## 🌾 AI Crop & Pest Diagnosis App: Phase-by-Phase Development Plan

### 🟢 Phase 1: MVP Development (Month 1–3)

📅 **Timeline:** Month 1 to 3  
🎯 **Goal:** Build a working minimum viable product (MVP) focused on core diagnosis functionality

**Key Features:**

* 📸 Image-based crop disease diagnosis using phone camera
* 🌐 Offline functionality with local data caching
* 🎙️ Multilingual voice support (Hausa, Yoruba, Igbo)
* 🧾 Farm profile creation (Name, crop type, location)

**Team Focus:**

* Mobile & backend developers: Core logic + offline mode
* AI team: Train and integrate first disease detection model
* UI/UX: Build onboarding, registration & diagnosis screens in Figma
* QA: Initial user testing and feedback collection



### 🟡 Phase 2: Core Enhancements (Month 4–6)

📅 **Timeline:** Month 4 to 6  
🎯 **Goal:** Add intelligence, expert feedback, and tracking system

**Key Features:**

* 📊 Farm history dashboard (track past diseases & solutions)
* 🧠 AI + human agronomist hybrid consultation system
* 💊 Disease treatment tips and solution suggestions
* 🗺️ Basic GPS-based farm location and mapping

**Team Focus:**

* Backend devs: Build user dashboard and data models
* AI: Improve accuracy, add more crop/disease types
* Partner agronomists: Advisory layer setup
* Design: Result screen, farm map integration



### 🟠 Phase 3: Expansion Features (Month 7–9)

📅 **Timeline:** Month 7 to 9  
🎯 **Goal:** Build engagement & knowledge-sharing layer

**Key Features:**

* 👨‍🌾 Community Q&A forum for farmers
* 📅 Smart tips: Planting calendar, rainfall alerts, market price updates
* 🧑‍🏫 Onboarding module for extension workers and NGOs

**Team Focus:**

* Devs: Implement Q&A system, user roles (farmers, extension agents)
* Data/AI: Weather API, market price logic
* UX: Improve user experience for low-literacy users
* Marketing: Start outreach to farmer groups & cooperatives



### 🔵 Phase 4: Marketplace Integration (Month 10–12)

📅 **Timeline:** Month 10 to 12  
🎯 **Goal:** Enable buying of farm inputs directly from the app

**Key Features:**

* 🛒 In-app shop for pesticides, fertilizers, and tools
* 🚚 Logistics support module for input delivery
* 🧾 Crop insurance information & advisory

**Team Focus:**

* Backend + payment gateway setup
* Partner onboarding: Vendors, logistics, insurers
* Legal: T&Cs, compliance for agri-commerce
* UI: Design marketplace and transaction interface



## 🛠️ Bonus: Design & Presentation Tools

* High-fidelity Figma wireframes for each feature
* Clickable prototype (optional for investor/demo sessions)
* Component library for frontend consistency



## 📅 Summary Timeline:

| **Phase** | **Duration** | **Focus** |
| --- | --- | --- |
| Phase 1 | Month 1–3 | MVP (Camera, Offline, Voice, Farm Profile) |
| Phase 2 | Month 4–6 | Dashboard, AI + Expert Feedback, GPS |
| Phase 3 | Month 7–9 | Forum, Smart Tips, Extension Workers |
| Phase 4 | Month 10–12 | Marketplace, Logistics, Insurance |

Based on the **AI Crop & Pest Diagnosis App** development plan you provided, here’s a recommended **GitHub project structure**. It’s modular and scalable, with folders and files aligning with the four development phases: MVP, Enhancements, Expansion, and Marketplace.

### **✅ Root Structure**

nomaapp-ai-crop-diagnosis/

├── README.md

├── LICENSE

├── .gitignore

├── docs/

│ └── development\_plan.md

├── frontend/

│ ├── mobile-app/

│ │ ├── src/

│ │ ├── assets/

│ │ ├── components/

│ │ ├── screens/

│ │ └── App.js

│ └── web-dashboard/

│ ├── src/

│ ├── public/

│ ├── components/

│ ├── pages/

│ └── index.js

├── backend/

│ ├── src/

│ │ ├── controllers/

│ │ ├── models/

│ │ ├── routes/

│ │ ├── services/

│ │ └── server.js

│ └── config/

│ ├── db.js

│ └── env/

├── ai-models/

│ ├── training-data/

│ ├── notebooks/

│ ├── models/

│ └── inference-api/

│ └── predict.py

├── data/

│ ├── cache/

│ └── mock/

├── gps-mapping/

│ ├── maps/

│ ├── api/

│ └── geolocation-utils.js

├── marketplace/

│ ├── shop/

│ ├── payments/

│ ├── vendors/

│ └── insurance/

├── voice-support/

│ ├── hausa/

│ ├── yoruba/

│ ├── igbo/

│ └── utils/

└── tests/

├── unit/

├── integration/

└── e2e/

### **📁 Folder Breakdown**

#### docs/

* Contains your phase-based dev plan, documentation, diagrams, and wireframes.

#### frontend/

* **mobile-app/**: React Native or Flutter source code.
* **web-dashboard/**: Admin/farmer dashboard interface (React/Vue).

#### backend/

* Express.js, Django, or Node.js REST API setup.
* Models include farms, users, diseases, consultations, etc.

#### ai-models/

* Training scripts, disease detection models, and the API that powers inference.

#### data/

* Caching logic for offline mode and mock farm data.

#### gps-mapping/

* Logic for GPS location, farm boundaries, and geospatial analysis.

#### marketplace/

* Ecommerce functionalities: vendors, logistics, shop items, payments.

#### voice-support/

* Multilingual voice interface resources and NLP handling for Hausa, Yoruba, and Igbo.

#### tests/

* Structured for unit, integration, and end-to-end testing across modules.

### **📌 Suggested Next Steps**

1. Add a **README.md** that outlines:  
   * Project purpose
   * Tech stack
   * Dev phases
   * Setup instructions
2. Use GitHub Issues + Projects to map out features by phase.

Create separate branches:  
  
 phase-1-mvp

phase-2-core

phase-3-expansion

phase-4-marketplace

If you want, I can generate the folder structure programmatically or provide starter boilerplate code for any of the modules (e.g. image diagnosis model, offline cache, payment integration). Just let me know!