

Project Documentation

1. Introduction

Project Title: Store Manager (Frontend)

Team ID: —Team ID : NM2025TMID39051

Team Leader : NUVERA HASEEN A

Email:24bsccs43@cttewc.edu.in

Team member : DHANUSHREE G

Email:24bsccs47@cttewc.edu.in

Team member : AFRIN ASIMA R

Email:24bsccs10@cttewc.edu.in

Team member : RINOSHA

Email: 24bsccs17@cttewc.edu.in

2. Project Overview

Purpose:

This project helps manage products, stock, and sales through a simple web-based system.

Features:

Product listing with images, price, and stock

Add/remove products to/from cart

Track sales using global state (Context API)

Dashboard for quick insights

3. Architecture

Frontend: React.js + Tailwind CSS

Backend: (Not included – frontend only project)

State Management: React Context API

Testing: Jest + React Testing Library

4. Setup Instructions

Prerequisites:

Node.js & npm

Visual Studio Code

Installation Steps:

Clone repository

git clone <repo-link>

Install dependencies

cd inventory-management-frontend

npm install

Start development server

npm start

5. Folder Structure

inventory-management-frontend/

- | -- public/ # Static files (index.html, images)
- | -- src/
 - | | -- App.js # Main app file
 - | | -- index.js # Entry point
 - | | -- Layout.jsx # Common layout
 - | | -- components/ # Reusable UI components (Cart, ProductList, etc.)
 - | | -- pages/ # Pages (Dashboard, Inventory, Sales)
 - | | -- context/ # Global state (SalesContext, InventoryContext)
 - | | -- hooks/ # Custom hooks
 - | | -- styles/ # Tailwind & CSS
- | -- package.json # Dependencies & scripts
- | -- tailwind.config.js # Tailwind configuration

6. Running the Application

Start frontend:

cd inventory-management-frontend

npm start

Access: <http://localhost:3000>

7. Features Explanation

Products: View product list with stock details

Cart: Add/remove products using Cart component

Sales: SalesContext stores and tracks sales globally

Dashboard: Displays inventory and sales overview

8. Authentication

Authentication is not included in this version.

It can be added in the future to provide secure access for different user roles such as Admin, Manager, or Staff.

9. User Interface

Dashboard Page

Product List Page

Cart Page

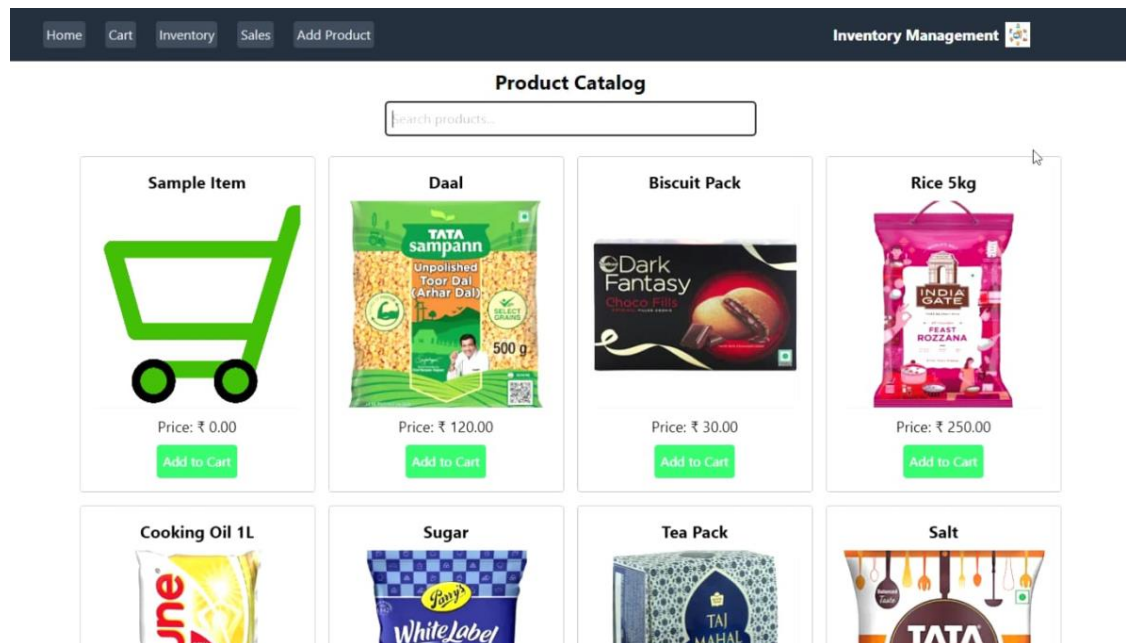
Sales Page

10. Testing

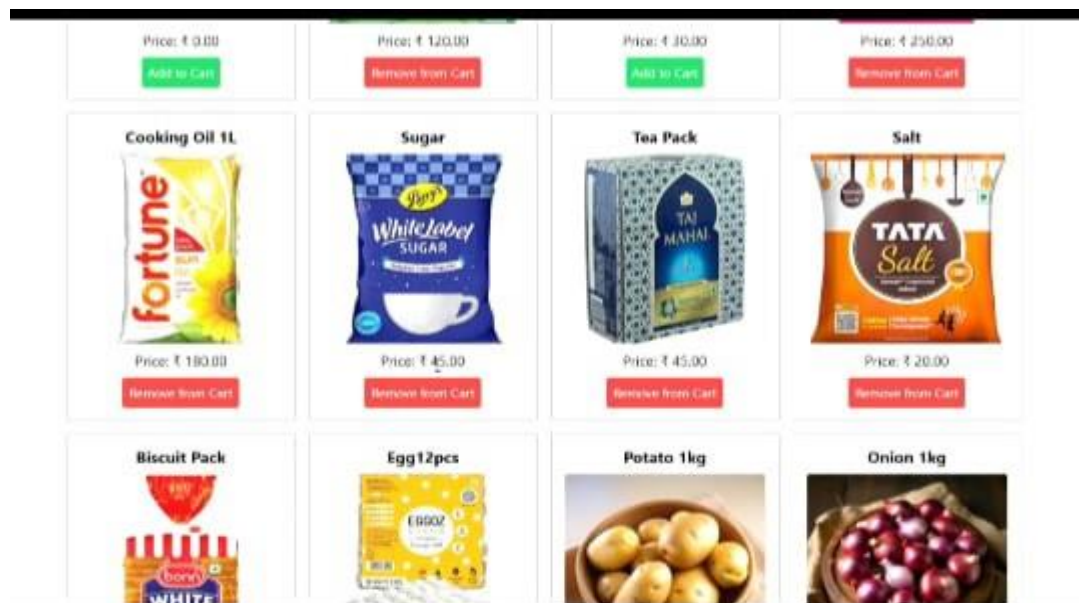
Tools: Jest, React Testing Library, Chrome DevTools

Approach: Manual and automated component tests

11. Screenshots / Demo



(Add screenshots of your Dashboard, Product List, and Cart UI)



HomeCartInventorySalesAdd Product

Inventory Management


Your Cart

No. of products in cart: 6 | Total Cart Value: 660.00

Checkout Cart

Clear Cart

Daal



Price: ₹ 120.00


1

+

-

Remove from Cart

Rice 5kg



Price: ₹ 250.00


1

+

-

Remove from Cart

Tea Pack



Price: ₹ 45.00


1

+

-

Remove from Cart

Salt



Price: ₹ 20.00


1

+


-

Remove from Cart

Sugar



Cooking Oil 1L



Sale #4		06/04/2025 13:29:15
Total Sale Value: ₹175.00		
Cart Details:		
<ul style="list-style-type: none">Rice 5kg (1) - ₹250.00Daal (1) - ₹120.00Sugar (1) - ₹45.00Cooking Oil 1L (1) - ₹190.00		
Sale #5		06/04/2025 13:30:01
Total Sale Value: ₹90.00		
Cart Details:		
<ul style="list-style-type: none">Rice 5kg (1) - ₹250.00Potato 1kg (1) - ₹30.00		
Sale #6		06/04/2025 14:00:11
Total Sale Value: ₹660.00		
Cart Details:		
<ul style="list-style-type: none">Daal (1) - ₹120.00Rice 5kg (1) - ₹250.00Tea Pack (1) - ₹45.00Salt (1) - ₹20.00Sugar (1) - ₹45.00Cooking Oil 1L (1) - ₹190.00		

12. Known Issues

No backend API integration (frontend only).

Data not persistent (lost on refresh).

13. Future Enhancements

Add backend with Node.js + MongoDB

Authentication (Login/Register)

Persistent storage of sales and inventory

Admin panel for inventory management