

# Test Case Template

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

<b>Project:</b> Test Plan	
<b>Course:</b> CSD-380 – Julie Sakai	
<b>Description:</b> A test plan to confirm the functionalities of the Todo application.	
<b>Date:</b> (2025/04/12)	

## TABLE OF CONTENTS

<b>Accessing the Website .....</b>	<b>Error! Bookmark not defined.</b>
<b>Adding a Task .....</b>	<b>2</b>
<b>Editing a Task .....</b>	<b>3</b>
<b>Removing a Task.....</b>	<b>3</b>
<b>Handling Invalid URL .....</b>	<b>4</b>

# Test Case Template

Test 1	<b>Accessing the Website</b>			
	<b>Test Objective:</b> Ensure that the website is accessible through the URL.	<b>Developer:</b> Julie Sakai <b>Date tested:</b> 2025/04/12	<b>Peer tester:</b> Tazmin Somerville <b>Date tested:</b> 2025/04/13	
<b>Step</b>	<b>Action</b>	<b>Expected results:</b> 2025/04/12	<b>Developer pass/fail</b>	<b>Tester pass/fail + Screenshot</b>
1	Visit the URL “ <a href="https://buwebdev.github.io/todo/">https://buwebdev.github.io/todo/</a> ”.	The page loads without any errors.	yes	yes
2	Refresh the browser window.	Confirm that the page reloads without any errors.	yes	yes
3	Interact with the clickable buttons.	The page function works without issues.	yes	yes
<b>Comments</b>	Nothing unexpected or different from the developer's results			

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 2	<b>Adding a Task</b>			
	<b>Test Objective:</b> Ensure the functionality of adding new tasks to the ToList is validating input handling.	<b>Developer:</b> Julie Sakai <b>Date tested:</b> 2025/04/12	<b>Peer tester:</b> Tazmin Somerville <b>Date tested:</b> 2025/04/13	
<b>Step</b>	<b>Action</b>	<b>Expected results:</b>	<b>Developer pass/fail</b>	<b>Tester pass/fail</b>

## Test Case Template

1	Navigate to the task form	The task form appears with fields to add a new task.	yes	yes
2	Enter a valid task and select "Add Task"	The task is added to the Todo List.	yes	yes
3	Refresh the page after entering a task and select "Add Task".	The added task appears in the Todo List after the page is refreshed.	no	no
<b>Comments</b>	Nothing unexpected or different from the developer's results			

Test 3	<b>Editing a Task</b>			
	<b>Test Objective:</b> Ensure the functionality of editing existing tasks and save changes.	<b>Developer:</b> Julie Sakai <b>Date tested:</b> 2025/04/12	<b>Peer tester:</b> Tazmin Somerville <b>Date tested:</b> 2025/04/13	
<b>Step</b>	<b>Action</b>	<b>Expected results:</b>	<b>Developer pass/fail</b>	<b>Tester pass/fail</b>
1	Select an existing task in the Todo List and click the edit button.	The selected task will appear in the "Edit Form" with the task field below.	yes	yes
2	Modify the task title and select "save".	The updated task appears in the Todo List.	yes	yes
3	Refresh the page after editing a task.	The edited task updates the changes on the Todo List.	no	no
<b>Comments</b>				

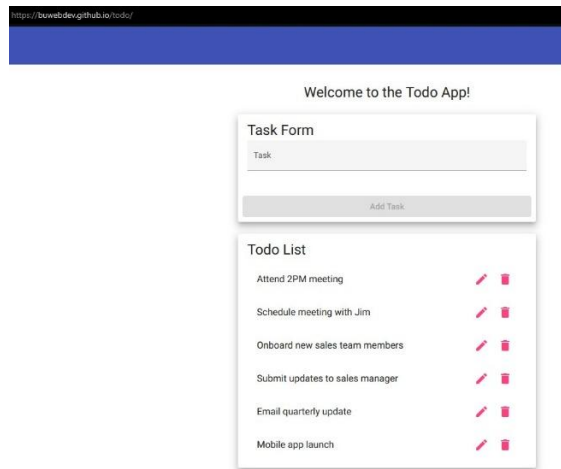
Test 4	<b>Deleting a Task</b>			
	<b>Test Objective:</b> Ensure the functionality of removing tasks for the Todo List.	<b>Developer:</b> Julie Sakai <b>Date tested:</b> 2025/04/12	<b>Peer tester:</b> Tazmin Somerville <b>Date tested:</b> 2025/04/13	
<b>Step</b>	<b>Action</b>	<b>Expected results:</b>	<b>Developer pass/fail</b>	<b>Tester pass/fail</b>

## Test Case Template

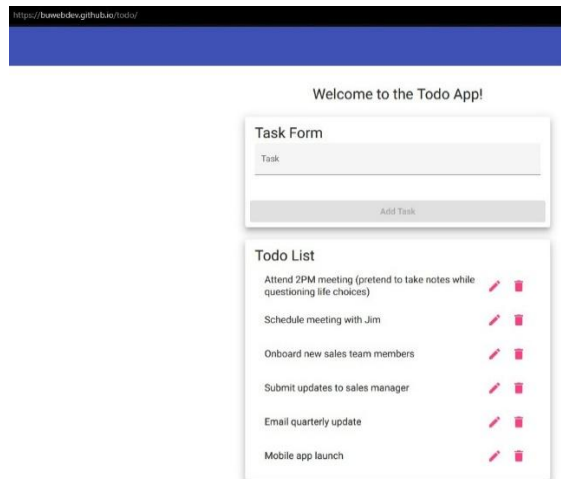
1	Select an existing task and click the delete button.	The selected task is removed from the Todo List.	yes	yes
2	Verify the Todo List after deleting a task.	The selected task is removed, and the other tasks remain on the Todo List.	yes	yes
3	Refresh the page after deleting a task.	The selected task to be deleted is removed and does not reappear in the Todo List.	no	yes
<b>Comments</b>				

Test 5	<b>Handling Invalid URL</b>			
	<b>Test Objective:</b> Ensure the invalid URL is displaying the 404 error page.	<b>Developer:</b> Julie Sakai <b>Date tested:</b> 2025/04/12	<b>Peer tester:</b> Tazmin Somerville <b>Date tested:</b> 2025/04/13	
<b>Step</b>	<b>Action</b>	<b>Expected results:</b>	<b>Developer pass/fail</b>	<b>Tester pass/fail</b>
<yes/no>	Enter the valid URL for the Todo List " <a href="https://buwebdev.github.io/todo/">https://buwebdev.github.io/todo/</a> ".	The page loads without any errors.	yes	yes
2	Enter the invalid URL for the Todo List " <a href="https://buwebdev.github.io/to/">https://buwebdev.github.io/to/</a> ".	The page displays a 404 error page.	yes	yes
<b>Comments</b>				

# Test Case Template

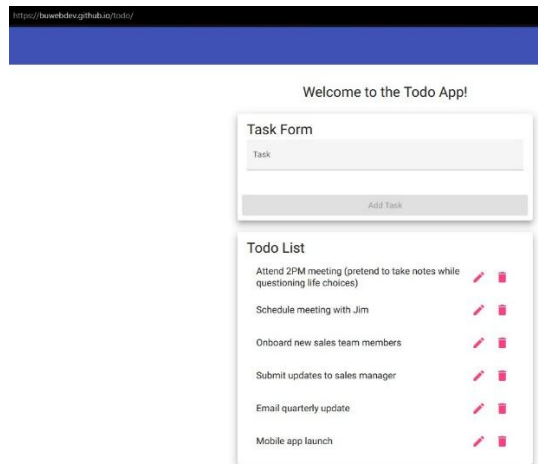


*Test 1 works without issues*

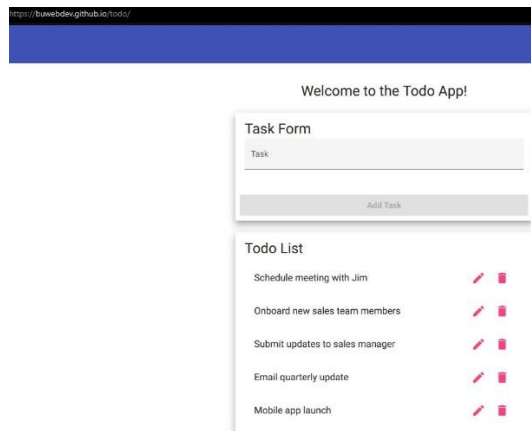


*Test 2 results same as Developers*

# Test Case Template



*Test 3 – Edited task performs as stated in test*



*Test 4 - Deletes as per Test case by Developer*

# Test Case Template

<https://buwebdev.github.io/to/>

# 404

There isn't a GitHub Pages site here.

If you're trying to publish one, [read the full documentation](#) to learn how to set up **GitHub Pages** for your repository, organization, or user account.

GitHub Status — @githubstatus



*Test 5 - 404 displays properly as stated by developer*