Requirement Analysis

Technology Stack (Architecture & Stack)

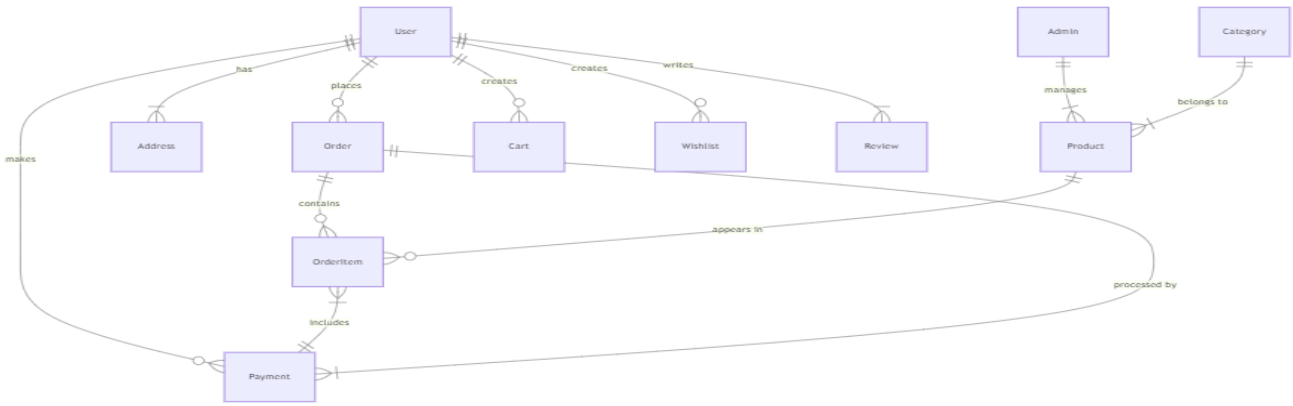
Date	19 June 2025
Team ID	LTVIP2025TMID48994
Project Name	ShopSmart: Your Digital Grocery Store Experience

Technical Architecture:

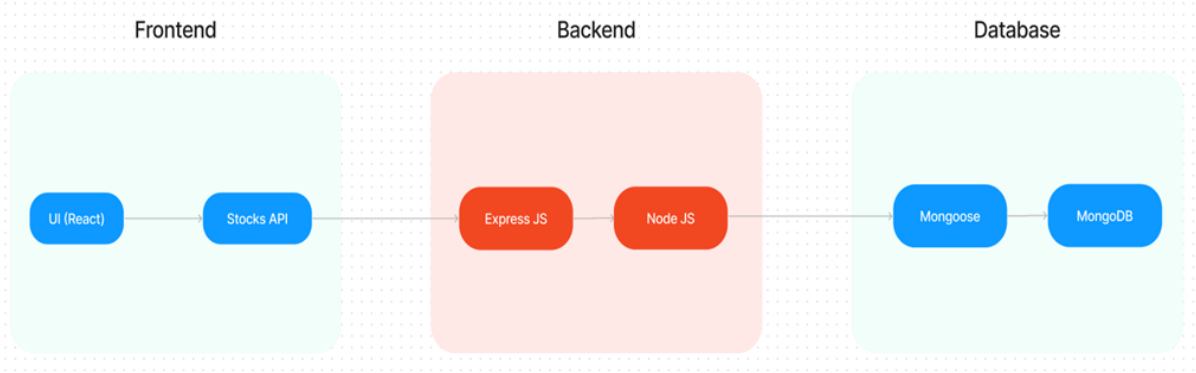
The technical architecture for ShopSmart is designed using the MERN stack, ensuring a scalable, modular, and responsive full-stack solution for grocery shopping. The architecture consists of user-facing interfaces, application logic handled by server-side code, and persistent data storage through a NoSQL database. This architecture ensures a seamless flow of information and interactions for customers, sellers, and administrators.

Architecture Diagram:

ER-Diagram:



Technical Architecture:



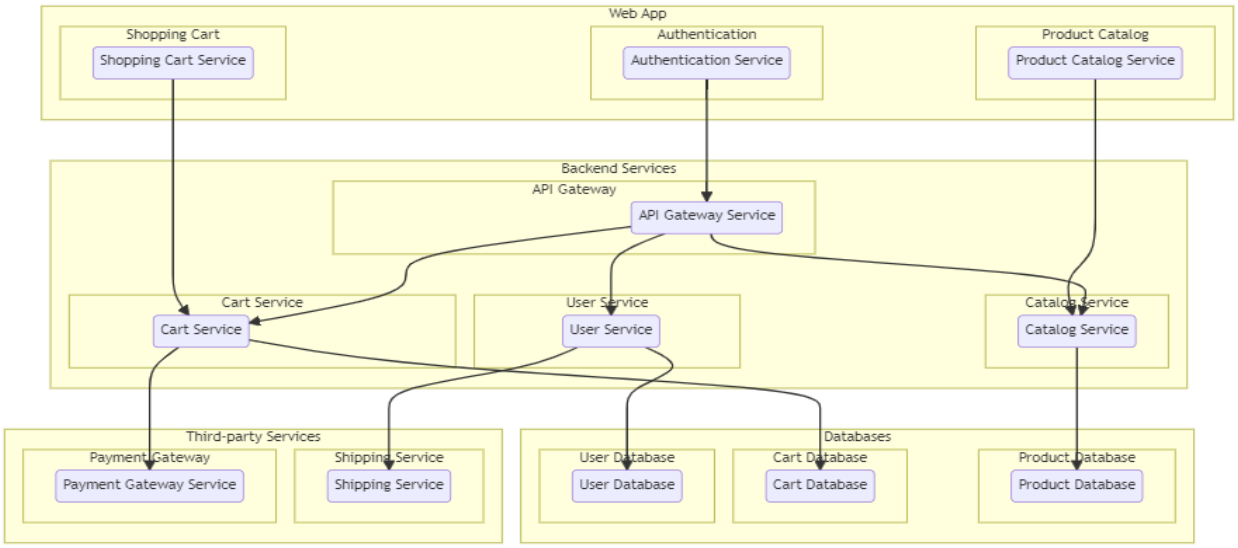


Table-1: Components & Technologies

S.No	Component	Description	Technology
1	User Interface	Web UI for customers, sellers, and admins	HTML, CSS, JavaScript, React.js
2	Application Logic-1	User authentication, registration, role-based access	Node.js, Express.js
3	Application Logic-2	Order processing, cart handling, inventory management	Node.js, Express.js, Mongoose
4	Application Logic-3	Admin control panel with full CRUD operations	Node.js, Express.js

5	Database	Persistent data storage for users, products, carts, orders	MongoDB
6	Cloud Database	Optional for deployment (MongoDB Atlas)	MongoDB Atlas
7	File Storage	Product images and static assets	Local filesystem or Cloudinary
8	External API-1	SMS/Email Notification for orders	Twilio API / Nodemailer
9	External API-2	(Optional) Payment integration	Razorpay / Stripe
10	Infrastructure	Application deployment	Localhost, Vercel (frontend), Render (backend)

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Used for frontend/backend development	React.js, Node.js, Express.js, MongoDB
2	Security Implementations	JWT Authentication, Bcrypt password hashing, role-based access	JWT, Bcrypt, CORS, HTTPS
3	Scalable Architecture	Modular design following a 3-tier architecture	MERN Stack, RESTful APIs
4	Availability	24/7 uptime goal, deployment on reliable services	Vercel, Render, MongoDB Atlas
5	Performance	Fast API responses, optimized rendering, efficient DB queries	Axios, React Hooks, Caching (local)