NAPTA

Graduation Project, Part-I (CS)

Computer Science Department

FCI LUXOR UNIVERSITY

Supervised By:

Dr. Mohammed Atta

Submitted by:

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


CHAPTER
1

1.1 Abstract

By the year 2100, the world need of food is expected to reach its peak due to continuous exponential increase of human population all over the world, exceeding 9 billion human being with basic need "Food" as the main global attention. On a hand the first proposal to overcome now and future food needs, the production rates must reach a **percentage of 70** by the year 2050. On the other hand, **plant/s pandemic/s** reduce the expected crop yield production with an estimation of **40%,** what is leading to a complete loss of outcomes, especially in distinct and far places where plant creatures superior and oracle knowledge in form of guidance doesn't exist.

Google Cloud Platform made it possible to simply design and deploy applications with multiple **API** guarantee ease of use through all available operating systems such as web-based or device-based**. Artificial Intelligence (AI)** combined with **Internet of things (IoT)** caused the most outstanding advances in both communication and information processing/exchange producing human programmed expert systems. Proposed solution strategy follows the recommended as stated in **Industry 5** by the modern manufacturing world for the fifth industrial revolution. "**NAPTA**" provides a solution to play the role of agricultural engineers giving informative guidance, advice and required support for all farmers around the world using their preferred language, plants they farm, and also a chat-bot to allow farmers to report their problems questions. **"NAPTA"** is a future application guiding farmers towards smart farming, hence the application provides a wide information database with enough knowledge covering all stages for planting, starting from the seed to harvesting instructions including irrigation, fertilization, disease both chemical & organic fighting solutions.

Table of Contents

Content **Page**

List of Figures

Content **Page**

1.2 Key Words

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.
Artificial Intelligence (AI)	Artificial intelligence (AI) is a wide-ranging branch of computer science concerned with building smart machines capable of performing tasks that typically require human intelligence.
Internet of things (IoT)	The Internet of Things (IoT) describes the network of physical objects—"things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. These devices range from ordinary household objects to sophisticated industrial tools. With more than 7 billion connected IoT devices today, experts are expecting this number to grow to 10 billion by 2020 and 22 billion by 2025. Oracle has a network of device partners.
Industry 5.0	Industry 5.0 is the digital transformation of manufacturing/production and related industries and value creation processes.
Deep Learning (DL)	Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.
Computer Vision (CV)	Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.
K-Means	K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine

	learning or data science. In this topic, we will learn what is K-means clustering algorithm, how the algorithm works, along with the Python implementation of k-means clustering.
CNN	CNNs are powerful image processing, artificial intelligence (AI) that use deep learning to perform both generative and descriptive tasks, often using machine vison that includes image and video recognition, along with recommender systems

1.3 Introduction

Population density has been increased 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 which is growing by up to **160** people per minute, with a greater than **90%** of them in developing countries. 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 using **AI** principles.

The use of technology in the detection and analysis process increases the accuracy and reliability of these processes. For example, the people who use the latest technology to analyze the diseases that arise unexpectedly are at a higher chance of controlling them than those that do not. In the recent occurrence of coronavirus, the world relied on the latest technology to develop preventive measures that have helped reduce the rate at which the disease is transmitted. Crop diseases are a significant threat to human existence because they are likely to lead to droughts and famines. They also cause substantial losses in cases where farming is done for commercial purposes. The use of **AI Platform** could improve the detection and fighting of diseases. Computer vision is a form of **Artificial Intelligence (AI)** that involves using computers to understand and identify objects. It is primarily applied in testing drivers, parking, and driving of self-driven vehicles and now in medical processes to detect and analyze objects. Computer vision helps increase the

accuracy of disease protection in plants making it easy to have food security.

One of the areas that CV has helped most is the detection of the severity of the diseases AI Platform is useful and promising in determining the severity of diseases in plants and animals. It is also used to classify diseases and avoid the late detection of diseases. Plant diseases are slightly different from those that affect human beings. Many factors make diseases similar as well. However, the diseases that can be transmitted from humans to plants and vice versa are rare. The analysis of the data related to this improved. The images of leaves and other parts of the plant scan be used to detect diseases in plants. The technology could be applied in analyzing images in human beings that also prove the presence of diseases and determine the extent of their destruction.

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.

1.4 Background

The rise of global population brings several challenges around global sustainability, including the need for more food. Agriculture is the dominant sector of the economy which contributes up to **6.4%** of the entire world's economic production.

Agriculture does not only provide the energy 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 (AI) 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 nowadays, so our application will make use of modern solutions that use internet and technologies to beat these problems, increasing the productivity of the crops, help farmers and providing them with the necessary experience to identify diseases and giving them suggestions to ensure the best productivity.

AI Platform image processing allows farmers to depend on digital tools to recognize plant types and to determine which crops are healthy and which ones are infested with disease caused by fungi, bacteria, or viruses.

We focus on using Ai Platform 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 all users.

2.1 Related Work

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 infected 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 infected 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 infected 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 infected and healthy leaves are collectively trained under Random Forest to classify the infected 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 Venkataraman an. -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 infected 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).


CHAPTER


3

3.1 Domain Analysis

The main issue is trying to help farmers know the diseases type that affect their plants and knowing the appropriate treatment for each disease as soon as possible. Some new farmers suffer from lack of experience in some agricultural matters, so they need someone to exchange experience to have the best crops. It can be obtained under these conditions and using the appropriate tools and equipment. The Application presents an experienced community where people can help each other in the faced problems. A chat-bot system will be available in order to provide plants information to users, so the user can ask the bot for any information about the plant among the common and predefined questions, then the bot will return the answer to the user.

Identification of the disease: The user will be able to capture/upload a plant leaf image with his smart phone to know if it is infected or not. The application analyzes the image and return to him the results of its analysis and suggests a treatment if it was infected.

Applying AI-Platform algorithms and classifiers, they showed promising results in image classification and decision-making to assist farmers to diagnose the plant diseases, which will serve as a boost in improving plant care services through effective image analysis of symptoms (pests) suffered by the plant.

Helping/Asking others for help: the user who wants an advice on his plant or anything related to the plant he is interested in can ask a general question (post) to all users and other users will respond to him.

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

Chat-Bot: A chat-bot system will be available in order to provide plants information to users, so the user can ask the bot for any information about the plant among the common and predefined questions, then the bot will return the answer to the user.

3.2 Risks and Constrains

- **Risks**

Risk	Strategy	Priority
User may not know how to use the application	Tips will appear to the user which explain how to use our application.	High
Malfunction with the servers that host the database, the posts, and users' information	Hosting the database and the several servers.	High
Malfunction with the server or the API hosting the model	Hosting the model and the API on several servers.	High

Constrains

- ✓ The application must provide friendly and easy interface with few icons and tips to help people who don't know use the app directly.
- ✓ User must scan objects clearly as possible
- ✓ User must have internet to deal with all application features.
- ✓ User must 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 AI Platform									
Testing and Final Discussion									

3.4 Quality Assurance Plan

Black box: In this stage we use test dataset as input to our AI Platform 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, AI 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 AI 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 Feasibility Study

Program Description:

Our program is an agriculture app that offer advice" treatment" to farmers and who have an interest to grow any plant without any cost and losing of time.

We will build our application to manage user upload a picture of his plant and we will show him a result that explain this plant is affected by any disease or not and if he is we will show him the type of disease and try to offer the treatment.

Our user can ask questions to our agriculture community without any cost and can find some of interesting services as calculation the percentage of fertilizers with his land or plants need.

We have a simple chatbot to help new users to use our application correctly and easily, and user can choose among many predefine questions and our chatbot will help him.

What's the benefits of this application?

- Our app aims to increase the knowledge of small farmers by protect their plants from diseases and make early detection of diseases easily.
- Our app aims to give advises without any cost and doesn't waste time of users by building a community among framers with each other and agriculture engineers
- Our app aims to ensure users use correctly amount and type of fertilizers to reduce money and losing in plants.

Who are the users of our application?

- Small farmers.
- Anyone cares to home farming or plants.

How will we promote to our app?

- We will upload our ios app store and google play and we will make it free.

What is the field which this app target?

- We target anyone have an interest to agriculture field.

3.6 System Requirements

3.6.1 Functional requirements

1- User interface enables users to create accounts, add their personal information and the plants in which they are interested.

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

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

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

5- The application shows to the user his posts or all posts of plants in which he is interested ordered by the latest date.

6- The application allows the users to create posts.

7- The application allows the users to interact with posts (like, comment).

8- User can search for posts with plant type.

9- AI-Platform allow user to capture / upload the plant leaf image for type detection and classification. The model is also used for plant examination from pests and diseases, if the plant is infected, AI-Platform returns details about this disease and recommends treatments.

10- A Server stores users' data and previous examined image results.

11- The user selects the plant type and enter agricultural area, and the application outputs the collection of best fertilization plan.

12- Chat bot to quickly get information about plants and answer of the common and predefined questions.

3.6.2 Nonfunctional requirements

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

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 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.

3.7 Techniques

AI-Platform: Applying AI Platform 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 fertilizing, and that's why fertilizing a plant is different from the other. The app will provide a user with the appropriate fertilization plans according to the agriculture area and plant type.
- **Posts Page:**
    - The user will be able to view other people's posts, and interact with them (like, comment), according to the latest date of the user interested plants.
    - He can also create a question (post) 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 type that will be available in any question or consultation.

❖ **Chat-bot Page:**

- A user can talk to the bot, by selecting the plant type, the bot responds with the predefined question about that plant, then user can select a question, bot responds the answer to the selected question.

3.8 System Request

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

Functionality:

- The ability to exanimate plants and recommend a treatment for free.
- The speed of obtaining the plants' diseases.
- The ability to generate full reports about plant diseases and treatments.
- The ability to exchange benefits with other users.
- The ability to get the best fertilization plans to get a good yield.

Expected Value:

- Less-cost plants examinations process.
- Lost cost plant cultivation plan.
- More organized posts about the benefits of plants.
- Ease-to-use application and friendly User-interfaces.
- Increase productivity.
- Customer-satisfaction.


CHAPTER
4

4.1 System Architecture

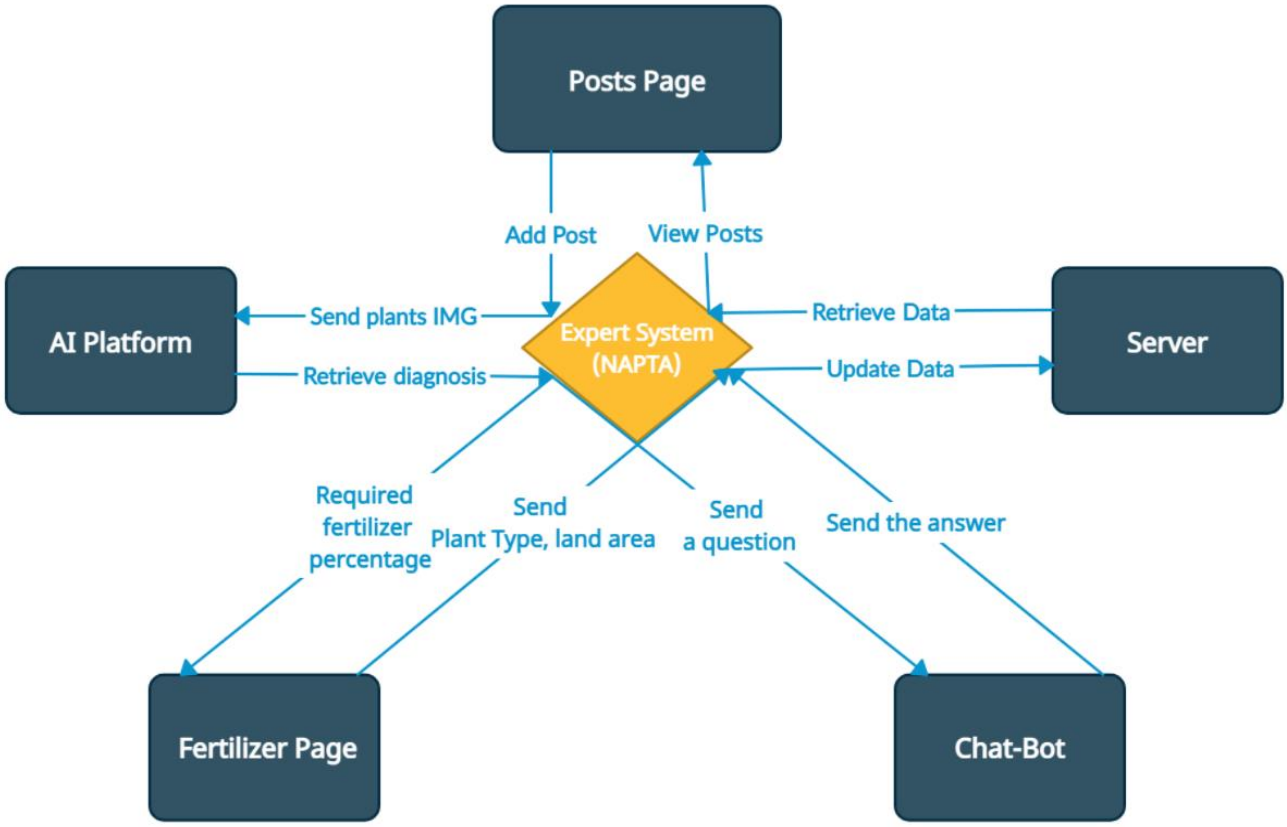


Figure 1. System architecture

- The main connection point is the Expert System (NAPTA) that connects all the system components.
- The user can create a post or can view the posts.
- Application sending a request to the Expert System (NAPTA) to save/return posts from/to the database server.
- User can upload the plant image to the Expert System (NAPTA) and the AI-Platform will retrieve the diagnosis.
- User can send the plant type and the cultivated area to the Expert System (NAPTA), and it will retrieve the required amount of fertilizer percentage.
- User selects one of the predefined questions about the selected plant and the bot returns the answer.

4.2 System Use-Case

4.2.1 Application/User Use case

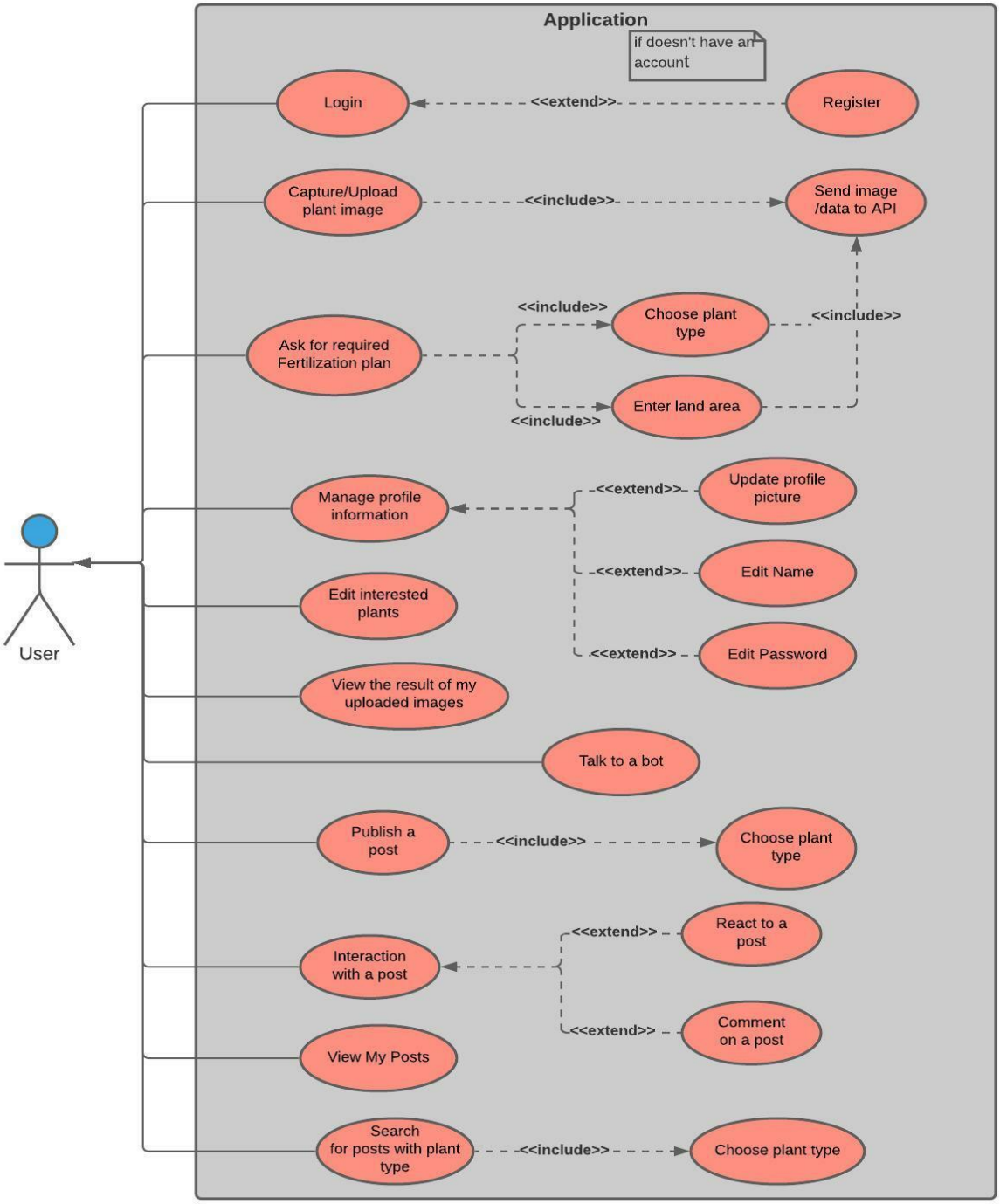


Figure 2. Application/User Use case

4.2.2 Server Use case

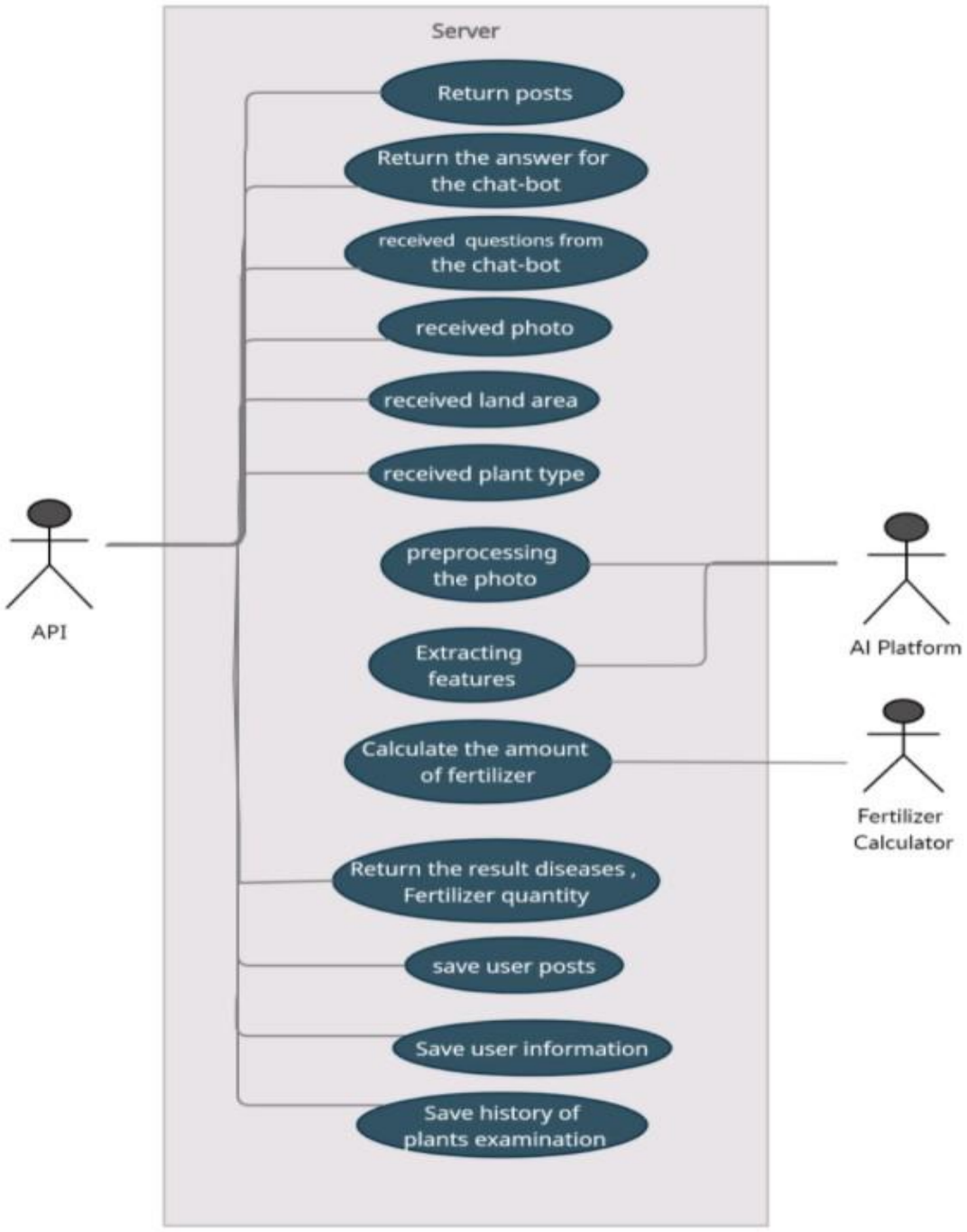
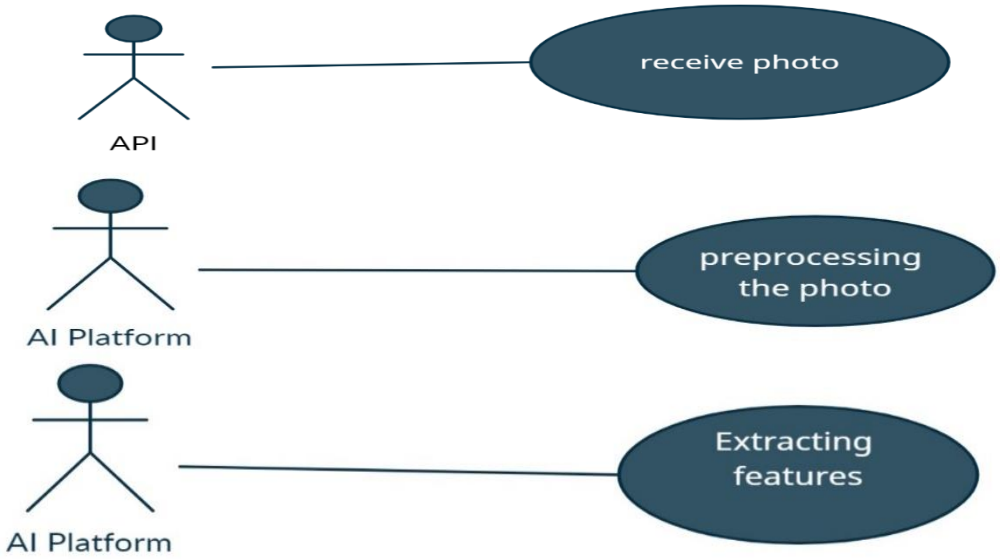
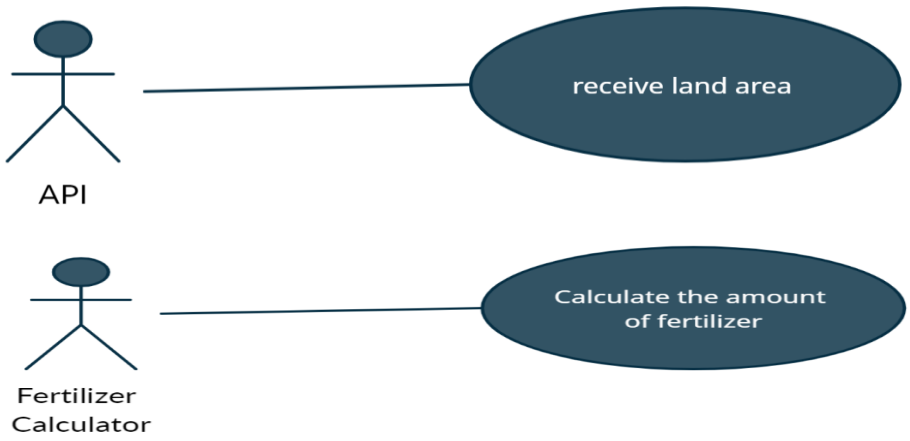


Figure 3. Server use case

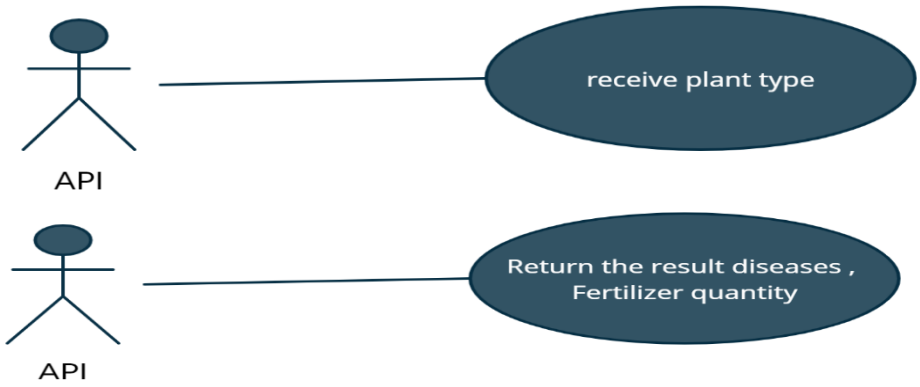
4.3 Use Case Description



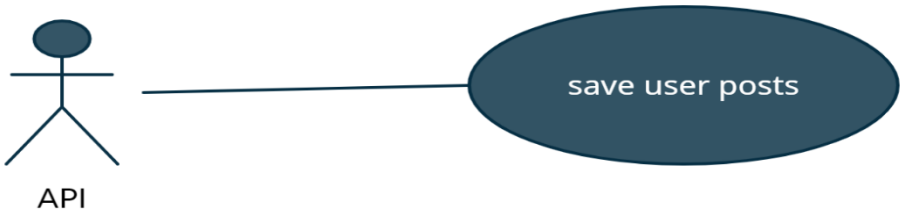
Use Case ID:	NAPTA-AI-Platform 01
Use Case Name:	Preprocessing the photo and Extract features
Area:	Server
Actor:	AI Platform
Description:	Extract the features from received photo
Preconditions:	A valid photo
Post conditions:	AI-Platform module has successfully extracted features from received photo
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
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.



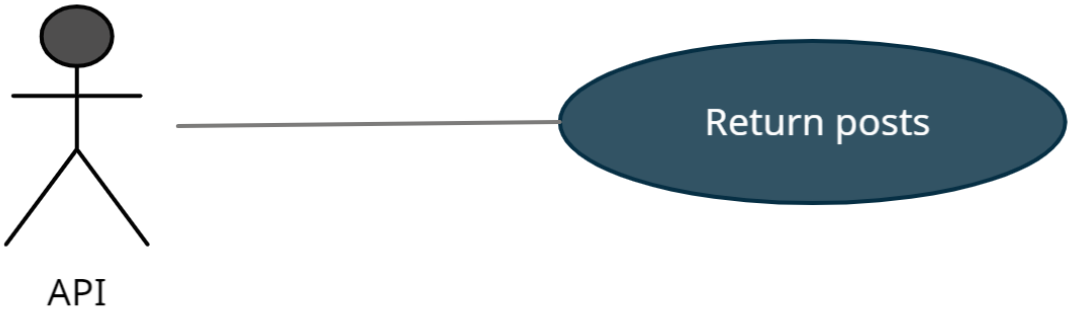
Use Case ID:	NAPTA-Fertilizer_Calc01
Use Case Name:	Calculate the amount of fertilizer.
Area:	Server
Actor:	Fertilizer Calculator
Description:	Getting the land area and the plant then type return the amount of fertilizer
Preconditions:	- land area must be existed. - plant type must be existed.
Postconditions:	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.
Alternative Flow:	-if the land area not valid number, a warning message should appear to the user.



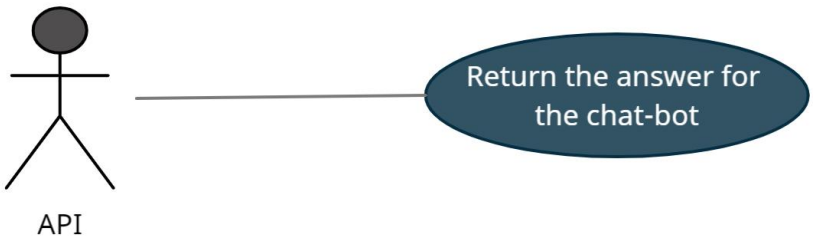
Use Case ID:	NAPTA-API01
Use Case Name:	Return the result
Area:	Server
Actor:	API
Description:	API returns the result diseases, Fertilizer quantity.
Preconditions:	- Photo must be existed. - Land area must be existed. - plant type must be chosen.
Postconditions:	API return the result of plant status.
Triggering Event:	Received a photo and plant type or land area.
Main Flow:	1. Receive diagnose of image or fertilizer quantity. 2. Send diagnose of image or fertilizer quantity to the user



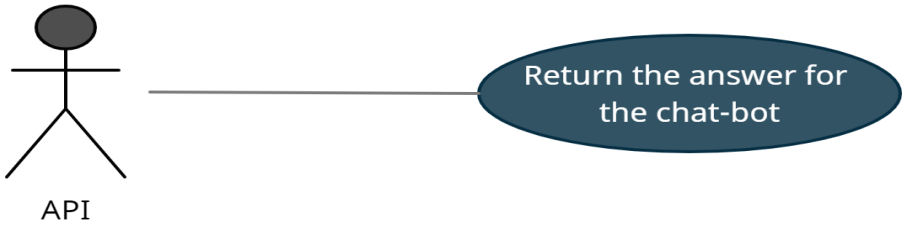
Use Case ID:	NAPTA-API02
Use Case Name:	Save user posts.
Area:	Server
Actor:	API
Description:	API save user posts.
Preconditions:	User created a post.
Postconditions:	Database stored the post.
Main Flow:	1. User creates a post 2. Application sends the post to API 3. API sends the post to the database 4. Database stores the post



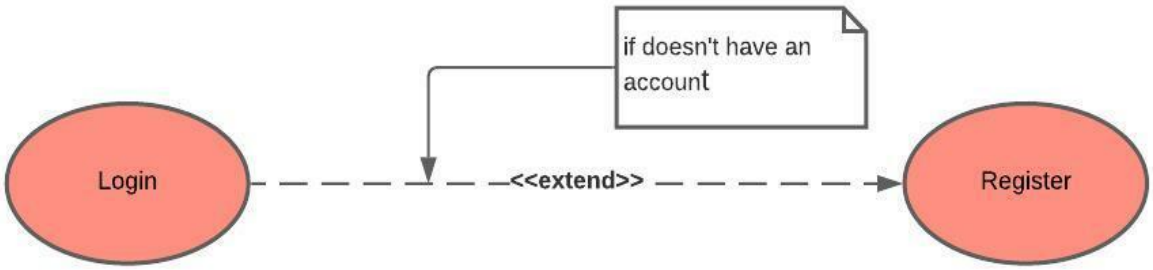
Use Case ID:	NAPTA-API-03
Use Case Name:	Return posts
Area:	Server
Actor:	API
Description:	Return posts to the application.
Preconditions:	Posts must be stored in the database.
Post conditions:	Posts are displayed in the application.
Main Flow:	1. API receives posts from the database 2. API returns posts to the application.



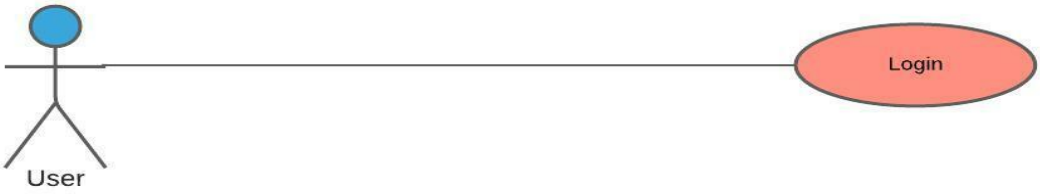
Use Case ID:	NAPTA -API-chatbot01
Use Case Name:	Received questions from user.
Area:	Server
Actor:	API
Description:	Received questions from and send it to the chat-bot.
Preconditions:	The question must be predefined.
Post conditions:	API return the answer to the selected question.
Triggering Event:	Clicks ‘send’ button
Main Flow:	1- user select pant type. 2- user select a question from the predefined questions. 3- API receive the question.
Alternative Flow:	If the user selects non-predefined plant type or non-predefined question, an error message should appear to him.



Use Case ID:	NAPTA-API-chatbot02
Use Case Name:	Return the answer to the question.
Area:	Server
Actor:	API
Description:	Return the answer to the selected question from the chat-bot.
Preconditions:	The question must be predefined.
Post conditions:	API return the answer to the selected question.
Main Flow:	1. API return the answer to the user.
Alternative Flow:	If the user selects non-predefined plant type or non-predefined question, an error message should appear to him.

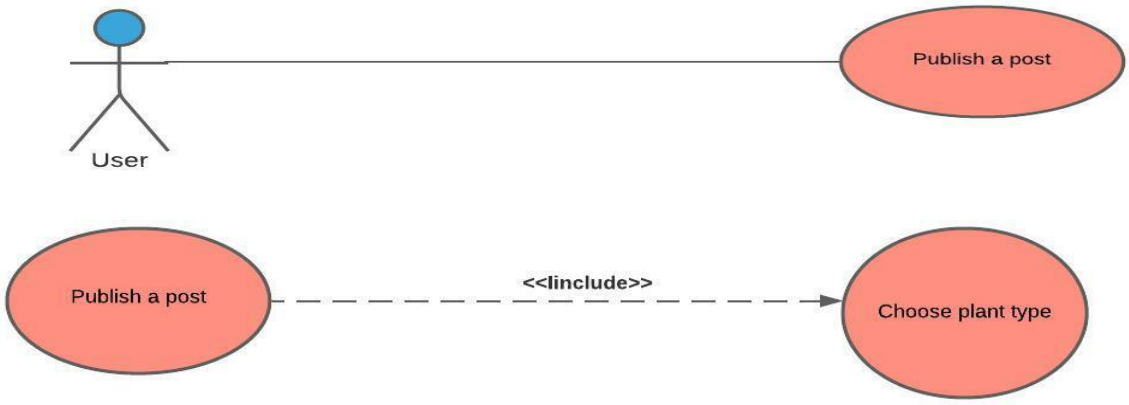


Use Case ID:	NAPTA-User001
Use Case Name:	Register
Area:	Application
Actor:	User
Description:	User creates an account
Preconditions:	• User needs to download the application • User needs an internet access
Post conditions:	User has successfully created an account
Triggering Event:	User clicks “Register” 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.

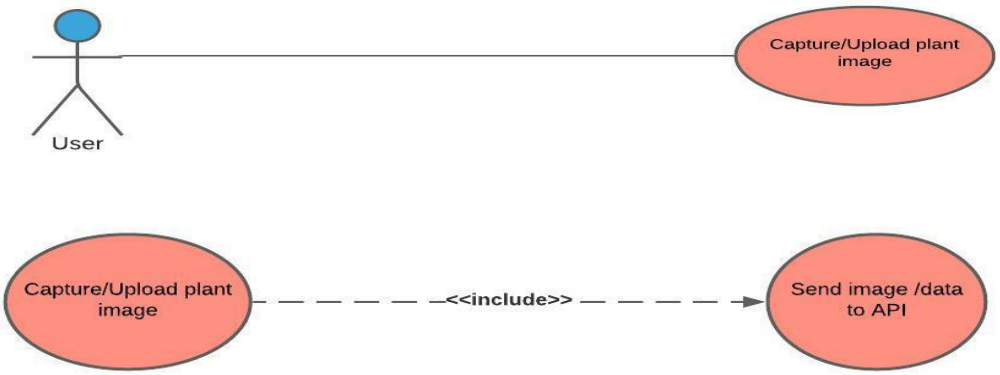


Use Case ID:	NAPTA-User002
Use Case Name:	Login
Area:	Application
Actor:	User
Description:	User login to the application using its account
Preconditions:	• User must download the application • User must an internet access • User must have an account
Post conditions:	• User has successfully logged into 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.

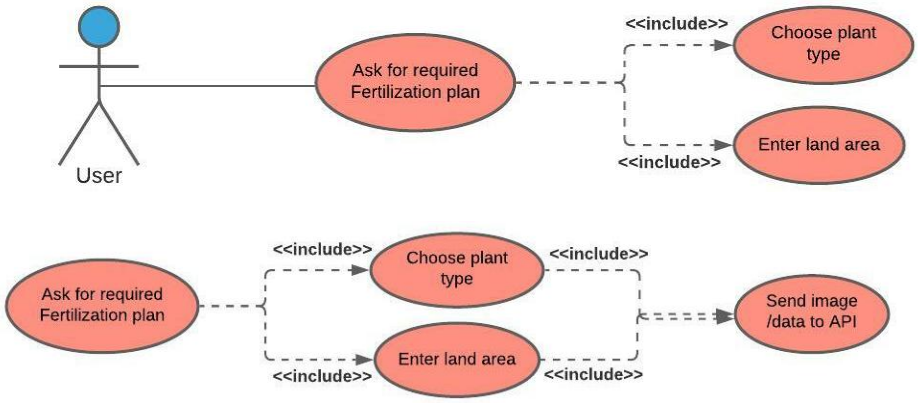
Use Case ID:	NAPTA-User003
Use Case Name:	Edit interested plants
Area:	Application
Actor:	User
Description:	Users edit the pre-selected interested plants.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login to the application
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.



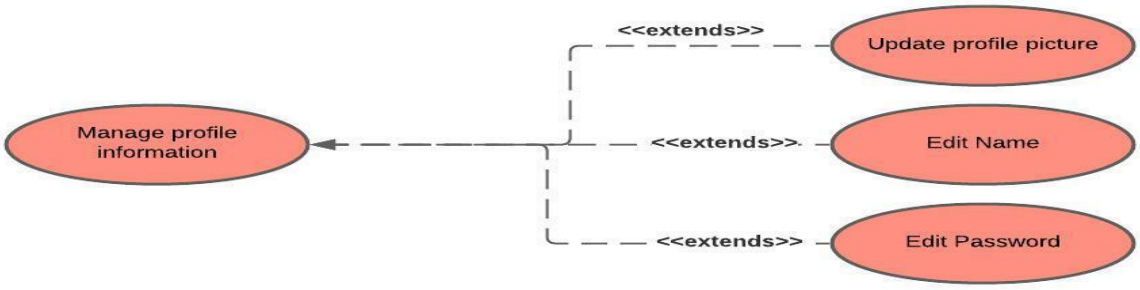
Use Case ID:	NAPTA-User004
Use Case Name:	Publish a post
Area:	Application
Actor:	User
Description:	Users publish a post in the community area to get help in specific plant.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User must login 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.



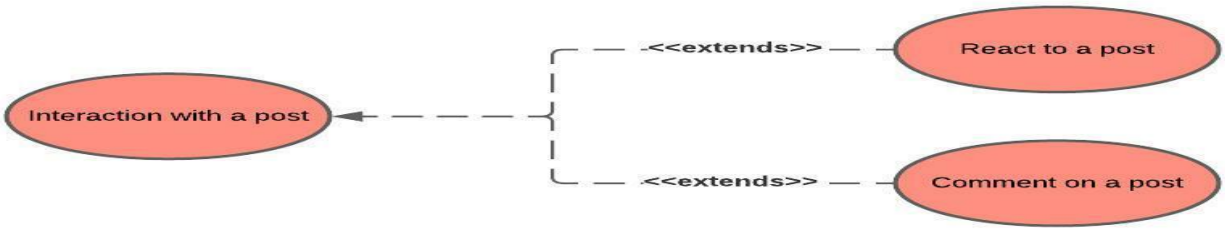
Use Case ID:	NAPTA-User005
Use Case Name:	Capture/Upload plant image
Area:	Application
Actor:	User
Description:	User upload or take an image of plant to see if it has a disease or pest.
Preconditions:	• User must download the application • User must an internet access • User must have an account • User 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.



Use Case ID:	NAPTA-User006
Use Case Name:	Calculate required Fertilizer
Area:	Application
Actor:	User
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.

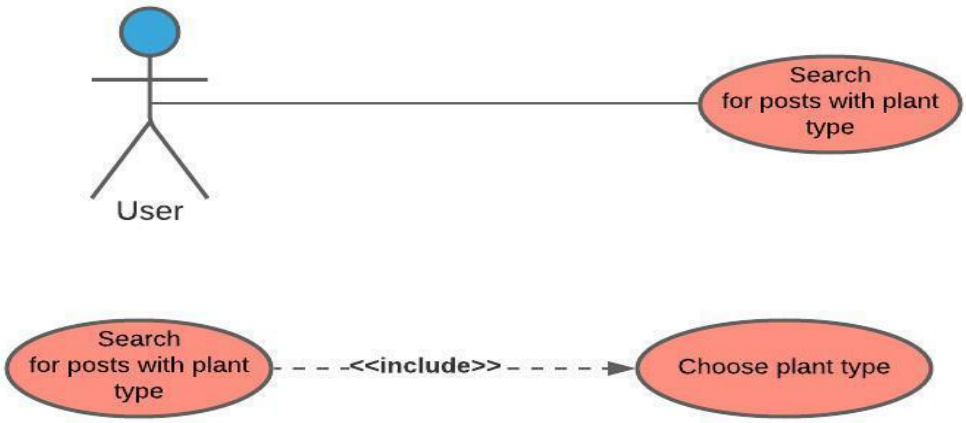


Use Case ID:	NAPTA-User007
Use Case Name:	User Information
Area:	Application
Actor:	User
Description:	Manage Profile Info
Preconditions:	• User must have an account on the application • User must be logged into the application successfully.
Post conditions:	User has successfully logged to his/her account
Triggering Event:	User clicks “Update” button.
Main Flow:	1.Open the application 2. click the update button 3. User updates his/her data 4. User submits his/her data 5. 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.



Use Case ID:	NAPTA-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. • A post must be existed to interact with.
Post conditions:	User has successfully logged to his/her account
Triggering Event:	User clicks “react” icon.
Main Flow:	1. Open the application 2. Select the posts page 3. Interact with a post
Additional info for steps:	Step 2: User can react on any post.

Use Case ID:	NAPTA-User009
Use Case Name:	Talk to a bot
Area:	Application
Actor:	User
Description:	Start talking to a bot, at first select the plant type then select one of the supplied questions
Preconditions:	• User must have an account on the application • User must be logged into the application successfully.
Post conditions:	User will know the answer for his question
Triggering Event:	User clicks “send” icon.
Main Flow:	1. Open the application 2. User opens the chat bot 3. User sends the type of plant desired to know information about 4. Bot response with the available questions for the plant 5. User sends another question 6. Bot send an answer to the user 7. Repeat step 4 to 6 until user stop asking questions



Use Case ID:	NAPTA-User010
Use Case Name:	Search for posts with plant type
Area:	Application
Actor:	User
Description:	User search for specific posts that relate to the selected plant type.
Preconditions:	• User must login to the application • User selects the plant type.
Post conditions:	Application returns the required posts.
Triggering Event:	User clicks “Search” icon.
Main Flow:	• User logins to the application • User selects the plant type • Application returns the required posts.

4.4 Analysis Class

4.4.1 Context Diagram

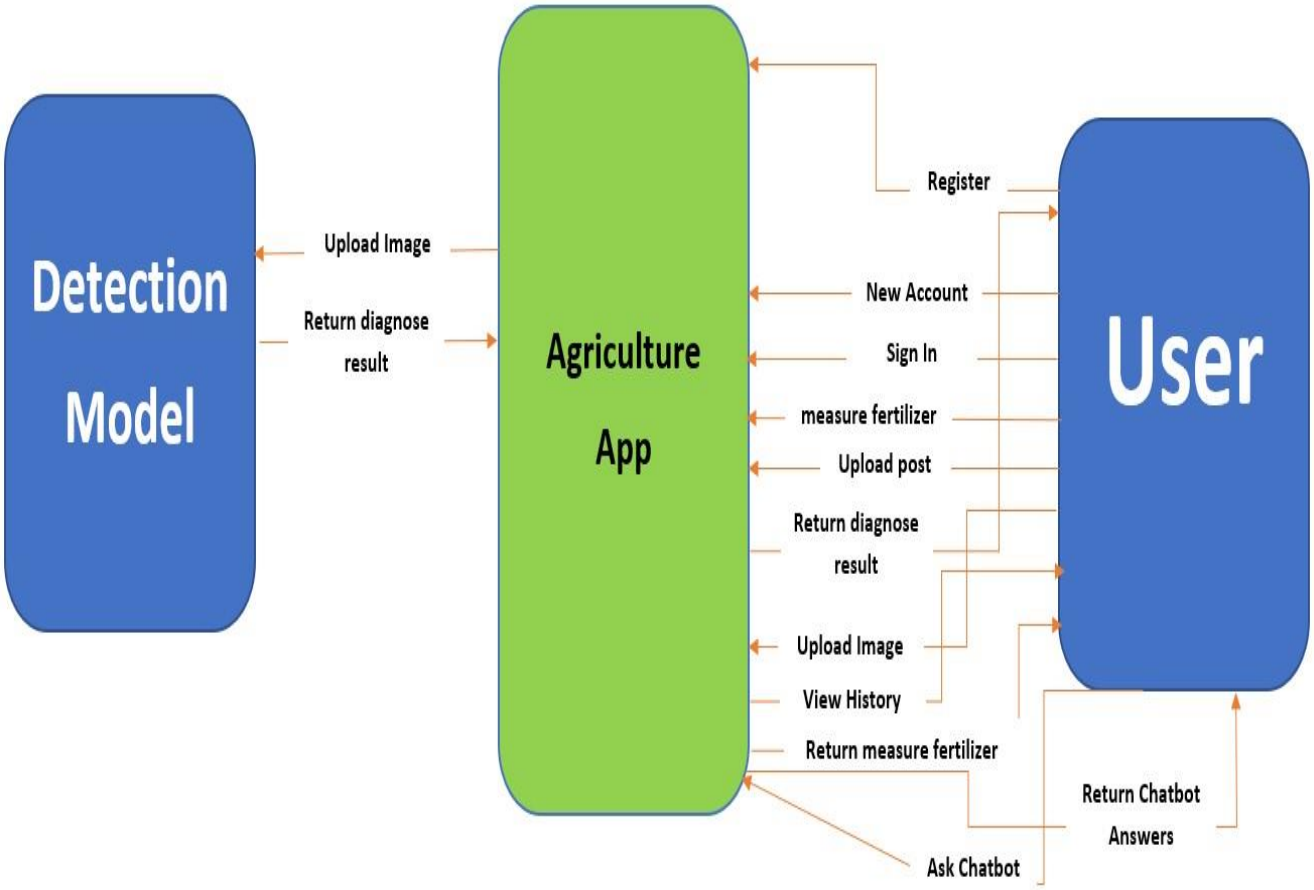


Figure 4. Context diagram

4.4.2 State Diagram

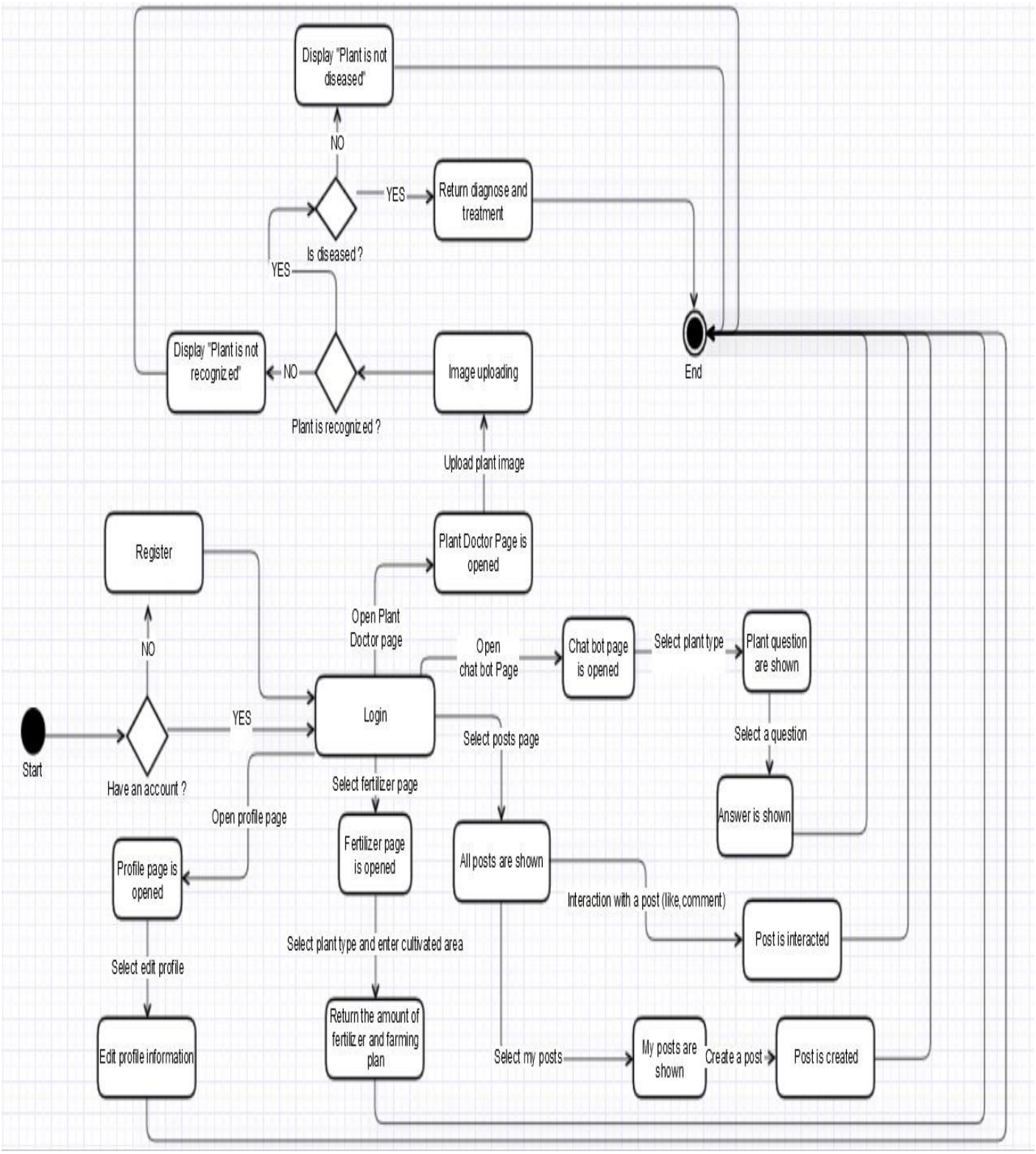


Figure 5. State diagram

4.4.3 Level 0 Diagram

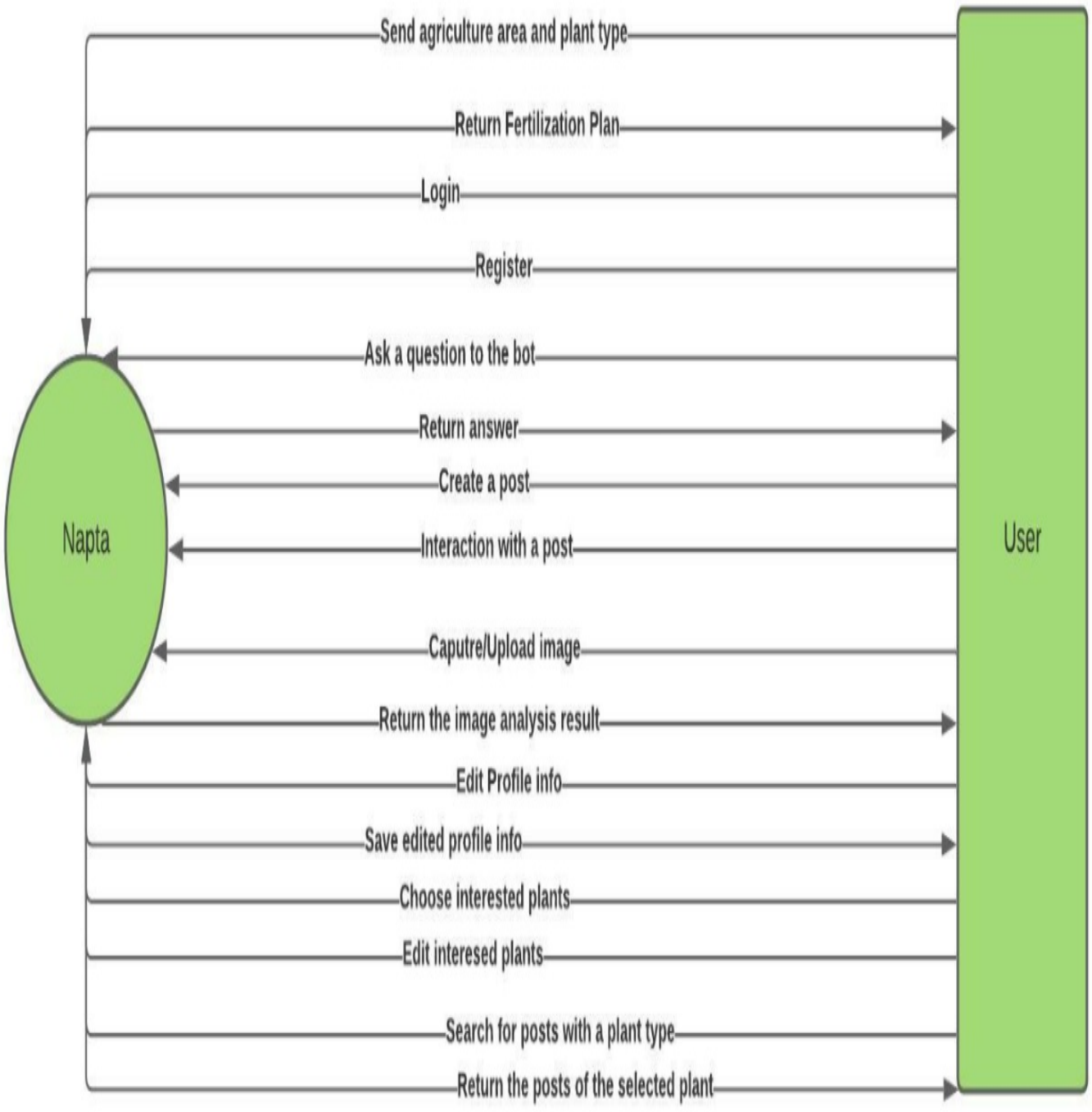


Figure 6. Level 0 diagram

4.5 Interaction Diagram (Sequence Diagram)

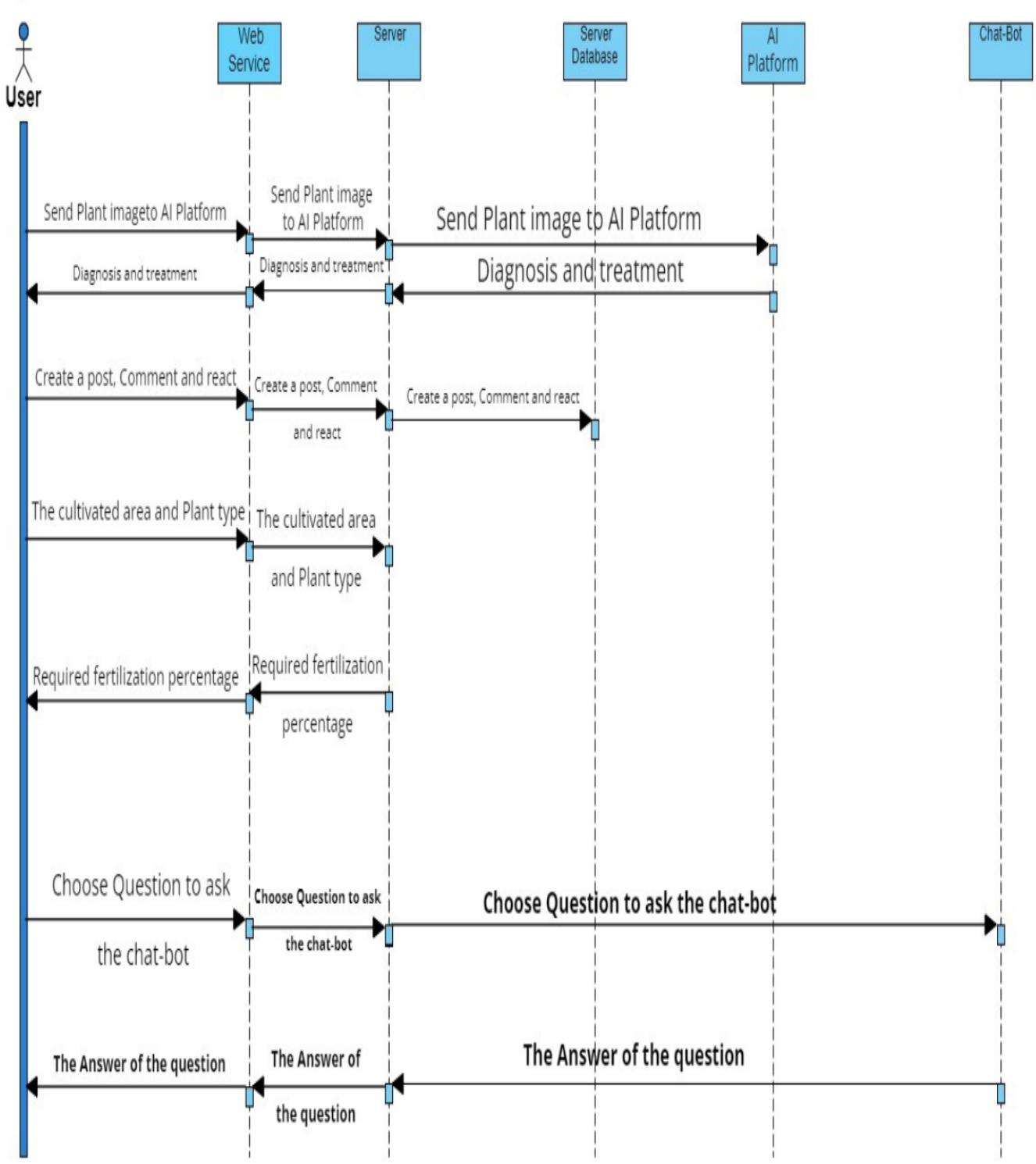


Figure 7. Sequence Diagram

4.6 Design Class

4.6.1 Class Diagram

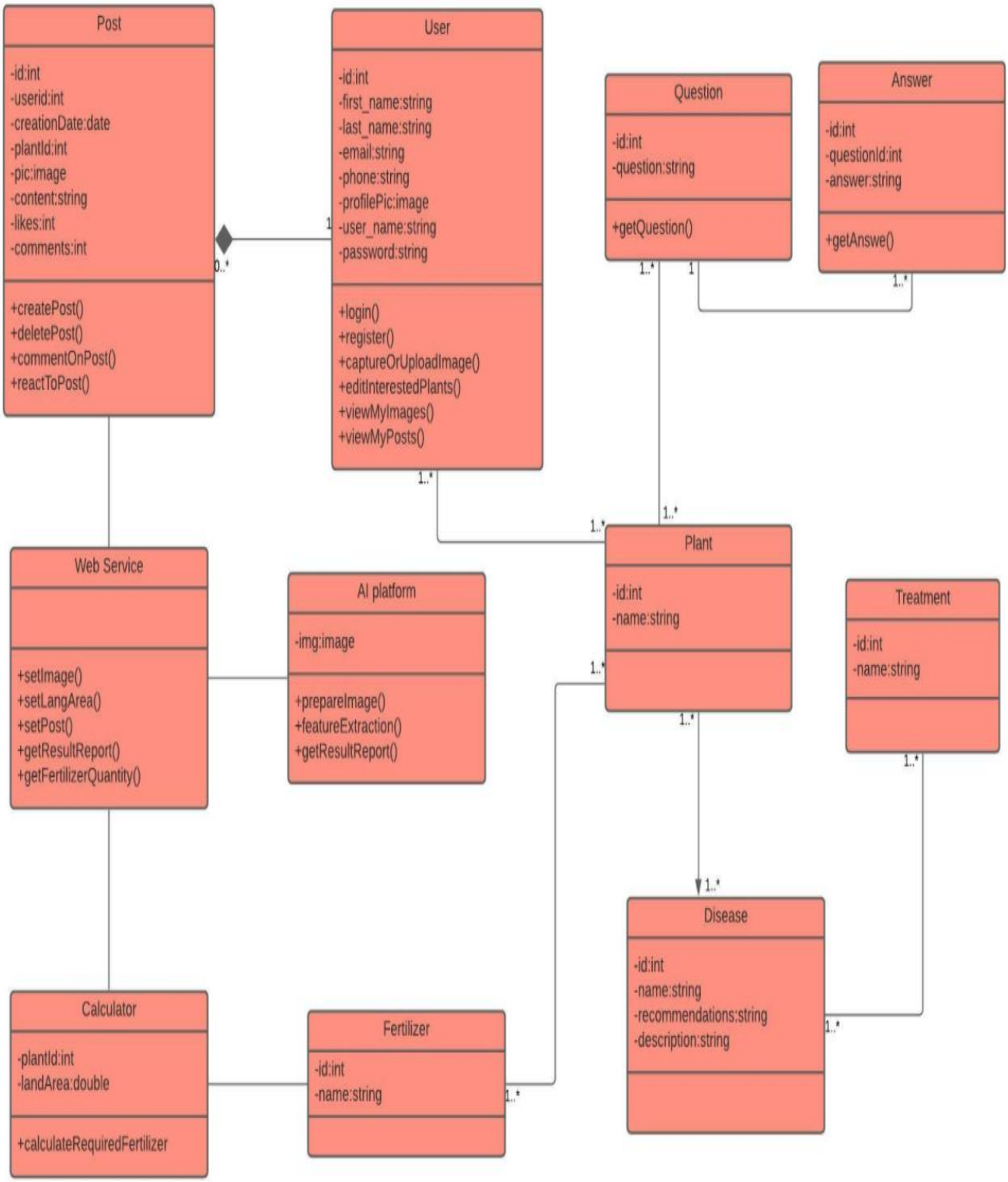


Figure 8. Class diagram

4.6.2 Domain Diagram

Figure 9. Domain diagram

4.7 ER Diagram

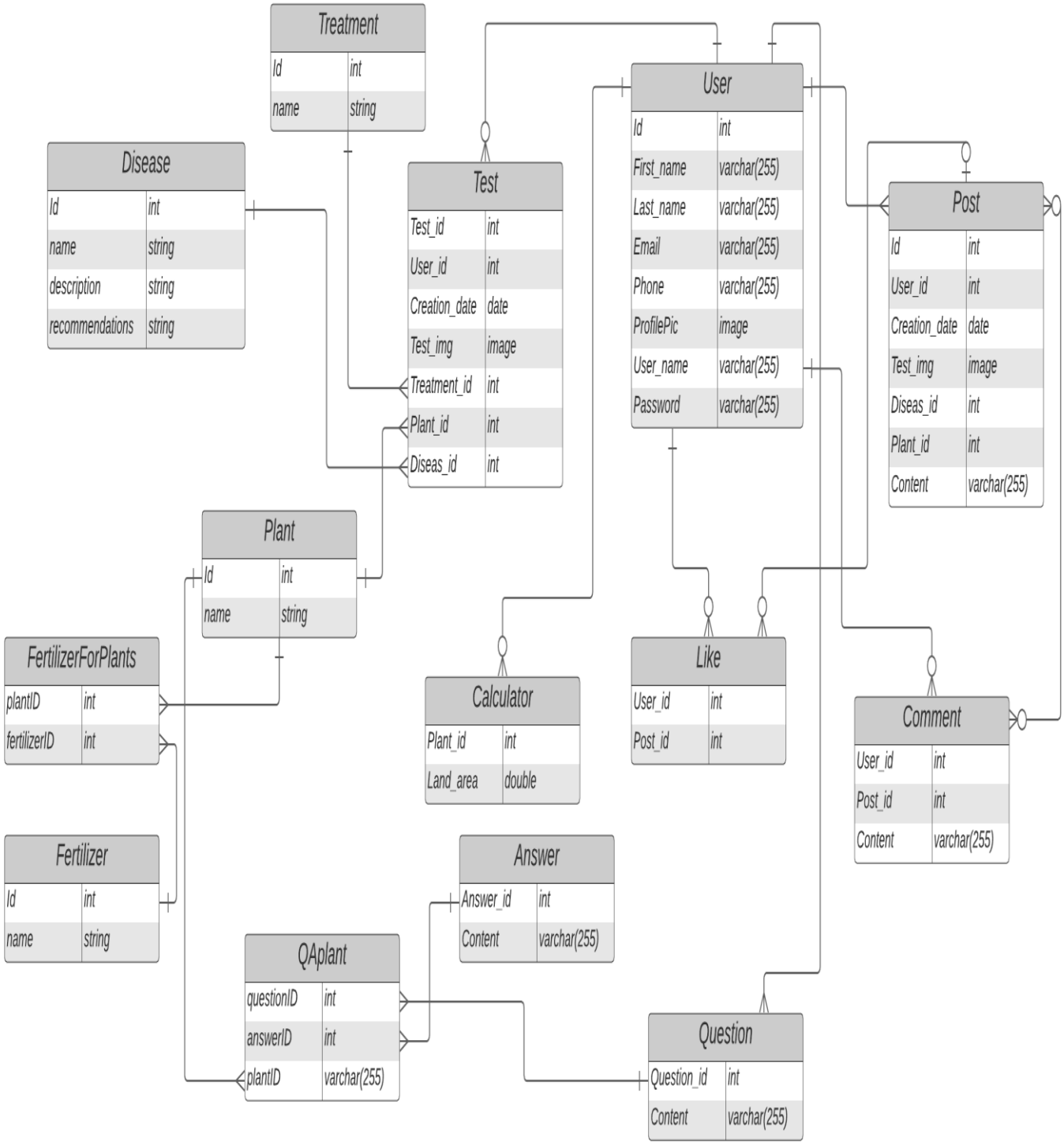


Figure 10. ER Diagram

4.8 Database Schema

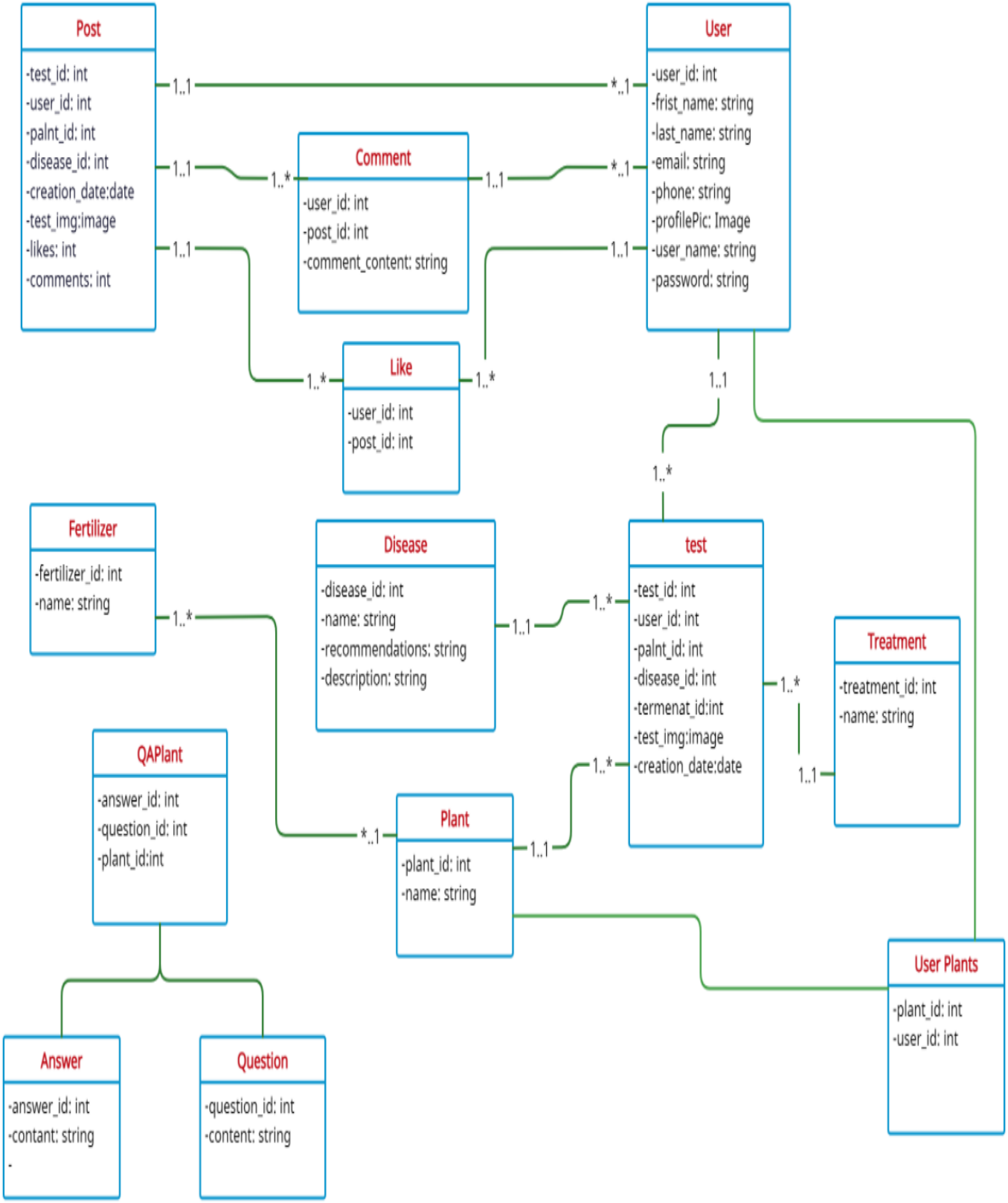


Figure 11. Database Schema

4.9 Design Mockup

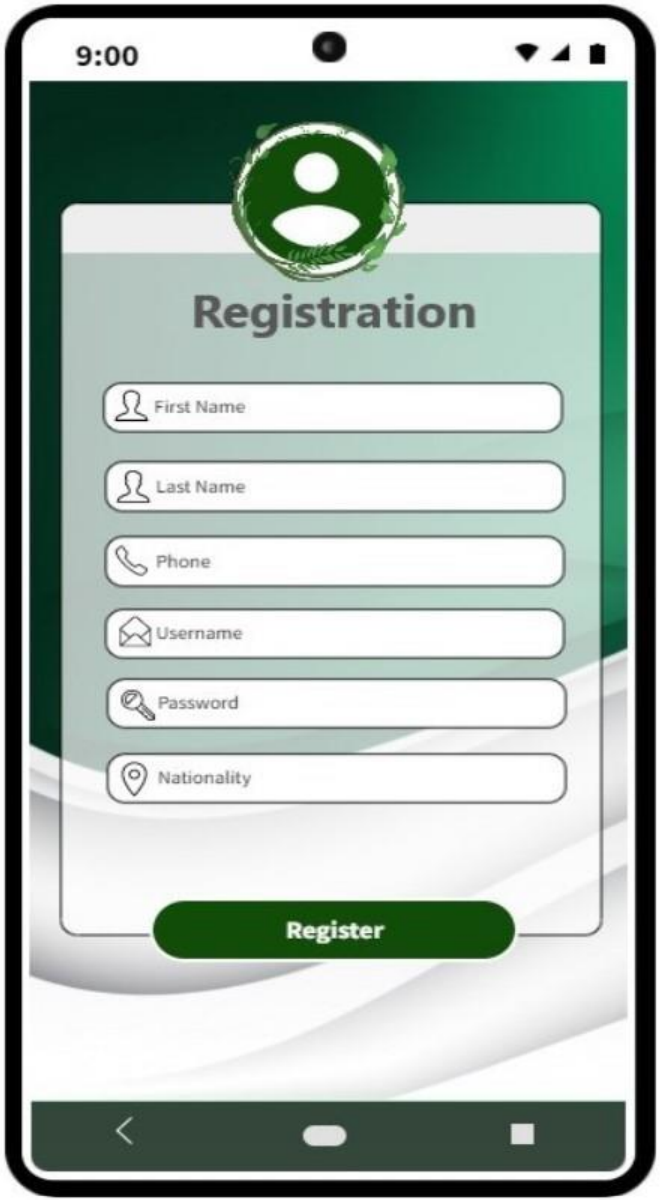


Figure 12. Registration page

Figure 13. Login page

Figure 14. profile page

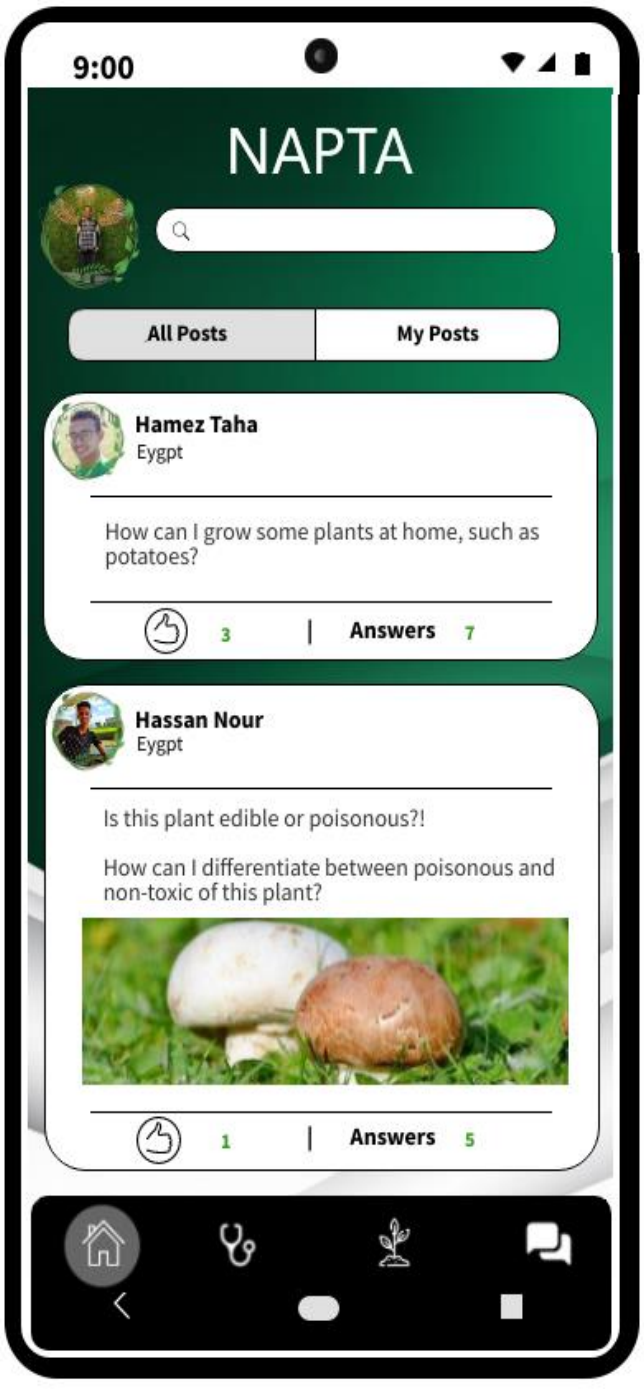


Figure 15. Posts page

Figure 16. Model page

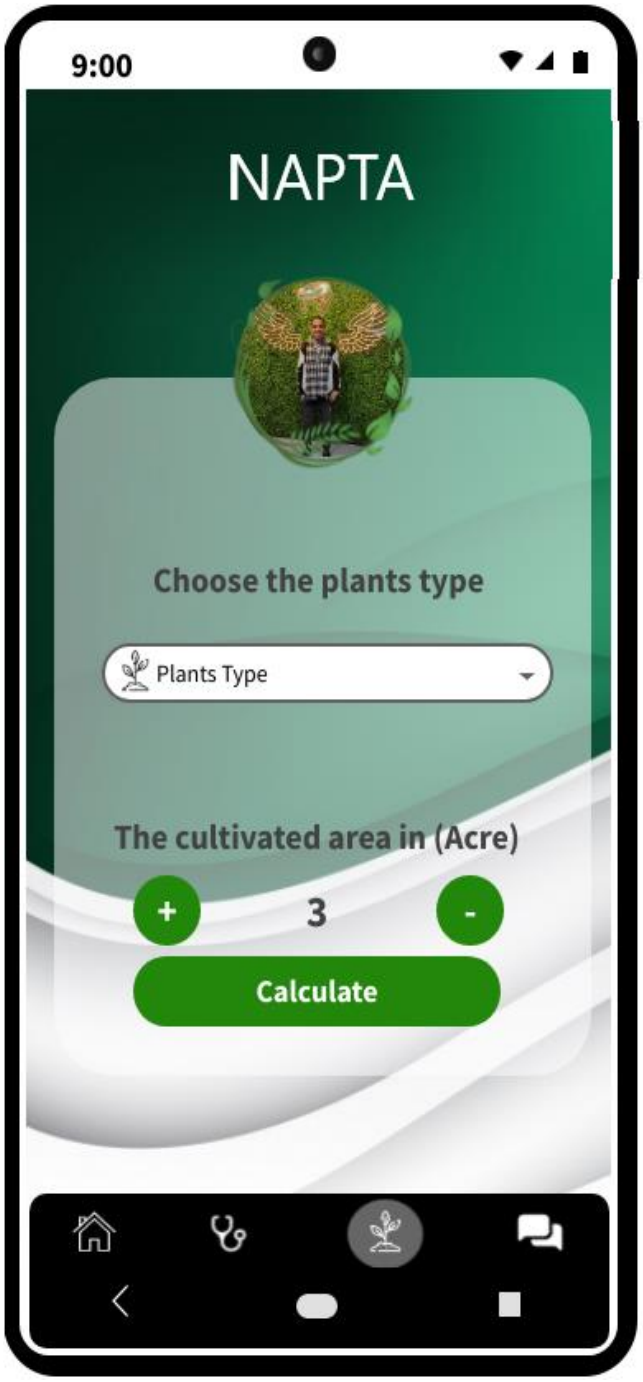


Figure 17. Fertilization Calculate page

Figure 18. Chat Bot page

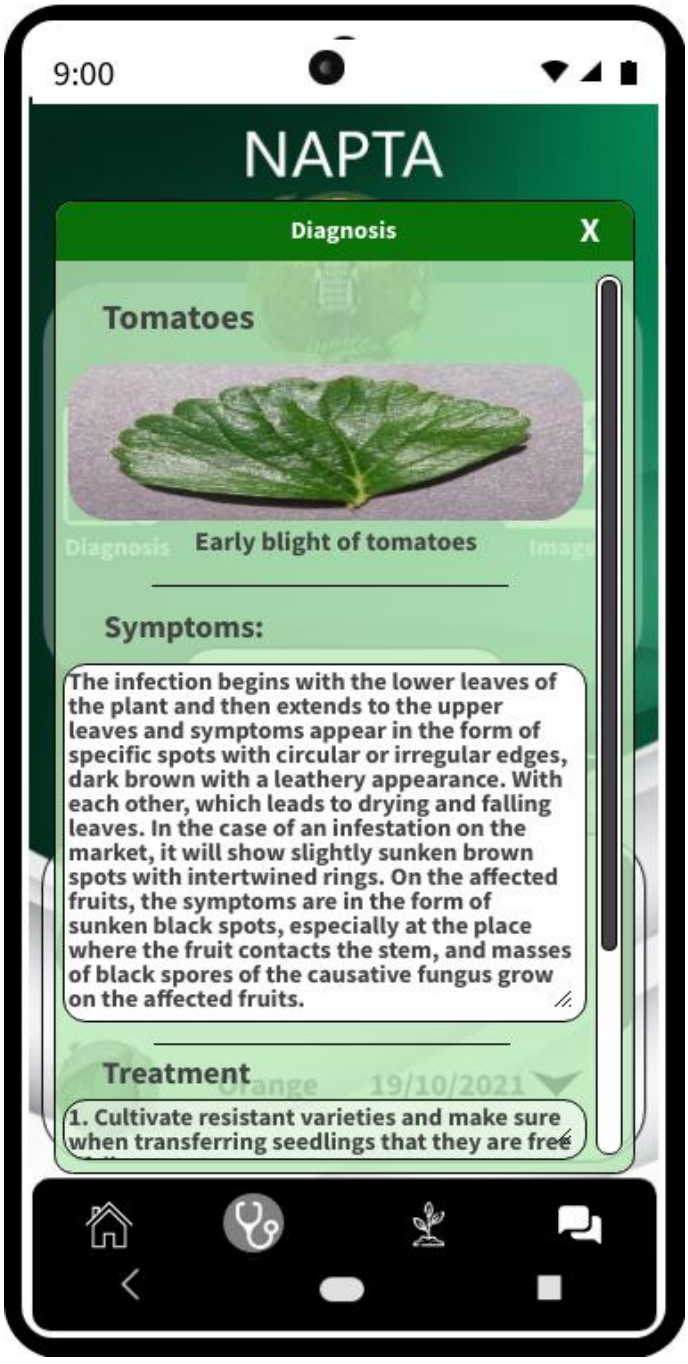


Figure 19. Diagnose result


CHAPTER
5

5.1 References

- Anjanadevi B, Charmila I, Akhil NS, Anusha R", An Improved Deep Learning Model for Plant Disease Detection", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020
- Sharada Prasanna Mohanty, David Hughes, and Marcel Salathé, "Using Deep Learning for Image-Based Plant Disease Detection", the manuscript was compiled on April 15, 2016
- Faye Mohameth, Chen Bingcai1, Kane Amath Sada, "Plant Disease Detection with Deep Learning and Feature Extraction Using Plant Village", School of Computer Science & Technology, Dalian University of Technology, Dalian, China School of Computer Science & Technology, Xinjiang Normal University, Urumqi, China ,Journal of Computer and Communications, 2020, 8, 10-22
- Shima Ramesh Maniyath, Vinod P V, Niveditha M, Pooja R, "Plant Disease Detection Using Machine Learning", 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control
- P.Prathusha, K.E.Srinivasa Murthy, K.Srinivas, "Plant Disease Detection Using Machine Learning Algorithms", Advances in Computational and Bio-Engineering (pp.213-220)
- Aravindhan Venkataramanan, Deepak Kumar P Honakeri, Pooja Agarwal, "Plant Disease Detection and Classification Using Deep Neural Networks", Aravindhan Venkataramanan et al. / International Journal on Computer Science and Engineering (IJCSE).
- G.Saradhambal, R.Dhivya, S.Latha, R.Rajesh, "Plant disease detection and its solution using image classification", January 2018International Journal of Pure and Applied Mathematics 119(14):879-883

- Rehan Ullah Khan, Khalil Khan, Waleed Albattah, Ali Mustafa Qamar, "Image-Based Detection of Plant Diseases: From Classical Machine Learning to Deep Learning Journey", June 2021Wireless Communications and Mobile Computing 2021(12):1-13.
- Konstantinos Ferentinos, "Deep learning models for plant disease detection and diagnosis", February 2018Computers and Electronics in Agriculture 145:311-318
- Isobel Tomlinson, "Doubling food production to feed the 9 billion: A critical perspective on a key discourse of food security in the UK", Journal of Rural Studies Volume 29, January 2013, Pages 81-90, Available online 19 October 2011.