

| | |
|-------------------------------|--|
| Project: ToDo | |
| Course: CSD380 DevOps | |
| Description: Test Case | |
| Date: 2024/11/17 | |

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

.

TABLE OF CONTENTS

| | |
|----------------------------------------------|---|
| Verify Access to Website | 2 |
| Create a New ToDo Task Item | 2 |
| Edit an Existing ToDo Task Item | 3 |
| Delete a ToDo Task Item | 3 |
| Invalid URL custom 404 Error | 4 |

| | | | | |
|-----------------|----------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|--------------------------------------------------|--------------------------------------|
| Test 1 | Verify Access to Website | | | |
| | Test Objective: Verifies input data meets requirements | Developer: Date tested: 2024/11/17 | Peer tester: Date tested: <yyyy/mm/dd> | |
| Step | Action | Expected results: | Developer pass/fail | Tester pass/fail + Screenshot |
| 1 | Open a browser and navigate to url https://buwebdev.github.io/todo/ | Landing page loads successfully | Yes | <yes/no> |
| 2 | Refresh the page | The page loads without errors | Yes | <yes/no> |
| Comments | This test ensures that users can access the website without issues. It also verifies that page reloads function correctly. | | | |

| | | | | |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|--------------------------------------------------|-------------------------|
| Test 2 | Create a New ToDo Task Item | | | |
| | Test Objective: Verify the users can add a new task to the ToDo list. | Developer: Date tested: 2024/11/17 | Peer tester: Date tested: <yyyy/mm/dd> | |
| Step | Action | Expected results: | Developer pass/fail | Tester pass/fail |
| 1 | Click on the task field | The cursor appears, ready for input | yes | <yes/no> |
| 2 | Enter "Test Task 1" | Task is populated | yes | <yes/no> |
| 3 | Click "Submit" | The task appears in the list | yes | <yes/no> |
| Comments | This test ensures users can create and add new tasks successfully. Feedback should focus on task input handling and list updates. | | | |

| | | | | |
|-----------------|------------------------------------------------------------------------------------------------------------|-----------------------------------------------------|---------------------------------------------------------|-------------------------|
| Test 3 | Edit an Existing ToDo Task Item | | | |
| | Test Objective: : Verify the users can edit existing tasks. | Developer: Date tested: 2024/11/17 | Peer tester: Date tested: <yyyy/mm/dd> | |
| Step | Action | Expected results: | Developer pass/fail | Tester pass/fail |
| 1 | Click the “Edit” icon next to an existing task | The edit form appears | Yes | <yes/no> |
| 2 | Change the tasks text to “Updated Task” | The edit field updates | Yes | <yes/no> |
| 3 | Click “Save” | The list reflects the updated task | Yes | <yes/no> |
| Comments | This test validated that task editing is functional. Focus on form usability and task updates in the list. | | | |

| | | | | |
|-----------------|-------------------------------------------------------------------------------------------------|-----------------------------------------------------|---------------------------------------------------------|-------------------------|
| Test 4 | Delete a ToDo Task Item | | | |
| | Test Objective: Verify users can delete tasks from the list | Developer: Date tested: 2024/11/17 | Peer tester: Date tested: <yyyy/mm/dd> | |
| Step | Action | Expected results: | Developer pass/fail | Tester pass/fail |
| 1 | Add a new task called “Task to Delete” | The tasks appears on the list | Yes | <yes/no> |
| 2 | Click the “Delete” icon next to the task | The task is removed from the list | Yes | <yes/no> |
| Comments | Ensure task deletion functionality works without errors. Feedback should cover user experience. | | | |

| | | | | |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|--------------------------------------------------|-------------------------|
| Test 5 | Invalid URL custom 404 Error | | | |
| | Test Objective: Verify users are redirected to a custom 404 page when invalid URL | Developer: Date tested: 2024/11/17 | Peer tester: Date tested: <yyyy/mm/dd> | |
| Step | Action | Expected results: | Developer pass/fail | Tester pass/fail |
| 1 | Open a browser and navigate to https://buwebdev.github.io/todo/ | Landing page loads successfully | Yes | <yes/no> |
| 2 | Add random special character to URL https://buwebdev.github.io%^/todo/ | A custom 404 error page is displayed | Yes | <yes/no> |
| 3 | Add a URL with long invalid path such as https://buwebdev.github.io/todo/testforlongurl | A custom 404 error page is displayed | Yes | <yes/no> |
| Comments | Test is meant to ensure that invalid URL's are handled. Verify the desing and clarity of the 404 page. Test performance under different scenarios. | | | |