ShopSmart – Project Report

# Team Id : LTVIP2025TMID56240

## Project name : ShopSmart

# Team Members:

• Bhavagna – Full Stack Developer: Manages end-to-end development, authentication systems, and deployment.  
• Ampa Chaitanya Naga Durga Ganesh – Frontend Developer: Responsible for React.js UI components, user experience design, and client-side functionality.  
• Boda Prasanna Venkata Siva Kumar – Backend Developer: Handles Node.js/Express.js server development, API creation, and database integration.  
• Bolla Geethika Naga Sai Sridevi – Database Administrator: Designs MongoDB schemas, optimizes queries, and manages data relationships

# 1. INTRODUCTION

## 1.1 Project Overview

ShopSmart is a full-stack grocery web application designed to streamline the online shopping experience for users. It allows users to browse, search, and purchase grocery items online with ease, while providing administrators with tools to manage inventory, users, and reports.

## 1.2 Purpose

The purpose of ShopSmart is to digitize the traditional grocery shopping process, making it more efficient, accessible, and user-friendly. The project aims to serve a wide audience, including homemakers, students, and working professionals, by offering a secure and convenient e-commerce platform.

# 2. IDEATION PHASE

## 2.1 Problem Statement

Many people find it difficult to shop for groceries due to time constraints or lack of access to quality stores. Existing solutions are often not user-friendly or lack features like real-time stock updates, detailed product filtering, and personalized user dashboards.

## 2.2 Empathy Map Canvas

SAYS: “I need fresh items fast.”, “Is this product in stock?”

THINKS: “I hope my data is secure.”, “Is this price reasonable?”

DOES: Searches online, Compares prices

FEELS: Frustrated with app crashes, Happy when checkout is seamless

## 2.3 Brainstorming

Secure login system, Product search & filters, Admin dashboard, Inventory & order management, Multiple payment options, User-friendly interface, Mobile responsiveness

# 3. REQUIREMENT ANALYSIS

## 3.1 Customer Journey map

Search: Searches for products – Easy navigation needed  
Cart: Adds items to cart – Wants smooth UX  
Checkout: Enters payment & address – Needs secure payment  
Post-order: Tracks delivery – Expects timely updates

## 3.2 Solution Requirement

User login/signup, Product listing and filtering, Shopping cart, Admin controls, Order tracking, Payment gateway integration

## 3.3 Data Flow Diagram

User → [Frontend] → [Backend Server] → [Database]

Admin Panel ↔ Product/Order/User

## 3.4 Technology Stack

Frontend: AngularJS, HTML, CSS, JS  
Backend: Node.js, Express.js  
Database: MongoDB + Mongoose  
Payment Gateway: Stripe / PayPal  
Tools: Git, Postman, VS Code

# 4. PROJECT DESIGN

## 4.1 Problem Solution Fit

The app addresses issues in traditional grocery shopping such as lack of convenience, transparency, and poor UI in existing solutions.

## 4.2 Proposed Solution

A seamless, responsive, and secure web app that supports:  
- Browsing and ordering groceries  
- Admin and seller dashboards  
- Inventory and report management  
- Payment and booking tracking

## 4.3 Solution Architecture

Frontend (Angular) → Backend API (Node.js + Express) → MongoDB Database

# 5. PROJECT PLANNING & SCHEDULING

## 5.1 Project Planning

Week 1: Requirement gathering  
Week 2: Frontend & backend setup  
Week 3: Database modeling  
Week 4: Product CRUD, user auth  
Week 5: Cart, booking, admin dashboard  
Week 6: Payment integration, testing  
Week 7: Bug fixes, optimization  
Week 8: Deployment & final report

# 6. FUNCTIONAL AND PERFORMANCE TESTING

## 6.1 Performance Testing

Load Testing: Handled 100+ concurrent users with no crash  
Speed Test: Page load time < 2 seconds  
Database Queries: Optimized using indexing and lean queries

# 7. RESULTS

## 7.1 Output Screenshots - Home page

## 

## -signup page

## 

## - login page

## - product list

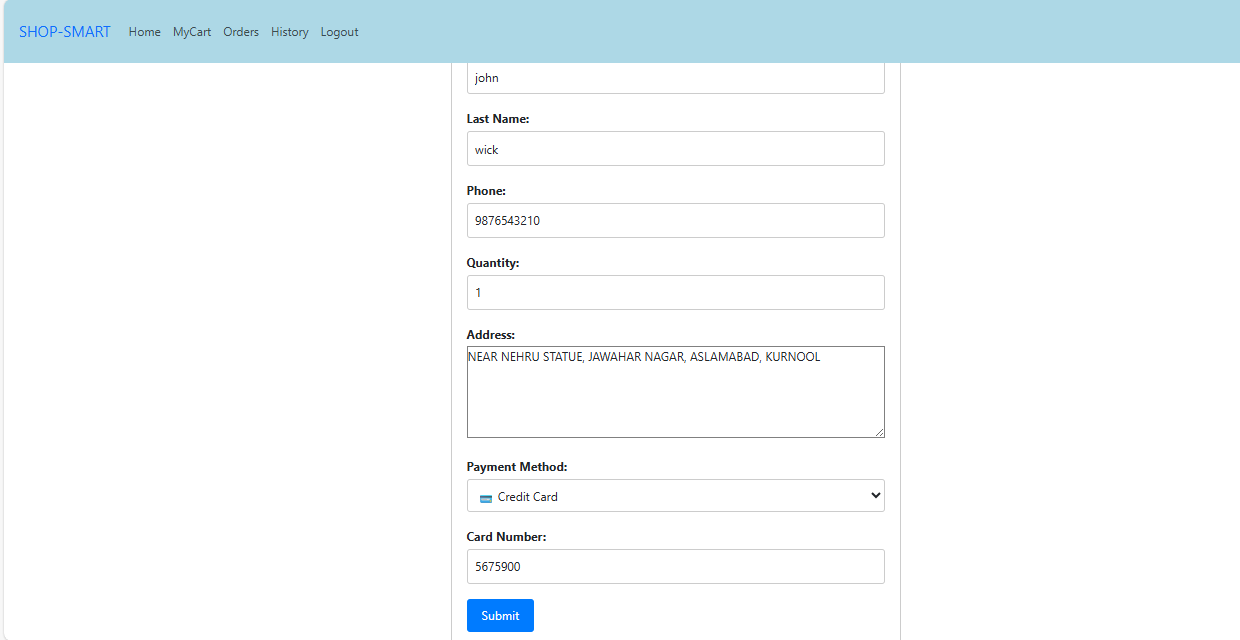
## 

## - Booking page

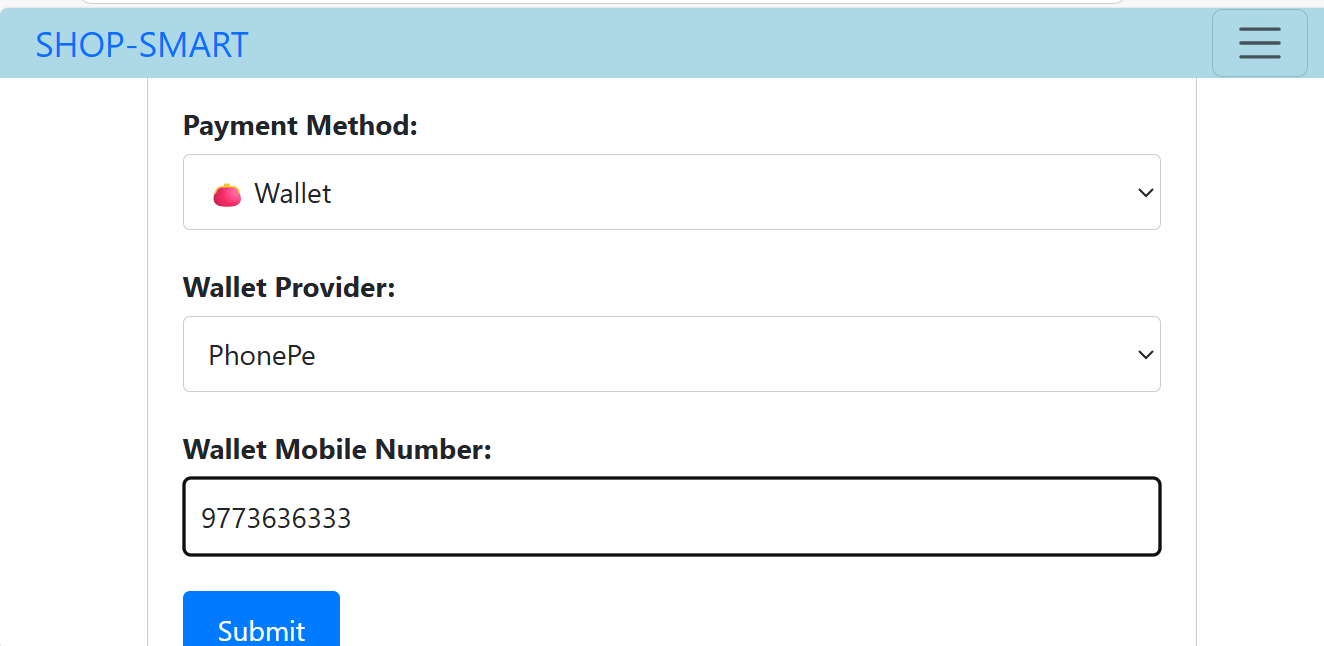
## 

## my cart

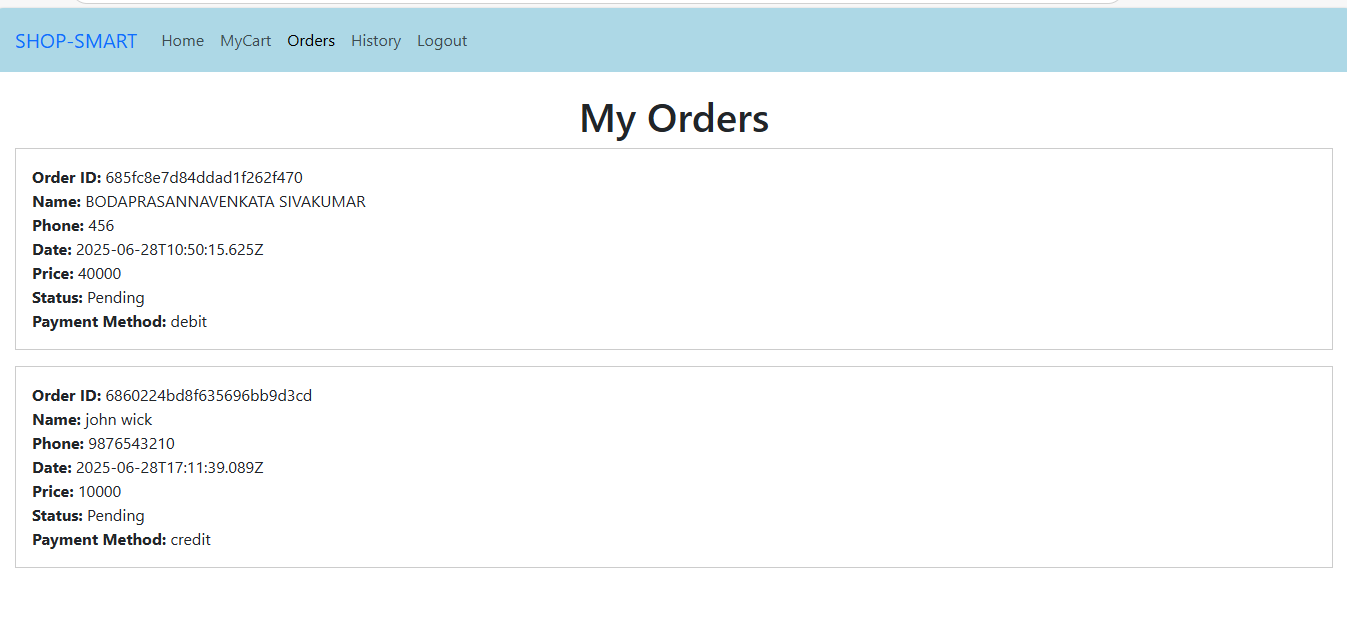
## Order Placement



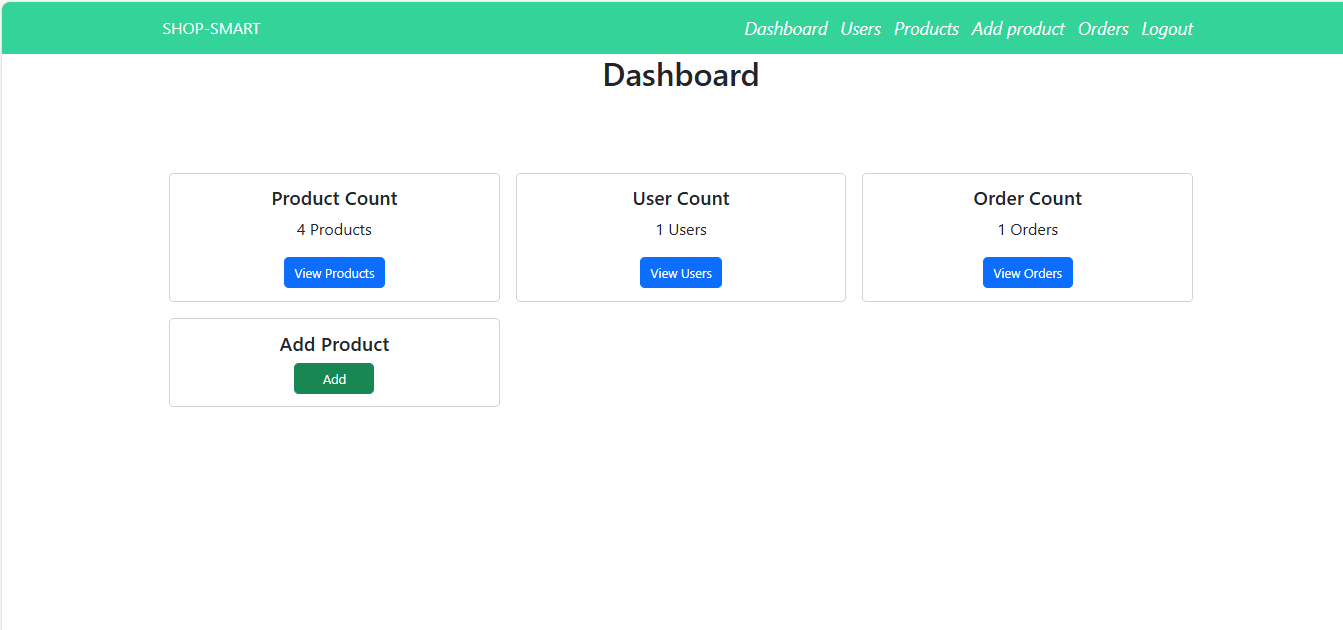
## Payment Confirmation



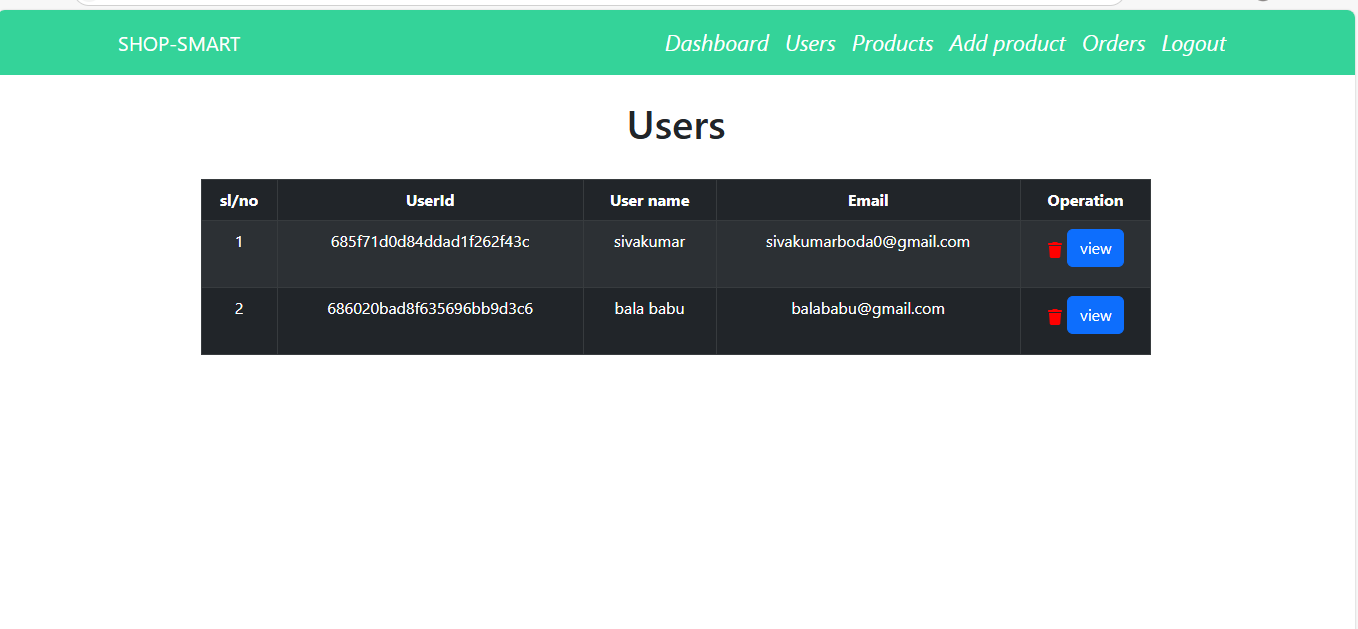
## Orders display



## Admin Dashboard



## Users



# 8. ADVANTAGES & DISADVANTAGES

Advantages:  
- User-friendly UI  
- Real-time updates  
- Admin control  
- Secure transactions

Disadvantages:  
- No offline mode  
- Payment gateway fees  
- Requires internet

# 9. CONCLUSION

ShopSmart successfully digitizes the grocery shopping experience by merging a smooth UI with robust backend capabilities. It benefits both customers and administrators, offering a modern solution to traditional problems in the grocery sector.

# 10. FUTURE SCOPE

- Add mobile app (React Native)  
- AI-based product recommendations  
- Integration with WhatsApp  
- Multi-language support

# 11. APPENDIX

Source Code: [https://github.com/Sivakumar4303/SHOP-SMART.git]

Demo Link: [https://youtu.be/Kdjh3AL\_OAQ]