

DAY 5 - TESTING, ERROR HANDLING, AND BACKEND INTEGRATION

REFINEMENT

| Test Case ID | Test Case Description | Test Steps | Expected Result | Actual Result | Status | Severity Level | Assigned To | Remarks | |
|--------------|---------------------------------------|--|---|---------------|--------|----------------|-------------|---------|-----------------------------------|
| TC001 | Validate cart items fetching from API | Load the cart page > Fetch cart items from API | Cart items should load successfully | | TBD | TBD | High | - | Check if the API request succeeds |
| TC002 | Handle cart fetch failure | Simulate API failure > Verify error handling | Error message should be logged in console | | TBD | TBD | Medium | - | Ensure UI does not break |
| TC003 | Validate item removal from cart | Remove an item > Verify API call and cart update | Item should be removed from the cart successfully | | TBD | TBD | High | - | Ensure UI updates correctly |
| TC004 | Handle item removal failure | Simulate API failure on delete request | Error should be logged and handled gracefully | | TBD | TBD | Medium | - | Check for console error logs |

Test Case 2: API Error Handling in Cart Actions

| Test Case ID | Test Case Description | Test Steps | Expected Result | Actual Result | Status | Severity Level | Assigned To | Remarks |
|--------------|--|---|--|---------------|--------|----------------|-------------|--|
| TC005 | Handle API failure when fetching cart items | 1. Simulate API failure (disconnect API). 2. Reload the page. 3. Verify error handling. | console.error("Error fetching cart items:", error) should appear in console. | TBD | TBD | High | - | Ensure proper fallback UI |
| TC006 | Handle API failure when removing an item from cart | 1. Add an item to the cart. 2. Simulate API failure when removing the item. 3. Verify error handling. | console.error("Error removing item from cart:", error) should be logged. | TBD | TBD | High | - | Ensure item is not removed from UI if API call fails |

Test Case 3: Cart Data Parsing in LocalStorage

| Test Case ID | Test Case Description | Test Steps | Expected Result | Actual Result | Status | Severity Level | Assigned To | Remarks |
|--------------|--|--|--|---------------|--------|----------------|-------------|------------------------------|
| TC007 | Validate cart data retrieval and parsing from localStorage | 1. Add products to the cart. 2. Refresh the page. 3. Check if the cart items are loaded correctly. | Cart items should be retrieved and set in state successfully. | TBD | TBD | Medium | - | TBD |
| TC008 | Handle corrupted cart data in localStorage | 1. Manually modify the cart data in localStorage to an invalid JSON format. 2. Refresh the page. 3. Check if an error is logged. | console.error("Error parsing cart data:", error) should be displayed in the console. | TBD | TBD | High | - | Fix JSON typo before testing |

For the "Actual Result" column, I wrote "TBD" (To Be Determined) because you haven't provided actual test results yet. Once you run the test cases, you can replace "TBD" with the actual outcome, such as:

- "Passed" (if everything works as expected)
- "Failed - API request failed" (if there's an issue)
- "Failed - UI did not update correctly" (if the cart UI doesn't reflect changes)