Hackathon 2025 - Marketplace Builder Day 2

Report: Planning the Technical Foundation for Q-Commerce Marketplace.

1. Introduction

The second day of the hackathon focused on refining the business goals from Day 1 into concrete technical solutions. The primary aim was to establish the architectural foundation of the Q-Commerce marketplace, ensuring seamless functionality, scalability, and efficiency.

2. Business Objectives and Technical Requirements

Based on our Day 1 research, the following key business objectives were identified and translated into technical requirements:

- **Instant Order Fulfillment:** Implement a real-time order processing system with inventory tracking.
- **Diverse Product Catalog:** Create a scalable database structure to handle multiple product categories.
- **Fast and Secure Payments:** Integrate multiple payment gateways ensuring secure transactions.
- **User-Friendly Experience:** Develop a responsive, intuitive UI with seamless navigation.
- **Vendor and Rider Management:** Build separate dashboards for vendors and delivery personnel.

3. Tech Stack Selection

After evaluating multiple technologies, we finalized the following stack to ensure performance, reliability, and scalability:

- **Frontend:** Next.js, React, Tailwind CSS (for an interactive and optimized user experience)
- **Backend:** Node.js with Express (for handling API requests and business logic)
- **Authentication:** Firebase Auth / NextAuth (for secure user authentication)
- **Payment Integration:** Stripe & PayPal (for flexible and secure payment processing)
- **Deployment:** Vercel (for frontend) (for backend and dat abase hosting)

4. System Architecture

A **microservices-based** approach was selected to ensure modularity and scalability. The high-level architecture consists of:

- **User Module:** Manages authentication, user profiles, and preferences.
- **Product & Inventory Module:** Handles product listing, stock management, and dynamic pricing.
- **Order Management Module:** Processes orders, tracks real-time status, and sends notifications.
- **Vendor & Rider Module:** Allows onboarding, performance tracking, and order assignment.
- **Payment Gateway Integration:** Ensures seamless and secure transactions.

5. API Design and Data Flow

The API endpoints were planned using REST principles to ensure smooth data exchange. The critical endpoints include:

- **User Authentication:** /api/auth/signup, /api/auth/login
- **Product Listing:** /api/products, /api/products/{id}
- **Cart Management:** /api/cart/add, /api/cart/remove, /api/cart/checkout
- **Order Processing:** /api/orders/create, /api/orders/status
- **Vendor & Rider Management:** /api/vendor/dashboard, /api/rider/assignments

8. Conclusion and Next Steps

The technical planning phase laid a solid foundation for development. The next phase will involve **prototyping key features** and building the **initial working version** of the Q-Commerce marketplace. Focus areas include **frontend UI design, backend API integration, and setting up the database schema.**