Plants Care Application

Graduation Project, Part-I (CS)

Computer Science Department

FCI LUXOR UNIVERSITY

Supervised By:

Dr. Mohmmed Atta

Submitted by:

1. Hany Abdou Saad
2. Mahmoud Ahmed Abdullah
3. Hazem Taha Basher
4. Moaz Wahed Ramadan
5. Mohmmed Ahmed Hashem
6. Hassan Nour Hassan

1.1 Abstract

The global food demand is expected to be increased sharply by 2100 in addition to the population size is predicted to be over 9 billion people, so human society needs to increase food production by an estimated 70% by 2050 to feed the expected population. On the other hand, Infectious diseases reduce the potential yield by an average of 40% which led to that many farmers in the developing world experiencing yield losses as high as 100%. Because of these problems, a farmer is usually using the normal solution, which is hiring agricultural engineers, to help him with these problems which are costly.

By the advancement of technology, we can make use of that by providing methods to help the farmers to get rid of these issues, such as using an application that can take a photo for a plant, and the application, in turn, presents you with the result of this plant is infected with a pest or disease, or it is healthy. This is our application, but it doesn't just do this process only, you can also ask about anything related to your plant by creating a post in the post area and get answers from other users who have a previous experience, finally, you can calculate the amount of required fertilizer to the agriculture land based on the type of plant and land area on different periods of plant growth.

Table of Contents

List of Figures

Content **Page**

1.2 Key Words

ML Model	A machine learning model is a file that has been trained to recognize certain types of patterns. You train a model over a set of data, providing it an algorithm that it can use to reason over and learn from those data.
Classifier	Classifier is an algorithm that automatically orders or categorizes data into one or more of a set of “classes”.
Fertilizer appropriate	They are the percentages that the application will give, which is a percentage of each fertilizer so that the farmer puts these percentages to get the best fertilizer without an increase or decrease in the amount of fertilizer.

1.3 Introduction

The global population has been expanding rapidly for many years, standing at around 7.3 billion in 2016, due to several factors, such as advanced maternity and healthcare so human society needs to increase food production by an estimated 70% by 2050 to feed an expected population. To feed a world population growing by up to 160 people per minute, with >90% of them in developing countries, will require an astonishing increase in food production. Unfortunately, Infectious diseases reduce the potential yield by an average of 40% with many farmers in the developing world experiencing yield losses as high as 100%. Plants act as an important resource for everyone in terms of food. If disease occurs, then it is very necessary to detect plant diseases in the early stage. There exist many models that help in detecting and classifying plant diseases. In recent days, Machine learning is a great way to detect diseases it gives the computers the ability to learn without being explicitly programmed. There exist many models in machine learning to detect plant diseases. Some of them are K-means, KNN for classifying the leaves are healthy or diseased. Later, there is advancement in the field of machine learning which results in the evolution of deep learning. Deep learning algorithms are learning the features from input images during the training stage and exhibit results with suitable metrics. The working of deep learning is as follows in deep learning the information is passed through some layers. The output of the previous layer is given as input to the next layer. It passes the learned features from one layer to the next layers using activation functions till the output layer produces the desired outcomes. This idea can be extended for plant disease detection systems to manage and monitor wirelessly in large-scale agriculture production with the use of drones for surveillance, the use of sensors for managing the quantity of water, as well as fertilizers and light necessary for a qualitative production outcome. For this issue using our app, you can take a photo of the plant and by- processing, if the plant is infected, or not if it is then showing the type of disease and methods of organic and chemical treatment.

In addition, you can create a post according to the category you have

chosen (plant type) or others. Asking for help or posting some information.

Finally, the calculator part of the amount of fertilizer you use is based on the nature of the plant and the area of the cultivated land. Plus, the best fertilization plan you can use.

1.4 Background

The ratio of food demand overproduction is the main factor affecting the level of food security. The global population has been expanding rapidly for many years, standing at around 7.3 billion in 2016, due to many factors, such as advanced maternity and healthcare. However, the rise brings with it several challenges around global sustainability, including the need for more food. Agriculture contributes to 6.4% of the entire world's economic production. In at least nine countries of the world, agriculture is the dominant sector of the economy.

Agriculture not only provides the fuel for billions of people but also employment opportunities for many people. The agricultural industries are seeking innovative approaches for improving crop yielding because of unpredictable climatic changes, the rapid increase in population growth, and food security concerns. Thus, artificial intelligence in agriculture also called “Agriculture Intelligence” is progressively emerging as a part of the industry's technological revolution.

The spread of agricultural diseases that destroy all agricultural crops and lack of arable land are the most problem which faces our planet, today's so we try by our project to use modern solutions using internet and technologies to beat these problems and increasing the productivity of the crops and trying to help farmers and providing them with the necessary experience to identify diseases before they occur and giving them suggestions to ensure the best productivity.

ML-driven image processing allows farmers to rely upon digital tools to recognize weed species and to determine which crops are healthy and which ones are infested with disease caused by fungi, bacteria, or viruses

We focus on using machine learning algorithms and computer vision to build models which have the ability to detect diseases and give users the best solution to its problem and try also to connect experienced agricultural engineers with simple farmers which helps to decrease time and cost to anyone who needs the advice by posting any questions, these posts appear to any user by its interesting and he can interact with these posts, save them if want.

2.1 Related Work

Through the application of artificial intelligence (AI) and machine learning (ML), growers can access increasingly sophisticated data and analytics tools, which enables better decisions, improved efficiencies, and reduced waste in food, all while minimizing negative environmental consequences.

Name	Authors	Year	Abstract
An Improved Deep Learning model for plant disease detection	-Anjanadevi Bondalapati	March 2020	In this paper, we have focused on plant data images in agricultural field. Agriculture is one of major living source in India. To increase the yield by preventing diseases and detection of diseases place major role in agriculture domain. By using Improved and customized DCNN model (improved-detect), We trained plant Doc and plant village datasets. Mainly we used Tomato, Corn and potato plant for model training and testing. we have experimented on plant image data set tomato leaves both healthy and diseased ones. Experimental results are compared with state of the architectures like Mobile Net, Dark Net-19, ResNet and proposed model in location and detection of plant diseases.

Using Deep Learning for Image-Based Plant Disease Detection	-Sharada Prasanna Mohanty. -David Peter Hughes. -Marcel Salathe	April 15 2016	Using a public dataset of 54,306 images of diseased and healthy plant leaves collected under controlled conditions, we train a deep convolutional neural network to identify 14 crop species and 26 diseases (or absence thereof). The trained model achieves an accuracy of 99.35% on a held-out test set, demonstrating the feasibility of this approach. When testing the model on a set of images collected from trusted online sources - i.e., taken under conditions different from the images used for training the model still achieves an accuracy of 31.4%. While this accuracy is much higher than the one based on random selection (2.6%), a more diverse set of training data is needed to improve the general accuracy.

Plant Disease Detection with Deep Learning and Feature Extraction Using Plant Village	-Mohamet Faye. -Chen Bingcai. -Kane Amath Sada	January 2020	The combination of high-end smart-phones and computer vision via Deep Learning has made possible what can be defined as “smartphone-assisted disease diagnosis”. In the area of Deep Learning, multiple architecture models have been trained, some achieving performance reaching more than 99.53%. In this study, we evaluate CNN’s architectures applying transfer learning and deep feature extraction. All the features obtained will also be classified by SVM and KNN. Our work is feasible using the open-source Plant Village Data set. The result obtained shows that SVM is the best classifier for leaf disease detection.

Plant Disease Detection Using Machine Learning	-Shima Ramesh Maniyath. -Vinod P V	April 2018	we make use of Random Forest in identifying between healthy and diseased leaf from the data sets created. Our proposed paper includes various phases of implementation namely dataset creation, feature extraction, training the classifier and classification. The created datasets of diseased and healthy leaves are collectively trained under Random Forest to classify the diseased and healthy images. For extracting features of an image, we use Histogram of an Oriented Gradient (HOG). Overall, using machine learning to train the large data sets available publicly gives us a clear way to detect the disease present in plants in a colossal scale

Plant Disease Detection Using Machine Learning Algorithms	-P. Prathusha. -K. E. Srinivasa -K. Srinivas	July 2020	Machine learning is a trending area where the technological benefits can be imparted to the agriculture field also. It is rather inexpensive to detect the diseases in plants using machine learning techniques rather than using chemical pesticides. This paper makes a review on the existing techniques and suggests the best technique which can be implemented by farmers to recognize the disease faster and which proves to be economical to them. In this work we use KNN algorithm which is one of the best machine learning algorithms.
Plant Disease Detection and Classification Using Deep Neural Networks	-Aravindhan Venkataramanan. -Pooja Agarwal	August 2019	we present a Deep Learning approach to detect and classify plant diseases by examining the leaf of a given plant. The classification is performed in multiple stages to eliminate possibilities at every stage, hence providing better accuracy during predictions. A YOLOv3 object detector is used to extract a leaf from the input image. The extracted leaf is analysed through a series of ResNet18 models. These ResNet18 models were trained using transfer learning. One layer identifies the type of leaf and the following layer checks for the possible diseases that could occur in the plant.

Plant disease detection and its solution using image classification	G.Saradhambl. R. Dhivya. S. Latha. R. Rajesh.	January 2018	We propose an enhanced k-mean clustering algorithm to predict the infected area of the leaves. A colour-based segmentation model is defined to segment the infected region and placing it to its relevant classes. Experimental analyses were done on samples images in terms of time complexity and the area of infected region. Our project is used to detect the plant diseases and provide solutions to recover from the disease. It shows the affected part of the leaf in percentage. We planned to design our project with voice navigation system, so a person with lesser expertise in software should also be able to use it easily.

Image-Based Detection of Plant Diseases: From Classical Machine Learning to Deep Learning Journey	-Rehanullah Khan. -Khalil Khan. -Waleed Albattah. -Ali Mustafa Qamar	June 2021	The technology used in medical procedures has not been adequate to detect all diseases on time, and that is why some diseases turn out to become pandemics because they are hard to detect on time. Our focus is to clarify the details about the diseases and how to detect them promptly with artificial intelligence. We discuss the use of machine learning and deep learning to detect diseases in plants automatically. Our study also focuses on how machine learning methods have been moved from conventional machine learning to deep learning in the last five years. Furthermore, different data sets related to plant diseases are discussed in detail. The challenges and problems associated with the existing systems are also presented.

Deep learning models for plant disease detection and diagnosis	-Konstantinos Ferentinos	February 2018	convolutional neural network models were developed to perform plant disease detection and diagnosis using simple leaves images of healthy and diseased plants, through deep learning methodologies. Training of the models was performed with the use of an open database of 87,848 images, containing 25 different plants in a set of 58 distinct classes of [plant, disease] combinations, including healthy plants. Several model architectures were trained, with the best performance reaching a 99.53% success rate in identifying the corresponding [plant, disease] combination (or healthy plant).

3.1 Domain Analysis

The widespread distribution of smartphones among crop growers around the world with an expected 5 billion smartphones by 2020 offers the potential of turning the smartphone into a valuable tool for diverse communities growing food. One potential application is the development of mobile disease diagnostics through machine learning and crowdsourcing.

The main issue: is trying to help farmers know the type of pests that affect their crops, in addition to knowing the appropriate treatment for this pest as soon as possible. Some new farmers also suffer from lack of experience in some agricultural matters, so they need someone to exchange experience within order to have the best crop. It can be obtained in these conditions and using the appropriate tools and equipment available, present an experienced community where people can help each other in the faced problems.

Identification of the pest: The farmer will be able to take a picture with his smart phone of the leaf of the plant that he wants to know the type of pest infecting it. And then he uploads this image to the application on his phone so that the application analyzes the image and gives him the results of its analysis of the image and gives him the best solutions and suggestions to treat this scourge.

Applying Machine learning algorithms and classifiers, they showed promising results in image classification and decision-making to assist growers in their diagnosis, which will serve as a boost in improving plant care services through effective image analysis of symptoms (pests) suffered by the plant.

Helping others/asking for help: Any farmer or anyone who wants opinion or even advice on his plant or anything related to the plant he is interested in can ask a general question to all present or users of the application and one of them will respond to him, which increases him with general information, or it saves him the trouble of searching in vain.

Fertilization: Calculating the appropriate amount of fertilizer for the cultivated land; some plants suffer from a lack or increase in the percentage of fertilizer used, so the application will provide the correct standards that you can use in the cultivated land area to obtain the best fertilization.

3.2 Risks and Constrains

➢ Risks

Risk	Strategy	Priority
User may not know how to use the application	Show electronic pictures to how use the application at the first use.	High
Malfunction with the servers that host the database, the posts, and users' information	Hosting the database and the several servers.	High
User may scan an untrained picture or unclear one to the application	We try to show him the related details about his picture.	High
Malfunction with the server or the API hosting the Machine Learning Model	Hosting the model and the API on several servers.	High
Internet for signed user may be failed	We will show him the saved posts and details about pictures he scanned them before.	High

➢ Constrains

- The application should provide friendly and easy interface with not many icons and images to help people who don’t know use the app directly.
- User should scan objects clearly as possible
- User should have internet to deal with all application features.
- User should have the application on his device
- Maintenance of an end-to-end encryption mechanism for providing confidentiality is a must.

3.3 Project Plan

phase	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
Gathering Information									
Define Requirements									
analysis									
design									
implementation									
Develop ML Model System									
Testing and Final Discussion									

3.4 Quality Assurance Plan

Black box: In this stage we use test dataset as input to our ML model to ensure the accuracy of output of the system.

White box:

Unity Testing: - In this stage of testing, we will take every component of our system such as web service, machine learning model, android application to test them separately.

Integration Testing: - In this stage of testing, we will take every component of our system such as web service, machine learning model, android application to test them separately.

Validation Testing: - Validation testing is the process of ensuring if the tested and developed application satisfies its functionality requirements. The business requirement logic or scenarios must be tested in detail. All the critical functionalities of an application must be tested here.

Alpha: - In this part, a group of testers in our team test the product in a laboratory environment to ensure efficiency of product and fix errors.

Beta: - At this stage we will test the application on real users, farmers and people who have some plants at home, retrieve feedback to our team.

3.5 System Requirements

3.5.1 Functional requirements

1- A user interface that enables users create accounts, add their information, the planets they are interested and the location of the users.

2- Verification email is sent to user whenever he/she registers for the first time on some software system.

3- The application allows user to enter password and username to login to app.

4- Authentication of a user when he/she tries to log into the system.

5- The application must store all Users' data

6- The application shows to the user all posts of plants in which he is interested.

7- Machine learning model to allow user to capture / upload the plant image for plant type detection and classification. The model is also used for plant examination from pests and diseases, if plant is diseased, it returns details about this disease and recommends Treatments.

8- A Server stores users' data and Machine Learning Models.

9- The user enters the plant type and agricultural area, and the application outputs the best time plan and quantity of fertilizer to grow well.

3.5.2 Nonfunctional requirements

1- Emails should be sent with a latency of no greater than 12 hours.

2- The application shows the post by the order of the tag plants and the location of the users.

3- User can search for posts with plant tag.

Performance requirement

- The application must respond any operation in less than 4 seconds.

Safety and security requirements

- The application must not affect, harm, or damage users and their mobiles.

Availability

- The application must be available within 24 hours every day.

Usability

- The application allows to users interact with the application to achieve required goals effectively and efficiently.

3.6 Techniques

ML Model: Applying Machine learning algorithms and classifiers, they showed promising results in image classification and decision making to assist growers in their diagnosis, which will serve as a boost in improving plant care services through effective image analysis of symptoms (pests) suffered by the plant.

Application components:

- ❖ **Login page:** This page has the username and password, and it also has a button for the registration page if the user is not registered.
- ❖ **Registration page:** A page that enables the user to register, to be able to use the application.
- ❖ **Plant Fertilization Page:** Every plant has certain conditions in its composting, and that's why fertilizing a plant is different from the other and that the app will provide you that you know the appropriate amounts of fertilizer in relation to the area of the land for each plant.
- ❖ **Posts Page:**

- each user will be able to view other people’s questions, advice, and responses to them, according to the latest date of the user interested plants, so that you can find environmental content that is close to him in the environment, climate, plants, and crops that you are related to, so that it will be more useful to you.
- He can also create a question (post) or inquiry that will appear to all users of the application; If you want to search for specific questions or inquiries related to a particular plant, you can search by the plant tag that will be available in any question or consultation.
- Anyone can interact with the post and comment on it.

3.7 System Request

The application helps framers to improve the quality of the plants by providing them good care, fertilizer and tips. It also helps farmers specially who grow a small area of the land or who grow certain plants at home to save money from consulting agricultural engineers, which makes it a good saver of money for those who do not have the financial ability to consult. The application is also a time saver, because it contributes to obtaining consultations quickly through the interaction of specialists on the post. This makes the application very important in critical times for plants that need quick consultations before they die.

➢ Functionality

1- The Ability to Examinate plants and recommend a treatment for free.

2- The speed of obtaining the plants diseases.

3- The ability to generate full reports about plant diseases and treatments.

4- The ability to share benefits by other users.

5- The ability to get best time plan and best fertilizer for the plant to get a good yield.

➢ Expected Value

1- Less-cost plants examinations process.

2- Lost cost plant cultivation plan.

3- More organized posts about benefits of plants.

4- Ease-to-use application and friendly User-interfaces.

5- Increase productivity.

6- Customer-satisfaction.

4.1 System Architecture

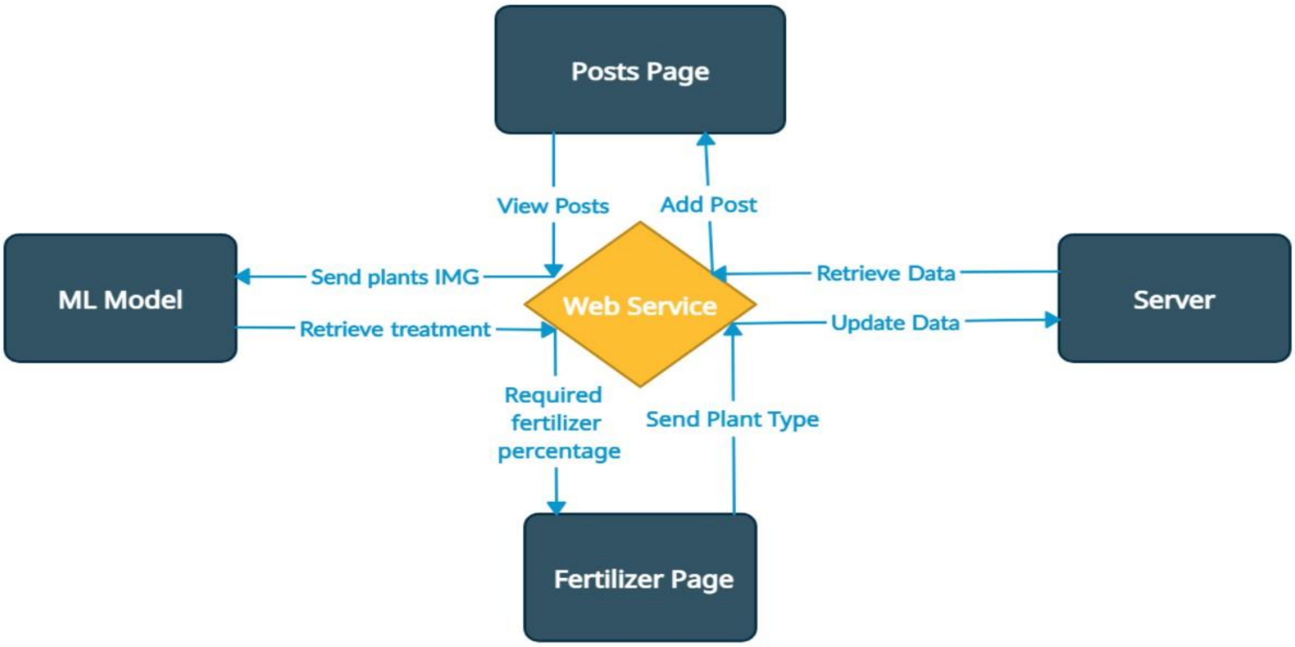


Figure 1. System architecture

- The main connection point is the Web Service (API) that connects all the system components.
- The user can make a post or can view the posts by sending a request to the API to make/view posts from/to the database server.
- User can upload the plant IMG to the API and the ML model will retrieve the treatment and the plant pest type.
- User can send the plant type and the cultivated area to the API, and it will retrieve the required fertilizer percentage.

4.2 System Use-Case

4.2.1 Application/User Use case

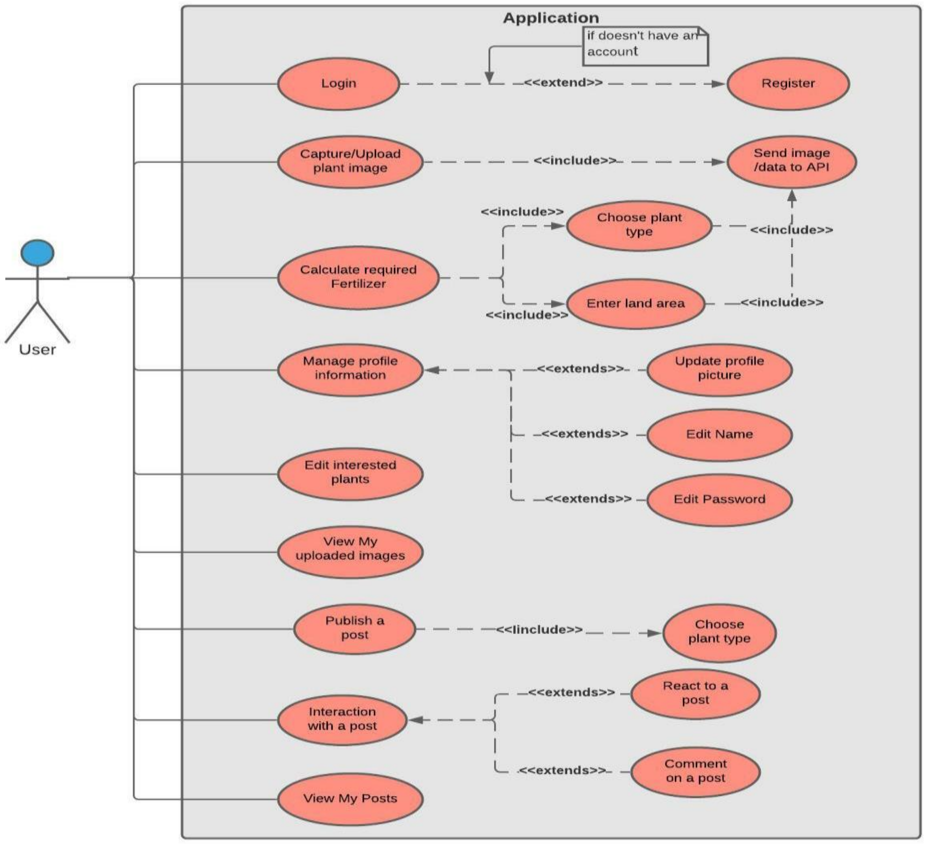


Figure 2. Application/User Use case

4.2.2 Server Use case

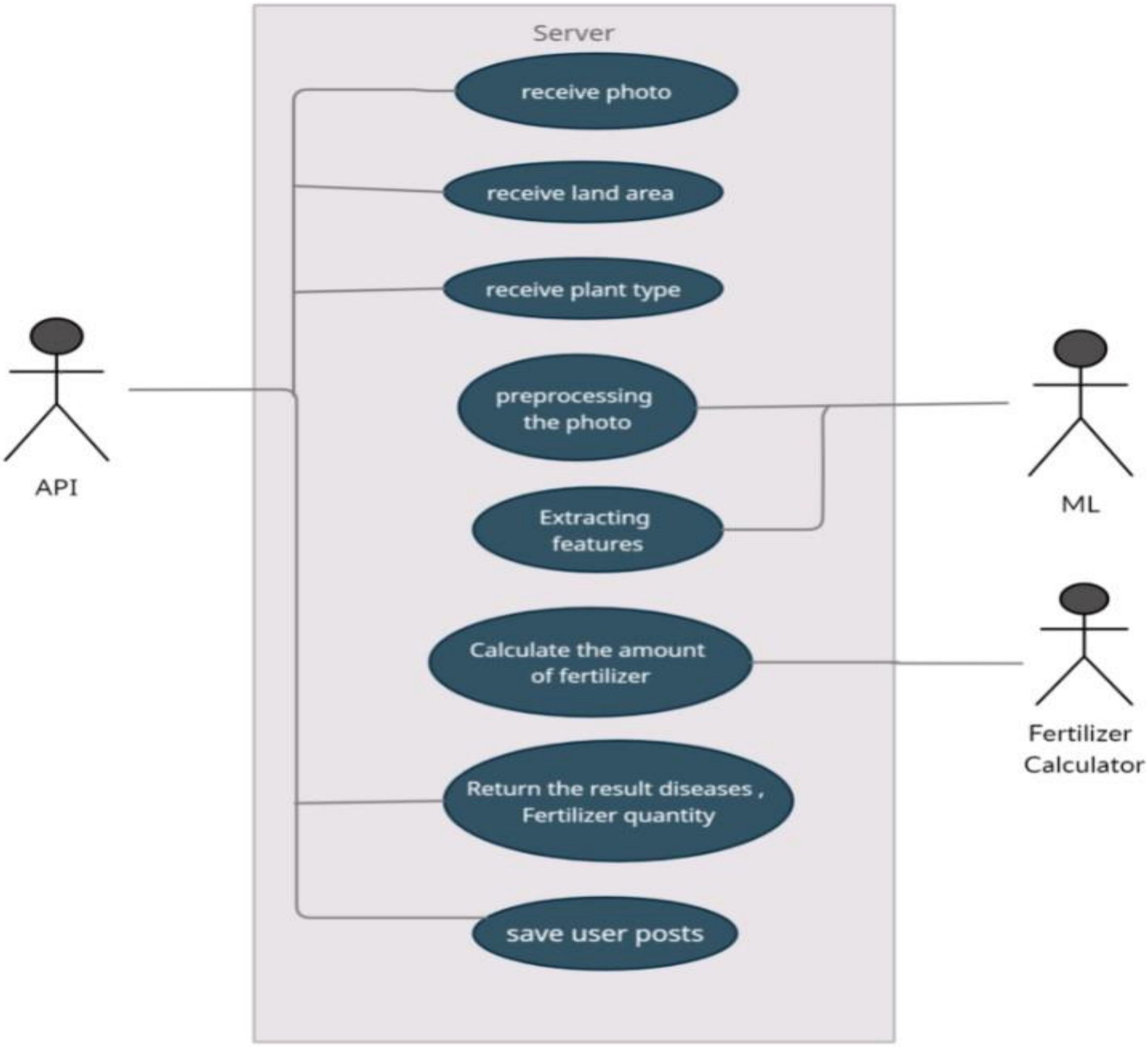
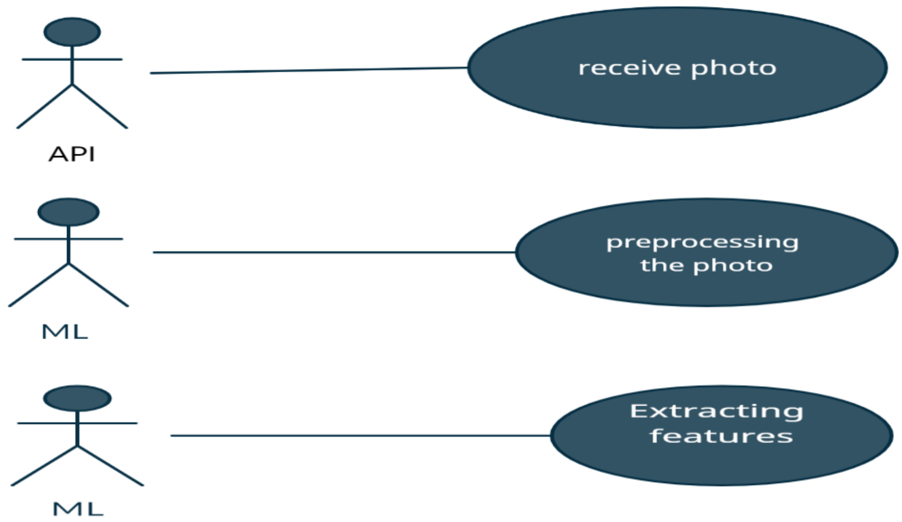
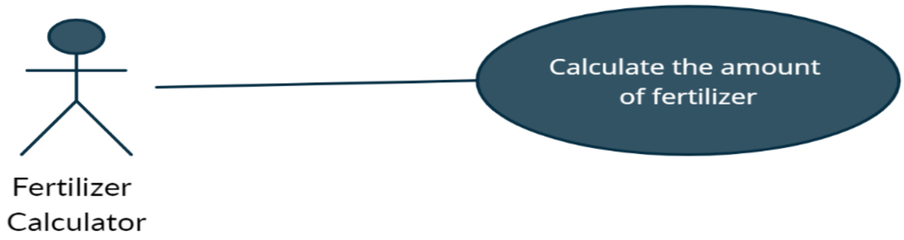


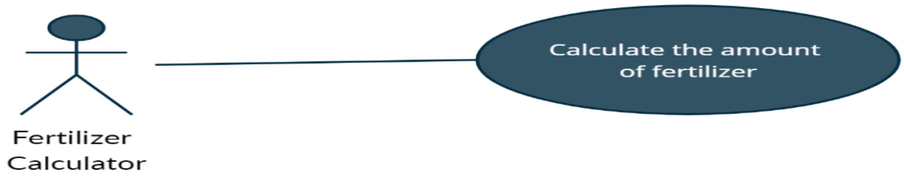
Figure 3. Server use case

4.3 Use Case Description

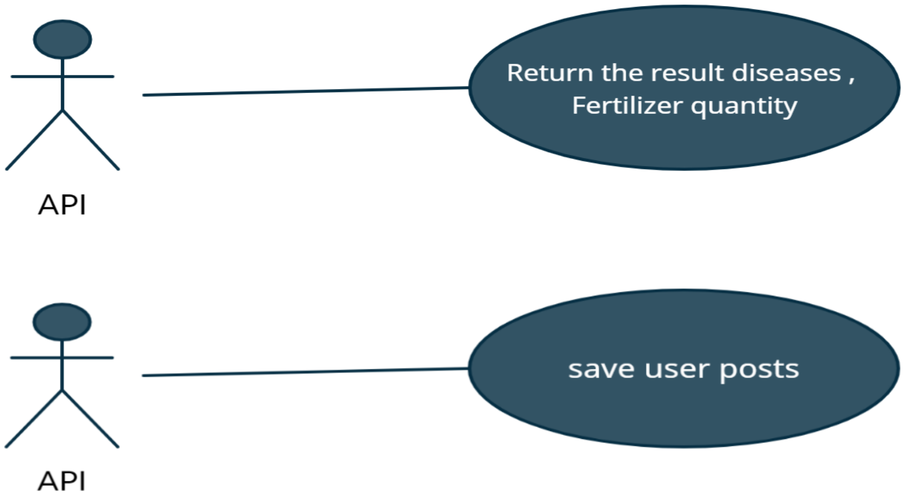


Use Case ID:	PlantsCare-ML01
Use Case Name:	Preprocessing the photo and Extract features
Area:	Server
Actor:	ML
Description:	Extract the features from received photo
Preconditions:	A valid photo
Postconditions:	ML module has successfully extracted features from received photo
Triggering Event:	API send a photo
Main Flow:	1. Receive the photo from API 2. Preprocess the photo 3. Extract the features from the photo 4. Return the result of the health status of the plant
Alternative Flow:	If user upload invalid photo, warning message should appear to the user to reupload the data. If the plant not included with our plants, warning message should appear to the user that app couldn't check the data.



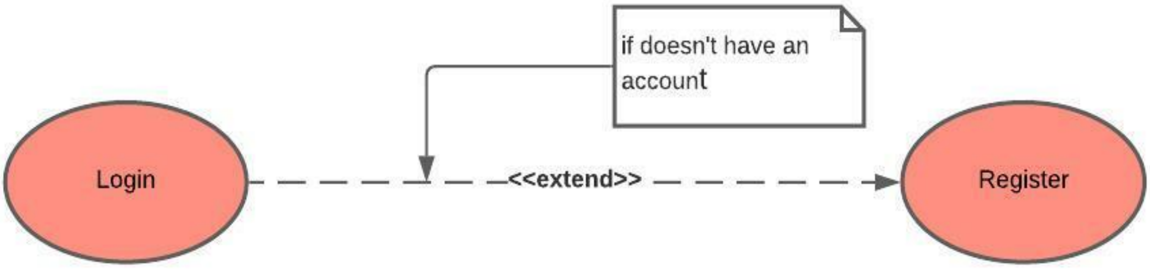


Use Case ID:	PlantsCare-Fertilizer_Calc01
Use Case Name:	Calculate the amount of fertilizer.
Area:	Server
Actor:	Fertilizer Calculator
Description:	Getting the land area and the plant then type return the amount of fertilizer
Preconditions:	There are no preconditions.
Postconditions:	Return the amount of the fertilizer.
Triggering Event:	API send the land area and plant type to Fertilizer Calculator.
Main Flow:	1. Received the land area. 2. Received the plant type. 3. Calculate the fertilizer amount. 4. Return the amount of the fertilizer amount.
Alternative Flow:	If the land area not valid number, a warning message should appear to the user.



Use Case ID:	PlantsCare-API01
Use Case Name:	Return the result and save posts.
Area:	Server
Actor:	API
Description:	API save user posts and return the result diseases, Fertilizer quantity.
Preconditions:	There are no preconditions.
Postconditions:	API return the result of plant status.
Triggering Event:	Received a photo and plant type or land area.
Main Flow:	Users take or upload a plant photo. API send a photo to ML module. Users enter land area and plant type. API send land area to Fertilizer Calculator
Alternative Flow:	If the land area not valid number, a warning message should appear to the user. If user upload invalid photo, warning message should appear to the user to reupload the data.

❖ Register



Use Case ID:	PlantsCare-User001
Use Case Name:	Register
Area:	Application
Actor:	User
Description:	User creates an account
Preconditions:	• User needs to download the application • User needs an internet access
Post conditions:	User has successfully created an account
Triggering Event:	User clicks "Register" button.
Main Flow:	1. Open the application 2. User enters his data 3. Users submit his data to system 4. The system validates the user data 5. The system sends a verification mail to the user
Additional info for steps:	Step 2: data is name, username, password, email.
Alternative Flow:	-if user enter non-valid data, an error message should appear to him.

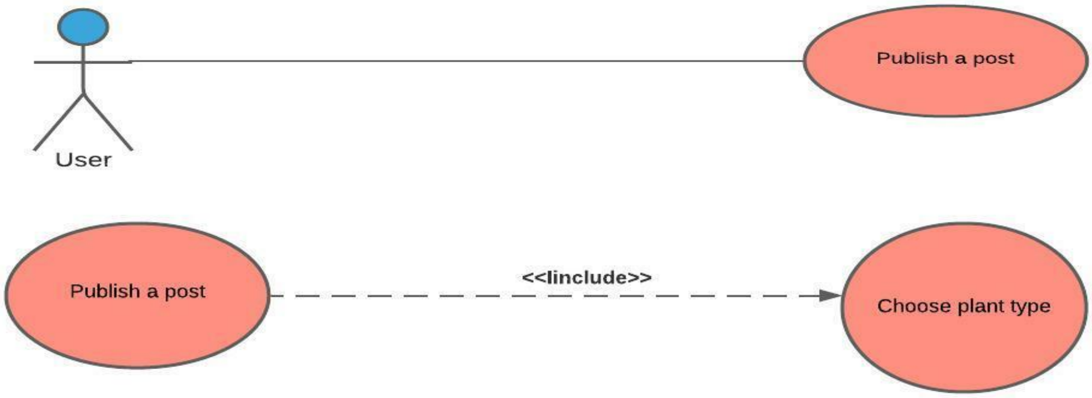
❖ Login

Use Case ID:	PlantsCare-User002
Use Case Name:	Login
Area:	Application
Actor:	User
Description:	User login to the application using it's account
Preconditions:	• User must download the application • User must an internet access • User must have an account
Post conditions:	• User has successfully logged into the application • The system displays the main page of application
Triggering Event:	User clicks "Login" button.
Main Flow:	1. Open the application 2. User enters his data 3. Users submit his data to system 4. The system validates the user data
Additional info for steps:	Step 2: data is username and password.
Alternative Flow:	If user enter non-valid data, an error message should appear to him.

❖ Edit interested plants

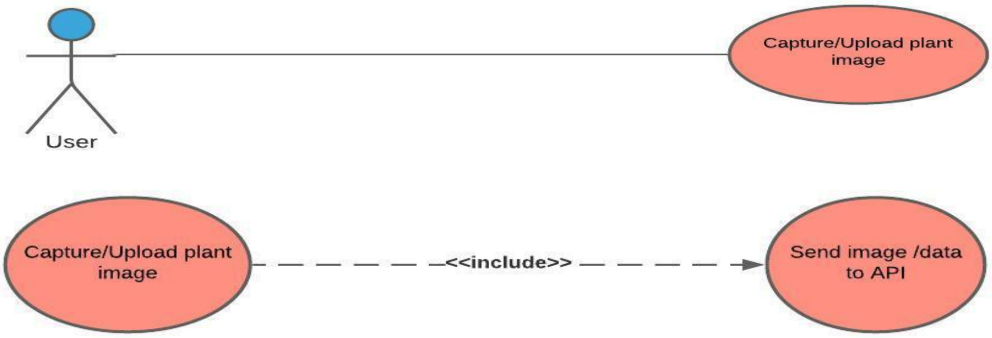
Use Case ID:	PlantsCare-User003
Use Case Name:	Edit interested plants
Area:	Application
Actor:	User
Description:	Users edit the pre-selected interested plants.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login to the application
Post conditions:	• Application updates the selected interested plants to the user.
Triggering Event:	Users click “Update” button.
Main Flow:	• User login to the application • Users select the items • Users submit the items • The application validates the items • The system updates the selected items
Additional info for steps:	Step 2, 3, 4, and 5: items are interested plants.
Alternative Flow:	If user doesn’t select any item, an error message will be appeared.

❖ Publish a post



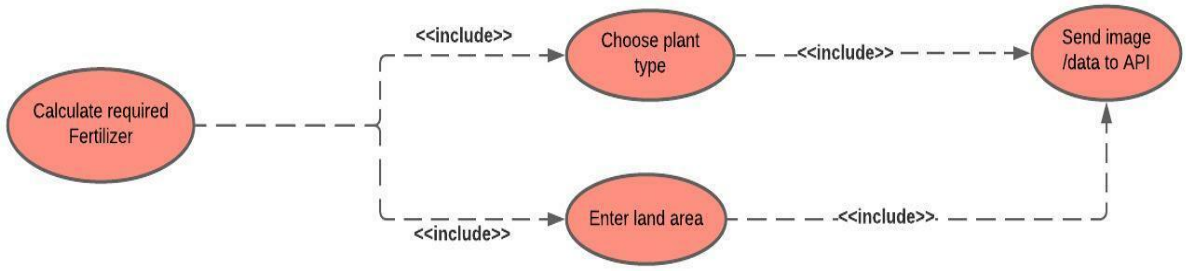
Use Case ID:	PlantsCare-User004
Use Case Name:	Publish a post
Area:	Application
Actor:	User
Description:	Users publish a post in the community area to get help in specific plant.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login to the application
Post conditions:	Application will make the post available to all users of the application
Triggering Event:	Users click “Post” button.
Main Flow	1. User login to the application 2. Users choose plant type 3. Users submit the post 4. The system validates the post
Alternative Flow:	If user doesn’t choose a plant type or doesn’t write any text in the post, an error message will be appeared.

❖ Capture/Upload plant image



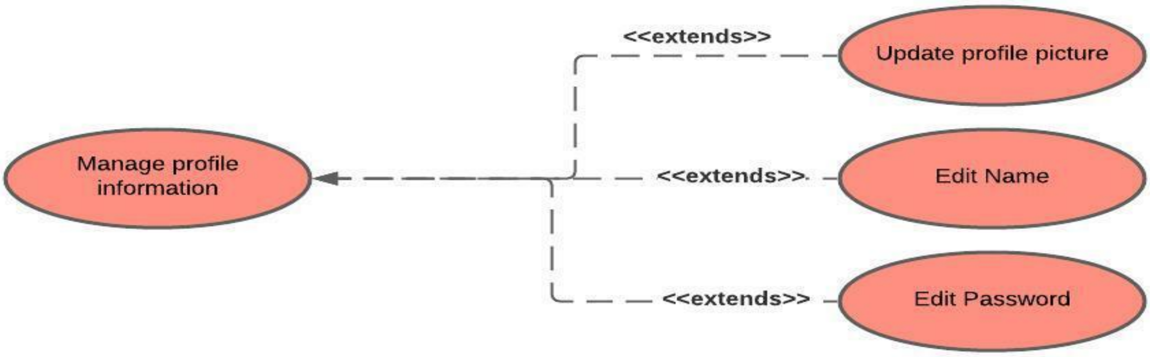
Use Case ID:	PlantsCare-User005
Use Case Name:	Capture/Upload plant image
Area:	Application
Actor:	User
Description:	User upload or take an image of plant to see if it has a disease or pest.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login to the application
Post conditions:	The User will have a paragraph shows the result.
Triggering Event:	Users click “Take Photo” button.
Main Flow	1. User login to the application 2. User upload or take a photo 5. Users submit the photo 6. The system validates the photo 7. The photo is sent to the API.
Alternative Flow:	If user doesn’t upload a plant image the application will display an error message.

❖ Calculate required Fertilizer



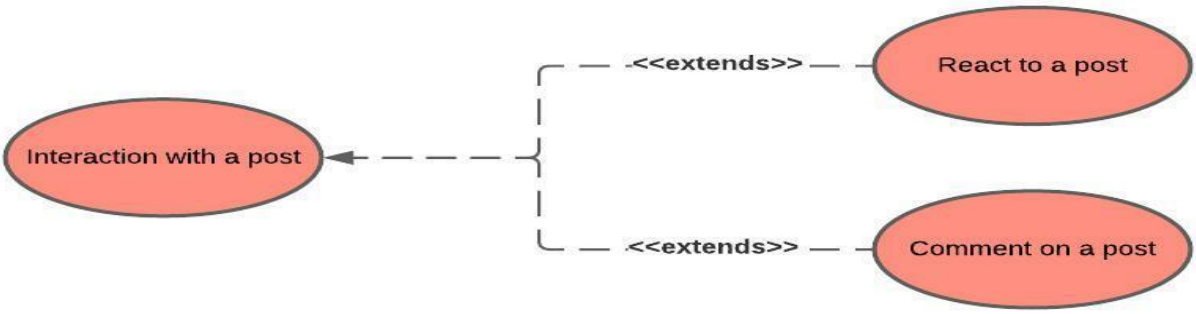
Description:	User wants to know the amount of fertilizer suitable to his land area.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login to the application
Post conditions:	User will have the required amount of fertilizer.
Triggering Event:	Users click “Calculate” button.
Main Flow	1. User login to the application 2. Users enter data 3. Users submit the data 4. The system validates the data 5. The data is sent to the API.
Additional info for steps:	Step 2, 3, 4, and 5: data is the plant type and the land area.
Alternative Flow:	If user enters invalid data, an error message will be appeared.

❖ Manage Profile information



Use Case ID:	PlantsCare-User007
Use Case Name:	User Information
Area:	Application
Actor:	User
Description:	Manage Profile Info
Preconditions:	• User must have an account on the application • User must be logged into the application successfully.
Post conditions:	User has successfully logged to his/her account
Triggering Event:	User clicks "Update profile data" button.
Main Flow:	1. Open the application 2. User enters his data 3. Users submit his data and photo to the system 4. The system validates the user data
Additional info for steps:	Step 2: data is name, profile image, password.
Alternative Flow:	If user enter non-valid data, an error message should appear to him. Ex: If the user uploads a file in the place designated for uploading a profile photo.

❖ Interaction With a post



Use Case ID:	PlantsCare-User008
Use Case Name:	Posts actions
Area:	Application
Actor:	User
Description:	Interaction with a post
Preconditions:	• User must have an account on the application • User must be logged into the application successfully.
Post conditions:	User has successfully logged to his/her account
Triggering Event:	User clicks “react” icon.
Main Flow:	5. Open the application 6. User enters his data 7. Users submit his data and photo to the system 8. The system validates the user data
Additional info for steps:	User can react on any post. Step 2:

4.4 Analysis Class

4.4.1 Context Diagram

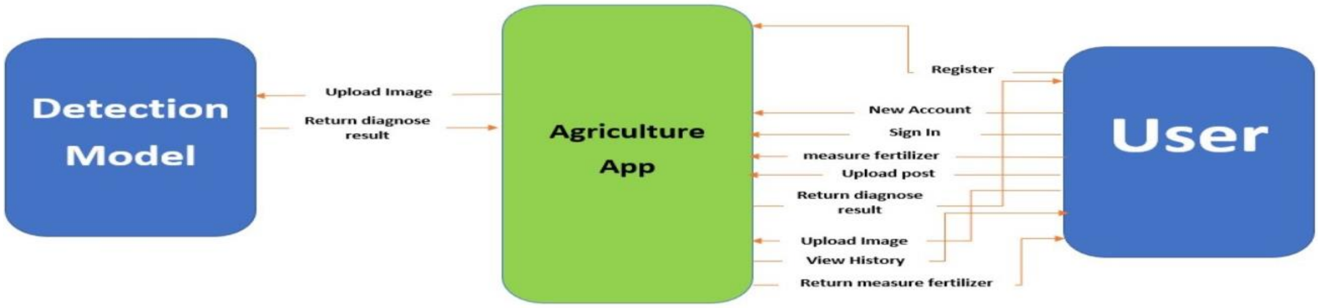


Figure 4. Context diagram

4.4.2 State Diagram

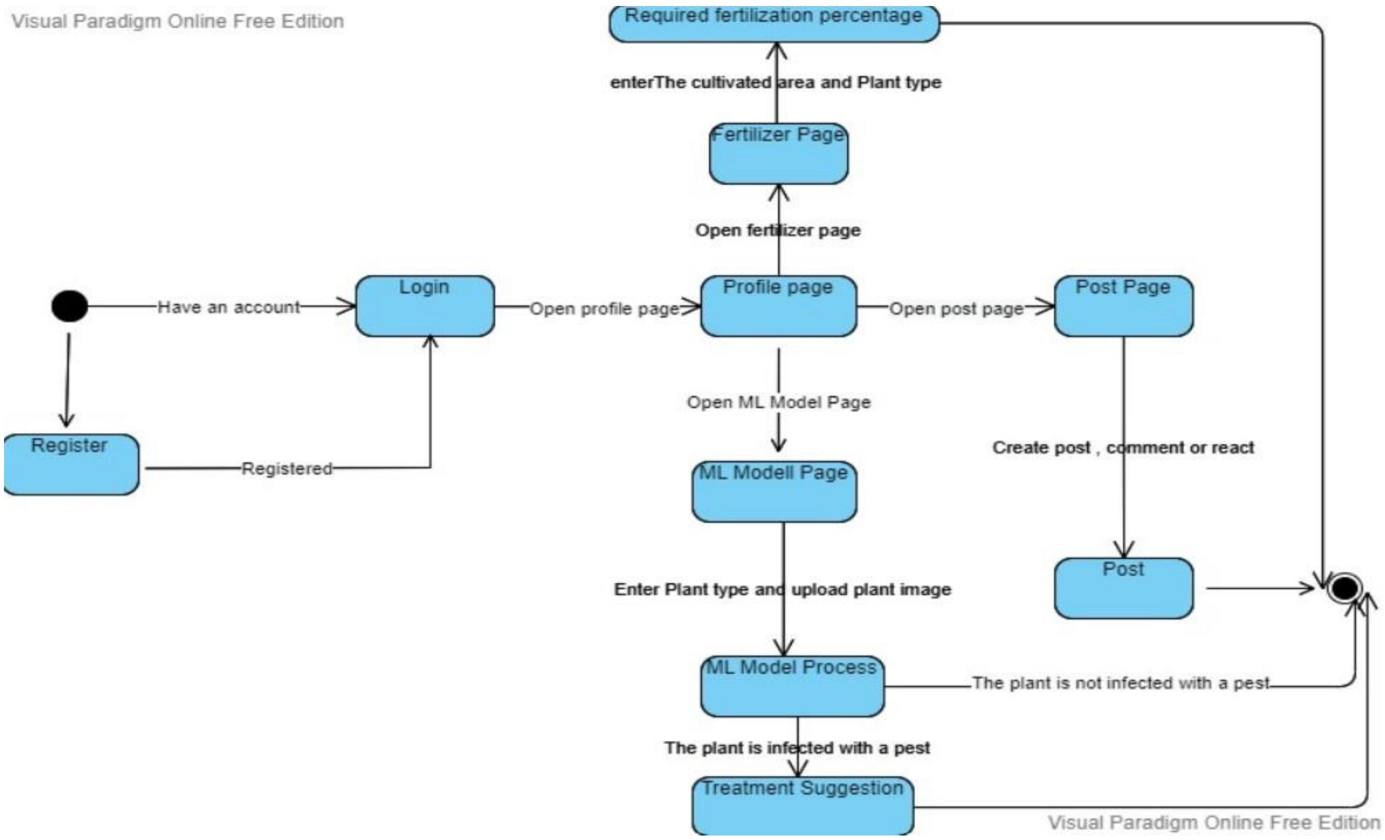


Figure 5. State diagram

4.4.3 Level 0 Diagram

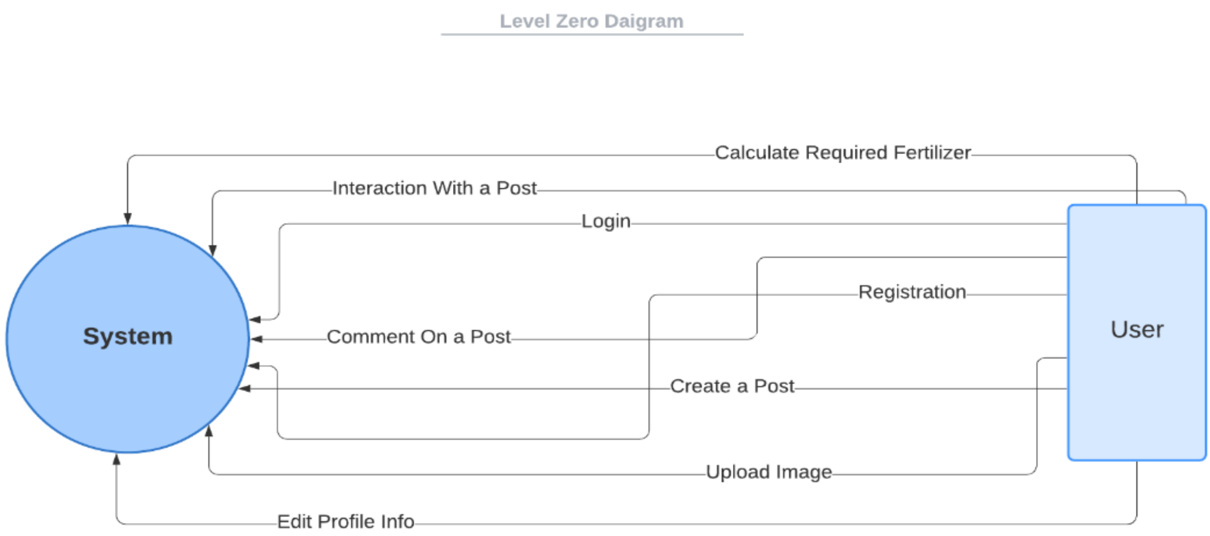


Figure 6. Level 0 diagram

4.5 Interaction Diagram (Sequence Diagram)

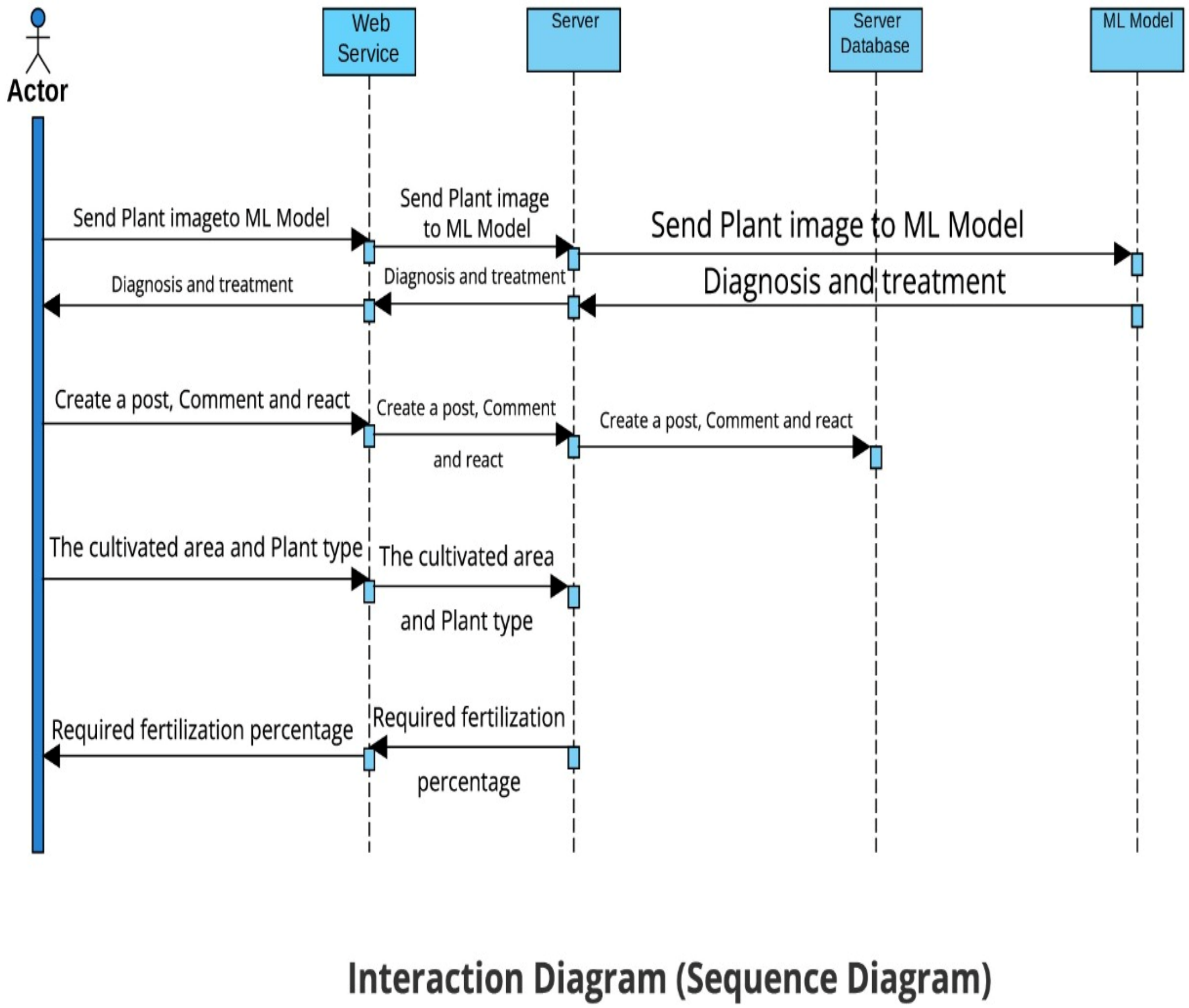


Figure 7. Sequence Diagram

4.6 Design Class

4.6.1 Class Diagram

Figure 8. Class diagram

4.6.2 Domain Diagram

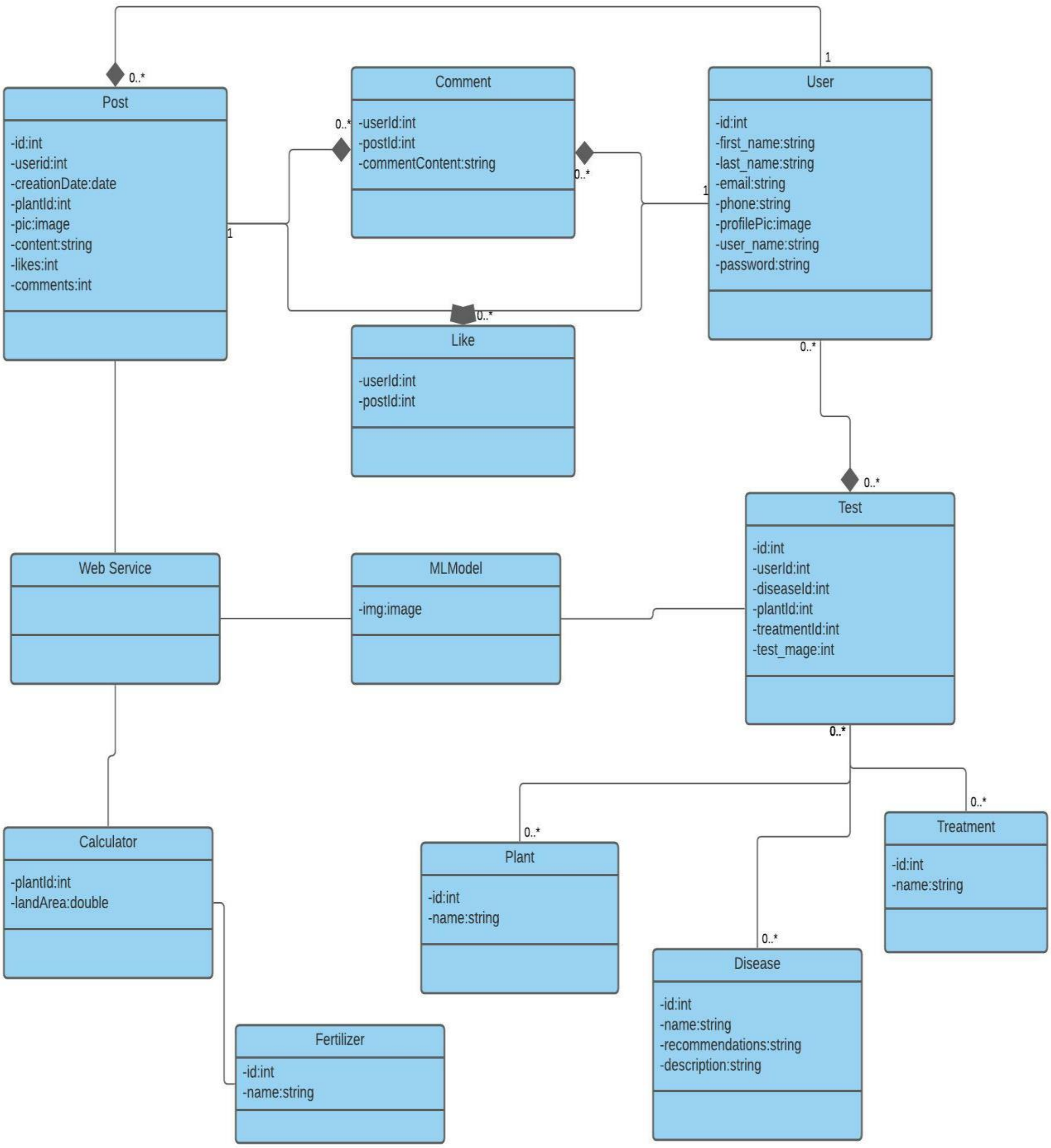


Figure 9. Domain diagram

4.7 ER Diagram

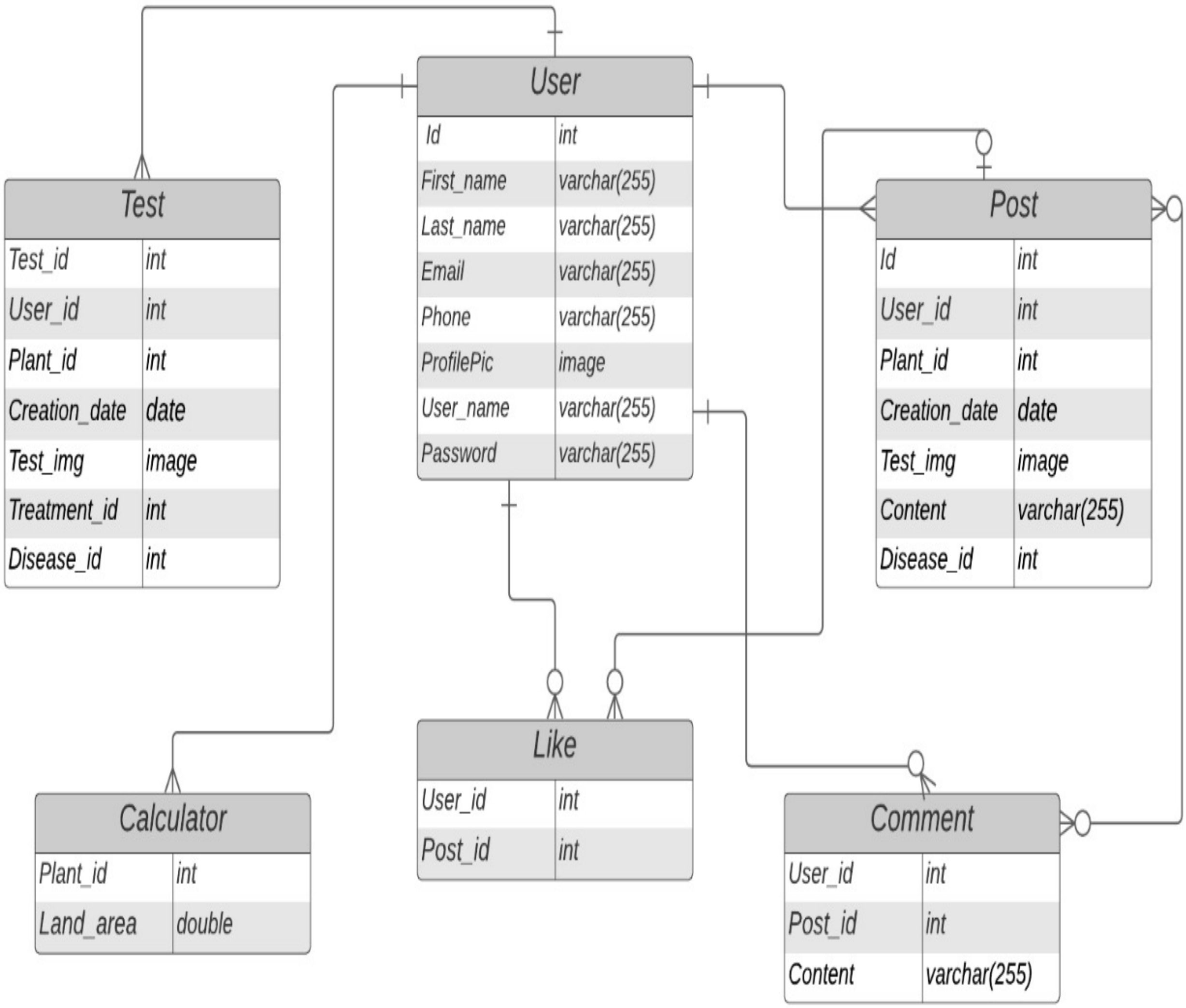


Figure 10. ER Diagram

4.8 Database Schema

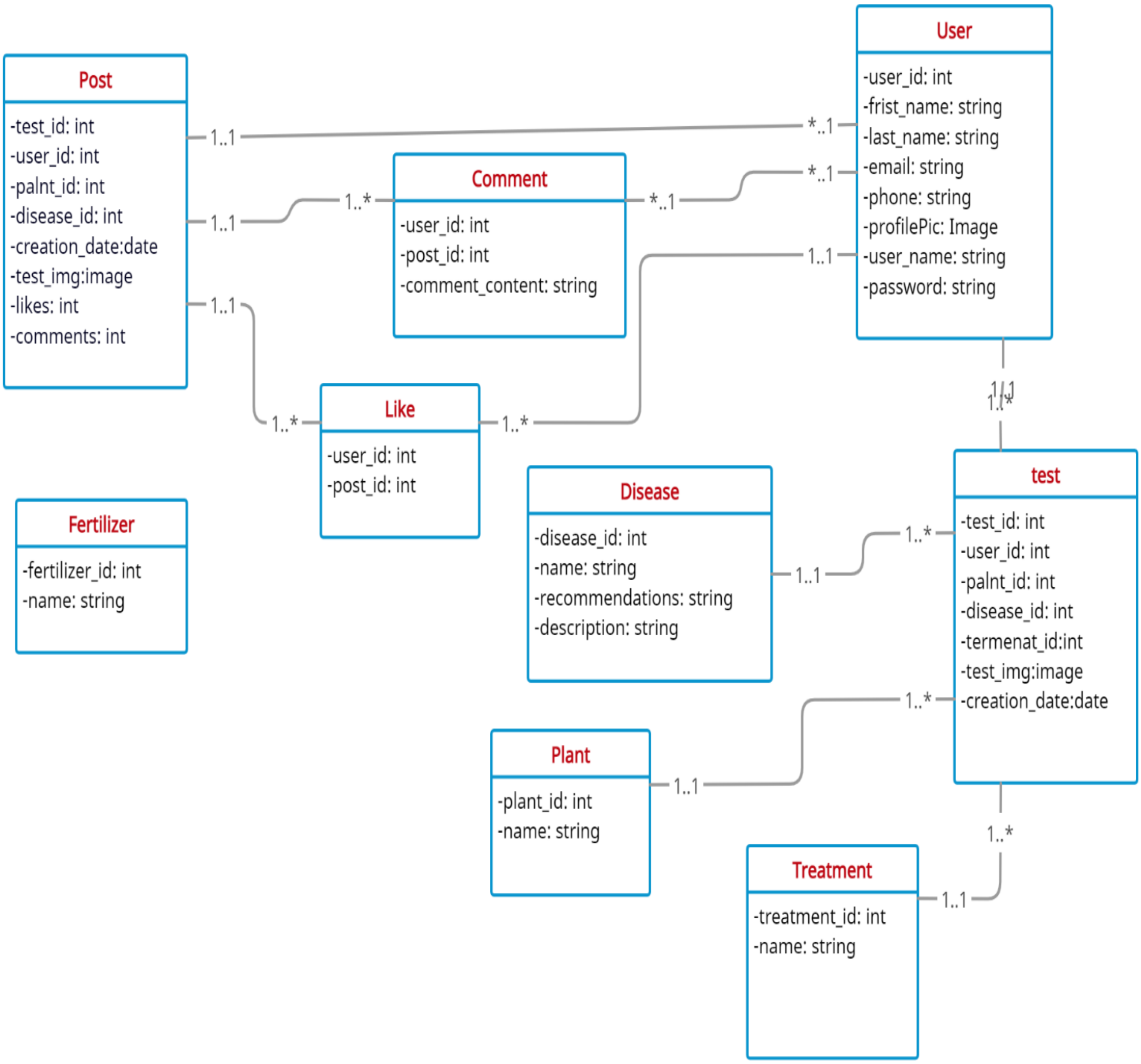


Figure 11. Database Schema

4.9 Design Mockup

Figure 12. Registration page

Figure 13. Login page

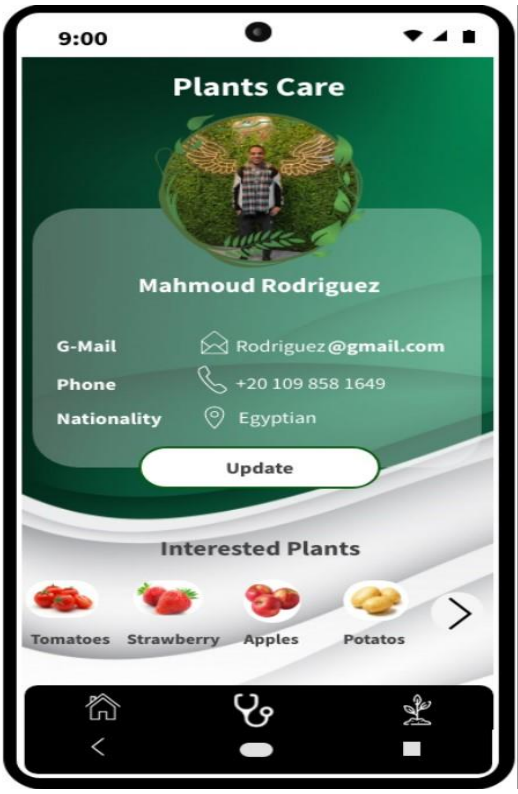


Figure 14. Profile page

Figure 15. Posts page

Figure 16. Model page

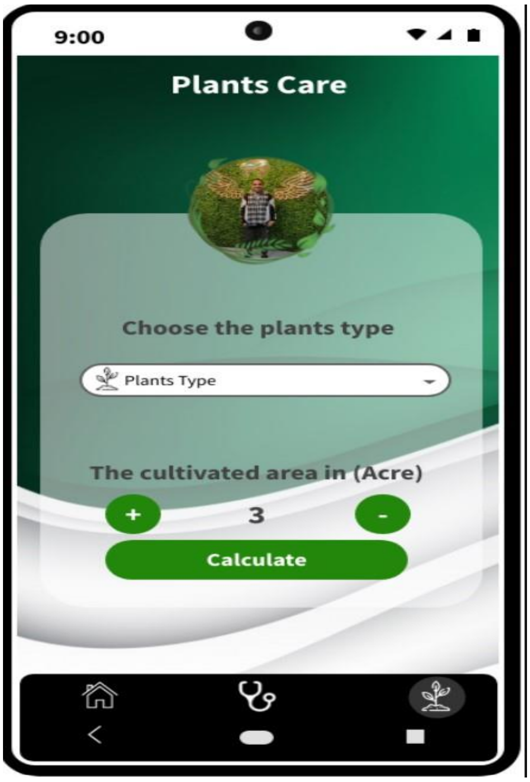


Figure 17. Fertilization Calculate page

5.1 References

- https://www.researchgate.net/publication/342601699_An_Improved_Deep_Learning_model_for_plant_disease_detection
- https://www.researchgate.net/publication/301879540_Using_Deep_Learning_for_Image-Based_Plant_Disease_Detection
- https://www.researchgate.net/publication/342234677_Plant_Disease_Detection_with_Deep_Learning_and_Feature_Extraction_Using_Plant_Village
- https://www.researchgate.net/publication/327065422_Plant_Disease_Detection_Using_Machine_Learning
- https://www.researchgate.net/publication/342710656_Plant_Disease_Detection_Using_Machine_Learning_Algorithms
- https://www.researchgate.net/publication/337858290_Plant_Disease_Detection_and_Classification_Using_Deep_Neural_Networks
- https://www.researchgate.net/publication/325580687_Plant_disease_detection_and_its_solution_using_image_classification
- https://www.researchgate.net/publication/352108134_Image-Based_Detection_of_Plant_Diseases_From_Classical_Machine_Learning_to_Deep_Learning_Journey
- https://www.researchgate.net/publication/322941653_Deep_learning_models_for_plant_disease_detection_and_diagnosis