Tomato Care

By:

Muhammad Salman Afaq
24775
Usman Afaq
24779
Rafaqat Ahmad
24784

Supervised by:
Mr. Muhammad Usman Karim

Faculty of Computing
Riphah International University, Islamabad

Spring 2024

A Dissertation Submitted To

Faculty of Computing,

Riphah International University, Islamabad

As a Partial Fulfillment of the Requirement for the Award of

the Degree of

Bachelors of Science in Computer Science

Faculty of Computing
Riphah International University, Islamabad

Date: [30 April 2024]

Final Approval

This is to certify that we have read the report submitted by ***Muhammad Salman Afaq (24775),Usman Afaq (24779), Rafaqat Ahmad (24784)*** for the partial fulfillment of the requirements for the degree of the Bachelors of Science in Computer Science (BSCS). It is our judgment that this report is of sufficient standard to warrant its acceptance by Riphah International University, Islamabad for the degree of Bachelors of Science in Computer Science (BSCS).

Committee:

1 _____

Mr. Muhammad Usman Karim
(Supervisor)

2 _____

Dr. Muhammad Musharraf
(Head of Department)

Declaration

We hereby declare that this document "**Tomato Care**" neither as a whole nor as a part has been copied out from any source. It is further declared that we have done this project with the accompanied report entirely on the basis of our personal efforts, under the proficient guidance of our teachers especially our supervisor **Mr. Muhammad Usman Karim**. If any part of the system is proved to be copied out from any source or found to be reproduction of any project from anywhere else, we shall stand by the consequences.

Muhammad Salman Afaq
24775

Usman Afaq
24779

Rafaqat Ahmad
24784

Dedication

Our final year project is dedicated to our parents, friends and teachers, whose love and support have been our pillars of strength. To our professors and especially supervisor "**Mr. Muhammad Usman Karim**", your guidance has shaped our academic journey.

Acknowledgement

First of all we are obliged to Allah Almighty the Merciful, the Beneficent and the source of all Knowledge, for granting us the courage and knowledge to complete this Project.

We extend our heartfelt gratitude to our project supervisor "**Mr. Muhammad Usman Karim"**, whose unwavering support, invaluable guidance, and continuous mentorship were indispensable to the successful completion of this project. Their dedication and commitment have been a driving force behind our work.

Furthermore, we want to say a big thank you to our family and friends. They have been our constant source of support and motivation, always encouraging us to do our best and be honest and hardworking.

Muhammad Salman Afaq
24775

Usman Afaq
24779

Rafaqat Ahmad
24784

Table of Contents

List of Tables

List of Figures

Abstract

The project, titled **"Tomato Care"** addresses the critical challenge of safeguarding tomato crops against diseases that threaten global agricultural sustainability. Focused on Early Blight, Late Blight, and Leaf Mold, our objective is to develop an efficient automated system for early disease detection, providing farmers with a tool for timely intervention.

Leveraging Convolutional Neural Networks (CNNs) and transfer learning on a diverse dataset, our model achieved a validation accuracy of 95%. Real-world testing demonstrated its robustness, showcasing high sensitivity and specificity.

This Tomato Disease Detection System stands as a practical solution, emphasizing the potential of machine learning in precision agriculture for sustainable food production.

Chapter 1

Introduction

Tomato Care is a complete agricultural solution which is designed to solve the problems of farmers. This advanced platform, which is available for Android, leverages cutting-edge technology to revolutionize the way farmers detect and manage diseases in their tomato crops. With current methods falling short in addressing the **complexities** of tomato crop health, **Tomato Care** aims to provide an efficient and innovative solution to empower farmers. By seamlessly integrating image-based disease detection, treatment recommendations, and real-time updates, **Tomato Care** ensures a holistic approach to crop health management.

This user-friendly platform not only enhances the **accuracy** of disease identification but also encourages a collaborative environment where farmers can make informed decisions for optimal yields. The old ways of finding diseases in tomato plants are not always good enough, and it is hard to find and fix problems quickly. But **Tomato Care** is here to help. It uses advanced technology to make sure we find and understand the diseases in tomato crops better and faster.

1.1 Goals And Objectives

1.1.1 Goals

To utilize artificial intelligence and image analysis to process images of tomato plants, training models to recognize visual cues associated with various diseases.

To make the efforts of farmer less and make the system to detect the disease and suggest the treatment.

1.2.1 Objectives

To develop a system using smart technology to analyze pictures of tomato plants and automatically spot signs of diseases.

Create a monitoring setup that can quickly identify early signs of diseases in tomato crops, enabling timely and targeted interventions to prevent the spread of infections and minimize crop damage.

1.2 Scope of the Project

The project aims to create a user-friendly mobile application tailored for farmers, facilitating the identification of tomato plant diseases through image capture. By leveraging advanced technologies like machine learning, the app will enable precise disease recognition. Beyond mere identification, it will offer valuable recommendations for treatment, empowering farmers to take proactive measures against potential threats to their crops. Ensuring the app remains updated with the latest information on plant diseases is integral to its functionality. Ultimately, the project seeks to support farmers in safeguarding their crops, ultimately contributing to enhanced harvests and bolstered food production on a larger scale.

Chapter 2

Literature Review

2.1 Introduction

Tomato Care is a complete agricultural solution which is designed to solve the problems of farmers. This advanced platform, which is available for Android, leverages cutting-edge technology to revolutionize the way farmers detect and manage diseases in their tomato crops. With current methods falling short in addressing the complexities of tomato crop health, Tomato Care aims to provide an efficient and innovative solution to empower farmers. By seamlessly integrating image-based disease detection, treatment recommendations, and real-time updates, Tomato Care ensures a holistic approach to crop health management. This user-friendly platform not only enhances the accuracy of disease identification but also encourages a collaborative environment where farmers can make informed decisions for optimal yields.

The Tomato Disease Detection market survey involves a thorough analysis of technology trends, and customer needs in the agricultural technology sector. By assessing market size, growth potential, and competitive dynamics, businesses can gain valuable insights for informed decision-making.

2.2 Background and Problem Elaboration

The literature review and technologies overview for Tomato Disease Detection look at what researchers and experts have already studied and developed in this area. They've been using technologies like machine learning to identify diseases in tomatoes. One good thing they use is image recognition, especially with something called CNNs. These technologies help to quickly spot and classify diseases just by looking at pictures of tomato leaves. Also, there are smart sensors that help farmers monitor their crops in real-time, catching diseases early. All of this technology together helps in managing diseases in tomatoes better, reducing losses, and making farming more sustainable. As scientists keep studying and improving these technologies, the way we detect and handle tomato diseases keeps getting better.

2.3 Litreature Review Summary Table

The Table below shows some researched papers

Table 1: Literature Review

Paper ID	Domain	Document	Algorithm	Accuracy	Reference
1	Machine Learning	Tomato Leaf Disease Detection Using Convolutional Neural Networks	Convolutional Neural Networks (CNNs)	91.52%	https://ieeexplore.ieee.org/document/9988540/
2	Deep Learning	Early Detection and Classification of Tomato Leaf Disease Using High-Performance Deep Neural Network	Inception V3, Rainbow concatenation	92.52%	https://www.mdpi.com/1424-8220/21/23/7987
3	Deep Learning	Tomato Leaf Disease Detection using Convolution Neural Network	Convolutional Neural Networks (CNNs), Transfer learning with Inception V3	91.2%	https://www.sciencedirect.com/science/article/pii/S1877050920306906
4	Deep Learning	Disease detection on the leaves of the tomato plants by using deep learning	Convolutional Neural Networks (CNNs)	91%	https://ieeexplore.ieee.org/document/9397001

2.3.1 Market Survey

Table 1.1: Market Survey

Ref	Applications	Tomato-Disease Detection	Treatment Suggestions	Voice and Text	Localized Alerts
1	Detailed tomato cultivation	✓	✕	✕	✕
2	Tomato Cultivation Tips	✕	✓	✕	✕
3	Diagno Plant Tomato	✕	✓	✕	✕

2.4 Problem Statement

Farmers face the critical challenge of timely and accurate detection of diseases in their tomato plants, which directly affects crop yield and livelihoods. Traditional methods of disease identification rely on visual inspection, which can be error-prone and time-consuming. Farmers need an accessible and reliable solution that leverages technology to quickly and effectively identify diseases in tomato plants, enabling them to take proactive measures for disease management and protect their agricultural investments.

Chapter 3

Requirements and Design

3.1 Requirements

In the area of agricultural technology, the search for efficient and accurate disease detection in tomato crops has led to the development of advanced solutions through Requirement Engineering. This process involves systematically capturing, analyzing, and defining the needs and specifications for a technology or system customize to detect diseases in tomatoes. Understanding the specific requirements is crucial for designing effective solutions that address the challenges faced by farmers in managing crop health.

From the types of diseases common in tomato plants to the technological capabilities required for timely detection, Requirement Engineering serves as the foundation for developing strong and user-friendly Tomato Disease Detection systems.

This introduction sets the stage for exploring the detailed requirements essential for creating effective and impactful solutions in the domain of tomato crop health management.

3.1.1 Functional Requirements

3.1.1.1 User / Farmer

ID	Requirements
1.1	User shall be able to Sign Up.
1.2	User shall be able to login to their account.
1.3	User shall be able to view his profile.
1.4	User shall be able to edit his profile.
1.5	User can upload images for disease detection.
1.6	User can access a personalized calendar for plant care.
1.7	User can give feedback.
1.8	User can also interact with the FAQs. It will provide relevant answers.
1.9	User can also view the reports.

3.1.1.2 Admin

ID	Requirements
2.1	Admin shall be able to login into the system.
2.2	Admin shall be able to manage diseases.
2.3	Admin shall be able to manage treatments.
2.4	Admin shall be able manage user's profiles.
2.5	Admin can also manage localized alerts.
2.6	Admin can review the reports.
2.7	Admin shall be able to manage the FAQs.

3.1.2 Non-Functional Requirements

3.1.2.1 Performance

The system should provide real-time or near-real-time processing for disease detection to ensure prompt results for farmers.

Response time for uploading images and receiving results should be within an acceptable range.

3.1.2.2 Security

User authentication and authorization mechanisms should be implemented to ensure that only authorized users (farmers and admins) can access the system.

Data privacy and integrity must be maintained, especially for sensitive information related to the farmer's crops.

3.1.3 Hardware and Software Requirements

3.1.3.1 Hardware Requirements

- Phone
- Server
- Storage

3.1.3.2 Software Requirements

- Database
- Programming Languages
- Framework
- Development Tools
- Version Control

3.2 Proposed Methodology

Tomato Care is a complete agricultural solution which is designed to solve the problems of farmers. This advanced platform, which is available for Android, leverages cutting-edge technology to revolutionize the way farmers detect and manage diseases in their tomato crops. With current methods falling short in addressing the **complexities** of tomato crop health, **Tomato Care** aims to provide an efficient and innovative solution to empower farmers. By seamlessly integrating image-based disease detection, treatment recommendations, and real-time updates, **Tomato Care** ensures a holistic approach to crop health management.

The methodology for simplifying disease detection for tomato plants and enhancing crop yields involves utilizing a user-friendly mobile app interface. Farmers capture photos of their tomato plants using smartphones, and the app employs image processing and machine learning algorithms to identify diseases and assess their severity. Based on the analysis, the app suggests appropriate treatment strategies tailored to the specific disease detected. It features a user-friendly interface with FAQs function.

Additionally, the app utilizes localized alerts and notifications about potential disease risks, enabling farmers to take proactive measures. By empowering farmers with accessible tools and information, the methodology aims to streamline disease management, ultimately contributing to improved crop yields and agricultural sustainability.

3.3 System Architecture

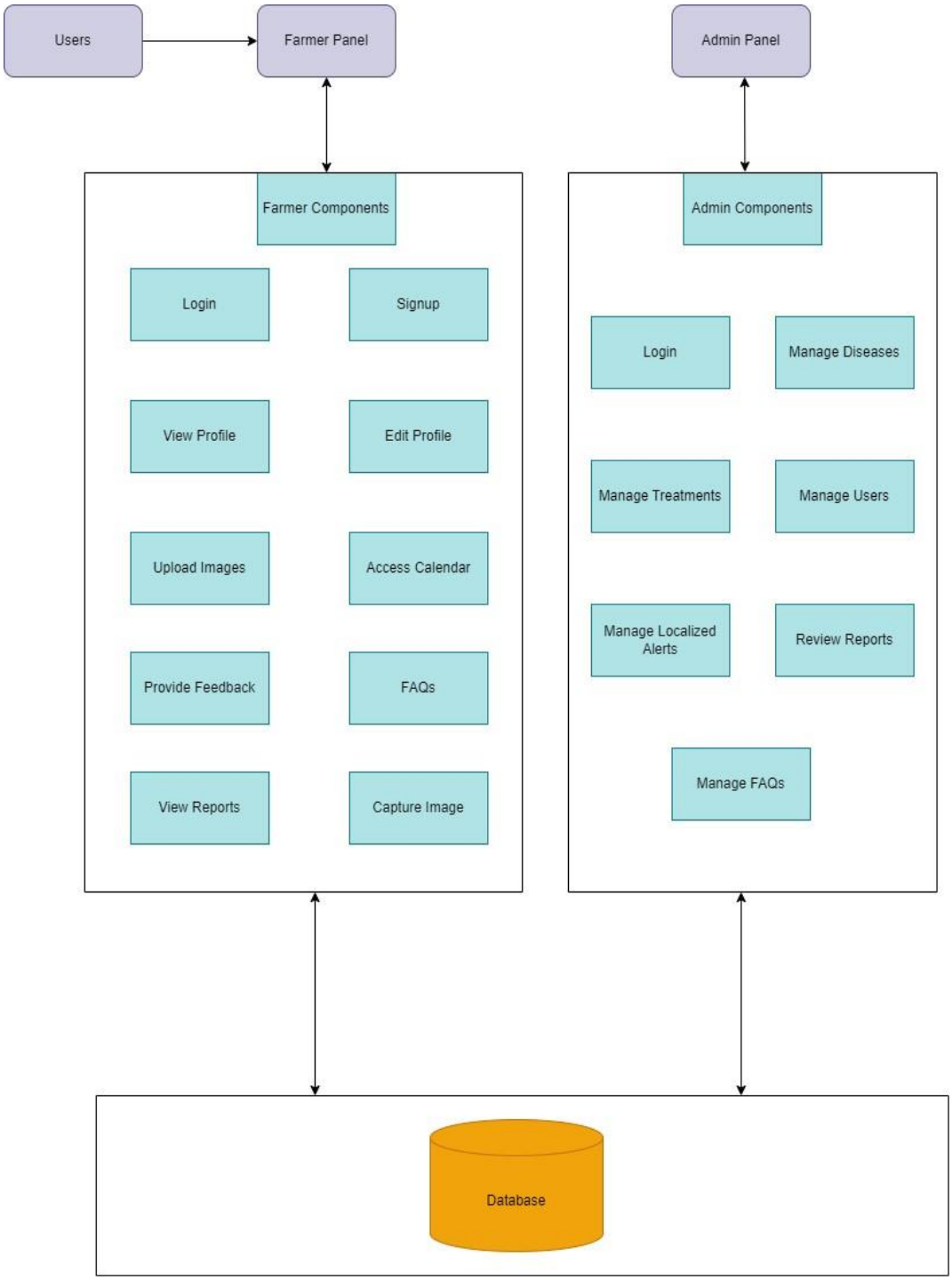

Figure 1: Architecture Diagram

3.4 Use Cases

3.4.1 Farmer

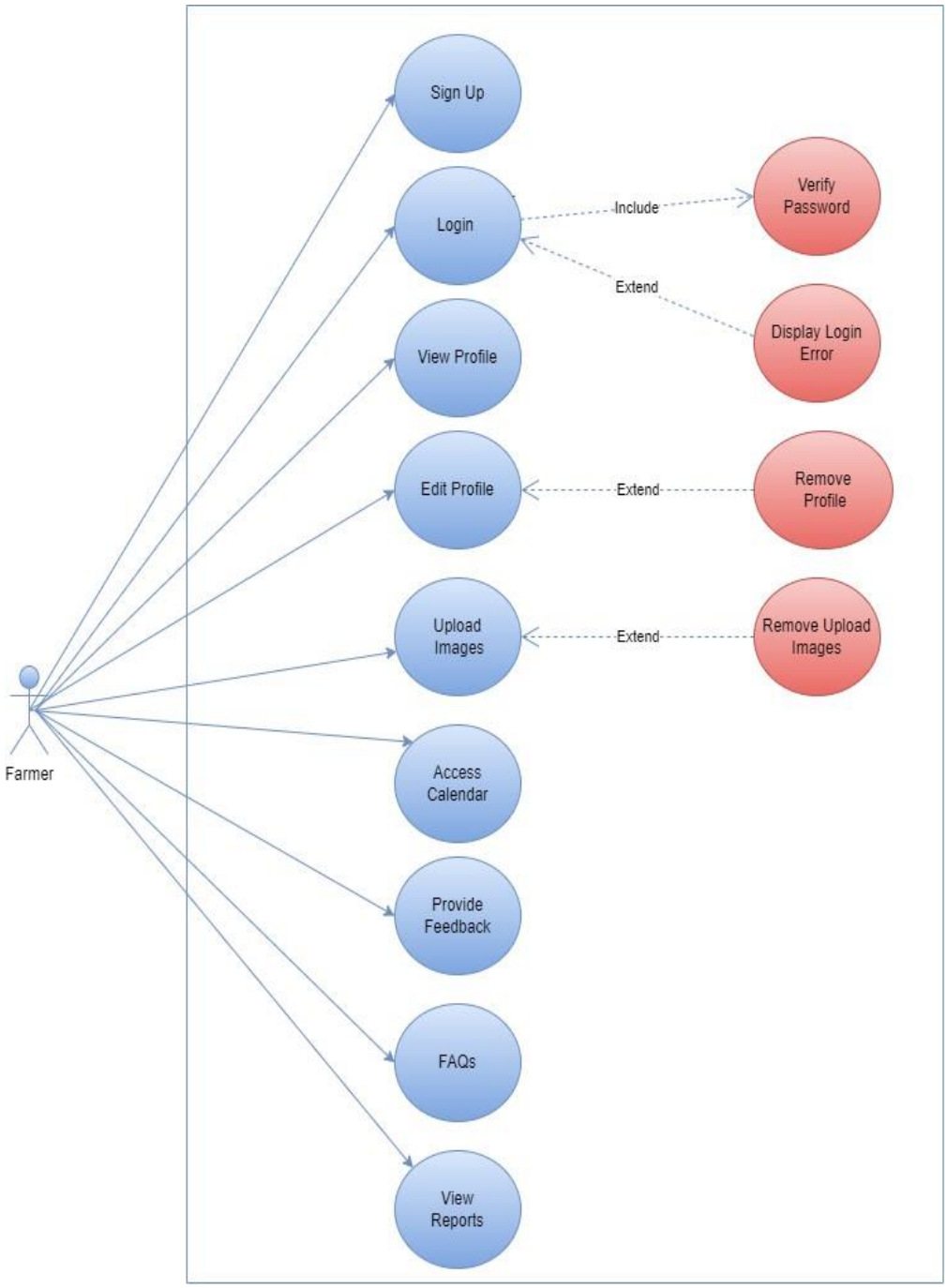

Figure 1.1: Farmer Use Case

3.4.2 Admin

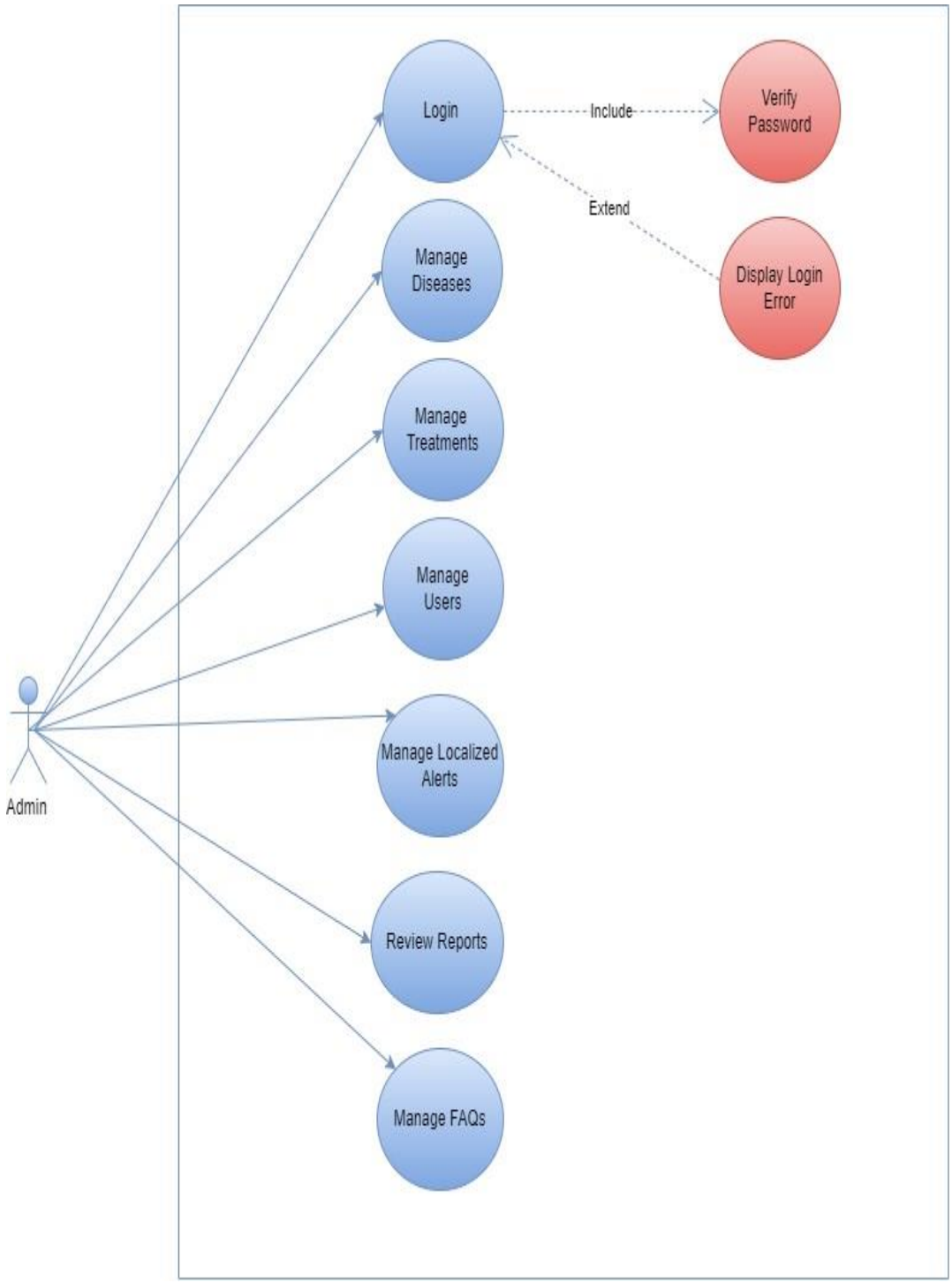

Figure 1.2: Admin Use Case

3.5 Fully Dressed Use Cases

3.5.1 Sign Up

Name	**Sign Up**
Actors	Farmer
Summary	The farmer registers a new account on the platform.
Pre-Conditions	The farmer is not logged in to the platform.
Post-Conditions	The farmer creates an account with basic information like name, email, phone number, and password.
Special Requirements	The email address should be unique and valid.

Basic Flow			
Actor Action		**System Response**	
1	The farmer opens the Sign Up page.	2	The system displays the Sign Up form.
3	The farmer fills out the form with required information.	4	The system validates the information and creates an account upon successful validation.
Alternative Flow			
1	If the entered email is already registered, the system displays an error message and prompts the farmer to use a different email.	2	If the password doesn't meet the complexity requirements, the system prompts the farmer to choose a stronger password.

3.5.2 Login

Name	**Login**
Actors	Farmer, Admin
Summary	The farmer or admin accesses their account by providing login credentials.
Pre-Conditions	The user has a registered account.

Post-Conditions	The user is logged into the platform and has access to their account features.		
Special Requirements	The password should be entered correctly.		
Basic Flow			
Actor Action		**System Response**	
1	The user opens the Login page.	2	The system displays the Login form.
3	The user enters their email address and password.	4	The system validates the credentials.
Alternative Flow			
1	If the entered credentials are incorrect, the system displays an error message and allows the farmer to attempt login again.	2	If the farmer forgets the password, there is a "Forgot Password" option that allows them to reset the password

3.5.3 View Profile

Name	**View Profile**		
Actors	Farmer, Admin		
Summary	The user views their profile information.		
Pre-Conditions	The user having securely authenticated their identity.		
Post-Conditions	The user can see their profile details like name, email address, phone number, and other relevant information.		
Special Requirements	None		
Basic Flow			
Actor Action		**System Response**	
1	The user clicks on the "Profile" link or icon.	2	The system displays the user's profile page with their information.
Alternative Flow			
1	None		

3.5.4 Edit Profile

Name	**Edit Profile**
Actors	Farmer, Admin
Summary	The user updates their profile information.
Pre-Conditions	The user has an access of his/her profile.
Post-Conditions	The user's profile information is updated with the changes.
Special Requirements	None

Basic Flow			
Actor Action		**System Response**	
1	The user clicks on the "Edit Profile" button.	2	The system displays the editable profile form with the user's current information.
3	The user modifies the desired information on the form.	4	The system validates the input and updates the user's profile with the changes.

Alternative Flow	
1	If the farmer attempts to submit invalid information, the system displays an error message and prompts the farmer to correct the information.

3.5.5 Upload Images

Name	**Upload Images**
Actors	Farmer
Summary	The farmer uploads images of their plants to identify potential diseases.
Pre-Conditions	The farmer accepts the permission to access camera.
Post-Conditions	The system analyzes the images and provides the farmer with potential disease diagnoses and information.

Special Requirements	The image format should be supported by the system.		
Basic Flow			
Actor Action		**System Response**	
1	The farmer clicks on the "Disease Detection" feature.	2	The system displays the image upload interface.
3	\The farmer selects and uploads the images of their plants.	4	The system analyzes the images using image recognition technology.
Alternative Flow			
1	If the uploaded images are of an unsupported format or size, the system prompts the farmer to upload valid images.	2	If there is an issue with the image processing, the system notifies the farmer and recommends re-uploading the images.

3.5.6 Access Calendar

Name	**Access Calendar**		
Actors	Farmer		
Summary	The farmer receives a personalized calendar with recommended tasks and schedules for their specific crops based on their location and climate.		
Pre-Conditions	The farmer is ready to access the calendar for planning crops and tasks.		
Post-Conditions	The farmer has a customized calendar with reminders for essential plant care activities like: Watering, Fertilization, Pest and disease control, Harvesting		
Special Requirements	Plant care calendar section.		
Basic Flow			
Actor Action		**System Response**	
1	The system automatically generates a personalized calendar after the farmer provides required information.	2	The farmer can filter and prioritize tasks based on urgency or category.

3	The calendar displays tasks with dates, deadlines, and instructions for each activity.	4	The farmer can set reminders for upcoming tasks.
Alternative Flow			
1	If there are no personalized recommendations for the farmer, the system displays a message suggesting that the farmer has no specific activities scheduled and encourages them to check back later.		

3.5.7 Provide Feedback

Name	**Provide Feedback**		
Actors	Farmer, Admin		
Summary	The user provides feedback on the platform's features and functionalities.		
Pre-Conditions	The application stands ready to receive and respond to the user's feedback.		
Post-Conditions	The feedback is communicated to the admin for review and consideration.		
Special Requirements	Feedback form.		
Basic Flow			
Actor Action		**System Response**	
1	The user opens the "Feedback" section.	2	Describe the issue or suggestion in detail. Attach screenshots or relevant media for clarity. The system sends the feedback to the admin for review.
3	Submits the feedback.	4	System acknowledges the feedback.
Alternative Flow			
1	If the feedback form submission fails due to technical issues, the system displays a message informing the farmer about the problem		

	and advises them to try again later.

3.5.8 FAQs

Name	**FAQs**
Actors	Farmer, Admin
Summary	The FAQs functionality will allow the user to ask the questions and will get relevant answers about the disease detection.
Pre-Conditions	The User can interact with the FAQs
Post-Conditions	The user receives answers about the asked questions
Special Requirements	Security measures to protect sensitive information. Also Search functionality for users to quickly find relevant FAQs.

Basic Flow			
Actor Action		**System Response**	
1	User accesses the FAQs page on the website.	2	System displays a list of frequently asked questions.

Alternative Flow	
1	User cannot find the answer to their question in the FAQs.

3.5.9 View Reports

Name	**View Reports**
Actors	Farmer, Admin
Summary	The user accesses and reviews reports generated by the platform to gain insights into their farming practices and crop performance.
Pre-Conditions	Users are all set to open and check out reports in the application.
Post-Conditions	The user gains valuable insights into their farming practices and crop performance based on the generated reports.

Special Requirements	Reports should be generated regularly and updated with new data.		
Basic Flow			
Actor Action		**System Response**	
1	The user accesses the reporting section of the platform.	2	The system displays available reports, categorized by type, such as Crop Performance Reports, Resource Consumption Reports, Soil Health Reports, Environmental Impact Reports
3	The user selects a specific report to view.	4	The system displays the report in an interactive format, allowing the user to Filter and analyze data based on specific criteria.
Alternative Flow			
1	The system prompts the user to provide additional information or filter parameters to refine the report.	2	The user provides the requested information or modifies filter settings.
3	The system updates the report with the refined data.		

3.5.10 Login

Name	**Login**
Actors	Admin
Summary	The administrator accesses the platform's administration panel to manage various functionalities.
Pre-Conditions	The administrator has a registered account and valid login credentials.
Post-Conditions	The administrator has access to all administrative features.
Special Requirements	Multi-factor authentication and secure session management.
Basic Flow	
Actor Action	**System Response**

1	The administrator opens the Login page.	2	The system displays the Login form.
3	The administrator enters their email address and password.	4	The system validates the credentials.
5	If credentials are valid	6	The system logs the administrator into the platform.
7	If credentials are invalid	8	The system displays an error message.
Alternative Flow			
1	The system displays an error message and prompts the user to re-enter their credentials.	2	The system informs the user that their account is locked and provides instructions for unlocking it.
3	The system sends a verification code to the administrator's registered phone number or email address for additional security.		

3.5.11 Manage Diseases

Name	**Manage Diseases**
Actors	Admin
Summary	The administrator adds, edits, and updates information about various diseases affecting crops within the platform.
Pre-Conditions	The admin gains the ability to manage diseases within the application.
Post-Conditions	The administrator manages the disease database, ensuring accurate and up-to-date information for users.
Special Requirements	Integration with a dynamic disease database, and support for classification.

Basic Flow			
Actor Action		**System Response**	
1	The administrator opens the Disease Management section.	2	The system displays a list of existing diseases.

3	The administrator can: Add a new disease, Edit an existing disease, Delete a disease:	4	The system updates the disease database accordingly.
Alternative Flow			
1	The system prompts the administrator to complete all required fields before submitting the new disease information.	2	The system warns the administrator that the disease already exists and suggests potential actions, such as merging information or creating a new entry with different details.
3	The system validates the format and accuracy of the imported data before adding it to the platform.		

3.5.12 Manage Treatments

Name	**Manage Treatments**
Actors	Admin
Summary	The administrator adds, edits, and updates information about various treatment options for different diseases.
Pre-Conditions	The admin is equipped to administer and oversee treatments within the application.
Post-Conditions	The administrator maintains a comprehensive list of treatment options for users to access and utilize.
Special Requirements	Integration with a treatment database, update mechanism, and metrics for effectiveness.

Basic Flow			
Actor Action		**System Response**	
1	The administrator opens the Treatment Management section.	2	The system displays a list of existing treatment options.

3	The administrator can: Add a new treatment, Edit an existing treatment, Delete a treatment	4	The system updates the treatment database accordingly.
Alternative Flow			
1	The system requires the administrator to provide clear warnings and disclaimers about potential risks associated with the treatment.	2	The system allows the administrator to edit and update the treatment information to ensure accuracy and currency.
3	The system prompts the administrator to provide more reliable references and evidence to support the treatment information.		

3.5.13 Manage Users

Name	**Manage Users**		
Actors	Admin		
Summary	The administrator reviews, approves, and manages user accounts within the platform.		
Pre-Conditions	The admin is empowered to oversee and manage users within the application.		
Post-Conditions	The administrator ensures a safe and productive environment for platform users.		
Special Requirements	Role-based access control, activity logs, and customizable profiles.		
Basic Flow			
Actor Action		**System Response**	
1	The administrator opens the User Management section.	2	The system displays a list of all registered users.
3	The administrator can: Review user profiles, Approve or reject user registrations, Suspend or ban users, Assign user roles and permissions	4	The system updates user accounts and access accordingly.
Alternative Flow			
1	The system prompts the user to complete all required	2	The system may request additional documentation or

	information before submitting the registration request.		proof of identity to verify the user's information.
3	The system ensures that user privacy is protected and sensitive information is not publicly accessible.		

3.5.14 Manage Localized Alerts

Name	**Manage Localized Alerts**
Actors	Admin
Summary	The administrator creates and sends localized alerts to users based on specific criteria.
Pre-Conditions	The admin gains direct authority to manage localized alerts in the application, demonstrating a high level of control and responsiveness.
Post-Conditions	Users receive timely and relevant notifications about potential threats or important information related to their location.
Special Requirements	Geolocation services, weather API integration, and user preferences.

Basic Flow			
Actor Action		**System Response**	
1	The administrator opens the Localized Alert Management section.	2	The system displays tools for creating and sending alerts.
3	The administrator can: Define alert criteria, Create alert messages, Schedule alert delivery.	4	The system delivers alerts to relevant users via push notifications, email, or other channels.
Alternative Flow			
1	The system informs the administrator that they need more data to define the alert criteria effectively.	2	The system allows users to control their notification preferences and manage the types of alerts they receive.
3	The system alerts the administrator of any technical issues affecting alert delivery and provides troubleshooting steps.		

3.5.15 Review Reports

Name	**Review Reports**
Actors	Admin
Summary	The administrator analyzes platform usage data and user feedback to gain insights and improve the platform.
Pre-Conditions	The administrator is logged into the platform and has access to reporting dashboards.
Post-Conditions	The administrator identifies trends, patterns, and areas for improvement based on data analysis.
Special Requirements	Comprehensive reporting tools and integration with analytics.

Basic Flow			
Actor Action		**System Response**	
1	The administrator selects specific reports from a dashboard.	2	The system displays visualizations and data analysis reports on various aspects: User activity, Content engagement, Disease management, Chatbot interactions.
3	The administrator analyzes the data to identify trends, patterns, and areas for improvement.	4	The system logs the administrator's actions and updates the platform accordingly.
Alternative Flow			
1	The administrator investigates further to identify the underlying causes and potential implications.	2	The administrator prioritizes addressing these issues and implementing improvements.
3	The administrator takes immediate action to address the identified risks and protect user data.		

3.5.16 Manage FAQs

Name	**Manage FAQs**		
Actors	Admin		
Summary	Admins can log in to the system and manage the FAQs by editing, deleting, or adding new entries.		
Pre-Conditions	Admin will manage the FAQs in the system		
Post-Conditions	FAQs are successfully managed, and the system reflects the changes.		
Special Requirements	None		
Basic Flow			
Actor Action		**System Response**	
1	Admin navigates to the "Manage FAQs" section.	2	System displays a list of existing FAQs with options to edit, delete, or add new entries.
Alternative Flow			
1	None		

3.6 Database Design

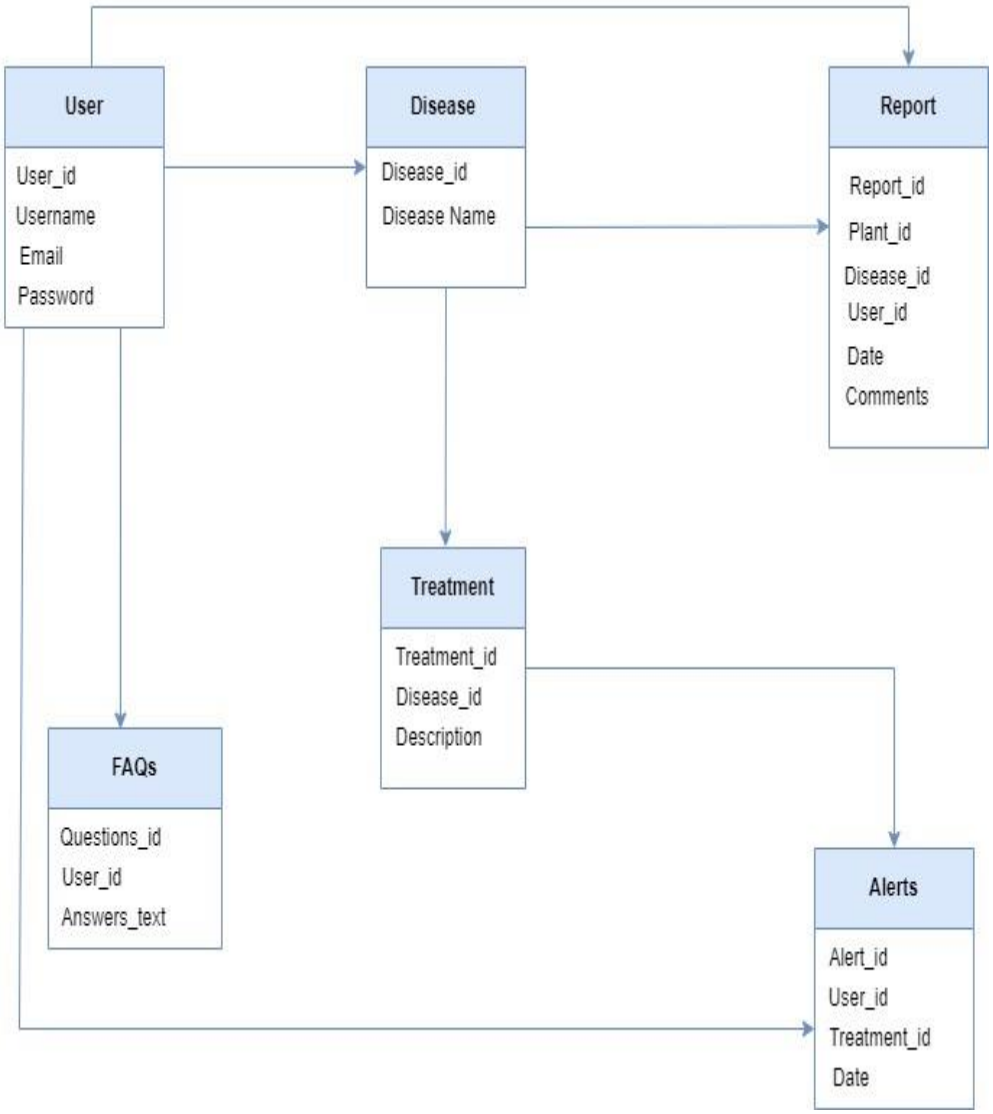

Figure 1.3: Database Design

3.7 Methodology Diagram

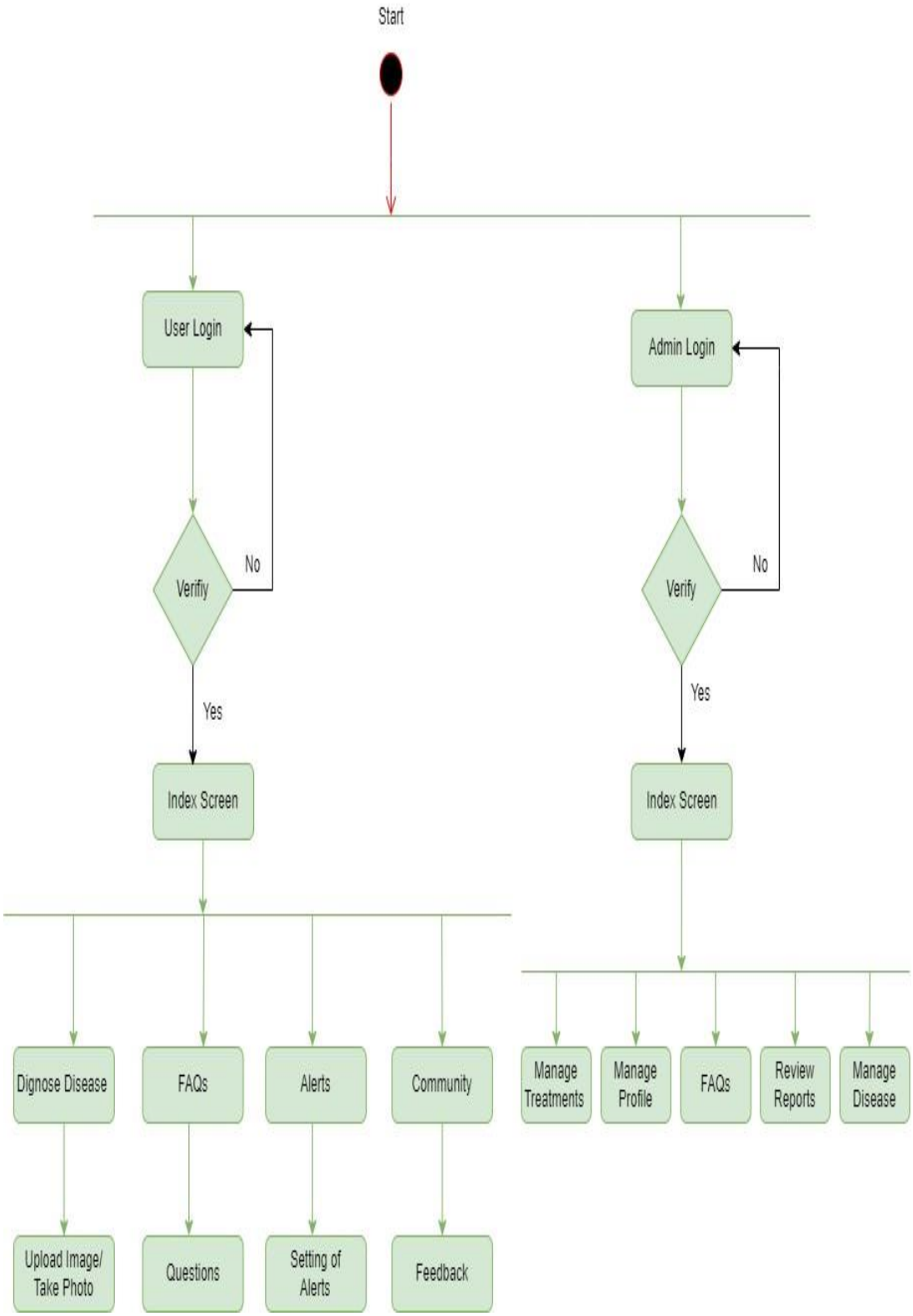

Figure 5: Methodology Diagram

Chapter 4

Implementation and Test Cases

4.1 Test Cases and Descriptions

4.1.1 Test Data

Test Data	TD-1
Form	Login
Stakeholder	User
Field	Email
Technique	Equivalence Partitioning
Valid	• Correct length (user@example.com) • Includes numeric characters (user123@example.com). • Includes country code (user@example.co.uk)
Invalid	• Does not contain '@' (userexample.com).Include characters • Includes characters (user@exampl*e.com).Not End with '.' • Includes special characters (user@ex!ample.com). • Does not end with '.' (user@examplecom).

Test Data	TD-2
Form	Login
Stakeholder	User
Field	Password
Technique	Equivalence Partitioning
Valid	• Password length should be >=6

	• Includes two special characters • Includes uppercase and lowercase character • Includes one numeric character
Invalid	• Password length < 6 • No uppercase and lowercase characters • No special character • No numeric character

More Test Data And Test Cases

Chapter 5

Experimental Results and Analysis

Model Accuracy graphs

Chapter 6

Conclusion and Future Directions

The tomato disease detection project aims to streamline the process of identifying and managing diseases affecting tomato plants to improve crop yields and agricultural sustainability. The proposed solution involves a user-friendly mobile app interface where farmers can capture photos of their tomato plants using smartphones. These images are then analyzed using image processing and machine learning algorithms to identify diseases and assess their severity. Based on the analysis results, the app provides farmers with tailored treatment recommendations for the detected diseases.

References

List all important sources of information which have been consulted for this project