STORE MANAGER

KEEP TRACK OF INVENTORY

1. Introduction

- **Project Title** : Store manager : Keep track of inventory
- **Team ID** : NM2025TMID35058
- **Team Leader** : KARTHIKA R

  (karthikar6324@gmail.com)

- **Team Members :**

  — PARAMESHWARI V

  (parameshvenkat2007@gmail.com)

  — BRINDHA S

  (sbrindha935@gmail.com)

  — JANANI P

  (janani567janu@gmail.com)

2. Project Overview

- **Purpose :** Store manager Works is a freelancing platform designed to connect clients and freelancers. The platform facilitates project postings, a bidding system for freelancers, and real time communication to streamline collaboration.
- **Key Features:**
  - Project posting and bidding system
  - Secure, real-time chat functionality
  - Feedback and review system for completed projects
  - Admin control panel for platform management

3. System Architecture

- **Frontend: React.js**, styled with Bootstrap and Material UI.
- **Backend** : Node.js with the Express.js framework, managing server logic and API endpoints
- **Database** : MongoDB is used for storing user data, project information, applications , and chat messages.

4. Setup Instructions

- **Pre requistes :** Before you begin, ensure you have the following installed:
  - Node.js
  - MongoDB
  - Git
  - Visual Studio Code (or another code editor)
- **Installation Steps :**
  - **Clone the repository :**

    Git clone [repository_url]

  - **Install client dependencies :**

    cd sb-works /client

    npm install

  - **Install Server dependencies :**

    cd .. / server

    npm install

5. Folder Structure :

The project is organized into a client-side and a server-side directory.

```
SB –Works /
|-- client /                    # React frontend
|    |--components /
|    ╰--pages /
╰-- server /                    # Node.js backend
     |-- modules /
     |-- routes /
     ╰-- controllers /
```

6. Running the Application :

To run the application , you need to start both the frontend and backend servers.

- **Frontend (from the client directory) :**

  npm start

- **Backend (from the server directory) :**

  npm start

- **Access :**

  Once both servers are running , you can access the application at

  http://localhost:3000

7. API Documentation

- **Product Management :**
  - ➢ POST /api/products (to add a new product)
  - ➢ GET /api/products (to get a list of all products)
  - ➢ PUT /api/products/ :id (to update product details like price or stock)
  - ➢ DELETE /api/products/ :id (to remove a product)
- **Inventory Management :**
  - ➢ POST /api/inventory/receive (to add stock for a product)
- **Order Management :**
  - ➢ GET /api/orders (to view all orders)
  - ➢ PUT /api/orders /:id (to update an order's status, eg., "shipped")
- **Customer Management :**
  - ➢ GET /api/customers (to see a list of cystomers)

8. Authentication

The application uses **JWT (JSON Web Token)** for authentication. This ensures secure login and protects private routes using middleware.

9. User Interface

- **Dashboard:** Sales, Products, Revenue, Stock alert
- **Manage :** Products, Orders, Customers, Categories
- **Reports :** Sales and Stock insight with charts

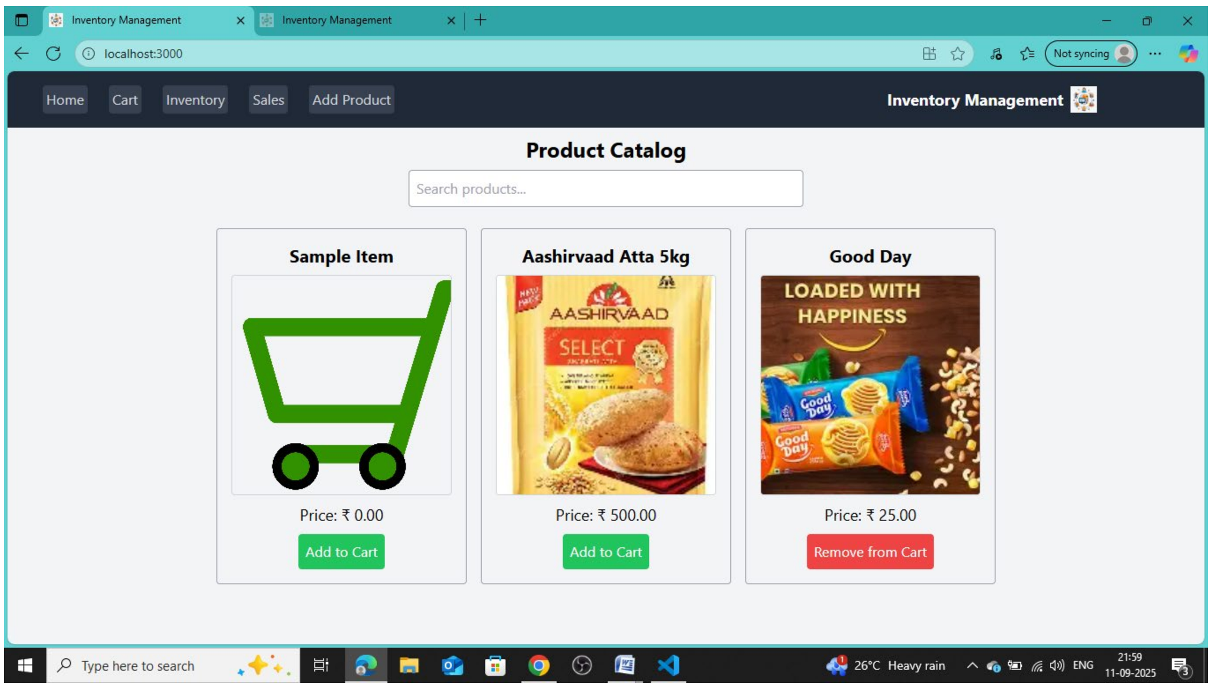
10. Testing

Manual testing was performed during project milestones. The following tools were used for debugging and API verification.

- Postman
- Chrome Dev Tools

11. Screenshots

- **Home page**



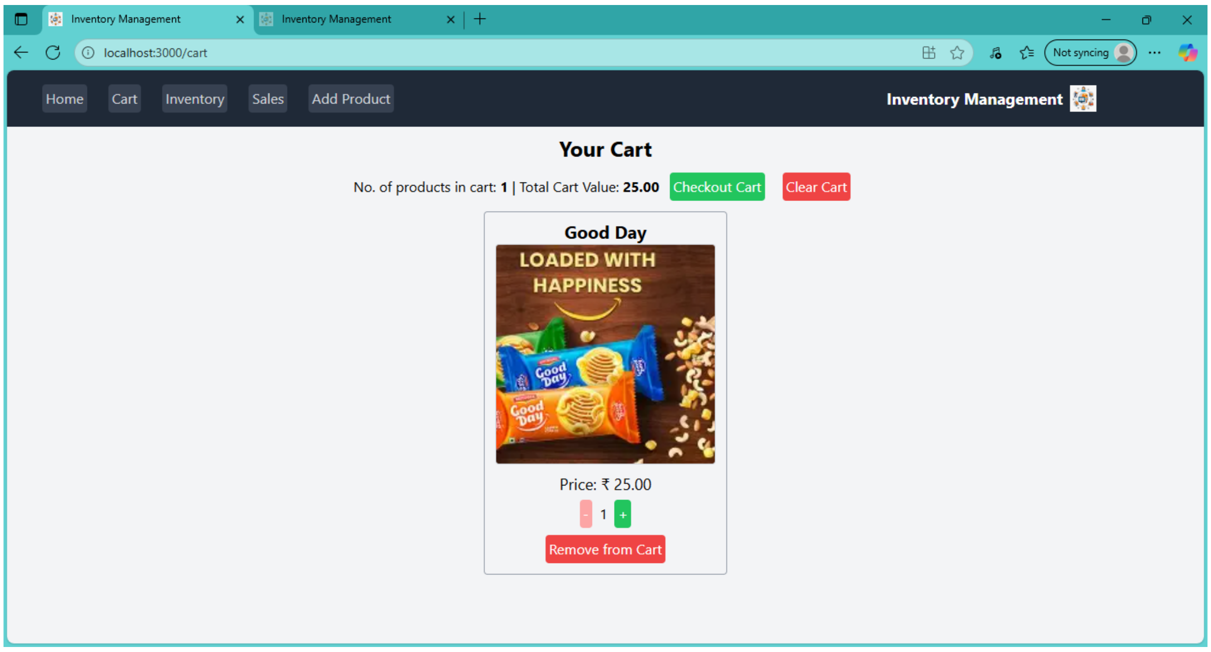
Product Catalog

Sample Item
Price: ₹ 0.00

Aashirvaad Atta 5kg
Price: ₹ 500.00

Good Day
Price: ₹ 25.00

- **Cart**



Inventory Management | Inventory Management

localhost:3000/cart

Home | Cart | Inventory | Sales | Add Product

Inventory Management

Your Cart

No. of products in cart: **1** | Total Cart Value: **25.00** Checkout Cart Clear Cart

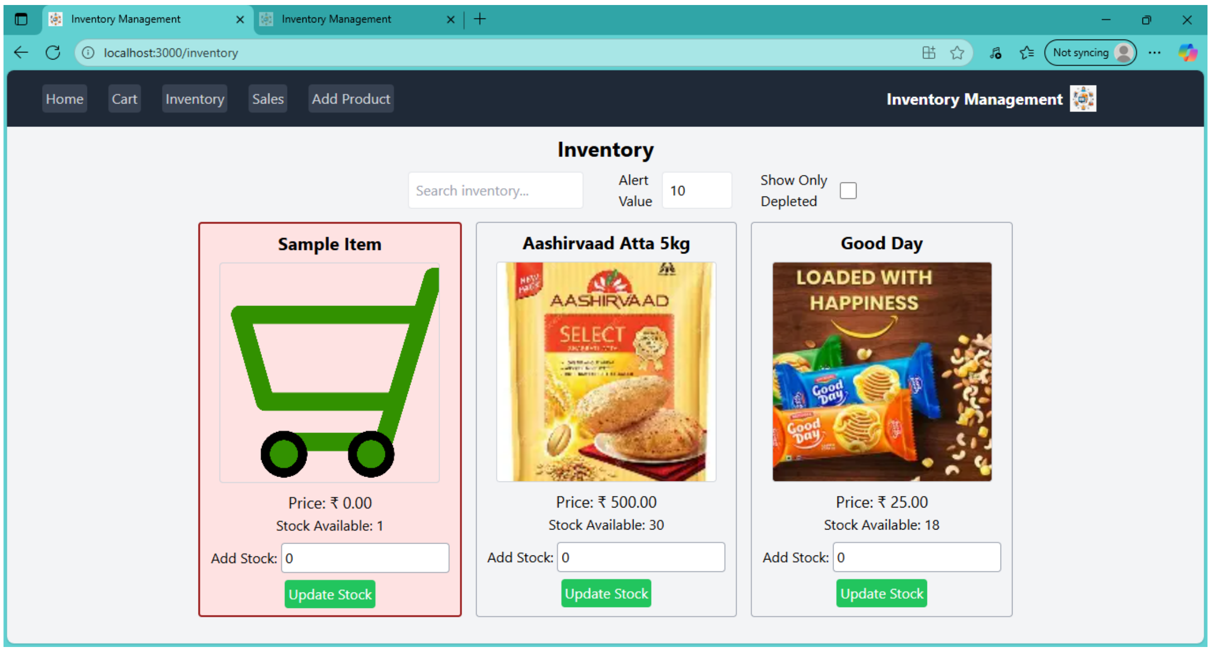
Good Day

Price: ₹ 25.00

- 1 +

Remove from Cart

- **Inventory**



Inventory Management | Inventory Management

localhost:3000/inventory

Home | Cart | Inventory | Sales | Add Product

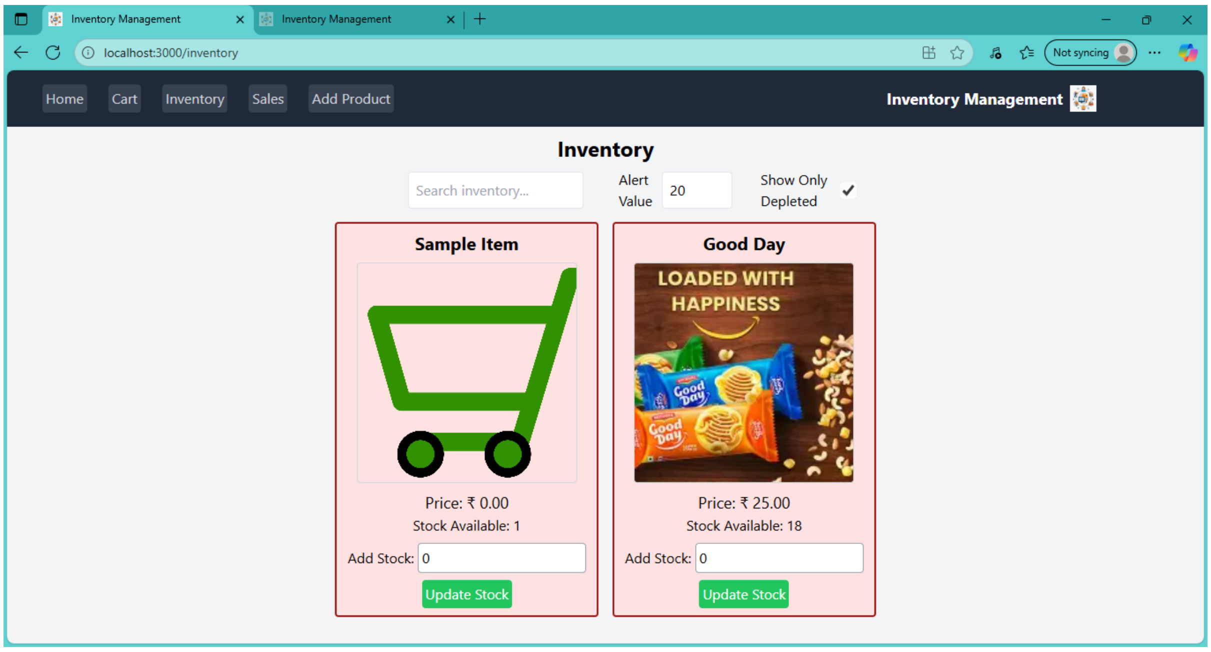
Inventory Management

Inventory

Search inventory... Alert Value 10 Show Only Depleted

Sample Item	**Aashirvaad Atta 5kg**	**Good Day**
Price: ₹ 0.00	Price: ₹ 500.00	Price: ₹ 25.00
Stock Available: 1	Stock Available: 30	Stock Available: 18
Add Stock: 0	Add Stock: 0	Add Stock: 0
Update Stock	Update Stock	Update Stock

- **Inventory (Show only Depleted)**



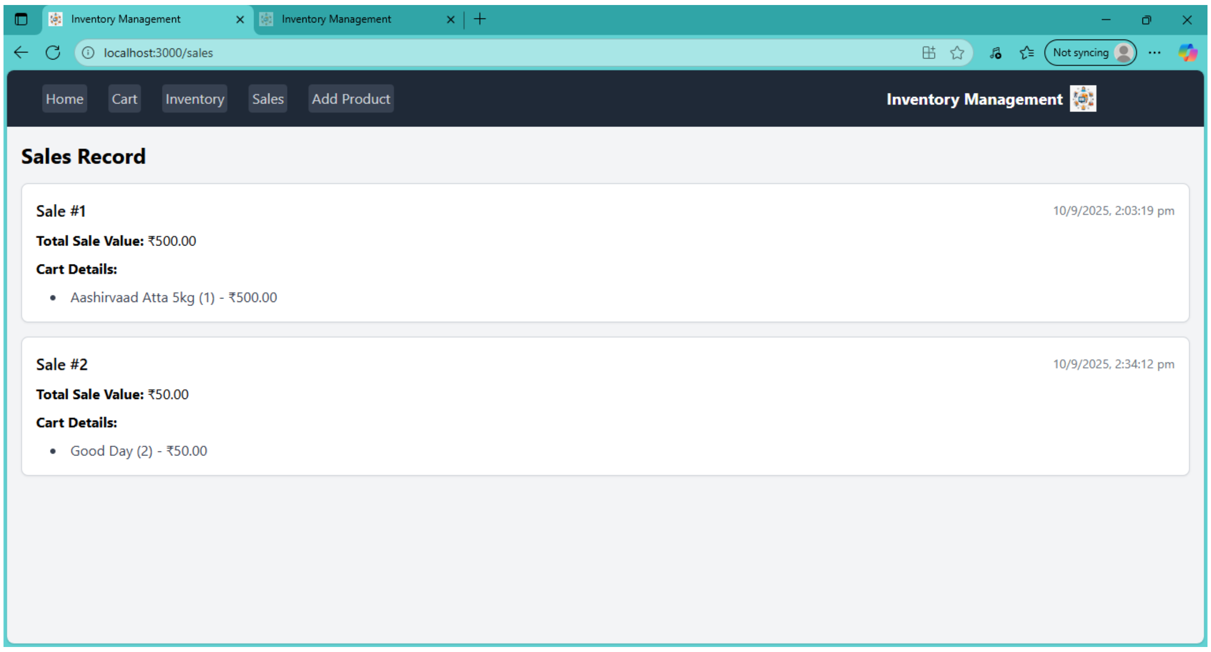
Inventory

Search inventory... Alert Value 20 Show Only Depleted ✓

Sample Item
Price: ₹ 0.00
Stock Available: 1
Add Stock: 0
Update Stock

Good Day
LOADED WITH HAPPINESS
Price: ₹ 25.00
Stock Available: 18
Add Stock: 0
Update Stock

- **Sales Record**

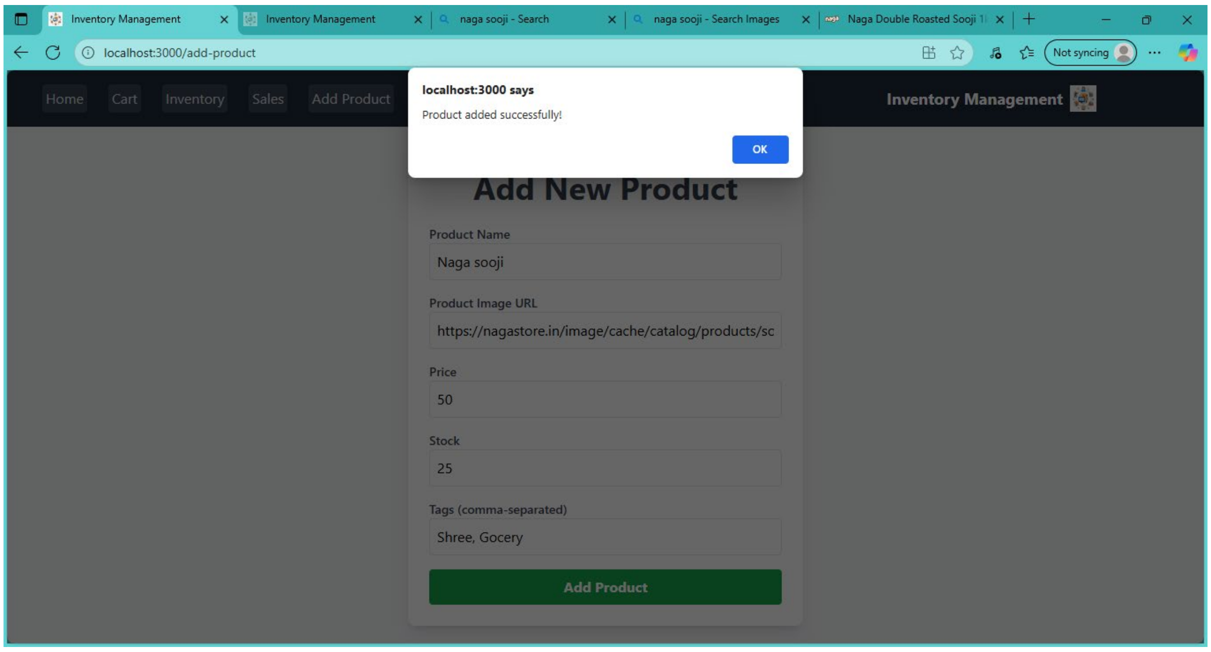


Sales Record

Sale #1 — 10/9/2025, 2:03:19 pm
Total Sale Value: ₹500.00
Cart Details:
- Aashirvaad Atta 5kg (1) - ₹500.00

Sale #2 — 10/9/2025, 2:34:12 pm
Total Sale Value: ₹50.00
Cart Details:
- Good Day (2) - ₹50.00

- **Add Product**


Inventory Management | Inventory Management | naga sooji - Search | naga sooji - Search Images | Naga Double Roasted Sooji 1k

localhost:3000/add-product

localhost:3000 says

Product added successfully!

Add New Product

Product Name: Naga sooji

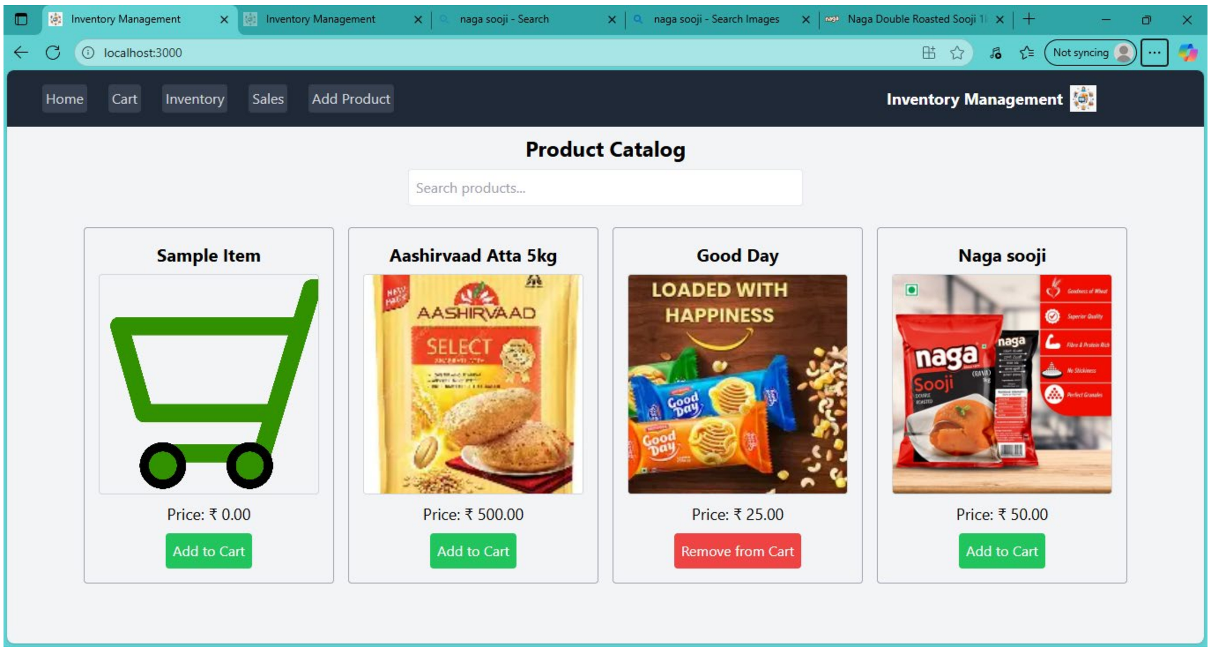
Product Image URL: https://nagastore.in/image/cache/catalog/products/sc

Price: 50

Stock: 25

Tags (comma-separated): Shree, Gocery

- **After Adding Product**


localhost:3000

Product Catalog

Search products...

Sample Item — Price: ₹ 0.00 — Add to Cart

Aashirvaad Atta 5kg — Price: ₹ 500.00 — Add to Cart

Good Day — Price: ₹ 25.00 — Remove from Cart

Naga sooji — Price: ₹ 50.00 — Add to Cart

12. Known Issues

- API Mismatch
- Data persistence Limitations
- Lack of Real-time Updates
- No input Validation
- Limitation Scalability

13. Future Enhancements

- Mobile support
- Advanced analytics for users and admin
- AI-drive project recommendations.