Software Requirement Specifications

AI Based Crop Disease Detection System

Submitted by

Muhabbat Ali Khan (F22BINFT1E02100)

Submitted to

Mr. Muzamil ur Rehman

Department of Information Technology

Faculty of Computing

The Islamia University of Bahawalpur

Meeting Details

Sr No	Details	Date	Supervisor Signature

Summary

This document is the Software Requirement Specification (or SRS) for our new AI project that helps find diseases in crops. The main goal of this system is to help farmers by quickly spotting what’s wrong with their plants just by looking at pictures of the leaves.

We are using Artificial Intelligence and some special image tools to make sure the app can tell exactly what disease a plant has. It doesn't just find the problem, though—it also tells the farmer how to fix it. We really want this to help stop crops from dying, keep people from using too many chemicals/pesticides, and help farms grow more food. This document explains exactly what the app needs to do, how it should work, and how we’re going to build it.

Table of Content

1. Introduction

Agriculture plays a vital role in the economy, especially in countries like Pakistan where a large population depends on farming. Crop diseases are a major cause of reduced agricultural yield and economic loss. Due to lack of technical knowledge and expert access, farmers often fail to identify diseases at an early stage.

The **AI-Based Crop Disease Detection System** is developed to solve this problem by using Artificial Intelligence and Image Processing. The system allows farmers to upload images of crop leaves and automatically detects diseases with high accuracy. It also provides disease details and treatment suggestions, making it a reliable decision-support tool for farmers.

1.1. Purpose

The purpose of this project is to develop an intelligent system that can.

- Detect and classify crop diseases from leaf images.
- Assist farmers with instant diagnosis.
- Provide disease control and treatment suggestions
- Improve crop yield and reduce financial loss

1.2. Scope

The scope of this project includes:

- Detection of crop diseases using leaf images
- Support for selected crops such as wheat, rice, maize, tomato, and potato
- Disease classification using AI models (CNN)
- Display of disease name, confidence level, and treatment advice

Out of Scope:

- Soil analysis
- Weather-based prediction
- Automatic pesticide application

1.3. Product Perspective

The system is a standalone AI-based application that integrates:

- Image processing module
- AI disease classification model
- Web-based user interface
- Database for storing user data and results
- It can later be integrated with mobile applications or agricultural advisory platforms.

1.4. User Characteristics

User Type	Description
Farmer	Uploads crop images and views results
Admin	Manages system, datasets, and users
Researcher	Improves AI model and dataset

Users are expected to have basic smartphone or web browsing knowledge.

1.5. Similar Apps and Systems / Literature Review

Several crop disease detection systems already exist, such as:

- Plantix App
- LeafSnap

- Agri AI systems based on CNN

Shortcomings of existing systems:

- Limited crop coverage
- High cost
- Internet dependency
- Complex interfaces

This project aims to provide a **simple**, **low-cost**, and **accurate** solution.

1.6. Proposed Technologies

Category	Technology
Programming Languages	Dart, Python
AI/ML Libraries	Tensorflow, Keras, OpenCV
Web Framework	FastAPI
Front-End	Flutter
Database	PostgreSQL
Dataset	PlantVillage Dataset
Platform	Mobile

2. Requirements

The system will allow users to register, log in, upload images of crop leaves, and receive disease detection results. AI models will analyze the images and provide accurate predictions along with treatment suggestions. Admin users will manage datasets and system configurations.

2.1. Functional Requirements

2.1.1. Sign Up

- **Name:** FR001
- **Purpose:** Allows a new user to register in the system.
- **User(s):** Farmer, Admin
- **Input:**
    - Name
    - Email

    - Password (minimum 8 characters)
    - Phone Number
- **Output:**
    - User account created successfully

2.1.2. Login

- **Name:** FR002
- **Purpose:** Allows registered users to log into the system.
- **User(s):** Farmer, Admin
- **Input:**
    - **Email**
    - **Password**
- **Output:**
    - Access to system dashboard

2.1.3. Upload Leaf Image

- **Name:** FR003
- **Purpose:** Upload crop leaf image for analysis.
- **User(s):** Farmer
- **Input:**
    - Leaf Image (JPEG/PNG)
- **Output:**
    - Image successfully uploaded

2.1.4. Disease Detection

- **Name:** FR004
- **Purpose:** Analyze the image using AI model.
- **User(s):** System
- **Input:**
    - Uploaded image
- **Output:**
    - Disease name and confidence level

2.1.5. View Result and Treatment

- **Name:** FR005
- **Purpose:** Display disease details and treatment suggestions.
- **User(s):** Farmer
- **Output:**
    - Disease description
    - Recommended treatment

2.2. Non-Functional Requirements

- The system should provide results within 5 seconds
- Accuracy should be above 85%
- The system should be user-friendly
- Secure user authentication
- High availability and reliability

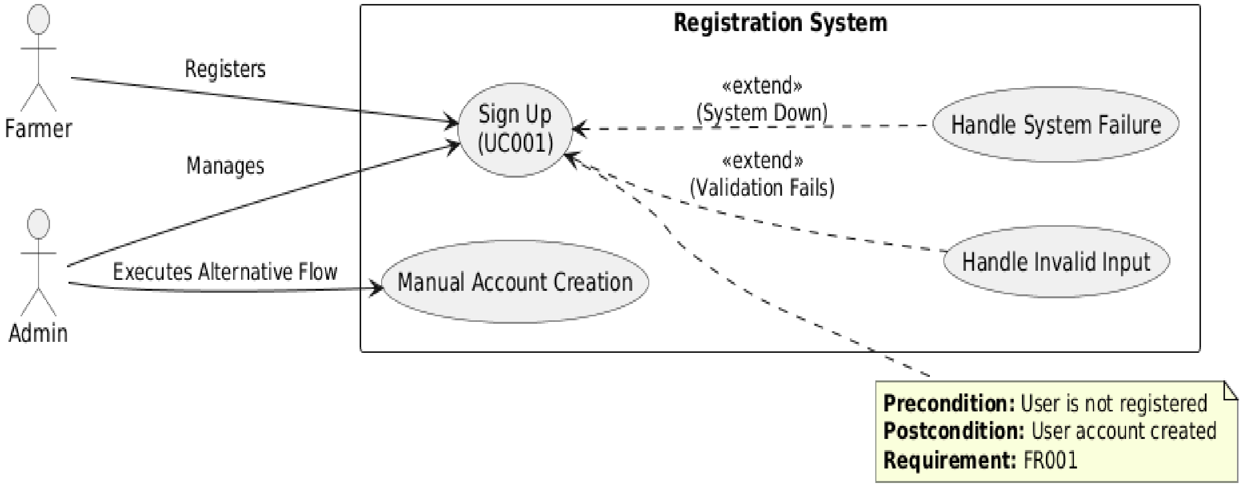
- Scalable for future crops and diseases

3. Use Case

3.1. Sign Up

Field	Description
Name	Sign Up
Description	User registration process
Actors(s)	Farmer, Admin
Precondition	User is not registered
Postcondition	User account created

- **Basic Flow:**
  a. User enters registration details
  b. System validates input
  c. System creates user account
- **Alternative Flow:**
  a. Admin creates user manually
- **Exceptions:**
  a. Invalid input
  b. System failure

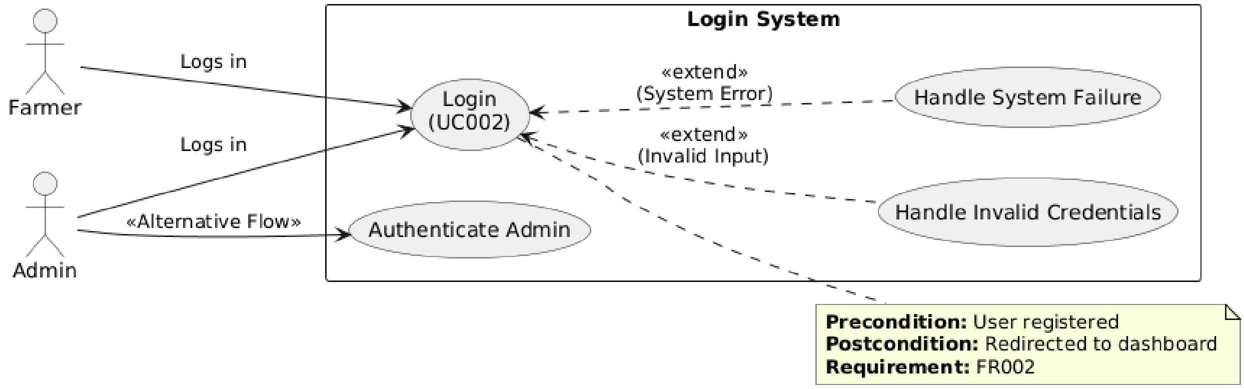


3.2. Login

Field	Description

Name	Login
Description	Registered users log into the system
Actors(s)	Farmer, Admin
Precondition	User must be registered
Postcondition	User is redirected to dashboard

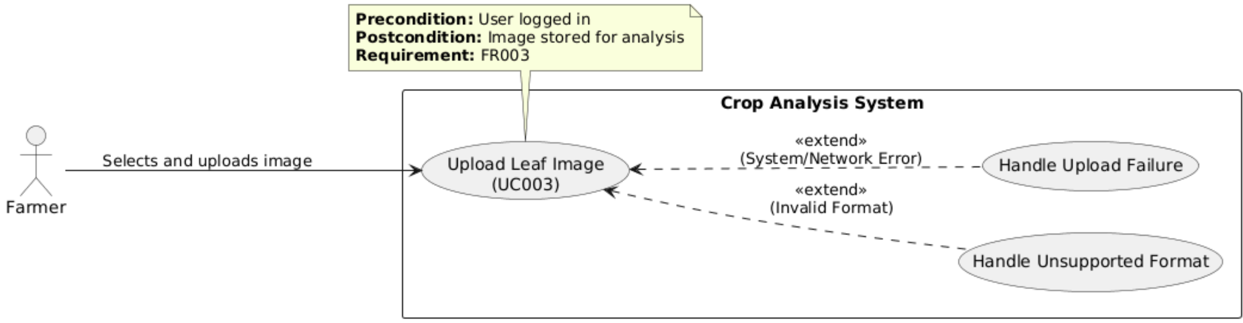
- **Basic Flow**
    a. User enters email and password
    b. System validates credentials
    c. System grants access
- **Alternative Flow**
    a. Admin logs in using administrator credentials
- **Exceptions**
    a. Invalid email or password
    b. System failure



3.3. Upload Leaf Image

Field	**Description**
Name	Upload Leaf Image
Description	Farmer uploads an image of a crop leaf
Actors(s)	Farmer
Precondition	User must be logged in
Postcondition	Image is stored for analysis

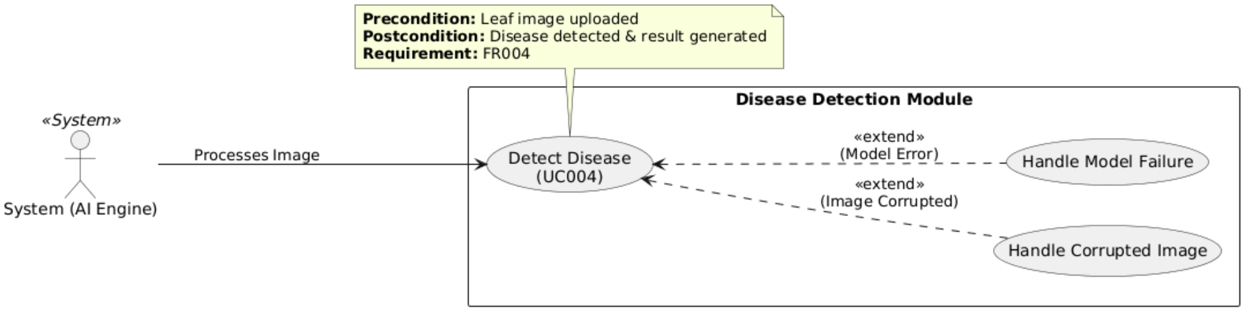
- **Basic Flow**
    a. Farmer selects image from device
    b. System checks image format
    c. Image is uploaded successfully
- **Exceptions**
    a. Unsupported file format
    b. Image upload failure


Precondition: User logged in
Postcondition: Image stored for analysis
Requirement: FR003
Crop Analysis System
Farmer
Selects and uploads image
Upload Leaf Image (UC003)
«extend» (System/Network Error)
Handle Upload Failure
«extend» (Invalid Format)
Handle Unsupported Format

3.4. Disease Detection

Field	**Description**
Name	Disease Detection
Description	AI model analyzes the uploaded image
Actors(s)	System
Precondition	Leaf image must be uploaded
Postcondition	Disease is detected

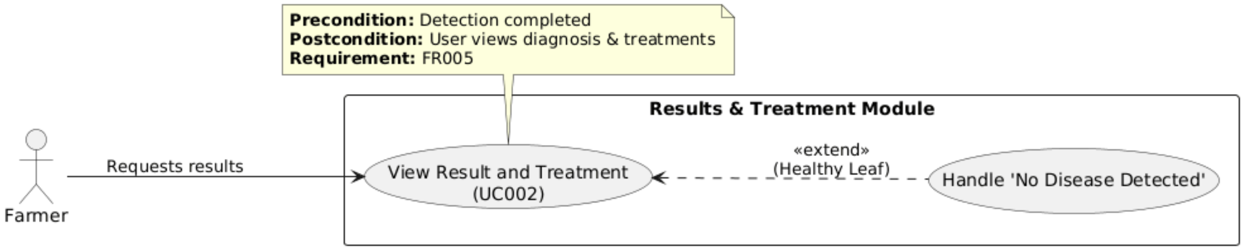
- **Basic Flow**
    a. System preprocesses the image
    b. CNN model predicts the disease
    c. Confidence score is generated
- **Exceptions**
    a. AI model failure
    b. Corrupted image



3.5. View Result and Treatment

Field	Description
Name	View Result and Treatment
Description	Display detected disease and treatment suggestions
Actors(s)	Farmer
Precondition	Disease detection is completed
Postcondition	User views diagnosis and recommendations

- **Basic Flow**
  a. System displays disease name
  b. System shows confidence level
  c. System suggests treatment options
- **Exceptions**
  a. No disease detected

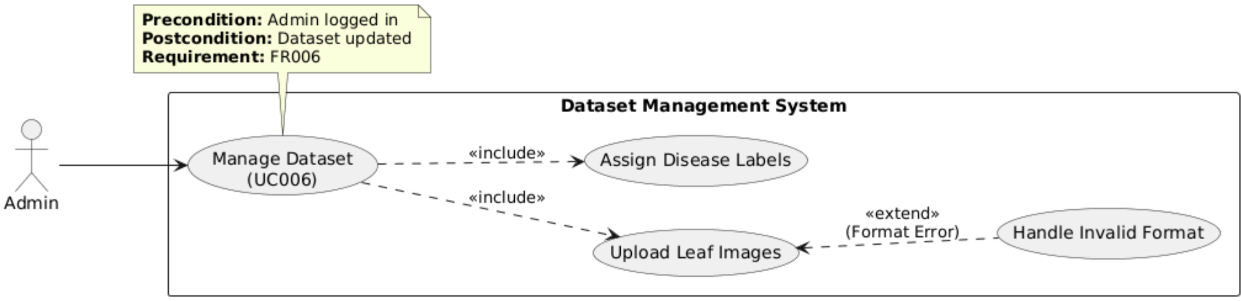

Precondition: Detection completed
Postcondition: User views diagnosis & treatments
Requirement: FR005
Farmer
Requests results
Results & Treatment Module
View Result and Treatment
(UC002)
«extend»
(Healthy Leaf)
Handle 'No Disease Detected'

3.6. Manage Dataset (Admin)

Field	Description

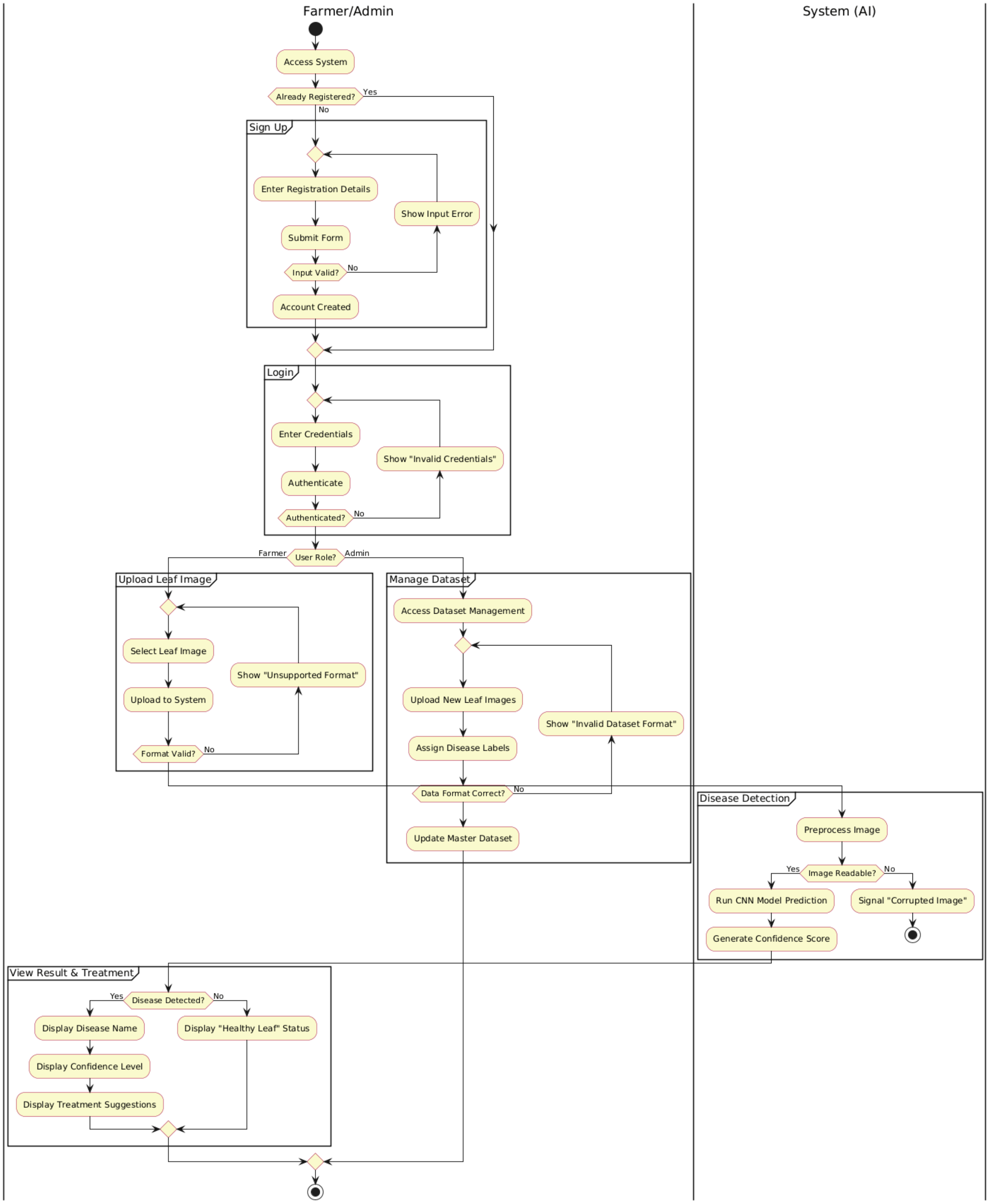
Name	Manage Datasets
Description	Admin manages crop disease dataset
Actors(s)	Admin
Precondition	Admin is logged in
Postcondition	Dataset is updated

- **Basic Flow**
    a. System displays disease name
    b. System shows confidence level
    c. System suggests treatment options
- **Exceptions**
    a. No disease detected



4. Flow Process

Crop Disease Detection System - Integrated Activity Diagram



5. References

5.1. Mohanty, S. P., Hughes, D. P., & Salathé, M. “Using Deep Learning for Image-Based Plant Disease Detection,” *Frontiers in Plant Science*, IEEE.

5.2. TensorFlow Documentation, IEEE Format.

5.3. PlantVillage Dataset Research Papers.