeleteBatch_ByID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| @DeleteRequest_DeleteProgram_ByID | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

| # | Feature Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|
| 1 | User Module | 30 | 0 | 0 | 30 | 1.094 s |

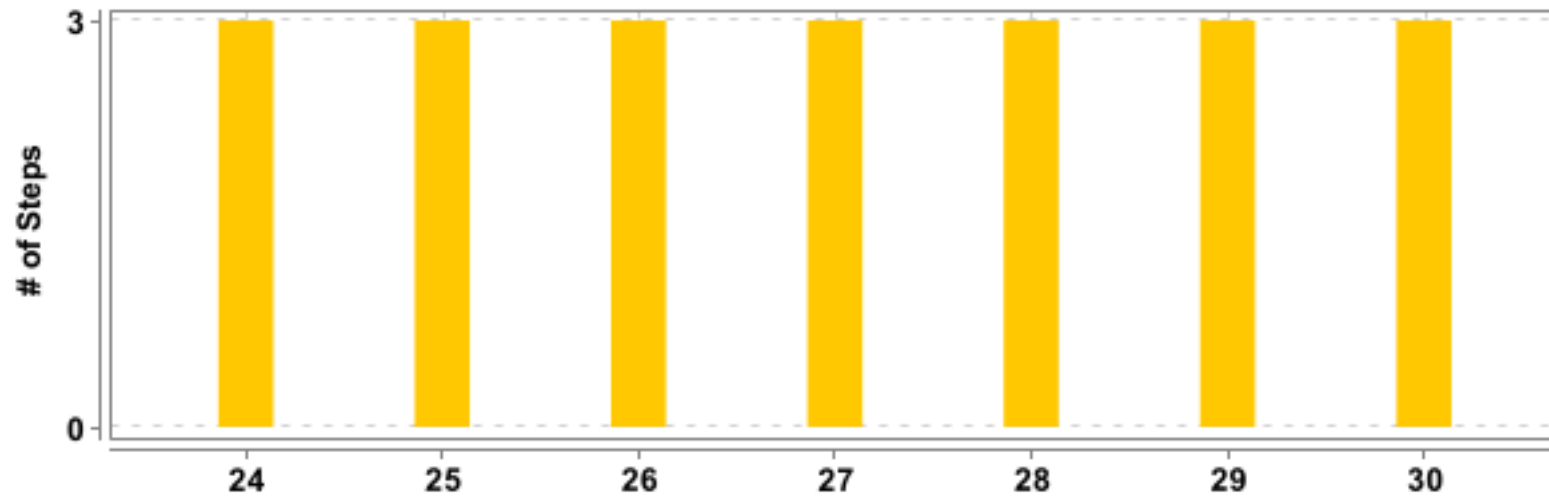| # | Feature Name | Scenario Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|---|
| 1 | *User Module* | *Check if user able to retrieve all user with valid LMS API* | 3 | 0 | 0 | 3 | 0.233 s |
| 2 | *User Module* | *Check if user able to retrieve all user with valid LMS API* | 3 | 0 | 0 | 3 | 0.018 s |
| 3 | *User Module* | *Check if user able to retrieve a user with valid LMS API* | 3 | 0 | 0 | 3 | 0.017 s |
| 4 | *User Module* | *Check if user able to create a User with valid endpoint and MissingMandatoryFields (non existing values)* | 3 | 0 | 0 | 3 | 0.016 s |
| 5 | *User Module* | *Check if user able to create a User with valid endpoint and request body (non existing values)* | 3 | 0 | 0 | 3 | 0.016 s |

| # | Feature Name | Scenario Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|---|
| 6 | User Module | Check if user able to create a User with valid endpoint and request body (non existing values) | 3 | 0 | 0 | 3 | 0.016 s |
| 7 | User Module | Check if user able to create a User with valid endpoint and request body (non existing values) | 3 | 0 | 0 | 3 | 0.015 s |
| 8 | User Module | Check if user able to retrieve a user with valid User ID | 3 | 0 | 0 | 3 | 0.012 s |
| 9 | User Module | Check if user able to update a user with missingFields and valid User ID | 3 | 0 | 0 | 3 | 0.012 s |
| 10 | User Module | Check if user able to update a user with valid User ID and request body | 3 | 0 | 0 | 3 | 0.012 s |
| 11 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.013 s |

| # | Feature Name | Scenario Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|---|
| 12 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.014 s |
| 13 | User Module | Check if user able to create a User with valid endpoint and ExistingPhoneNumber(non existing values) | 3 | 0 | 0 | 3 | 0.015 s |
| 14 | User Module | Check if user able to delete a user with valid User Id | 3 | 0 | 0 | 3 | 0.013 s |
| 15 | User Module | Check if user is not able to retrieve user info with Invalid User ID | 3 | 0 | 0 | 3 | 0.012 s |
| 16 | User Module | Check if user able to update a user with Invalid User ID and request body | 3 | 0 | 0 | 3 | 0.014 s |
| 17 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.013 s |

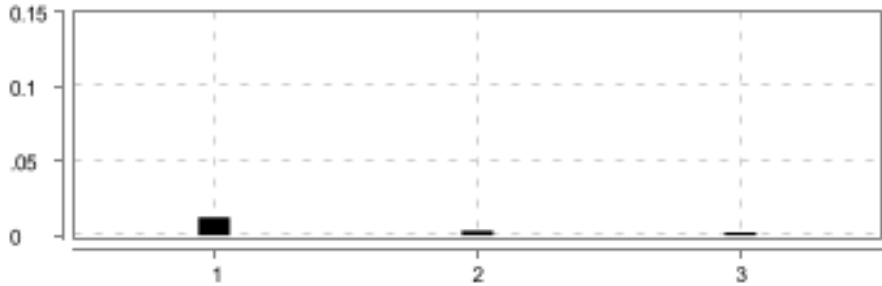| # | Feature Name | Scenario Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|---|
| 18 | User Module | Check if user able to delete a user with Invalid User Id | 3 | 0 | 0 | 3 | 0.011 s |
| 19 | User Module | Check if user able to delete a user with valid User Id | 3 | 0 | 0 | 3 | 0.012 s |
| 20 | User Module | Check if user able to delete a user with valid User Id | 3 | 0 | 0 | 3 | 0.011 s |
| 21 | User Module | Check if user able to create a program with valid endpoint and request body (non existing values) | 3 | 0 | 0 | 3 | 0.010 s |
| 22 | User Module | Check if user able to create a Batch with valid endpoint and request body (non existing values) | 3 | 0 | 0 | 3 | 0.013 s |
| 23 | User Module | Check if user able to create a User with valid endpoint and request body (non existing values) | 3 | 0 | 0 | 3 | 0.014 s |

| # | Feature Name | Scenario Name | T | P | F | S | Duration |
|---|---|---|---|---|---|---|---|
| 24 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.013 s |
| 25 | User Module | Check if user able to assign a user with program and batch with invalid UserId | 3 | 0 | 0 | 3 | 0.009 s |
| 26 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.010 s |
| 27 | User Module | Check if user able to update a user with valid User Id and request body | 3 | 0 | 0 | 3 | 0.010 s |
| 28 | User Module | Check if user able to delete a user with valid User Id | 3 | 0 | 0 | 3 | 0.011 s |
| 29 | User Module | Check if user able to delete a user with valid Batch Id | 3 | 0 | 0 | 3 | 0.012 s |
| 30 | User Module | Check if user able to delete a user with valid Program Id | 3 | 0 | 0 | 3 | 0.010 s |

## User Module

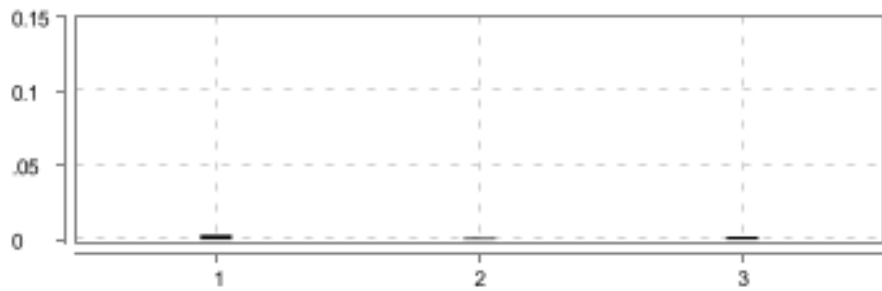| SKIPPED | DURATION - 1.094 s | Scenarios<br>Total - 30<br>Pass - 0<br>Fail - 0<br>Skip - 30 | 30 | Steps<br>Total - 90<br>Pass - 0<br>Fail - 0<br>Skip - 90 | 90 |
|---------|--------------------|-----|------|-----|----|
| / 3:02:26.134 AM // 3:02:27.228 AM / | | | | | |

### Check if user able to retrieve all user with valid LMS API

| SKIPPED | DURATION - 0.233 s | | Steps<br>Total - 3<br>Pass - 0<br>Fail - 0<br>Skip - 3 | 3 |
|---------|--------------------|--|-----|---|
| / 3:02:26.138 AM // 3:02:26.371 AM / | | | | |
| *User Module* | | | | |
| @GETRequest_AllUsers | | | | |

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.012 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS Request GetAllUser | SKIPPED | 0.003 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body for viewing all Users | SKIPPED | 0.002 s |
|   | Step undefined | | |
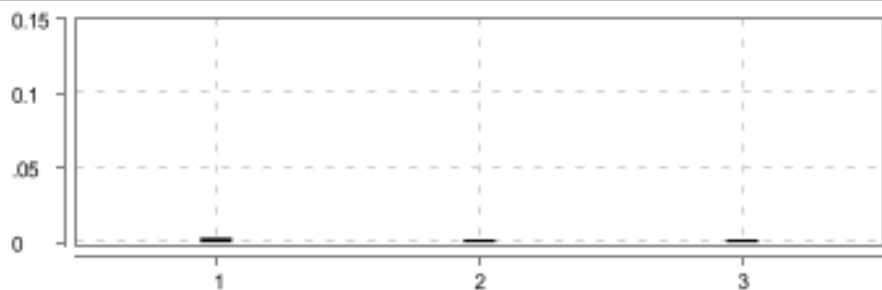
### Check if user able to retrieve all user with valid LMS API

| SKIPPED | DURATION - 0.018 s |
|---------|--------------------|

*/ 3:02:26.497 AM // 3:02:26.515 AM /*

*User Module*

@GETRequest_Userby_Role
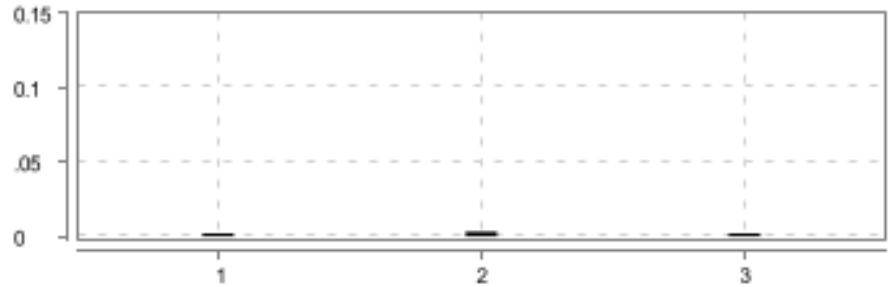
**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.003 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS GET Request to view user by Roles | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body for viewing an User by Role | SKIPPED | 0.002 s |
|   | Step undefined | | |

### Check if user able to retrieve a user with valid LMS API

| SKIPPED | DURATION - 0.017 s |
|---------|--------------------|

*/ 3:02:26.531 AM // 3:02:26.548 AM /*

*User Module*

@GETRequest_GetAllStaff
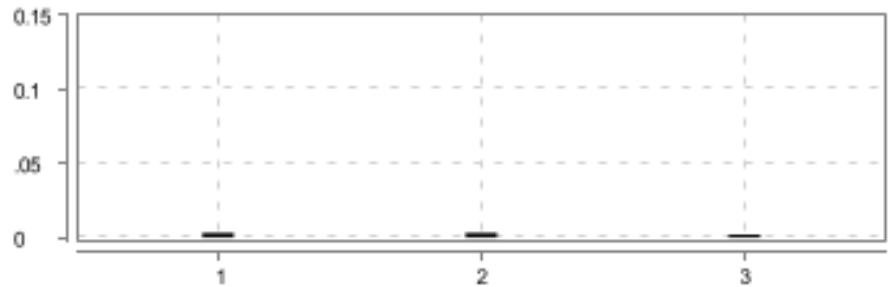
**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.003 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS GET Request to view all Staff | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body to view all Staff | SKIPPED | 0.002 s |
|   | Step undefined | | |

## Check if user able to create a User with valid endpoint and MissingMandatoryFields (non existing values)

| SKIPPED | DURATION - 0.016 s |
|---|---|
| / 3:02:26.571 AM // 3:02:26.587 AM / | |
| *User Module* | |
| @POSTRequest_UserWith_MissingMandatoryFields | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.002 s |
| 2 | When User sends HTTPS POST Request for new User with Missing Mandatory Fields<br>Step undefined | SKIPPED | 0.003 s |
| 3 | Then User receives status code 400 with response body for User with Missing Mandatory Fields<br>Step undefined | SKIPPED | 0.002 s |

## Check if user able to create a User with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.016 s |
|---|---|
| / 3:02:26.604 AM // 3:02:26.620 AM / | |
| *User Module* | |
| @POSTRequest_UserWith_AdminRole | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.003 s |
| 2 | When User sends HTTPS POST Request for new User for Admin role<br>Step undefined | SKIPPED | 0.003 s |
| 3 | Then User receives status code 201 with response body for creating an User for Admin role | SKIPPED | 0.002 s |

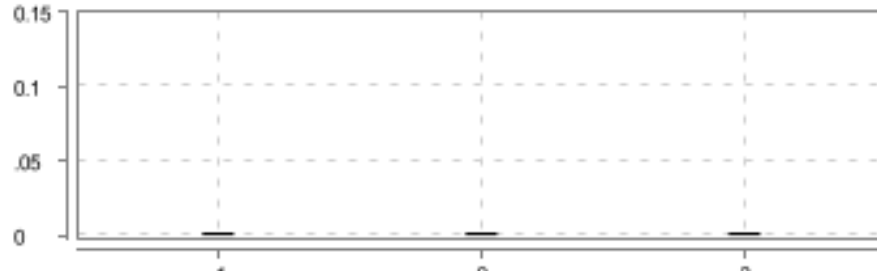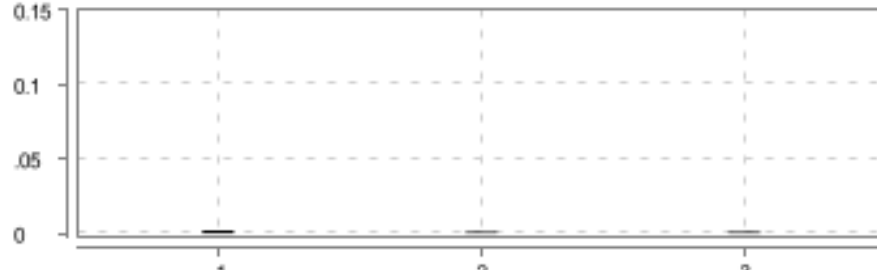| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
|   | Step undefined |  |  |

## Check if user able to create a User with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.016 s |  | Steps |
|---------|--------------------|----------|-------|
| / 3:02:26.637 AM // 3:02:26.653 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @POSTRequest_UserWith_StaffRole | | | |

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS POST Request for new User for Staff role | SKIPPED | 0.003 s |
|   | Step undefined | | |
| 3 | Then User receives status code 201 with response body for creating an User for Staff role | SKIPPED | 0.002 s |
|   | Step undefined | | |

## Check if user able to create a User with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.015 s |  | Steps |
|---------|--------------------|----------|-------|
| / 3:02:26.671 AM // 3:02:26.686 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @POSTRequest_UserWith_StudentRole | | | |

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
|   | Step undefined | | |

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 2 | When User sends HTTPS POST Request for new User for Student role | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 201 with response body for creating an User for Student role | SKIPPED | 0.002 s |
| | Step undefined | | |
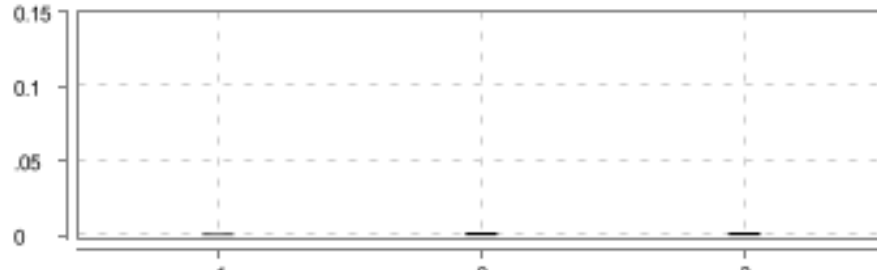
### Check if user able to retrieve a user with valid User ID
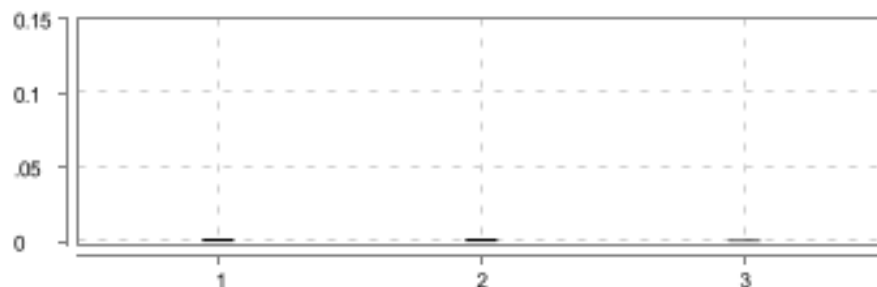
| SKIPPED | DURATION - 0.012 s | | | Steps | |
|---------|--------------------|---|---|-------|---|
| / 3:02:26.704 AM // 3:02:26.716 AM / | | | | Total - 3 | |
| | | | | Pass - 0 | |
| User Module | | | | Fail - 0 | |
| | | | | Skip - 3 | |
| @GETRequest_Userby_UserID | | | | | |

| # | Step / Hook Details | Status | Duration |
|---|---------------------|--------|----------|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
| | Step undefined | | |
| 2 | When User sends HTTPS GET Request to view user by userID | SKIPPED | 0.001 s |
| | Step undefined | | |
| 3 | Then User receives status code 200 with response body for viewing an User by ID | SKIPPED | 0.001 s |
| | Step undefined | | |

### Check if user able to update a user with missingFields and valid User ID

| SKIPPED | DURATION - 0.012 s | | | Steps | |
|---------|--------------------|---|---|-------|---|
| / 3:02:26.731 AM // 3:02:26.743 AM / | | | | Total - 3 | |
| | | | | Pass - 0 | |
| User Module | | | | Fail - 0 | |
| | | | | Skip - 3 | |
| @PUTRequest_UpdateUserByMissingFields | | | | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS PUT Request to update User by UserId with missingFields | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 3 | Then User receives status code 400 with response body for updating User with missingFields | SKIPPED | 0.002 s |
|   | Step undefined | | |
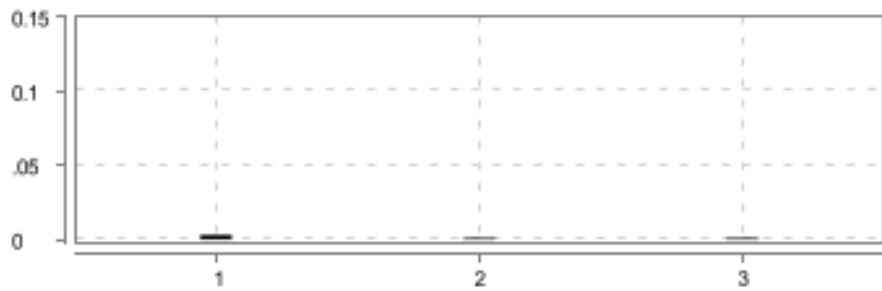
## Check if user able to update a user with valid User ID and request body

| SKIPPED | DURATION - 0.012 s |
|---|---|
| / 3:02:26.756 AM // 3:02:26.768 AM / | |
| User Module | |
| @PUTRequest_UpdateUserById | |



Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS PUT Request to update User by UserId | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body for updating User by ID | SKIPPED | 0.001 s |
|   | Step undefined | | |

## Check if user able to update a user with valid User Id and request body

| SKIPPED | DURATION - 0.013 s |
|---|---|
| / 3:02:26.779 AM // 3:02:26.792 AM / | |
| User Module | |
| @07PUTRequest_UpdateUserRoleStatusById_Missing-Fields | |
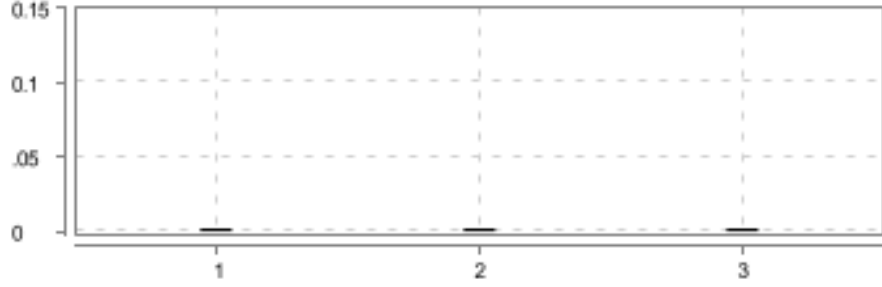


**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.003 s |
| 2 | When User sends HTTPS PUT Request to update User role status by User ID for missing fields<br>Step undefined | SKIPPED | 0.001 s |
| 3 | Then User receives status code 400 with response body for updating UserRoleStaus<br>Step undefined | SKIPPED | 0.001 s |

### Check if user able to update a user with valid User Id and request body

| SKIPPED | DURATION - 0.014 s |
|---|---|
| / 3:02:26.803 AM // 3:02:26.817 AM / | |
| User Module | |
| @PUTRequest_UpdateUserRoleStatusById | |
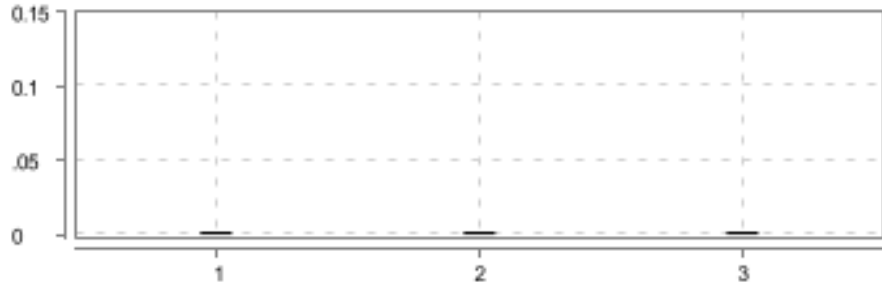


**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.002 s |
| 2 | When User sends HTTPS PUT Request to update User role status by User ID<br>Step undefined | SKIPPED | 0.003 s |
| 3 | Then User receives status code 200 with response body for updating UserRoleStaus<br>Step undefined | SKIPPED | 0.003 s |

### Check if user able to create a User with valid endpoint and ExistingPhoneNumber(non existing values)

| SKIPPED | DURATION - 0.015 s |
|---|---|
| / 3:02:26.829 AM // 3:02:26.844 AM / | |
| User Module | |
| @POSTRequest_UserWith_ExistingPhoneNumber | |

Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

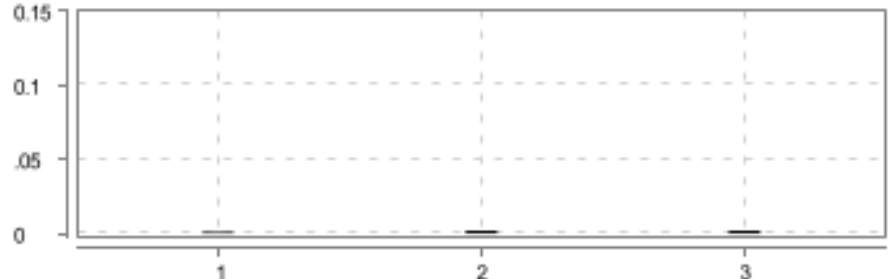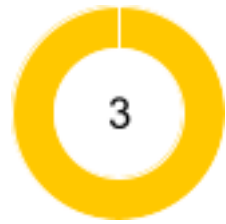| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
| | Step undefined | | |
| 2 | When User sends HTTPS POST Request for new User with Existing PhoneNumber | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 400 with response body for User with ExistingPhoneNumber | SKIPPED | 0.002 s |
| | Step undefined | | |

### Check if user able to delete a user with valid User Id

| SKIPPED | DURATION - 0.013 s |
|---|---|
| / 3:02:26.856 AM // 3:02:26.869 AM / | |
| User Module | |
| @DeleteRequest_DeleteAdminUser_ByID | |

Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

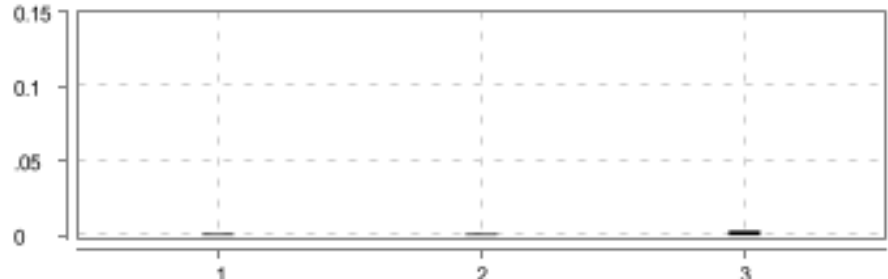| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
| | Step undefined | | |
| 2 | When User sends HTTPS DELETE Request to delete a user | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 200 with response body to delete User by ID | SKIPPED | 0.002 s |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| | Step undefined | | |

## Check if user is not able to retrieve user info with Invalid User ID

| SKIPPED | DURATION - 0.012 s | | Steps |
|---|---|---|---|
| / 3:02:26.882 AM // 3:02:26.894 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @GETRequest_Userby_InvalidUserID | | | |

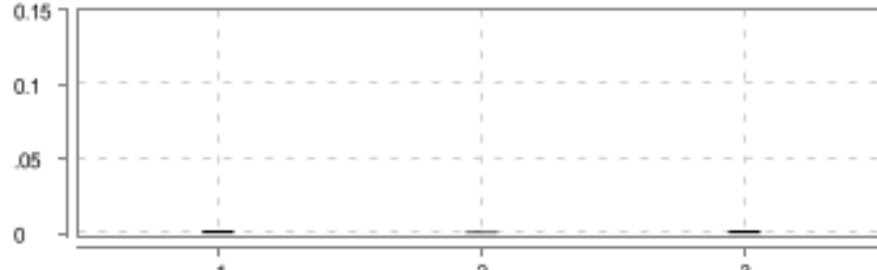| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
| | Step undefined | | |
| 2 | When User sends HTTPS GET Request to view user by invalid userID | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 404 Bad Request with response body with Invalid UserID | SKIPPED | 0.002 s |
| | Step undefined | | |

## Check if user able to update a user with Invalid User ID and request body

| SKIPPED | DURATION - 0.014 s | | Steps |
|---|---|---|---|
| / 3:02:26.906 AM // 3:02:26.920 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @PUTRequest_UpdateUserByInvalidId | | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
| | Step undefined | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 2 | When User sends HTTPS PUT Request to update User by Invalid userId | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 3 | Then User receives status code 404 with response body for updating User by InvalidID | SKIPPED | 0.003 s |
|   | Step undefined | | |
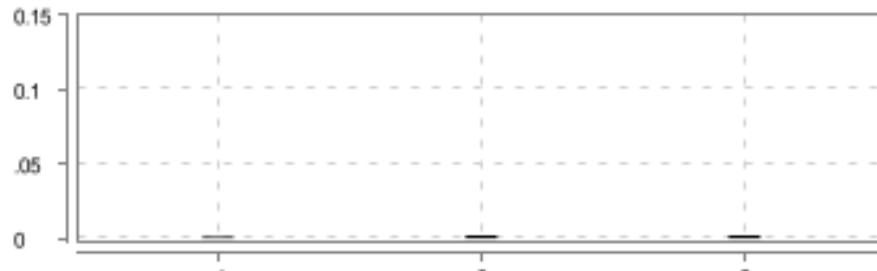
### Check if user able to update a user with valid User Id and request body

| SKIPPED | DURATION - 0.013 s | | Steps | |
|---|---|---|---|---|
| / 3:02:26.929 AM // 3:02:26.942 AM / | | | Total - 3 | |
| | | | Pass - 0 | |
| User Module | | | Fail - 0 | |
| | | | Skip - 3 | |
| @PUTRequest_UpdateUserRoleStatusById_invalidid | | | | |



| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS PUT Request to update User role status by User ID for missing fields | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 3 | Then User receives status code 400 with response body for updating UserRoleStaus | SKIPPED | 0.002 s |
|   | Step undefined | | |

### Check if user able to delete a user with Invalid User Id

| SKIPPED | DURATION - 0.011 s | | Steps | |
|---|---|---|---|---|
| / 3:02:26.953 AM // 3:02:26.964 AM / | | | Total - 3 | |
| | | | Pass - 0 | |
| User Module | | | Fail - 0 | |
| | | | Skip - 3 | |
| @DeleteRequest_DeleteUser_ByInvalidID | | | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
|  | Step undefined |  |  |
| 2 | When User sends HTTPS DELETE Request to delete a user with Invalid User Id | SKIPPED | 0.002 s |
|  | Step undefined |  |  |
| 3 | Then User receives status code 404 with response body to delete User by Invalid User Id | SKIPPED | 0.002 s |
|  | Step undefined |  |  |
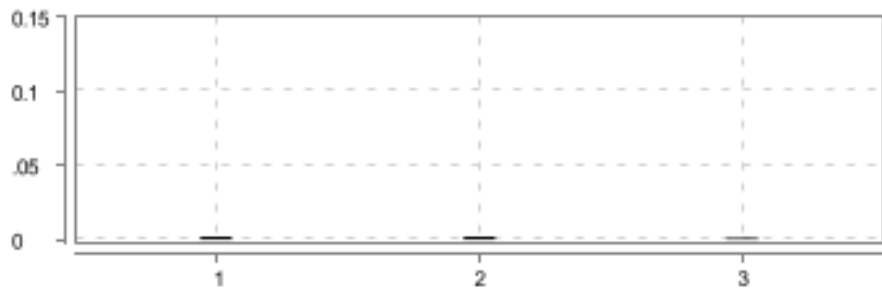
### Check if user able to delete a user with valid User Id

| SKIPPED | DURATION - 0.012 s | | Steps |
|---|---|---|---|
| / 3:02:26.975 AM // 3:02:26.987 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @DeleteRequest_DeleteStaffUser_ByID | | | |
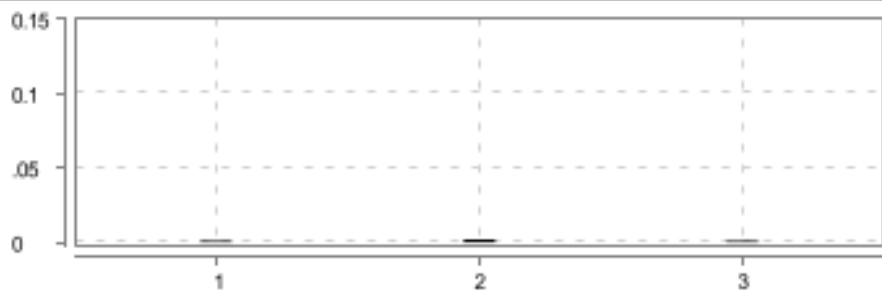
**3**

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
|  | Step undefined |  |  |
| 2 | When User sends HTTPS DELETE Request to delete a Staff user | SKIPPED | 0.002 s |
|  | Step undefined |  |  |
| 3 | Then User receives status code 200 with response body to delete User by ID | SKIPPED | 0.002 s |
|  | Step undefined |  |  |

### Check if user able to delete a user with valid User Id

| SKIPPED | DURATION - 0.011 s | | Steps |
|---|---|---|---|
| / 3:02:26.998 AM // 3:02:27.009 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @DeleteRequest_DeleteStudentUser_ByID | | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
| | Step undefined | | |
| 2 | When User sends HTTPS DELETE Request to delete a Student user | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 200 with response body to delete User by ID | SKIPPED | 0.001 s |
| | Step undefined | | |

### Check if user able to create a program with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.010 s | | Steps |
|---|---|---|---|
| / 3:02:27.018 AM // 3:02:27.028 AM / | | | Total - 3 |
| | | | Pass - 0 |
| User Module | | | Fail - 0 |
| | | | Skip - 3 |
| @PostProgram | | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
| | Step undefined | | |
| 2 | When User sends HTTPS POST Request to create new Program | SKIPPED | 0.002 s |
| | Step undefined | | |
| 3 | Then User receives status code 201 with response body for creating a Program | SKIPPED | 0.001 s |
| | Step undefined | | |
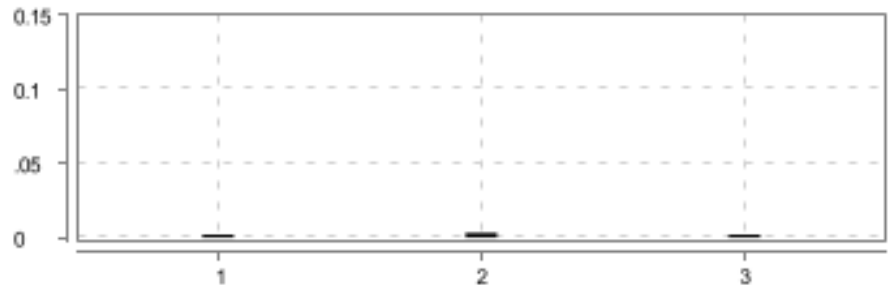
### Check if user able to create a Batch with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.013 s |
|---|---|
| / 3:02:27.039 AM // 3:02:27.052 AM / | |
| *User Module* | |
| @PostBatch | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

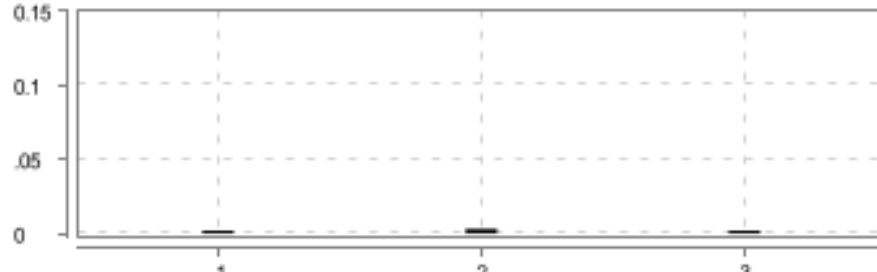| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.002 s |
| 2 | When User sends HTTPS POST Request to create new Batch<br>Step undefined | SKIPPED | 0.002 s |
| 3 | Then User receives status code 201 with response body for creating a Batch<br>Step undefined | SKIPPED | 0.002 s |

### Check if user able to create a User with valid endpoint and request body (non existing values)

| SKIPPED | DURATION - 0.014 s |
|---|---|
| / 3:02:27.062 AM // 3:02:27.076 AM / | |
| *User Module* | |
| @POSTRequest_UserWith_AdminRole | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

3

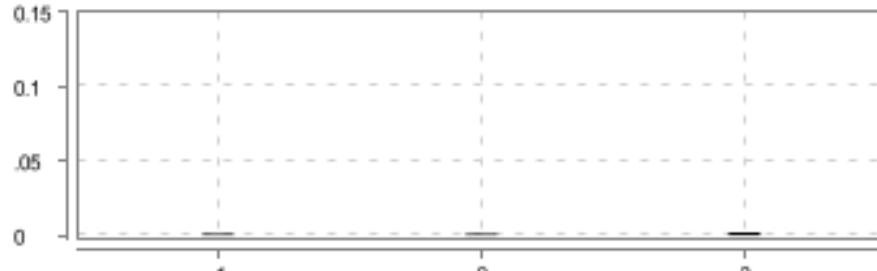| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.002 s |
| 2 | When User sends HTTPS POST Request for new User for Admin role<br>Step undefined | SKIPPED | 0.003 s |
| 3 | Then User receives status code 201 with response body for creating an User for Admin role | SKIPPED | 0.002 s |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| | Step undefined | | |

## Check if user able to update a user with valid User Id and request body

| SKIPPED | DURATION - 0.013 s | | |
|---|---|---|---|
| / 3:02:27.087 AM // 3:02:27.100 AM / | | | |
| User Module | | | |
| @PUTRequest_UpdateUserRoleStatusByDiffRoleID | | | |

Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3

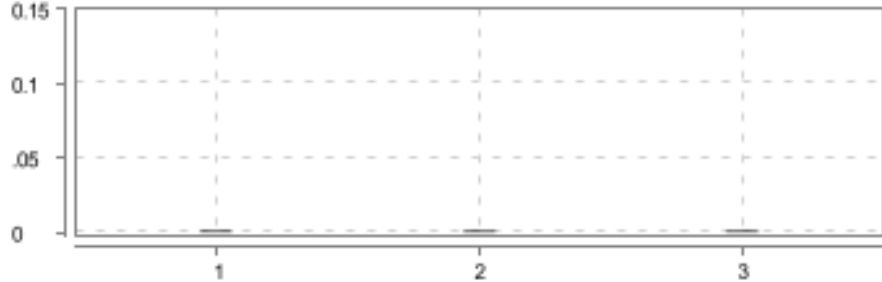| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
| | Step undefined | | |
| 2 | When User sends HTTPS PUT Request to update User role status by different Role ID | SKIPPED | 0.003 s |
| | Step undefined | | |
| 3 | Then User receives status code 400 with response body for updating UserRoleStaus | SKIPPED | 0.002 s |
| | Step undefined | | |

## Check if user able to assign a user with program and batch with invalid UserId

| SKIPPED | DURATION - 0.009 s | | |
|---|---|---|---|
| / 3:02:27.111 AM // 3:02:27.120 AM / | | | |
| User Module | | | |
| @PUTRequest_AssignUserToProgBatchByMissingfield | | | |

Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3
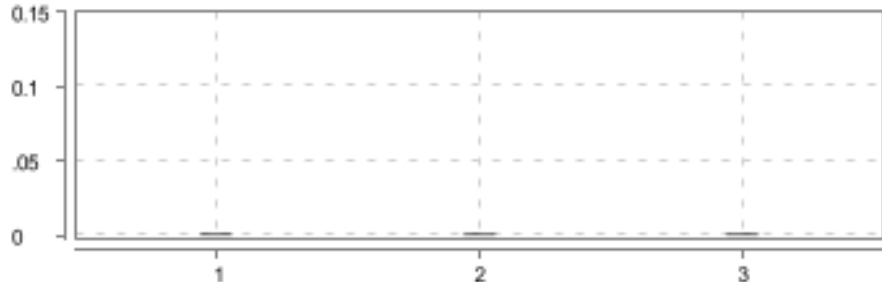
| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
| | Step undefined | | |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 2 | When User sends HTTPS PUT Request to assign User to given program and Batch for invalid UserId | SKIPPED | 0.001 s |
|  | Step undefined |  |  |
| 3 | Then User receives status code 400 with response body for assigning Program and Batch with invalid User | SKIPPED | 0.002 s |
|  | Step undefined |  |  |

### Check if user able to update a user with valid User Id and request body

| | | |
|---|---|---|
| SKIPPED | DURATION - 0.010 s | |
| / 3:02:27.131 AM // 3:02:27.141 AM / | | |
| User Module | | |
| @PUTRequest_AssignUserToProgBatchById | | |
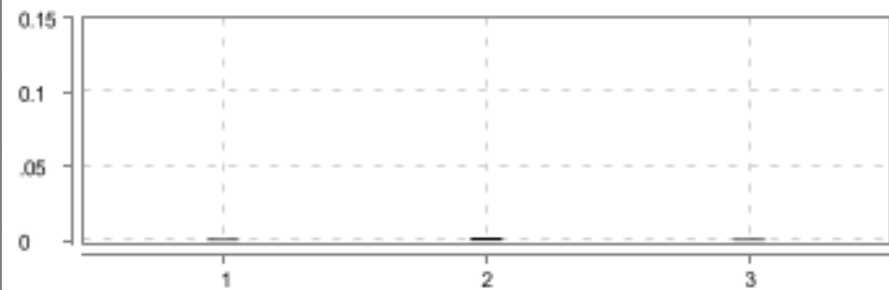
Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3



| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
|  | Step undefined |  |  |
| 2 | When User sends HTTPS PUT Request to assign User to given program and Batch | SKIPPED | 0.001 s |
|  | Step undefined |  |  |
| 3 | Then User receives status code 200 with response body for assigning Program and Batch to User | SKIPPED | 0.001 s |
|  | Step undefined |  |  |

### Check if user able to update a user with valid User Id and request body

| | | |
|---|---|---|
| SKIPPED | DURATION - 0.010 s | |
| / 3:02:27.152 AM // 3:02:27.162 AM / | | |
| User Module | | |
| @PUTRequest_AssignUserToProgBatchByInvalidID | | |
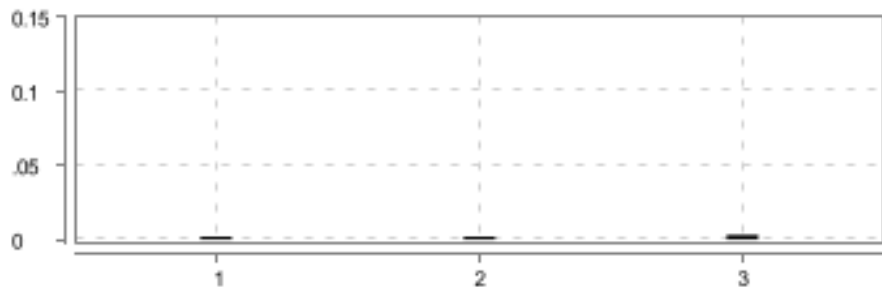
Steps
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.001 s |
| 2 | When User sends HTTPS PUT Request to assign User to given program and Batch by InvalidID<br>Step undefined | SKIPPED | 0.001 s |
| 3 | Then User receives status code 200 with response body for assigning Program and Batch to User<br>Step undefined | SKIPPED | 0.001 s |

## Check if user able to delete a user with valid User Id

| SKIPPED | DURATION - 0.011 s | | Steps |
|---|---|---|---|
| / 3:02:27.174 AM // 3:02:27.185 AM / | | | Total - 3 |
| *User Module* | | | Pass - 0 |
| | | | Fail - 0 |
| @DeleteRequest_DeleteAdminUser_ByID | | | Skip - 3 |

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint<br>Step undefined | SKIPPED | 0.001 s |
| 2 | When User sends HTTPS DELETE Request to delete a user<br>Step undefined | SKIPPED | 0.002 s |
| 3 | Then User receives status code 200 with response body to delete User by ID<br>Step undefined | SKIPPED | 0.001 s |
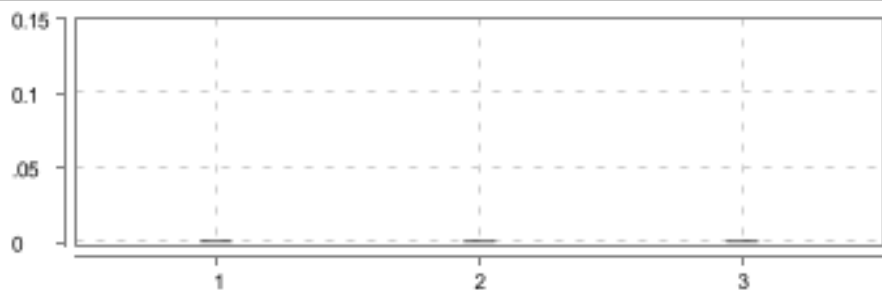
## Check if user able to delete a user with valid Batch Id

| SKIPPED | DURATION - 0.012 s |
|---|---|
| / 3:02:27.195 AM // 3:02:27.207 AM / | |
| User Module | |
| @DeleteRequest_DeleteBatch_ByID | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS DELETE Request to delete a Batch | SKIPPED | 0.002 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body to delete Batch by ID | SKIPPED | 0.003 s |
|   | Step undefined | | |

## Check if user able to delete a user with valid Program Id

| SKIPPED | DURATION - 0.010 s |
|---|---|
| / 3:02:27.218 AM // 3:02:27.228 AM / | |
| User Module | |
| @DeleteRequest_DeleteProgram_ByID | |

**Steps**
Total - 3
Pass - 0
Fail - 0
Skip - 3

| # | Step / Hook Details | Status | Duration |
|---|---|---|---|
| 1 | Given User creates  Request for the LMS API endpoint | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 2 | When User sends HTTPS DELETE Request to delete a Program | SKIPPED | 0.001 s |
|   | Step undefined | | |
| 3 | Then User receives status code 200 with response body to delete Program by ID | SKIPPED | 0.001 s |
|   | Step undefined | | |