David Amos

Module 4.3 Assignment

1/31/2025

Tests from developer Alex Zayas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | **Ensure To Do List is Displayed** | | | |
|  | **Test Objective: Make Sure users can view the list of todo items when they access the app.** | **Developer: Alex Zayas** | **Peer tester: David Amos** |  |
|  |  | **Date tested:** <2025/01/30> | **Date tested:** <2025/01/31> |  |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
|  |  |  |  | **+ Screenshot** |
| 1 | Open https://buwebdev.github.io/todo/ in a browser | The page loads successfully without errors | Pass | Yes |
| 2 | Check if the Todo List section is visible | A list of existing todo tasks is displayed | Pass | Yes |
| 3 | Ensure the tasks are presented in a readable format | Tasks should be listed clearly with edit and delete icons next to each task | Pass | Yes |
| **Comments** | The test verifies that the user can access the todo list and view existing tasks. If tasks are not displayed, verify the API call or database connection.Tester notes: Page loads with no errors. All tasks appear with correct icons. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Create a new Todo Task** | | | |
|  | **Test Objective: Validate that a user can successfully create a new todo task.** | **Developer: Alex Zayas** | **Peer tester: David Amos** |  |
|  |  | **Date tested:** <2025/01/30> | **Date tested:** <2025/01/31> |  |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
|  |  |  |  | **+ Screenshot** |
| 1 | Navigate to the Todo App | The application loads successfully | Pass | Yes |
| 2 | Enter a new task into the Task Form field | The input field accepts text input | Pass | Yes |
| 3 | Click the “Add Task” button | The new task appears in the Todo List section | Pass | Yes |
| 4 | Refresh the page | The newly added task should still be present in the list | Fail | Fail |
| **Comments** | This test ensures that tasks can be created and persist in the list after refreshing. If the task does not appear, recheck logic/ debug. Tester notes: Adding a new task was successful. The newly added task however is no longer present after a refresh. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Edit an Existing Todo Task** | | | |
|  | **Test Objective: Confirm that users can edit an existing todo task.** | **Developer: Alex Zayas** | **Peer tester: David Amos** |  |
|  |  | **Date tested:** <2025/01/30> | **Date tested:** <2025/01/31> |  |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
|  |  |  |  | **+ Screenshot** |
| 1 | Click the edit icon next to an existing task | An input field appears allowing the user to edit the task | Pass | Yes |
| 2 | Modify the task text and save changes | The updated task should replace the old text | Pass | Yes |
| 3 | Refresh the page | The edited task should still reflect the changes | Fail | Fail |
| **Comments** | If edits are not saving, verify that the JavaScript event handlers are working correctly and that updates persist in local storage or the database. Tester notes: Editing a task was successful. The update however is no longer present after a refresh. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Delete a Todo Task** | | | |
|  | **Test Objective: Ensure users can delete an existing task from the Todo List.** | **Developer: Alex Zayas** | **Peer tester: David Amos** |  |
|  |  | **Date tested:** <2025/01/30> | **Date tested:** <2025/01/31> |  |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
|  |  |  |  | **+ Screenshot** |
| 1 | Click the trash icon next to an existing task | The task is removed from the list | Pass | Yes |
| 2 | Refresh the page | The deleted task should no longer appear in the list | Fail | Fail |
| **Comments** | If deleted tasks reappear upon refresh, check if the deletion is correctly updating the data source. ester notes: Deleting a task was successful. The deleted task however reappears after a refresh. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Prevent Empty Task Submission** | | | |
|  | **Test Objective: Validate that users cannot submit an empty task.** | **Developer: Alex Zayas** | **Peer tester: David Amos** |  |
|  |  | **Date tested:** <2025/01/30> | **Date tested:** <2025/01/31> |  |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
|  |  |  |  | **+ Screenshot** |
| 1 | Click the “Add Task” button without entering text | The system should prevent empty task submission | Pass | Yes |
| 2 | Enter spaces (without characters) and click “Add Task” | The system should still prevent submission | Pass | Fail |
| **Comments** | If empty tasks are being accepted, check form validation logic. Note from tester: Page allows a task to be added if 4 or more spaces are entered. Works as intended for empty or only 1-3 spaces. | | | |

Tests from developer Caitlan Nichols:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | **Verify Users Can Visit the Website** | | | |
|  | **Test Objective:**  Verify that users can access the website through URL. | **Developer: Caitlan Nichols**  **Date tested:** 2025/01/29 | **Peer tester: David Amos**  **Date tested:** 2025/01/31 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Open a browser and navigate to <https://buwebdev.github.io/todo/> | The landing page loads without errors | yes | Yes |
| 2 | Refresh page to see if it reloads | Page continues to show no errors | yes | Yes |
| 3 | Enter an invalid URL (e.g., <https://buwebdev.github.io/todo/invalid>) | A custom 404 error page is displayed. | yes | Yes |
| **Comments** | All steps worked as intended. Page loads correctly and an error will display when using an incorrect URL. The error message could be improved to better explain the issue to the user. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Verify Users Can Use Website** | | | |
|  | **Test Objective:** Verify that the system displays the Todo list | **Developer: Caitlan Nichols**  **Date tested:** 2025/01/29 | **Peer tester: David Amos**  **Date tested:** 2025/01/31 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the landing page | A list of Todo task items is visible. | Yes | Yes |
| 2 | Check if task list updates after refresh | Task list remains displays saved tasks. | yes | Yes |
| **Comments** | Tests work as stated. However, refreshing the page will reset the items to their original values. No updates are saved. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Creation of new task** | | | |
|  | **Test Objective:** ensure tasks can be added to the list | **Developer: Caitlan Nichols**  **Date tested:** 2025/01/29> | **Peer tester: David Amos**  **Date tested:** 2025/01/31 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Click the ‘Add task’ button | Task creation form appears | yes | Yes |
| 2 | Enter Valid test in task description and submit form | New task appears in the task list | yes | Yes |
| 3 | Submit form without entering a desc. | Error message displayed | yes | Fail |
| **Comments** | Adding a task is not in a creation form but rather on the home screen. No error message appears when add task button is shown, the button is just not interactable. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Editing a task** | | | |
|  | **Test Objective: :** ensure tasks can be modified | **Developer: Caitlan Nichols**  **Date tested:** 2025/01/29> | **Peer tester: David Amos**  **Date tested:** 2025/01/31 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Click the 'Edit' button next to an existing task. | The edit form appears with current details already filled from prior use. | yes | Yes |
| 2 | Modify the task description and submit the form. | The task list updates to reflect the new description. | yes | Yes |
| 3 | Submit the form with an empty description. | An error message is displayed. | yes | Fail |
| **Comments** | When submitting the edit form with an empty description no error displays. Instead the item is updated but is blank. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Deleting a task** | | | |
|  | **Test Objective: :** ensure tasks can be removed | **Developer: Caitlan Nichols**  **Date tested:** 2025/01/29> | **Peer tester: David Amos**  **Date tested:** 2025/01/31 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Click the 'Delete' button next to a task. | A confirmation prompt appears. | yes | Fail |
| 2 | Confirm the deletion. | The task is removed from the list. | yes | Yes |
| 3 | Attempt to delete a non-existent task. | No action occurs or an appropriate error message is displayed. | yes | Yes |
| **Comments** | No confirmation appears as stated in expected results. Item is deleted successfully. Non-existent tasks cannot be deleted as there would be no delete button to click. | | | |