# IOT Equipped and AI enabled Wireless Sensor Nodes for Precision Agriculture

Project Supervisor: Dr. Ihsan Ullah

Submitted By

Hamza Ishtiaq

CUI/FA19-EEE-028/ATD

Mobasher Ahmed

CUI/FA19-EEE-009/ATD

Malik Nabeel Tariq

CUI/FA19-EEE-008/ATD

BS Thesis

In

Electrical (Electronics) Engineering

COMSATS University Islamabad
Abbottabad Campus-Pakistan

Spring 2023

**COMSATS University Islamabad-Abbottabad Campus**

# IOT Equipped and AI enabled WSN nodes for Precision Agriculture

A Thesis Presented to

COMSATS University Islamabad-Abbottabad Campus

In partial fulfillment

of the requirement for the degree of

## BS Electrical (Electronics) Engineering

By

Hamza Ishtiaq

CUI/FA19-EEE-028/ATD

Mobasher Ahmed

CUI/FA19-EEE-009/ATD

Malik Nabeel Tariq

CUI/FA19-EEE-008/ATD

Spring 2023

# IOT Equipped and AI enabled WSN nodes for Precision Agriculture

---

An Under Graduate Thesis submitted to Electrical and Computer Engineering Department as partial fulfillment of the requirement for the award of Degree of Bachelor of Science in Electrical (Electronics) Engineering.

| Name | Registration Number |
|---|---|
| Hamza Ishtiaq | CUI/FA19-EEE-028/ATD |
| Mobasher Ahmed | CUI/FA19-EEE-009/ATD |
| Malik Nabeel Tariq | CUI/FA19-EEE-008/ATD |

## Supervisor

Dr. Ihsan Ullah
Associate professor, Electrical and Computer Engineering
Abbottabad Campus
COMSATS University Islamabad (CUI)
Abbottabad Campus
July 2023

# Final Approval

This thesis titled

# IOT Equipped and AI enabled WSN nodes for Precision Agriculture

By

*Hamza Ishtiaq*
*CUI/FA19-EEE-028/ATD*
*Mobasher Ahmed*
*CUI/FA19-EEE-009/ATD*
*Malik Nabeel Tariq*
*CUI/FA19-EEE-008/ATD*
Has been approved

For the COMSATS University Islamabad, Abbottabad Campus

Supervisor: ________________________________________
Dr. Ihsan Ullah, Associate Professor

Department of Electrical and Computer Engineering Engineering/CUI,
Abbottabad Campus

HOD: Owais
Dr. Owais, Associate Professor

Department of Electrical and Computer Engineering, CUI, Abbottabad
Campus

# Declaration

We Hamza Ishtiaq(CUI/FA19-EEE-028/ATD), Mobasher Ahmed(CUI/FA19-EEE-009/ATD), Malik Nabeel Tariq(CUI/FA19-EEE-008/ATD). We hereby declare that We have produced the work presented in this thesis, during the scheduled period of study. We also declare that we have not taken any material from any source except referred to wherever due that amount of plagiarism is within acceptable range. If a violation of HEC rules on research has occurred in this thesis, we shall be liable to punishable action under the plagiarism rules of the HEC.

Date: 06/01/2023

Signature of the student:

Hamza Ishtiaq
CUI/FA19-EEE028/ATD

Hamzaishtiaq

Mobasher Ahmed
CUI/FA19-EEE009/ATD

Mobasher Ahmed

Malik Nabeel Tariq
CUI/FA19-EEE008/ATD

Nabeel tariq

# Certificate

It is certified that Hamza Ishtiaq(CUI/FA19-EEE-028/ATD), Mobasher Ahmed(CUI/FA19-EEE-009/ATD), Malik Nabeel Tariq(CUI/FA19-EEE-008/ATD has carried out all the work related to this report under my supervision at the Department of Electrical and Computer engineering, COMSATS University Islamabad, Abbottabad Campus and the work fulfills the requirement for award of BS degree.

Date: 06/01/2023

Supervisor:

___________________________

Dr. Ihsan Ullah, Associate Professor
Department of Electrical and Computer Engineering

Head of Department:

___________________________

Dr. Owais, Associate Professor
Department of Electrical and Computer Engineering

# DEDICATION

First of all we dedicated this thesis to Almighty ALLAH and then thesis is dedicated to our parents, who have provided us with so much financial and moral support in order to meet all of our needs while we were developing our system, to our most honorable faculty, particularly our Supervisor "Dr. Ihsan Ullah" who has supported us at every stage, as well as to our subordinates and all those who believe in the fertility of education. This degree is dedicated to our parents, family, friends, and respected faculty members who have supported us throughout our life.

# ACKNOWLEGDGEMENTS

We would like to acknowledge and give our warmest thanks to our supervisor Dr. Ihsan Ullah who made this work possible. His guidance and advice carried us through all the stages of writing our thesis. We would also like to thanks our committee members for letting our defense be an enjoyable moment, and for brilliant comments and suggestions.

Finally, we would like to acknowledge and thank PEC for providing funding for this project under the initiative *"Final Year Design Projects – FYDP for Industrial Technical problems."*

**Hamza Ishtiaq (CUI/FA19-EEE-028/ATD)**

**Mobasher Ahmed (CUI/FA19-EEE-009/ATD)**

**Malik Nabeel Tariq (CUI/FA19-EEE-008/ATD)**

# ABSTRACT

## IOT Equipped and AI enabled WSN nodes for Precision Agriculture

The rapid advancements in Internet of Things (IoT) technology and Artificial Intelligence (AI) have opened up new possibilities for optimizing agricultural practices and increasing crop yields. This project focuses on the development of IoT-equipped and AI-enabled Wireless Sensor Network (WSN) nodes specifically designed for precision agriculture. Precision agriculture aims to enhance farming efficiency by precisely monitoring and managing various environmental parameters such as soil moisture, temperature, humidity, light intensity, and crop health. Traditional farming methods often lack the precision and real-time monitoring capabilities required for optimal crop growth. By leveraging IoT and AI, this project aims to address these limitations and provide farmers with actionable insights to make informed decisions. The IoT-equipped WSN nodes developed in this project will consist of low-power and cost-effective sensor modules that collect data from the agricultural field. These nodes will be interconnected through wireless communication, forming a network that enables seamless data exchange and aggregation. The collected data will be processed and analyzed using advanced AI algorithms to extract valuable information and provide real-time feedback to farmers. The proposed system offers several advantages over conventional farming techniques. By monitoring environmental parameters with high precision, farmers can optimize irrigation schedules, ensure proper fertilization, and detect early signs of plant diseases or pests. This proactive approach enables targeted interventions, minimizing resource wastage and maximizing crop productivity. The AI-enabled analysis of sensor data enables the development of predictive models and decision support systems. By leveraging historical data and machine learning algorithms, the system can provide personalized recommendations and insights tailored to specific crop types, local weather conditions, and soil characteristics. This empowers farmers to make data-driven decisions and optimize their agricultural practices for better outcomes.

# TABLE OF CONTENTS

**CHAPTER 1** ........................................ **1**

**INTRODUCTION** ........................................ **1**

1.1 INTRODUCTION ........................................ 2
1.2 MOTIVATION ........................................ 3

**CHAPTER 2** ........................................ **6**

**STUDY AND METHODS** ........................................ **6**

2.1 INTRODUCTION ........................................ 7
    2.1.1 Precision Agriculture (PA) ........................................ 8
    2.1.2 Wireless sensor network (WSN) ........................................ 8
    2.1.3 Internet of things (IOT) ........................................ 10
    2.1.4 Artificial Intelligence (AI) ........................................ 10
    2.1.5 Cloud computing ........................................ 10
    2.1.6 Similar WSN projects ........................................ 11
    2.1.7 WSN and communication technologies ........................................ 12
    2.1.8 Infield sensors ........................................ 13
    2.1.9 Advantages of energy management ............ **Error! Bookmark not defined.**
    2.1.10 Current monitoring ........................................ 14
2.2 INITIAL DESIGN: ........................................ **ERROR! BOOKMARK NOT DEFINED.**
    2.2.1 Interfacing Raspberry Pi ........................................ 16
    2.2.2 Interfacing ESP32 Using Arduino IDE ........................................ 18
    2.2.3 Blynk App for Data Storing ........................................ 20
    2.2.4 Integrating all Sensors and Creating a string on Blynk App ........................................ 22
    2.2.5 Modeling AI modules on Google Colab ........................................ 24
    2.2.6 Transfer datasets and have an accuracy ........................................ 27
    2.2.7 Deploying H5 file on Raspberry Pi: ........................................ 31
    2.2.8 Testing Deployed file by attaching camera to Raspberry Pi ........................................ 32

**CHAPTER 3 RESULTS** ........................................ **35**

**PROTOTYPE DEVELOPMENT AND EXPERIMENTAL TESTS** ........................................ **35**

3.1 INTRODUCTION: ........................................ 36
3.2 PROTOTYPE DEVELOPMENT AND ASSEMBLING: ........................................ 36
3.3 DESIGNED PROTOTYPE AND RESULTS: ........................................ 36

**CHAPTER 4 DISCUSSION** ........................................ **46**

**CHAPTER 5** ........................................ **48**

**REFERENCES** ........................................ **48**

# LIST OF FIGURES

Fig 2.1 WSN nodes General Structure…… ………………………….….………...11

Fig 2.2 Current monitoring of the project ………………………….….……...11

Fig 2.3 Flowchart…………………………………………….….…..…….........13

Fig 2.4 Interfacing Raspberry Pi….……………………….….…...….……….15

Fig 2.5 Selecting ESP-32 Board in Arduino IDE ……………….…..…………..13

Fig 2.6 Dashboard Blynk App…………… ……………………..….….…….13

Fig 2.7 Instances of Tomato leaf pictures of Plant Dataset……….……......….24

Fig 2.8 Under, Appropriate and Overfitting…………….……….….…......….26

Fig 2.9 Accuracy Graph……………..……………………………..….……..28

Fig 3.1 Prototype…………….…………………………………….….…..……..30

Fig 3.2 Working of Circuit…..…………………………………….….………….32

Fig 3.3 Values Displayed on LCD………..……………………….…….…………32

Fig 3.4 Values when no Voltage is supplied……………….….…….…….……..33

Fig 3.5 Experimented Values…….……………………………….…………….....34

# LIST OF TABLES

TABLE 2.1 SIMILAR WSN PROJECTS ........................................................................ 10
TABLE 2.3 CONNECTIVITY OPTIONS ........................................................................ 11
TABLE 2.3 IN FIELD SENSORS ........................................................................ 11

# ABBREVIATIONS

| | |
|---|---|
| IOT | Internet of things |
| AI | Artificial Intelligence |
| WSN | Wireless sensor network |
| PA | Precision Agriculture |
| GDP | Gross domestic product |
| ICT | Information and Communication Technological |
| MQTT | Message Queuing Telemetry Transport |

# Chapter 1

# Introduction

## 1.1 Introduction

The quick progressions in innovation have prepared for the joining of Internet of Things (IoT) and Man-made consciousness (simulated intelligence) in different fields, and one such encouraging application is accuracy horticulture. Accuracy farming expects to enhance the utilization of assets and work on horticultural efficiency by utilizing information driven dynamic strategies. In this specific situation, the use of IoT and simulated intelligence empowered Remote Sensor Organization (WSN) hubs holds huge potential.

The title of this last year project, "IoT and artificial intelligence empowered WSN Hubs for Accuracy Farming," centers around utilizing the capacities of IoT and simulated intelligence to improve the observing and the executives of rural practices. By consolidating these advances, ranchers and farming specialists can get ongoing and exact data about their harvests, soil conditions, and natural elements. This significant information can engage them to go with informed choices in regards to water system, preparation, bother control, and generally speaking harvest wellbeing.

IoT-empowered WSN hubs allude to little, remote gadgets furnished with sensors that can gather information connected with different farming boundaries, for example, temperature, dampness, soil dampness, pH levels, and supplement content. These hubs speak with one another and communicate information to a focal center or cloud stage, where the information is handled and dissected utilizing simulated intelligence calculations. Simulated intelligence calculations empower the distinguishing proof of examples, patterns, and oddities in the gathered information, empowering exact bits of knowledge and significant suggestions.

The combination of IoT and computer based intelligence in accuracy horticulture offers a few key benefits. Right off the bat, it empowers constant and remote observing of yields and ecological circumstances, taking out the requirement for manual reviews. This continuous checking takes into consideration early identification of issues like plant infections, water pressure, or supplement inadequacies, empowering convenient mediations and limiting yield misfortunes.

Also, the use of simulated intelligence calculations on the gathered information empowers prescient examination and choice emotionally supportive networks. By breaking down authentic information and taking into account ecological variables, artificial intelligence can give suggestions to advanced water system plans, exact supplement the board, and powerful vermin control techniques. This information driven approach upgrades crop yields as well as lessens the natural effect by limiting the utilization of assets and synthetics. Also, IoT and man-made intelligence empowered accuracy horticulture frameworks can be coordinated with existing homestead the executives programming, making an exhaustive and brought together stage. This reconciliation permits ranchers to get to and examine the gathered information, screen the situation with their yields, and get alarms and warnings on their cell phones or PCs.

Taking everything into account, the proposed project, "IoT and simulated intelligence empowered WSN Hubs for Accuracy Horticulture," investigates the capability of joining IoT and man-made intelligence advances to change agribusiness. The task plans to configuration, create, and send a framework that flawlessly coordinates remote sensor organizations, computer based intelligence calculations, and information examination to empower accuracy cultivating rehearses. By bridling the force of IoT and artificial intelligence, ranchers can pursue information driven choices, streamline asset use, and accomplish higher harvest yields while advancing economical and harmless to the ecosystem rural practices.

## 1.2 Motivation

The advancement in the technology will help out the farmers to increase in crop gain.

The new concepts in technology includes

I. IOT (Internet of things).
II. WSN (wireless sensor network).
III. AI (Artificial Intelligence).

The internet of things is the network in which real world objects are connected to each other which tends to form many embedded systems including fields such as electronics and sensors through which the data can be transferred and received reliably. A real world thing/object in Internet of Thing in terms of animal farming can be an animal with Bio chip transponder which when assigned an IP address and ability to reliable data transfer over the network can be helpful to the farmer. Also with the use of sensor, application on the mobile phones and the transfer of useful data generated by the system will make it easy to use. The system has wide area of applications like Open Farm, Greenhouse Farming. In Open Farming, irrigation, water level etc. can be managed with this system whereas in Greenhouse Farming, temperature control, moisture control is the applications of this system. The system is also useful in the field of Gardening.

## 1.3 Objectives

The term “precision agriculture” refers to a contemporary approach of applying new technologies utilizing sensors in order to optimize the agricultural cultivation processes. Precision agriculture-based system design principles are increasingly used in research projects and commercial products to provide solutions for crop status monitoring and pest control. Such system will save time and results in cost reduction by automation. Another significant feature that precision agriculture provides to farmers is the ability to prevent hazardous incidents and to proactively monitor their crops and the local environmental conditions. The effectiveness of precision agriculture is based on the analysis of accurate sets of measurements in soft real-time. Parameters such as the soil condition and humidity are aggregated and analyzed, in order to extract useful information that a farmer can use as a recommendation or guidance; or even to apply fully automated procedures to the crop cultivation process chain.

# Chapter 2

# Study and Methods

## 2.1 Introduction

Precision agriculture is an emerging field that combines advanced technologies to optimize farming practices and enhance crop yield. One of the key components of precision agriculture is the use of Internet of Things (IoT) and Artificial Intelligence (AI) enabled Wireless Sensor Network (WSN) nodes. These nodes play a crucial role in collecting and analyzing data from various sources within agricultural ecosystems, enabling farmers to make informed decisions and improve resource management. The integration of IoT and AI in precision agriculture has revolutionized traditional farming practices by providing real-time data on environmental conditions, crop growth, soil moisture, and other vital parameters. IoT-equipped WSN nodes act as sensory agents, gathering information from sensors placed strategically throughout the agricultural fields. The collected data is then transmitted wirelessly to a central system for analysis and decision-making. In recent years, advancements in IoT and AI technologies have facilitated the development of intelligent WSN nodes that are capable of autonomous decision-making, adaptive learning, and predictive analytics. These nodes are equipped with sophisticated sensors, communication modules, and AI algorithms, enabling them to monitor and control various aspects of agricultural operations. They can automatically adjust irrigation schedules, detect pest infestations, optimize fertilizer usage, and provide early warning systems for adverse weather conditions. The benefits of using IoT-equipped and AI-enabled WSN nodes in precision agriculture are manifold. Firstly, they enable farmers to monitor and manage their crops in real-time, eliminating the need for manual data collection and reducing human error. Secondly, by providing actionable insights and recommendations, these nodes help optimize resource allocation, resulting in improved efficiency and cost savings. Thirdly, they contribute to sustainable agriculture practices by minimizing water usage, reducing chemical inputs, and mitigating environmental risks. This literature aims to explore the various aspects of IoT-equipped and AI-enabled WSN nodes for precision agriculture. It delves into the underlying technologies, such as sensor networks, communication protocols, AI algorithms, and data analytics techniques. Additionally, it examines the potential applications of these nodes in different agricultural domains, including crop monitoring, livestock management, and greenhouse cultivation. Furthermore, the literature investigates the challenges and limitations associated with the deployment of IoT-enabled WSN nodes in agricultural settings and proposes possible solutions. By understanding the capabilities and limitations of IoT-equipped and AI-enabled WSN nodes in precision agriculture, we can pave the way for innovative solutions that address the evolving needs of modern farming.

### 2.1.1 Precision Agriculture (PA)

In Precision Agriculture information technology is used to improve the yield by precisely monitoring the various crop parameters. In result, better crop irrigation, suitable herbicides, pesticides and sowing techniques are achieved [2].

### 2.1.2 Wireless sensor network (WSN)

Wireless Sensor Network is a wireless network of separately placed sensor nodes which gather and share data with each other forming a network. Wireless Sensor Networks (WSN) emerged from advancements in the areas of micro electro-mechanical system (MEMS) technology, wireless communication, and digital electronics. WSNs devices are small in size, low cost, and require low power to work. The basic structure of WSN sensor nodes as identified by [3]. shown below see Fig.1.

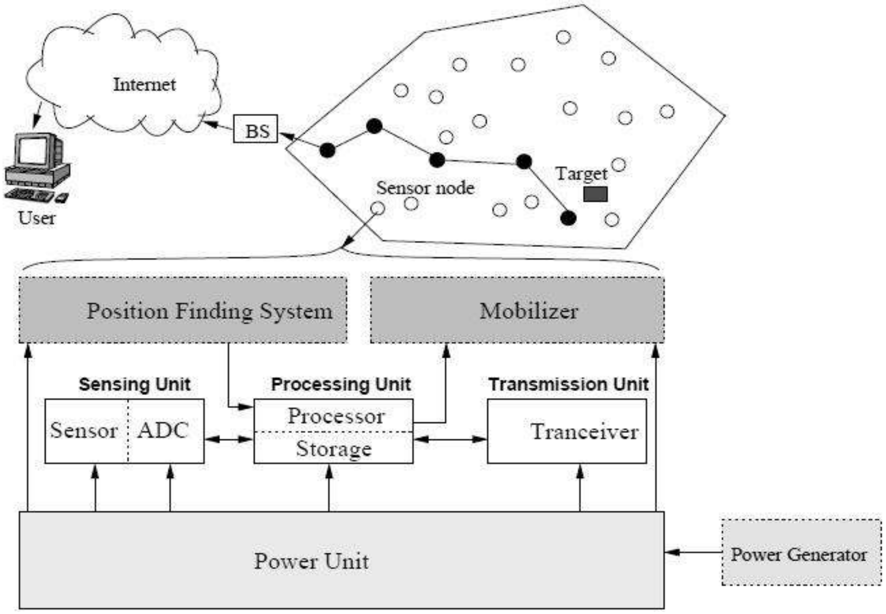


Figure 2.1: WSN sensor nodes general structure

According to [3]. There are four main components that make up a sensor node. The parts are namely: a sensing unit, a processing unit, a transmission unit and a power unit. Depending on the type of application a sensor node may have additional parts such as a position finding system, mobilize and a power generator. Sensing unit usually takes the burden of sensing and gathering sensor data and then passes the data to the processing unit. The processing unit receives the sensed data and processes it according to a set procedure or program. A transmission unit connects the sensor not with a network. The power unit supplies power required to run a sensor node. Akylidiz et al. [4] identified five areas of application of WSN. These are for military applications, environmental applications, health applications, home applications and other commercial applications in offices and buildings. Other studies conducted show that WSN technologies can be used in the area of agriculture as well. There is a growing demand for technological application in the developing world and expansion of telecommunication infrastructure is growing rapidly. The rapid growth in the field of telecommunication in the developing world is making farmer get access to

Information and Communication Technological (ICT) Infrastructures. The potential of applicability of WSN in the agricultural sector couple with the increasing growth of ICT in developing countries create an opportunity to explore the benefit of such application for farmers.

### 2.1.3 Internet of things (IOT)

The internet of things (IOT) is a network of globally connected devices and machines which communicate with each other. IOT is one of the latest technology trends that are gaining popularity among vast majority of industries. It has been forecast that till the year 2020 IOT based units will reach up to 26 billion. IOT applications include human to device and device to device interactions in reliable way. Main application areas are categorized as data and business analysis, monitoring and control, and collaboration and information sharing [5].

### 2.1.4 Artificial Intelligence (AI)

Now a day artificial intelligence is commonly used being used in several technologies. For real time health monitoring of crop like health is done by the help of Artificial intelligence. Camera is deployed in the fields to capture the real time condition of the crop and module is trained with datasets by comparing the real time data send by the camera. In this way farmers are updated with the current condition of their respective crops over the large area.

### 2.1.5 Cloud computing

IOT applications need space for data storage and huge data manipulations for real time decision making [6]. Cloud computing is a platform that provides solution for high data transfer and data storage in real time. Data related to the parameters responsible for crop growth are collected by placing sensor nodes in random locations. Solar panel with rechargeable battery is used to fulfill the power needs. These nodes are interconnected according the cluster tree topology. Wireless communication protocol. Each node consists of several sensors capable of measuring parameters such as soil moisture and humidity, Temperature, Air quality and pests. The data from these sensors is transmitted through a network of interconnected nodes to the server, which then uploads the data to a cloud.

### 2.1.6 Similar WSN projects

The table below contains some projects and researches that used WSN components to improve different aspects in agriculture. The table shows list of projects and researches, their description, sensor used, crops involved in the study or project and devices used to access the information. We carried out a thorough investigation of these projects and researches in order to get an idea of what has been done and what is lacking.

Table 1: WSN projects

| Name | Description | Research Prototype | Sensors used | Crops | Devices | Reference |
|---|---|---|---|---|---|---|
| An Enegyefficient WSN for farming | Using WSN for PA in an energy efficient way | ✓ | | | PC | **[7]** |
| Application of WSNs for Greenhouse parameter control in PA | Use of programmable system on chip tech. to control the parameter of | ✓ | Zig-Bee sensor network(temperature, pressure,light,humidity,CO2, wind speed, and wind direction) | | PC | **[8]** |

| | | | | | | |
|---|---|---|---|---|---|---|
| | greenhouse for PA | | | | | |
| Wireless Farming | Using WSN to monitor a farm field and assist farmer make a decision | ✓ | Soil moisture Soil temperature Humidity temperature | Onion and tomato | PC, Mobile | |
| Agro-sense | Automated fruit harvesting | ✓ | Soil PH/moisture/Temperature<br>Soil Electrical conductivity | Fruits | Automation | **[9]** |

### 2.1.7 WSN and communication technologies

A WSN is composed of several sensor nodes that have the capacity of sensing and gathering data. The sensor nodes can sense varying types of parameters and send it to a central gateway [10] . WSN sensor and processing boards have the capability of working with various communication technologies. WSN can be linked to external servers or services both with wires or wirelessly (see Table 2: Connectivity options). Amongst others some of connection options could be using Ethernet connection, Wi-Fi, Bluetooth, or GSM-GPRS. In our case we plan to use ESP32s Wi-Fi based development board connectivity option.

**Table 2: Connectivity options**

| **Communication Techniques** | **Ranges** |
|---|---|

|  |  |
|---|---|
| Bluetooth | 30m-100m |
| ESP32s Wifi module | 50m-200m |

### 2.1.8 Infield sensors

Field sensors are devices used to accumulate and screen different sorts of information from the outer climate. They assume a pivotal part in various enterprises, including natural checking, modern computerization, horticulture, medical care, and the sky is the limit from there. Field sensors are intended to identify actual properties, or different boundaries and afterward convert this data into an electrical sign or information that can be handled and broke down.

Here are a few normal sorts of field sensors:

**Temperature and Humidity Sensor:**
Temperature and humidity sensor work by measuring the capacitance or resistance of air samples. Most of these sensors utilize capacitive measurement to determine the amount of dampness in the air.

**Gas Sensor:**
Gas sensors distinguish the presence and convergence of explicit gases in the climate. These sensors are fundamental for observing air quality, recognizing poisonous gases, and guaranteeing safe working circumstances in modern settings.

**Soil Moisture Sensor:**
These sensors measure the dampness content in soil and are fundamental for accuracy horticulture and proficient water system frameworks.

**Light Sensor:**
Light sensors, also known as photodetectors or photodiodes, measure light force. They are utilized in programmed lighting frameworks, cameras, sunlight powered chargers, and light-delicate cautions.

**Table 3: In field sensors**

| Parameters | Sensors |
|---|---|
| Temperature/ Humidity | DHT11 |
| Air Quality | MQ 135 |
| Soil moisture | A0009 |
| Sunlight Sensor | UV- Light |

### 2.1.9 Current monitoring

The proposed and implemented methodologies in present in this .fig.2. where different sensor nodes and camera among these nodes are deployed for data collection. These nodes and camera are connected with each other according to cluster tree topology. Different sensors gather information related to temperature, humidity, air quality check, soil moisture and camera are directly connected to ESP32S Wi-Fi based development board, further ESP32S Wi-Fi based development board send data by the mean if Wi-Fi to Raspberry Pi 3 model B.

Wireless sensor node is powered by solar panel. And connected to ESP32S Wi-Fi based development board. Different environmental parameter values obtained from sensors and camera are fed to ESP32S Wi-Fi based development board.

The measurement from the sensors and camera is transmitted wirelessly in the form of frames to the Raspberry pi 3 model B. further artificial intelligence is also implemented in this project for the sake of pest detection, our module is trained with different kind of datasets the basic working is like the camera which is deployed in the field capture real time images and compare it with the dataset by which our module is trained.

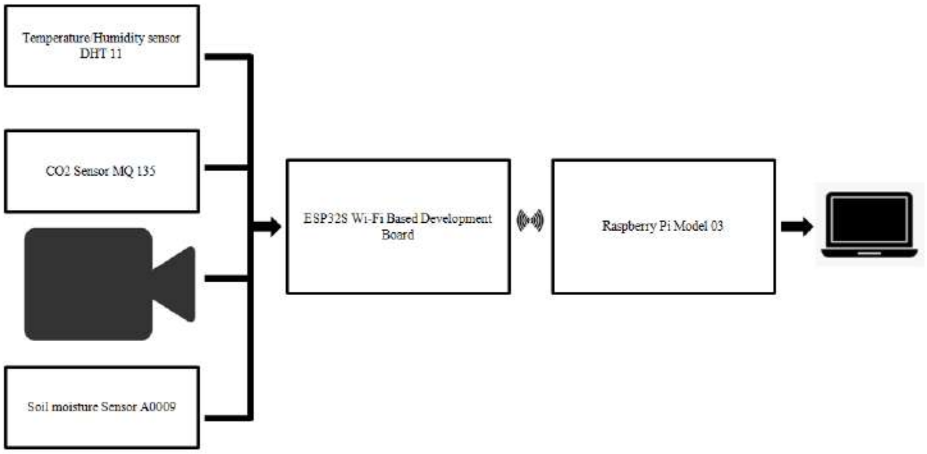


Figure 1.2: Current monitoring of the project

## 2.2 Initial Design:

Designing IoT-equipped and AI-enabled Wireless Sensor Network (WSN) nodes for precision agriculture requires careful consideration of various factors.

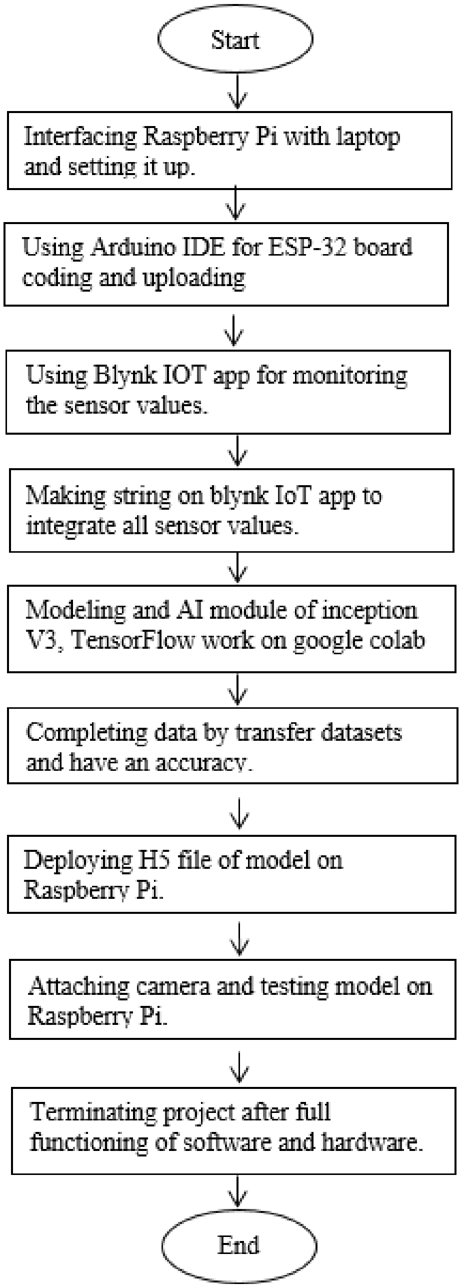


**Figure 2.3: Flowchart**

### 2.2.1 Interfacing Raspberry Pi

To communicate a Raspberry Pi with a PC, you have a couple of choices relying upon what you need to accomplish:

Far off Work area Association: to control your Raspberry Pi's graphical UI (GUI) straightforwardly from your PC, you can set up a distant work area association. How it's done:

Interface your Raspberry Pi to a similar neighborhood network as your PC utilizing an Ethernet link or Wi-Fi.

Empower Far off Work area on your Raspberry Pi by going to the Raspberry Pi Arrangement menu and empowering the VNC (Virtual Organization Processing) server.

Introduce a VNC client on your PC, like Real VNC. Launch the VNC client on your PC and enter the IP address of your Raspberry Pi to lay out the association.

SSH (Secure Shell): On the off chance that you favor an order line interface (CLI) and need to remotely access and control your Raspberry Pi's terminal from your PC, you can utilize SSH. This is how it's done:

Guarantee your Raspberry Pi is associated with a similar neighborhood network as your PC.

Empower SSH on your Raspberry Pi by going to the Raspberry Pi Design menu and empowering SSH.

On your PC, open a terminal or order brief.

Enter the accompanying order: ssh pi@<Raspberry Pi IP address>, supplanting <Raspberry Pi IP address> with the IP address of your Raspberry Pi.

When incited, enter the secret key for the "pi" client (of course, it is "raspberry").

Headless Arrangement: On the off chance that you don't have an extra screen, console, or mouse, you can set up your Raspberry Pi without straightforwardly interfacing them. How it's done:

Streak the Raspberry Pi operating system (recently called Raspbian) onto a SD card utilizing a PC or PC.

Make a vacant document named "ssh" (no record augmentation) in the boot parcel of the SD card. This empowers SSH as a matter of course.

Design Wi-Fi by making a document named "wpa_supplicant.conf" in the boot parcel. The document ought to contain the organization design subtleties. Here is a model:

```
country=<country_code>
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=net
deve_config=1
 network={
```

```
    ssid="<your_SSID>"
    psk="<your_password>"
}
```

Discharge the SD card from your PC and addition it into the Raspberry Pi.
Power up the Raspberry Pi, and it ought to naturally associate with your Wi-Fi organization and empower SSH.
Use SSH to interface with your Raspberry Pi from your PC as made sense of in the past SSH method.Once you have laid out an association with your Raspberry Pi, you can design and set it up further as per your particular prerequisites.

Figure 2.4: Interfacing Raspberry Pi

## 2.2.2 Interfacing ESP32 Using Arduino IDE

To involve the Arduino IDE for coding and transferring projects to an ESP32 board,

1. Introduce the Arduino IDE: Download and introduce the most recent form of the Arduino IDE from the authority Arduino site (https://www.arduino.cc/en/programming).

2. Add ESP32 Board Backing: Open the Arduino IDE and go to Record - > Inclinations. In the Inclinations window, see as the "Extra Sheets Administrator URLs" field and enter the accompanying URL:

arduino

https://dl.espressif.com/dl/package_esp32_index.json

Click "Alright" to save the inclination.

3. Introduce ESP32 Board Backing: Go to Devices - > Board - > Sheets Chief. In the Sheets Administrator window, type "esp32" in the pursuit bar. Search for "esp32" by Espressif Frameworks and snap on it. Click the "Introduce" button to introduce the ESP32 board support.

4. Select the ESP32 Board: After the establishment is finished, go to Apparatuses - > Board and select the suitable ESP32 board from the rundown. Pick the board variation that matches your particular ESP32 board (e.g., ESP32 Dev Module).

5. Pick the Sequential Port: Interface your ESP32 board to your PC through USB. Go to Apparatuses - > Port and select the sequential port that relates to your ESP32 board. In the event that the port isn't identified, ensure you have the important USB drivers introduced.

6. Compose and Transfer the Code: Presently, you can compose your code in the Arduino IDE. Begin with an essential model or compose your custom code. At the point when you're prepared to transfer the code to your ESP32 board, click on the "Transfer" button (right bolt symbol) in the Arduino IDE toolbar.The IDE will gather the code and transfer it to the ESP32 board. You can screen the advancement in the status bar at the lower part of the Arduino IDE. Once the transfer is finished, you ought to see the "Done transferring" message.

7. Confirm and Test: In the wake of transferring the code, the ESP32 board will begin running the program. Assuming that everything is working accurately, you ought to see the normal result or conduct in light of your code. You can involve the Chronic Screen in the Arduino IDE to troubleshoot and screen the result from the ESP32 board.

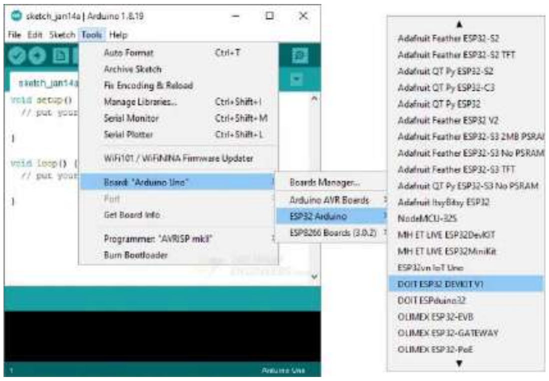


Figure 2.5: Selecting ESP-32 Board

### 2.2.3 Blynk App for Data Storing

To use the Blynk IoT app for monitoring sensor values,

1. Create a Blynk Account: Download and install the Blynk app from your app store and create a Blynk account. Alternatively, you can use the Blynk web dashboard by visiting https://dashboard.blynk.cc/. Sign in using your Blynk account credentials.
2. Create a New Project: In the Blynk app or web dashboard, create a new project by clicking on the "+" or "New Project" button. Give your project a name and select the hardware device you will be using (e.g., ESP32).
3. Obtain Auth Token: After creating the project, Blynk will assign an authorization token (Auth Token) to your project. This token is required for your hardware device to connect to the Blynk server. Make a note of this token as you will need it later.
4. Add Widgets: In your Blynk project, you can add various widgets to monitor sensor values. To do this:

Click on the "+" button to add a widget.

Choose the desired widget based on the type of sensor value you want to monitor. For example, you can use a Gauge widget to display analog sensor readings, or a Value Display widget for displaying numeric values.

Configure the widget properties, such as the pin number (virtual or physical), label, and other settings as per your requirements.

5. Connect Your Hardware Device to Blynk: In your Arduino code, make sure you have the Blynk library installed. You can install it by going to Sketch -> Include Library -> Manage Libraries and searching for "Blynk".

Here's a basic example code snippet to get you started:

Replace "your_auth_token" with the Auth Token obtained from your Blynk project.

6.Configure Wi-Fi: In the code snippet above, enter your Wi-Fi credentials by replacing "your_wifi_ssid" and "your_wifi_password" with your network's SSID and password, respectively.

7. Upload the Code: Upload the code to your ESP32 board using the Arduino IDE, as explained in the previous response.

8. Monitor Sensor Values: Once the code is uploaded and the ESP32 is connected to Blynk, you can open the Blynk app or web dashboard. It should start receiving data from your device.

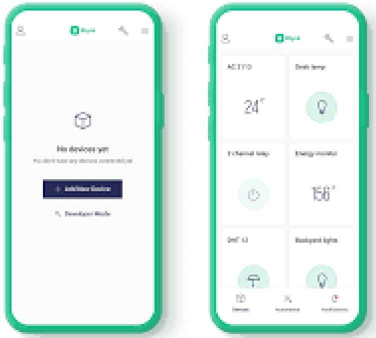


Figure 2.6: Dashboard Blynk App

### 2.2.4 Integrating all Sensors and Creating a string on Blynk App

To make a string on the Blynk IoT application that coordinates all sensor values, you can utilize the Blynk's Virtual Pins and the Blynk.virtualWrite() capability in your Arduino code.

1. In the Blynk application or web dashboard, add a "Terminal" gadget to your venture. This gadget will be utilized to show the coordinated sensor values as a string.

2. In your Arduino code, incorporate the Blynk library in the event that you haven't as of now:

```
#incorporate <WiFi.h>

#incorporate <BlynkSimpleEsp32.h>
```

3. Proclaim factors to store the sensor values:

```
int sensor1Value;

float sensor2Value;
```

```
// Add more factors for other sensor values on a case by case basis
```

4.In the arrangement() capability, introduce the Blynk association and set up the terminal gadget:

```
void arrangement()
{
Serial.begin(9600);
Blynk.begin(auth, ssid, pass);
\\Add this line to set up the terminal gadget
Blynk.virtualWrite(V1, "Sensor Values:");//V1 is the virtual pin allocated to the terminal gadget
}
```

5. In the know() capability, read the sensor values and update the terminal gadget with the coordinated string:

```
void circle()
{
// Peruse sensor values and store them in factors
sensor1Value = analogRead(A0);
sensor2Value = analogRead(A1) * 0.1;
// Make the incorporated string with sensor values
```

String sensorString = "Sensor 1: " + String(sensor1Value) + ", Sensor 2: " + String(sensor2Value);

// Update the terminal gadget with the sensor string

Blynk.virtualWrite(V1, sensorString);//V1 is the virtual pin allocated to the terminal gadget Blynk.run();

}

In the code above, A0 and A1 address the simple information pins for the particular sensors. Change them in light of your particular sensor associations.

Transfer the code to your ESP32 board.

Presently, when you open the Blynk application or web dashboard and view the terminal gadget, you ought to see the coordinated sensor values being shown as a string. The string will refresh ceaselessly as the circle capability runs and peruses the sensor values.

### 2.2.5 Modeling AI modules on Google Colab

Indeed, you can utilize the Beginning V3 model and TensorFlow in Google Colab. Google Colab is a cloud-based Python journal climate that gives free admittance to GPUs and TPUs, making it helpful for running profound learning models.

To utilize the Beginning V3 model in Google Colab, you really want to follow these means:

1. Open Google Colab: Go to https://colab.research.google.com/and make another journal.

2. Set up TensorFlow: TensorFlow is as of now introduced in Google Colab, so you don't have to introduce it. Be that as it may, you can really take a look at the variant by running !pip show tensorflow in a code cell.

3. Import the vital libraries: In a code cell, import the necessary libraries by running:

```
import tensorflow as tf
from tensorflow.keras.applications.inception_v3 import InceptionV3
from tensorflow.keras.preprocessing import picture
From tensorflow.keras.applications.inception_v3 import preprocess_input, decode_predictions
import numpy as np
```

4. Load the Beginning V3 model: In a code cell, load the Origin V3 model utilizing the accompanying code:

python

Duplicate code

```
model = InceptionV3(weights='imagenet')
```

This will download the pre-prepared loads of the Beginning V3 model.

5. Utilize the model for expectations: You can utilize the stacked model to make forecasts on pictures. Here is a model:

```
# Load and preprocess the picture
img_path = 'path_to_your_image.jpg' # Supplant with the way to your picture
img = image.load_img(img_path, target_size=(299, 299))
x = image.img_to_array(img)
x = np.expand_dims(x, axis=0)
x = preprocess_input(x)
```

```
# Make expectations
preds = model.predict(x)
decoded_preds = decode_predictions(preds, top=3)[0]
for _, class_name, prob in decoded_preds:
print(f'{class_name}: {prob * 100}%')
```

Supplant 'path_to_your_image.jpg' with the genuine way to your picture record.

That is all there is to it! You can show each code cell in Google Colab to squeezing Shift+Enter. This will execute the code and give you the result. Make a point to transfer your picture to research Colab in the event that it's not currently in the virtual machine's record framework.

Note that you could have to empower the GPU in Google Colab for quicker calculations. You can do this by going to "Runtime" > "Change runtime type" and choosing "GPU" as the equipment gas pedal.

If it's not too much trouble, remember that running profound learning models on huge datasets or for a drawn out period might have asset restrictions in Google Colab's complementary plan.

**Figure 2.7: Instances of tomato leaf pictures of the Plant dataset**

Tomato leaf images of the Plant dataset dimensions are primarily modified to be acceptable to feed the input layers of the three compact CNNs. As these layers only admit images of specific sizes; thus, the new size of the images is 224 × 224 × 3 for the three CNNs. Thereafter, these images are augmented using various data augmentation approaches. The effectiveness of CNN models is strongly dependent upon the training dataset. On adequately large datasets, these models show improved results and high generalizability. The datasets currently available for tomato plant disease typically lack sufficient images in a variety of conditions, which is required for developing high-accuracy models. Given the small size of the dataset, the model could overfit and perform poorly on real-world test data. Hereafter, numerous data augmentation methods, such as flipping, rotation, shearing, and scaling are being employed to boost the amount of images utilized to train the CNNs; thus, improving training performance and preventing overfitting.

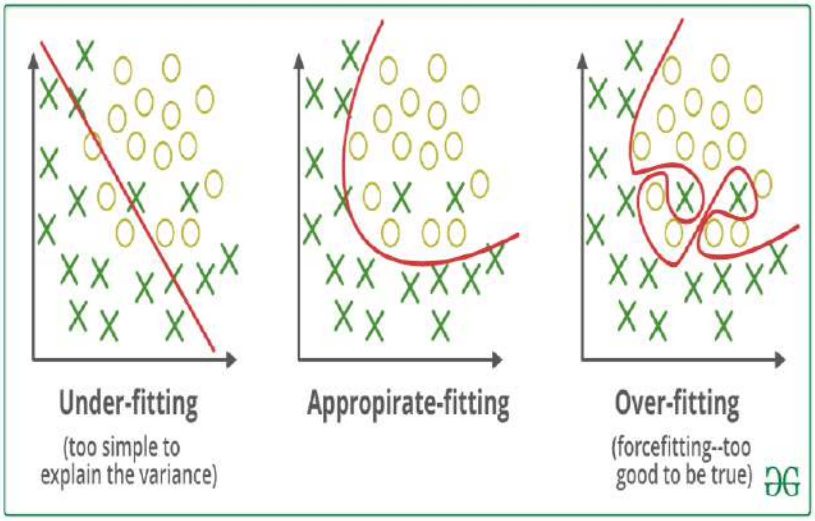


Figure 2.8: Under Appropriate and Over-Fitting

## Under-fitting in Machine Learning:

A measurable model or an AI calculation is said to have underfitting when it can't catch the fundamental pattern of the information, i.e., it just performs well on preparing information however performs ineffectively on testing information. (It's very much like attempting to fit small jeans) Underfitting annihilates the exactness of our AI model. Its event just implies that our model or the calculation doesn't fit the information alright. It as a rule happens when we have less information to fabricate an exact model and furthermore when we attempt to construct a direct model with less non-straight information. In such cases, the guidelines of the AI model are excessively simple and adaptable to be applied to such negligible information, and subsequently the model will presumably make a ton of wrong forecasts. Underfitting can be abstained from by utilizing more information and furthermore lessening the highlights by include choice.

More or less, Underfitting alludes to a model that can neither performs well on the preparation information nor sum up to new information.

## Reasons for Underfitting

- High bias and low variance.
- The size of the training dataset used is not enough.
- The model is too simple.
- Training data is not cleaned and also contains noise in it.

## Techniques to Reduce Underfitting

- Increase model complexity.
- Increase the number of features, performing feature engineering.
- Remove noise from the data.
- Increase the number of epochs or increase the duration of training to get better results.

## Over-fitting in Machine Learning:

A factual model is supposed to be over-fitted when the model doesn't make precise forecasts on testing information. At the point when a model gets prepared with such a lot of information, it begins gaining from the commotion and wrong information passages in our informational index. Furthermore, while testing with test information brings about High difference. Then the model doesn't order the information accurately, in light of such a large number of subtleties and commotion. The reasons for overfitting are the non-parametric and non-straight techniques on the grounds that these sorts of AI calculations have more opportunity in building the model in view of the dataset and thusly they can truly assemble ridiculous models. An answer for stay away from overfitting is utilizing a straight calculation in the event that we have direct information or utilizing the boundaries like the maximal profundity assuming that we are utilizing choice trees.

## Reasons for Overfitting:

- High variance and low bias.
- The model is too complex.
- The size of the training data.

## Techniques to Reduce Overfitting:

- Increase training data.
- Reduce model complexity.
- Early stopping during the training phase (have an eye over the loss over the training period as soon as loss begins to increase stop training).
- Ridge Regularization and Lasso Regularization.
- Use dropout for neural networks to tackle overfitting.

### 2.2.6 Transfer datasets and have an accuracy

To finish missing information utilizing move learning and assess the precision of the model, you can follow these means:

1. Set up the dataset: Gather or get a dataset that contains the significant information for your errand. Guarantee that the dataset has an adequate number of tests with missing qualities that should be finished.
2. Part the dataset: Split the dataset into preparing and testing sets. The preparation set will be utilized to prepare the model, while the testing set will be utilized to assess its exactness.
3. Preprocess the information: Play out any fundamental preprocessing steps on the dataset, like dealing with missing qualities, normalizing the information, and parting it into input elements and target factors.
4.Load the pre-prepared model: Pick a pre-prepared model that is reasonable for your undertaking. For instance, you can utilize Beginning V3, as referenced prior, or some other pre-prepared model accessible in TensorFlow.
5. Change the model: Eliminate the first result layer of the pre-prepared model and add another layer that matches the quantity of classes or elements in your dataset.

6. Train the model: Train the changed model utilizing the preparation set. During preparing, the model will figure out how to finish the missing information in view of the accessible data.
7. Assess the model: When preparing is finished, assess the precision of the model utilizing the testing set. This will provide you with a gauge of how well the model can finish the missing information.

8. Calibrate the model (discretionary): In the event that the underlying exactness isn't acceptable, you can tweak the model by changing hyper boundaries, changing the engineering, or expanding the preparation information.
9. Rehash stages 6-8 on a case by case basis: Emphasize the preparation, assessment, and calibrating ventures until you accomplish a palatable precision. It's critical to take note of that the particular execution subtleties might fluctuate relying upon the idea of your datasets and the main job. Also, move learning approaches for finishing missing information might contrast in light of the particular necessities of your datasets.

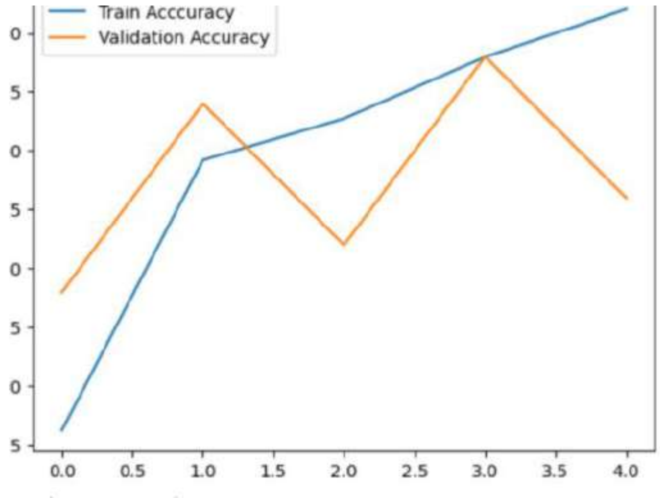


Figure 2.8: Accuracy Graph

## 2.2.7 Deploying H5 file on Raspberry Pi:

To send a H5 record normally connected with AI models on a Raspberry Pi, you'll have to play out a couple of steps.

1. Set up your Raspberry Pi: Guarantee that your Raspberry Pi is appropriately set up, running a viable working framework (like Raspbian). Ensure you have the important peripherals associated, including a console, mouse, show, and web network.
2. Introduce conditions: Prior to conveying the H5 record, you'll have to introduce the

necessary conditions on your Raspberry Pi. These conditions could incorporate Python, Tensorflow, and some other libraries or bundles your H5 model depends on. You can normally introduce them utilizing bundle administrators like pip or well-suited get.

3. Move the H5 document to your Raspberry Pi: Move the H5 record from your improvement machine to the Raspberry Pi. You can utilize different strategies like USB stockpiling gadgets, SCP (secure duplicate), or record sharing over the organization. Make a note of the record's area on the Raspberry Pi.
4. Compose a sending script: Make a Python script that heaps and conveys the H5 model on the Raspberry Pi. You'll have to utilize a reasonable system like TensorFlow or Keras to stack the model from the H5 record and play out any vital preprocessing or post-handling.
5. Run the arrangement script: Execute the sending script on your Raspberry Pi utilizing the suitable Python translator. For instance, assuming you have Python 3 introduced, you can run the content with the order python3 your_script.py. Guarantee that the essential conditions and libraries are accessible to the Python climate.
6. Test the arrangement: Check that your H5 model is effectively conveyed on the Raspberry Pi by testing its usefulness with test inputs. This step guarantees that the arrangement cycle was fruitful and that the model is functioning true to form in the Raspberry Pi climate.

### 2.2.8 Testing Deployed file by attaching camera to Raspberry Pi

To connect a camera to a Raspberry Pi and test your sent model, you'll have to follow these means:

1. Interface the camera module: In the event that you're utilizing an authority Raspberry Pi camera module, find the camera connector on the Raspberry Pi board and tenderly addition the strip link into it. Ensure the link is appropriately adjusted, and secure it by flipping the

connector's lock down. On the off chance that you're utilizing a USB camera, interface it to one of the USB ports on the Raspberry Pi.

2. Empower the camera interface: On your Raspberry Pi, you really want to guarantee that the camera connection point is empowered. To do this, follow these means:

- Open a terminal or interface with your Raspberry Pi through SSH.
- Run the order sudo raspi-config to open the Raspberry Pi arrangement instrument.
- Explore to "Point of interaction Choices" or "Connecting Choices" and select "Camera". Empower the camera connection point and leave the design device.

3. Introduce camera libraries: Contingent upon the sort of camera you're utilizing, you might have to introduce explicit libraries or bundles to interact with it. For instance, in the event that you're utilizing the authority Raspberry Pi camera module, you can introduce the expected libraries by running the order sudo able get introduce python-picamera. In the event that you're utilizing a USB camera, it could sort out of the container without requiring any extra establishment.

4. Catch pictures or video: Compose a Python content to catch pictures or video utilizing the camera and feed them to your sent model for derivation. You can utilize the picamera library for the authority Raspberry Pi camera module or OpenCV for USB cameras. Here is a model content utilizing the picamera library:

```
import time
import picamera
# Initialize the camera
camera = picamera.PiCamera()
# Set camera parameters (e.g., resolution, framerate)
#
# Capture and process images
while True:
  try:
```

```
       # Capture image
        image_path = '/path/to/save/image.jpg'
 camera.capture(image_path)
# Perform any necessary preprocessing on the image
   # …
 # Load the image and use the model for inference
      # result = model.predict(image)
# Perform any desired actions with the model's output
 # …
  # Wait before capturing the next image
     time.sleep(1)
     except KeyboardInterrupt:
 # Clean up resources
     camera.close()
    break
```

Customize the script based on your specific camera and model requirements.

5. Run the content: Execute the Python script on your Raspberry Pi utilizing the suitable Python translator. For instance, in the event that you have Python 3 introduced, you can run the content with the order python3 your_script.py. Guarantee that the important conditions and libraries are accessible to the Python climate.
6. Test the camera and model: Check that the camera is catching pictures or video and taking care of them to your sent model for induction. Screen the content's result or play out any ideal activities with the model's forecasts in light of your particular use case.

By following these means, you ought to have the option to connect a camera to your Raspberry Pi, catch pictures or video, and utilize your sent model for induction on the caught information

# Chapter 3
# Results

## Prototype Development and Experimental Tests

## 3.1 Introduction:

To validate the theoretical concepts, prototype for IoT Equipped and AI Enables WSN nodes for Precision Agriculture is developed and tested practically, and the detail is mentioned in this chapter.

## 3.2 Prototype Development and Assembling:

The discussion in the previous chapter investigates the detailed working performance of the initial optimized design of IoT Equipped and AI Enabled WSN nodes for Precision Agriculture. The effectiveness of the proposed prototype design and concept is validated with a tested prototype.

## 3.3 Designed Prototype and Results:

### For Internet of Things

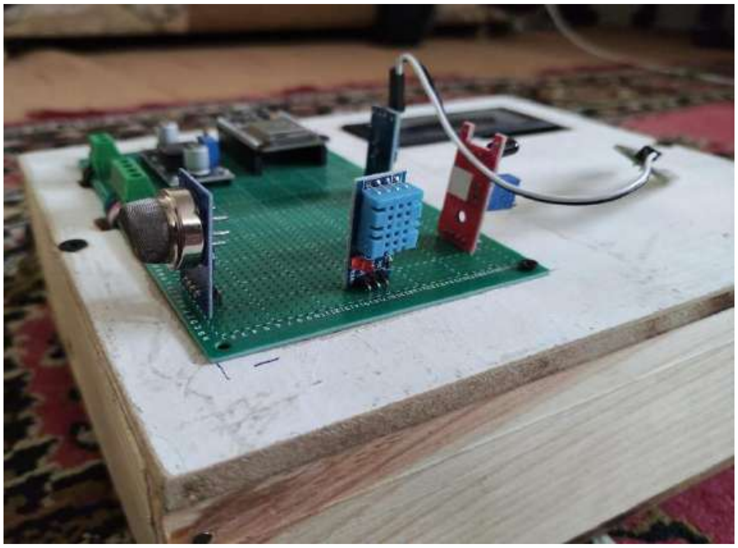


Figure 3.1: Prototype

The above figure 3.1 shows the prototype of different sensors connected wirelessly to the development board ESP-32. Here we are using MQ135 co2 sensor, DHT11 Temperature and humidity sensor, Soil Moisture Sensor and Sunlight rays detector.

Following are the steps required for the connection of the sensors with ESP-32.

1. Assemble the necessary parts:

ESP32 microcontroller
MQ135 CO2 sensor
DHT11 temperature and dampness sensor
Soil dampness sensor
Daylight beams finder (contingent upon the particular module you're utilizing)
Jumper wires
Veroboard

2. Associate the parts to the ESP-32:

Interface the VCC pin of every sensor to a 3.3V pin on the ESP32.
Interface the GND (ground) pin of every sensor to a GND nail to the ESP32.
Interface the information/yield pins of every sensor to any suitable computerized pins on the ESP32. Make a point to note down the pins you decide for every sensor.
Set up the important libraries for the sensors.
You'll require libraries for the MQ135, DHT11, and any extra libraries expected for the daylight beams finder or soil dampness sensor.
Introduce the sensors in the code, appointing the proper advanced pins you associated them to.
Utilize the library capabilities to peruse information from every sensor.

You can print the sensor readings to the chronic screen or send them to a server or show gadget, contingent upon your undertaking prerequisites.

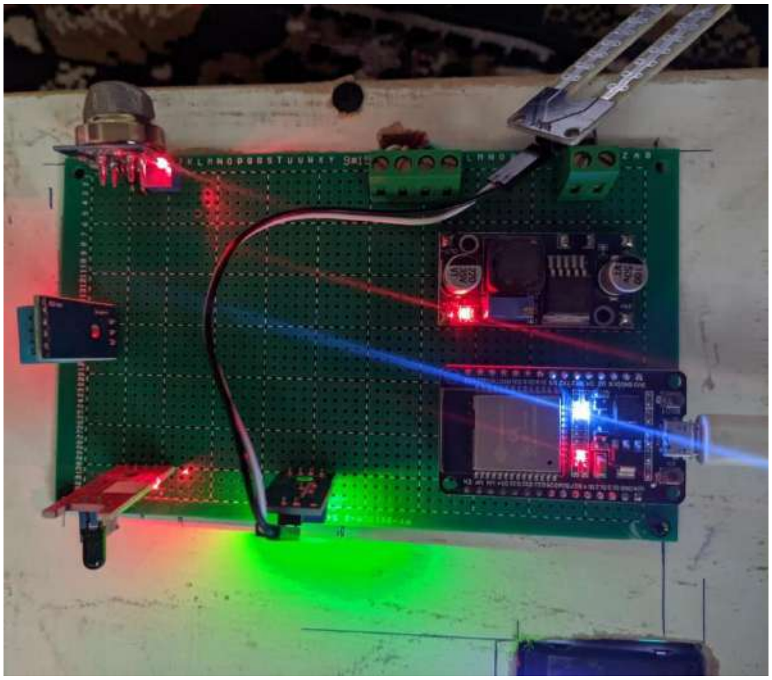


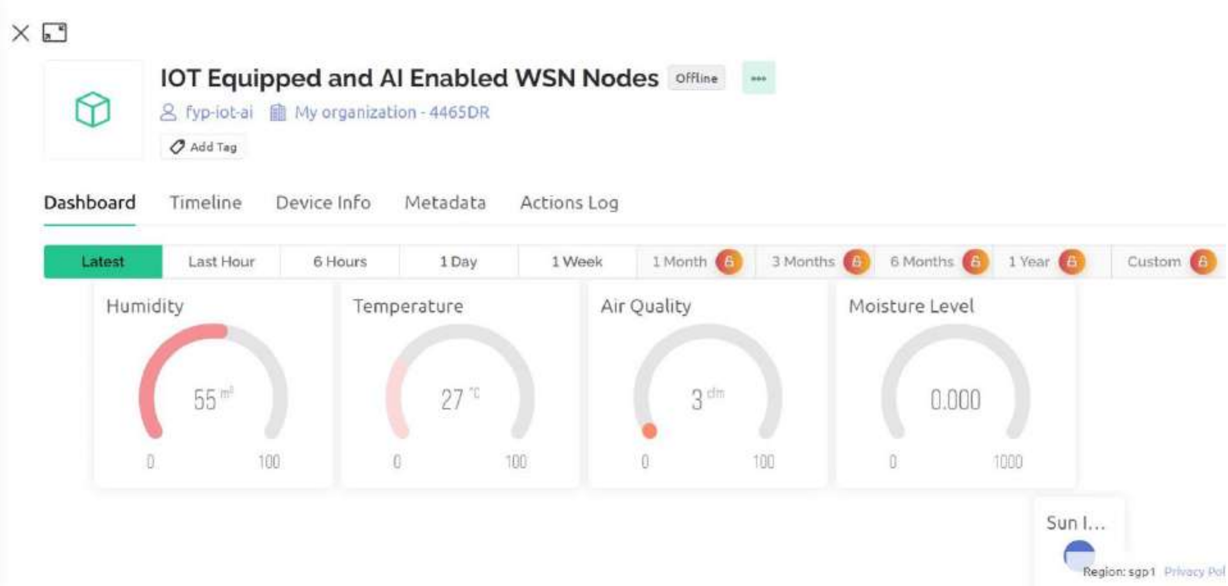
Figure 3.2: Working of Circuit

We have applied a voltage of 5v to the development board ESP-32 as shown in the figure 3.2, all the other sensors attached to development board energized.

```
Temperature:28.40
Humidity:59.00
Air Quality:10
Moisture :19.946
```

Figure 3.3: Value Displayed on LCD

The above figure 3.3, the display screen, which shows us the values of the given sensors in our prototype.This example assumes you have a 16x2 character LCD screen connected to development board ESP-32 using the Liquid Crystal library. It reads an analog sensor value, converts it to a string, and displays it on the LCD screen.



IOT Equipped and AI Enabled WSN Nodes Offline

fyp-iot-ai My organization - 4465DR

Figure 3.4: Values when no voltage is supplied

At the point when no voltage is provided to an ESP-32 , it won't work as shown in the figure 3.4, and the Blynk IoT application's on the web or disconnected status become offline. The ESP-32 requires a power supply to work, and without power, it can't play out any errands or speak with outer gadgets or applications.

On the off chance that the ESP-32 isn't getting any voltage, it can not lay out an association with the Blynk IoT application, whether or not the application is on the web or disconnected. The shortfall of force implies that the microcontroller can't execute any code or speak with the Blynk server.

In such a situation, the ESP-32 can not send or get information from the Blynk IoT application. The LCD show or some other associated gadgets will likewise not get any guidelines or show any qualities since the microcontroller isn't controlled.

To guarantee legitimate working of the ESP-32 and Blynk IoT application, try to give a dependable power supply to the ESP-32, permitting it to work and lay out correspondence with the application.

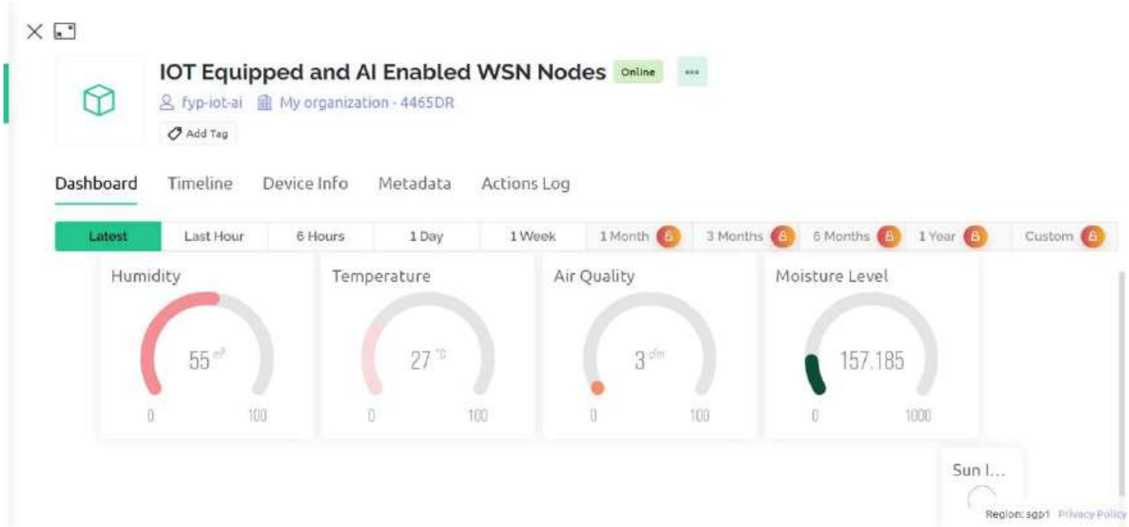


Figure 3.5: Experimented Values

As the prototype is tested by applying co2 to MQ135 sensor, by applying water to the soil moisture sensor and light to sunlight sensor the values changes as shown in the figure 3.5.

## For Artificial Intelligence

## Image Pre-Processing and Labelling:

Before training the model, image pre-processing was used to change or boost the raw images that needed to be processed by the CNN classifier. Building a successful model requires analyzing both the design of the network and the format of input data. We pre-processed our dataset so that the proposed model could take the appropriate features out of the image. The first step was to normalize the size of the picture and resize it to 224 × 224 pixels. The images were then transformed into grey. This stage of pre-processing means that a considerable amount of training data are required for the explicit learning of the training data features. The next step was to group tomato leaf pictures by type, then mark all images with the correct acronym for the disease. In this case, the dataset showed ten classes in test collection and training.

## Training Dataset:

Preparing the dataset was the first stage in processing the existing dataset. The Convolutional Neural Network process was used during this step as image data input, which eventually formed a model that assessed performance. Normalization steps on tomato leaf images are shown in Figure below.

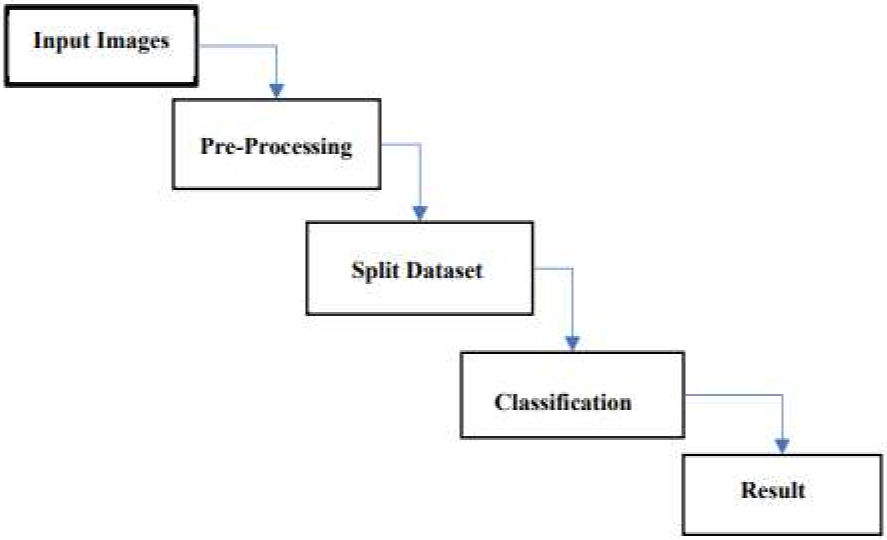


Figure 3.6: Phases of Training Datasets

## Convolutional Neural Networks:

The CNN is a neural network technology widely employed today to process or train the data in images. The matrix format of the Convolution is designed to filter the pictures. For data training, each layer is utilized in the Convolution Neural Network, including the following layers: input layer, convo layer, fully connected layer pooling layer, drop-out layer to build CNN, and ultimately linked dataset classification layer. It can map a series of calculations to the input test set in each layer.

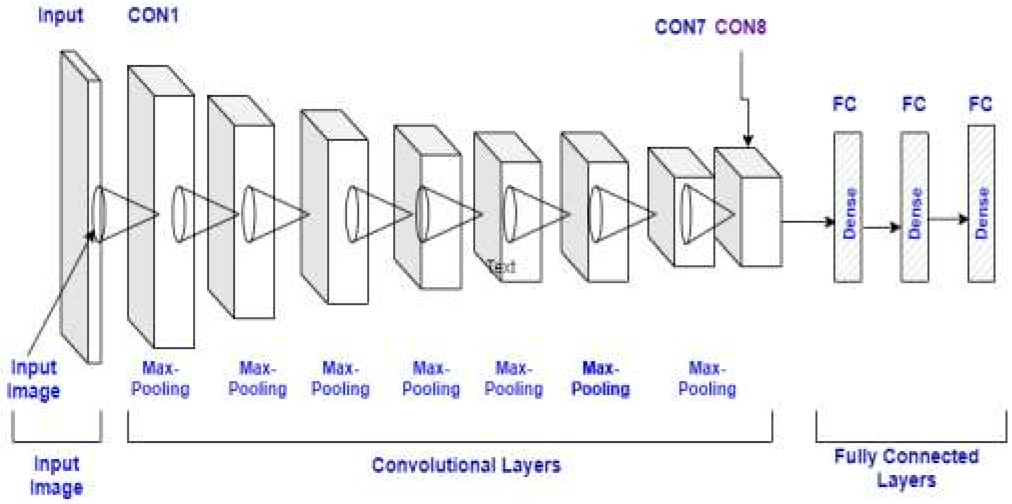


Figure 3.7 CNN Architecture

## Convolutional Layer

A convolution layer is used to map characteristics using the convolution procedure with the presentation layer. Each function of the map is combined with several input characteristics. Convolution can be defined as a two-function operation and constitutes the basis of CNNs. Each filter is converted to each part of the input information, and a map or 2D function map is generated. The complexity of the model encounters significant layer convolutional performance optimization. Calculated in the following equation for input $z$ of the $i$th coalescent layer.

## Pooling Layer:

The pooling layer increases the number of parameters exponentially to maximize and improve precision. Furthermore, with growing parameters, the size of the maps is reduced. The pooling layer reduces the overall output of the convolution layer. It reduces the number of training parameters by considering the spatial properties of an area representing a whole country. It also distributes the total value of all R activations to the subsequent activation in the chain. In the m-th max-pooled band, there are J-related filters that are combined.

## Fully Connected Layers:

Each layer in the completely connected network is connected with its previous and subsequent layers. The first layer of the utterly corresponding layer is connected to each node in the pooling layer's last frame. The parameters used in the CNN model take more time because of the complex computation; it is the critical drawback of the fully linked sheet. Thus, the elimination of the number of nodes and links will overcome these limitations. The dropout technique will satisfy deleted nodes and connections.

## Dropout:

An absence is an approach in which a randomly selected neuron is ignored during training, and they are "dropped out" spontaneously. This means that they are briefly omitted from their contribution to the activation of the downstream neurons on the forward transfer, and no weight changes at the back are applied to the neuron. Thus, it avoids overfitting and speeds up the process of learning. Overfitting is when most data has achieved an excellent percentage through the training process, but a difference in the prediction process occurs. Dropout occurs when a neuron is located in the network in the hidden and visible layers.

## Results Analysis and Discussion For AI:

The complete experiment was performed on Google Colab. The result of the proposed method is described with different test epochs and learning rates and explained in the next sub-section.

This research used epoch 50 and epoch 100 for comparison, though learning rates were 0.0001. Figure 3.8(a) shows the comparison between training and validation loss, and Figure 3.8(b) shows the comparison between training accuracy and validation accuracy.

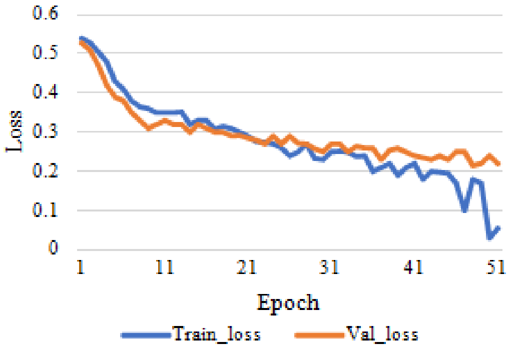


Figure 3.8(a): Loss Graph and

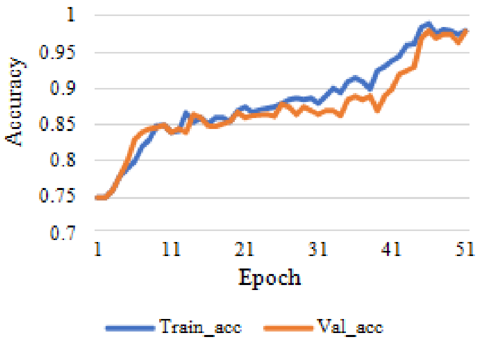


Figure 3.8(b): Accuracy Graph

# Chapter 4
# Discussion

In discussion, the project "IoT Equipped and AI-Enabled WSN Nodes for Precision Agriculture" aims to enhance agricultural practices through the implementation of advanced technologies. By integrating Internet of Things (IoT) and Artificial Intelligence (AI) techniques, the project offers significant benefits for precision agriculture. The use of IoT-enabled wireless sensor network (WSN) nodes allows for the collection and monitoring of real-time data from various environmental parameters crucial for successful crop management. These parameters may include temperature, humidity, soil moisture, light intensity, and more. The WSN nodes communicate and transmit this data to a central hub or cloud-based platform for further analysis and decision-making.

Furthermore, the incorporation of AI algorithms and machine learning techniques enables intelligent analysis and interpretation of the collected data. AI algorithms can detect patterns, anomalies, and correlations in the data, providing valuable insights for optimizing agricultural processes. This includes predicting optimal irrigation schedules, identifying crop diseases, managing pest control, and optimizing resource allocation.

By leveraging IoT and AI technologies, the project offers several benefits for precision agriculture. These include improved resource management, increased crop yield, optimized water and energy usage, enhanced disease and pest detection, and reduced environmental impact. Farmers and agricultural practitioners can make data-driven decisions based on the real-time information gathered from the WSN nodes, leading to more efficient and sustainable farming practices. In conclusion, the "IoT Equipped and AI-Enabled WSN Nodes for Precision Agriculture" project demonstrates the potential of advanced technologies in revolutionizing the agricultural sector. It presents a holistic approach that combines IoT, wireless sensor networks, and AI algorithms to address the challenges faced by modern farmers and contribute to the advancement of precision agriculture for a more productive and sustainable future.

# Chapter 5

# References

1. Survey 2 Agriculture%2C by finance Government of Pakistan%2C 2018. .
2. M. R. Mohd Kassim, I. Mat, and A. N. Harun, “Wireless Sensor Network in Precision Agriculture Application,” 2014 International Conference on Computer, Information and Telecommunication Systems (CITS), 2014. doi:10.1109/cits.2014.6878963
3. M. A. Matin and M. M. Islam, “Overview of wireless sensor network,” Wireless Sensor Networks - Technology and Protocols, 2012. doi:10.5772/49376
4. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless Sensor Networks: A survey,” Computer Networks, vol. 38, no. 4, pp. 393–422, 2002. doi:10.1016/s1389-1286(01)00302-4
5. I. Lee and K. Lee, “The internet of things (IOT): Applications, investments, and challenges for enterprises,” Business Horizons, vol. 58, no. 4, pp. 431–440, 2015. doi:10.1016/j.bushor.2015.03.008
6. R. van Kranenburg and A. Bassi, “IOT challenges,” Communications in Mobile Computing, vol. 1, no. 1, 2012. doi:10.1186/2192-1121-1-9
7 H. Sahota, R. Kumar, A. Kamal, and Jing Huang, “An energy-efficient wireless sensor network for Precision Agriculture,” The IEEE symposium on Computers and Communications, 2010. doi:10.1109/iscc.2010.5546508
8. D. D. Chaudhary, S. P. Nayse, and L. M. Waghmare, “Application of wireless sensor networks for greenhouse parameter control in Precision Agriculture,” International Journal of Wireless &amp; Mobile Networks, vol. 3, no. 1, pp. 140–149, 2011. doi:10.5121/ijwmn.2011.3113
9. D. Anurag, S. Roy, and S. Bandyopadhyay, “Agro-sense: Precision Agriculture using sensor-based Wireless Mesh Networks,” 2008 First ITU-T Kaleidoscope Academic Conference - Innovations in NGN: Future Network and Services, 2008. doi:10.1109/kingn.2008.4542291
10. G. J. Pottie, "Wireless sensor networks," *1998 Information Theory Workshop (Cat. No.98EX131)*, Killarney, Ireland, 1998, pp. 139-140, doi: 10.1109/ITW.1998.706478.
11. S. Slijepcevic and M. Potkonjak, “Power Efficient Organization of Wireless Sensor Networks,” ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No.01CH37240). doi:10.1109/icc.2001.936985
12. “Network management for Wireless Sensor Networks,” Wireless Sensor Networks, pp. 262–272. doi:10.1002/9780470112762.ch9