Build a responsive web-based delivery system for KwaMhlanga residents to order **food and groceries** from local stores and restaurants. The system will enable:

* Users to place and track orders.
* Store managers to receive and manage orders.
* Drivers to view and complete deliveries.
* Admins to oversee operations.

Other services (documents, parcels, etc.) will be added in future updates.

**👥 Stakeholders (Refined)**

* **Customers**
* **Drivers**
* **Store Managers**
* **Admin**
* **Developer/Technical Team**

**👤 User Roles & Stories**

**1. Customer**

*“As a customer, I want to...“*

* Register/login using email or phone number.
* Browse a list of restaurants and grocery stores.
* View products/menus with prices.
* Add items to a cart and place an order.
* Pay online via card (credit/debit).
* Track my order live.
* View order history.
* Receive delivery notifications (email/SMS/in-app).

**2. Store Manager**

*“As a store manager, I want to...”*

* Login securely.
* Upload and manage menus/products.
* Accept or reject incoming orders.
* Set estimated prep time.
* View order history and revenue reports.
* Communicate with drivers or admin.

**3. Driver**

*“As a driver, I want to...”*

* Login and see available delivery requests.
* Accept or reject delivery tasks.
* View delivery address and customer contact.
* Use maps to navigate to the customer.
* Mark orders as delivered.

**4. Admin**

*“As an admin, I want to...”*

* Manage users (customers, drivers, managers).
* Add/edit/remove stores.
* View reports on system usage, sales, deliveries.
* Handle disputes or support queries.
* Monitor delivery performance.
* Set service availability (e.g., turn off document delivery for now).

**🔧 System Features (Functional Requirements)**

* User registration/login (Email/Phone/Social)
* Store and menu/product listings
* Shopping cart and checkout
* Online payment (credit/debit card)
* Live delivery tracking (Map API)
* Notifications (order updates)
* Order management (store manager)
* Delivery assignment (manual or auto)
* Admin panel with full control
* “Coming Soon” banner for other services

**⚙️ Non-Functional Requirements**

* **Scalability:** Able to add other services later (e.g., documents, parcels).
* **Security:** Protect personal and payment info (HTTPS, authentication).
* **Performance:** Load under 2s; handle concurrent users efficiently.
* **Reliability:** Minimal downtime; alerts in case of errors.
* **Mobile Responsiveness:** Should work well on phones and tablets.

**📐 System Architecture Overview**

* **Frontend:** HTML, CSS, JavaScript
* **Backend:** Node.js + Express.js
* **Database:** MongoDB (recommended for flexibility) or MySQL
* **Third-party Services:**
  + Google Maps API (for live tracking)
  + Stripe or Paystack (for payments)
  + Firebase or Twilio (for notifications, optional)

**📊 Data Models (Sample Entities)**

* **User:** name, email, phone, password, role
* **Store:** name, manager, products/menu, location
* **Product:** name, category, price, availability
* **Order:** items, total, user, store, driver, status, timestamps
* **Driver:** name, availability, current location, deliveries

**🚧 Scope Constraints**

* Focused only on food and grocery deliveries for launch.
* Admin manually approves stores and drivers for quality control.
* Live tracking limited to driver’s phone GPS for MVP.

**📌 Coming Soon (future services)**

* Document delivery
* Parcel delivery
* Mobile app (Android/iOS)
* Loyalty program
* Reviews and ratings