|  |  |
| --- | --- |
| **Project:** Module 4 Test case | Jacob Cannamela |
| **Course:** CSD-380 |  |
| **Description:** Module 4 Test cases |  |
| **Date:** 06/19/2025 |  |

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

.

**TABLE OF CONTENTS**

[Allow users to visit the website through https://buwebdev.github.io/todo/ 2](#_Toc201234271)

[Prevent users from submitting a task form with without entering a todo item 2](#_Toc201234272)

[Edit a todo task item 3](#_Toc201234273)

[Delete a todo task item 3](#_Toc201234274)

[Prevent users from submitting the edit form without entering a todo item 4](#_Toc201234275)

For each test, the developer should: provide a test description, a test objective, the developer name and date tested. For each step, fill out the actions to be taken and describe the expected results, and indicate yes or no for each step depending upon the result. The peer tester should provide their name, date tested, indicate yes or no for each step depending upon the result, and a screenshot (thumbnail) of the result.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | Allow users to visit the website through <https://buwebdev.github.io/todo/> | | | |
|  | **Test Objective:** Verifies the website loads correctly from the proper link | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Truman Forey**  **Date tested:** <2025/06/21> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Visit the site | Site loads | <yes/no> | Yes |
| 2 | Confirm the page loaded properly | No errors | <yes/no> | Yes |
| 3 | Refresh the page | Site loads | <yes/no> | Yes |
| 4 | Confirm the page reloaded correctly | No errors | <yes/no> | Yes |
| **Comments** | Pages loads normally, as expected. Items on page are effectively ‘reset’ to the page’s defaults when the page is loaded/reloaded. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | Prevent users from submitting a task form with without entering a todo item | | | |
|  | **Test Objective: ensure input validation works on the todo form** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Truman Forey**  **Date tested:** <2025/06/21> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Load web page | Site loads properly | <yes/no> | Yes |
| 2 | Attempt to submit an empty todo task | User clicks the add task button | <yes/no> | Yes |
| 3 | validate how the system reacts | System throws an error | <yes/no> | No |
| 4 | Enter a valid todo task | User clicks the add task button | <yes/no> | Yes |
| 5 | Validate how the system reacts | New task is added to the todo list | <yes/no> | Yes |
| **Comments** | Entering an empty task is impossible – pressing the ‘add task’ button with no text in the task field produces no result. In addition, tasks must have at least 4 characters in order to be submitted. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | Edit a todo task item | | | |
|  | **Test Objective: validate users can edit the items on the todo list** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Truman Forey**  **Date tested:** <2025/06/21> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Create the todo list by submitting items | Todo list is built | <yes/no> | Yes |
| 2 | Click the edit button on a todo list item | Todo list item creates a text box to allow editing of the item | <yes/no> | Yes |
| 3 | Make any changes and hit save | The todo list item is changed | <yes/no> | Yes |
| **Comments** | Edit function works as expected. Adding too many characters leads to the edit and delete buttons floating off of the todo list field. Strangely, any spaces in the front of the first word are removed. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | Delete a todo task item | | | |
|  | **Test Objective: validate users can delete the items on the todo list** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Truman Forey**  **Date tested:** <2025/06/21> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Create the todo list by submitting items | Todo list is built | <yes/no> | Yes |
| 2 | Click the delete button on a todo list item | Todo list item is deleted | <yes/no> | Yes |
| **Comments** | Delete function works as expected. All items on todo list can be deleted without causing an issue. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | Prevent users from submitting the edit form without entering a todo item | | | |
|  | **Test Objective: ensure input validation works on the todo form when editing tasks** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Truman Forey**  **Date tested:** <2025/06/21> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Load web page | Site loads properly | <yes/no> | Yes |
| 2 | Attempt to edit a task, and leave it empty, hit save | System throws an error | <yes/no> | No |
| **Comments** | This function does not work – todo tasks can be left empty while editing them, despite being unable to be created empty. Site does not throw an error when attempting to leave a task empty while editing it. | | | |