Mansoura University

Faculty of Computer & Information Sciences

Computer Science Department

2023-2024

ACADEMIC YEAR

GRADUATION PROJECT

AGRICULTURE ENHACEMENT USING AI AND IOT WITH SATELITE IMAGERY

CS-10

Supervisor
Dr. Mohamed Handosa

Team Members

Name	Track
Seif Hesham Gomaa Ali (Team Leader).	Artificial Intelligence
Mohamed Ahmed Sabry Ali.	GIS/System Analysis
Abd El-Rahman Ebrahim El-Said Karawia.	Artificial Intelligence
Eman Sami Mahmoud Mohamed.	Artificial Intelligence
Ahmed Bahgat Mohamed Elbaz.	Mobile App (Flutter)
Samir Magdy Ishak Yakop.	Back End (.Net)
Mohamed Diaa Abd El-Aziz Abd El-Qadir.	Embedded System
Haidy Essam abd El-kader abd El-hamid.	Embedded System
Fatma Al-Zahraa Ahmed Fathy Abd El-Aziz.	Front End/ UI-UX
Fatma Al-Zahraa Attiya Al-bahar.	UI/UX

AGRICULTURE ENHANCEMENT USING AI AND IOT WITH SATELLITE IMAGERY

Department of Computer Science
Mansoura University, FCIS
Mansoura

GP Documentation

May 5, 2022

Abstract: Agricultural crops are greatly affected by the changes in environmental factors and the different appropriate conditions for each plant in terms of irrigation, the appropriate nutrients, the impact of this on the country's agricultural economy, and raising the proportion of imports. In global history, there are many disasters in the agriculture field, one of the most direct ways disasters affect agriculture is through lower-than-expected production. This causes direct economic loss to the farmers which can cascade along the entire value chain even affecting the growth of the sector or entire national economies [1]. Therefore, our project aims to help farmers and make the cultivation process easier by following up on the surrounding environmental changes. Via sensors, AI, and satellites, we can do this by following the plant day-by day, controlling the amount of water for irrigation, fertilizers needed at each part, and early detection of plant and soil diseases. Using deep-learning to detect wheat diseases and harmless insects and grasses, making use of all these images from different ways of capturing images like Satellite Imagery or Drone embedded with an embedded camera to capture images, so for images from the satellite we are going to follow up the health of the plants with monitoring some different factors like vegetation index, … etc., and by using images of the plants from the Drone we can make early detection of any kind of the diseases could affect plants using AI algorithms for object detection and the detection of the harmful weed and insects by coordinating or targeting them exactly then eliminate and this process is done by recommending the right pesticides based on the type of plant and insects on that position.

Keywords: Machine Learning, Deep Learning, Computer Vision, Image Classification, Multi-Class Classification, CNN, Tensor Flow, Satellite Imagery, IoT, Sensors, Agriculture, Plants, Crops, Irrigation, Soil. Drone

Acknowledgment

Apart from our efforts, the success of the project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We would like to express our deep and sincere gratitude to our supervisor:

Dr. Mohamed Handosa

We would like to thank our supervisor, Professor Handosa, for bringing the weight of his considerable experience and knowledge to this project. His high standards have made us better at what we do.

Eng. Hassan Khaled

We would like to express our special thanks to our mentor Eng. Hassan Khaled for the time and efforts he provided throughout the year. Your useful advice and suggestions were really helpful to us during the project's completion.

Table of Contents

Chapter 1

Introduction

1.1 Introduction

Agriculture is a key sector in the Egyptian economy, providing livelihoods for 57 percent of the population and directly employing about 26 percent of the labor force. Though its share of gross domestic product (GDP) has fallen to about 11 percent, farming is a vital source of exports and foreign exchange accounting for 20 percent of export revenue. Egypt's total agricultural crop production has increased by more than 20 percent in the past decade, and the area used for the cultivation of crops and the production of animals has generally increased in the period under review.

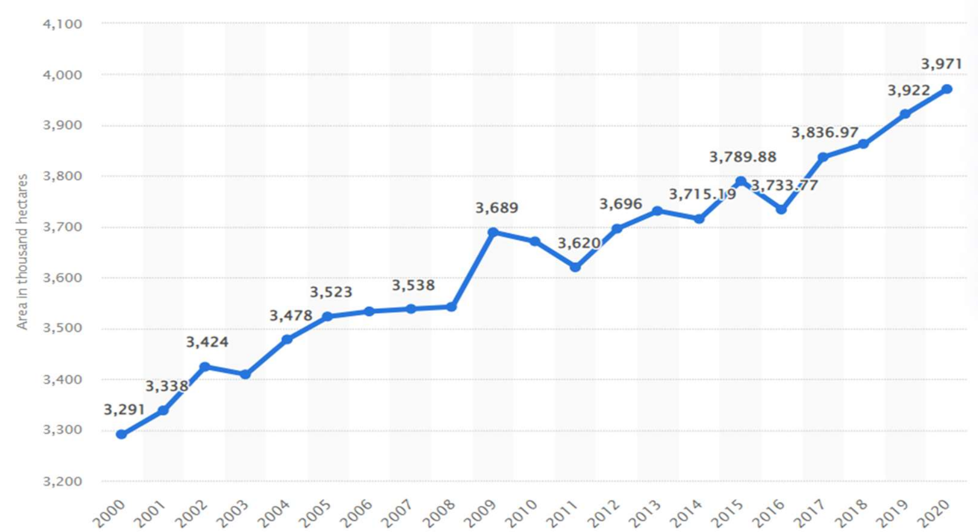


Figure 1. Agricultural land in Egypt from 2000 to 2020

The data indicate that the wheat crop is exposed to many factors of damage and loss, which may reach 30%, meaning that about one-third of the average individual consumption of wheat is wasted and lost, which constitutes a great loss to the Egyptian national economy. A high percentage of these losses is occurring because the presence of many soil and crop diseases and the presence of harmful weeds and insects that negatively affected the crops. So, we decided to participate in solving these problems to reduce these mass losses, and because of the modern technology and its role in solving many large problems in our daily life, we aim to use this modern technology and the rapid improvements in the artificial intelligence field and other modern fields such as remote sensing and IoT to solve the agriculture problems.

Artificial intelligence (AI) helps us to solve many of these problems, as it solves many crop diseases, especially using deep learning and computer vision by capturing images of crops using an esp32 camera sensor embedded in an auto-driven car and the deep learning model using CNN will early detect if this area of the crop has a disease before it spreads out. Using machine learning techniques, we can also monitor anomalies in plants and soil.

And using CNN we can detect harmful weeds and insects. And our application will recommend the correct treatment for these detected weeds and insects.

The United Nations expects Egypt to suffer from water scarcity by 2025, and a high percentage of Egypt’s water is consumed for irrigation and agriculture.

Egypt's Water Consumption by percentage

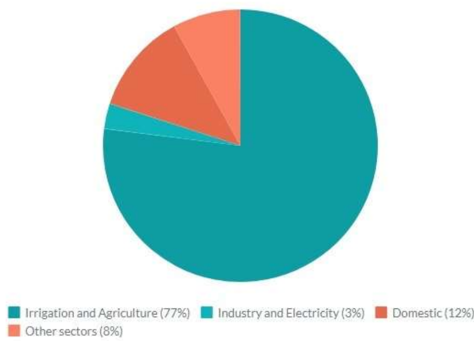


Figure 2. Egypt Water Consumption

So, we will use a water level detection sensor to know the amount of water in the field and determine the required amount for irrigation to rationalize the consumption of water in fields to decrease Egypt's consumption of water to solve this water scarcity problem.

1.2 Problem Definition

Agriculture is considered one of the most occupations in the world nowadays, being the main source of feeding the world, but it is facing many difficulties and problems in recent times that affect the mass production of agricultural crops. In global history, there are many disasters in the agriculture field, they cause serious damage to the global economy and cause many famines some of these problems are:

- **Un-Rehabilitation of the soil nutrients**: One of the most problems that the crops suffer from is that the soil nutrients ratios are disproportionate with the needs of the crops' nutrients. This problem negatively affects the growth of crops.
- **Plants and soil diseases**: The presence of plant diseases on an agricultural farm costs farmer a lot of money such as:
  I. **Soil rot** occurs due to excessive watering.
  II. **Fungi**, there are some types of fungi that prevent the uptake and flow of water.
  and nutrients through the plant, which may cause the plant to wilt.
  III. **Rust diseases** that appear on leaves reduce the percentage of crops by 20%.
- **Detecting weeds**: Many weeds and broadleaf weeds are a problem in wheat fields, they absorb a high percentage of soil nutrients and water by acting as a host to pests and diseases, as well as pollute the purity of wheat grains that are harvested and thus lowering their quality, crop losses caused by weeds usually range from 10% to 80%.
- **Wasting a lot of water and fertilizer**: Most farmers consume large amounts of water and fertilizers without considering the plant's need for them. We cannot accurately control the amount of water added to the plant, which makes the distribution efficiency very low. Soil erosion is impossible to avoid. Therefore, the amount of water lost is very large and requires a lot of labor to repair.

- **Lack of knowledge**: A high percentage of farmers have little knowledge of crops and soil diseases, so they cannot handle these problems well. They need agronomists to help them deal with these problems.

All these problems affect this $5 trillion industry which aims to feed an estimated 9.7 billion people by 2050, therefore we must find an innovative way to eliminate or at least reduce these problems.

Artificial Intelligence can be considered the best technological solution to handle the exponential growth rate of population and unpredicted climate changes. AI has been developed in a significant way in the last 15 years in a way that makes it possible to make a real and significant change in the agriculture process.

1.3 Project Objectives

- **Participating in the Egyptian vision for agriculture improvement**: Egypt has reclaimed 217,700 acres of agricultural lands in different areas over the recent years that produce around 550,000 tons of wheat annually. Egypt has also reclaimed 63,900 acres of wheat in Toshka, 145,200 acres in Sharq el-Owainat, 4,550 acres in the Dalla Spring area, and 4,050 acres in Farafra Oasis. Therefore, we are trying to participate in this vision by using modern technologies such as artificial intelligence and remote sensing to improve the quality of crops, especially wheat.
- **Utilizing the developments and improvements in satellites and AI fields to improve precision agriculture**: Our project uses advances in deep learning and computer vision algorithms to detect some of the wheat diseases and soil problems, also we will use satellite imagery, image processing, and remote sensing to detect changes in the field and crops and solve the problems whenever they pop.
- **Detecting soil type before farming**: Our project will detect the soil type to know if the soil is suitable for this type of plant or not.
- **Observing and monitoring crops using remote sensing**: we aim to use optical (VIR) sensing to observe the fields and make timely crop management decisions, as optical (VIR) sensing allows us to see beyond visible wavelengths like infrared, and these wavelengths are very sensitive to crop vigor, damage, and stress.
- **Detecting soil nutrients such as N, K, Pa, etc.**: Using sensors for measuring nutrients in the soil, we can detect if there is any problem with the percentage of nutrients required for this specific type of plant, and if there is a problem the application will inform the farm owner about this to find the suitable solution before it becomes so risky.
- **Early detection of plant and soil diseases**: Using artificial intelligence and deep learning, our project can detect plant and soil diseases using an existing camera sensor that tracks the plants in real-time day by day.
- **Preventing plants from harmless insects and grasses**: Computer vision techniques will be used to detect harmless insects and weeds before spreading and damaging the crops. So, we aim to prevent the plants from these dangers, by informing the farm owner if there is one of these problems to get rid of these insects and grasses.
- **Rationalize consumption of water in fields**: We will use a humidity sensor and a water level sensor to determine the amount of water in that soil and then we can determine the required amount of water at this moment. This will save a high percentage of water required for irrigation.
- **Facilitation following the agriculture market and economy**: We will provide application users with the current news about the agriculture market and its economy.

1.4 Project Scope

1.4.1 Identify the target audience:

The target audience for the project we divided our target audience into two segments 1st income and 2nd Geography According to Income we target Agricultural business owners and Agricultural Engineers 2nd According to Geography we target Farmland owners seeking extra improve especially in Delta Egypt.

1.4.2 Project Deliverables:

we develop mobile Application connect with satellite data and IOT systems.

1.4.3 Acceptance Criteria:

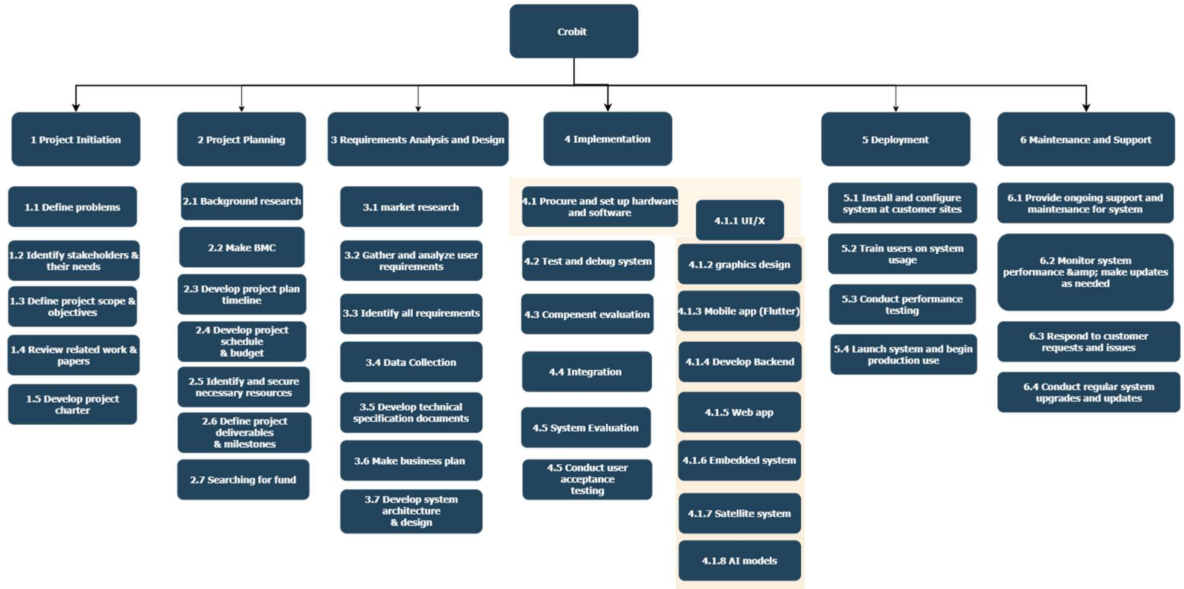
We solve the problems facing farmers or owners of agricultural land such as trying to follow the land, wasting a lot of water and fertilizer, and the main goal is to make farming easy.

1.4.3.1 Establish timeline and budget:

- **Budget Plan**

Item	Quantity	price	Total cost
ESP32 CAM	2	350	700
FTDI	1	100	100
Soil moisture sensor	1	60	60
Soil NBK sensor	1	4400	4400
DHT-22	1	130	130
Water level sensor	1	25	25
3d printing	1	1500	1500
Brushless Motor 2200KV/6T	4	350	1400
RYLR896	1	617	617
18650 Lion Battery	8	60	480
tp4056 Lion battery charger	1	65	65
ESP	1	290	290
			9767 EGP

1.4.3.2 Tasks and Timeline



#	Task name	Duration
1.1	Define problems	1 week
1.2	Identify stakeholders and their needs	3 day
1.3	Define project scope and objectives	3 day
1.4	Review related work and papers	1 week
1.5	Develop project charter	15 day
2.1	Background research	1 week
2.2	Make BMC	1 day
2.3	Develop project plan timeline	3 day
2.4	Develop project schedule and budget	3 day
2.5	Identify and secure necessary resources	3 day
2.6	Define project deliverables and milestones	3 day
2.7	Searching for fund	1 week
3.1	Market research	3 week
3.2	Gather and analyze user requirements	1 week
3.3	Identify all requirements	1week

3.4	Data collection	1 month
3.5	Develop technical specification documents	1 week
3.6	Make business plan	3 moths
3.7	Develop system architecture and design	2 week
4.1	Procure and set up hardware and software	
4.1.1	UI/X	1 month
4.1.2	graphics and design	3 weeks
4.1.3	Mobile app (Flutter)	2 months
4.1.4	Develop Backend	2 months
4.1.5	Web app	5 days
4.1.6	Embedded system	1.5 months
4.1.7	AI models	2 months
4.1.8	Satellite system	2 months
4.2	Test and debug system	3 days
4.3	Component evaluation	2 days
4.4	Integration	4 days
4.5	System evaluation	3 days
4.6	Conduct User acceptable testing	1 week

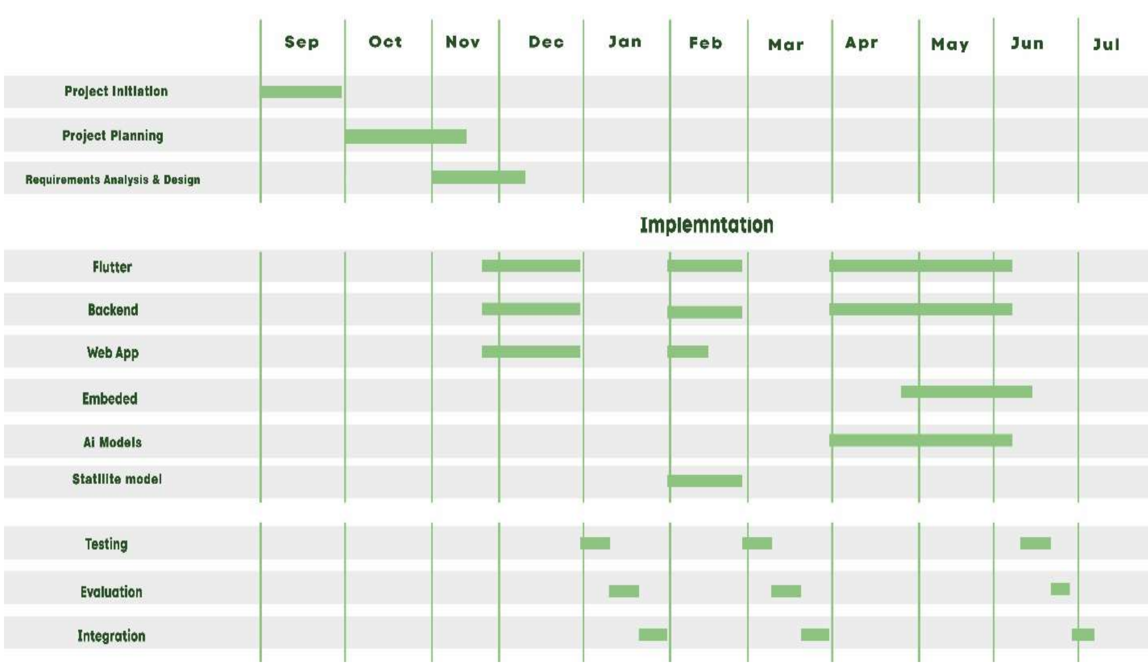


Figure 3 Timeline

Chapter 2

Literature Review

Literature Review

2. Introduction

In this chapter, we provide a background about the tools and techniques needed to build our system. We also review the previous related work to our system, and the common technologies that are used. In the end of the chapter, we provide a comparison between the system we are going to build and the related systems.

2.1 Background

As we have previously mentioned that we are in the era of artificial Intelligence and IOT. Artificial intelligence (AI) helps us to solve many of these problems, as it solves many crop diseases, especially using deep learning and computer vision by capturing images of crops using an esp32 camera sensor embedded in an auto-driven car and the deep learning model using CNN will early detect if this area of the crop has a disease before it spreads out. Using machine learning techniques, we can also monitor anomalies in plants and soil. And using CNN we can detect harmful weeds and insects. And our application will recommend the correct treatment for these detected weeds and insects.

2.2 Review of Relevant Work

Lots of projects and systems have been developed over the last 10 years, these systems are not in the maturity level we are hoping for so far, but the curve of the developing rate is great makes it possible in the following 10 years to make a full AI system for farms with no aid from farmers. In the following list there is a sample of the research and competitors:

2.3 Researches

2.3.1 agriculture "Survey and challenges"

Undoubtedly, high demands for food from the world-wide growing population are impacting the environment and putting many pressures on agricultural productivity. Agriculture, as the fourth evolution in the farming technology, puts forward four essential requirements: increasing productivity, allocating resources reasonably, adapting to climate change, and avoiding food waste. As advanced information systems and Internet technologies are adopted in Agriculture, enormous farming data, such as meteorological information, soil conditions, marketing demands, and land uses, can be collected, analyzed, and processed for assisting farmers in making appropriate decisions and obtaining higher profits. Therefore, agricultural decision support systems for Agriculture have become a very attractive topic for the research community. The objective of this paper aims at exploring the upcoming challenges of employing agricultural decision support systems in Agriculture.

For Egypt, the country has been aiming to expand the agricultural area for a long time through the so-called Toshka agricultural project, which seeks to reclaim deserts to the agricultural area. Indeed, it has contributed to adding an area of up to 540 thousand acres to the agricultural area.

2.4 Technology in agriculture

2.4.1 IoT Based Smart Agriculture System

Smart Agriculture system is an aborning topic in this materialistic world. This paper describes the concept of featuring and elating an agriculture platform to the internet world. Agriculture is the most important of human life so it can be improvised by using IoT technology. IoT technology gives a grasp to enhance the power of automation systems in agriculture. Smart agriculture System that uses the advantages of cutting-edge technologies such as Arduino and Wireless Sensor Network. This paper proposes the concept and features of the sensor world in the internet of things for agriculture which is used to enhance the production of crops.

The agriculture stick being proposed through this paper is integrating with Arduino Technology, Breadboard and mixed with different various sensors and live data feed can be obtained online through mobile phone. India Monitoring environmental conditions are the major factor to improve the yield of efficient crops. The feature of this paper includes the development of a system that can monitor temperature, humidity, moisture, and even the movement of animals which may destroy the crops in agricultural fields through sensors using Arduino board. With its energy autonomy and low cost, the system has the potential to be useful in water-limited geographically isolated areas.

2.4.2 Adoption of the Internet of Things (IoT) in Agriculture and Smart Farming

It is essential to increase the productivity of agricultural and farming processes to improve yields and cost-effectiveness with new technology such as the Internet of Things (IoT). In particular, IoT can make agricultural and farming industry processes more efficient by reducing human intervention through automation. In this study, the aim to analyze recently developed IoT applications in the agriculture and farming industries to provide an overview of sensor data collections, technologies, and sub-verticals such as water management and crop management. In this review, data is extracted from 60 peer-reviewed scientific publications (2016-2018) with a focus on IoT sub-verticals and sensor data collection for measurements to make accurate decisions. Our results from the reported studies show water management is the highest sub-vertical (28.08%) followed by crop management (14.60%) then smart farming (10.11%). From the data collection, livestock management and irrigation management resulted in the same percentage (5.61%). Regarding sensor data collection, the highest result was for the measurement of environmental temperature (24.87%) and environmental humidity (19.79%). There are also some other sensor data regarding soil moisture (15.73%) and soil pH (7.61%). Research indicates that of the technologies used in IoT application development, Wi-Fi is the most frequently used (30.27%) followed by mobile technology (21.10%). As per our review of the research, we can conclude that the agricultural sector (76.1%) is researched considerably more than compared to the farming sector (23.8%). This study should be used as a reference for members of the agricultural industry to improve and develop the use of IoT to enhance agricultural production efficiencies. This study also provides recommendations for future research to include IoT systems' scalability, heterogeneity aspects, IoT system architecture, data analysis methods, size or scale of the observed land or agricultural domain, IoT security and threat solutions/protocols, operational technology, data storage, cloud platform, and power supplies.

2.4.3 Smart Agriculture Applications using Deep Learning Technologies

Agriculture is considered an important field with a significant economic impact in several countries. Due to the substantial population growth, meeting people's dietary needs has become a relevant concern. The transition to smart agriculture has become inevitable to achieve these food security goals. In recent years, deep learning techniques, such as convolutional neural networks (CNN) and recurrent neural networks (RNN), have been intensely researched and applied in various fields, including agriculture. This study analyzed the recent research articles on deep learning techniques in agriculture over the previous five years and discussed the most important contributions and the challenges that have been solved. Furthermore, we investigated the agriculture parameters being monitored by the internet of things and used them to feed the deep learning algorithm for analysis.

Additionally, we compared different studies regarding focused agriculture area, problems solved, the dataset used, the deep learning model used, the framework used, data preprocessing and augmentation method, and results with accuracy. We concluded in this survey that although CNN provides better results, it lacks in early detection of plant diseases. To cope with this issue, we proposed an intelligent agriculture system based on a hybrid model of CNN and SVM, capable of detecting and classifying plant leaves disease early.

2.4.4 A comprehensive review on automation in agriculture using artificial intelligence.

Agriculture automation is the main concern and emerging subject for every country. The world population is increasing at a very fast rate and with increase in population the need for food increases briskly. Traditional methods used by farmers aren't sufficient enough to serve the increasing demand and so they have to hamper the soil by using harmful pesticides in an intensified manner. This affects the agricultural practice a lot and in the end the land remains barren with no fertility. This paper talks about different automation practices like IOT, Wireless Communications, Machine learning and Artificial Intelligence, Deep learning.

There are some areas which are causing the problems to agriculture field like crop diseases, lack of storage management, pesticide control, weed management, lack of irrigation and water management and all these problems can be solved by above mentioned different techniques.

Today, there is an urgent need to decipher the issues like use of harmful pesticides, controlled irrigation, control on pollution and effects of environment in agricultural practice. Automation of farming practices has proved to increase the gain from the soil and also has strengthened the soil fertility. This paper surveys the work of many researchers to get a brief overview about the current implementation of automation in agriculture. The paper also discusses a proposed system which can be implemented in botanical farm for flower and leaf identification and watering using IOT.

2.5 Competitors

2.5.1 Hits Solutions

HITS is an international company that was founded in 1998 as one of the pioneer software companies (ISV), specialized in the field of Business Applications, Cloud, AI, IoT and Bioinformatics. It provides a smart agricultural system with over than 100 sensors that allow customers to track, monitor and manage the farm.

Figure 4. HITS Solutions Logo

2.5.2 PlatFarm

Platfarm is a UAE company, it is a data platform that consumes and analyzes satellite imagery to enable variable rate irrigation and fertilization for sustainable agriculture in the MENA region. It provides weather forecast data daily and according to each farm's location, it supports weather data like temperature, humidity and wind, it also monitors plant growth and vegetation rates.

Figure 5. PlatFarm Logo

1. VAIS:

Visual and Artificial Intelligence Solutions (VAIS) is a deep-tech Egyptian company that develops innovative and proprietary AI algorithms and solutions to use in the domains of Precision Agriculture and Geospatial Intelligence. They provide centralized farm, field management, monitoring, on-the-fly generation and display of aggregated/detailed farm/field statistics.

Figure 6. VAIS Logo

2.5.3 Plantix:

The Plantix app covers 30 major crops and detects 400+ plant damages just by taking a photo of a sick crop. It's available in 18 languages. This makes Plantix one of the most popular agricultural apps for damage detection, pest and disease control and yield improvement for farmers. The application failed because it didn't focus on specific diseases, but rather all diseases and that's why its accuracy was reduced.

Figure 7. Plantix Logo

2.5.4 Agrio:

Helps growers and crop advisors to forecast, identify and treat plant diseases, pests and nutrient deficiencies. Agrio leverages and deploys proprietary artificial intelligence and computer vision algorithms to help farmers, crop advisors and agronomists to manage crops and improve yield. The digital plant doctor contains the knowledge of numerous agriculture experts from all over the world and continuously improves. They offer a multi-layered Plant diagnosis app that acts as a plant doctor and helps to identify plant diseases and problems in smartphone-captured images.

Figure 8. Argio Logo

2.5.5 Relationship between relevant work and our work

All Agricultural applications have the same goal, which is provide increasing in the amount of the crop, improving the usage of the water efficiency, reducing input costs or any other desirable outcome related to farming. Each Application uses a different way to achieve these goals. For Example, “VAIS” depends on ground sensors only to display of aggregated/detailed farm/field statistics and detect diseases. “Plantix” detects damage, pest and disease without using either sensors or satellites, it only depends on scanning the crop using camera and use ai model to do this. “Platfarm” centralizes, manages and optimizes the crop monitoring and management activities & operations by using satellite and sensors, but there is no treatment suggestions and ai consultant.

Here in “Crobit” application it differs from other applications in that it focuses all its attention on the most problems in the field of agriculture. Unlike other applications we use Hybrid system using satellites and Sensors for crops observation to detect its diseases, weeds and the soil diseases, improve their efficiency and productivity, reduce costs and use Free features and Competitive prices. We are working on solving agriculture problems like “Un-Rehabilitation of the soil nutrients, Plants and soil diseases, weeds and wasting a lot of water and fertilizer”. We use cameras to follow the plant growth, NPK sensor to measure the nutrients of the soil so the farm can specify the amount of fertilizer it needs, water level sensor to detect the amount of water in the soil, https://store.fut-electronics.com/products/temperature-humidity-sensor-dht22Temperature & humidity sensor to measure the humidity of the atmosphere and soil moisture sensor to measure the humidity of the soil. We use the same technology with increasing models’ accuracy. we are going to follow up the health of the plants with monitoring some different factors using images from the satellite.

2.5.6 Competitors Analysis

Applications	Hits Solutions	PlatFarm	VAIS	Plantix	Agrio	Crobit
Satellites	✗	✓	✗	✗	✗	✓
Sensors	✓	✓	✓	✗	✓	✓
Suggestion	✓	✗	✗	✓	✓	✗
Open Lands	✓	✓	✓	✓	✓	✓
Closed Lands	✓	✗	✗	✓	✓	✓
AI Consultant	✗	✗	✗	✗	✗	✓
Diseases	✓	✓	✓	✓	✗	✓
Awareness	✓	✓	✓	✓	✓	✓
Digital Twin	✗	✓	✗	✗	✗	✗
Spray Advisory	✗	✓	✗	✗	✗	✗

Figure 9. Competitors Analysis

2.5.7 Summary

This chapter talks about the role of information system in addition to review of literature. In this chapter we have identified important attributes to analyze the searches findings in agriculture applications. Our survey shows the strength and weakness of other applications, here we try to show our steps to cover all the weaknesses of most of the apps as well as we combine all the strengths of most of the apps.

Chapter 3

System Analysis

3. System Analysis

3.1 System Requirements

System requirements are the needed configurations for the system to operate efficiently. The next three subsections will discuss the functional requirements, Nonfunctional requirements, and user requirements.

3.1.1 Functional Requirements

In systems engineering, functional requirements are directly concerned with the system services, where a function is described as a specification of behavior. In this subsection, we list the functions required in our system. We, also, provide a description for each function.

1- **Data collection**: The system should be able to collect and store data from the satellite imagery and IoT devices. This data may include information on weather conditions, soil moisture levels, crop growth, and other factors that impact agriculture.
2- **Data analysis**: The system should be able to analyze the collected data and identify trends, patterns, and different that may be useful for optimizing agriculture practices. This may involve using machine learning algorithms or other data analysis techniques.
3- **Detection modeling**: The system should be able to use the collected data and analysis results to build predictive models that can forecast future crop yields, identify potential problems, and suggest solutions.
4- **Decision support**: The system should be able to provide decision support to farmers or other stakeholders by presenting recommendations based on the collected data and analysis results.
5- **Reporting**: The system should be able to generate reports and visualizations that summarize the collected data, analysis results, and recommendations in a clear and understandable way.

3.1.2 Non-Function Requirements

The system needs to operate efficiently and meet the requirements. Any failure of the components of the systems may lead to one or more of the functions to stop or be misused. A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system.

1- **Performance**: The system should be able to process and analyze data quickly and efficiently, with minimal latency.
2- **Reliability**: The system should be able to operate continuously and consistently, without downtime or errors.
3- **Usability**: The system should be easy to use and understand, with a user-friendly interface and clear instructions.
4- **Maintainability**: The system should be easy to maintain and update, with clear documentation and a robust architecture.
5- **Scalability**: The system should be able to handle large amounts of data and be easily scalable as the needs of the project evolve.
6- **Security**: The system should have robust security measures in place to protect the collected data and ensure the integrity of the analysis and decision-making process.
7- **Legal compliance**: The system should adhere to any relevant laws, regulations, and standards related to data privacy and security.
8- **Integration with other systems**: The system should be able to integrate with other systems, such as GIS software, to enable more comprehensive analysis and decision-making.

3.1.3 User Requirements

The users of the application are a farmland owners or Agriculture Engineering or Agriculture hobbyist who wants to know the result of his crops و wants to enhance the agricultural process. And this includes Chat with the chatbot for some advice, Crop scanning and disease detection.

The following list will show the requirements.

1- reports
2- Get the explanation of the result of reports.
3- Alerts
4- Speaking with chatbot
5- Real time scan for crops
6- Guidance
7- Ease of use
8- Accuracy
9- Timeliness
10- Customization
11- Support

Explain some Requirements.

1- **Ease of use**: The system should be easy to use, with a user-friendly interface and clear instructions.
2- **Accuracy**: The system should provide accurate and reliable data and analysis results.
3- **Timeliness**: The system should provide data and analysis results in a timely manner, so that users can make informed decisions.
4- **Customization**: The system should allow users to customize the types of data and analysis they receive, based on their specific needs and interests.
5- **Support**: The system should provide adequate support and assistance to users, including training, documentation, and technical support.
6- **Guidance**: The system should guide user and give him an advice and how to care about his plants.

3.2 System Architecture

System architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. Systems Architecture is a response to the conceptual and practical difficulties of the description and the design of complex systems. Systems Architecture helps to describe consistently and design efficiently complex systems.

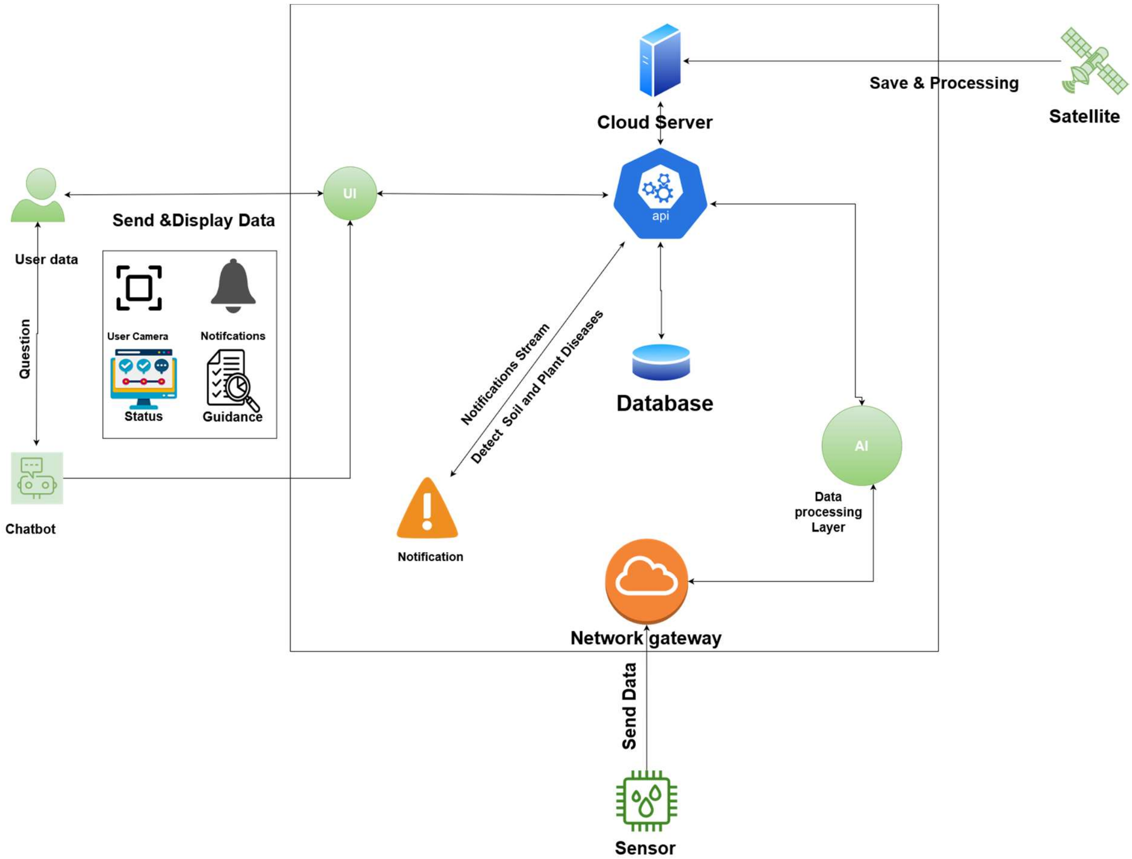


Figure 10. System Architecture

3.3 Development Methodology

After we knew the basic structure of the system. We use Agile methodology, this methodology is iterative and allows for continuous reviews. It is also less focused on rigorous documentation which can be useful when working with new technologies such as AI, IOT and Remote sensing.

We are going to view all of its functions, the relation between them and the sequence of their executions in the following subsections.

3.3.1 Context Diagram

A context diagram is a high-level diagram that represents the system being developed and its relationship to the external environment.

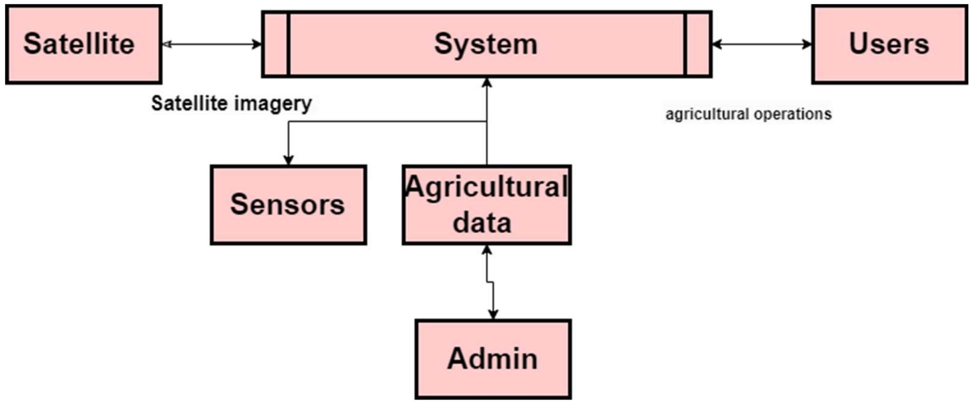
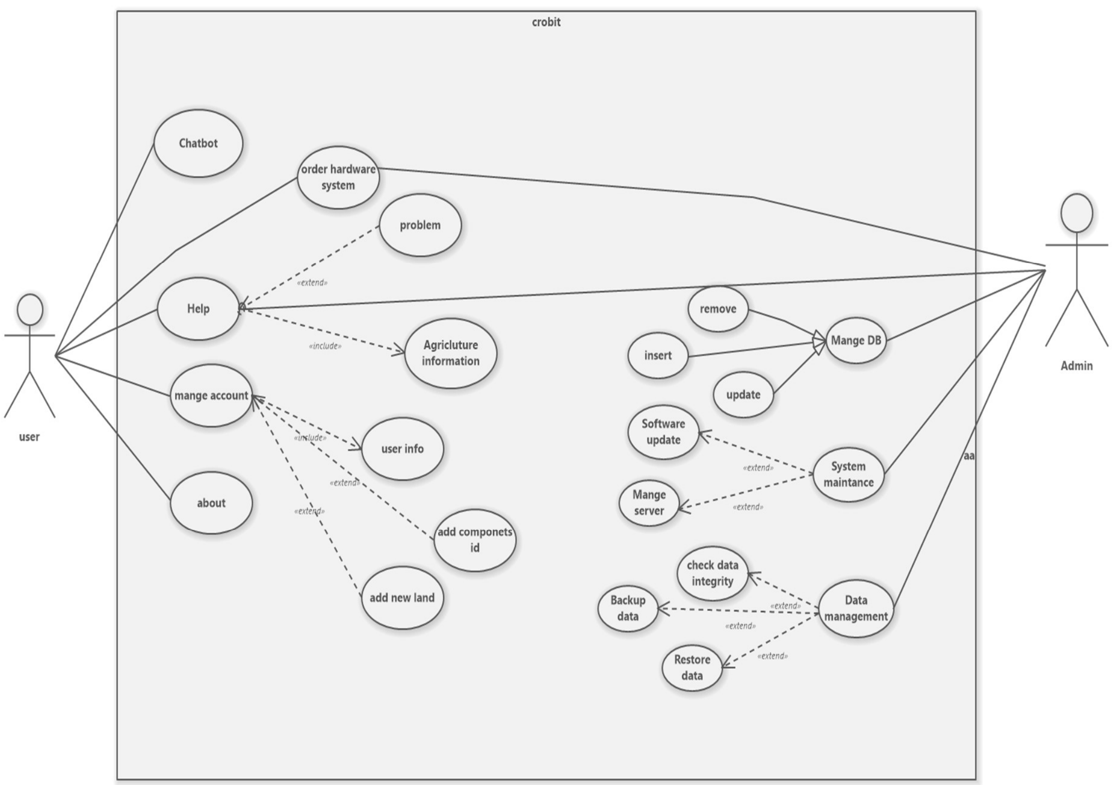


Figure 11. Context Diagram

3.3.2 Use case diagrams

with a use case diagram, we will specify the expected behavior (what) of the system, not the exact method of making it happen (how). This helps us to design the system from the end user's perspective.



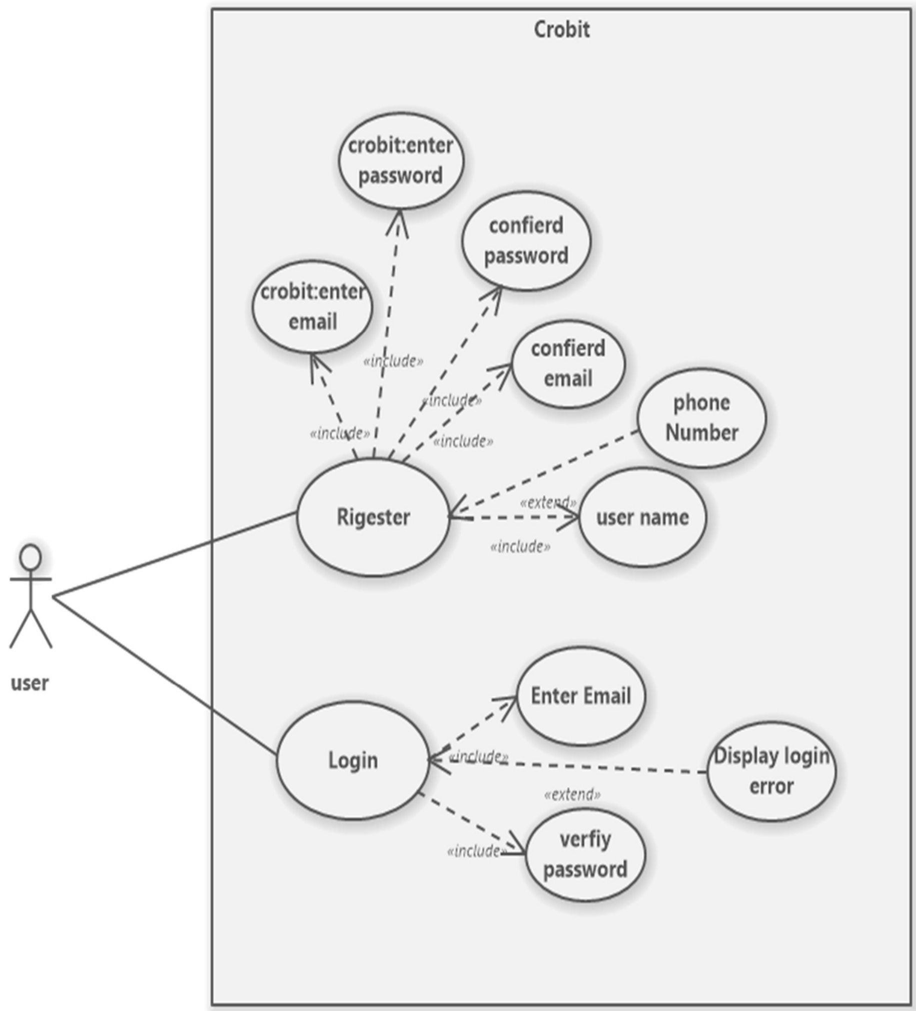
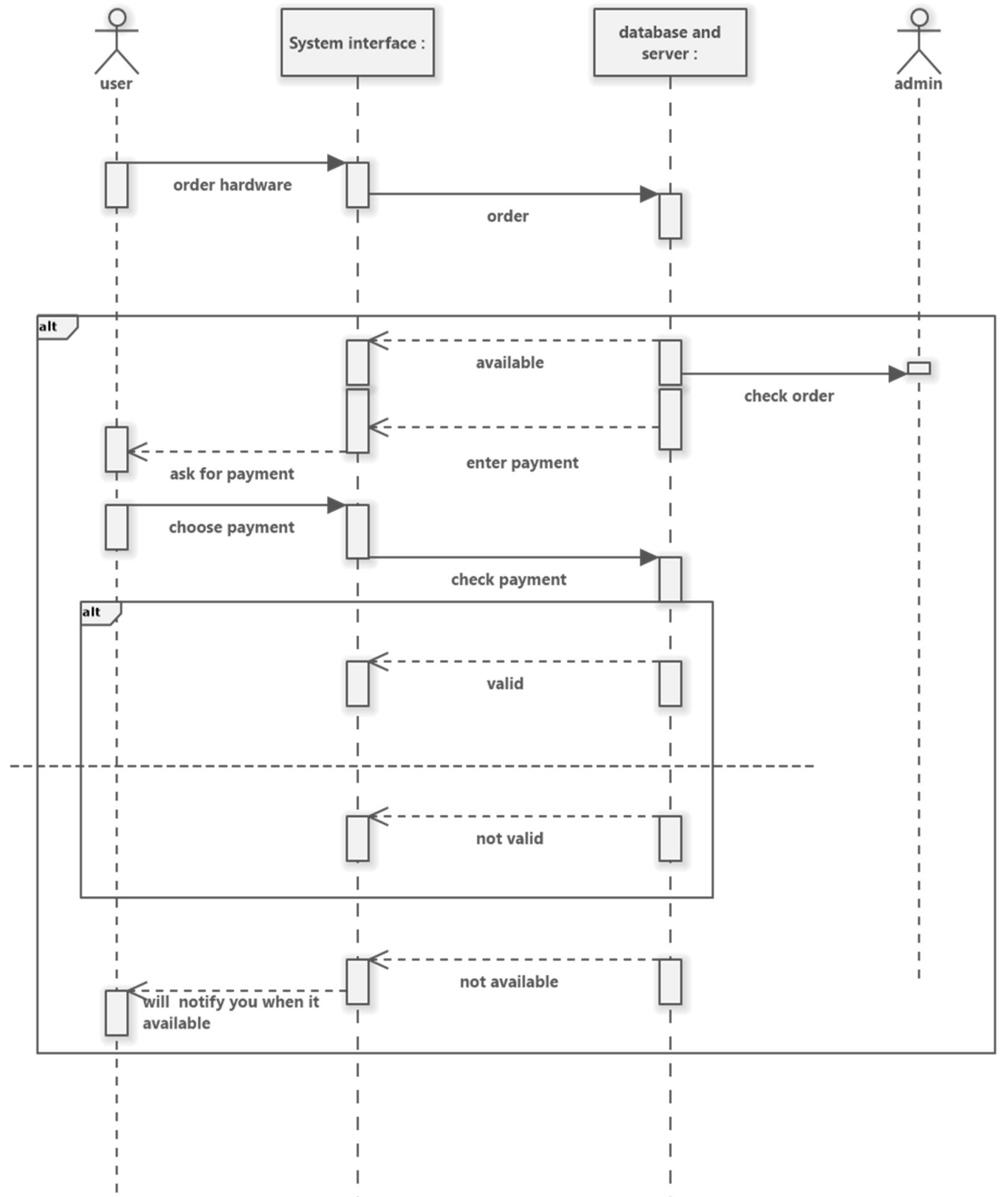
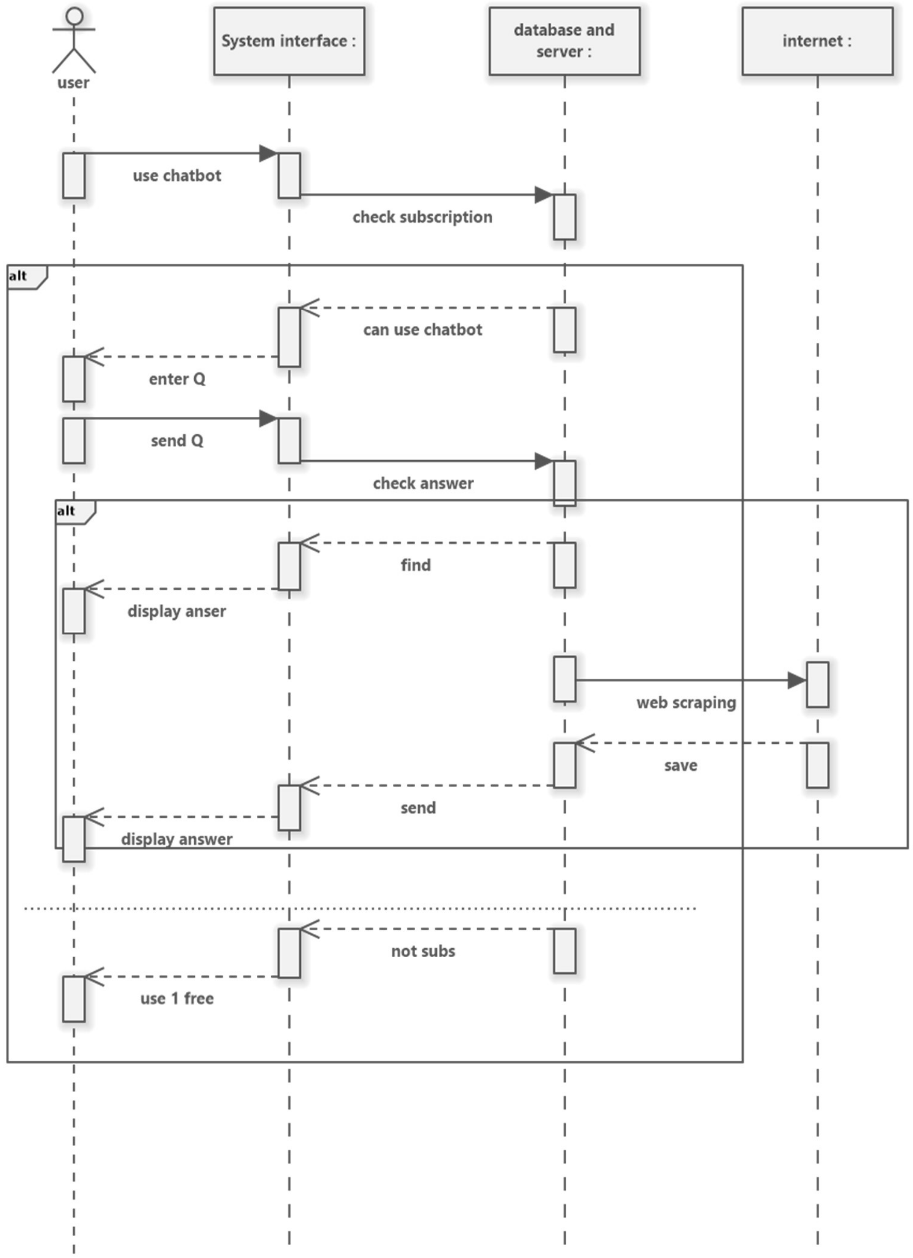
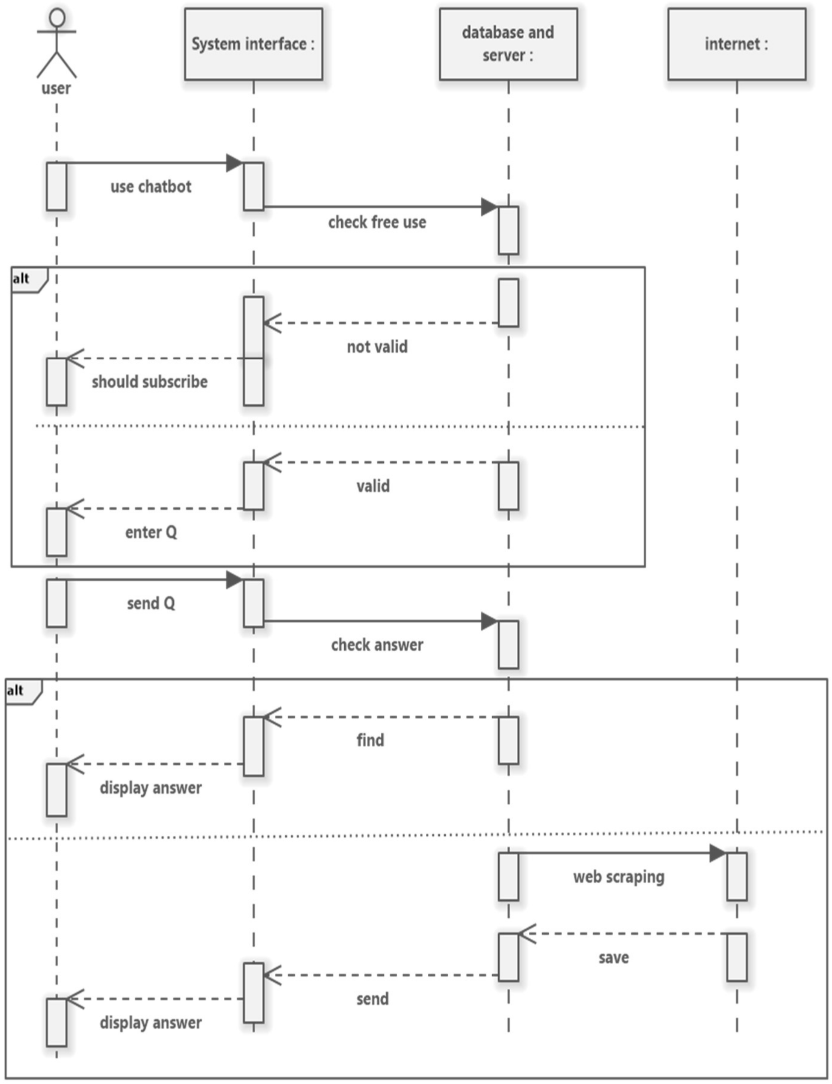


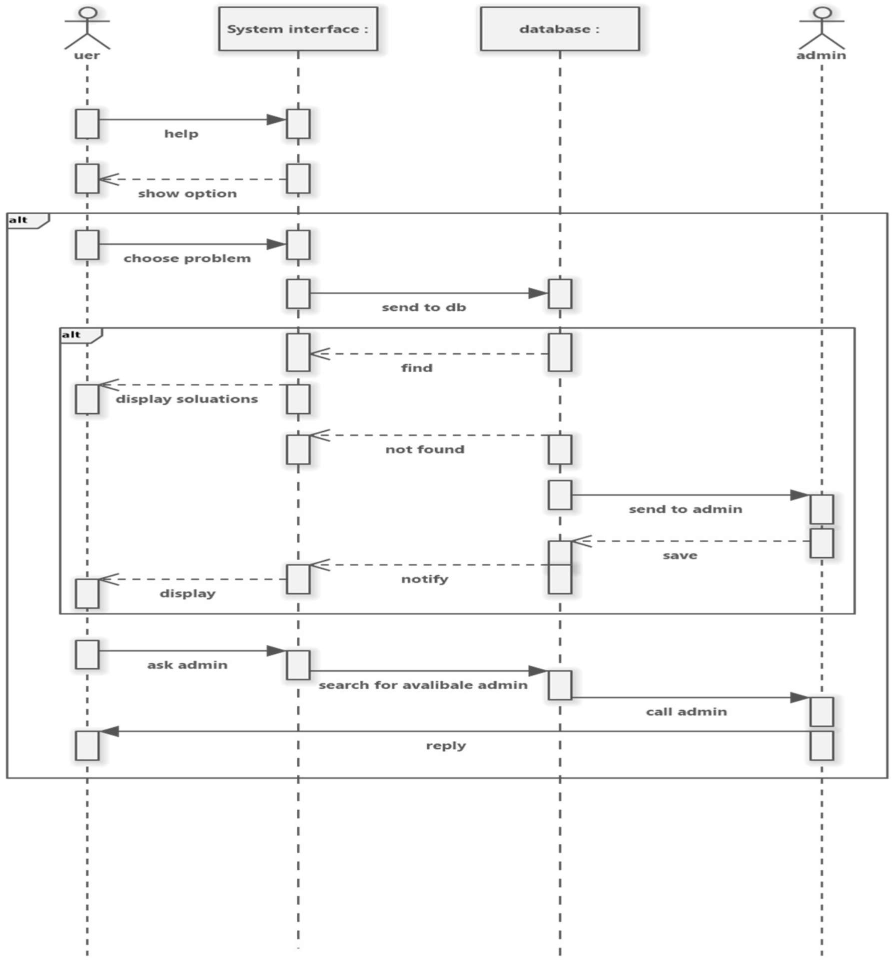
Figure 12. Use case diagrams

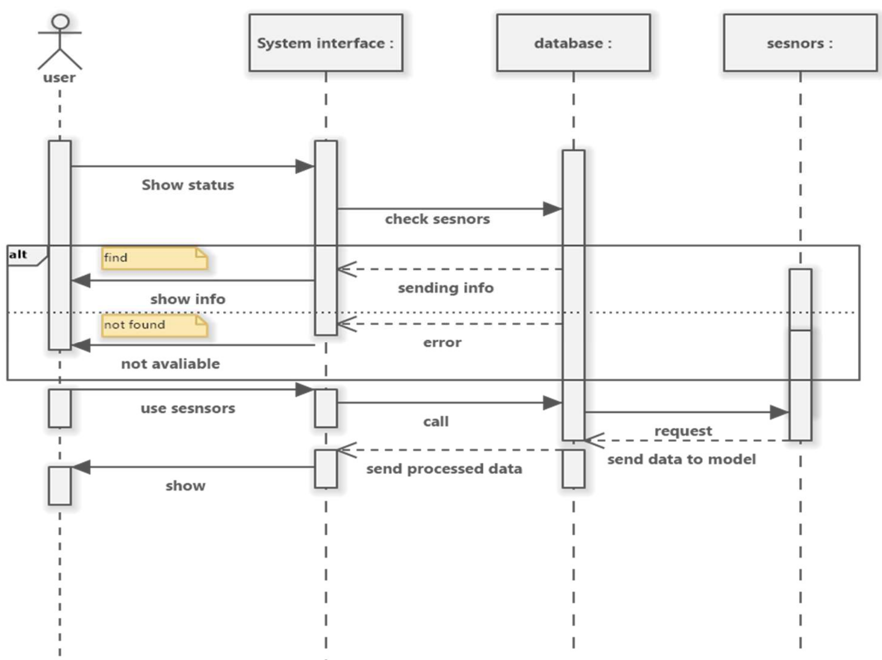
3.3.3 Sequence diagrams:

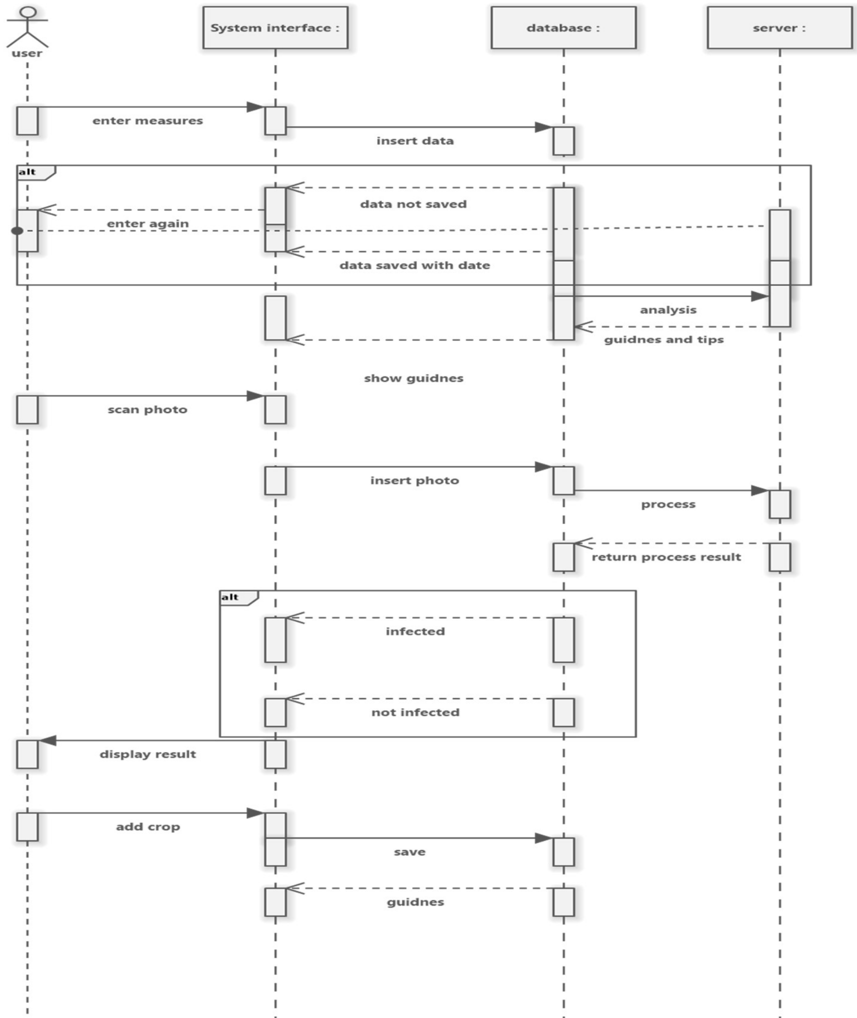



user
System interface :
database and server :
internet :
use chatbot
check subscription
alt
can use chatbot
enter Q
send Q
check answer
alt
find
display anser
web scraping
save
send
display answer
not subs
use 1 free

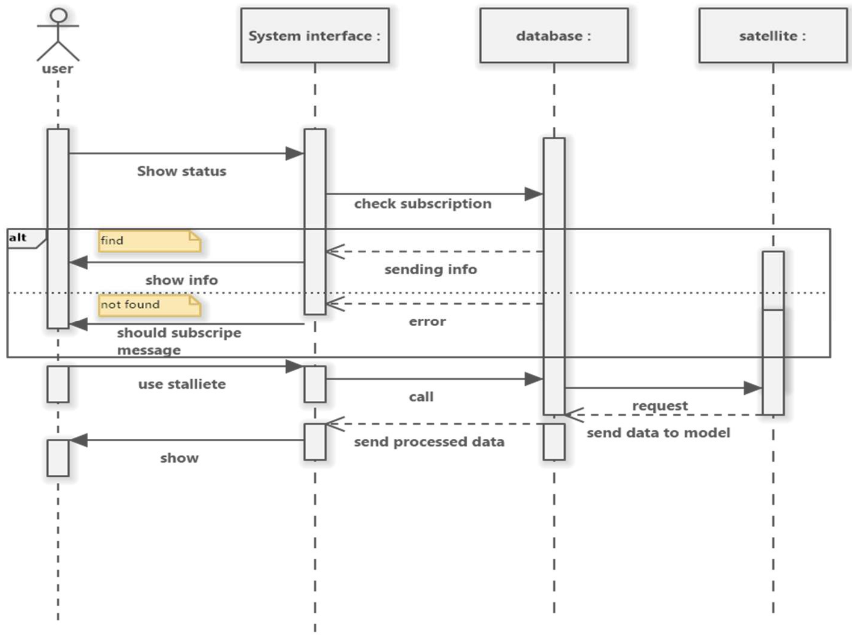

user
System interface :
database and server :
internet :
use chatbot
check free use
alt
not valid
should subscribe
valid
enter Q
send Q
check answer
alt
find
display answer
web scraping
save
send
display answer


uer
System interface :
database :
admin
help
show option
alt
choose problem
send to db
alt
find
display soluations
not found
send to admin
save
notify
display
ask admin
search for avalibale admin
call admin
reply


user
System interface :
database :
sesnors :
Show status
check sesnors
alt
find
sending info
show info
not found
error
not avaliable
use sesnsors
call
request
send data to model
send processed data
show



user
System interface :
database :
server :
enter measures
insert data
alt
data not saved
enter again
data saved with date
analysis
guidnes and tips
show guidnes
scan photo
insert photo
process
return process result
alt
infected
not infected
display result
add crop
save
guidnes


user
System interface :
database :
satellite :
Show status
check subscription
alt
find
sending info
show info
not found
error
should subscripe
message
use stalliete
call
request
send data to model
send processed data
show

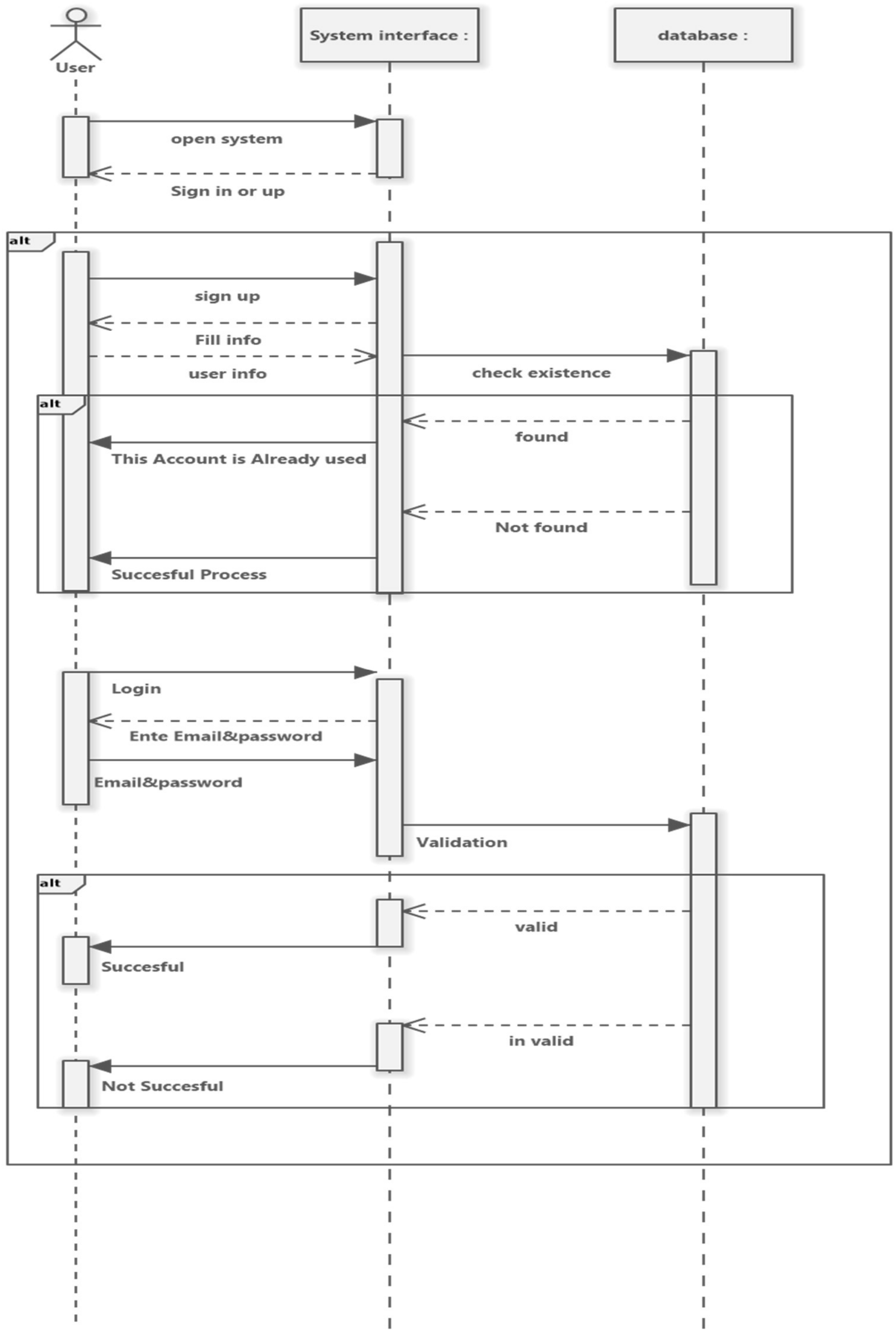


Figure 13. Sequence diagrams

3.3.4 Activity diagrams:

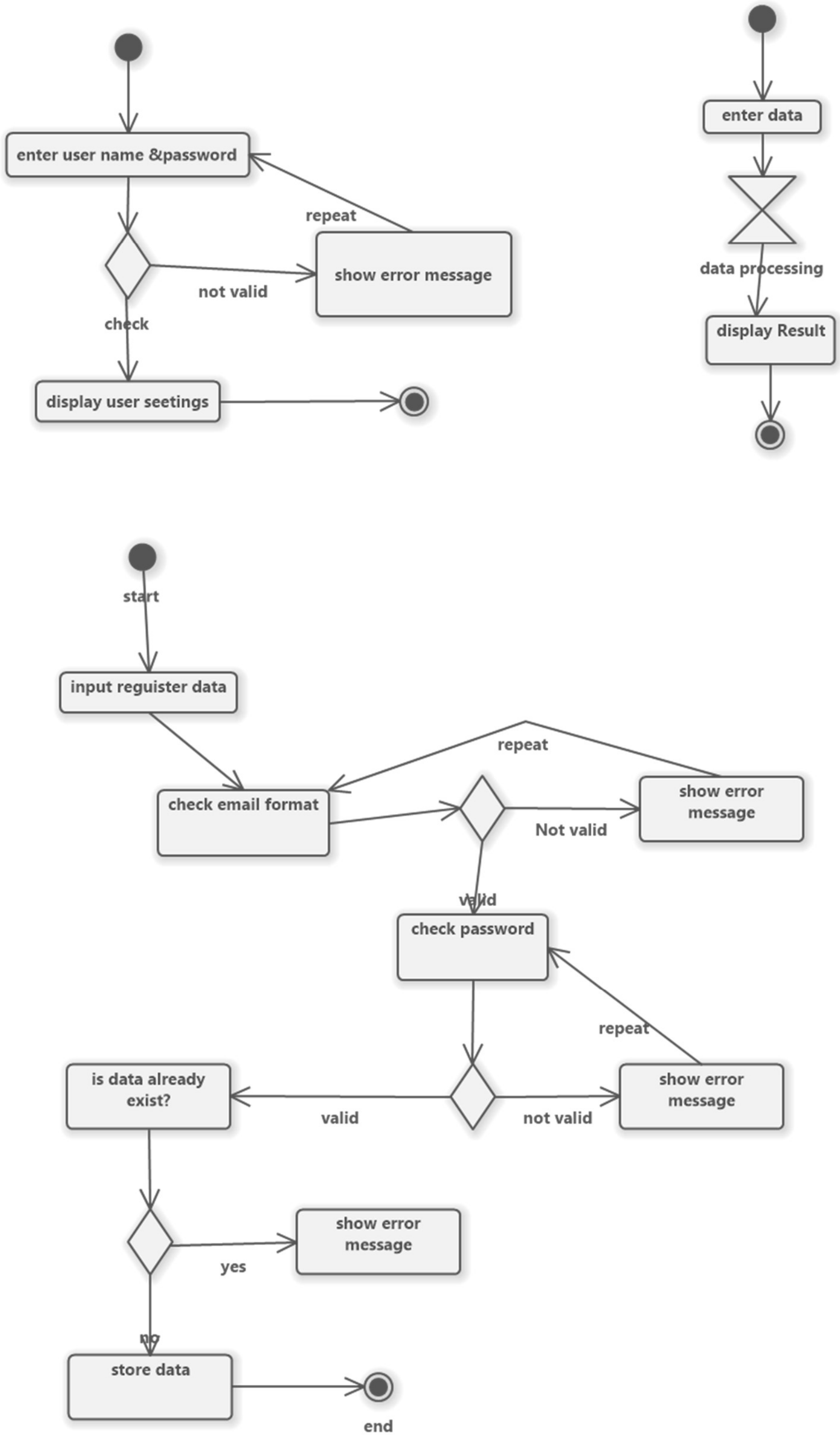
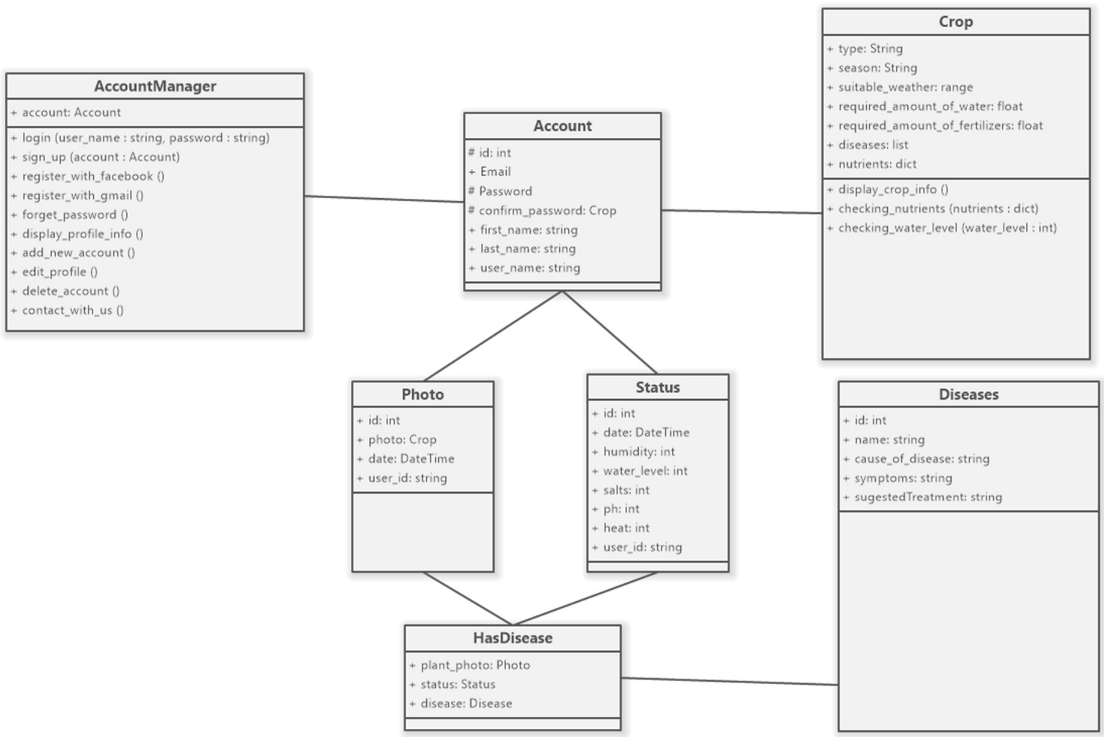


Figure 14. Activity diagrams

3.3.5 Class diagram:

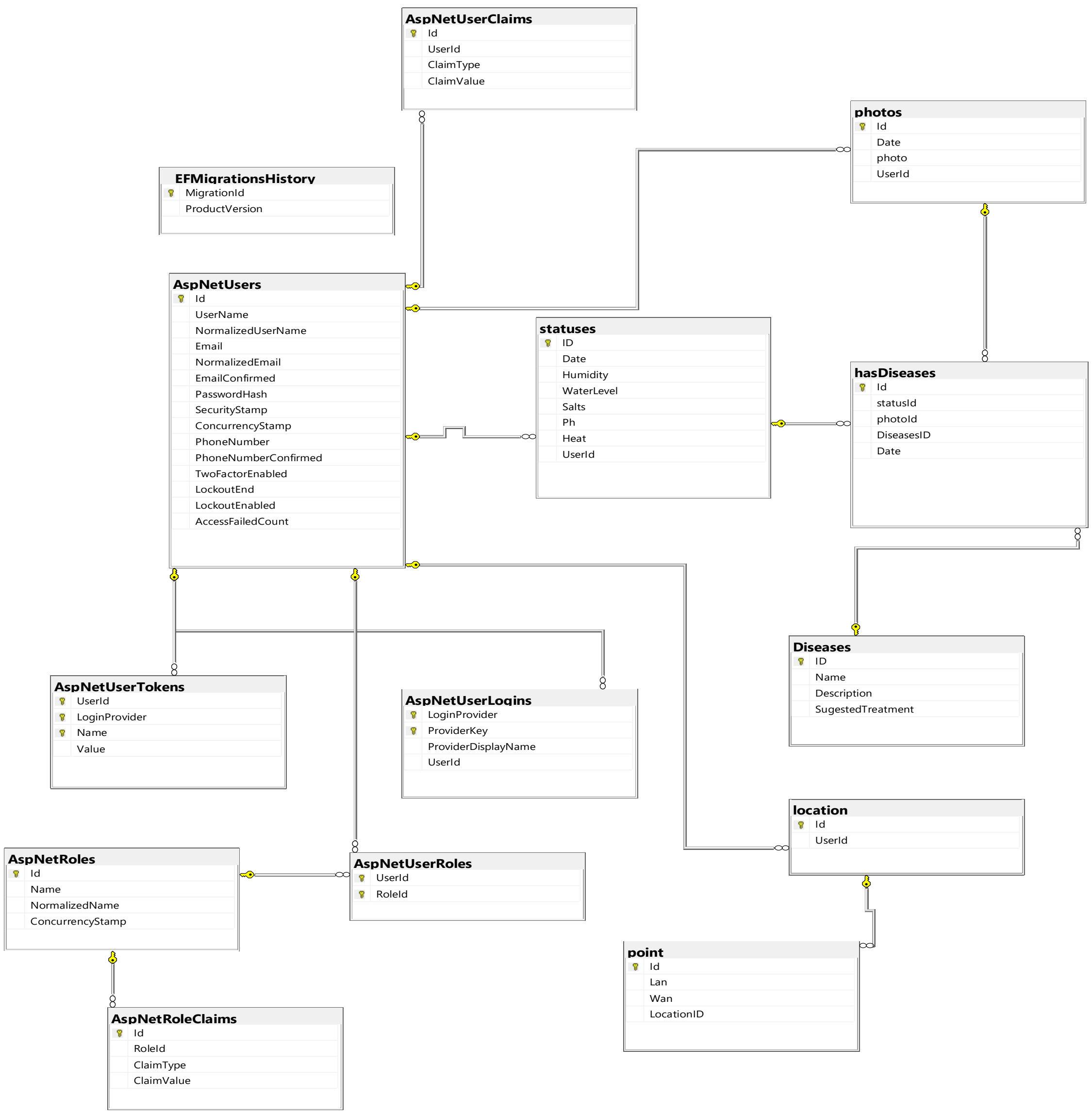
A class diagram describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects:



3.3.6 Database diagram:

- **Database:** is absolutely an integral part of software systems. To fully utilize ER Diagram in database engineering guarantees you to produce high-quality database design to use in database creation, management, and maintenance. An ER model also provides a means for communication.

- produce high-quality database design to use in database creation, management, and maintenance. An ER model also provides a means for communication.



3.3.7 SWOT Analysis:

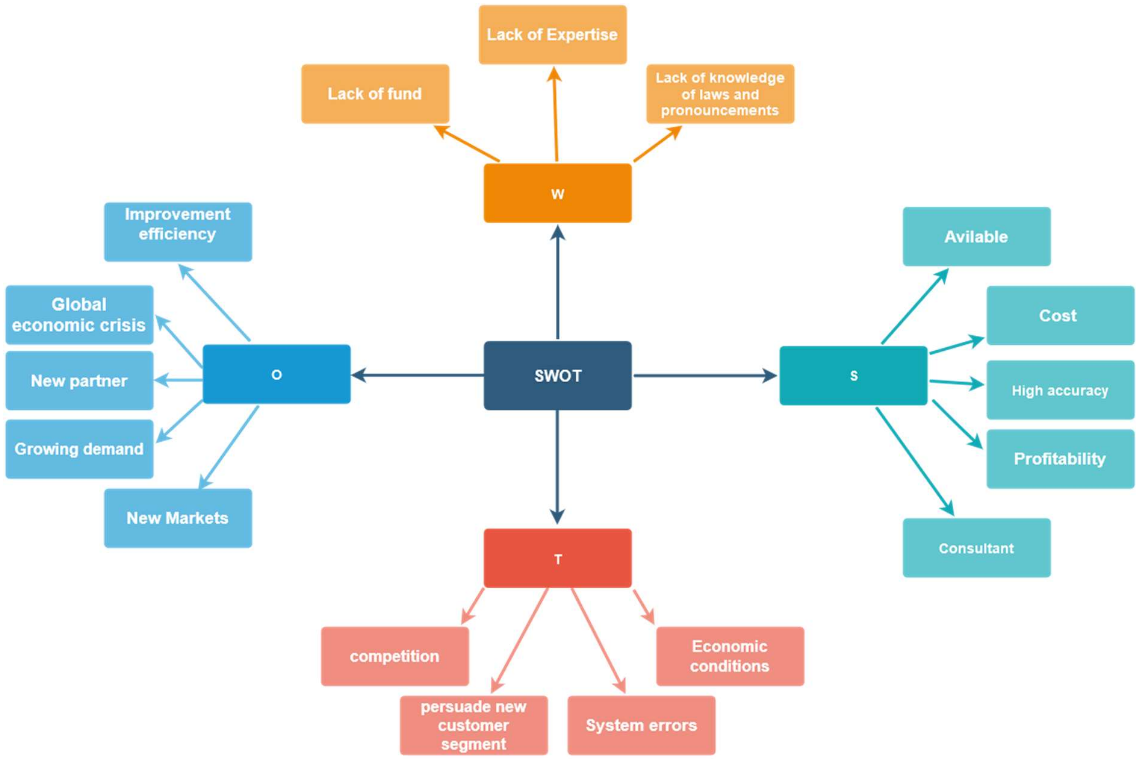


Figure 15.SWOT Analysis

3.3.7.1 Strength

- **Available:** the system is available for different users and have some free features to Agriculture hobbyist.
- **Cost:** The project offers a different price depending on the period of use of the satellite.
- **Advanced technology:** The project will use cutting-edge artificial intelligence, IoT, and satellite imaging technologies, which can give it a competitive advantage in the market.
- **High accuracy:** The system has good AI models to score high for analyzing real data and detecting different problems facing plants.
- **Consultant:** The project contains a smart chatbot that can guide the user.
- **Profitability:** the system will help user to increasing his profit and we will deal with him.

3.3.7.2 Weakness

- **lack of fund:** that there is a current or projected deficiency in the funds to maintain current or projected levels of staffing and operations.
- **Lack of expertise:** The project team does not have experience with all of Agriculture's knowledge, which can lead to challenges during implementation.
- **Lack of knowledge** of laws and pronouncements

3.3.7.3 Opportunities

- **Improvement efficiency:** The **project could** significantly improve the efficiency of agricultural operations, which could lead to cost savings and increased profits for customers.
- **Global economic crisis**
- **New partner:** The project can bring in new partners who are excited to take on a new challenge in our country.
- **New markets:** The project could open new markets, such as developing countries where the use of advanced technologies in agriculture is limited.
- **Growing demand:** The use of artificial intelligence, IoT, and satellite imaging in agriculture is increasing, which could create opportunities for the project to expand and grow.

3.3.7.4 Threats

- **Persuade new customer segment:** Failure to convince one of the target groups.
- **Competition:** The project faces competition from other companies offering similar solutions.
- **System errors:** There may be regulatory barriers or hurdles that could impact the project, such as data privacy or cybersecurity concerns.
- **Economic conditions:** Changes in the economic environment, such as market shifts or changes in commodity prices, could impact the demand for the project's solutions

3.4 Business Model Canvas:

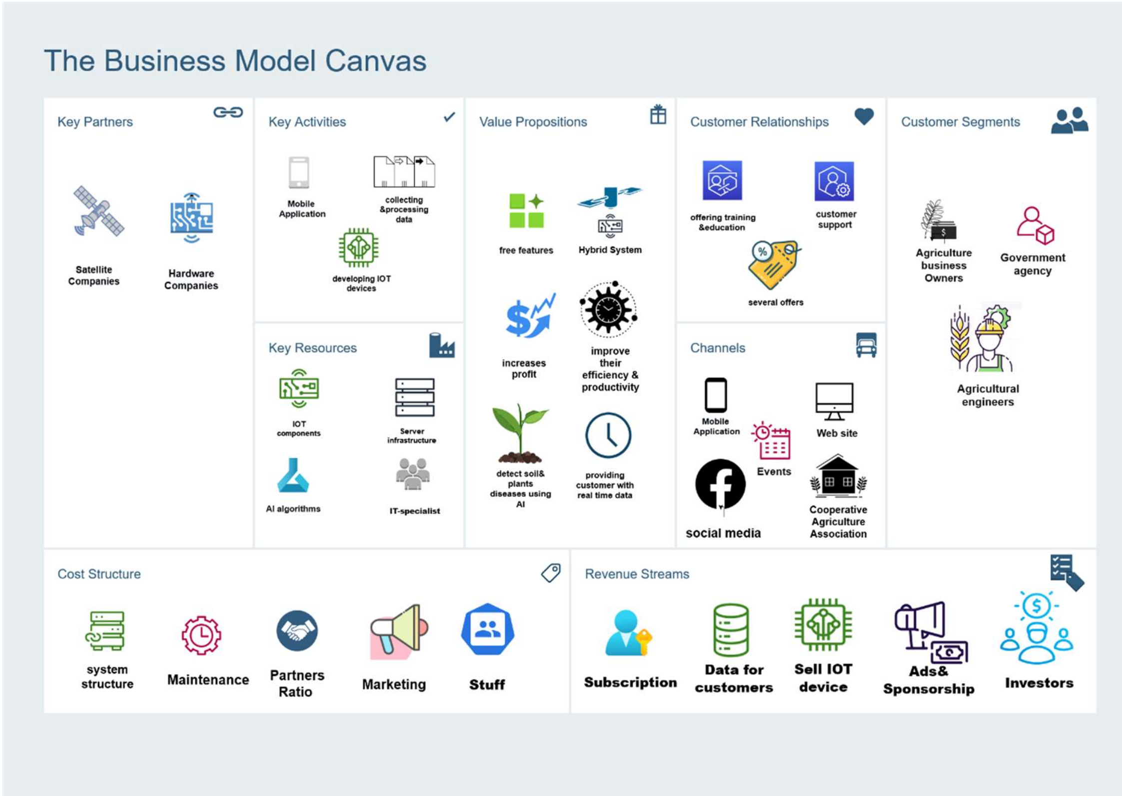


Figure 16. Business Model

3.5 Tools and Languages:

- **SoftwareIdeasModeler:** It is used to draw the UML diagrams.
- ***Blender****: It is used to design hardware products*
- ***Figma:*** *It is used to design user interface with user experience and prototypes.*
- ***Python:*** *Python is among the most widely used programming languages that developers use in the present. it is a key part of AI programming languages due to the fact that it has good frameworks.*
- ***Keras :*** *is the most used deep learning framework among top-5 winning teams on Kaggle. Because Keras makes it easier to run new experiments*
- ***TensorFlow:*** *makes it easy for beginners and experts to create machine learning models for desktop, mobile, web, and cloud. See the sections below to get started.*
- ***OpenCV:*** *is an open-source computer vision and machine learning software library. it was built to provide a common infrastructure for computer vision applications.*

- ***ASP.NET Core:*** *is a cross-platform, high-performance, open-source framework for building modern, cloud-enabled, Internet-connected apps. With ASP.NET Core, you can Build web apps and services, Internet of Things (IoT) apps, and mobile backends*
- ***Jupyter Notebook*** *:is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience.*
- ***Flutter:*** *is an open-source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase.*
- ***Dart:*** *is a client-optimized language for fast apps on any platform.*
- ***Arduino:*** *is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects.*
- ***Adobe Illustrator:*** *is a software application for creating drawings, illustrations, and artwork using a Windows or MacOS computer.*
- ***Visual Studio 2022:*** *is a complete and ideal integrated development environment (IDE) for creating large, complex, and scalable applications. It is one of the most complete tools available for development, especially with Microsoft.*
- ***Microsoft SQL Server:*** *is a relational database management system (RDBMS) that supports a wide variety of transaction processing, business intelligence and analytics applications in corporate IT environments.*
- ***Entity Framework Core:*** *is a modern object-database mapper for .NET. It supports LINQ queries, change tracking, updates, and schema migrations. EF Core works with many databases, including SQL Database (on-premises and Azure), SQLite, MySQL, PostgreSQL, and Azure Cosmos DB*

3.7 Summary:

In this chapter we provide the reader with detailed knowledge about our system. Section 3.1 provides system requirements Which is divided into functional and non-functional requirements, and user requirements which specify some different specifications for users, Section 3.2 includes system architecture which describe its major components, their relationships, and how they interact with each other, Section 3.3 provides development methodology which includes UML diagrams that shows the details of how the system will function. At the end of the chapter, we have included a SWOT analysis, BMC and tools to build the system.

Chapter 4

System Design

In this chapter we start to convert the analysis system to actual User interface Screens according to User experience that makes it easy, efficient, and enjoyable for users to interact with our product.

4. Software Application Design

Software application design is the method of planning and creating a software solution to address specific business needs regarding our app “agriculture improvement”. Our app aims to make farming easier. At first there are onboarding pages that help the user to understand the purpose of Crobit app before using it.

The Cycle of Crobit app to benefit from our services is: First, user should sign-up or log-in to be able to using Crobit App. Second, User should determine the location and select the field then he will be able to uses our features also he can select multiple fields and enter or knowing information about these fields.

After logging in, there is a home page that contains basic features “Diagnose diseases – Soil status – Satellite monitoring – Consultant – Scan crop” that user can choose from these, user should add crop before using one of these features and when choosing a crop, useful information about this crop will appear. When choosing “Diagnose crop” the diseases, weeds & insects in the crop & field will be shown and if there is no disease a pop message will appear.

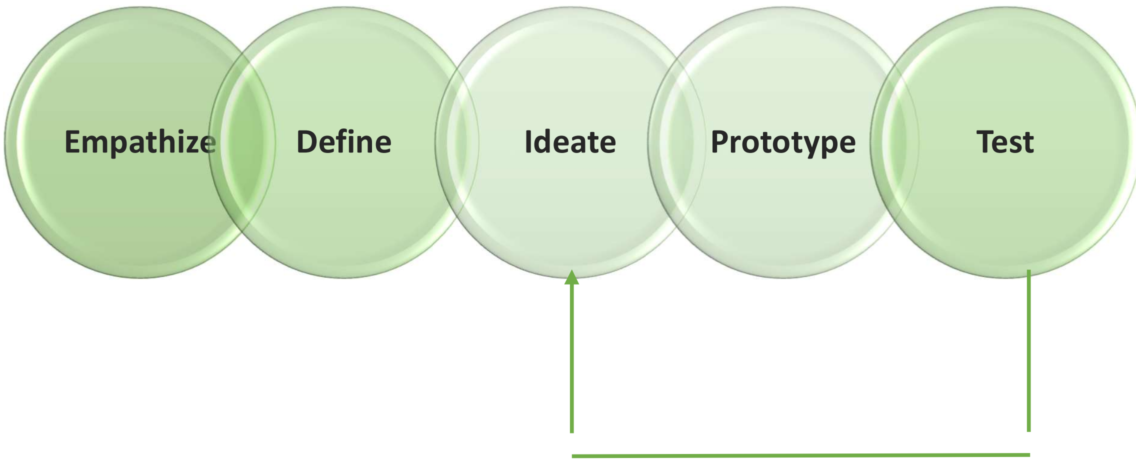
When choosing “Soil status”, this page contains two types of information weekly information about soil nutrients “N-P-K” and daily information about “Humidity -Salts- Water level- Heat – Ph.”. When choosing “Satellite monitoring”, this page Contain three classes (NDVI, NDMI, SAVI). When choosing “Consultant” there is Chabot between user and Ai consultant and user can inquire about any questions. When choosing “Scan crop”, user will be able scan his crop from any disease using his camera.

There are also some pages that help the user in dealing with the application in general like “forget his password and link the app with an email to make new password to confirm security, edit his profile, edit his fields information, knowing privacy policy, help center, logout and finally there's a notification page that notify user when there's something new in his plant".

4.1 UX in system design

UX stands for User Experience that's the subjective experience that a user has with a mobile application. This includes both good and bad experience, as well as emotions, also it's the process of creating a product that's delightful to interact with and provides a meaningful and relevant experience. We inspire the green color for the application based on the color of the plant and inspire our logo "Crobit" based on two words with important meaning in our project "CROP" that's indicates the plant and "ORBIT" that's indicates the satellite orbit that's used in the project.

4.2 Design process



For our project, we adopted a simple process that begins with understanding our potential user's pain points, defining the specific problems we wanted to solve, brainstorming to come up with solutions to these problems, converting these ideas to interactive prototypes then testing our design. As always, there was room for extra ideation after testing.

4.3 Visual Concept

Color plays a vitally important role in Graphical User Interface design. Color helps the users see and interpret the content of the mobile app, interact with the correct elements, and understand actions. Every mobile app's design has a color scheme. Using the right color is not only good for aesthetic purposes but also it can help users navigate and use mobile apps more easily. The choosing of color combinations that complement each other will make mobile apps more appealing to the user and it will be easier to use. The application of color in these mobile apps is dominated by green color with code #4BA26A R: 75 G: 162, B: 106. Green symbolizes balance, refreshment, environmental awareness, and peace.

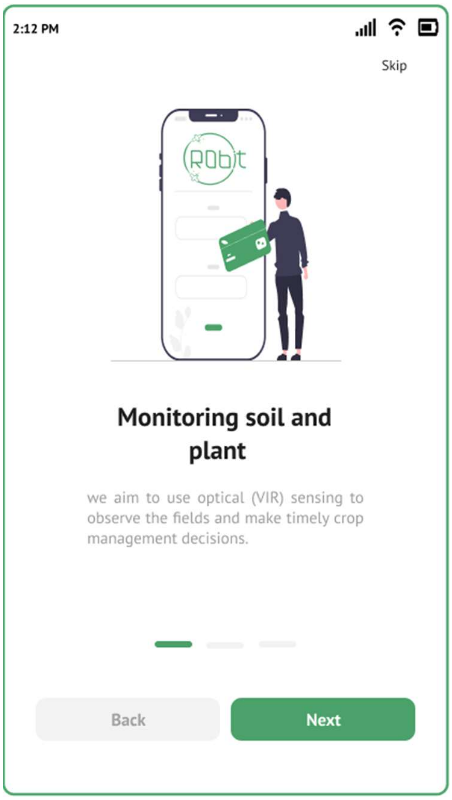
4.4 Typeface

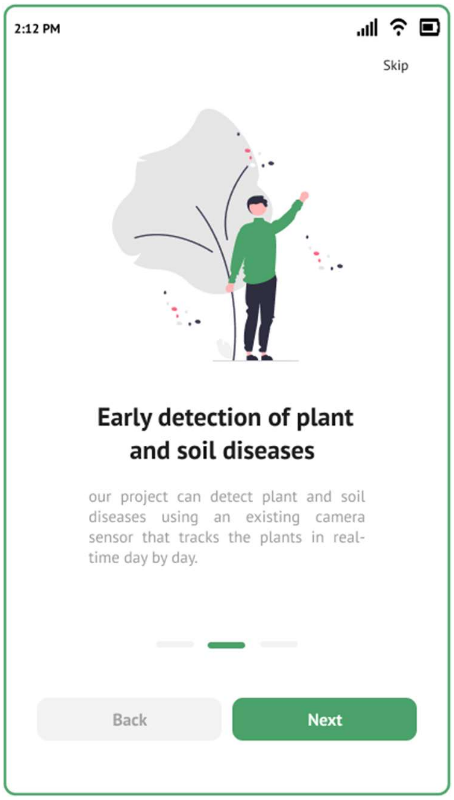
Mobile design requires high details and elaboration. Although images and videos are dynamic and colorful, the users still need to gain information throughout the text. It means the application of typography. The font is one of the most important elements of user interface design. Different type of content requires different font because font can express feelings and emotions. Mobile is the dominant screen today so the user interface should ideally design with responsive typography in mind. It means the typography has a geometric and scalable outline. Another important aspect to improve the readability is the spacing between the fonts. PT Sans is a type of family of universal use. It consists of 8 styles: regular and bold weights with corresponding italics form a standard computer font family; two narrow styles (regular and bold) are intended for documents that require tight set; two caption styles (regular and bold) are for texts of small point sizes. These features beside conventional use in business applications and printed stuff made the fonts quite useable for direction and guide signs, schemes, screens of information kiosks and other objects of urban visual communications.

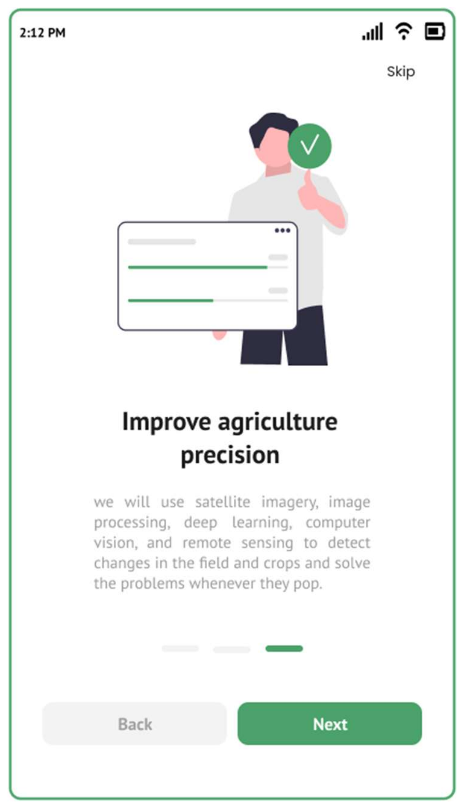
4.5 UI in system design

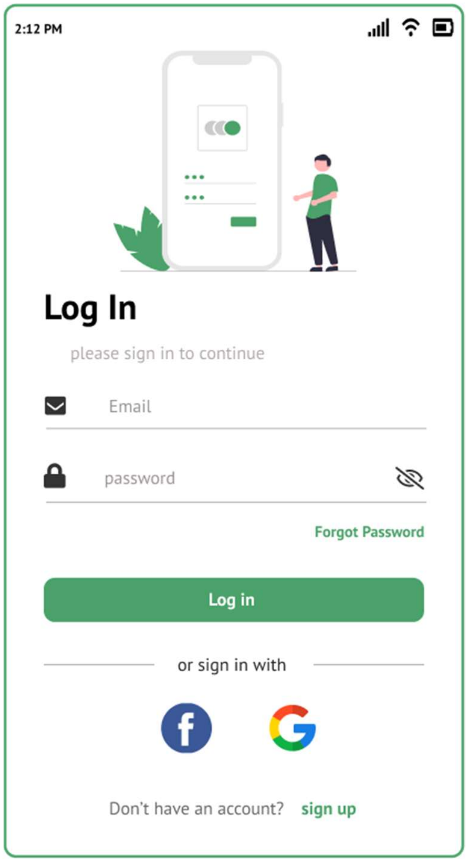
UI stands for User Interface that it's a space where interaction between user and device's application, features, content, and functions.

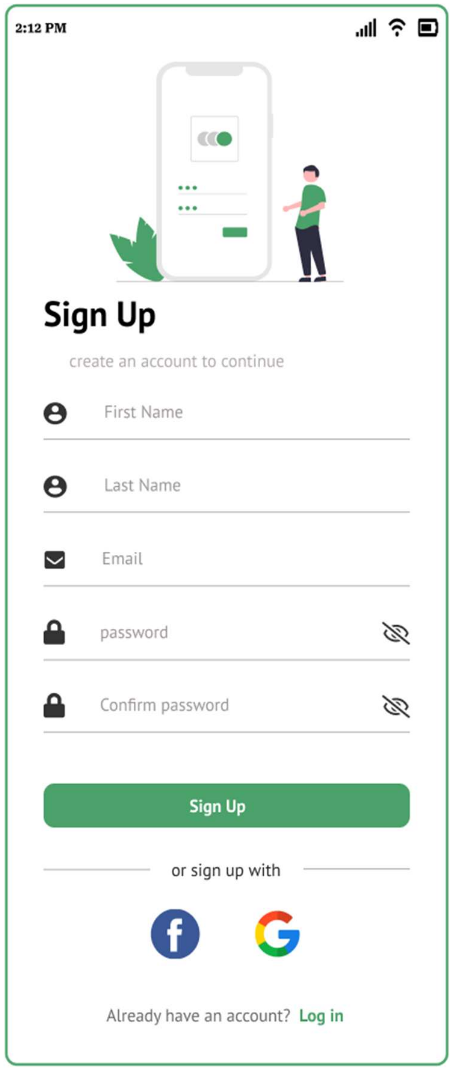
4.5.1 Our UI design (Application)

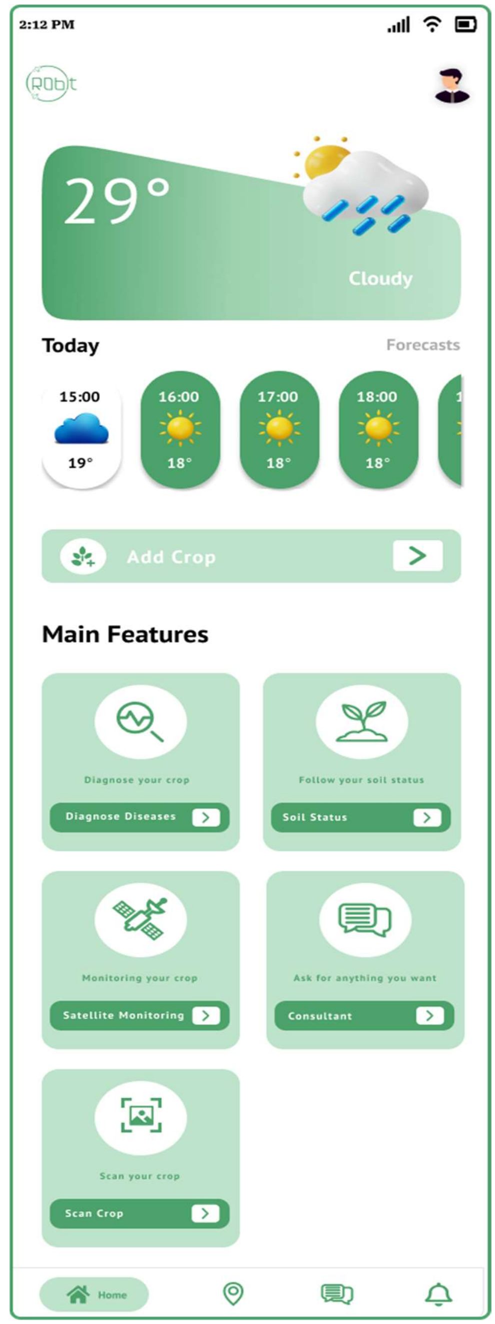

2:12 PM
Skip
RObit
Monitoring soil and plant
we aim to use optical (VIR) sensing to observe the fields and make timely crop management decisions.
Back
Next

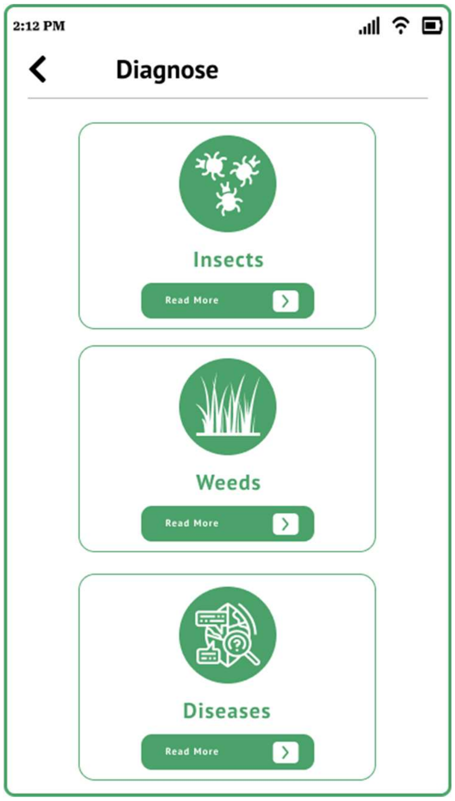

2:12 PM
Skip
Early detection of plant and soil diseases
our project can detect plant and soil diseases using an existing camera sensor that tracks the plants in real-time day by day.
Back
Next

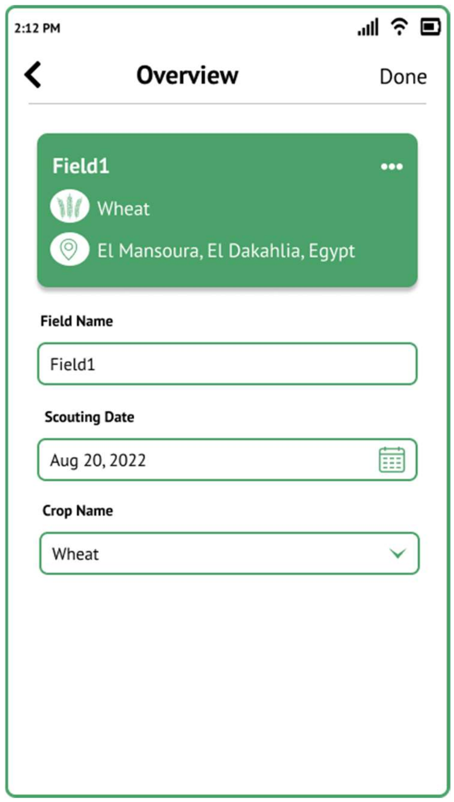

2:12 PM
Skip
Improve agriculture precision
we will use satellite imagery, image processing, deep learning, computer vision, and remote sensing to detect changes in the field and crops and solve the problems whenever they pop.
Back
Next


2:12 PM
Log In
please sign in to continue
Email
password
Forgot Password
Log in
or sign in with
Don't have an account? sign up

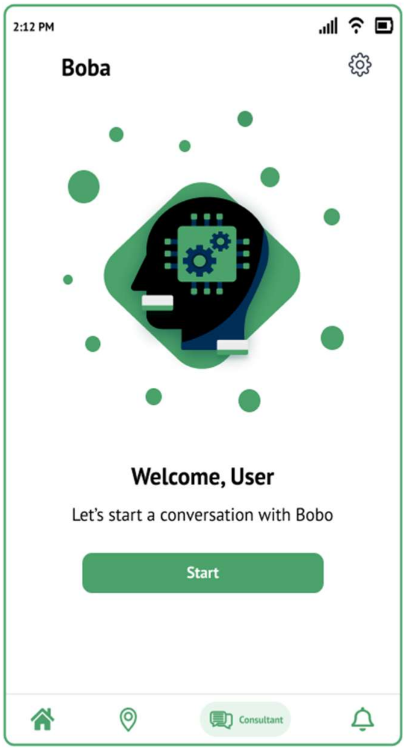

2:12 PM
Sign Up
create an account to continue
First Name
Last Name
Email
password
Confirm password
Sign Up
or sign up with
Already have an account? Log in

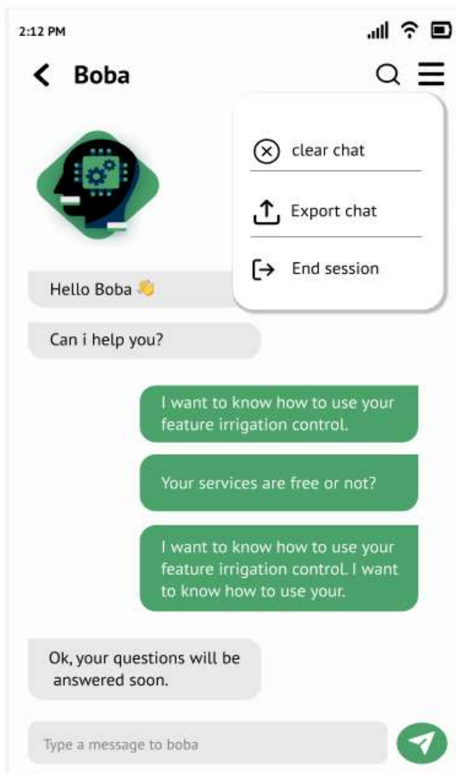

2:12 PM
29°
Cloudy
Today
Forecasts
15:00
19°
16:00
18°
17:00
18°
18:00
18°
Add Crop
Main Features
Diagnose your crop
Diagnose Diseases
Follow your soil status
Soil Status
Monitoring your crop
Satellite Monitoring
Ask for anything you want
Consultant
Scan your crop
Scan Crop
Home

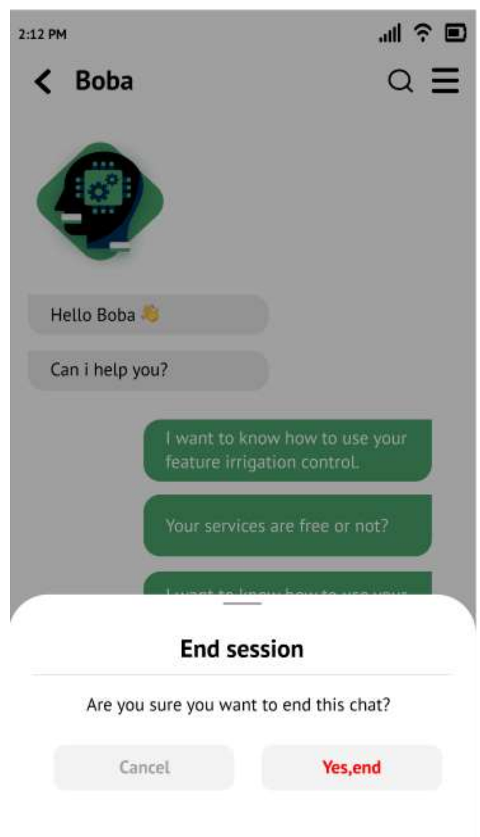

2:12 PM
Diagnose
Insects
Read More
Weeds
Read More
Diseases
Read More

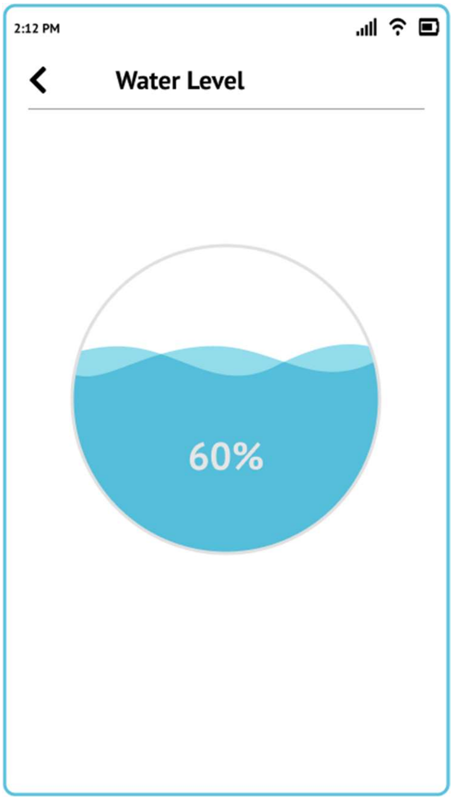

2:12 PM
Overview
Done
Field1
Wheat
El Mansoura, El Dakahlia, Egypt
Field Name
Field1
Scouting Date
Aug 20, 2022
Crop Name
Wheat

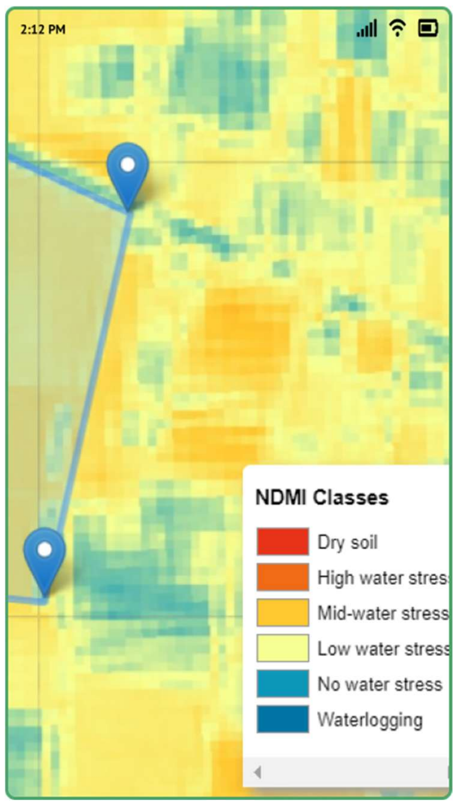

2:12 PM
Location
Field1
Add

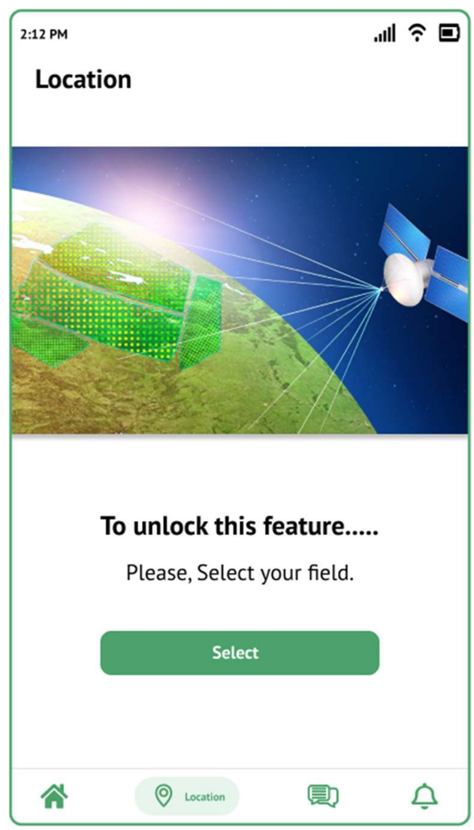

2:12 PM
Boba
Welcome, User
Let's start a conversation with Bobo
Start
Consultant

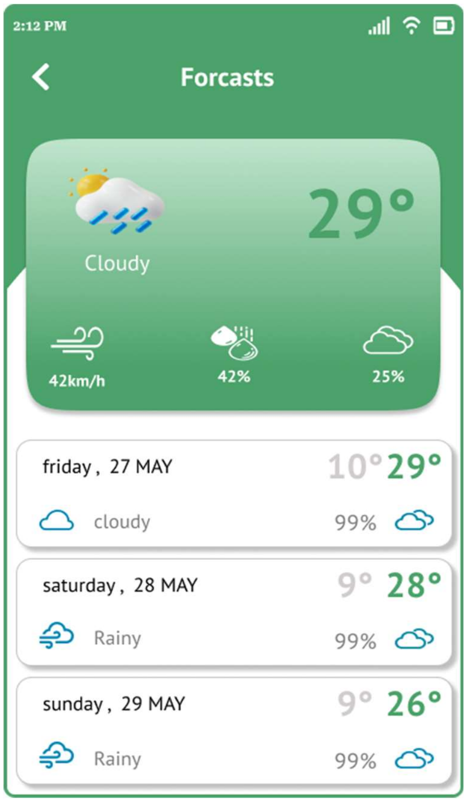

2:12 PM
Boba
clear chat
Export chat
End session
Hello Boba
Can i help you?
I want to know how to use your feature irrigation control.
Your services are free or not?
I want to know how to use your feature irrigation control. I want to know how to use your.
Ok, your questions will be answered soon.
Type a message to boba

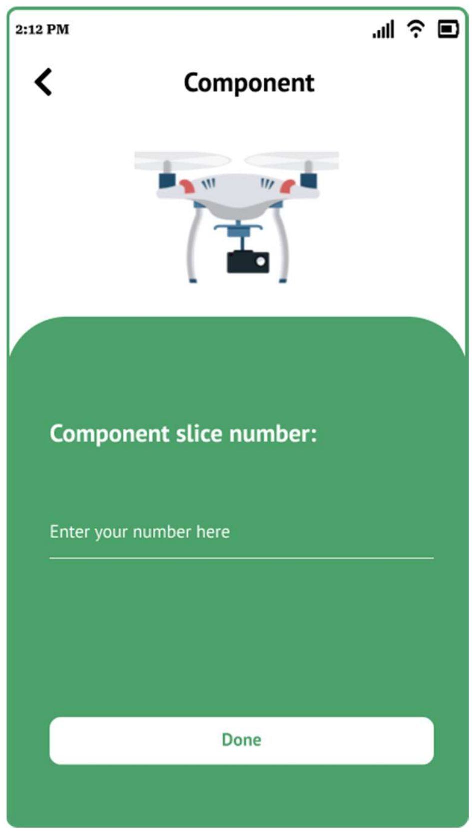

2:12 PM
Boba
Hello Boba
Can i help you?
I want to know how to use your feature irrigation control.
Your services are free or not?
End session
Are you sure you want to end this chat?
Cancel
Yes,end

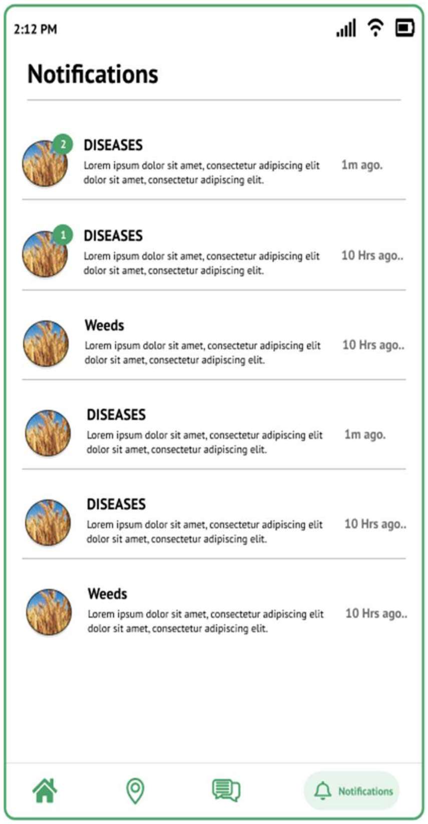

2:12 PM
Water Level
60%

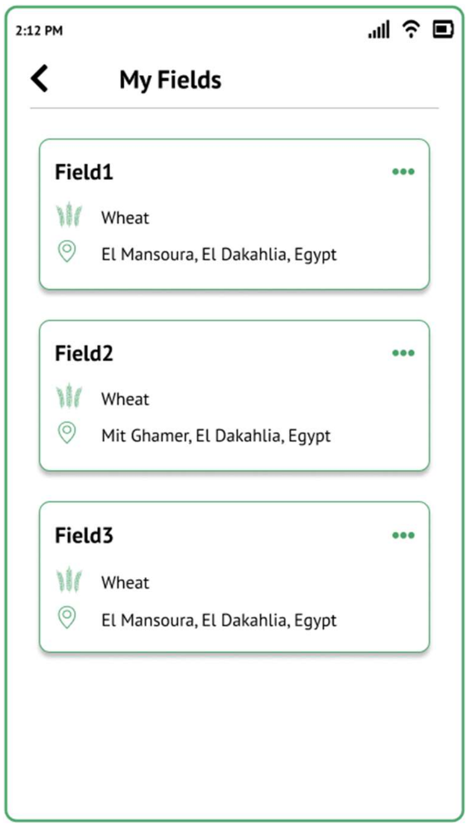

2:12 PM
NDMI Classes
Dry soil
High water stres
Mid-water stress
Low water stress
No water stress
Waterlogging

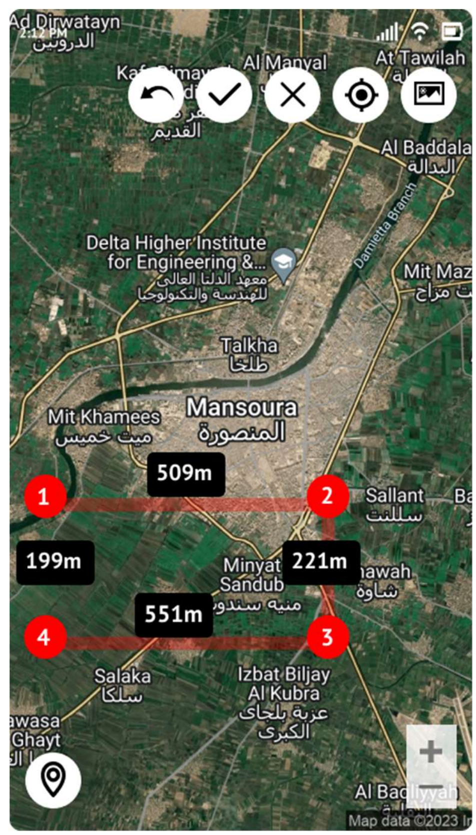

2:12 PM
Location
To unlock this feature.....
Please, Select your field.
Select
Location

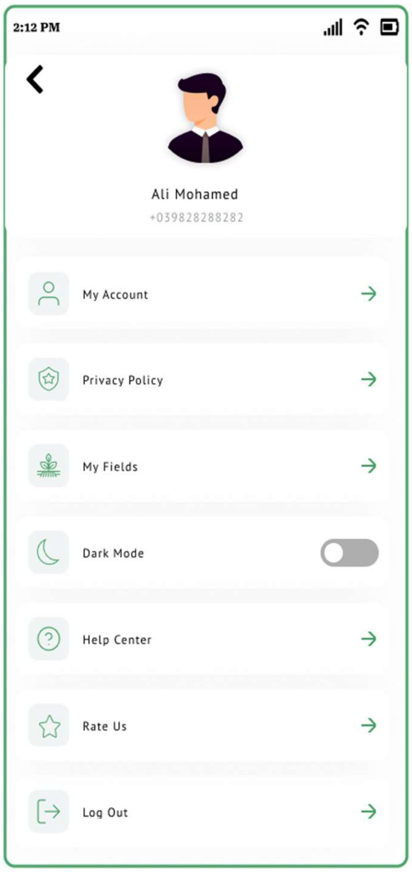

2:12 PM
Forcasts
29°
Cloudy
42km/h
42%
25%
friday , 27 MAY
10° 29°
cloudy
99%
saturday , 28 MAY
9° 28°
Rainy
99%
sunday , 29 MAY
9° 26°
Rainy
99%

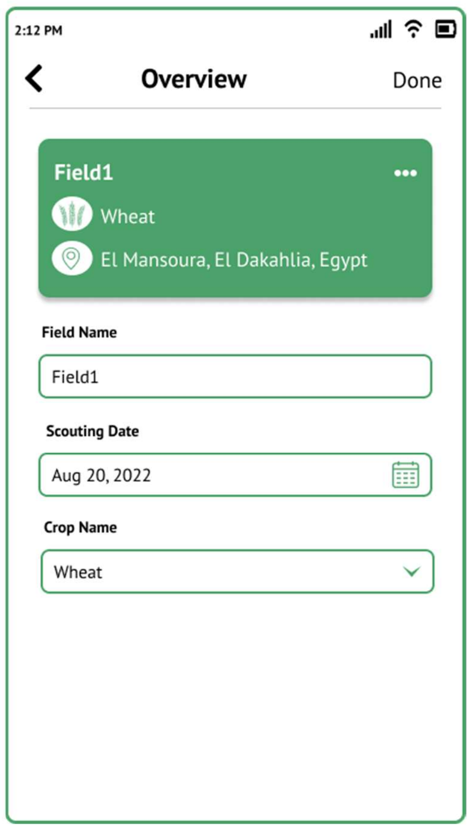

2:12 PM
Component
Component slice number:
Enter your number here
Done

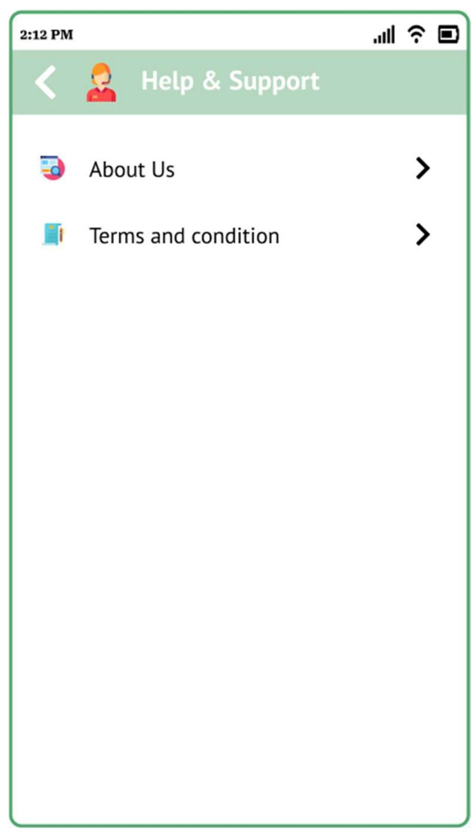

2:12 PM
Notifications
DISEASES
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
1m ago.
DISEASES
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
10 Hrs ago..
Weeds
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
10 Hrs ago..
DISEASES
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
1m ago.
DISEASES
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
10 Hrs ago..
Weeds
Lorem ipsum dolor sit amet, consectetur adipiscing elit dolor sit amet, consectetur adipiscing elit.
10 Hrs ago..
Notifications

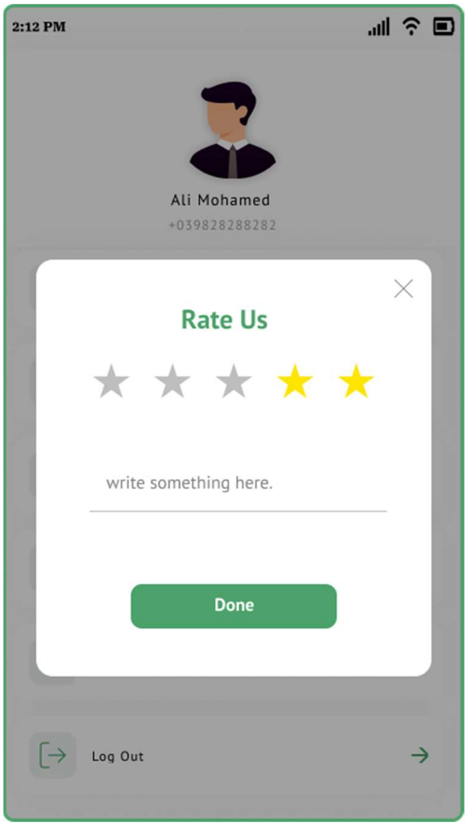

2:12 PM
My Fields
Field1
Wheat
El Mansoura, El Dakahlia, Egypt
Field2
Wheat
Mit Ghamer, El Dakahlia, Egypt
Field3
Wheat
El Mansoura, El Dakahlia, Egypt

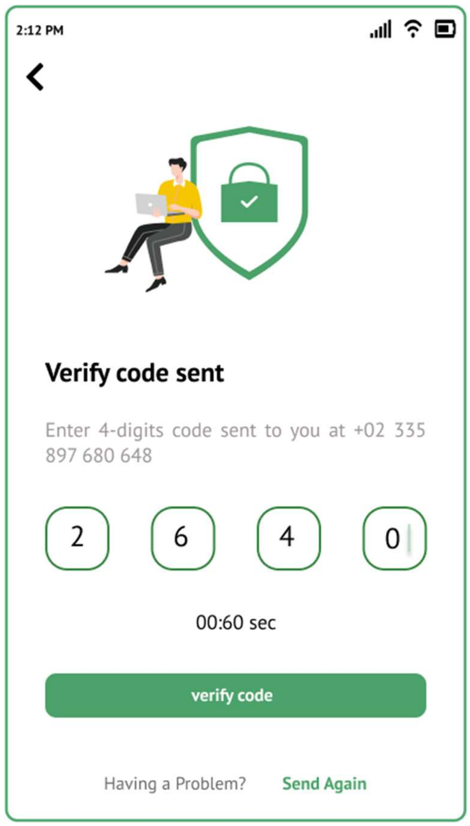

Ad Dirwatayn
Al Manyal
At Tawilah
Al Baddala
Delta Higher Institute for Engineering &...
Talkha
Mansoura
المنصورة
Mit Khamees
509m
199m
221m
551m
1
2
3
4
Sallant
Minyat Sandub
Salaka
Izbat Biljay Al Kubra
Map data ©2023

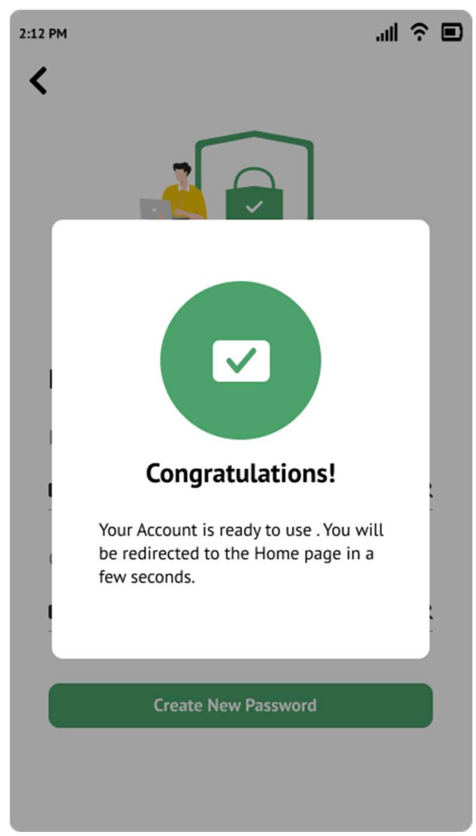

2:12 PM
Ali Mohamed
+039828288282
My Account
Privacy Policy
My Fields
Dark Mode
Help Center
Rate Us
Log Out


2:12 PM
Overview
Done
Field1
Wheat
El Mansoura, El Dakahlia, Egypt
Field Name
Field1
Scouting Date
Aug 20, 2022
Crop Name
Wheat

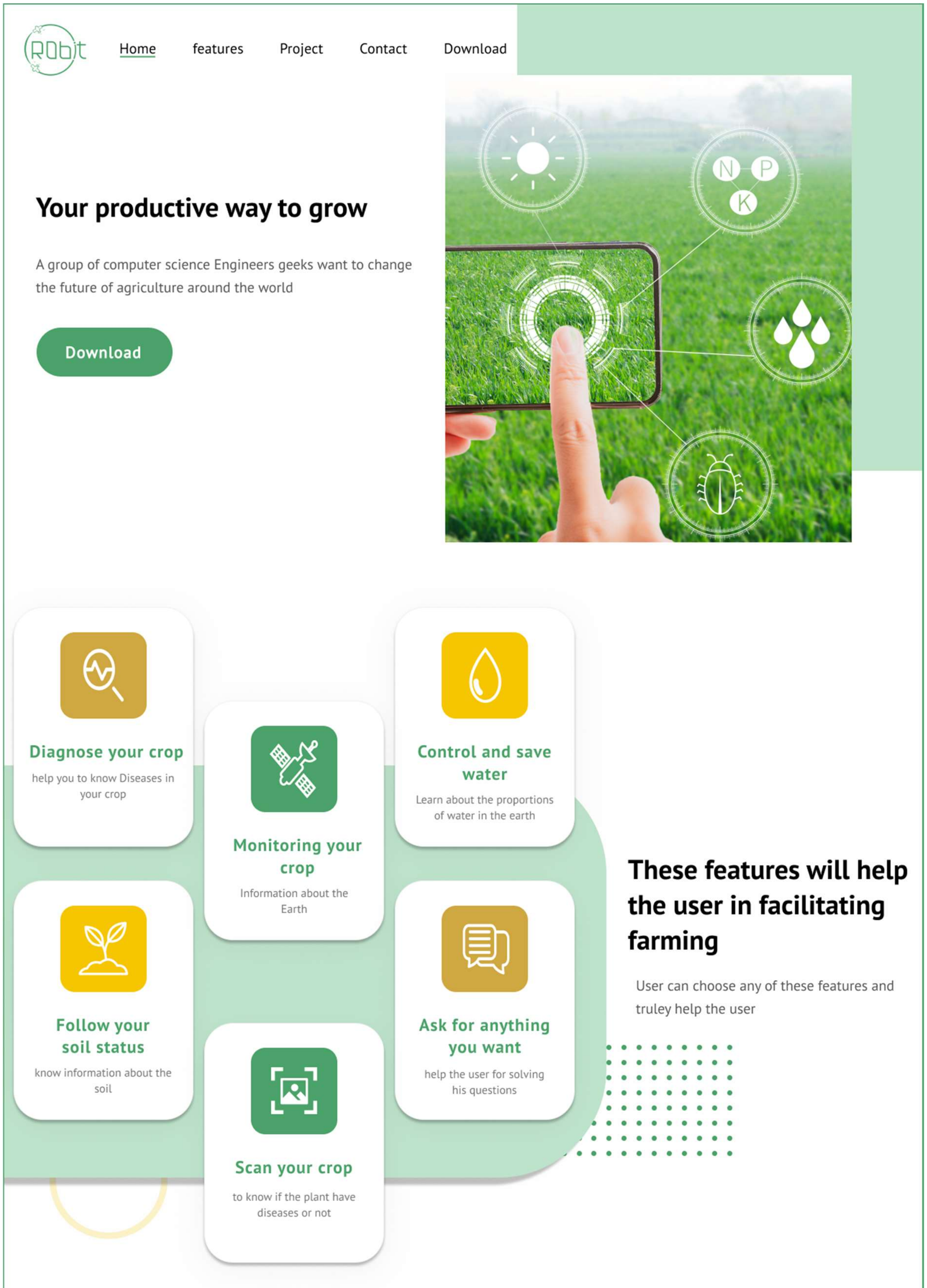

2:12 PM
Help & Support
About Us
Terms and condition


2:12 PM
Ali Mohamed
+039828288282
Rate Us
write something here.
Done
Log Out


2:12 PM
Verify code sent
Enter 4-digits code sent to you at +02 335 897 680 648
2
6
4
0
00:60 sec
verify code
Having a Problem?
Send Again


2:12 PM
Congratulations!
Your Account is ready to use . You will be redirected to the Home page in a few seconds.
Create New Password

- **Our UI design (Website)**



Our project aims to make farming easier and lossless

We help the user to learn about everything related to implantation using modern technologies such as AI , IOT and Satellite Imagery.

How Crobit app works?

01 — 02 — 03

Determine the location and select the field

You Should determine your farm location to be able to use our app and benefit fromour features.


Our Mobile App
Download Our Mobile App
follow your crop easily and productively using smart farming.
Play Store
Subscribe Newsletter
I will update good news and promotion service not spam
Enter your email address..
Contact Now
RObit
We aim to help farmers and make the cultivation process easier by following up on the surrounding environmental changes. Via sensors, AI, and satellites, we are able to do this by following the plant day-by-day, controlling the amount of water for irrigation, fertilizers, and early detection of plant and soil diseases.
What We Do
Detecting Plants & soil diseases
Irrigation control
Mobile Application
Contact
WhatsApp
Support 24
Chatbot
Support
FAQ
Policy
Business
Copyright © 2023 Crobit team

Chapter 5
IoT

5. Introduction

We use the Internet of Things nowadays in various things in our daily lives, IOT is a system of interrelated computing devices, mechanical and digital machines, objects, animals, or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction, so in this chapter, we will talk about the implementation of IOT in our project.

5.1 IoT in Agriculture

The main job of IoT in Agriculture is to collect data about the Soil, the surrounding environment, and the plants to take action built on this data.

Sensors

- Camera
- NPK
- Soil moisture sensor
- DHT22
- Water level detection sensor
  A single sensor of each type is required for a single Stick, then a drone is used to deliver this stick to a different area of the agricultural land.

5.2 Hardware components

5.2.1 ESP32_CAM

The ESP32-CAM is a full-featured microcontroller that also has an integrated video camera and microSD card socket. It’s inexpensive and easy to use and is perfect for IoT devices requiring a camera with advanced functions like image tracking and recognition. **Invalid source specified.**

Figure 17. ESP32_CAM

5.3 Sensors

5.3.1 NPK

Soil NPK sensor is designed for detecting the content of nitrogen 'N', phosphorus 'P', and potassium 'K' in the soil, determining the fertility of the soil, and facilitating the evaluation of the soil condition by the customer system.**Invalid source specified.**

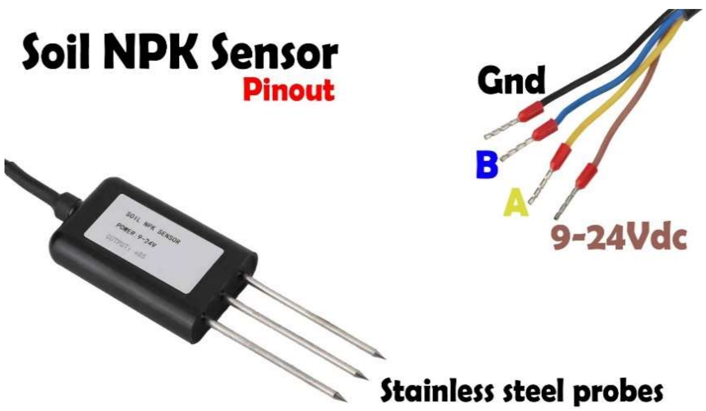


Figure 18 NPK sensor

5.3.1.1 Soil moisture

A soil moisture sensor can read the amount of moisture in the surrounding soil. It is ideal for monitoring gardens, or plant's water level.**Invalid source specified.**

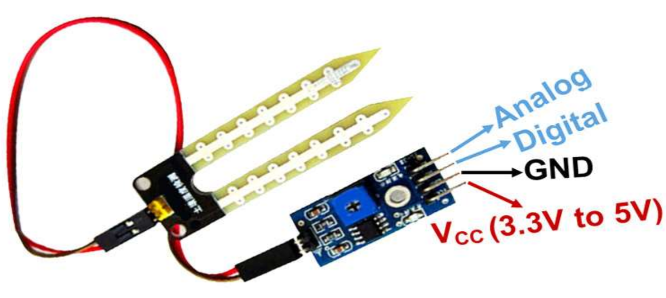


Figure 19 Soil moisture

5.3.1.2 DHT22 Sensor

low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analog input pins needed). It's fairly simple to use but requires careful timing to grab data.**Invalid source specified.**

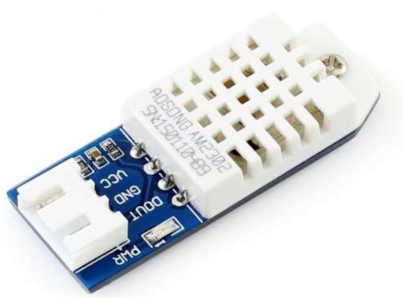


Figure 20 DHT22

5.3.1.3 Water level detection sensor

A device that measures the liquid level in a fixed container that is too high or too low.**Invalid source specified.**

Figure 21 Water level detection sensor

5.3.1.4 A6 GPRS GSM module

A6 GPRS GSM Module is a device that can be used to connect to the internet, send and receive SMS messages, and make and receive phone calls using a 2G SIM card. It can be integrated into many IoT projects that require wireless communication. It supports quad-band network and has a TTL serial port and a micro-USB interface. It also has a SIM card holder and an antenna interface.**Invalid source specified.**

Figure 22 A6 GPRS

Other hardware components

- FTDI USB to serial TTL converter
- Breadboard
- Jumpers
- Batteries
- Stick body

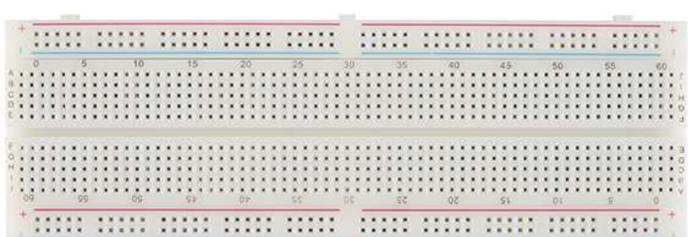


Figure 23 Breadboard

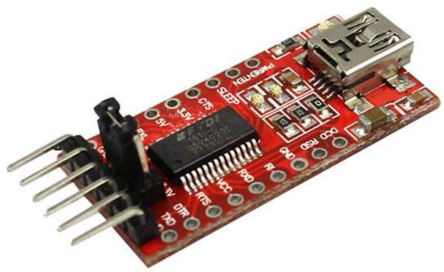


Figure 24 FTDI

5.4 Implementation Steps

5.4.1 Stick body

The first step is to make the stick body with 3d printing.

5.4.2 Camera

The camera is used to send pictures periodically to the AI model to detect if there are insects or harmful weeds.

First, we connect the ESP32_CAM with the FTDI to start the camera as follows:

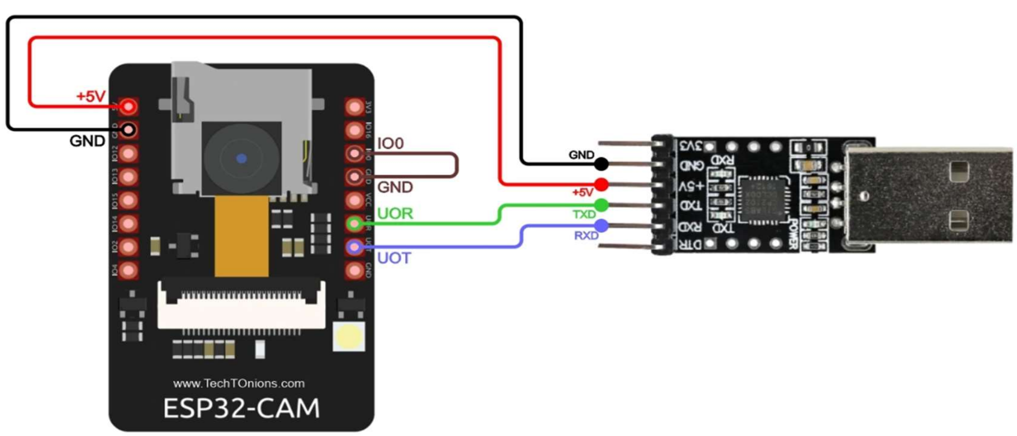


Figure 25 ESP32_CAM Connections

Second the code which is used to store the pictures in Firebase then send the picture's link to the backend server.

5.4.3 DHT22

We use DHT22 to measure the humidity and temperature of the surrounding environment to predict the weather conditions that affect the plants.

First, wiring DHT22 with the ESP32_CAM and the breadboard as follows:

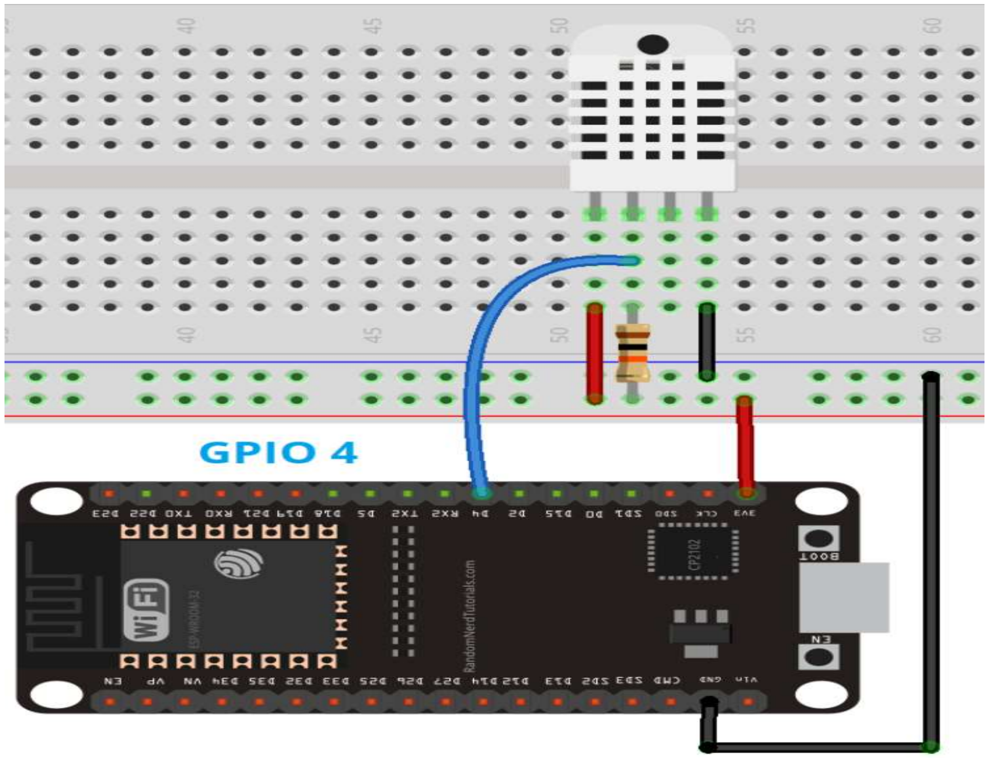


Figure 26 DHT22 with ESP32_CAM Connections

Second the code which sends the data measured by the sensor to the backend server.

5.4.4 Soil Moisture Sensor

We use the soil moisture sensor to measure the humidity of the soil to detect the exact amount of water that is still needed by the soil.

First wire the sensor with the ESP32_CAM as follows:

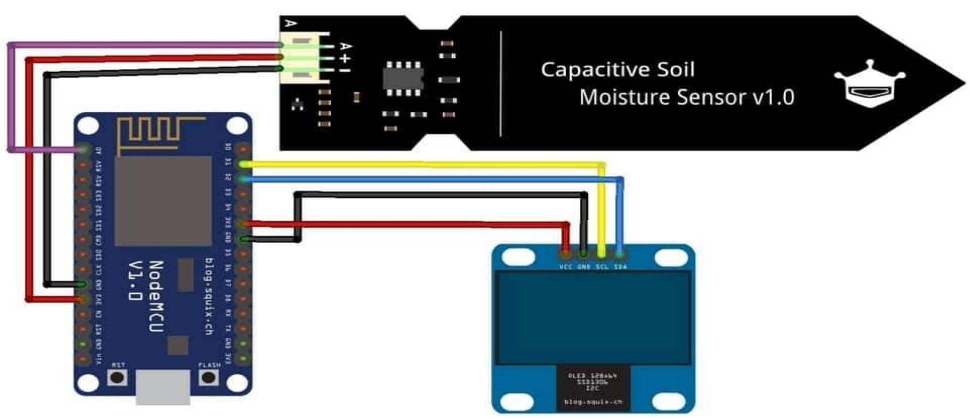


Figure 27 Soil moisture with ESP32_CAM Connections

Second the code which sends the data measured by the sensor to the backend server.

5.4.5 Water level detection

We use water level detection sensor to measure the water level above the ground, which helps to control the irrigation process with the soil moisture sensors.

First wire the sensor with the ESP32_CAM as follows:

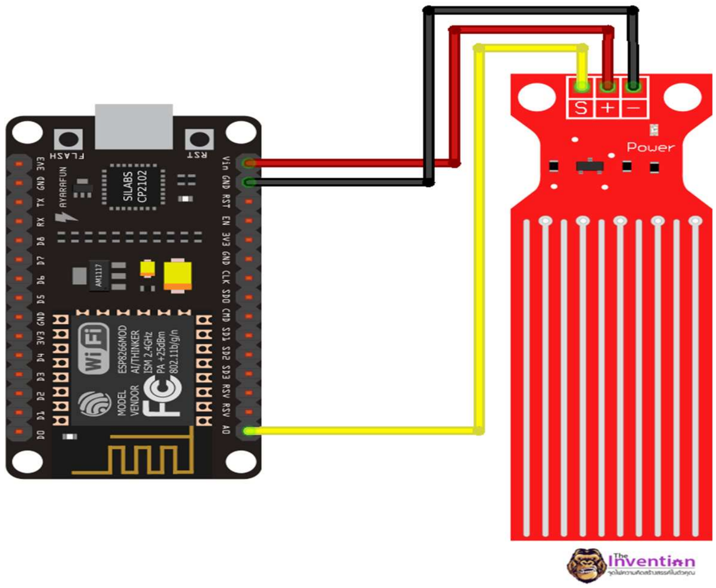


Figure 28 Water Level Detection sensor with ESP32_CAM Connections

Second the code which sends the data measured by the sensor to the backend server.

5.5 Mechanical specifications

5.5.1 Mechanical design of the drone

We use a drone as a solution to make our product portable, to hold a Stick with our sensors, and travel over the agricultural land of the user to get measurements in specific regions.

Now we will discuss the parts of the drone.

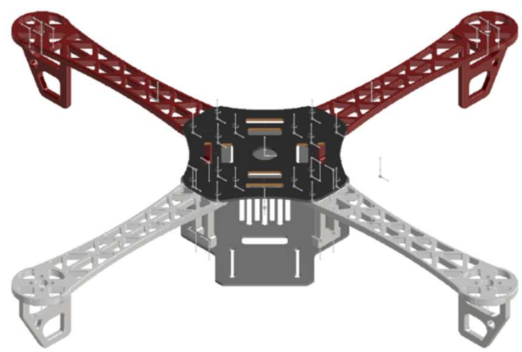


Figure 29 Drone Design

5.5.1.1 Bottom Area

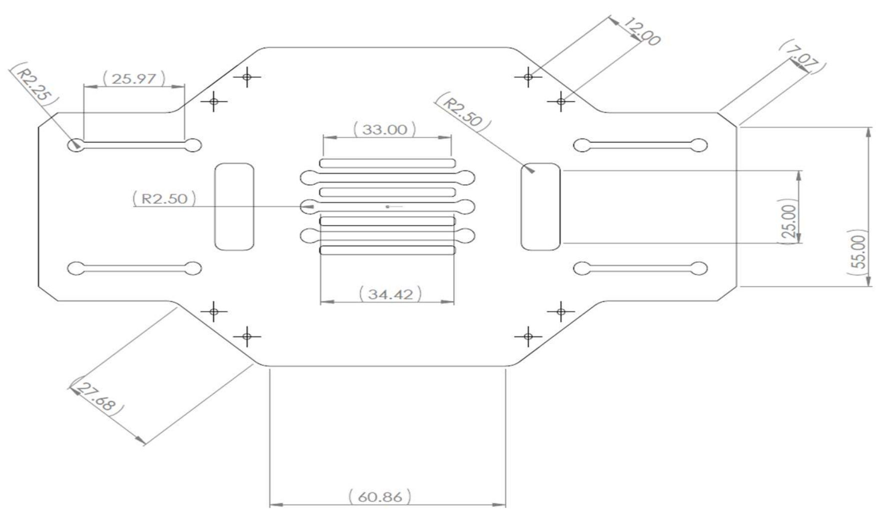


Figure 30 Diagram for the Bottom area of the drone

We split this part into three regions front, middle, and back. Front and back are similarly used to hold cameras, so there is an option to hold more than one camera with a hole to access wires and cable to have cable management for your camera.

In the middle, we have different shapes of holes that are made to have various options to mount the motor that holds the stick which we will discuss later. Also, these three regions use the upper side as a mount for the components of the drone, like raspberry pi shown in Figure 5.15**Invalid source specified.** and Pixhawk Flight Controller shown in Figure 5.16 which is the flight controller that handles our drone flight.**Invalid source specified.**

Figure 31 Raspberry Pi with its camera module

Figure 32 Pixhawk Flight Controller

5.5.2 Upper area

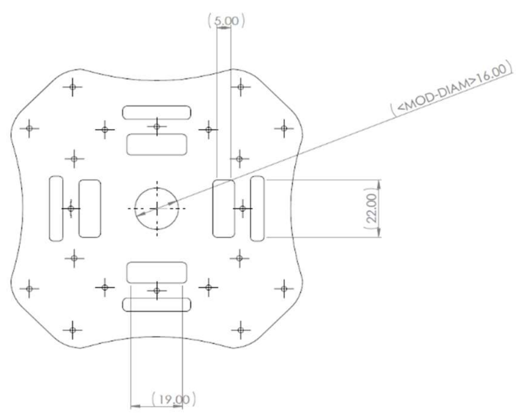


Figure 33 Upper area

This part we use to assembly wings and provide more stability, it is also used as accessory for Pixhawk as GPS show in figure 5.18**Invalid source specified.**

Figure 34 Pixhawk GPS

Features and Specifications of Pixhawk GPS

- Ublox Neo-M8N module
- Industry-leading –167 dBm navigation sensitivity
- Cold starts: 26s
- LNA MAX2659ELT+
- 25 x 25 x 4 mm ceramic patch antenna
- Rechargeable Farah capacitance
- Low noise 3.3V regulator
- Current consumption: less than 150mA @ 5V
- Fix indicator LEDs
- Protective case
- Cable Length: 26cm (42cm cable can be purchased here)
- Diameter 50mm total size, 32 grams with case

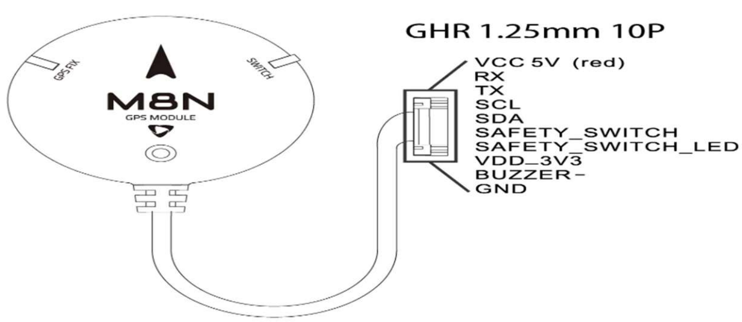


Figure 35 will show a diagram of pins

5.5.3 FrSky 2.4GHz Access Archer R8 Pro Receiver

This part is used for receiving data to control the drone.

- ACCESS protocol with Over the Air (OTA) Anti-interference in spark-ignition
- Supports signal redundancy (SBUS In) Full control range with telemetry.
- S.Port / F.Port External battery/device voltage detection
- Voltage Measurement Range via AIN2 (External device): 0-36V Compatibility: All FrSky ACCESS transmitters**Invalid source specified.**

Figure 36 FrSky 2.4GHz Access Archer R8 Pro Receiver

5.5.4 FrSky Taranis X9D Plus

We use it to send data to drone for control.

Features**:**

- Compatibility: ACCST D16 and ACCESS receivers
- Classic Taranis form factor design Easy launch momentary button Program navigation button
- High-speed module digital interface Installed with ACCESS protocol Supports spectrum analyzer function.
- Supports SWR indicator warning G9D potentiometer gimbal Haptic vibration alerts and voice speech outputs.
- The FrSky Taranis X9D Plus 2019 is a re-designed version with additions like an additional momentary button placed on the top left shoulder making it ergonomically friendly for DLG pilots to activate launch mode, and features a program scroll wheel making it even easier to navigate the menus.**Invalid source specified.**

Figure 37 FrSky Taranis X9D Plus

5.5.5 Wings

Figures 5.22 and 5.23 show the two respective of the wing which functionality is holding motors and motor drivers to make the drone fly.

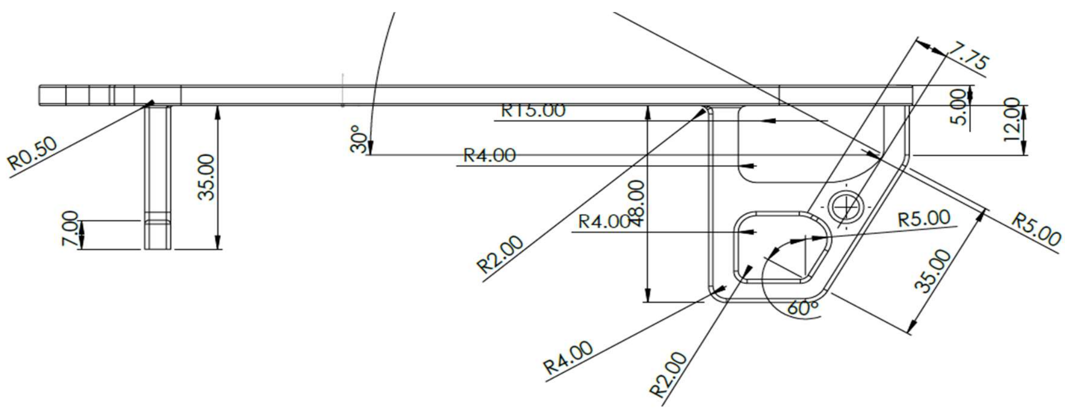


Figure 38 Design of wing 1

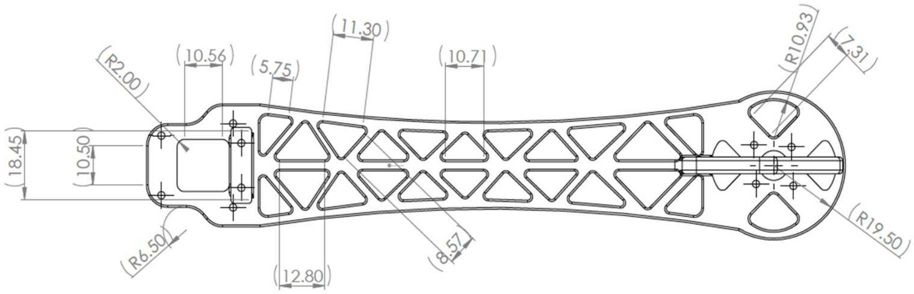


Figure 39 Design of wing 2

We use A2212 6T 1400KV Brushless Motor for Drone shown in figure 5.24 and, its louds in the table in figure 5.25**Invalid source specified.**

Figure 40 Brushless motor

MODEL	KV (rpm/V)	Voltage (V)	Prop	Load Current(A)	Power (W)	Pull (g)	Efficiency (g/W)	Lipo Cell	Weight (g)
A2212	930	11.1	1060	9.8	109	660	6.1	2-4S	52
	1000		1047	15.6	173	885	5.1		
	1400		9050	19.0	210	910	4.3		
	1800		8060	20.8	231	805	3.5		
	2200		6030	21.5	239	732	3.1	2-3S	
	2450		6030	25.2	280	815	2.9		

Figure 41 Tables show loads of motor

In this wing we have multiple cuts or holes in the body of the arm of the wing this is used to decrease the weight of the drone which makes the drone more capable of lifting more weight and consuming less power. It also holds the driver which delivers specific amounts of power to the motor it needs and for the control of speed we use a Standard 30A BLDC ESC Electronic Speed Controller shown in Figure 5.26

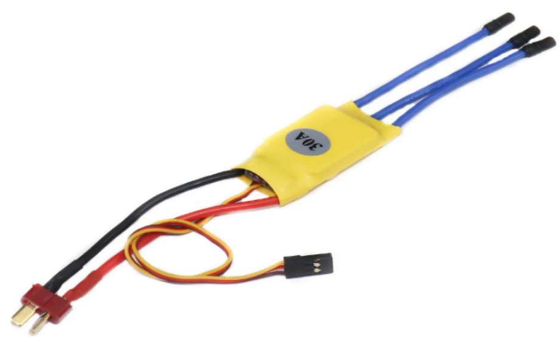


Figure 42 Standard 30A BLDC ESC Electronic Speed Controller

Standard BLDC 30-amp ESC Electronic Speed Controller with Connector is specifically made for quadcopters and multi-rotors. It provides faster and better motor speed control giving better flight performance compared to other available ESCs.

Standard 30-amp ESC Electronic Speed Controller can drive motors that consume up to 30A current. It works on 2-3S LiPo batteries. The onboard BEC provides regulated 5V (2A max draw) to power the flight controller and other onboard modules. This is useful to control our brushless motors with a 2-3S LiPo (make sure the motor doesn't draw more than 30A).

ESC CONNECTORS:

Battery Side – Male Deans (T) Connector.

Motor Side – Female 3.5mm Gold Connector.

Features:

- This 30-amp ESC for BLDC has onboard BEC
- This ESC comes with a male dean T-connector.
- Auto Low BATTERY Slow down at 3.0V/cell Lipo, cut-off at 2.9V/cell lipo
- APPLICATION: Multirotor, Rc Airplanes, etc

Mechanical design of the stick

This stick design will be printed with 3D printing, then holds our sensors.

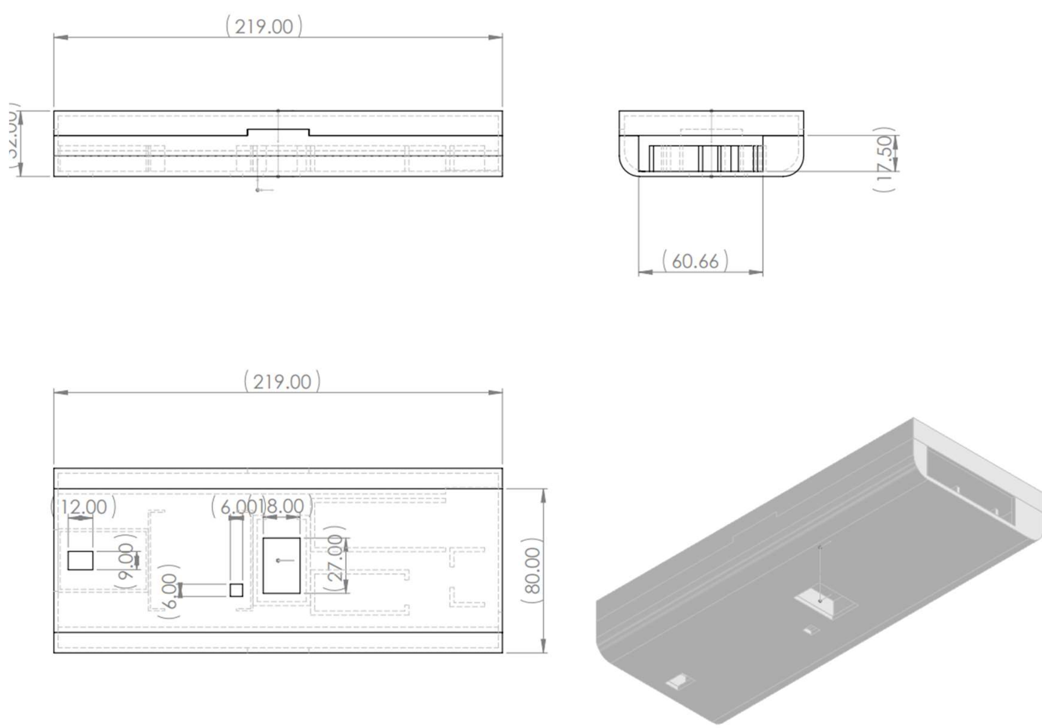


Figure 43 stick design

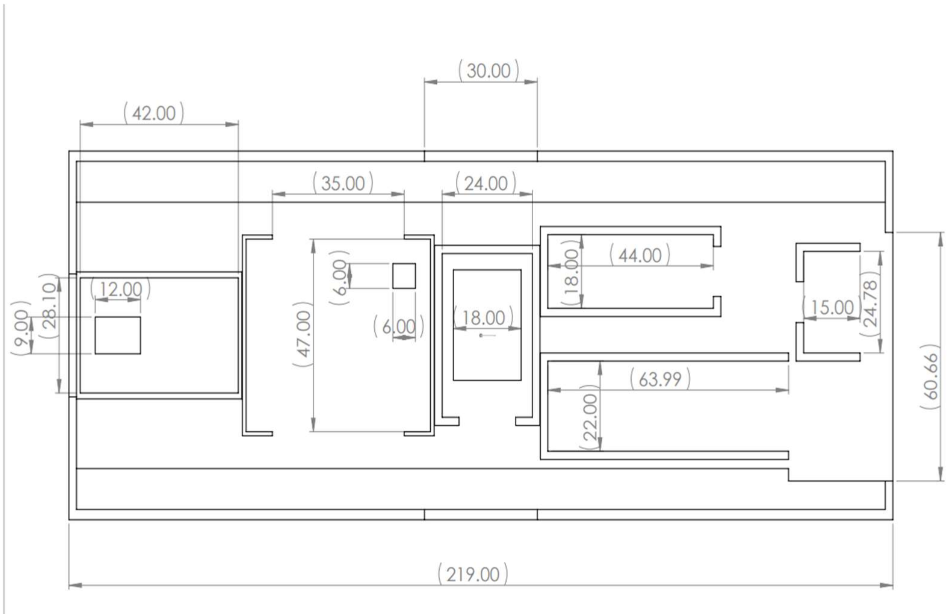


Figure 44 stick design

6. Weeds Detection Model

Weeds are unwanted plants that grow in agriculture fields and compete with crops for nutrients, sunlight, and water. They can cause significant reductions in crop yield and quality if left unchecked. Therefore, it is crucial to detect and remove weeds from agriculture fields to preserve the quality and quantity of crops [1].

Detecting weeds in agriculture fields has many advantages such as:

- **Prevent crop yield loss:** Weeds can reduce crop yield by competing with crops for essential resources. If left uncontrolled, weeds can cause significant reductions in crop yield and quality, leading to economic losses for farmers.
- **Reduce the use of herbicides:** Detecting weeds early in the growing season can reduce the need for herbicides. Herbicides are expensive, and their overuse can lead to environmental pollution and harm to non-target organisms, including humans.
- **Improve soil health:** Weeds can have adverse effects on soil health by altering soil structure, nutrient availability, and water-holding capacity. By removing weeds, farmers can improve soil health and maintain the long-term productivity of their fields.

So, one of our goals is to detect weeds in agriculture land and inform the farmer of their location to get rid of them. We can detect weeds with a binary classification model that takes an image from the drone and classify this image if it is weeded or not, and if it weeds the model will return this image to the mobile application with its location in the field.

6.1 Dataset

The DeepWeeds dataset consists of 17,509 unique 256x256 color images in 9 classes, 8 classes of weeds, and one negative class (not weed). These images were collected in situ from eight rangeland environments across northern Australia.

Because of the high similarity between classes, and our concern is to detect if there are weeds or not, we merged these 8 classes into one class. So, now the dataset consists of two classes, weed or not weed, and we can do a binary classification for this dataset.

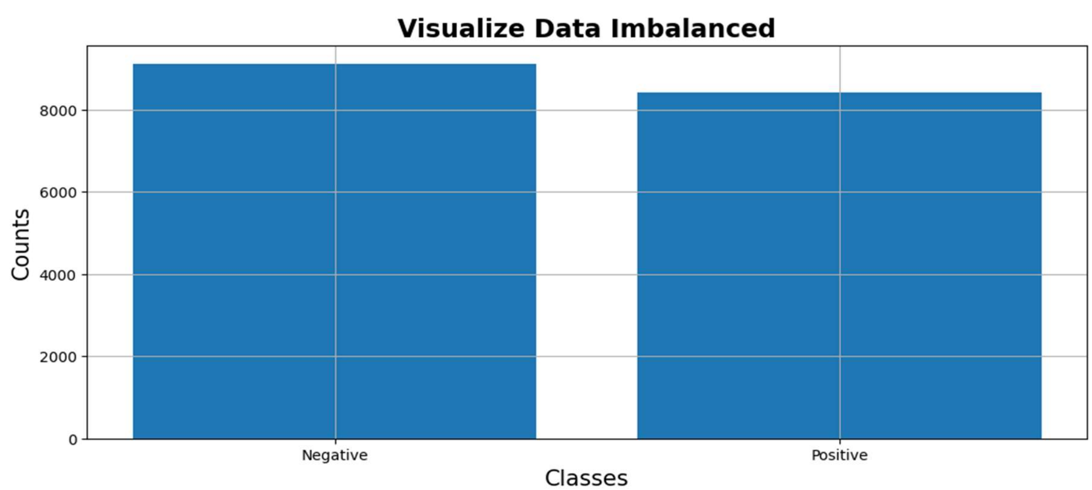


Figure45 Dataset classes and their counts

The dataset is split into three parts for training, validation, and testing. The split is as follows:

- 70% for training
- 15% for validation
- 15% for testing

6.1.1 Training Set

The training set consists of 12,257 images. This set is used to train the model and optimize its parameters. The training set is large enough to ensure that the model learns a diverse range of features and can generalize to new images.

6.1.2 Validation Set

The validation set consists of 2,626 images. It is used to evaluate the model's ability to distinguish between weeds and non-weeds. During the training process, the model is trained on the training set and its performance is evaluated on the validation set. The purpose of the validation set is to provide an estimate of the model's performance on unseen data. This is important because if the model is overfitting to the training data (i.e. memorizing the training set rather than learning generalizable features), its performance on the validation set will be worse than its performance on the training set. By monitoring the performance of the model on the validation set during training, we can make decisions about when to stop training or adjust the hyperparameters of the model.

6.1.3 Test Set

The test set consists of 2,627 images. It is used to evaluate the model's ability to correctly classify new images as either a weed or not a weed. The test set is composed of images that are representative of the overall dataset and should not be used for any training or fine-tuning of the model. The purpose of the test set is to evaluate the overall performance of the model and provide an unbiased estimate of the model's performance on new, unseen data. This is important because if the model has overfit to the training data or the validation set, its performance on the test set will be worse than its performance on the training and validation sets.

6.1.4 Image Format

All the images in the dataset are 256x256 pixels in size and are in color format. The color format is important as it provides additional information that can help the model make better predictions.

6.1.5 Image Labeling

Each image in the dataset is labeled as either a weed or not a weed. The labeling of the images is important as it provides the ground truth for the model to learn from and evaluate its performance.

6.2 Data Preprocessing

Data preprocessing is an important step that is crucial to ensure that the model can learn relevant patterns and generalize well to new data. The primary goal of data preprocessing is to prepare the dataset for training by applying appropriate augmentation and normalization techniques.

6.2.1 Data Augmentation

Data augmentation is a technique that is used to increase the size and diversity of the training dataset by applying various transformations to the original images [2]. In this project, data augmentation techniques such as random flip horizontal, random flip vertical, random rotation, and random zoom are applied to the dataset. These techniques are applied randomly to each image in the training set, creating new images that are slightly different from the original ones. This increases the size and diversity of the training set and helps the model to generalize better to new data.

6.2.2 Data Normalization

Normalization is a technique that is used to scale the pixel values of the images to a specific range [3]. In this project, the pixel values of each image in the dataset are scaled by dividing each pixel value by 255. This normalization technique ensures that the pixel values of the images are between 0 and 1, which is a common range for image data.

6.3 Model Architecture and Selection

The model architecture plays a crucial role in the performance of a deep learning model. In this task, we tried several architectures such as ResNet50, InceptionV3, and ConvNeXtSmall.

6.3.1 ResNet50

ResNet, short for "Residual Network," is a type of convolutional neural network (CNN) that was developed by researchers from Microsoft in 2015 [4]. ResNet is based on the idea of "residual learning," which allows for the training of much deeper neural networks than was previously possible.

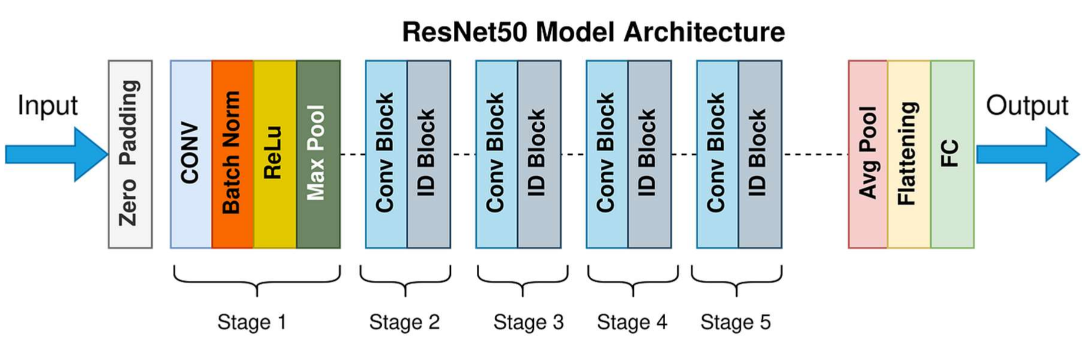


Figure46 . ResNet50 Architecture

ResNet has had a significant impact on the field of computer vision, particularly in the area of image classification. Prior to ResNet, researchers found that as CNNs became deeper, they were more difficult to train, and their performance would often plateau or even degrade. By using residual connections, ResNet50 was able to significantly increase the depth of the network while maintaining or improving its accuracy. This architecture has become a standard benchmark for image recognition tasks and has been used as a base architecture for many other deep learning models.

ResNet50 is composed of several building blocks known as "Residual Units". Each Residual Unit consists of two or three convolutional layers, followed by a shortcut connection that bypasses the convolutional layers and adds the input directly to the output of the unit.

Skip/Shortcut connection

Skip connections are a type of connection used in neural networks to make it easier for the network to learn complex patterns. In a typical neural network architecture, each layer receives input from the previous layer and passes its output to the next layer. However, this can sometimes result in the loss of information, making it difficult for the network to learn complex patterns.

Skip connections solve this problem by allowing information to bypass one or more layers in the network and be passed directly to a later layer. This allows the network to preserve important information and makes it easier to learn complex patterns.

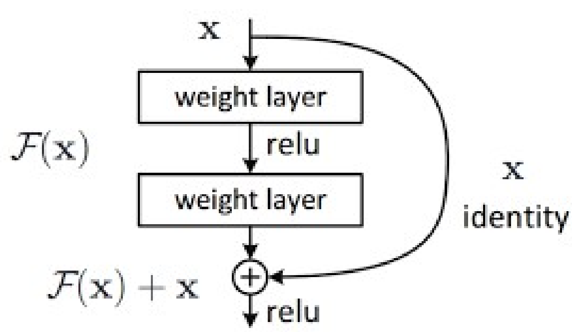


Figure47 . Skip Connection

How do Skip Connections Work?

Skip connections work by adding a shortcut connection between two layers in a neural network. This shortcut connection allows the output of one layer to be added directly to the output of another layer, bypassing any layers in between.

The output of the skip connection is then passed through a non-linear activation function before being fed into the next layer. This helps to ensure that the output of the skip connection is transformed in a way that is compatible with the output of the next layer.

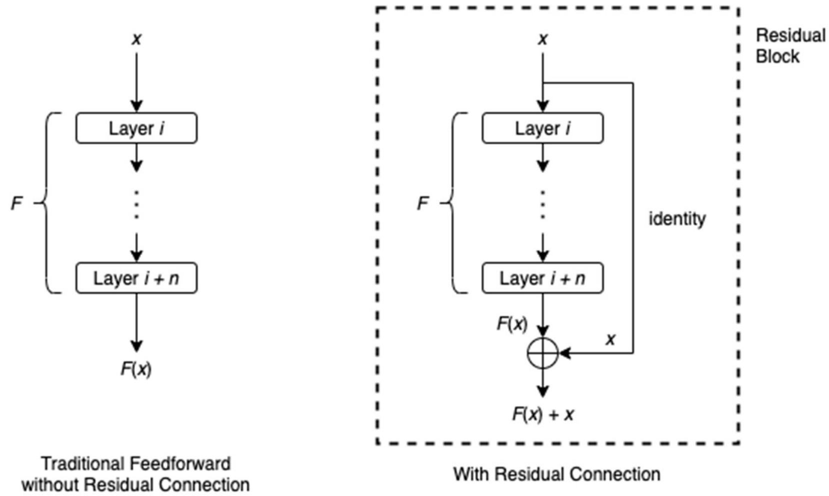


Figure48 . Difference between Neural Networks with and without Skip Connections

The main difference between neural networks with and without skip connections is their ability to learn complex patterns. Neural networks with skip connections are better able to learn complex patterns because they are able to preserve important information and make it easier for the network to learn from that information.

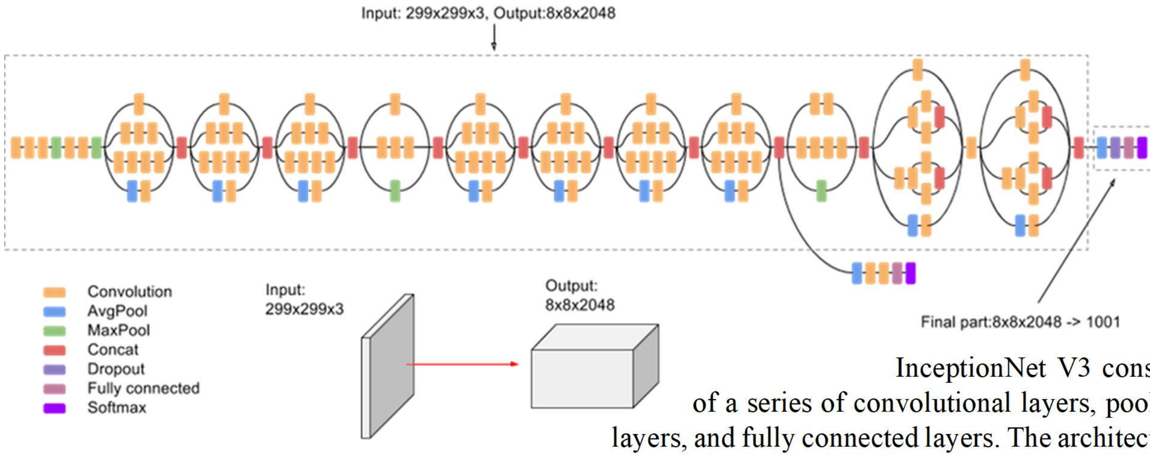
In contrast, neural networks without skip connections can sometimes struggle to learn complex patterns because they are limited by the information that is passed from one layer to the next. This can result in the loss of important information and make it difficult for the network to learn complex patterns.

Another difference between neural networks with and without skip connections is their depth. Neural networks with skip connections can be much deeper than those without, as skip connections help to alleviate the problem of vanishing gradients. This allows the network to learn more complex patterns and achieve higher levels of accuracy.

6.3.2 Inception Network

InceptionNet V3 is a neural network architecture for image classification developed by Google [5]. It is a deep convolutional neural network (CNN) that was introduced in 2015 and is an extension of the original InceptionNet architecture. InceptionNet V3 is designed to improve accuracy and reduce the number of parameters in the network.

InceptionNet V3 is an important development in the field of computer vision because it is able to achieve state-of-the-art performance on various image classification tasks. The architecture is designed to be both accurate and efficient, making it ideal for use in real-world applications where speed and accuracy are both important. InceptionNet V3 has been used in a number of applications, including image recognition, object detection, and visual search.



InceptionNet V3 consists of a series of convolutional layers, pooling layers, and fully connected layers. The architecture

Figure49 . Inception V3 Architecture

is designed to be modular, with each module consisting of a set of parallel convolutional layers with different filter sizes. The outputs of these parallel layers are then concatenated and fed into the next module.

Inception Module

The Inception module is a key component of the InceptionNet V3 architecture, and it is designed to allow the network to learn representations at multiple scales. It is a type of convolutional neural network (CNN) module that consists of a set of parallel convolutional layers with different filter sizes, followed by a pooling layer and a set of 1x1 convolutional layers.

The main idea behind the Inception module is to allow the network to learn representations at multiple scales by using filters of different sizes. By using filters of different sizes, the network can learn both fine-grained and coarse-grained features, which can improve the accuracy of the network on various image classification tasks.

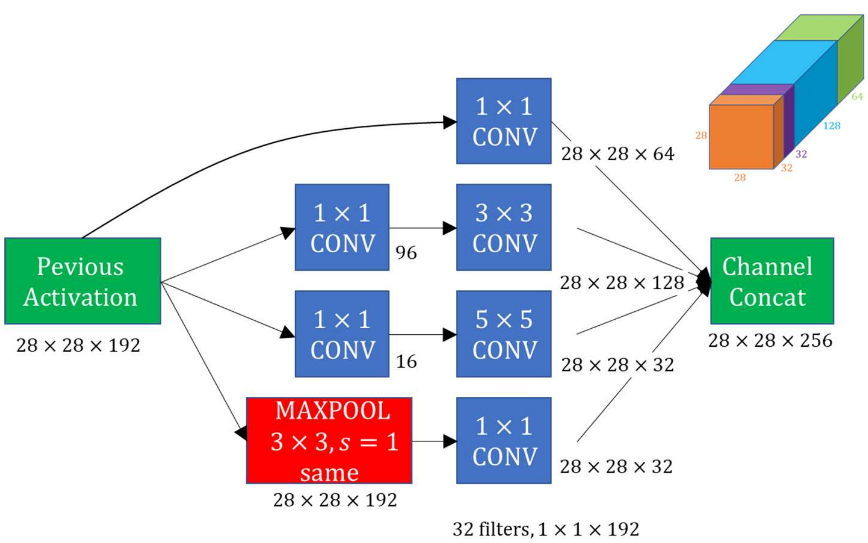


Figure50 . Inception Module

Inception modules are typically composed of 1x1, 3x3, and 5x5 convolutional layers, as well as pooling layers. The 1x1 convolutional layers are used to reduce the depth of the input, while the 3x3 and 5x5 convolutional layers are used to capture features at different scales. The pooling layer is used to reduce the spatial dimensions of the input and is typically used to reduce the computational cost of the network. The output of the pooling layer is then passed through the 1x1 convolutional layers, which are used to reduce the depth of the input.

The outputs of the parallel convolutional layers are then concatenated along the depth dimension and passed through the pooling layer and the 1x1 convolutional layers. The output of the Inception module is then passed to the next layer in the network.

Auxiliary Classifiers

InceptionNet V3 also includes auxiliary classifiers, which are used to improve the training of the network. The auxiliary classifiers are designed to encourage the network to learn discriminative features early in the network, which can improve the overall accuracy of the network.

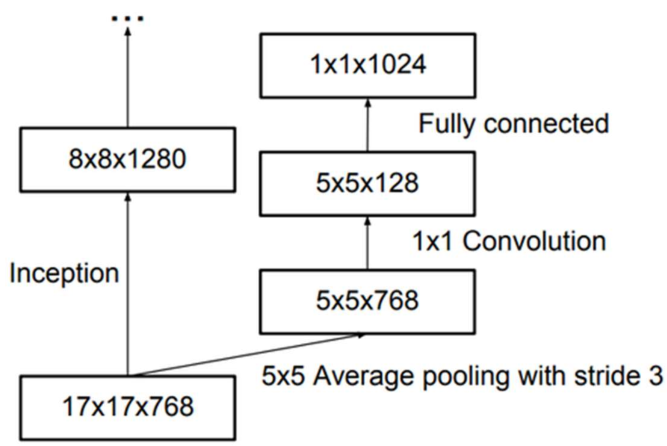


Figure51 . Auxiliary Classifier Architecture

The auxiliary classifiers are added at intermediate layers of the network and consist of a pooling layer, a set of convolutional layers, a fully connected layer, and a softmax layer. The output of the auxiliary classifier is combined with the output of the main classifier to produce the final output of the network.

6.3.3 Vision Transformer

Transformers are a type of neural network architecture that have shown outstanding results in natural language processing (NLP) tasks. Recently, transformers have been adapted to computer vision tasks with promising results, challenging the traditional convolutional neural network (CNN) architecture [6].

Transformers are based on the self-attention mechanism that allows them to learn long-range dependencies in sequential data. In computer vision, the transformer architecture operates on the image pixels. This Transformer model introduced a patch layer that splits an image into a sequence of patches of 16 by 16 pixels. This Transformer model is called the Vision Transformer (ViT).

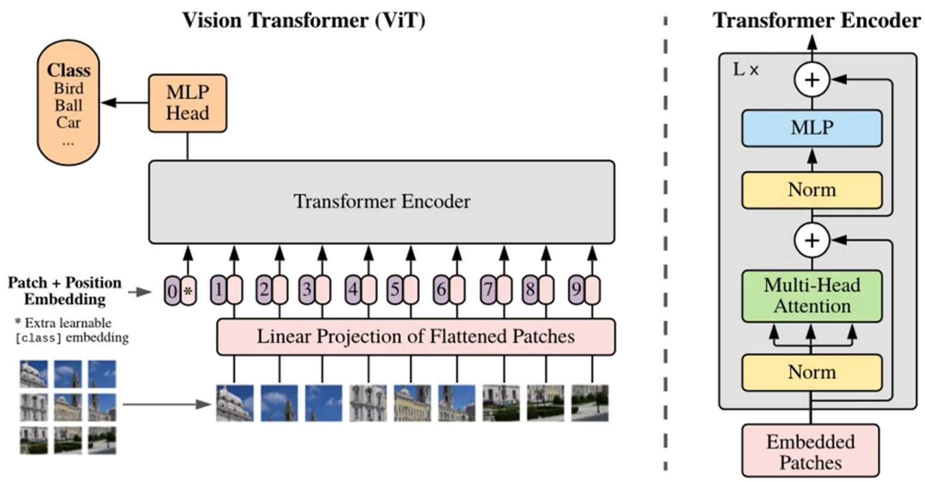


Figure52 . Vision Transformer (ViT)

6.3.4 ConvNeXt

ConvNeXt is a type of convolutional neural network (CNN) which is a pure convolutional model (ConvNet), inspired by the design of Vision Transformers, that claims to outperform them. ConvNeXt is designed to combine the strengths of both depth-wise and point-wise convolutions, which are two common types of convolutional operations used in CNNs. Depth-wise convolutions are used to capture spatial features within each channel of the input feature maps, while point-wise convolutions are used to combine information across different channels.

The key idea behind ConvNeXt is to use a split-transform-merge strategy to combine the strengths of depth-wise and point-wise convolutions. In this strategy, the input feature maps are split into multiple groups, and each group is processed separately using a combination of depth-wise and point-wise convolutions. The outputs of each group are then merged together to produce the final output feature maps.

The split-transform-merge strategy allows ConvNeXt to capture both spatial and channel-wise information effectively and has been shown to be highly effective in image classification tasks. In particular, ConvNeXt has achieved state-of-the-art performance on the ImageNet dataset, which is a large dataset of images used for benchmarking image classification algorithms.

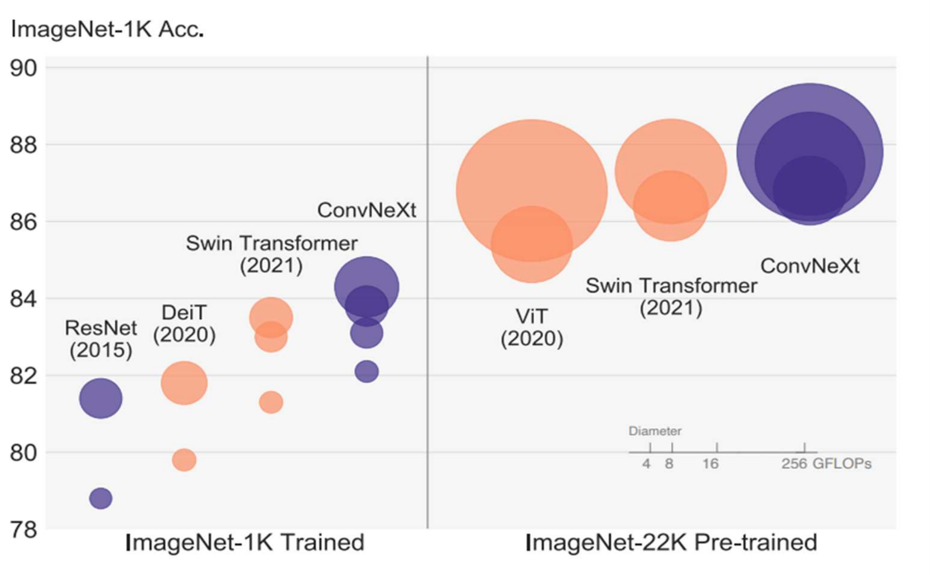


Figure53 . ImageNet-1K/22K classification results for ConvNets and vision Transformers

In our project, we used ConvNeXt architecture because ConvNeXt has several advantages over other CNN architectures we tried, such as ResNet and Inception. For example, ConvNeXt has a lower computational cost than ResNet, while achieving similar or better performance. ConvNeXt also has better performance than Inception.

ImageNet

ImageNet is a large-scale dataset of images that is widely used for benchmarking image classification algorithms. The dataset contains over 14 million images that are classified into more than 20,000 categories. Each image is labeled with one of the categories, which include objects, animals, plants, and scenes.

Figure54 . ImageNet Dataset

Transfer Learning

Transfer learning is a technique in deep learning where a pre-trained model is used as a starting point to solve a new task. In the context of convolutional neural networks (CNNs), transfer learning involves using a pre-trained CNN model as a feature extractor on a new dataset [7].

In our project, we used "imagenet" pre-trained weights as transfer learning for our chosen architecture which is the ConvNeXt architecture. The "imagenet" weights are pre-trained weights that have been trained on the ImageNet.

These pre-trained weights are used to initialize the weights of the ConvNeXt-S architecture, which helps the model to learn relevant features of the images more quickly and effectively.

Our chosen model (ConvNeXt-S) consists of 295 layers, and the first 230 layers are frozen while the remaining layers are unfrozen except for the top layers that were not included in the model architecture. This is done to fine-tune the model's weights and improve its ability to classify weeds accurately.

A functional model was created using the Keras API, which included the data augmentation and scaling steps, followed by additional layers such as GlobalAveragePooling2D, Dropout with a dropout probability of 0.2, a Dense layer with 10 units and a ReLU activation function, and an output layer with a Dense layer with 2 units and a softmax activation function.

6.4 Training

During training, the **EarlyStopping** callback was used with a patience of 5 to prevent overfitting of the model during training. The **SparseCategoricalCrossentropy** loss function was used to calculate the loss by summing the logarithmic loss of the predicted probabilities for the two classes, multiplied by the true label. And the **Adam** optimizer with a learning rate of 0.001 was used to optimize the model's parameters. The model was trained for 20 epochs, with the training process monitored using the validation accuracy metric.

6.5 METHODS

6.5.1 Datasets

6.5.1.1 Wheat Leaf Disease Detection Dataset

Our model trained on a dataset of **wheat leaf disease** images to be classified, collected manually from diverse number of sources for each class to generalize the process and cover most of wheat leaf diseases with enough images to prevent some serious problems such as **overfitting.**

This dataset consists of sex different diseases, which are the most known and common ones such as brown rust, loose smut, mildew, septoria, stem rust, and yellow rust with quite good amount of instances as could as possible due to the obstacles showed up when collecting data as usual as shown in Table 1.

Table 1. Wheat Leaf Diseases

Disease	No. of Instances
Brown Rust	**1256**
Loose Smut	**939**
Mildew	**161**
Septoria	**349**
Stem Rust	**376**
Yellow Rust	**1395**
Healthy	**1658**
Total	**6134**

6.5.2 Insects Dataset

This dataset used to detect insects in a certain field, regardless the type of the insect, aiming to detect harmful types of insects which can damage the farm and locate their position to make an action by the means of diverse tools or pests specialized for this type of elimination.

This dataset was originally created by Nirmani with approx. 1000 images used to train our model here, this dataset is part of RF100, an Intel-sponsored initiative to create a new object detection benchmark for model generalizability [8].

6.6 Models

For classification part in the wheat leaf disease, which is proposed to classify a colossal number of major diseases that could affect our yield, so we need to use the appropriate algorithm for that in terms of time and performance as a trade-off. There are tremendous number of algorithms that can be used but countable number of common verified algorithms to choose between.

The key difference between plain NN or Connected layers NN and CNN, is that for plain NN as shown in **Error! Reference source not found.**, the input pass through each unit in the next layer by two steps one is linear step and the second is adding some non-linearity to the system (activation function) and the number of input in the previous layer is equal to the number of parameters of that node which are going to be trained, therefore it has a huge number of parameters to train.

The first work on modern convolutional neural networks (CNNs) occurred in the 1990s, inspired by Noncognition. Yann LeCun et al., [9], which applied that which converts simple features in the early stages or layers to complex one's latter can used efficiently to recognize handwritten character recognition.

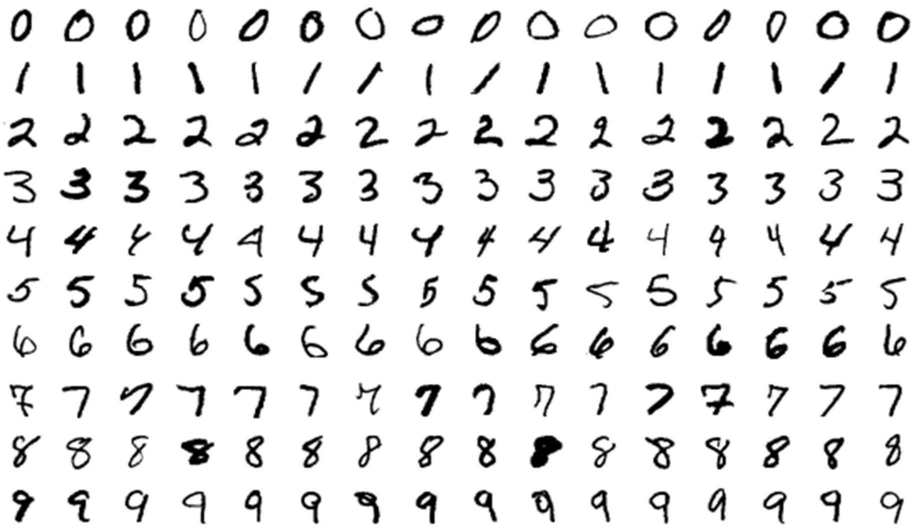


Figure 55. MNIST Dataset Sample [10]

A small gap of development, around 2012 was in a huge popularity especially with the existence of ImageNet dataset [11] which is a freely- available dataset for images and currently it consists of more than 14 million high quality images with different 21,000 classes which make it a great space and environment for validation process of algorithms and architectures.

One key advantage of Convolutional Neural Networks (CNNs) is their ability to effectively extract and learn spatial hierarchies of features from input data. This makes CNNs particularly well-suited for tasks involving images and other grid-like data, such as audio spectrograms or 2D sensor data.

We use the following architecture to compare the number of parameters of both CNN and FC layers. After handling the process of extracting features from data or image for the architecture proposed in Figure 56, there is a necessary evil here (like friction to cars) which is the fully connected layers at the end of the architecture as it's useful for classification part but it takes more than 58 m parameters of total 60 m, so what if the whole network is FC layers.

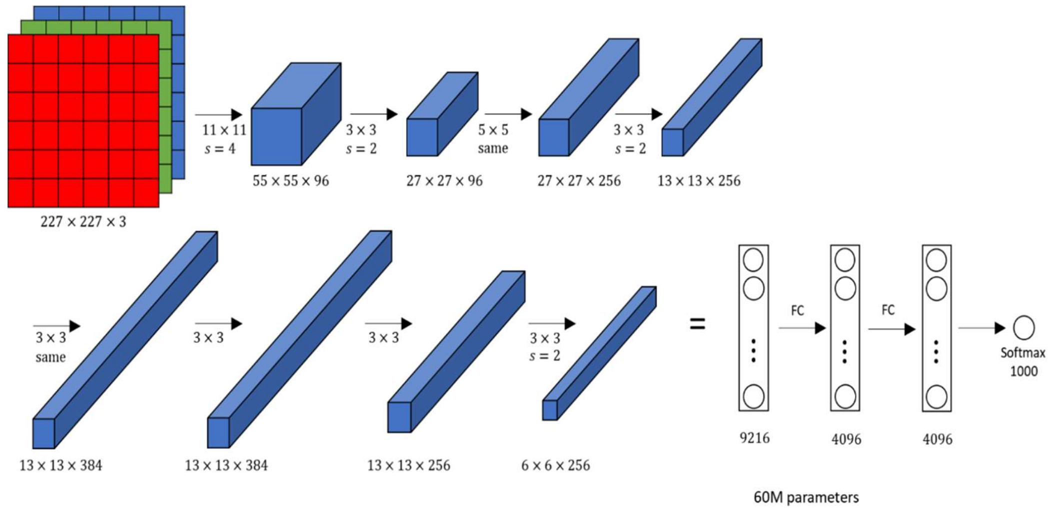


Figure 56. AlexNet Architecture introduced in the following paper [12]

with vast critical algorithms as well introduced in this paper, and here showing the number of parameters which is appx. 60 m

Therefore, we got to use the CNN concept here to save time and for performance as well, specially, we are doing a computer vision task.

Moving to the next part, which is object detection task, a model of recognizing disparate number of harmful insects and locate them for elimination purposes. Object detection has seen significant improvements over the years, thanks to advancements in deep learning and computer vision techniques. Several algorithms have contributed to these improvements such as RCNN, Fast and Faster RCNN, YOLO, and SSD.

Every single algorithm mentioned above has its capabilities and it's hard to choose between as it depends on various factors requirements of your application, available computational resources, and desired trade-offs between speed and accuracy. For our project we choose to use YOLO (You only look once) v8 [13], and as the name described it's about looking and gridding with disparate number of anchors and do a simple classification after that.

YOLO always has that concept of development as it has lots of versions to choose between and selecting the version to use is important as well. YOLOv8 is designed to be fast, accurate, and easy to use, making it an excellent choice for a wide range of object detection and tracking, instance segmentation, image classification and pose estimation tasks.

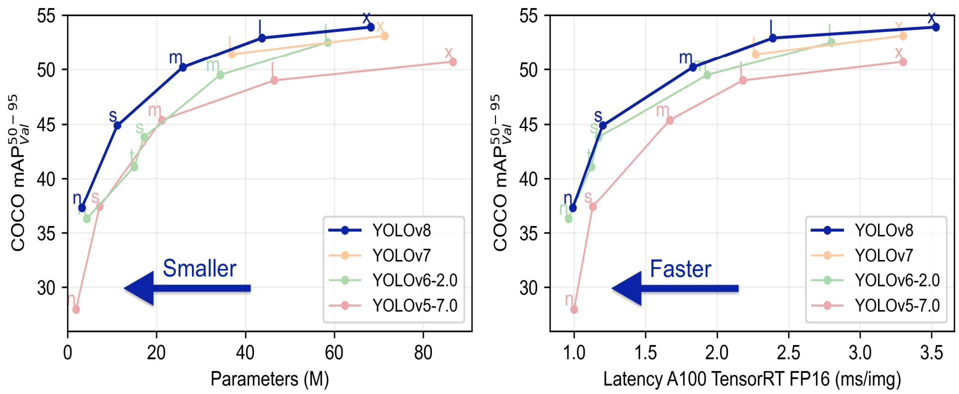


Figure 57. Comparison between different YOLO versions in terms of Latency, accuracy, and size [14]

6.6.1 Wheat Leaf Disease Detection Model

Wheat is **susceptible** to various leaf diseases that can significantly impact crop yield and quality, management of these diseases is challenging process. We aim to ease this process and reduce the load and dependency on local agricultural extension services or plant pathologists. We must automate the process of detecting these diseases on a daily basis for the sake of early detection which is critical to manage these diseases.

With seven disparate common diseases we made our model with approx. 1% validation error rate after a sequence of trials and playing with different hyper parameters and architectures exist, below we are going to demonstrate a sample of that sequence which yield to these outstanding results.

Overfitting is an expected problem which normally came by intuition due to the bigger networks existing these days, maybe the data couldn't handle that size in some fields such as our problem here. Lack of data and resources, we got to collect every class images manually from disparate sites and databases, nevertheless, there is always a panacea such as Dropout technique and Data Augmentation [12] .

Data Augmentation, we used that technique by intuition as we are dealing with a computer vision task, and it is the easiest and most common method to reduce overfitting on image data is to artificially enlarge the dataset using label-preserving transformations and it helps improving the generalization and robustness of machine learning models. We did the following methods image flipping, rotations, scaling, zooming and cutout as shown in Figure 58.

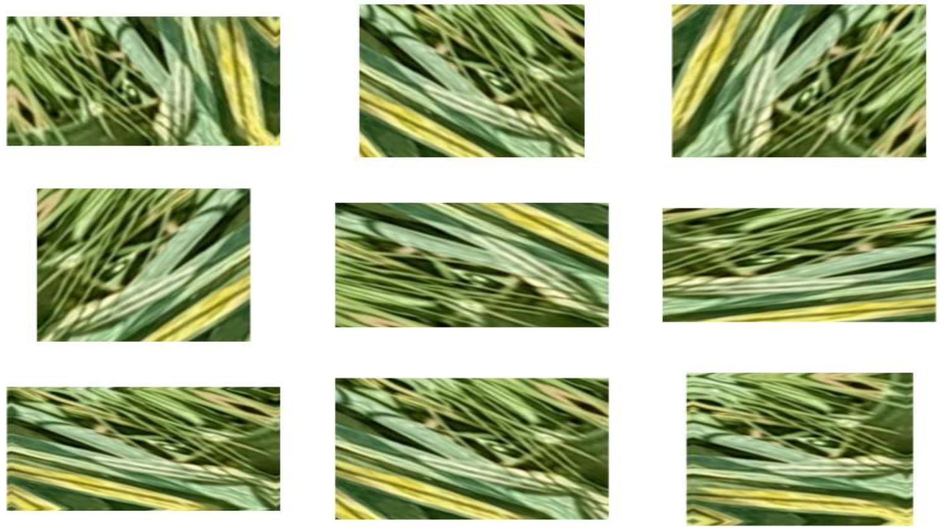


Figure 58. Enlarging Date with Data Augmentation Technique

Here’s another problem showed up after collecting data as there was some limitations such as the lack of resources and images of some specific diseases which led to **imbalanced data**, the problem with imbalanced data that it could be tricky, you may got a high accuracy overall but for internally the accuracy of the specific class which is likely to be the small, it’s accuracy will be bad. There are two common solutions for that, the first one is to augment the small class to balance the data but it’s not effective for generalization purposes.

What really happens is the weights of the small class is low compared with large class weights, then a great solution showed up which is **class weighting** by enlarging the classes of the weights of small class by a certain factor.

For class weighting we make the model pays more attention to minority class by increasing its weights by some ratio not explicitly identified but there some ways to deal with these class weighs, in our model we used this form:

$$class\ weight = \frac{1}{examples_{class}} * \frac{examples_{total}}{number\ of\ class} - const \tag{1}$$

The most technique was used earlier was over and under Sampling (Random or Manual) (Not preferable), Under sampling is a technique which deletes examples from the majority class which may leads to lose information from the model, on the other hand Over sampling is the opposite which duplicates the minority class images. “Random over-sampling, a random set of copies of minority class examples is added to the data. This may increase the likelihood of overfitting, especially for higher over-sampling rates. Moreover, it may decrease the classifier performance and increase the computational effort” [15]. But they did not work well as they have some drawbacks such as information loss and increasing the risk of discarding valuable data. Finally, class weighting gives the best results.

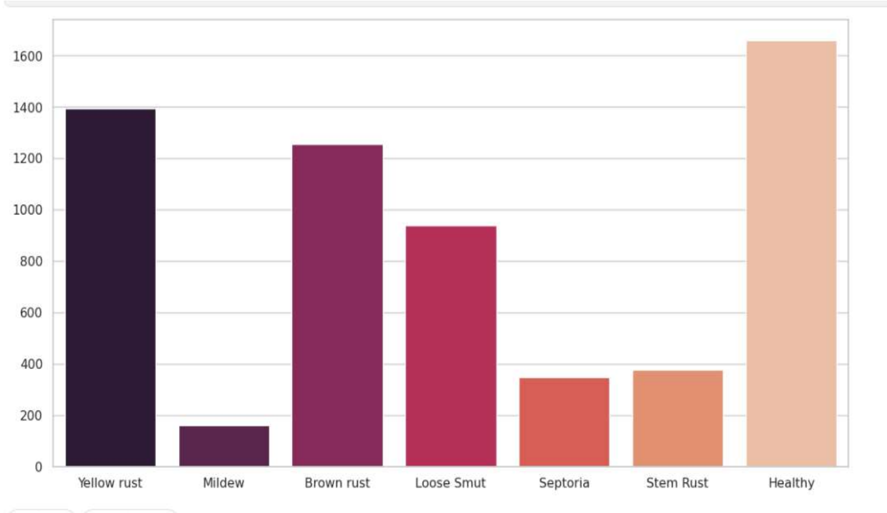


Figure 59. Imbalancing of Wheat Diseases Dataset

The idea is how to tune the hyperparameters you have for instance choosing the right optimizer for your model like Adam Algorithm or specifying the best layer from pretrained model to make trainable or deciding which loss function to use based on the nature of the model we are working on, all of that is very exhausting process but necessarily.

Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can doing well with noisy problems, after missing around here and there by different algorithms Adam is the choice for every model we trained already, and this might be the only property or hyperparameter won’t change with models.

Trying a mass of pretrained models such as ResNet, Inception, and more, but the best one with a noticeable change in error rate was ConvNeXt [16] began with the introduction of Vision Transformers (ViTs), which quickly superseded ConvNets as the state-of-the-art image classification model. The outcome of this exploration is a family of pure ConvNet models dubbed ConvNeXt. Constructed entirely from standard ConvNet modules.

Image Resizing also was an advantage to a certain large enough dimension affects the accuracy to a favorable level and more of hyperparameters tuned led us to the final results that would describe. As shown in Figure 60.

```
Epoch 1/10
38/38 - 207s - loss: 1.0445 - accuracy: 0.7111 - val_loss: 0.2131 - val_accuracy: 0.9282 - 207s/epoch - 5s/step
Epoch 2/10
38/38 - 174s - loss: 0.2066 - accuracy: 0.9357 - val_loss: 0.1277 - val_accuracy: 0.9574 - 174s/epoch - 5s/step
Epoch 3/10
38/38 - 175s - loss: 0.0946 - accuracy: 0.9697 - val_loss: 0.0921 - val_accuracy: 0.9716 - 175s/epoch - 5s/step
Epoch 4/10
38/38 - 172s - loss: 0.0524 - accuracy: 0.9852 - val_loss: 0.0932 - val_accuracy: 0.9683 - 172s/epoch - 5s/step
Epoch 5/10
38/38 - 171s - loss: 0.0296 - accuracy: 0.9925 - val_loss: 0.0868 - val_accuracy: 0.9724 - 171s/epoch - 4s/step
Epoch 6/10
38/38 - 171s - loss: 0.0162 - accuracy: 0.9962 - val_loss: 0.0729 - val_accuracy: 0.9766 - 171s/epoch - 5s/step
Epoch 7/10
38/38 - 215s - loss: 0.0108 - accuracy: 0.9983 - val_loss: 0.0781 - val_accuracy: 0.9766 - 215s/epoch - 6s/step
Epoch 8/10
38/38 - 176s - loss: 0.0066 - accuracy: 0.9994 - val_loss: 0.0667 - val_accuracy: 0.9808 - 176s/epoch - 5s/step
Epoch 9/10
38/38 - 176s - loss: 0.0043 - accuracy: 0.9998 - val_loss: 0.0601 - val_accuracy: 0.9816 - 176s/epoch - 5s/step
Epoch 10/10
38/38 - 177s - loss: 0.0030 - accuracy: 1.0000 - val_loss: 0.0628 - val_accuracy: 0.9816 - 177s/epoch - 5s/step
```

Figure 60. Validation and Training Accuracy and Loss with Epochs (epoch by epoch)

The model trained has the following characteristics:

$$Training\ Accuracy = 100\ \%, \quad Validation\ Accuracy = 98.2\ \%$$
$$Training\ Loss = 0.003, \quad Validation\ Loss = 0.0628 \tag{2}$$

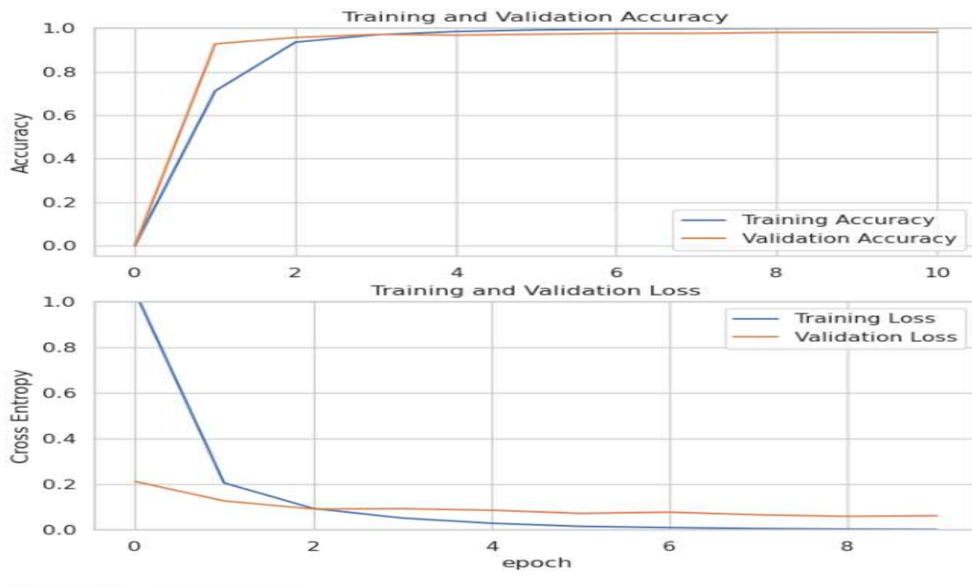


Figure 61. Loss and Accuracy with epochs graph

Training the model and getting results is important but testing it is critical to ensure the validation of training, Figure 62 shows some predicted test images embed with the correctness of the prediction, test images is important and completely different from validation dataset as they are misunderstanding all the way,

Validation dataset is helping the model to choose the best hyperparameter like train dataset helps with the weights themselves. So, we cannot refer to validation dataset as test dataset, test images are absolutely brand new to the model. From 25 samples we got only one wrong prediction.

Figure 62. Predicting and Validation Test Images

Actual Negative | True Negative (TN) | False Positive (FP) | Actual Positive | False Negative (FN) | True Positive (TP)

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \quad , \; Recall = \frac{TP}{(TP + FN)} \tag{4}$$

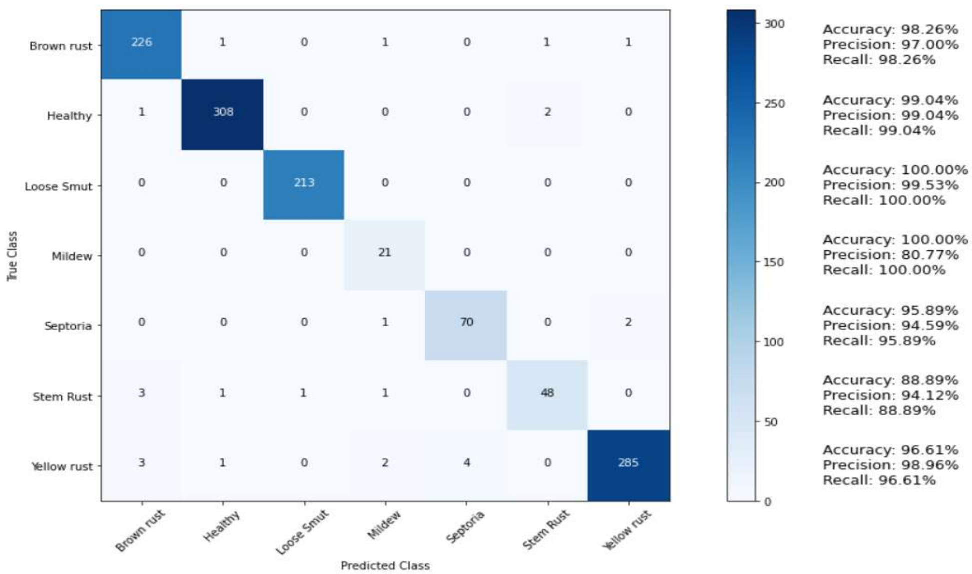


Figure 63. Confusion Matrix for Test Dataset (Each class characteristics embedded)

In summary, **accuracy** gives an overall measure of correctness, **precision** focuses on the proportion of correctly predicted positive instances, and **recall** emphasizes the proportion of actual positive instances correctly predicted by the model. And as we have imbalanced data we would consider recall and precision over accuracy as they give more meaningful insights, and the results are far better with **class weighting**.

There is a way not completely implemented has a great potential to affect the process of training which is Train Dataset processing with some tricky way to detect the most variations in the image and it would be greatly the region of interest specially in colorful images and defects problems. Then we could remove the rest to train over small pixels which could save time and computation resources.

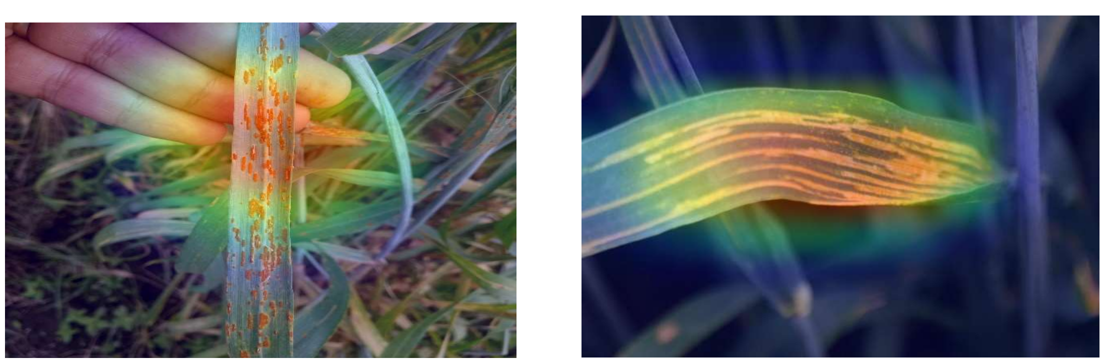


Figure 64. Image Processing for Train Dataset

6.6.2 Insects Detection Model

The insects and mites that attack wheat have **complex biological systems**, The insects and mites that attack wheat exhibit complex biological systems, diverse food habits, varied life cycles, powers of dispersal, and different ecological requirements for **survival**. While cultural practices have been manipulated to **control aphids on wheat**, the success of these methods has been mixed [10]. Therefore, effective pest detection in wheat is crucial for ensuring optimal yield and quality of this vital staple crop, traditional methods such as visual inspection and scouting have long been the primary approaches for pest monitoring in wheat fields. However, advancements in technology have opened promising avenues for more efficient and **accurate pest detection.** Emerging technologies such as remote sensing, **molecular diagnostics, sensor technologies**, and Unmanned Aerial Vehicles (UAVs) are revolutionizing the field of pest detection, providing real-time data, precise identification, and rapid surveillance capabilities.

One specific technology that has gained traction in object detection is the **You Only Look Once (YOLO) algorithm. YOLO is an object detection framework** that uses deep learning techniques to identify and locate objects within images or video frames. It operates by dividing the input image into a grid and predicting bounding boxes and **class probabilities for each grid cell**. This approach allows for real-time **object detection with high accuracy**. [17]

By incorporating YOLO or similar object detection algorithms into pest detection systems, it becomes possible to identify **and track insect pests in wheat fields.** High-resolution images captured by UAVs or other remote sensing devices can be analyzed using YOLO to detect and classify pests based on their appearance and movement patterns.

However, along with the potential benefits of these technologies, challenges need to be addressed. Cost-effectiveness, data interpretation, and integration with existing agricultural practices are key considerations. The future of pest detection in wheat cultivation lies in harnessing advancements in **machine learning, AI, and precision agriculture.** By continuously improving detection strategies and integrating them into crop management systems, we can safeguard wheat crops, mitigate pest-related risks, and **ensure food security for a growing global population**.

It is important to understand the nature of **the damage caused by pests** and mites in wheat fields, as well as the available methods for their control. Additionally, while this discussion focuses on pests specifically targeting wheat, it is worth considering the economic importance of some serious field pests of wheat in other parts of the world. By studying and addressing these challenges, researchers and agricultural practitioners can develop effective pest management strategies to protect wheat crops and **enhance agricultural productivity.**

During the object detection task, a total of 199 images were processed, each containing various instances of agricultural pests and insects. In total, there were **204 instances detected across the dataset.** The model showcased impressive performance across different evaluation metrics.

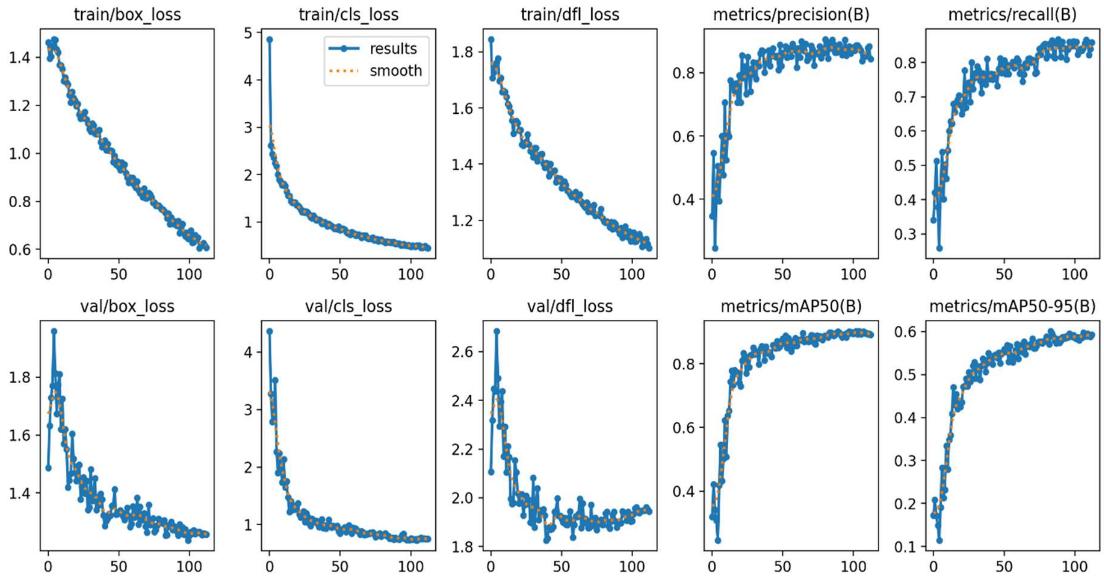


Figure 65. Performance metrics such as precision, recall, and mAP

At an IoU threshold of 0.50, the object detection model achieved an average precision (AP) of 0.875, indicating a high level of accuracy in localizing and classifying objects within bounding boxes. The recall (R) at the same **IoU threshold was 0.846**, indicating the model's ability to successfully identify a substantial number of instances present in the images.

The mean average precision (mAP) at an IoU threshold of 0.50 was **measured at 0.918, demonstrating the model's strong overall performance.** The mean average precision (mAP) across a range of IoU thresholds from 0.50 to 0.95 was 0.62, indicating the model's ability to maintain accuracy across varying levels of object overlap.

Analyzing the performance of specific classes, the legume blister beetle exhibited remarkable precision, with a value of 0.981, indicating a high proportion of correctly identified instances. The associated **AP for this class was 0.929,** showcasing the model's effectiveness in localizing and identifying legume blister beetles in the images.

Another notable class was the rice leaf roller, which **achieved perfect precision (1.0) and an AP of 0.945**. This indicates the model's exceptional performance in accurately identifying and localizing instances of rice leaf rollers, a significant pest in agriculture.

The **object detection model also demonstrated strong performance** for other classes, such as army worm, red spider, rice gall midge, rice leafhopper, rice water weevil, wheat phloeothrips, white backed plant hopper, and yellow rice borer, achieving respectable AP scores for each class.

In terms of speed, the **object detection process was efficient**, with preprocessing taking an average of 1.1ms per image, inference requiring 10.0ms per image, and postprocessing consuming 2.7ms per image. The absence of loss calculation time per image suggests a well-optimized and streamlined model implementation.

Figure 66.Validation Batch of Images

After applying the YOLO algorithm with accurate **predictions on a validation batch of images**, the resulting collection of images would typically contain various annotated bounding boxes around detected objects. These bounding boxes indicate the locations and sizes of objects that **the YOLO algorithm successfully recognized.**

Each image in the collection may have multiple bounding boxes, depending on the number of objects detected. The bounding boxes are typically represented by four coordinates: the top-left corner's (x, y) coordinate and **the bottom-right corner's (x, y) coordinate**. Additionally, the bounding boxes are often accompanied by class labels and confidence scores.

Overall, the YOLO algorithm's performance on the Val batch demonstrates its effectiveness in accurately identifying and localizing objects of interest within images. The high mAP score of 92%indicates a **strong ability to generalize and make reliable predictions across different object classes.**

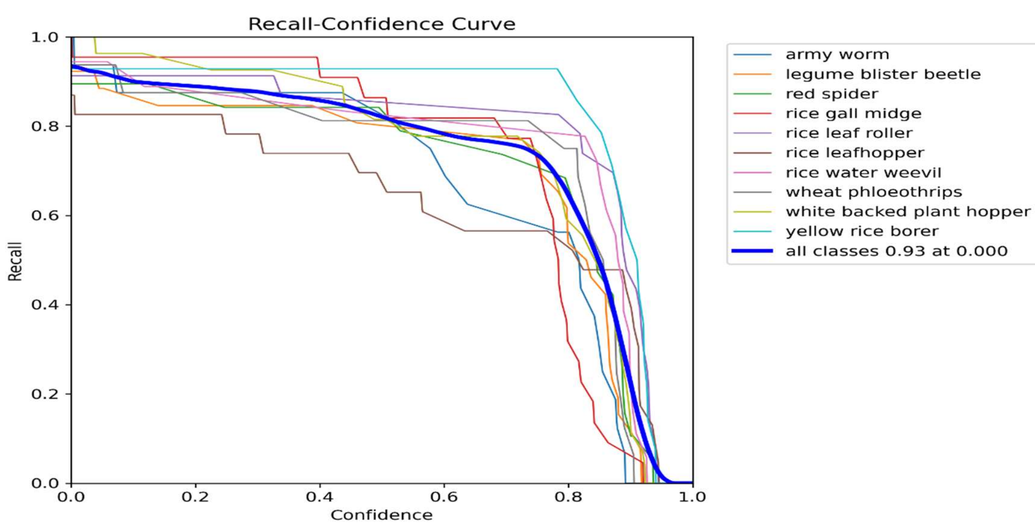


Figure 67.Recall curve

The Recall curve in YOLO **(You Only Look Once)** is a plot that illustrates the relationship between the detection threshold and the corresponding recall rate for object detection across different classes. Recall represents the ability of the **model to detect all instances** of a particular class within an image.

The **recall curve is a common evaluation metric** used in object detection tasks, including the YOLO algorithm. It measures the trade-off between the recall rate and the number of false positives generated by the model at different confidence thresholds.

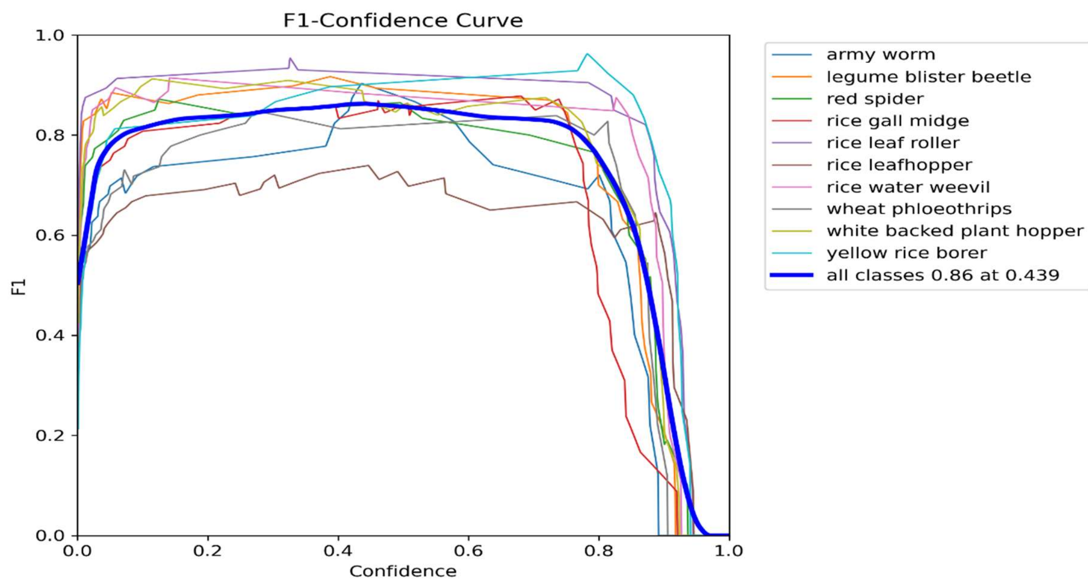


Figure 68 . F1-Confidence curve is a performance metric.

The F1-Confidence curve is a performance metric commonly used in object detection tasks, including the YOLO (You Only Look Once) algorithm. It provides insights into the **trade-off between precision and recall** at different confidence thresholds.

The F1-Confidence curve plots the F1 score against different confidence thresholds. By adjusting the confidence threshold, **you can control the number of detections made by the model.** A high confidence threshold will lead to fewer detections but higher precision, while a lower confidence threshold will result in more detections but potentially lower precision.

In the **context of YOLO**, the information you provided, "All classes **0.86 at 0.439**," suggests that for all object classes, when setting the **confidence threshold at 0.439**, the model achieves an **F1 score of 0.86**. This indicates a good overall balance between precision and recall for the given confidence threshold.

6.7 Deployment

The aim of this **mobile application** that utilizes image classification using WebView, where the backend is built with Streamlit, and the frontend is developed using Flutter. The application allows users to capture or select an image from their device's gallery and perform image classification on it.

6.7.1 Streamlit

The project consists **of two main components: the backend and the frontend**.

- **Backend:** The backend of the application is built using Streamlit, a powerful Python library for building interactive web applications. Streamlit provides an intuitive way to create user interfaces and visualize data. In this project, Streamlit is responsible for hosting a machine learning model that performs image classification. The backend utilizes a pre-trained image classification model, such as a Convolutional Neural Network (CNN), which has been trained on a large dataset of labeled images. Streamlit integrates seamlessly with machine learning frameworks like TensorFlow, allowing you to load the model and make predictions on new images.

- **Frontend: The frontend of the application** is developed using Flutter, a popular cross-platform framework for building native mobile applications. Flutter allows you to create beautiful and responsive user interfaces that work seamlessly on both Android and iOS devices. In the Flutter app, a **WebView component is integrated**, which acts as a bridge between the Flutter UI and the Streamlit backend. The WebView displays the Streamlit application hosted on a local server or a remote server. Users can interact with the WebView, capture an image using their device's camera, or select an image from the gallery. The selected image is then sent to the **Streamlit backend for image classification**.

Once the **classification results are received from the Streamlit backend,** the **Flutter app displays the predicted class or classes to the user** along with any additional information or visualizations provided by Streamlit.

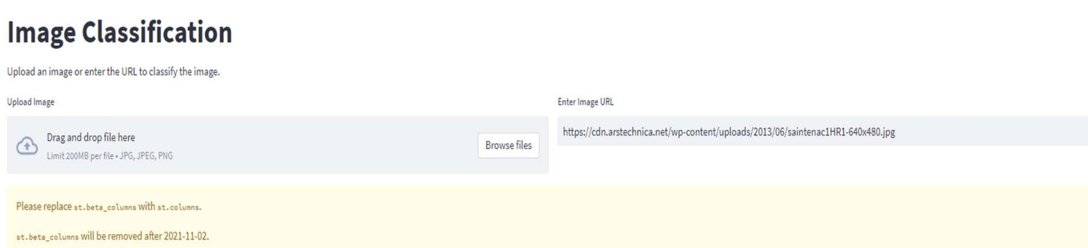


Figure 69. Front-End Application 1

Figure 70. Front-End Application 2

6.8 Pest Detection with TFLite

Deploying a TensorFlow Lite (TFLite) model, specifically YOLO v8, for pest detection in a Flutter app involves training and converting the model, preparing the deployment environment, integrating the TFLite model into Flutter, capturing or selecting images, preprocessing them, performing inference using the TFLite model, displaying the detection results, testing and validating the app, building and deploying the Flutter app, and maintaining and updating it over time. By combining the power of TFLite for efficient on-device inference and Flutter for cross-platform app development, this deployment enables real-time pest detection in a user-friendly and accessible mobile application.

6.9 Introduction

One of the most critical problems people faces is the lack of experience in agricultural matters. If a person wants to farm and does not have information or experience, it is possible that the person is an amateur or a beginner in agricultural matters, so he needs information and does a search. The search methods differ between searching on Google, and sometimes this is useless. Some sites contain wrong information, and the person does not know that it is wrong. Therefore, it is possible to harm the implant or the thing that needs to be done. Another way is to search on YouTube and follow the steps, and it may take a lot of time to verify all the steps in the video, and it may not find the thing you are looking for. It is possible to ask a person who is an expert in agricultural matters, whether a farmer or an agricultural engineer and this matter is also difficult because it is possible that he does not find someone to give him experience. Those reasons were sufficient for the existence of the chatbot, which will make it easier for the user to access the information he wants, because he collects the information accurately, verifies it, and puts it inside the application, in any one place that the user can access instead of searching in more than one place such as: Google, YouTube, Agricultural engineer so that having a Chatbot is important. Our Chatbot ("Agri Chatbot") is integrated with a mobile application("Crobit") and the second thing for our Chatbot is to introduce us and tell the user who we are and what we are exactly doing.

6.10 METHODS:

6.10.1 Datasets

There was a problem with the existence of the dataset because we did not find data related to agriculture in general, nor the plant that we work on in principle, which is the wheat plant in sites that are famous for the presence of data such as "Kaggle" and searching for the dataset took a long time and finally we did not find a suitable dataset so that we collected it manually to become suitable for us to worked on it and this took a long time also to collect it and the size of the dataset is about 1700 label.

The dataset is split into three parts for training, validation, and testing. The split is as follows:

70% for training
15% for validation
15% for testing

6.10.2 Training Set

The training set consists of 1190 examples. This set is used to train the model and optimize its parameters. The training set is large enough to training examples to capture the nuances of conversational interactions.

6.10.3 Validation Set

The validation set consists of 255 examples. The chatbot is trained on a separate training set, During the development of a chatbot, it is essential to evaluate its performance and ensure that it generates appropriate and relevant responses in a conversational context. The validation set plays a crucial role in this evaluation process. It consists of a collection of conversational pairs that are used to assess the chatbot's performance and make informed decisions about its improvement. It is used in:

- **Performance Evaluation:** The primary purpose of the validation set is to evaluate the performance of the chatbot. By comparing the chatbot's generated responses with the expected or desired responses in the validation set, you can measure its accuracy, relevance, and overall quality. This evaluation helps identify strengths and weaknesses, allowing you to make informed decisions about the chatbot's improvement.

- **Model Selection and Fine-tuning:** The validation set can be used to compare the performance of different chatbot models or configurations. By evaluating their outputs on the validation set, you can determine which model or configuration performs better and select the most suitable one for deployment. Additionally, the validation set can guide the fine-tuning process by providing feedback on specific areas that require improvement.
- **Error Analysis and Debugging:** Analyzing the chatbot's performance on the validation set can help identify common errors, patterns of incorrect responses, and areas of improvement. This error analysis provides insights into the limitations and challenges of the chatbot, allowing you to refine its algorithms, address specific issues, and enhance its overall performance.

6.10.4 Test Set

The test set consists of 255 examples, The test set is a crucial component in the development and evaluation of a chatbot. It serves as an independent dataset used to assess the chatbot's performance and generalization capabilities. The test set consists of a collection of input queries or prompts and their expected or desired responses. It is used in:

6.10.4.1 Performance Evaluation: It is to evaluate the chatbot's performance in real-world scenarios. By presenting the chatbot with unseen queries and assessing its responses, we can measure its ability to handle a wide range of user inputs and provide accurate and meaningful answers.

6.10.4.2 Generalization Assessment: The test set helps us understand how well the chatbot generalizes to new and unseen data. It assesses the chatbot's capability to handle variations in input phrasing, context, and user intents, ensuring that it can provide consistent and reliable responses across different scenarios.

6.10.4.3 Final Verification: The test set acts as a final verification step before deploying the chatbot to production or releasing it to end-users. It allows us to ensure that the chatbot meets the desired quality standards, performs as expected, and provides satisfactory responses across various test cases.

6.11 Data Preprocessing

Data preprocessing is an essential step in the machine learning workflow, including natural language processing tasks such as: Chatbots.

It helps with dimensionality reduction and normalization, enabling better model performance. It enhances the data's quality, relevance, and suitability, leading to more accurate and reliable predictions.

6.12 Models

Selecting the best model for a task is crucial for achieving optimal performance and obtaining accurate results so we tried several ones such as: T5 (Text-To-Text), GPT-2(Generative Pre-trained Transformer), Dialog flow.

6.11.1 Dialogflow:

Dialogflow, a platform by Google Cloud, provides a user-friendly interface and powerful tools for natural language processing tasks such as: building chatbots and conversational agents. It offers pre-built agents, intent recognition, and integration with various messaging platforms, making it a popular choice for chatbot development.

conversational experiences are hard to implement. Interpreting and processing natural language requires a very robust language parser. Dialogflow handles this for you, so you can provide a high-quality conversational end-user experience. Here is the key features and components of Dialogflow:

Intents An intent categorizes an end-user's intention for one conversation turn. For each agent, you define many intents, where your combined intents can handle a complete conversation. When an end-user writes or says something, referred to as an end-user expression, Dialogflow matches the end-user expression to the best intent in your agent. Matching an intent is also known as intent classification.

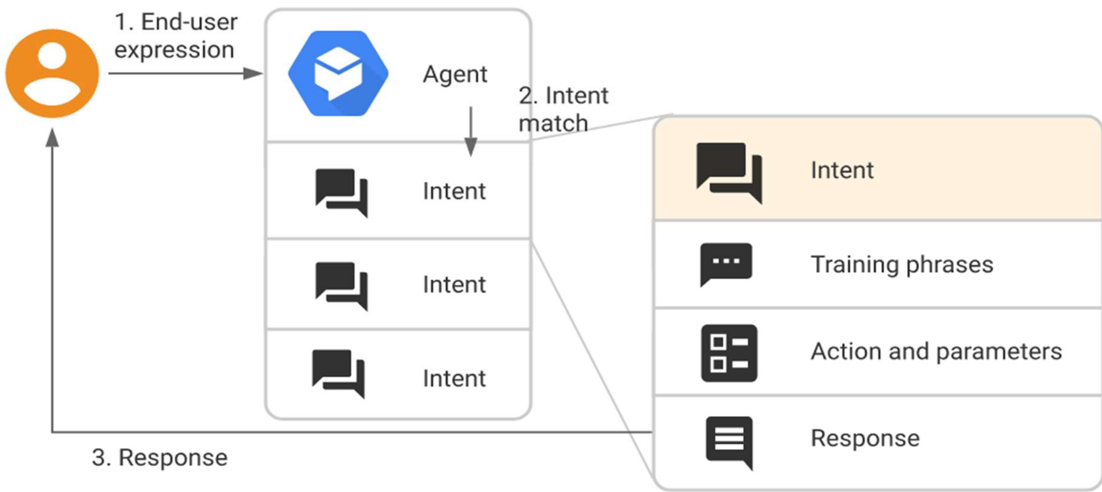
A basic intent contains the following:

6.11.1.1 Training phrases: These are example phrases for what end-users might say. When an end-user expression resembles one of these phrases, Dialogflow matches the intent. You don't have to define every possible example, because Dialog flow's built-in machine learning expands on your list with other, similar phrases.

6.11.1.2 Action: You can define an action for each intent. When an intent is matched, Dialogflow provides the action to your system, and you can use the action to trigger certain actions defined in your system**.**

6.11.1.3 Parameters: When an intent is matched at runtime, Dialogflow provides the extracted values from the end-user expression as parameters. Each parameter has a type, called the entity type, which dictates exactly how the data is extracted. Unlike raw end-user input, parameters are structured data that can easily be used to perform some logic or generate responses.

6.11.1.4 Responses: You define text, speech, or visual responses to return to the end-user. These may provide the end-user with answers, ask the end-user for more information, or terminate the conversation.



6.11.1.5 Entities

Each intent parameter has a type, called the entity type, which dictates exactly how data from an end-user expression is extracted.

Dialogflow provides predefined system entities that can match many common types of data, Entities are used to extract specific information or parameters from user inputs. They allow developers to define and identify relevant entities in user queries, such as dates, locations, or specific objects.

6.11.1.6 Contexts

Contexts help the chatbot maintain the context of a conversation by providing information about the current state or topic. They allow for more accurate understanding of user inputs and enable the chatbot to provide contextually relevant responses.

Dialogflow contexts are like natural language context. If a person says to you "they are orange", you need context to understand what "they" is referring to. Similarly, for Dialogflow to handle an end-user expression like that, it needs to be provided with context to correctly match an intent.

Using contexts, you can control the flow of a conversation. You can configure contexts for an intent by setting input and output contexts, which are identified by string names. When an intent is **matched, any configured *output contexts* for that intent become active. While any contexts** are active, Dialogflow is more likely to match intents that are configured with input contexts that correspond to the currently active contexts.

- **Fulfillment**: Dialogflow supports fulfillment, which allows developers to integrate external services or backend logic to handle complex tasks or retrieve dynamic information. Fulfillment can be used to perform actions, make API calls, or fetch data based on user queries.
- **Integration: Dialogflow** provides integration options for various channels and platforms, such as websites, Mobile apps, and more. It supports popular platforms like Facebook Messenger, Slack, and Google Assistant, allowing chatbots to be deployed on multiple channels.
    - **Machine Learning:** Dialogflow incorporates machine learning techniques to improve the understanding and accuracy of user inputs over time. It can learn from user interactions and adapt its responses based on training data.

6.11.1.7 Accuracy

The accuracy of Dialogflow, like any other conversational AI platform, depends on several factors, including the quality of training data, the complexity of the conversational scenarios, and the effectiveness of the configuration and customization with us the accuracy was good and it was enough that response time is better than T5.

6.12 Transformers

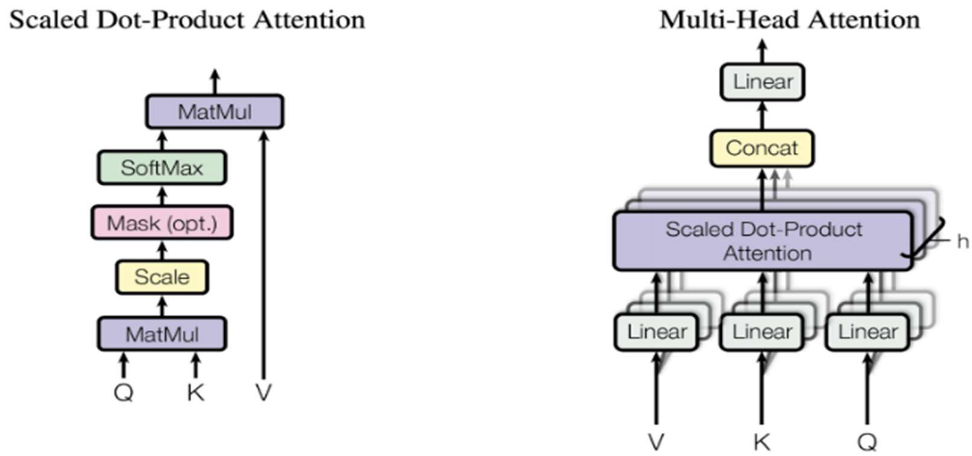
Transformers refer to a type of deep learning model architecture that has gained significant popularity in natural language processing (NLP) tasks. It appeared in the paper "Attention Is All **You Need" by Vaswani et al. in 2017. Transformers are designed to handle sequential data, such as sentences or text, and excel at capturing contextual relationships and dependencies. The key innovation of transformers i**s the self-attention mechanism, which allows the model to weigh the importance of different words or tokens in a sequence when making predictions or generating output.

This self-attention mechanism enables transformers to capture long-range dependencies and contextual information effectively. Transformers have revolutionized NLP tasks by achieving state-of-the-art results in various applications, including machine translation, sentiment analysis, question answering, text summarization, Chatbots and more. One of the most well-known transformer models is the **T5, which has been influential in advancing NLP research** and applications.

The transformer architecture has been the basis for various influential models, such as T5 (Text-To-Text Transfer Transformer), **BERT** (Bidirectional Encoder Representations from Transformers) and **GPT** (Generative Pre-trained Transformer).

For the Chatbot **T5 was the best one of the transformer models,** T5 has a **large-scale transformer architecture with multiple encoder and decoder layers**. It employs the self-attention mechanism to capture contextual dependencies and relationships within the input sequences. By training T5 on a vast corpus of text data and using different pre-training objectives, it learns to generate high-quality responses for a wide range of NLP tasks. Yes, there are different variants or versions of T5 that have been developed based on specific objectives or enhancements such as: **T5 Base, T5.1.0, T511B, T5 XL, T5 Large, T5 Small.**

Sample preprocessed conversation pair



Attention architecture diagrams from Attention is All You Need

Figure 71. Attention architecture diagram from Attention is All You Need

6.12.1 Attention

Like many sequence-to-sequence models, Transformer also consist of encoder and decoder. However, instead of recurrent or convolution layers, Transformer uses multi-head attention layers, which consist of multiple scaled dot-product attention.

6.12.2 Multi-head attention

It allows the model to capture different types of information and learn more nuanced representations. Multi-head attention is a mechanism used in transformer-based models to process and capture multiple aspects of input sequences simultaneously. It enables the model to attend to different positions within the input sequence to extract different types of information. This mechanism involves splitting the input representation into multiple heads and performing attention operations independently on each head. Each head has its own weight matrices, allowing it to focus on different relationships and patterns.
By employing multi-head attention, the model can effectively capture various types of dependencies and relationships within the input data. It enables the model to attend to both local and global contexts, learn long-range dependencies, and model complex interactions. The outputs from different heads are then combined, typically through concatenation or linear transformations, to provide a comprehensive representation of the input sequence.
Multi-head attention consists of four parts:

- Linear layers and split into heads.
- Scaled dot-product attention.
- Concatenation of heads.
- Final linear layer.

Each multi-head attention block takes a dictionary as input, which consists of query, key and value.

6.12.3 Scaled dot product attention.

The scaled dot-product attention function takes three inputs: Q (query), K (key), V (value). The equation used to calculate the attention weights is:

$$Attention(Q,K,V) = softmax_k\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

As the SoftMax normalization being applied on the key, its values decide the amount of importance given to the query. The output represents the multiplication of the attention weights and value. This ensures that the words we want to focus on are kept as is and the irrelevant words are flushed out.

For making our chatbot we used one of Dialogflow, GPT-2 and T5 Small and we are going to talk about each one and which one we used permanently in our application.

Transformer architecture diagram from **Attention is All You Need**

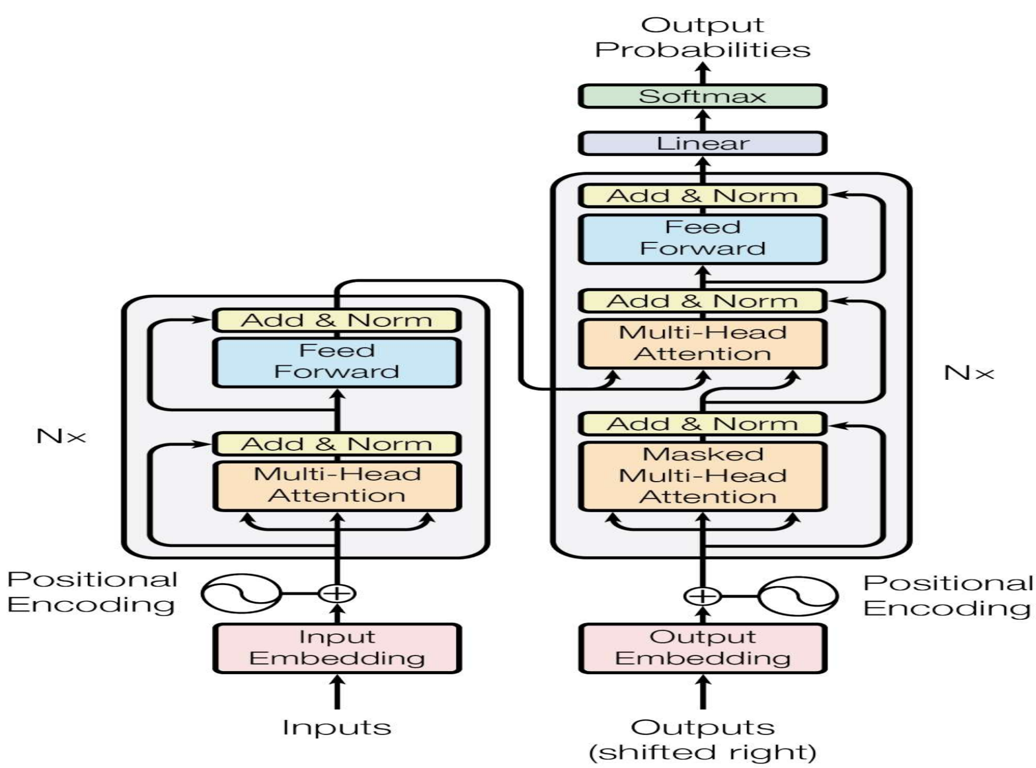


Figure 72. The Transformer - Model Architecture

Transformer uses stacked multi-head attention and dense layers for both the encoder and decoder. The encoder maps an input sequence of symbol representations to a sequence of continuous representations. Then the decoder takes the continuous representation and generates an output sequence of symbols one element at a time.

6.12.4 Positional Encoding

Since Transformer doesn't contain any recurrence or convolution, positional encoding is added to give the model some information about the relative Position of the words in the sentence.

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}})$$

Figure 73. The formula for calculating the positional encoding.

The positional encoding vector is added to the embedding vector. Embeddings represent a token in a d-dimensional space where tokens with similar meaning will be closer to each other. But the embeddings do not encode the relative position of words in a sentence. So after adding the positional encoding, words will be closer to each other based on the similarity of their meaning and their position in the sentence, in the d-dimensional space. To learn more about Positional Encoding.

Transformer consists of the encoder, decoder and a final linear layer. The output of the decoder is the input to the linear layer and its output is returned.

6.12.5 Comparing between T5&GPT-2:

As we mentioned before there are many transformer models to ensure that T5 is best one for us we use GPT-2 and when we compare the resulting accuracy of the two was different, the accuracy of GPT-2 was less than the accuracy of the T5 so that we used T5.

6.12.6 T5(Text- To-Text Transfer Transformer)

It is known that T5 is good with text-to-text sentences, so we use it with its version T5 base because it was the best one with our data (xlsx file), T5 base refers to a large variant of the T5 model. It consists of millions of parameters that define the architecture and behavior of the model the exact number of parameters and the architecture details may vary depending on the specific implementation or research context, The T5 Base model has a moderate model size, making it suitable for many practical applications. While it may not have as many parameters as larger variants like T5 Large or T5 3B, T5 Base still exhibits strong performance across different NLP tasks. It strikes a balance between computational efficiency and model capacity, making it a popular choice for many researchers and practitioners. T5 base can still be effective in handling a range of natural language processing (NLP) tasks and generating meaningful responses T5 base allows for faster training and inference times, making it more practical for applications where speed is a priority or where real-time interaction is required.

6.12.6.1 Accuracy:

At first there was a problem because of GPU so I had to put the batch size at first with 1, the resulting accuracy was not the best thing it was 85%, but after changing the batch size with 4 the accuracy increased.

6.12.6.2 Problem of T5:

While T5 is a powerful model for natural language processing (NLP) tasks, including chatbot development, it does have certain challenges and limitations. Here are some of the common problems associated with using T5 for chatbots that faced us:

6.12.6.3 Response Consistency:

T5, being a generative model, can sometimes produce inconsistent responses. It may generate different responses for similar inputs, leading to a lack of coherence or inconsistency in the conversation. Ensuring response consistency is an ongoing challenge in chatbot development.

6.12.6.4 Lack of Control:

T5 generates responses based on the input and its pre-training, but it may not always adhere to specific guidelines or constraints. It can produce answers that are factually incorrect, off-topic, or inappropriate. Controlling the output and ensuring accurate and contextually appropriate responses can be challenging with T5.

6.12.6.5 Response Time:

it was very slow, and this problem could be enough to use another thing because chatbot is real time Text-To-Text.

6.13 GPT-2 (Generative Pre-trained Transformer 2):

is a state-of-the-art language model developed by OpenAI. It is part of the GPT series, which includes earlier versions like GPT and the more recent version GPT-3. GPT-2 is known for its impressive ability to generate coherent and contextually relevant text when we used it, it was

good but less than expected in accuracy and the response time.

6.13.1 Accuracy:

The accuracy of using it was about 83%.

6.13.2 Problems of GPT-2:

GPT-2 is powerful, but has many problems such as:

6.13.3 Lack of Control:

GPT-2 tends to generate responses based on the patterns it learned during training, without strict adherence to specific guidelines or constraints This lack of control can result in generating responses that are factually incorrect, off-topic, or inappropriate in certain contexts. Ensuring control over the generated output can be challenging.

6.13.4 Response Coherence:

Although GPT-2 generally generates coherent and contextually relevant text, it may occasionally produce responses that lack coherence or fail to provide meaningful answers to specific queries. It can sometimes generate responses that seem plausible but lack substantive information or fail to address the user's intent effectively.

6.13.5 Inference and Comprehension Errors:

GPT-2 may occasionally make inference errors or misunderstand certain inputs, leading incorrect or nonsensical responses. It may struggle with

disambiguation or handling complex queries that require deep understanding or reasoning.

For us T5 was good than GPT-2 because of many things like: the accuracy and the spread of real-time response.

6.13.6 Comparing between Dialogflow & T5

Dialog flow is more accurate than T5 so at the end we use dialog flow because Chatbot is real time Text-To-Text so that response time is very important In addition to the accurate response that it retrieves based on the similarity between the text that the user enter and the text inside it.

6.14 Deployment:

Finally, we deployed the chatbot within the Mobile application "Crobit" that is specific for agriculture to provide people with the information or experience that they want to do the agricultural thing they want and empower farmers on-the-go with a powerful combination of mobile technology and AI-driven chatbot capabilities tailored specifically for the agriculture industry and make them know us from our chatbot."

6.15 Introduction

Remote sensing techniques using satellite data can provide valuable information for agriculture monitoring, allowing farmers and agricultural managers to make informed decisions about crop management, irrigation, and other agricultural practices**Invalid source specified.**. NDVI, NDMI, and SAVI are vegetation indexes commonly used in agriculture to monitor crop health and sparsity, as well as to estimate vegetation water content. Real-time data analysis using these indexes can provide valuable insights for improving crop productivity and sustainability, leading to more efficient use of resources and informed decision-making.

6.16 Working of Remote Sensing in Agriculture

The sun emits energy in the form of electromagnetic radiation, which includes visible light, ultraviolet radiation, and other forms of radiation that are not visible to the human eye. This energy travels through space and reaches the Earth, where it can be used for various purposes, including agriculture.

In agriculture, electromagnetic radiation is used for remote sensing, which is a technique that involves using sensors to collect information about plants without physically touching them. The electromagnetic spectrum is divided into different regions based on the wavelength of the radiation. The regions that are most useful for agricultural remote sensing are the visible and near-infrared regions of the spectrum, which have wavelengths between 400 and 2200 nanometers.

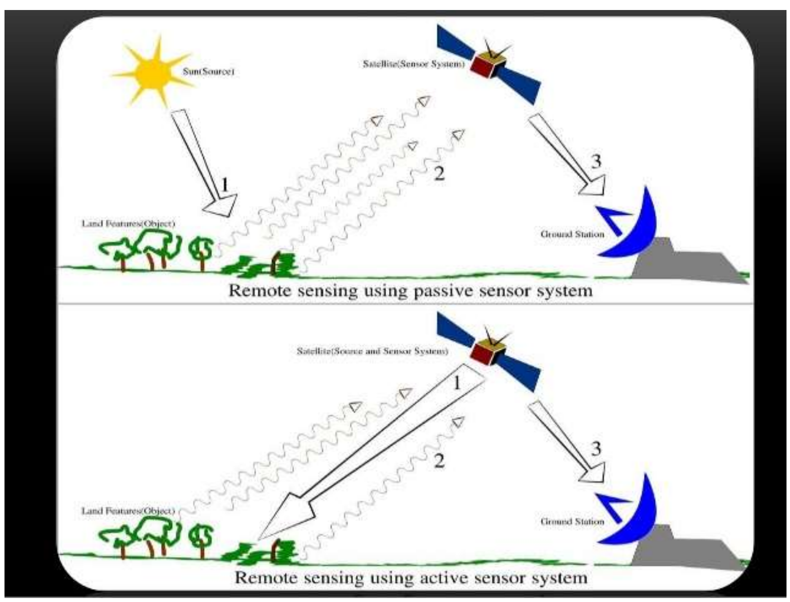


Figure 74. How remote sensing works in agriculture

When electromagnetic radiation from the sun reaches a plant, it interacts with the plant in different ways depending on the wavelength of the radiation and the properties of the plant. Some of the radiation is reflected back into the atmosphere, some is absorbed by the plant, and some is transmitted through the plant.

The amount of radiation that is reflected, absorbed, and transmitted by a plant depends on its optical properties, which are determined by factors such as its pigments, moisture

content, and leaf structure. These optical properties create a unique spectral signature for each plant species, which can be used to identify the plant and gather information about its properties.

Remote sensing sensors can detect the radiation that is reflected, absorbed, and transmitted by plants and use this information to create spectral signatures. By analyzing these spectral signatures, researchers can determine important information about the plant, such as its health, growth rate, and moisture level.

6.17 Satellite Imagery Acquisition

Remote sensing is the process of acquiring information about the Earth's surface using sensors mounted on satellites orbiting the planet. This information often comes in the form of images, These images are processed and analyzed to extract information about the Earth's environment, and the results are visualized and interpreted to communicate the findings to others.

In our project, we acquire satellite images through the Google Earth Engine (GEE) API.

6.17.1 Google Earth Engine (GEE)

Google Earth Engine (GEE) is a cloud-based platform for analyzing and processing geospatial data**Invalid source specified.**. It provides a powerful set of tools for geospatial analysis, including raster and vector data processing, machine learning algorithms, and data visualization. The GEE API is a programming interface that allows developers to access and use GEE's functionality in their own applications. Prior to using the Earth Engine Python client library, we need to authenticate (verify our identity) and use the resultant credentials to initialize the Python client. Then, we need to initialize the google earth environment, this initialization step verifies that valid credentials have been created and populates the Python client library with methods that the backend server supports.

After authenticating and initializing the environment, we need to define and process the agricultural land location (location of interest).

6.17.2 Defining the Agricultural Land Location

The first step in processing agricultural land location is to define the area of interest. In our project, this step is done by accessing our database which contains information about the agricultural land. Once we have accessed the database, we can retrieve the location data as an array of points, each containing a lat, lon pair. Using the array of points retrieved from the database, you can define the boundaries of the agricultural land.

The GEE API requires the location to be in a lon, lat format instead of being in a lat, lon format. So, the first step we do after getting the location from the database is converting each location point in the array from a lat, lon pair to be a lon, lat pair. Then we convert this location to a Geometry type to be used for acquiring satellite images. We create a geometry with ee.Geometry which is a GeoJSON object where the type member’s value can be Point, Polygon, LineString ... We use the Polygon type to be more flexible with different lands’ shapes.

Figure 75. Define the agricultural land location

Then we use this Polygon location to filter the images that we acquire from the satellite. There are many different satellites offered by the google earth engine that we can use such as Landsat, Sentinel-2, and MODIS. In our project, we use the Sentinel-2 satellites because of their high-resolution multispectral imager with 13 spectral bands.

6.17.3 Sentinel-2

Sentinel-2 is an Earth observation mission from the Copernicus Program that systematically acquires optical imagery at high spatial resolution (10 m to 60 m) over land and coastal waters. The mission is currently a constellation with two satellites, Sentinel-2A and Sentinel-2B; a third satellite, Sentinel-2C, is currently undergoing testing in preparation for launch in 2024. The mission supports a broad range of services and applications such as agricultural monitoring, emergencies management, land cover classification, and water quality. We use it in our project for agriculture monitoring**Invalid source specified.**.

Sentinel-2 has been developed and is being operated by the European Space Agency (ESA).

The Sentinel-2 mission has the following key characteristics:

- Multi-spectral data with 13 bands in the visible, near infrared, and short wave infrared part of the spectrum.
- Systematic global coverage of land surfaces from 56° S to 84° N, coastal waters, and all of the Mediterranean Sea.
- Spatial resolution of 10 m, 20 m, and 60 m.
- Revisiting every 10 days under the same viewing angles.
- Free and open data policy

To achieve frequent revisits and high mission availability, two identical Sentinel-2 satellites (Sentinel-2A and Sentinel-2B) operate together. The satellites are phased 180 degrees from each other on the same orbit. This allows for what would be a 10day revisit cycle to be completed in 5 days.

Figure 76. SENTINEL-2

6.17.4 Acquiring a Satellite Image for the Specified Location

After we have the location of interest in a Geometry type, then we can acquire data from the Sentinel-2 satellites for this location. We use the Earth Engine Python API to search for Sentinel-2 images for the location we are interested in using an Image Collection with a unique ID for each satellite and filtering the collection by location and date range. We acquire an image for this location with the current date to process this image and calculate the vegetation indexes we are interested in.

6.18 Calculating Indexes

One of the most powerful tools in remote sensing for agriculture is the use of vegetation indices, such as the Normalized Difference Vegetation Index (NDVI), Normalized Difference Moisture Index (NDMI), and Soil-Adjusted Vegetation Index (SAVI). In our project, we calculate these indexes using remote sensing data to provide valuable information on vegetation health, moisture content, and other important parameters.

Some key benefits of using vegetation indexes in agriculture:

1. **Monitoring crop health:** Vegetation indices such as NDVI can provide an accurate measure of vegetation health, which is a key indicator of crop performance. By monitoring changes in NDVI over time, farmers and researchers can identify areas of the crop that are under stress and take corrective actions to improve yield and quality.
2. **Assessing water stress:** Vegetation indices such as NDMI can provide an indication of water stress in crops. By monitoring changes in NDMI over time, farmers and researchers can identify areas of the crop that are suffering from drought or other water-related stress and adjust irrigation practices accordingly.
3. **Detecting nutrient deficiencies:** Vegetation indices such as SAVI can provide an indication of nutrient deficiencies in crops. By monitoring changes in SAVI over time, farmers and

researchers can identify areas of the crop that are lacking in essential nutrients and take corrective actions to improve crop health and yield.

6.18.1 Normalized Difference Vegetation Index (NDVI)

The Normalized Difference Vegetation Index (NDVI) is a numerical indicator that uses the red and near-infrared spectral bands. NDVI is highly associated with vegetation content. High NDVI values correspond to areas that reflect more in the near-infrared spectrum. Higher reflectance in the near infrared corresponds to denser and healthier vegetation**Invalid source specified.**.

The formula of **NDVI** = $\frac{\boldsymbol{NIR - Red}}{\boldsymbol{NIR + Red}}$

In Sentinel-2,

NIR corresponds to ➔ Band 8

Red corresponds to ➔ Band 4

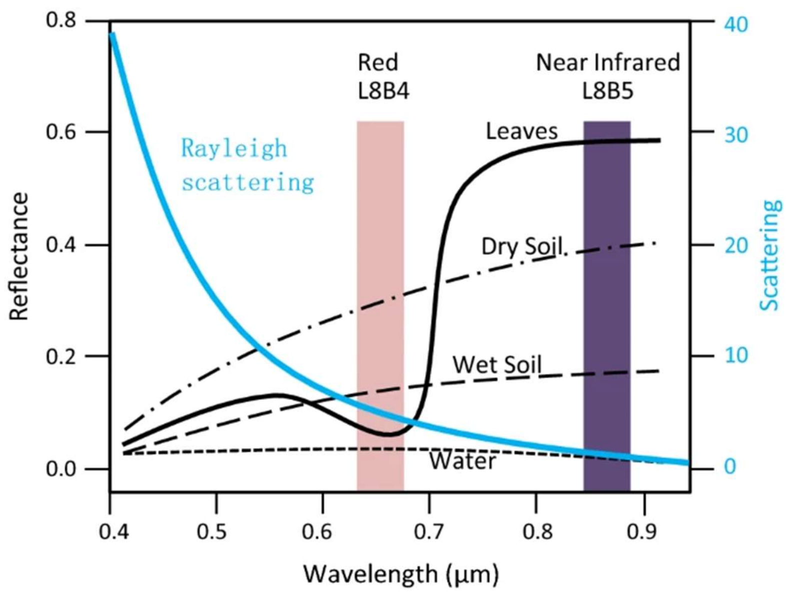


Figure 77. spectral response curves and their intersections with Red and NIR as well as the wavelength-dependent scattering

The NIR band is sensitive to the amount of vegetation present, while the red band is sensitive to the amount of chlorophyll in the vegetation. Thus, the NDVI value is an indication of the amount and vigor of green vegetation in each area.

NDVI values range from -1 to +1, with negative values indicating non-vegetated surfaces, such as water or barren land, and positive values indicating vegetated surfaces. The higher the NDVI value, the greener and healthier the vegetation is.

6.18.2 Normalized Difference Moisture Index (NDMI)

The Normalized Difference Moisture Index (NDMI) is a vegetation index that is used to estimate the amount of moisture in vegetation and soil. NDMI is calculated by taking the difference between the shortwave infrared (SWIR) and NIR band reflectance values and dividing by their sum**Invalid source specified.**.

The formula of **NDMI** = $\frac{\boldsymbol{NIR - SWIR}}{\boldsymbol{NIR + SWIR}}$

In Sentinel-2,

NIR corresponds to ➔ Band 8

SWIR corresponds to ➔ Band 11

The NIR band is sensitive to the amount of vegetation present, while the SWIR band is sensitive to the amount of water present in vegetation and soil. Thus, the NDMI value is an indication of the amount of water in vegetation and soil.

NDMI values range from -1 to +1, with negative values indicating dry surfaces, such as bare soil, and positive values indicating surfaces with high moisture content, such as vegetation cover. The higher the NDMI value, the more moisture there is in the vegetation and soil.

6.18.3 Soil-Adjusted Vegetation Index (SAVI)

The Soil-Adjusted Vegetation Index (SAVI) is a vegetation index that is used to estimate vegetation health and vigor while taking into account soil brightness in areas where vegetative cover is low. SAVI is calculated by taking the difference between the near-infrared (NIR) and red band reflectance values, multiplied by a vegetation adjustment factor, and dividing by the sum of the NIR and red band reflectance values, plus the vegetation adjustment factor**Invalid source specified.**.

The formula of **SAVI** = $\frac{\boldsymbol{NIR - RED}}{\boldsymbol{NIR + RED + L}} * \boldsymbol{(1 + L)}$

The vegetation adjustment factor, L, is used to adjust for the effects of soil brightness on the index. L ranges from 0 to 1, with higher values indicating brighter soils, but defined as 0.5 to accommodate most land cover types.

SAVI values range from **-1** to **+1**, with negative values indicating non-vegetated surfaces, such as water or barren land, and positive values indicating vegetated surfaces. The higher the SAVI value, the greener and healthier the vegetation is.

6.18.4 Adding the Indexes' Layers to the Basemap

After calculating the NDVI, NDMI, and SAVI indexes, we add their layers to the basemap with a legend for each index to clarify the classes of this index and their corresponding colors.

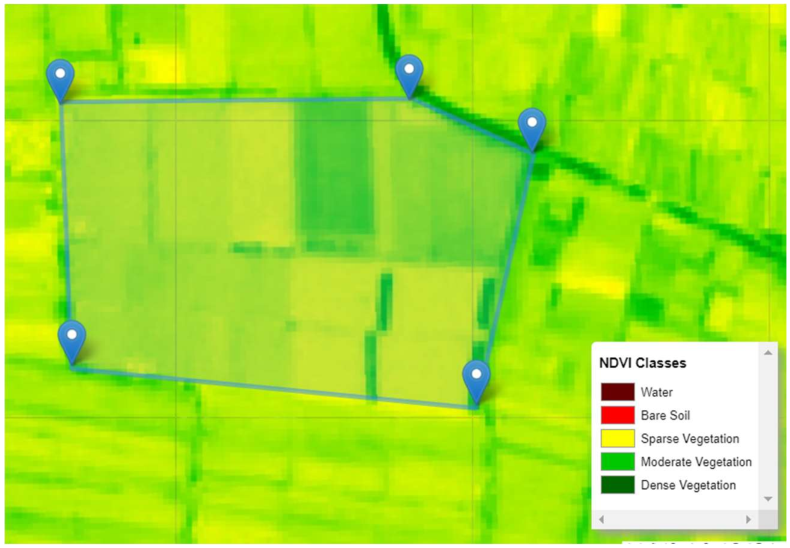


Figure 78. NDVI

Note: At the time of writing this documentation, this land is in the harvest season, so the dominant color is the light green/yellow.

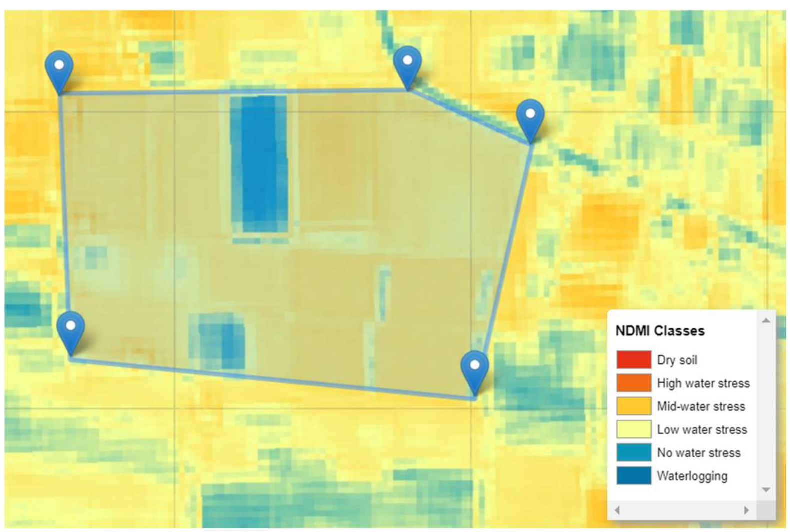


Figure 79. NDMI

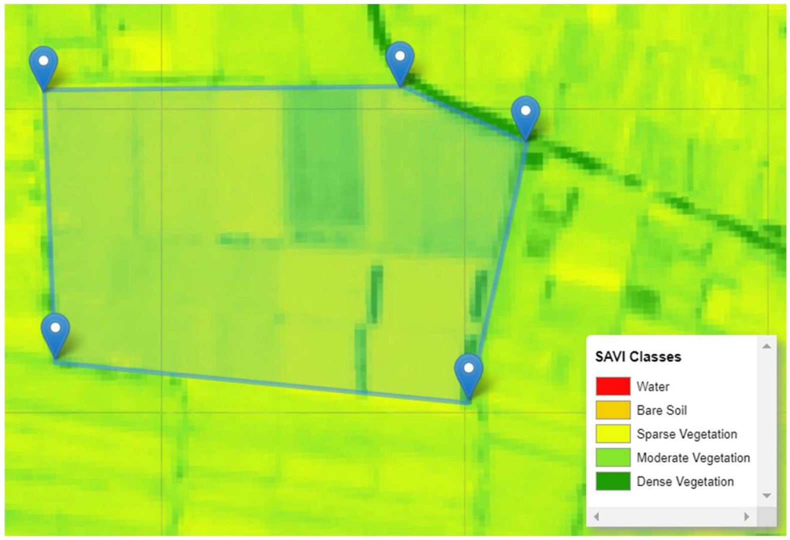


Figure 80. SAVI

Finally we convert this map which have all these indexes' layers, into an HTML file which can be easily viewed to the user as an interactive map and the user can switch easily between these layers.

References

[1] M. M. R. S. W. C.-Y. X. A. M. S. A. W. S. M. S. M. R. Nahina Islam, "Early Weed Detection Using Image Processing and Machine Learning Techniques in an Australian Chilli Farm," 25 April 2021. [Online]. Available: https://www.mdpi.com/2077-0472/11/5/387.

[2] J. W. Luis Perez, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," 13 December 2017. [Online]. Available: https://arxiv.org/abs/1712.04621.

[3] F. H. N. M. G. P. L. E. Z. Ivan Miguel Pires, "Homogeneous Data Normalization and Deep Learning: A Case Study in Human Activity Classification," 10 November 2020. [Online]. Available: https://www.mdpi.com/1999-5903/12/11/194.

[4] X. Z. S. R. J. S. Kaiming He, "Deep Residual Learning for Image Recognition," 10 December 2015. [Online]. Available: https://arxiv.org/abs/1512.03385.

[5] W. L. Y. J. P. S. S. R. D. A. D. E. V. V. A. R. Christian Szegedy, "Going Deeper with Convolutions," 17 September 2014. [Online]. Available: https://arxiv.org/abs/1409.4842.

[6] H. M. C.-Y. W. C. F. T. D. S. X. Zhuang Liu, "A ConvNet for the 2020s," 2 March 2022. [Online]. Available: https://arxiv.org/abs/2201.03545.

[7] H. K. K. Sajja Tulasi Krishna, "Deep-Learning-and-Transfer-Learning-Approaches-for-Image-Classification," February 2019. [Online]. Available: https://www.researchgate.net/profile/Hemantha-Kumar-Kalluri/publication/333666150_Deep_Learning_and_Transfer_Learning_Approaches_for_Image_Classification/links/5cfbeeb9a6fdccd1308d6aae/Deep-Learning-and-Transfer-Learning-Approaches-for-Image-Classificatio.

[8] Roboflow, "Github," 8 2022. [Online]. Available: https://github.com/roboflow/roboflow-100-benchmark.

[9] Y. Lecun, "Gradient-based learning applied to document recognition," *IEEE Xplore,* vol. 86, no. 11, 1998.

[10] Y. Lecun, "MNIST handwritten digit database," *ATT Labs,* vol. 2, 2010.

[11] J. Deng, "ImageNet: A large-scale hierarchical image database," *IEEE Xplore,* 2019.

[12] A. Krizhevsky, "ImageNet Classification with Deep Convolutional Neural Networks," 2015.

[13] J. Redmon, "You Only Look Once: Unified, Real-Time Object Detection," *arxiv,* 2015.

[14] Ultralytics, "ultralytics," [Online]. Available: https://ultralytics.com/.

[15] L. T. R. R. Paula Branco, "A Survey of Predictive Modelling under Imbalanced Distributions," *Cornell University ,* 2015.

[16] Z. Liu, "A ConvNet for the 2020s," *arxiv,* 2022.

[17] N. K. ,. J.-H. K. Chee Sun Won, "High-Speed Drone Detection Based On Yolo-V8," [Online]. Available: https://arxiv.org/abs/2004.10934.