**Project Title:** AI-Driven Crop Disease Detection for Sustainable Agriculture

**Team Members**

● Abenezer Tesfaye

● Shewanek Zewdu

● Emran Kamil

● Hewan

● Biniyam Berga

## **Concept Note**

### **1. Project Overview**

Crop diseases threaten yields, food security, and farmer livelihoods worldwide. By leveraging deep learning for automated image‑based disease detection, our project aims to identify infections in leaves at early stages, enabling timely interventions that reduce crop loss and pesticide use. This aligns with **SDG 2 (Zero Hunger)** by boosting food production efficiency and **SDG 12 (Responsible Consumption and Production)** by minimizing waste.

### **2. Objectives**

* **Early Detection:** Build a model that classifies healthy vs. diseased leaves with ≥95% accuracy.
* **Scalability:** Design a pipeline that can adapt to multiple crops and regions.
* **Accessibility:** Deliver a user‑friendly web/mobile interface for farmers and extension officers.
* **Impact:** Reduce time to diagnosis by at least 30% compared to manual inspection.

### **3. Background**

Traditional visual inspections are labor‑intensive, error‑prone, and often too late to prevent significant yield loss. Recent studies demonstrate that CNNs (e.g. ResNet, Inception) can achieve >99% classification accuracy on benchmark datasets like PlantVillage. However, real‑world deployment faces challenges of variable lighting, backgrounds, and device constraints. A robust AI solution must bridge this lab‑to‑field gap by incorporating diverse training data, augmentation, and lightweight model architectures.

### **4. Methodology**

1. **Data Preprocessing:** Resize, normalize, augment leaf images (rotations, color jitter).
2. **Modeling:**
   1. **Base Architecture:** Transfer‑learn from ImageNet‑pretrained ResNet50.
   2. **Fine‑Tuning:** Unfreeze top layers, apply domain‑specific augmentation.
   3. **Optimization:** Use early stopping and learning‑rate scheduling.
3. **Evaluation:** K‑fold cross‑validation, confusion matrix analysis, and field‑collected test set.
4. **Deployment:**
   1. **Server‑Side:** Flask/FastAPI serving TensorFlow/PyTorch model.
   2. **Client‑Side:** React web dashboard + optional edge deployment on mobile devices.

### **5. Architecture Design Diagram**

flowchart LR  
 A[Image Acquisition] --> B[Preprocessing]  
 B --> C[Model Training & Validation]  
 C --> D[Inference API]  
 D --> E[User Interface]  
 A --> F[Edge Device Deployment]

* **Image Acquisition:** Smartphone/camera or drone captures leaf images.
* **Preprocessing:** Cleansing, resizing (224×224), normalization, augmentation.
* **Model Training & Validation:** Transfer learning on ResNet50; hyperparameter tuning.
* **Inference API:** Exposes prediction endpoint via REST.
* **User Interface:** Web/mobile app for uploading images and viewing results.
* **Edge Device Deployment:** Optional TensorFlow Lite model for offline use.

### **6. Data Sources**

We will use the **PlantVillage** dataset—a collection of over 50,000 labeled images covering healthy and diseased leaf samples across 14 crop species. Images are in JPEG/PNG format. Preprocessing includes resizing to 224×224 pixels, normalizing RGB channels, and augmenting to simulate field conditions (rotation, brightness/contrast adjustments).

### **7. Literature Review**

Convolutional Neural Networks (CNNs) are the de facto standard for image‑based disease classification, achieving up to 99.35% accuracy on lab‑captured leaf images via transfer learning and data augmentation citeturn0file1. Drone‑mounted multispectral/hyperspectral imaging further expands coverage but adds complexity and cost. The PlantVillage dataset serves as a benchmark, though it under‑represents field variability. Addressing these gaps requires domain adaptation, robust augmentation, and lightweight architectures for edge deployment.

## **Implementation Plan**

### **1. Technology Stack**

* **Languages:** Python, JavaScript (TypeScript)
* **ML Frameworks:** TensorFlow 2.x or PyTorch
* **Data Processing:** OpenCV, Pillow, NumPy, Pandas
* **API & Backend:** FastAPI or Flask, Docker
* **Frontend:** React (Next.js optional)
* **Deployment:** AWS (EC2, S3, Lambda) or GCP; Docker & Kubernetes for scaling
* **Mobile/Edge:** TensorFlow Lite, React Native (optional)
* **Version Control:** Git & GitHub/GitLab
* **CI/CD:** GitHub Actions

**Task Distribution Matrix**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task** | **Olana** | **Heaven** | **Selemon** | **Yonas** | **HaileMaryam** |
| Data Collection & Augment | X |  |  | X |  |
| Model Development |  | X | X |  |  |
| API & Backend |  |  |  | X | X |
| Frontend/UI |  |  | X |  | X |
| Deployment & Edge Build | X |  |  | X | X |
| Testing & Validation | X | X | X |  | X |
| Documentation & Reporting |  |  | X |  | X |

### **3. Milestones**

* **M1:** Data pipeline operational (05/01/2025)
* **M2:** Baseline model trained (05/10/2025)
* **M3:** Optimized model ≥95% accuracy (05/152025)
* **M4:** API + UI ready for testing (05/25/2025)
* **M5:** Edge deployment prototype (05/27/2025)
* **M6:** Field testing completed (06/01/2025)

### **4. Challenges and Mitigations**

* **Data Quality & Variability:** Mitigate via extensive augmentation and domain‑adaptive fine‑tuning.
* **Model Generalization:** Use cross‑validation, field‑collected samples, and regularization to avoid overfitting.
* **Resource Constraints:** Employ model pruning and TensorFlow Lite for edge devices; leverage cloud GPU instances.
* **User Adoption:** Conduct farmer workshops; design intuitive UI; gather feedback for iterative improvements.

### **5. Ethical Considerations**

* **Data Privacy:** Ensure anonymity of farm locations; obtain consent for image collection.
* **Bias & Fairness:** Include diverse crop species and severity levels to prevent model bias.
* **Socio‑Economic Impact:** Offer low‑cost or open‑source deployment to benefit smallholder farmers equally.

### **6. References**

1. Barbedo J.G.A., Plant Disease: A Growing Threat to Global Food Security - MDPI, accessed April 10, 2025, https://www.mdpi.com/2073-4395/14/8/1615
2. Hughes D.P., Salathe M., “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” *PNAS*, 2025.
3. Hughes D., “Agriculture and Food Sustainability with AI – Disease Detection,”