|  |  |
| --- | --- |
| **Project:** Module 4.2 – To Do Tests | Peer Tester: Greih Murray |
| **Course:** CSD 380 |  |
| **Description:** Test cases for Module 4.2 BUWebDev ToDo App. |  |
| **Date:** 2024-09-04 |  |
| **Name**: Ryan Norrbom |  |

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

.

**TABLE OF CONTENTS**

[**Test Case 1** 2](#_Toc132718195)

[**Test Case 2** 2](#_Toc132718196)

[**Test Case 3** 3](#_Toc132718197)

[**Test Case 4** 3](#_Delete_ToDo_Task)

[**Test Case 5** 4](#_Prevent_Empty_Task)

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 | **Create New ToDo Task Item** | | | |
|  | **Test Objective:** Verifies input data meets requirements | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Greih Murray**  **Date tested:** 2024/09/08 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Navigate to the URL: [Todo Application](https://buwebdev.github.io/todo/). | Website Loads to Todo App | <yes/no> | Yes |
| 2 | Click on the Task test field | Text field will allow input | <yes/no> | Yes |
| 3 | Enter a task title in the input field (e.g., "Buy groceries"). | Entered task is visible in text field and submit button is clickable | <yes/no> | Yes (See above) |
| 4 | Click the “Add Task" button. | Task is entered to todo list | <yes/no> | Yes |
| 5 | Review Added task in ToDo list | Task is seen in todo list | <yes/no> | Yes (See above) |
| **Comments** | Each component functions as expected and the test passes. Given that there were some steps with duplicate validation it was decided to use a single image as opposed to duplicating the image within the document | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **View a List of ToDo Task Items** | | | |
|  | **Test Objective:** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Greih Murray**  **Date tested:** 2024/09/08 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the URL: [Todo Application](https://buwebdev.github.io/todo/). | Website Loads to Todo App | <yes/no> | Yes |
| 2 | Review Todo list items | Items show as visible on the todo list | <yes/no> | Yes (See above) |
| **Comments** | The url works as expected and the page loads with the default to do list items pre-populated. Again, both steps require the same validation so a single image was used. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Edit a ToDo Task Item** | | | |
|  | **Test Objective:** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Greih Murray**  **Date tested:** 2024/09/08 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the URL: [Todo Application](https://buwebdev.github.io/todo/). | Website Loads to Todo App | <yes/no> | Yes |
| 2 | View Todo list | Todo list items are visible on the page | <yes/no> | Yes (See above) |
| 3 | Select the Edit button | Task item opens in text form | <yes/no> | Yes |
| 4 | Add text to Todo task | New text is added to input field | <yes/no> | Yes |
| 5 | Click save | Task appears on Todo list with new text | <yes/no> | Yes |
| **Comments** | Each step within the test passes as expected, editing the item causes its edits to appear immediately. Of note is the lack of length limits on the edit field which is inconsistent with the requirements on the add item field | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Delete ToDo Task Item** | | | |
|  | **Test Objective:** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Greih Murray**  **Date tested:** 2024/09/08 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the URL: [Todo Application](https://buwebdev.github.io/todo/). | Website Loads to Todo App | <yes/no> | Yes |
| 2 | Click the Trash icon in the Todo list | Task disappears from the Todo list | <yes/no> | Yes |
| **Comments** | Test passes as expected, the garbage icon works for deleting items. Might recommend a confirmation pop up of some form to prevent accidental deletion of items. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Prevent Empty Task Item Submission** | | | |
|  | **Test Objective:** | **Developer:**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Greih Murray**  **Date tested:** 2024/09/08 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the URL: [Todo Application](https://buwebdev.github.io/todo/). | Website Loads to Todo App | <yes/no> | Yes |
| 2 | Click “Add Task” | Nothing happens | <yes/no> | Yes (See above) |
| 3 | Look for empty task in Todo list | No new or empty Task in todo list | <yes/no> | Yes |
| **Comments** | Test passes successfully, the user is unable to add a blank item to the to-do list. Difficult to get a picture of clicking on the button as I click, however the button is disabled when no input is given | | | |