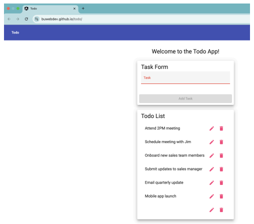
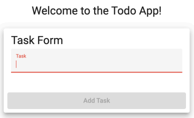
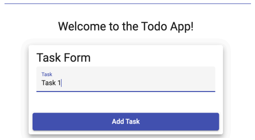
Project: Test Plan	
Course: CSD380 DevOps	
Description: Test plan based on the business requirements for the Todo Business Requirements website. Includes 5 functional tests that address the requirements.	
Date: 2025/01/28	

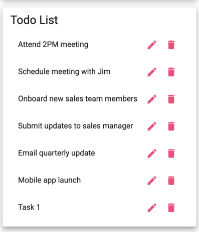
TABLE OF CONTENTS

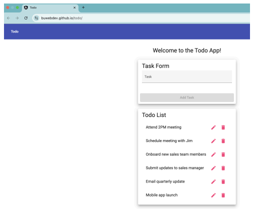
Test Case Template

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.

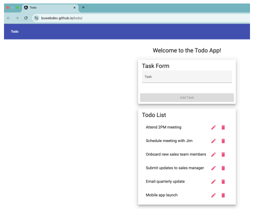
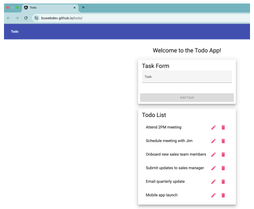
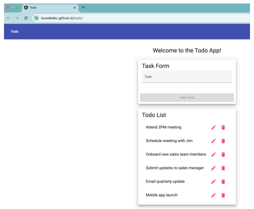
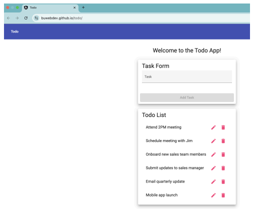
Test 1	**Create a new todo task item**			
	Test Objective: Verify that users are able to create a new task and that it is displayed within the task list.	**Developer:** **Date tested:** 2025/01/28	**Peer tester: Austen Rhyce Erickson** **Date tested:** 2/1/2025	
Step	**Action**	**Expected results:**	**Developer pass/fail**	**Tester pass/fail** **+ Screenshot**
1	Navigate to the Todo App homepage.	The web page loads correctly.	Yes	yes  buwebdev.github.io/todo/ Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
2	Locate the "Task Form" input field.	Page is functioning properly and that users can add new tasks.	Yes	yes  Welcome to the Todo App! Task Form
3	Enter a valid task name.	Users can add input text.	Yes	yes  Welcome to the Todo App! Task Form Task 1

Test Case Template

4	Click the “add task” button.	Users can submit text information.	Yes	yes
5	Verify that the new task is displayed in the list of tasks.	User input (tasks) are displayed in the Todo List.	Yes	yes  Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch Task 1
Comments	This test tests the main MVP of the app. It is the main central requirement and therefore an important test. I notice you test all the main core functional requirements of the app in this test suite – add, edit, delete. Nice work.			

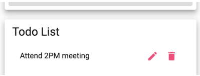
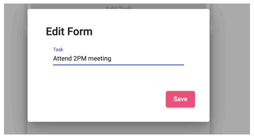
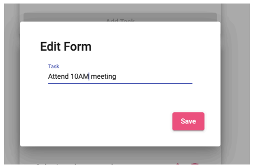
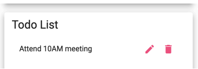
Test 2	**View a List of Todo Task Items**			
	Test Objective: Verify that users are able to view a list of existing tasks within the application.	**Developer:** **Date tested:** 2025/01/28	**Peer tester: Austen Rhyce Erickson** **Date tested:** 2/1/2025	
Step	**Action**	**Expected results:**	**Developer pass/fail**	**Tester pass/fail**
1	Navigate to the Todo homepage.	The web page should load properly.	Yes	yes  buwebdev.github.io/todo/ Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch

Test Case Template

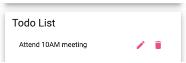
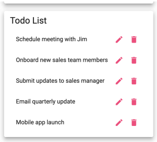
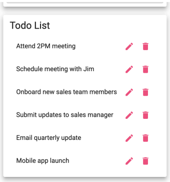
2	Confirm that the page loads and displays the task list.	The web page should automatically display the task list to users.	Yes	yes  Todo buwebdev.github.io/todo/ Todo Task Add Task Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
3	Check that each task is clearly visible and properly formatted on the webpage.	Each task should be properly formatted and visible.	Yes	yes  Todo buwebdev.github.io/todo/ Todo Task Add Task Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
4	Verify that the list contains the tasks that were previously created.	The page should display all existing tasks.	Yes	yes  Todo buwebdev.github.io/todo/ Todo Task Add Task Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
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  Todo buwebdev.github.io/todo/ Todo Task Add Task Welcome to the Todo App! Task Form Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch

Test Case Template

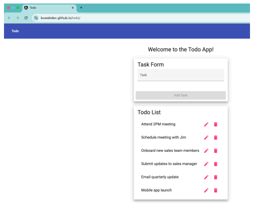
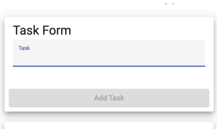
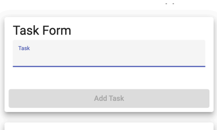
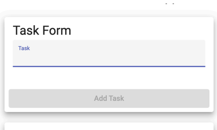
Comments	This test feels a little redundant. All the steps test more or less the same thing. I think this test could be taken down from 5 steps to 2 or 3.

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: Austen Rhyce Erickson** **Date tested:** 2/1/2025	
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  Todo List Attend 2PM meeting
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  Edit Form Task Attend 2PM meeting
3	Modify the task to have a different value.	Users should be able to edit task information.	Yes	 Edit Form Task Attend 10AM meeting
4	Click the “save” button.	Save button should be functional.	Yes	yes
5	Confirm that the updated item appears in the list, properly displaying the modified information.	The task item should be properly saved.	Yes	yes  Todo List Attend 10AM meeting
Comments	This test tests another core functionality and is important to include in a test suite for this app. Good work.			

Test Case Template

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: Austen Rhyce Erickson** **Date tested:** 2/1/2025	
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  Todo List Attend 10AM meeting
2	Locate the “Delete” icon on the task.	The trash icon should appear for users.	Yes	yes
3	Click the “Delete” icon on the task.	The user should be able to delete items by clicking this icon.	Yes	yes
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  Todo List Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
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  Todo List Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
Comments	I found a way that this test could pass when it should fail. Since this page returns to the same state after every refresh, any edited item that gets deleted will appear to have been successfully deleted because that exact item will not be there anymore.			

Test Case Template

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: Austen Rhyce Erickson** **Date tested:** 2/1/2025	
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  buwebdev.github.io/todo/ Welcome to the Todo App! Attend 2PM meeting Schedule meeting with Jim Onboard new sales team members Submit updates to sales manager Email quarterly update Mobile app launch
2	Locate the “Add Task” input field.	Users should be able to input text to create a new task.	Yes	yes 
3	Leave the input field blank.	Leaving the text field blank should not prompt the button for task submission to the user.	Yes	yes 
4	Click the submit button.	Users should not be able to submit empty fields.	Yes	yes
5	Confirm that the form is not submitted.	The form was not submitted.	Yes	yes 

Comments	Nice job testing another business requirement. I noticed that the “add task” button only activates after four characters are added. This kind of detail was not specified in the business requirements. I wonder how/why the developers decided to do it this way.