|  |  |
| --- | --- |
| **Project:** Test Plan |  |
| **Course:** CSD380 DevOps |  |
| **Description:** Test plan based on the business requirements for the Todo Business Requirements website. Includes 5 functional tests that address the requirements. |  |
| **Date:** 2025/01/28 |  |

**TABLE OF CONTENTS**

[**Create a new todo task item** 2](#_Toc132718195)

[**View a list of todo task items** 2](#_Toc132718196)

[**Edit a todo task item** 3](#_Toc132718197)

[**Delete a todo task item** 3](#_Toc132718197)

[**Prevent a submission without a todo task item** 3](#_Toc132718197)

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 a new todo task item** | | | |
|  | **Test Objective: Verify that users are able to create a new task and that it is displayed within the task list.** | **Developer:**  **Date tested:** 2025/01/28 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Navigate to the Todo App homepage. | The web page loads correctly. | Yes | <yes/no> |
| 2 | Locate the “Task Form” input field. | Page is functioning properly and that users can add new tasks. | Yes | <yes/no> |
| 3 | Enter a valid task name. | Users can add input text. | Yes | <yes/no> |
| 4 | Click the “add task” button. | Users can submit text information. | Yes | <yes/no> |
| 5 | Verify that the new task is displayed in the list of tasks. | User input (tasks) are displayed in the Todo List. | Yes | <yes/no> |
| **Comments** | Comments should be substantive; this means there should be at least 2-3 well-structured sentences with constructive feedback. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **View a List of Todo Task Items** | | | |
|  | **Test Objective: Verify that users are able to view a list of existing tasks within the application.** | **Developer:**  **Date tested:** 2025/01/28 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the Todo homepage. | The web page should load properly. | Yes | <yes/no> |
| 2 | Confirm that the page loads and displays the task list. | The web page should automatically display the task list to users. | Yes | <yes/no> |
| 3 | Check that each task is clearly visible and properly formatted on the webpage. | Each task should be properly formatted and visible. | Yes | <yes/no> |
| 4 | Verify that the list contains the tasks that were previously created. | The page should display all existing tasks. | Yes | <yes/no> |
| 5 | Scroll through the list to ensure that it fully and properly displays all tasks. | The page should be visible and does not cut off or shorten any tasks that exist. | Yes | <yes/no> |
| **Comments** | Comments should be substantive; this means there should be at least 2-3 well-structured sentences with constructive feedback. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Edit a task item** | | | |
|  | **Test Objective: Verify that users are able to edit existing tasks and edit their contents.** | **Developer:**  **Date tested:** 2025/01/28 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Locate a pre-existing task on the list. | Tasks should appear in list format on the webpage. | Yes | <yes/no> |
| 2 | Click the “edit” icon associated with the class. | Once the edit icon has been clicked, the edit form should open. Users should be able to edit the task. | Yes | <yes/no> |
| 3 | Modify the task to have a different value. | Users should be able to edit task information. | Yes | <yes/no> |
| 4 | Click the “save” button. | Save button should be functional. | Yes | <yes/no> |
| 5 | Confirm that the updated item appears in the list, properly displaying the modified information. | The task item should be properly saved. | Yes | <yes/no> |
| **Comments** | Comments should be substantive; this means there should be at least 2-3 well-structured sentences with constructive feedback. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Delete a todo task item** | | | |
|  | **Test Objective: Verify that users can delete existing tasks from the task list.** | **Developer:**  **Date tested:** 2025/01/28 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Identify and locate any existing task within the task list. | The ToDo list should be loaded for the user. | Yes | <yes/no> |
| 2 | Locate the “Delete” icon on the task. | The trash icon should appear for users. | Yes | <yes/no> |
| 3 | Click the “Delete” icon on the task. | The user should be able to delete items by clicking this icon. | Yes | <yes/no> |
| 4 | Verify that the item has been removed from the task list. | The task should no longer appear in the list for the user. | Yes | <yes/no> |
| 5 | Refresh page to ensure that it has been deleted. | When the page is refreshed, the deleted item should be reflected to the user. | No | <yes/no> |
| **Comments** | Comments should be substantive; this means there should be at least 2-3 well-structured sentences with constructive feedback. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Prevent from submission without a todo item** | | | |
|  | **Test Objective: Verify that users are unable to submit forms with empty input fields.** | **Developer:**  **Date tested:** 2025/01/28 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Navigate to the website home page. | Home page should load properly for the user. | Yes | <yes/no> |
| 2 | Locate the “Add Task” input field. | Users should be able to input text to create a new task. | Yes | <yes/no> |
| 3 | Leave the input field blank. | Leaving the text field blank should not prompt the button for task submission to the user. | Yes | <yes/no> |
| 4 | Click the submit button. | Users should not be able to submit empty fields. | Yes | <yes/no> |
| 5 | Confirm that the form is not submitted. | The form was not submitted. | Yes | <yes/no> |
| **Comments** | Comments should be substantive; this means there should be at least 2-3 well-structured sentences with constructive feedback. | | | |