Lecture Topic Task: #313 Smart UI vs Layered Architecture (This example will be for marking for Sold)

Assignee: Daniella Melero

Objective: Implement a small feature in two ways to compare the architecture of both styles:

- Smart UI: All logic inside the component
- Layered: Core logic extracted into a helper file.

Feature: When the seller clicks “Mark as Sold”, the item changes it status from available to sold.

The rules:

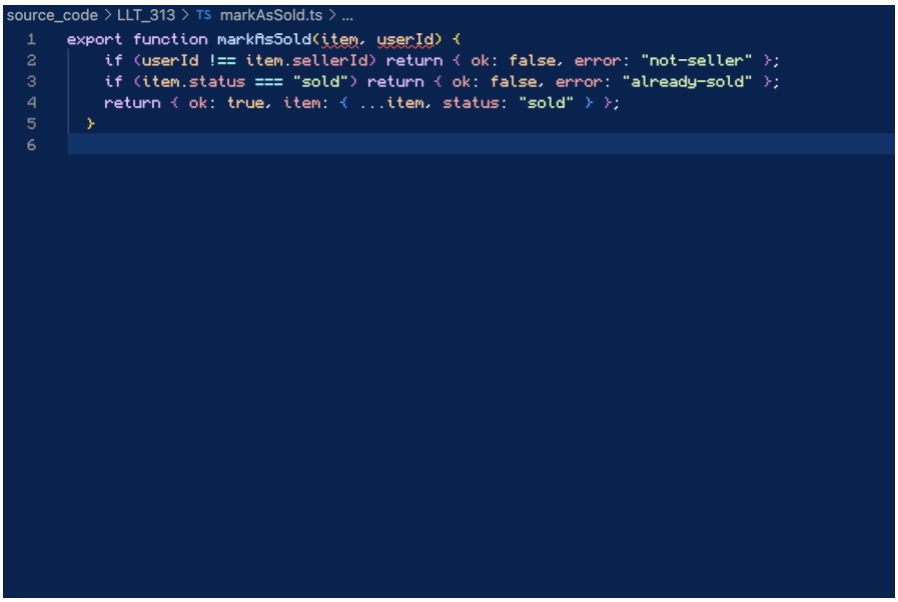
- Item starts as available
- Only the seller can mark the item as Sold
- Once sold, it cannot be sold again UNLESS Cancelled order
- No database, it is a simulated item object

Smart UI Version(SmartUIMarkAsSold.tsx)

```tsx
"use client";
import { useState } from "react";

export function SmartUIMarkAsSold({ currentUserId }: { currentUserId: string }) {
  const [item, setItem] = useState({
    title: "Vintage College Hoodie",
    status: "available",
    sellerId: "user-123",
  });

  const isSeller = currentUserId === item.sellerId;
  const isSold = item.status === "sold";

  const handleMarkAsSold = () => {
    if (!isSeller) return alert("Only the seller can mark this item as sold.");
    if (isSold) return;
    setItem({ ...item, status: "sold" });
  };

  return (
    <div>
      <p>Status: <strong>{item.status}</strong></p>
      <button onClick={handleMarkAsSold} disabled={!isSeller || isSold}>
        {isSold ? "Sold" : "Mark as Sold"}
      </button>
    </div>
  );
}
```

Layered Version

Logic Helper(markAsSold.ts)



UI using the helper

Comparison Table for demonstration

Approach	Best For	Pros	Cons
Smart UI	Small features	Fast to write and everything in one file	Harder to reuse and to test
Layered	Growing projects	Reusable Logic, easier for testing	Requires extra files

It is better layered because it is reusable and update can follow anywhere where the feature is implemented.