|  |  |
| --- | --- |
| **Project:** <Enter a project name> | Test Plan for Todo App |
| **Course:** <Enter the course number and name> | CSD380-DevOps |
| **Description:** <Enter a description> | Module 4, Assignment 4.2, Test Plan for the Todo APP |
| **Date:** <Enter a date (yyyy/mm/dd)> | 2024/09/05 |

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

.

**TABLE OF CONTENTS**

[**Delete a todo task item** 2](#_Toc176458699)

[**Submitting a task form without entering a todo item** 2](#_Toc176458700)

[**Submitting the edit form without entering a todo item** 3](#_Toc176458701)

[**Edit a todo task item** 3](#_Toc176458702)

[**Custom 404 error page for invalid URLs** 4](#_Toc176458703)

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 | **Delete a todo task item** | | | |
|  | **Test Objective:** Verifies a todo task is deleted when the “trashcan” icon is clicked. | **Developer: Ivan Pryymak**  **Date tested:** <2024/09/05> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Visit <https://buwebdev.github.io/todo/> | Todo app landing page | yes | <yes/no> |
| 2 | Confirm that there are items in the todo list | A list with todo task items | yes | <yes/no> |
| 3 | Click on the “trashcan” icon next to the todo task item | The task item disappears from the list | yes | <yes/no> |
| 4 | Verify that the todo task item is no longer in the todo list | The task item is no loner in the list | yes | <yes/no> |
| **Comments** | Deleting a todo task item functions as expected. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Submitting a task form without entering a todo item** | | | |
|  | **Test Objective:** Verifies a user is unable to submit a task form without entering a todo item | **Developer: Ivan Pryymak**  **Date tested:** <2024/09/05> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Visit <https://buwebdev.github.io/todo/> | Todo app landing page | yes | <yes/no> |
| 2 | Click in the textfield of the Task form | The textfield gains focus | Yes | <yes/no> |
| 3 | Enter between 0 and 3 characters in the textfield | The characters appear in the textfield, and the Add Task button remains inactive | Yes | <yes/no> |
| 4 | Click on Add Task Button | The task is not added to the todo list | Yes | <yes/no> |
| **Comments** | The app prevents users from submitting a task form without entering a todo item. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Submitting the edit form without entering a todo item** | | | |
|  | **Test Objective:** Verifies that users can not update a todo task item to an empty line | **Developer: Ivan Pryymak**  **Date tested:** <2024/09/05> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Visit <https://buwebdev.github.io/todo/> | Todo app landing page | yes | <yes/no> |
| 2 | Confirm that there are items in the todo list | A list with todo task items | yes | <yes/no> |
| 3 | Click on the “Pen” icon next to the todo task item | The edit form pops up | Yes | <yes/no> |
| 4 | Erase the task item description | Edit form with an empty task | Yes | <yes/no> |
| 5 | Click “Save” | Error message that says the task description can’t be empty | no | <yes/no> |
| **Comments** | The app allows users to edit todo task items to an empty line and save them. The code needs to be rewritten to ensure the app prevents users from doing that. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Edit a todo task item** | | | |
|  | **Test Objective:** Verifies that users are able to edit and save a task item | **Developer: Ivan Pryymak**  **Date tested:** <2024/09/05> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Visit <https://buwebdev.github.io/todo/> | Todo app landing page | yes | <yes/no> |
| 2 | Confirm that there are items in the todo list | A list with todo task items | yes | <yes/no> |
| 3 | Click on the “Pen” icon next to the todo task item | The edit form pops up | Yes | <yes/no> |
| 4 | Edit the task item description | Edit form with the task description | Yes | <yes/no> |
| 5 | Click “Save” | The Todo list shows the task with updated description | Yes | <yes/no> |
| **Comments** | The app allows users to edit todo task items. The functionality is as expected. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Custom 404 error page for invalid URLs** | | | |
|  | **Test Objective:** Verifies that users receive a custom 404 error page when they type in an invalid URL | **Developer: Ivan Pryymak**  **Date tested:** <2024/09/05> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Visit <https://buwebdev.github.io/todo/> | Todo app landing page | yes | <yes/no> |
| 2 | Edit the URL by appending any characters to the end of the URL (except for “index.html”) | A custom “Not-found” error page | yes | <yes/no> |
| **Comments** | The App correctly displays a custom “Not-found” error page. | | | |