|  |  |
| --- | --- |
| **Project:** ToDo Application | Tester Jason Schriner |
| **Course:** CSD-380 DevOps | 2025/18/6 |
| **Description:** Functional test cases based on the Todo Business Requirements document |  |
| **Date:** (2025/06/16) |  |

Jose Franco

Module 4.2 Assignment – Test Plan

**TABLE OF CONTENTS**

[**Verify Access to Website via Standard URL >** 2](#_Toc200662917)

[**<Create New Todo Task>** 3](#_Toc200662918)

[**<Edit a Task>** 4](#_Toc200662919)

[**<Validate Empty Task Form Submission>** 5](#_Toc200662920)

[**<Custom 404 Page for Invalid URLs>** 6](#_Toc200662921)

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 | **<** **Verify Access to Website via Standard URL >** | | | |
|  | **Test Objective:**  Ensure users can access the Todo site using the standard URL and that the page loads correctly. | **Developer: Jose Franco**  **Date tested:** 2025/06/16 | **Peer tester: Jason Schriner**  **Date tested:** 2025/06/18 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Open browser and enter URL | Page loads without error | Yes | Yes |
| 2 | Refresh the browser | Page reloads successfully | Yes | Yes |
| 3 | Check the URL uses HTTPS | HTTPS connection is active and secure | Yes | Yes |
| 4 | Confirm navigation is responsive | Menu and layout adjust properly on window resize | Yes | Yes |
| 5 | Click on any button or link | Page responds and navigates correctly | Yes | Yes |
| **Comments** | Step one browser opened, and I was able to enter the web address and successfully navigate to the site. I was also able to prove security certificate was present. Page continues to work correctly even when refreshed or resized. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **<Create New Todo Task>** | | | |
|  | **Test Objective:** Ensure users can successfully add a new task. | **Developer:Jose Franco**  **Date tested:** 2025/06/16 | **Peer tester: Jason Schriner**  **Date tested:** 2025/06/18 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Load the Todo page | Page loads with input form visible | Yes | Yes |
| 2 | Enter a task name in the input box | Input accepted | Yes | Yes |
| 3 | Click the “Add” or “+” button | Task is added to the list below | Yes | Yes |
| 4 | Confirm the new task appears in list | Task is listed with no errors | Yes | Yes |
| 5 | Refresh the page | Task remains if stored persistently | No | No |
| **Comments** | The input works correctly and allows me to click on it and enter text. Clicking the submit button adds the new task to the list. However, the task disappears if the refresh button is hit. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **<Edit a Task>** | | | |
|  | **Test Objective:** Confirm that existing tasks can be edited and changes saved | **Developer: Jose Franco**  **Date tested:** 2025/06/16 | **Peer tester: Jason Schriner**  **Date tested:** 2025/06/18 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Add a new task | Task is created successfully | Yes | Yes |
| 2 | Click the “Edit” icon or button | Edit form or input becomes active | Yes | Yes |
| 3 | Change the task text | Input accepted | Yes | Yes |
| 4 | Click “Save” | Task text updates to new value | Yes | Yes |
| 5 | Reload the page | Edited text remains (if data is persisted) | No | No |
| **Comments** | The edit button opens a new task box which allows users to edit the task. It is possible to add and delete text from this box and the new task is entered correctly when the save button is clicked. The task reverts back to the original when the refresh button is hit. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **<Validate Empty Task Form Submission>** | | | |
|  | **Test Objective:** Ensure the app blocks users from submitting empty todo entries. | **Developer: Jose Franco**  **Date tested:** 2025/06/16 | **Peer tester: Jason Schriner**  **Date tested:** 2025/06/18 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Open the Todo page | Input field and add button are visible | Yes | Yes |
| 2 | Leave input field blank | Input remains empty | Yes | Yes |
| 3 | Click the “Add” or submit button | No task is added to the list | Yes | Yes |
| 4 | Check for validation message | Message or alert shown (if implemented) | Yes | No |
| 5 | Enter valid task after that | Task added successfully | Yes | Yes |
| **Comments** | It is not possible to input a blank task. The add task button is not clickable until information is added. It does become clickable after text is added and ads the new task to the list when hit. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **<Custom 404 Page for Invalid URLs>** | | | |
|  | **Test Objective:** Confirm that the site displays a custom 404 page for invalid routes. | **Developer: Jose Franco**  **Date tested:** 2025/06/16 | **Peer tester: Jason Schriner**  **Date tested:** 2025/06/18 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Open browser | Navigate to homepage | Yes | Yes |
| 2 | Manually type an invalid sub-path (e.g., `/xyz`) | Page loads with custom 404 error | Yes | Yes |
| 3 | Check for branding/message on 404 page | Page contains project-specific 404 content | Yes | Yes |
| 4 | Click "Back to home" (if present) | Redirects back to main page | No | No |
| 5 | Try another invalid URL | Same 404 page behavior observed | Yes | Yes |
| **Comments** | The page loads correctly when the correct address is entered. If bad information is added to the link, it provides a custom 404 error to let users know they have an invalid address. | | | |