Product Requirements Document (PRD): AgriConnect

Version: 1.0

Status: Draft

Type: Web Application (B2B/B2C Marketplace + Farm ERP)

1. Executive Summary

AgriConnect is a dual-purpose platform designed to empower farmers.

1. **Management Side:** Helps farmers track crop cycles, inventory, and expenses.
2. **Marketing Side:** Connects farmers directly with buyers (wholesalers, retailers, or consumers), eliminating middlemen and ensuring fair pricing.

Goal: To digitize the agricultural supply chain, providing transparency in pricing and helping farmers manage their yield effectively.

---

2. User Personas

Role	Description	Key Goals
The Farmer	The producer of goods.	Manage crop schedules, track expenses, list harvest for sale, receive payments.
The Buyer	Wholesaler, Retailer, or Consumer.	Search for specific crops, negotiate bulk prices, track delivery.
The Admin	Platform moderator.	Verify farmer IDs (KYC), resolve disputes, manage crop categories.

---

3. Functional Requirements (MVP)

3.1 Module A: Farm Management (The "ERP")

- **Farm Profile:** Farmers register their land details (Size, Location/Coordinates, Soil Type).
- **Crop Calendar:**
  - Input: Planting Date, Crop Variety, Expected Harvest Date.
  - Output: System estimates harvest volume.
- **Expense Tracker:**
  - Farmers log costs (Seeds, Fertilizers, Labor).
  - **Dashboard:** Shows "Cost of Production per kg" so farmers know their break-even price.

3.2 Module B: Marketing & Sales (The "Marketplace")

- **Produce Listings:** Farmers convert "Harvested Crops" into "Sale Listings" (Price/kg, Available Quantity, Photos).
- **Smart Search:** Buyers filter by:
  - **Distance:** "Show farmers within 50km."
  - **Crop Type:** "Organic Tomatoes."
  - **Harvest Date:** "Freshly harvested today."
- **Cart & Checkout:** Standard e-commerce flow for buyers.

3.3 Module C: Logistics & Updates

- **Order Status:** Pending -> Confirmed -> Packed -> Out for Delivery -> Completed.
- **Digital Bill:** Auto-generate a PDF Invoice for every transaction.

---

4. Technical Architecture

4.1 Tech Stack

- **Backend:** Python, Django 5.x
- **Database:**ANY DBMS.
- **Frontend:** Tailwind CSS (Lightweight and mobile-responsive for rural internet speeds).
- **Location Services:** GeoDjango + PostGIS (To calculate distances between Farmer and Buyer).

---

5. Database Schema (Core Models)

This schema bridges the gap between internal management and external sales.

A. Users & Profiles

- CustomUser: role (Farmer/Buyer), phone_number.

- FarmProfile: user (FK), location (PointField), acreage, certification_image.

B. Management Models

- CropCycle:
  - farm (FK), crop_name, planting_date, expected_harvest_date.
  - status (Growing, Ready, Harvested).
- Expense:
  - cycle (FK), item_name, cost, date.

C. Market Models

- Listing:
  - farmer (FK), crop_cycle (FK - links sales to production), price_per_kg.
  - min_order_qty, images.
- Order:
  - buyer (FK), listing (FK), quantity_ordered, total_price, status.

---

6. Development Roadmap

- **Phase 1 (Setup):** Configure GeoDjango and PostGIS. Set up User Authentication (Farmer vs Buyer).
- **Phase 2 (Farm ERP):** Build the "My Farm" dashboard where farmers log crops and expenses.
- **Phase 3 (Marketplace):** Build the public listing page and search filters (Distance/Crop).
- **Phase 4 (Transactions):** Implement the Order and Cart models.