Project Documentation: Store Manager – Keep Track of Inventory

1. Introduction

Project Title: Store Manager – Keep Track of Inventory

Team ID:NM2035TMID36732

Team Leader Id: SOFIYA.M(sofiyamanikandan2007@gmail.com)

S.no	Team Members	Mail id
1	SOFIYA.M	sofiyamanikandan2007@gmail.com
2	SANGEETHA.M	sangeethamurugan01@gmail.com
3	SAKTHI.M	msakthi6126@gmail.com
4	SUMITHRA.E	sumithraelumalai172@gmail.com

2. Project Overview

Purpose:

This project is designed to help small business owners efficiently manage inventory. The application allows adding, updating, deleting, and viewing product stock levels in real-time.

Features: - Add new products to the inventory

- Edit product details (price, quantity)
- Delete products
- Track current stock levels
- Search products by name
- View product details in a tabular format

3. Architecture

Component Structure: - App: Main component wrapping the entire application

- Header: Displays application title and navigation
- ProductList: Displays a list of all inventory items
- ProductItem: Shows individual product data and action buttons (edit, delete)
- ProductForm: Form for adding or editing a product

- SearchBar: Allows filtering products by name

- Footer: Footer of the app

State Management: - Global State:

- Used React Context API to manage inventory data globally

- Inventory state contains an array of products:

js const [products, setProducts] = useState([]);

- Local State:

- Controlled form states in ProductForm using useState for fields (name, price, quantity)

Routing:

- Used react-router-dom for navigation between: - / → Product list page

- /add → Add new product

- /edit/:id → Edit existing product

4. Setup Instructions

Prerequisites:

- Node.js installed

- npm installed

Installation: bash # Clone repository git clone https://github.com/your-repo/store-manager.git

Navigate to project folder cd store-manager

Install dependencies npm install

Start development server npm start

5. Folder Structure

store-manager/

```
|
├── public/
├── src/
|   ├── components/
|   |   ├── Header.js
|   |   ├── Footer.js
|   |   ├── ProductList.js
|   |   ├── ProductItem.js
|   |   ├── ProductForm.js
|   |   └── SearchBar.js
|   |
|   ├── context/
|   |   └── InventoryContext.js
|   |
|   ├── pages/
|   |   ├── Home.js
|   |   ├── AddProduct.js
|   |   └── EditProduct.js
|   |
|   ├── utils/
|   |   └── helperFunctions.js
|   |
|   ├── App.js
|   └── index.js
|
├── package.json
└── README.md
```

6. Running the Application

Run in terminal: bash npm start

7. Component Documentation

Key Components: - ProductList: Displays a table of products.

- ProductItem: Displays product name, price, quantity, and edit/delete buttons.

- ProductForm: Handles adding and editing product data.

Reusable Components: - Header, Footer, and SearchBar are reusable across multiple pages.

8. State Management

Global State: Inventory is stored in Context API.

Example: js const { products, setProducts } = useContext(InventoryContext);

Local State: Each form field uses useState to control input values.

9. User Interface

The UI contains: - Product list page with a search bar and table

- Add/Edit product forms

- Responsive design

(Screenshots should be inserted here.)

10. Styling

Used plain CSS (or specify if using Styled-Components, Sass, etc.)

No third-party frameworks used.

Theming: Basic light theme implemented with consistent colors and fonts.

11. Testing

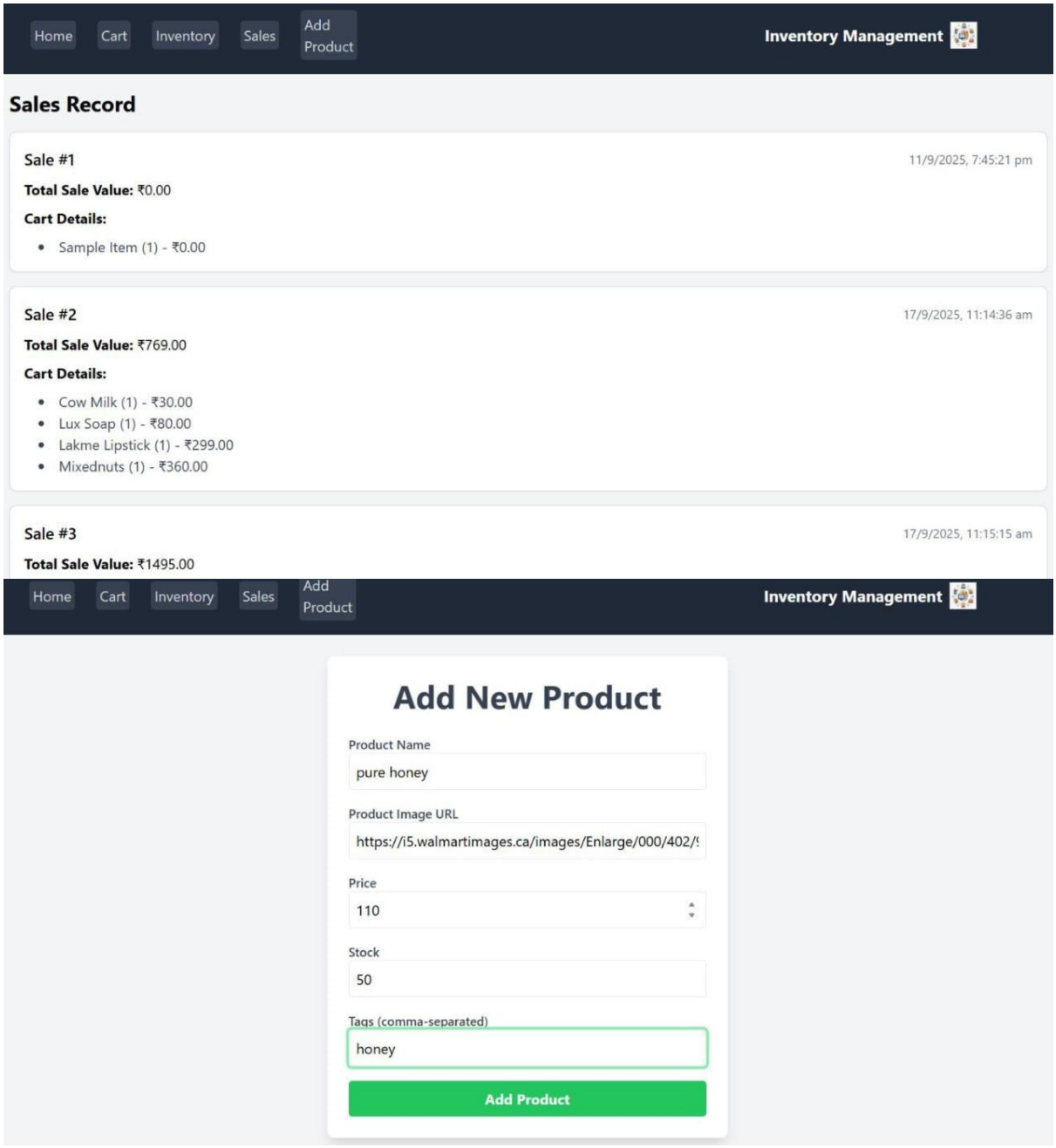
Testing Strategy: Used Jest and React Testing Library for unit tests. Example test:

```
js test('renders product list correctly', () => { render(<ProductList />);
  expect(screen.getByText('Product Name')).toBeInTheDocument(); });
```

Code Coverage:

Configured Jest to provide coverage reports.

12. Screenshots or Demo


Home
Cart
Inventory
Sales
Add Product
Inventory Management
Sales Record
Sale #1
11/9/2025, 7:45:21 pm
Total Sale Value: ₹0.00
Cart Details:
Sample Item (1) - ₹0.00
Sale #2
17/9/2025, 11:14:36 am
Total Sale Value: ₹769.00
Cart Details:
Cow Milk (1) - ₹30.00
Lux Soap (1) - ₹80.00
Lakme Lipstick (1) - ₹299.00
Mixednuts (1) - ₹360.00
Sale #3
17/9/2025, 11:15:15 am
Total Sale Value: ₹1495.00
Home
Cart
Inventory
Sales
Add Product
Inventory Management
Add New Product
Product Name
pure honey
Product Image URL
https://i5.walmartimages.ca/images/Enlarge/000/402/
Price
110
Stock
50
Tags (comma-separated)
honey
Add Product

Home Cart Inventory Sales Add Product **Inventory Management**

Product Catalog

Search products...

Sample Item

Price: ₹ 0.00

Add to Cart

Dairy Milk Oreo

Price: ₹ 375.00

Add to Cart

Cow Milk

Price: ₹ 30.00

Add to Cart

Lux Soap

Price: ₹ 80.00

Add to Cart

localhost:3000/inventory Sign in

Price: ₹ 0.00

Stock Available: 10

Add Stock: 6

Update Stock

Price: ₹ 375.00

Stock Available: 82

Add Stock: 30

Update Stock

Price: ₹ 30.00

Stock Available: 43

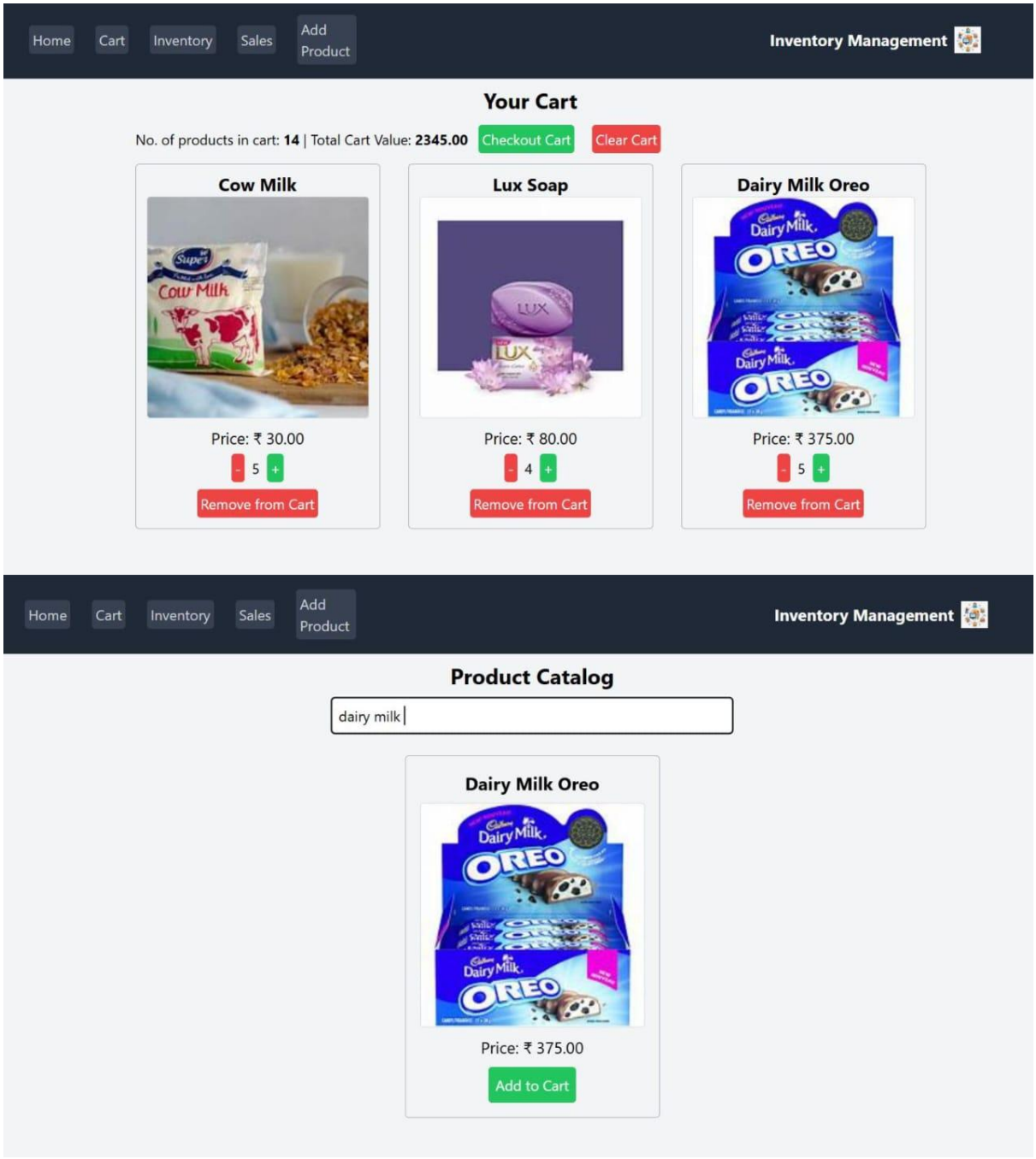
Add Stock: 20

Update Stock

Lux Soap

Mixednuts

Lakme Lipstick



13. Known Issues

Edit page doesn’t validate empty fields

Search performance drops when data exceeds 1000 products

14. Future Enhancements

Implement pagination for large inventories

Add authentication for multiple users

Add animations when products are added or removed

Integrate backend API for persistent data storage