# Project Proposal Document

### Project Title

**Smart Paddy Disease Detection using AI** (for Sri Lankan Farmers)

### Problem Statement

Farmers in Sri Lanka often suffer significant crop losses due to late detection of paddy diseases. Most rely on manual inspection, which is time-consuming and inaccurate. There is a lack of accessible and affordable tools to help farmers detect diseases early.

### Objective

To develop an AI-powered tool that identifies diseases in paddy leaves using image classification. The tool will allow users (ex, farmers, agricultural officers) to upload images of leaves and receive instant feedback on whether they are healthy or diseased.

### Users

* Farmers
* Agricultural Officers

### Feasibility Analysis

|  |  |
| --- | --- |
| **Type** | **Analysis** |
| Technical | Users have proven CNN models (like MobileNetV2), transfer learning, and Python libraries like TensorFlow and Streamlit. No complex hardware required. |
| Time | Can be built in 2 weeks using an open dataset and pre-trained models |
| Operational | Streamlit web app makes it usable on laptops/mobiles by farmers/agriculture officers. |
| Economic | No cost for training (uses Google Colab). All tools and data are open-source and free. |

### Type of Project

AI/ML-based image classification tool for smart agriculture.

### Development Plan

|  |  |
| --- | --- |
| **Week** | **Tasks** |
| Week 1 | Data collection, preprocessing, and model training using transfer learning |
| Week 2 | Build and test the Streamlit app, integrate the model, create documentation, and finalize the demo. |

### Tools and Technologies

**Language**: Python

**Libraries**: TensorFlow / Keras, Numpy, OpenCV, Streamlit

**Platform**: Google Colab (for training), Streamlit (for app)

**Model**: MobileNetV2 or ResNet50 (CNNs)

**Frontend**: Streamlit Web App

**OS**: Cross-platform (Web browser)

### Dataset

|  |  |
| --- | --- |
| **Source** | **Labels** |
| Rise Leaf Disease Dataset | Healthy, Brown Spot, Bacterial Blight, Leaf smut |

### System Architecture

User Uploads Image ➝ Image Preprocessing ➝ CNN Model ➝ Prediction ➝ Display Result + Information

### Machine Learning Model

**Type**: Image Classification

**Technique**: Transfer Learning with MobileNetV2

**Process**:

1. Input: Preprocessed leaf image
2. Feature Extraction using MobileNetV2
3. Classification Layer: Outputs disease type

**Training**: On Google Colab using GPU

### User Interface (UI)

**Home Page**: Introduction to the app

**Upload Section**: Upload leaf image

**Prediction Output**: Disease type + Confidence

**Information Box**: Disease description + Solution

### Expected Outcome

* A functional web application that can accurately detect paddy diseases from leaf images.
* Helps farmers make timely decisions, reducing crop loss.
* Open-source and extendable for other crops or regions.
* Give the output in the Sinhala language.

### Conclusion

This project uses machine learning to solve a real-world agricultural issue faced by Sri Lankan farmers. It is technically feasible, affordable, impactful, and aligned with national digital agriculture goals.