Cypress.io

Test Protocol

[1 Site Search Bar Testing 3](#_Toc53735300)

[2 Dashboard Side Bar Testing 5](#_Toc53735301)

[3 Language Testing 6](#_Toc53735302)

# Site Search Bar Testing

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Test Case Name | Test Case Steps | Test Outcome |
| 1 | Search results should be relevant to searched term | Search for existing term via Search functionality | Correct search result should be displayed in the right corner |
| 2 | Verify that special characters won’t break the Search | Try using symbols and special characters to verify that Search functionality will not crash | Search results should either contain proper result, or show some default values. |
| 3 | Verify that suggested search results are generated dynamically | Write down some phrases and check if search results will get more specific as you finish writing the keyword | Search results should get from broader search to more specific one as you finish the keyword |
| 4 | Verify that clicking on Search Result redirects to the proper site | Write down some search keyword and check if the redirect will move to correct site | Correct site should be displayed |
| 5 | Verify that return from Search Results site redirects to the previous site. | Search for some keyword, click on the search result, click on “back” button in browser to check if you get back to the correct site. | Correct site should be displayed |
| 6 | Verify that deleting keyword from Search Bar deletes suggested search results | Write down some keyword and delete it. | Search results should disappear after deleting keyword. |
| 7 | Verify that proper amount of search results is displayed | Write down some keyword and check if the correct number of search results is shown. Consider using as specific and as broad keywords as you can. | You should see 5 search results. |
| 8 | Verify that keyword is highlighted in Search Result | Search for some specific keyword and check if the keyword is highlighted in Search Result | Keyword should be highlighted in search result. |
| 9 | Verify that reset button works correctly | Write down some keyword and delete it by using X symbol at the end of search bar | Keyword should be erased |
| 10 | Verify that reset button resets search results | Write down some keyword and delete it by using X symbol at the end of search bar | No search results should be kept under the search bar. |
| 11 | Verify that clicking outside the search bar closes the search results | Write down some keyword and click outside the search bar | Search results should be closed and reopened only if you click back into search bar |

# Dashboard Side Bar Testing

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Test Case Name | Test Case Steps | Test Outcome |
| 1 | Verify that “Dashboard” is collapsible | Click on Dashboard and check if it collapses and reopens properly | Dashboard should collapse or reopen after clicking |
| 2 | Verify that all links are redirecting to the proper pages | Check if all hyperlinks are working and redirecting properly | All hyperlinks should redirect to the proper pages |
| 3 | Verify that sidebar highlights the proper site after clicking on hyperlink | Click on some hyperlink in Dashboard and check if proper item is highlighted on the sidebar. | Proper keyword is highlighted after site redirect |
| 4 | Verify that “Dashboard” isn’t overlapping neighboring panels | Click on Dashboard and check if it’s not overlapping other panels. Consider also different resolutions for Mobile Phones. | Visibility is intact. |

# Language Testing

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Test Case Name | Test Case Steps | Test Outcome |
| 1 | Check if Time And Date format is correct for the defined region | Check if all Time And Date related notes are in correct format | Time And Date should have the correct format |
| 2 | Check if correct currency is shown | Check if proper currency is shown | Proper currency should be shown |
| 3 | Check if button text is not overlapping outside boundaries | Check visually if some texts are not out of bounds, or overlapping other buttons | Proper size should be used for buttons, etc. |
| 4 | Check if terminology used in translation is consistent on whole site | Check if terminology used in translation is consistent all over the website | No discrepancies in terminology should be there |
| 5 | Check for glaring grammatical errors in translations | Check if some gibberish is not displayed in translation | No grammatical errors should be found. |
| 6 | Check for character corruption | Try using special characters in translation and check if there aren’t any graphical bugs. | Special characters should be displayed properly |
| 7 | Go through basic behavior tests | Check if correct behavior is still working correctly | Everything should work the same way as is in default language |
| 8 | Check if all hyperlinks are redirecting to correct language page layout | Check all hyperlinks if they redirect to proper language version | Correct language hyperlink should be in place |
| 9 | Check if Search Bar works properly | Check if search bar searches for proper language versions, etc. For more tests regarding search bar, please check Task 1. | Proper test results should be returned. |

# Feature XX Testing

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Test Case Name | Test Case Steps | Test Outcome |
| 1 | Verify correct status codes | Try sending some commands and check if you get proper HTML status codes back, e.g. 404,200,500, etc. | Correct status code should be returned. |
| 2 | Verify correct payload | Try sending DELETE command and see if correct payload is removed from database | Correct payload should be removed |
| 3 | Verify response header | Check response header from previous step for any discrepancies or security weaknesses | Response header should have correct format |
| 4 | Verify correct application state | Please check if the application won’t get stuck in some state | API should be responsive after finishing task |
| 5 | Verify that response is sent in timely manner | Send DELETE command and measure time it takes | Payload is deleted in timely manner |
| 6 | (FUTURE) Test GET functionality | Send GET command and check response | Correct response is received |
| 7 | (FUTURE) Test POST functionality | Send POST command and check response | Correct response is received |
| 8 | (FUTURE) Test PUT functionality | Send PUT command and check response | Correct response is received |
| 9 | Test DELETE functionality | Send DELETE command and check response | Correct response is received |
| 10 | Test Positive Flow for DELETE | Send DELETE Command with all additional parameters and check the response | DELETE command should be finished with response 200 OK |
| 11 | Test Negative Flow for DELETE – CORRECT DATA | Send DELETE Command with additional parameters that were already deleted and check the response | DELETE command should be finished with response 500, 501, 404 etc. |
| 12 | Test Negative Flow for DELETE – INCORRECT DATA | Send DELETE Command with wrong parameters and check the response | DELETE command should be finished with response 500, 501, 404 etc. |
| 13 | Verify that empty payload won’t break API | Send empty DELETE command | API should return with error code and continue working |
| 14 | Verify that oversized payload won’t break API | Send oversized DELETE command (1000+characters long, etc.) | API should return with error code and continue working |
| 15 | Verify that illegal characters in payload won’t break API | Send DELETE command with forbidden characters | API should return with error code and continue working |
| 16 | Check for security weaknesses | Make sure you need to send commands in secure manner | No issues should be found |