|  |  |
| --- | --- |
| **Project:** Todo Application Functional Test Plan |  |
| **Course:** CSD-380 DevOps |  |
| **Description:** Functional Tests validating the Todo web app |  |
| **Date: 9/10/2025** |  |

.

**TABLE OF CONTENTS**

[**<TC1: Access and View Tasks** 2](#_Toc132718195)

[**<TC2: Create Task and Block Empty Submission**](#_Toc132718196) 3

[**<TC3: Edit Task and and Block Empty Submission** 4](#_Toc132718197)

[**<TC4: Delete Task** 5](#_Toc132718195)

[**<TC5: Custom 404 for Invalid URL** 5](#_Toc132718196)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | **Access and View Tasks** | | | |
|  | **Test Objective:** Verifies users can access the site via the URL and see list of tasks | **Developer: Andy Stiles**  **Date tested:** <2025/09/10> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Navigate to https://buwebdev.github.io/todo/ | Page Loads without errors | Yes | <yes/no> |
| 2 | Observe initial task list | Existing tasks display, layout is readable | Yes | <yes/no> |
| 3 | Refresh the browser | Page reloads and tasks remain visible | Yes | <yes/no> |
| 4 | Click any visible navigation or logo (if present) | Remains within the Todo app without errors | Yes | <yes/no> |
| **Comments** | Access via the standard URL works and the list view renders as expected. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Edit Task and Block Empty Submission** | | | |
|  | **Test Objective:** | **Developer: Andy Stiles**  **Date tested:** <2025/09/10> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | In the “New Task” input, type: Buy milk | Text appears in the input field | yes | <yes/no> |
| 2 | Click **Add Task** | Task Buy milk appears in the list | Yes | <yes/no> |
| 3 | Refresh the page | |  | | --- | |  |   “Buy Milk” Still appears on in the list | No | <yes/no> |
| 4 | |  | | --- | |  |  |  | | --- | | Clear the input and click **Add** with the field empty | | App blocks submission and shows a message or prevents adding a blank row | No | <yes/no> |
| **Comments** | Database no updating internally. Only editing on surface level. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Create Task and Block Empty Submission** | | | |
|  | **Test Objective:** Verify users can edit an existing task and cannot save an empty task during edit. | **Developer: Andy Stiles**  **Date tested:** <2025/09/10> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | |  | | --- | |  |  |  | | --- | | Select an existing task (for example, Buy milk) and click **Edit** | | Task switches to edit mode | Yes | <yes/no> |
| 2 | |  | | --- | |  |  |  | | --- | | Change text to Buy milk and eggs, click **Save** | | List shows updated text | Yes | <yes/no> |
| 3 | Re-open edit on the same task, delete all text, click **Save** | App blocks saving an empty value and keeps prior text | No | <yes/no> |
| 4 | |  | | --- | |  |  |  | | --- | | Refresh the page | | |  | | --- | |  |  |  | | --- | | Edited value persists | | No | <yes/no> |
| **Comments** | Edit flow succeeds except doesn’t block empty editing, and doesn’t keep tasks when refreshed. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Delete Task** | | | |
|  | **Test Objective:** Verify users can delete an existing task. | **Developer: Andy Stiles**  **Date tested:** <2025/09/11> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Identify an existing task (Or Create One if needed) | Task is visible in the list | Yes | <yes/no> |
| 2 | Click “Delete” for that task | Task disappears from list | Yes | <yes/no> |
| 3 | Refresh the page | Deleted task does not reappear | No | <yes/no> |
| **Comments** | Deletion removes the task but returns upon refresh. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Custom 404 for Invalid URL** | | | |
|  | **Test Objective:** Verify an invalid URL under the site shows a custom 404 page. | **Developer: Andy Stiles**  **Date tested:** <2025/09/11> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to https://buwebdev.github.io/todo/invalidpage | Custom 404 page is displayed | Yes | <yes/no> |
| 2 | |  | | --- | |  |  |  | | --- | | Use the browser Back button | | Returns to prior page without error | Yes | <yes/no> |
| **Comments** | Invalid routes correctly show the custom 404 experience. No issues found. | | | |