

## Bug Report 1: Cart allows increasing quantity beyond available stock

### Description

While the Categories page correctly prevents users from adding more items when the stock is zero or insufficient, the Cart page still allows users to increase the product quantity beyond the available stock. This results in unrealistic cart values and violates the stock control logic.

### Steps to Reproduce

1. Go to **Categories** page.
2. Select a product that is **out of stock** or has low stock.
3. Attempt to add it — UI correctly prevents adding more.
4. Now go to the **Cart** page.
5. Increase the item quantity using the **+ button**.
6. Cart allows quantities that exceed the actual stock.

### Actual Behavior

- The Cart page allows users to increase product quantity **even when the stock is 0 or insufficient**.
- This results in values like **30 units** even when the product stock is 0

### Expected Behavior

- The Cart page should **validate stock limits** just like the Categories page.
- Quantity increase should stop when:
  - Stock = 0
  - OR current quantity = stock amount
- A warning or disabled “+” button should appear when stock limit is reached.

### Severity

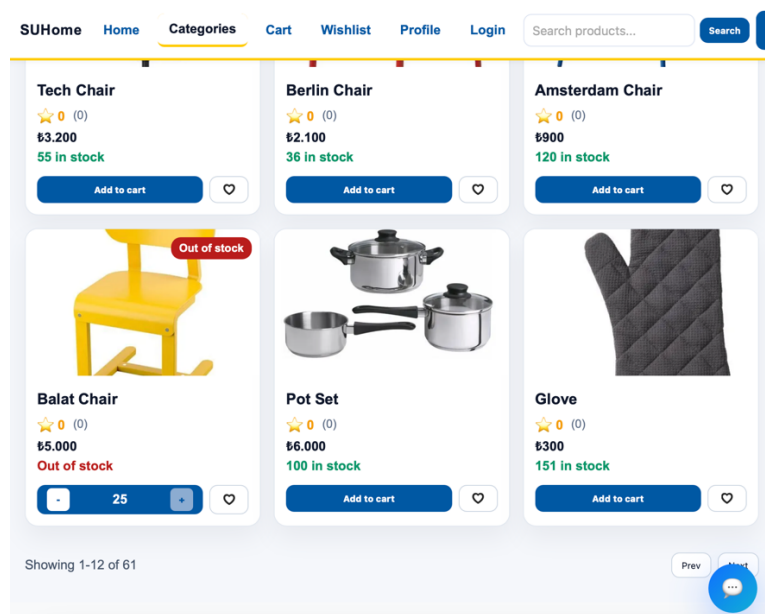
#### High

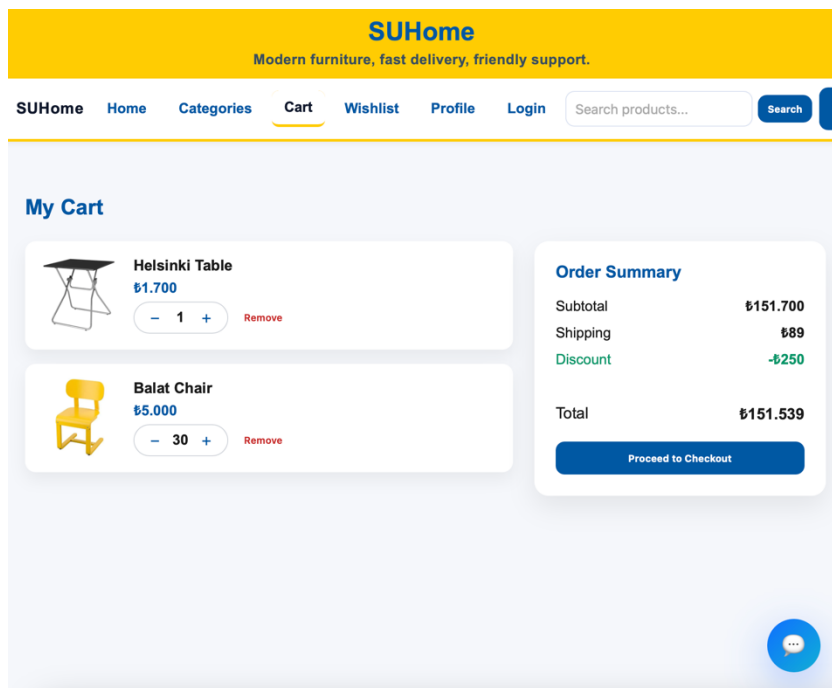
This bug affects:

- Ordering accuracy
- Stock integrity
- Business logic consistency
- User experience

And may cause backend errors later during checkout.

### Screenshot





### Possible Causes

- Cart UI does not check product.stock before incrementing.
- Backend API for updating cart may not validate stock.
- Missing condition in the increment logic:

Possible missing check:

```
if (quantity < product.stock) {  
    setQuantity(quantity + 1);  
}
```

## Bug Report 2: Product detail page fails to load

### Description

When clicking on any product card, the product detail page does not load and an error message is displayed.

### Steps to Reproduce

1. Go to **Categories** page.
2. Open any product list.
3. Click on a product.
4. Error page appears instead of product details.

### Expected Behavior

The product detail page should load normally and show:

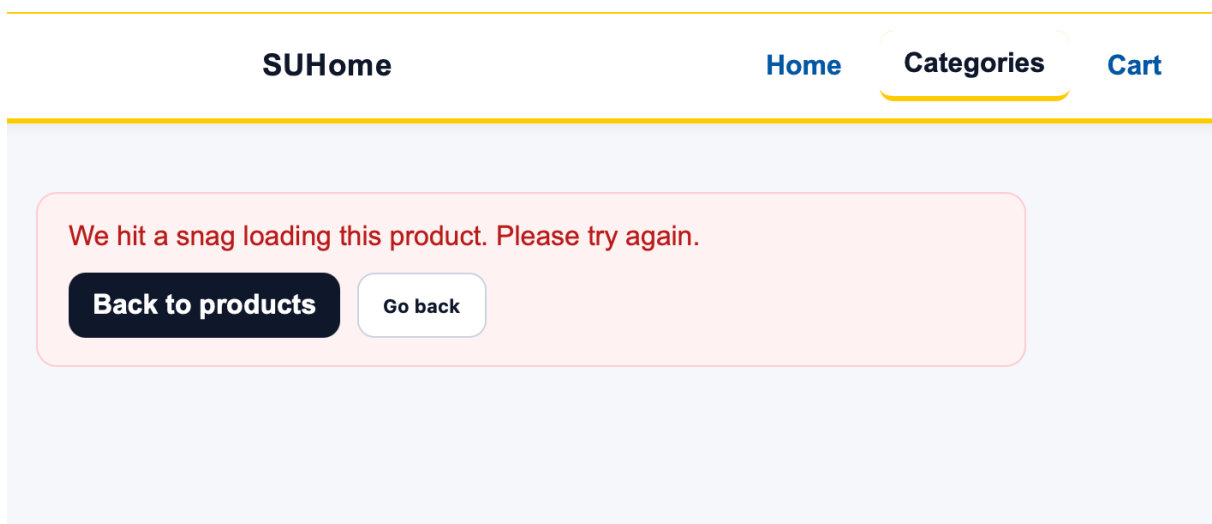
- Product name
- Description
- Images
- Price
- Stock quantity
- Ratings & reviews

### Actual Behavior

The following error message is shown:

*"We hit a snag loading this product. Please try again."*

No product data is displayed.



## Bug Report 3: Clearing search input does not reset product list

### Description

When the user searches for a keyword (e.g., “bed”), the correct filtered products are shown.

However, when the user clears the search input (removes the text), the product list **does NOT reset** back to the full catalog.

The UI continues showing only the previously filtered items.

### Steps to Reproduce

1. Navigate to **Categories** page.
2. Type “bed” in the search bar.
3. Click the **Search** button → filtered results appear correctly.
4. Now delete the text in the search input (make it empty).
5. **Expected:** All products should be shown again.
6. **Actual:** Only the filtered products (“bed” results) are still shown.

### Actual Behavior

- The product list remains filtered even after clearing the search text.
- UI does not refresh or reload the full product list.

### Expected Behavior

When the search input becomes empty:

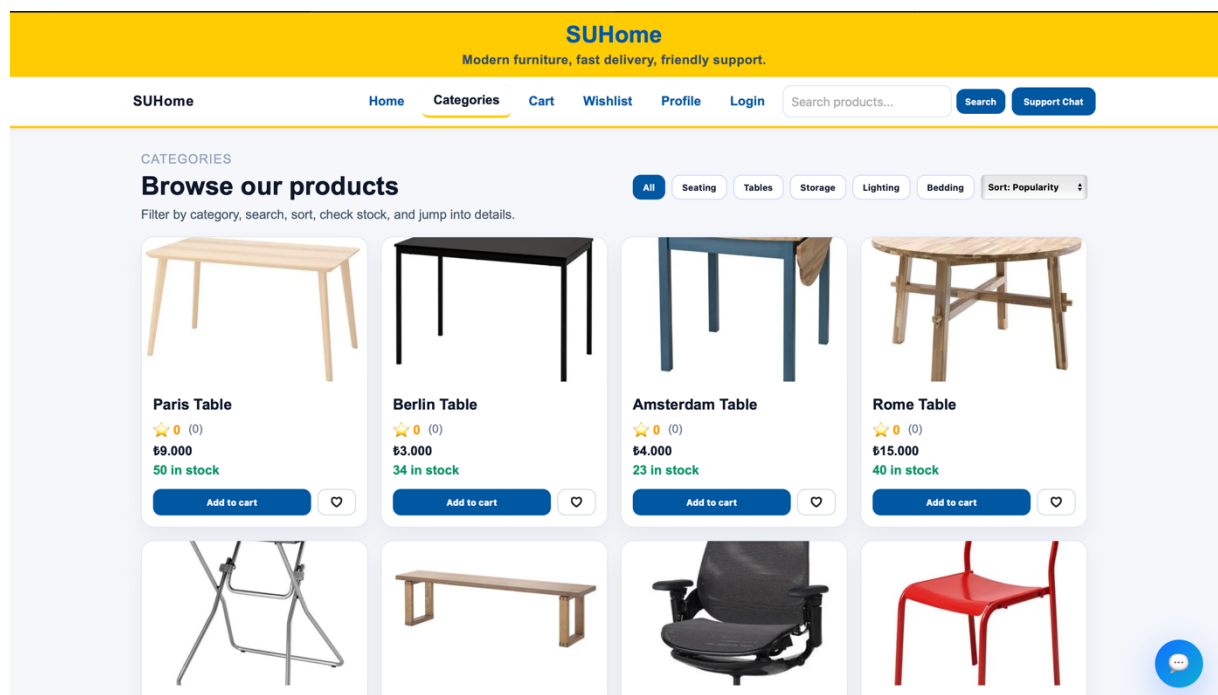
The system should automatically reload all products

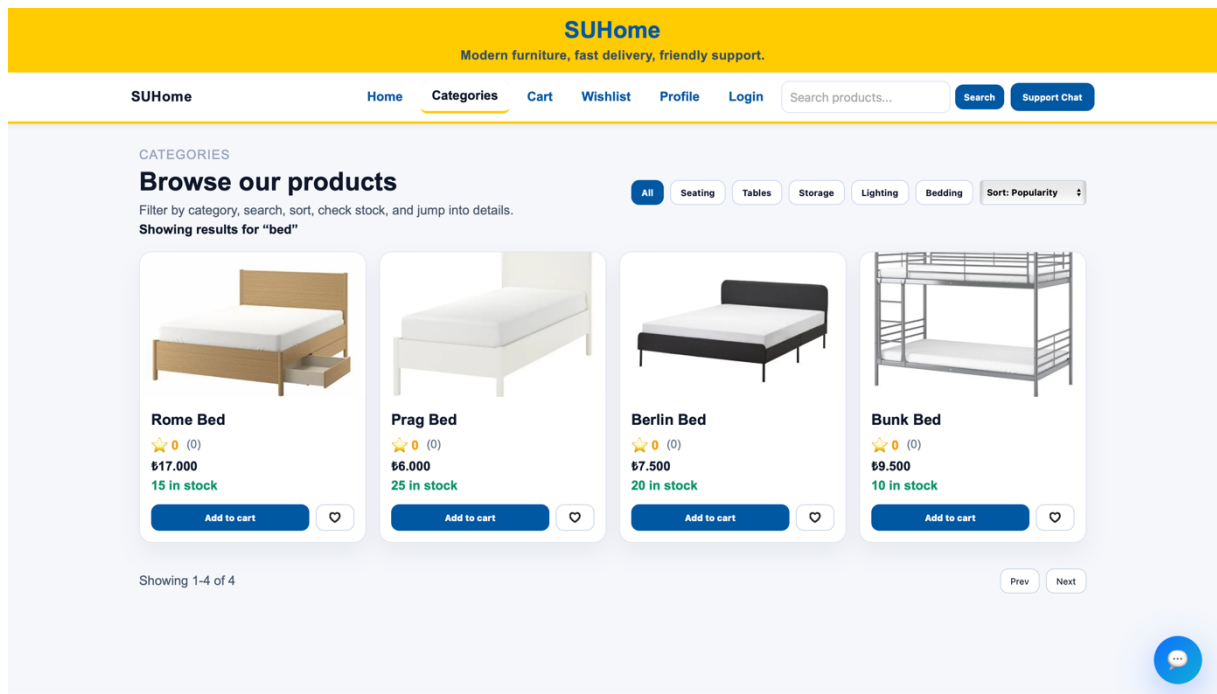
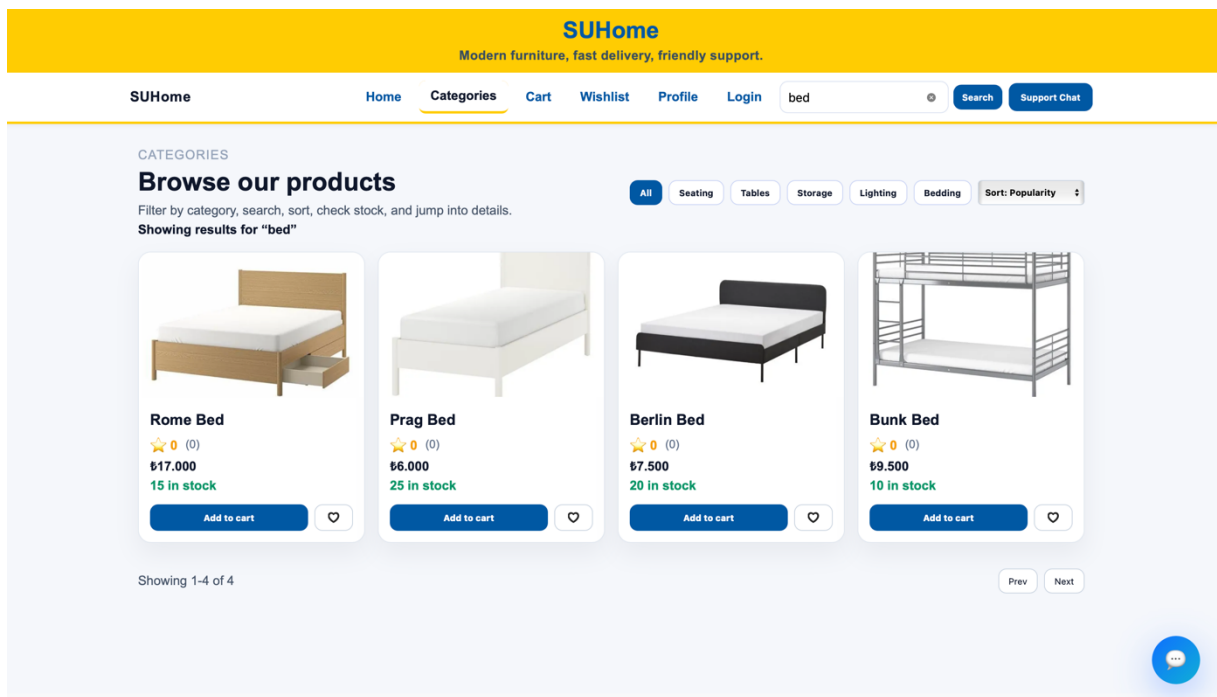
**OR**

Search button should reset the product list to the default state

The user should see the full product catalog again.

### Screenshots





## Severity

## Medium to High

Reason:

- Breaks the search UX
- Traps the user in a filtered state
- Makes it impossible to return to normal browsing unless they refresh the whole page

## Possible Causes

Likely in search logic:

### Missing condition:

```
if (searchTerm === "") {
  setProducts(allProducts);
}
```

```
return;  
}
```

**Or:**

- API is only called when clicking “Search” and not when input is emptied
- State variable for "filteredProducts" is not reset
- Search component does not trigger refresh on change

## Bug Report 4: Cart quantity doubles after login during checkout

### Summary

When a user adds items to the cart as a guest and proceeds to checkout, the cart quantity unexpectedly increases after the user logs in. The number of items in the cart becomes higher than what the guest originally added.

---

### Description

The cart state is not being handled consistently between guest mode and logged-in mode.

Items added before login are duplicated or merged incorrectly once the user signs in at the checkout step. As a result, the quantity shown in the cart after logging in is different from the quantity that the guest user selected.

This leads to incorrect order totals and inconsistent checkout behavior.

---

### Steps to Reproduce

1. Log out (ensure you are in guest mode).
  2. Add **1 item** (e.g., Paris Table) to the cart.
  3. Go to **Cart** → quantity is **1** (correct).
  4. Click **Proceed to Checkout** → redirected to Login (expected).
  5. Log in.
  6. After login, you are redirected back to Checkout/Cart.
  7. Observe that the cart now shows **2 items** instead of 1.
- 

### Expected Result

- The cart should preserve the exact quantity the guest user selected.
  - After login, cart should remain **1 item**, not 2.
  - Cart should not merge, duplicate, or re-add items unless explicitly chosen by the user.
- 

### Actual Result

- Logging in causes the cart to **double the quantity of the item** (1 → 2).
  - Checkout subtotal and totals become incorrect.
  - This may cause users to unintentionally purchase more items than intended.
- 

### Severity

#### High

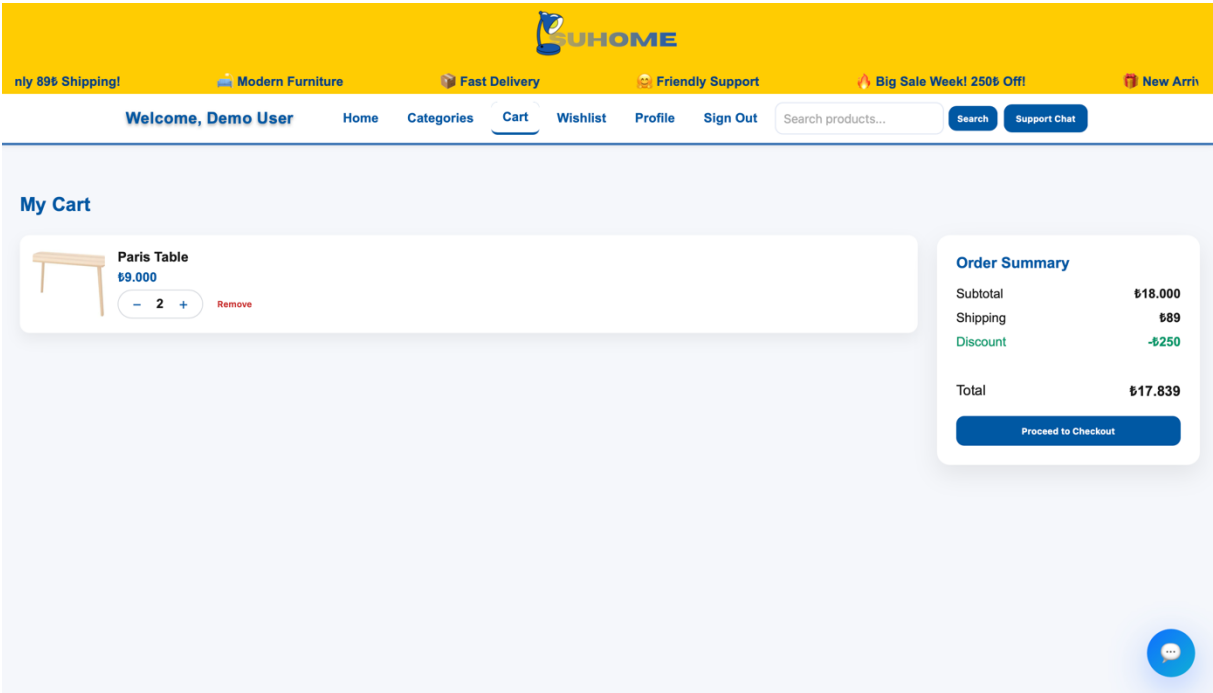
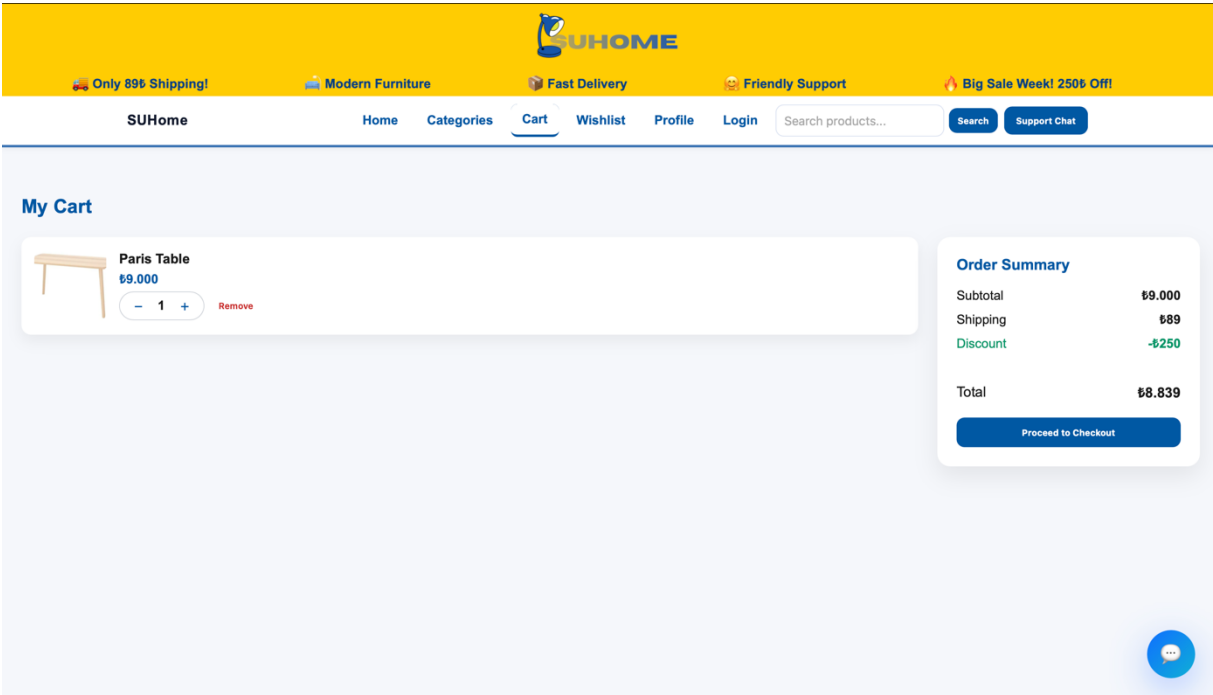
This bug affects the checkout process and directly impacts the correctness of orders.

---

### Possible Cause

- Cart state is stored both in **frontend (local state)** and **backend/localStorage**, and both are being merged on login.
- When login occurs, the system:
  - Restores a saved cart **AND**
  - Preserves the guest cart → causing duplication
- Cart is initialized twice: once from guest state, once from user profile.

Screenshots





## **Bug Report 5: Stock decreases when adding to cart instead of after completing purchase**

### **Summary**

Stock decreases immediately when a user adds an item to the cart, even though the project specification requires stock reduction to occur only *after the purchase is completed*.

---

### **Description**

According to the project requirements, stock should be deducted when the shopping process is completed (after payment), not when items are added to the cart.

However, in the current implementation, stock is reduced as soon as the user clicks “Add to Cart”.

If the user abandons the cart or removes the item later, the stock remains incorrectly lowered.

This violates the expected workflow and leads to inaccurate stock tracking.

---

### **Steps to Reproduce**

1. Open the Products page.
2. Add any product to the cart.
3. Observe that the product’s stock decreases immediately.
4. Remove the item from the cart or leave checkout.
5. Stock does *not* return to its original value.

---

### **Expected Behavior**

- Stock should only decrease **after** the user completes the checkout and the order is successfully placed.
- Adding items to the cart should not impact inventory levels.
- Cart removal or abandonment should not modify stock.

---

### **Actual Behavior**

- Stock drops immediately when an item is added to the cart.
- Users can reduce stock without purchasing the item.
- Inventory becomes inaccurate and does not reflect confirmed orders.

---

### **Severity**

**High** — This affects essential business logic and breaks order → delivery flow.

---

### **Possible Cause**

- Stock is updated using `updateStock(id, - 1)` inside `handleAddFirst()` and `handleIncrease()`.
- Cart operations are incorrectly tied to inventory updates.

---

### **Suggested Fix**

- Remove stock updates from `ProductList.jsx` cart functions.
- Deduct stock **only inside Checkout → handleSubmit**, once payment is processed.