**Task 2:**

* Console Tab Analysis: The browser's console records no JavaScript errors, warnings, or any activity related to the change event. This indicates the field's change event is either not being captured or has no JavaScript function attached to it.
* Network Tab Analysis: Upon attempting to edit the quantity of an item in the cart, no new network requests (API calls) are triggered. This includes no calls to an expected endpoint that would communicate the change to the backend server.

The absence of both network activity and console errors suggests that this is a front-end functional defect. The application's user interface presents an editable field, but the underlying JavaScript code is missing the necessary event listeners to capture the user's input and send that updated data to the backend via an API call.

**Task 3:**

## **Title:** Cart Icon Not Fixed in Header

**Severity:** Medium  
**Component:** Navigation/Header

**Description:** The Cart icon scrolls out of view when navigating down the page due to non-fixed header positioning.

**Step To Reproduce:**1. Open the app saucedemo.com

2. Scroll down the page and observe the cart icon

**Expected Result:** Cart icon remains visible and accessible at all times via fixed header positioning.

**Actual Result:** Cart icon disappears from view during page scrolling, requiring users to scroll back to the top for cart access.

## **Title:** Missing Quantity Controls in Cart

**Severity:** High  
**Component:** Shopping Cart

**Description:** The Cart page lacks quantity modification functionality for added products.

**Expected Result:** Users can adjust item quantities directly in the cart using increment/decrement controls or input fields.

**Actual Result:** No quantity editing options available; users cannot modify product quantities once items are added to the cart.

**Impact:** Both issues negatively affect checkout workflow efficiency and overall user experience.

**Task 4:**

# **Test Cases for Cart Feature**

## **Test Case 1: Verify Cart Item Display and Information**

**Objective:** Ensure all cart items display correctly with complete product information

**Test Steps:**

* Add multiple different products to the cart
* Navigate to the cart page
* Verify each item shows: quantity, product name, description, price, and the Remove button
* Confirm the cart icon shows the correct item count

**Expected Result:** All products display with accurate information, proper formatting, and functional Remove buttons

**Test Case 2: Verify Remove Item Functionality**

**Objective:** Confirm users can successfully remove individual items from the cart

**Test Steps:**

* Add multiple items to the cart
* Click the "Remove" button for one specific item
* Verify the item is deleted from the cart
* Confirm cart total and item count update accordingly
* Verify other items remain unaffected

**Expected Result:** Selected item is removed, cart updates correctly, and remaining items stay intact

**Test Case 3: Verify Empty Cart State**

**Objective:** Test cart behavior when all items are removed

**Test Steps:**

* Add items to cart
* Remove all items one by one using the Remove buttons
* Observe the cart page state when empty
* Verify cart icon badge updates to show zero items

**Expected Result:** Cart displays appropriate empty state message, cart badge shows 0 or disappears

## **Test Case 4: Verify Cart Data Persistence Across Navigation**

**Objective:** Ensure cart contents persist when navigating between pages

**Test Steps:**

* Add items to the cart from product pages
* Navigate to different pages (home, product details, etc.)
* Return to cart page
* Verify all previously added items remain in the cart with correct quantities and details

**Expected Result:** Cart contents remain unchanged across page navigation, maintaining data integrity

**Task 6:**

## **Missing Quantity Controls in Cart**

This is likely a **frontend-related** issue based on the visual evidence from the cart page. The cart displays a static "QTY" column header with hardcoded quantity values of "1" for each item, but lacks any interactive controls (input fields, +/- buttons, or dropdowns) that would allow users to modify quantities. Since the basic cart functionality (displaying items, Remove buttons) works correctly, this appears to be a UI/UX implementation gap rather than a backend data handling problem.