|  |  |
| --- | --- |
| **Project:** Module 4.2 - Todo Application Testing |  |
| **Course:** CSD 380 |  |
| **Description:** Testing core functionalities of the Todo application based on the business requirements. |  |
| **Date:** 2025/01/26 |  |

To complete the test case plan, fill out the information for Project, Course, Description and Date in the header below.

.

**TABLE OF CONTENTS**

[**View Todo Task List** 2](#_Toc188801077)

[**Create New Todo Task** 2](#_Toc188801078)

[**Edit an Existing Todo Task** 3](#_Toc188801079)

[**Delete a Todo Task** 3](#_Toc188801080)

[**Invalid URL Handling** 4](#_Toc188801081)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | **View Todo Task List** | | | |
|  | **Test Objective:** Verify that users can view a list of Todo tasks | **Developer:** Gerald Oden  **Date tested:** 2025/01/26 | **Peer tester:** Vincent Ryan  **Date tested:** 2025/02/02 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Navigate to <https://buwebdev.github.io/todo/> | The landing page loads without errors, displaying the todo task list. | Yes |  |
| 2 | Confirm that the task list contains all existing tasks. | The task list displays correct and complete data. | Yes | There are 6 tests |
| 3 | Refresh the page. | The task list reloads and retains its state. | Yes | The page looks the same as the original landing. |
| **Comments** | The test ran successfully. The task list contains 6 line entries, and displays them in the same order with multiple refreshes. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Create New Todo Task** | | | |
|  | **Test Objective:** Verify that users can create a new task successfully. | **Developer:** Gerald Oden  **Date tested:** 2025/01/26 | **Peer tester:** Vincent Ryan  **Date tested:** 2025/02/02 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to https://buwebdev.github.io/todo/ | The application loads without errors. | Yes | Yes |
| 2 | Enter a valid todo item in the task creation field and click the submit button. | The new task appears in the task list. | Yes | Yes |
| 3 | Verify the persistence by refreshing the page. | The new task remains visible after the page reloads. | No | No |
| 4 | Attempt to submit an empty todo item. | The system prevents submission and displays an error message. | Yes | Yes |
| **Comments** | New task items can be created. They cannot be blank. To be specific they can’t have fewer than 4 characters. 4 spaces does act as valid, and can create a blank task. Tasks are also not saved on refresh as seen above. Only the default tasks are saved. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Edit an Existing Todo Task** | | | |
|  | **Test Objective:** Verify that users can edit an existing task. | **Developer:** Gerald Oden  **Date tested:** 2025/01/26 | **Peer tester:** Vincent Ryan  **Date tested:** 2025/02/02 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Select an existing task from the list and click the edit button. | The task details load into an editable form. | Yes | Yes |
| 2 | Modify the task content and save changes. | The task list updates to reflect the edited content. | Yes | Yes |
| 3 | Refresh the page. | The edited task persists after reloading. | No | No |
| 4 | Attempt to save an empty edit. | The system prevents the update and displays an error message. | No |  |
| **Comments** | Editing tasks functions with limitations. We don’t want blank line items, but it will allow you to delete the contents and save them as empty as shown above. Saved edits are overwritten upon refreshing the page. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Delete a Todo Task** | | | |
|  | **Test Objective:** Verify that users can delete a task. | **Developer:** Gerald Oden  **Date tested:** 2025/01/26 | **Peer tester:** Vincent Ryan  **Date tested:** 2025/02/02 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the task list and select a task to delete. | The task is removed from the list. | Yes | Yes |
| 2 | Refresh the page. | The deleted task does not reappear. | No | No |
| 3 | Attempt to delete a task and immediately refresh during the operation. | The task remains deleted after the operation completes. | No | No (same as above) |
| **Comments** | Same error with deletion as there was with editing or adding; changes are not saved upon refreshing the page. Deleting items works as expected and the list adjusts correctly. There is no “confirmation” or “undo” to prevent accidental deletions which may need to be considered. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Invalid URL Handling** | | | |
|  | **Test Objective:** Verify that the system displays a custom 404 error page for invalid URLs. | **Developer:** Gerald Oden  **Date tested:** 2025/01/26 | **Peer tester:** Vincent Ryan  **Date tested:** 2025/02/02 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to an invalid URL, such as <https://buwebdev.github.io/todo/emptypage> | The application displays a custom 404 error page. | Yes | Yes |
| 2 | Verify that the custom page includes navigation links to return to the main site. | Links are functional and redirect users to the homepage. | No | No. |
| **Comments** | There is a not-found page but the page doesn’t allow you to navigate back to home. It works to deliver a custom missing page, but doesn’t function to return the user where they should have gone. | | | |