Shri B. V. V. Sangha's

BASAVESHWAR ENGINEERING COLLEGE BAGALKOTE

2024-25

DEPARTMENT OF ELECTRONICS AND COMMUINCATION ENGINEERING

A MINI PROJECT REPORT [22UEC511P] ON

SMART IRRIGATION AND PEST DETECTION

PROJECT GUIDE	PROJECT COORDINATORS	HOD
DR. A.V.SUTAGUNDAR	DR. MAMATA SATARADDI	DR.J.D.MALLAPUR
	DR. B.M.ANGADI	

Project Associates

AMITSING.A.HUNNUR	2BA22EC012
ANKITA HULLOLI	2BA22EC015
ARUN HADIMANI	2BA22EC018
BHAGYASHRI KUMBAR	2BA22EC021

Shri B. V. V. Sangha's

BASAVESHWAR ENGINEERING COLLEGE BAGALKOTE

587102

2024-2025

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

1. This is to certify that this mini project report entitled **"SMART IRRIGATION AND PEST DETECTION"** submitted by **Amitsing.A.Hunnur [2BA22EC012], Ankita Hulloli [2BA22EC015], Arun Hadimani [2BA22EC018] , Bhagyashri Kumbar [2BA22EC021]** studying in V semester during year 2024-25 from Basaveshwar Engineering College, Bagalkot, in partial fulfilment for the award of degree **Bachelor of Engineering in Electronics and Communication Engineering** is a Bonafide record work done by them under my supervision.

GUIDE	HOD	PRINCIPAL
Dr. A.V.Sutagundar	**Dr. J. D. Mallapur**	**Dr. Basavaraj Hiremath**

Examiner	**Name of Examiner**	**Signature**
HOD/Nominee	____________________	________________
Internal Examiner	____________________	________________
External Examiner	____________________	________________

DECLARATION

We hereby declare that this project report titled **"Smart Irrigation and Pest Detection"** submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Electronics and Communication Engineering, is a record of original work carried out by us under the supervision of **Dr. A.V.Sutagundar.** This work has not formed the basis for the award of any other degree or diploma in this or any other institution or university. In accordance with ethical practices in reporting scientific information, due acknowledgments have been made wherever the findings of others have been cited.

Amitsing.A.Hunnur(2BA22EC012)

Ankita Hulloli (2BA22EC015)

Arun Hadimani(2BA22EC018)

Bhagyashri Kumbar(2BA22EC018)

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this project would be incomplete without the mention of the people who made it possible, without whose constant endurance and encouragement would have made efforts go in vain. We considered privileged to mess gratitude and respect towards all those who guided us through the completion of this project.

We convey thanks to our project guide **Dr. A.V.Sutagundar** Department of Electronics and Communication Engineering, for their support and advice to get and complete the project on "Smart Irrigation And Pest Detection".

We would like to thank project Co-ordinators **Prof. M. J. Sataraddi** and **Dr. B. M. Angadi** Department of Electronics and Communication Engineering, for providing encouragement, constant support and guidance which was of a great help to complete the project successfully.

We are grateful to **Dr. Jayashree D. Mallapur**, Professor and Head of Department of Electronics and Communication Engineering, for giving us the support and encouragement that was necessary for completion of this project.

We would also like to express our gratitude to **Dr. B. R. Hiremath** Principal, Basaveshwar Engineering College for providing us congenial environment to work in.

We grateful to faculty and staff members of Department of Electronics and Communication Engineering, for providing the facilities which have helped in making this project a success.

Finally, we would like to thank our parents, friends and who helped to complete this project directly and indirectly, for their constant encouragement with moral and material support.

INDEX

SL NO.	TITLE	PAGE NO.
1.	Introduction	6
2.	Problem definition	6-7
3.	Objectives	7-8
4.	Literature review	8-9
5.	Methodology	9-12
6.	Block diagram	13
7.	Tools and techniques used	13-28
8.	Results- Work done	29
9.	References	29

Introduction

Agriculture is one of the major domains of economics of growing interest to be integrated with a smart system. One of the major processes for smart integration is irrigation. Over 90% of the overall growth and development of a healthy plant depends on the quantity of water provided and the micro and macro nutrients required. Therefore, by integrating it with a smart system these minute details can be monitored and maintained to gain high quantity and quality of produce. This simple upgrade can reduce the labour and time used by the farmer to maintain and water a large field and also helps in reducing water loss by providing precise amount of water such that the plant is neither waterlogged nor under-nourished.

The second most important most important factor is pest or insect control. Even with proper nutrients and water availability a good produce of a crop can be destroyed by pest if left unattended. Therefore, to detection of these pests and harmful are equally important to get good produce. Though these pests can be detected, it is extremely difficult to pinpoint which plant are infected without using high grade sensors. Hence, a warning or detection system can be used to alert the farmer in case of pest invasion and provide necessary instruction on how to handle it. This can reduce chance of produce being destroyed by the pest and help in the farmer detecting these pests early so that proper steps can be taken to mitigate it.

Problem definition

2.1 Problem statement

This project aims to reduce the water loss during irrigation. This also ensures that the growth and yield of the crop is as optimum as ever by ensuring controlled water and pest detection. This also reduce the labour work as the farmer can manage the crops from the comfort of his house.

The rise of global population has put increasing pressure on the agriculture industry to meet the demand for food. However, the growing use of pesticides and insecticides in conventional farming practices has caused significant harm to the environment and human health. Thus, there is a growing interest in using sustainable agriculture practices that reduce the use of these harmful chemicals. One such practice is pest detection, which enables farmers to detect pests in their crops before they cause significant damage.

2.2 Background information

Both these steps are important to ensure a proper growing field which requires less labour which helps the farmer save up on money as well as reach the required standards of quality and quantity of food that satisfies the people as well as increase the economy of the country.

Irrigation is one of the oldest practices in the world and now that there is a population boom the amount of food required has shot up exponentially but the land available for cultivation is decreasing due to industrialisation or over exploitation of soil. To make use of the given plot of land and utilise it to the maximum cannot be achieved with traditional methods.

To counter this, lab grown plants from tissue culture and genetically modified crops were used which increased the yield and quality but still the problem persisted ass it required huge quantity of water, pesticide and insecticides which was harmful.

Then came aquaponics which was a break through in science but it would be extremely costly to maintain crops as well as fish with high end materials.

As the rise of AI came about, most of the items started getting ‘smart’ due to its integration into day to day uses. Similarly, smart or IoT based irrigation system became a popular choice among farmers as it was cheap and reduced the time they had to spend in the field and even increase the yield and security of the plant.

Objectives

3.1 Primary objectives:

- Minimize water waste by delivering the right amount of water to crops precisely when and where it is needed.
- Ensure that crops receive adequate moisture to support healthy growth and maximize yield potential.

- Real time data of soil moisture content, temperature, humidity and water level in the tank is provided to the farmer.
- Reduce the labour for farmer by automating the process of irrigation.
- Prevent loss of crops due to pest attacks by detecting pests in the field and warning the farmer about it.
- Providing a chatbot which gives precise information of any question related to agriculture so that the farmer can take advice and suggestion about anything that he is unsure of.

3.2 Secondary objectives:

- This system has an integrated camera with 24/7 live stream with motion detection, hence this can act as a security camera as well which detects human or animal invaders.
- Chatbot can also be used to search other information but it might not be as accurate to the question relating to agriculture.

Literature review

1. Title: Smart Irrigation Systems: Overview and Challenges
   - Authors: S. Sharma and R. Kumar
   - Summary: Explores the benefits and challenges of smart irrigation systems, emphasizing their efficiency and adaptability.
2. Title: Affordable Solar-Powered Irrigation for Precision Agriculture
   - Authors: J. Smith, S. Johnson, and M. Kumar
   - Summary: Proposes a cost-effective solar-powered irrigation system designed to meet the needs of precision agriculture.

3. Title: Pest Detection and Recognition: An Approach Using Deep Learning Techniques
    - Authors: Not Mentioned (Conference Publication)
    - Summary: Demonstrates how deep learning models can detect and classify pests, aiding in improved agricultural yield management.
4. Title: Machine Learning for Plant Leaf Disease Detection and Classification—A Review
    - Authors: L. Sherly Pushpa Annabel, T. Annapoorani, and P. Deepalakshmi
    - Summary: Reviews machine learning methods for identifying and classifying plant leaf diseases, facilitating early diagnosis and prevention.

Methodology

5.1 Approach

For irrigation system,

- Collection of Moisture level in soil of various types of crops.
- Preparation and storage of Arduino code to detect moisture in the soil.
- Data is collected from soil via sensor and is provided to the Arduino.
- Data are compared and instructions are provided to the Arduino to turn the motor to pump water if the moisture is low.
- It checks continuously until the moisture level is high and switched off the motor.

For pest detection,

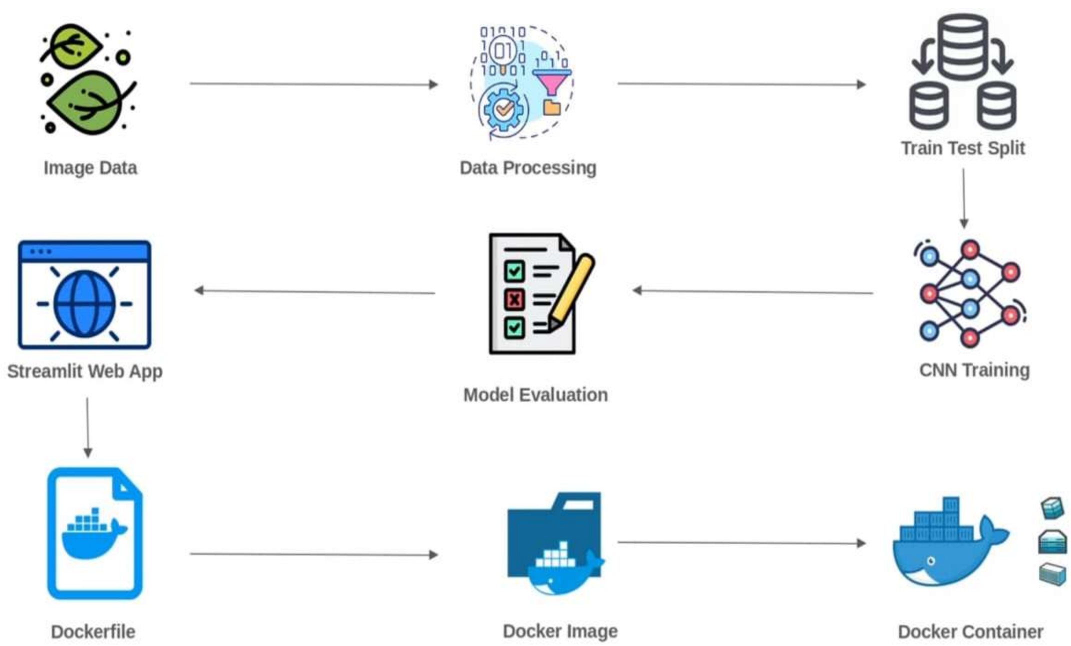
- IR sensor scans at regular interval for movements of pests.
- If a pest is found, buzzer goes off and the camera module shows a livestream of the area.
- The picture is sent to the users mobile via a wifi module.

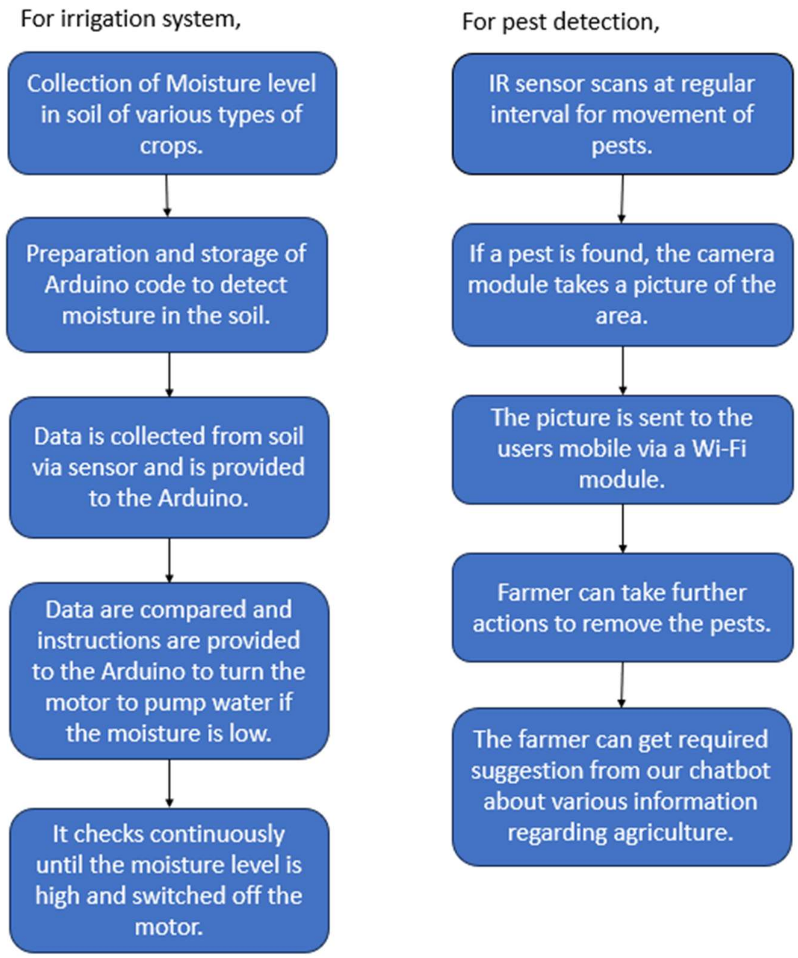
- Farmer can take further actions to remove the pests.
- The farmer can get required suggestion from our chatbot about various information regarding agriculture.

For disease detection,

- Collection of image data
- Data processing of image
- Train Test Split
- CNN training
- Model Evaluation
- Streamlit Web App
- Docker file
- Docker image
- Docker container

For disease detection,





5.2 Procedure

For irrigation system,

First the system reads the soil moisture from which is take from the sensor and sends it to the Arduino UNO bord. Arduino compares the received value to the threshold value and if it greater than the threshold, which indicates the soil is dry, it says that the soil is dry and sends a message to relay station to turn the motor on. The relay station turns the motor for 2 seconds and switches it off. After 10 seconds the moisture is checked again and the process is repeated until the moisture is below the threshold level when the motor is not turned on and no water is pumped. All in while an ultrasonic sensor detects the amount of water present the tank and if it is low, it indicates the user to turn on the motor.

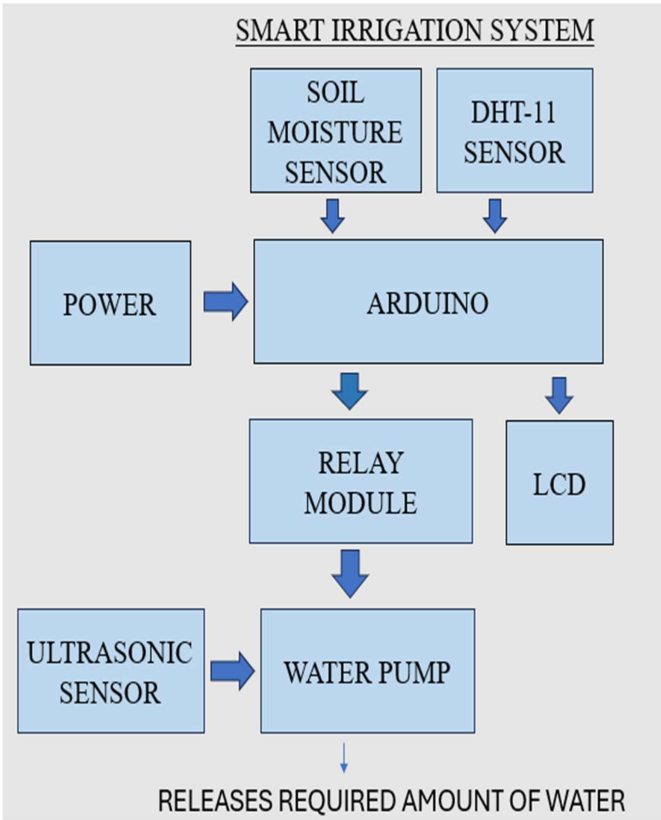
For pest detection,

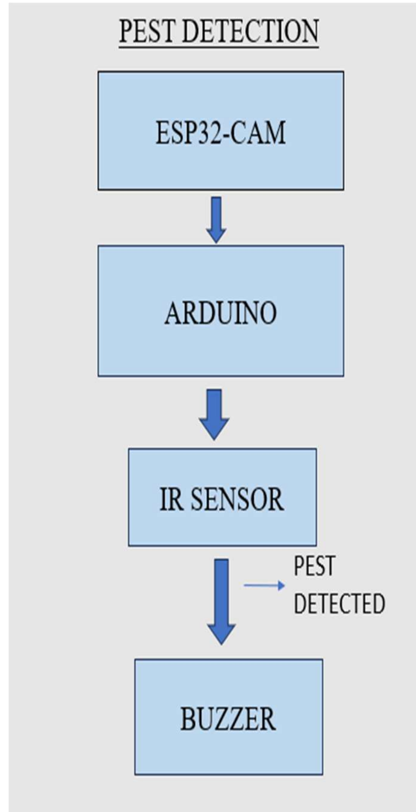
The IR sensor sends out continues beams of light and when an object disturbs the beam it indicates that something is moving and turns the buzzer on to warn the user. The Wi-Fi controlled camera module live streams the area 24/7 and the farmer can look through the camera to check what it is. For further reference he can use the chatbot to find how to get rid of the pest or any other information that he requires.

For disease detection,

The first step in this workflow is gathering the image data, and through various preprocessing techniques such as resizing, augmentation, and labeling, ensuring the data is clean and in a ready state for a model to be trained on. Once the data has been preprocessed, the data is then separated into training and testing sets, where the training set is fed to a convolutional neural network which learns image features while the testing set is kept aside for later assessing the model's performance through various metrics such as Accuracy and Precision. After the model has been trained and evaluated, it is deployed into a Streamlit web application that allows users to upload images and get predictions in real time through an interactive interface. To ensure robustness and consistency between environments, a Dockerfile is created to encapsulate the application and its dependencies into a single Docker image, which is then converted into a docker container. This container runs the web application and allows users to access it for accurate and fast predictions. This framework guarantees that the workflow is flexible, efficient, and easy to use.

BLOCK DIAGRAM





Tools and Techniques Used

6.1 Tools:

Tools Used:

- Soil moisture sensor
- Arduino UNO
- Wires
- Motor Pump
- DHT Sensor
- Ultra sonic sensor
- Relay module
- LCD display

- Battery
- Buzzer

SOFTWARE USED AND ITS PURPOSE:

- Arduino IDE:
  We used this integrated development environment for writing, compiling, and uploading the code to the microcontroller.

- LiquidCrystal_I2C Library:
  This library is used to control the LCD display, allowing it to show information like soil moisture, temperature, and humidity.

- DHT Library:
  This library is used to interface with the DHT11 sensor to read humidity and temperature.

HARDWARE USED AND ITS PURPOSE

- ESP32-CAM:
  Acts as the main controller for the irrigation system, reading sensor data, controlling outputs, and handling WiFi communication.

- Soil Moisture Sensor:
  Measures the moisture level in the soil. It helps determine when to turn the water pump on or off.

- DHT11 Sensor:
  Measures the ambient temperature and humidity, providing environmental data for the system.

- Relay Module:
  Controls the water pump. It acts as a switch to turn the pump on or off based on soil moisture levels.

- Water Pump:
  Waters the plants when the soil moisture sensor detects dry soil.

- Buzzer:
  Provides an audible alert when the soil is dry or other conditions are met.

- LCD Display:
  Displays real-time data about the soil moisture, temperature, and humidity.

- LED:
  Indicates when the water pump is activated.

- Ultrasonic Sensor:
  Measures the water level in the tank to ensure there's enough water for the pump to use.

TECHNIQUES AND METHOD USED:

- Sensor Data Reading:
  Collects data from the soil moisture sensor, DHT11 sensor, and ultrasonic sensor to monitor environmental conditions.

- Conditional Logic:
  Uses if-else statements to decide actions based on sensor data, such as turning the water pump on or off and providing alerts.

- Display Information on LCD:
  Shows current soil moisture, temperature, and humidity readings on the LCD for easy monitoring.

- Buzzer Alerts:
  Provides an audible alert when the soil is dry to draw immediate attention.

EXPLAINATION FOR WHY THESE TECHNIQUES WERE USED:

- Sensor Data Reading:
  - To obtain real-time environmental data for informed decision-making.
  - Analog and digital readings from sensors are processed in the loop to continuously monitor conditions.

- Conditional Logic:
  - To automate responses to changing environmental conditions.
  - If-else statements control the relay and buzzer based on sensor thresholds.

- Display Information on LCD:
  - To provide a user-friendly interface for monitoring the system.
  - The LCD displays relevant data updated regularly in the loop.

- Buzzer Alerts:
  - To ensure immediate attention is drawn when critical conditions are detected.
  - The buzzer is activated when soil moisture falls below a certain threshold, indicating dry soil.

TOOLS AND SOFTWARE USED FOR PEST AND SURVEILLANCE:

LIST OF TOOLS:

- 1.ESP32-CAM
- 2.WiFi Library
- 3.IR Sensor
- 4.Buzzer

SOFTWARE USED AND ITS PURPOSE:

- Arduino IDE:
  - Used for coding and uploading programs to the ESP32-CAM.

- WiFi Library:
  - Manages WiFi connectivity for the ESP32-CAM, enabling remote monitoring and control.

HARDWARE USED AND ITS PURPOSE:

- ESP32-CAM:
  - Main microcontroller for image capturing and WiFi communication.

- IR Sensor:
  - Detects pests by sensing motion in the monitored area.

- Buzzer:
  - Provides an alert when pests are detected.

- TECHNIQUES AND METHOD USED:

- Camera Initialization and Configuration:
  - Sets up the camera to capture images and stream them over the network.

- WiFi Connectivity:
  - Connects the ESP32-CAM to a WiFi network for remote monitoring.

- Pest Detection with IR Sensor:
  - Detects motion indicating the presence of pests and triggers an alert.

EXPLAINATION FOR WHY THESE TECHNIQUES WERE USED:

- Camera Initialization and Configuration:
  - To enable image capturing for monitoring the field visually.
  - Camera settings are configured in the setup function to capture high-quality images.

- WiFi Connectivity:
  - To allow remote monitoring and control of the system.
  - WiFi credentials are used to connect to the network, enabling access to the camera feed from a web interface.

- Pest Detection with IR Sensor:
  - To provide a simple and effective way to monitor for pests.
  - The IR sensor detects motion, and the buzzer provides an immediate alert if pests are detected.

SOFTWARE USED FOR DISEASE DETECTION:

- Google Colab (Collaboratory): It is a free, cloud-based platform by Google.
- Use of image dataset from Kaggle.
- We used it to create model using python code.
- PyCharm: It is a powerful integrated development environment (IDE) developed by JetBrains, specifically designed for Python development.
- We deployed the web app using PyCharm and Streamlit..

EXPLAINATION FOR WHY THESE SOFTWARE WERE USED:

- Google Colab is a free cloud platform that gives you access to GPUs/TPUs, needs no installation of libraries and has a Jupyter Notebook environment to write and run Python code. It also supports collaboration, has seamless Google Drive integration and is widely used by data scientists for machine learning, data science and research purposes.

- Kaggle provides a vast collection of datasets for machine learning, data analysis, and research purposes.

- PyCharm is ideal for building web apps with support for frameworks like Django and Flask, intelligent code completion, and debugging tools. It also offers seamless database integration and version control for efficient project management.

Code for Smart Irrigation

```
#include<LiquidCrystal_I2C.h>
#include<Wire.h>
#include<DHT.h>
#define dhtpin 9
#define dhttype DHT11

DHT dht(dhtpin,dhttype);
LiquidCrystal_I2C lcd(0x27,16,2);

int sensor_pin=A0;
int relay_pin = 7;
int SDA_pin=A4;
int SCL_pin=A5;
int Buzzer=2;
int LED_pin=3;
int trigPin=13;
int echoPin=12;

void setup() {
  Serial.begin(9600);
  lcd.init();  // Initialize the LCD
  delay(50);   // Add a short delay
  lcd.backlight();
  dht.begin();
  pinMode(sensor_pin, INPUT);
  pinMode(relay_pin, OUTPUT);
  pinMode(echoPin,INPUT);
  pinMode(trigPin,OUTPUT);
  pinMode(Buzzer,OUTPUT);
```

```
  digitalWrite(relay_pin, HIGH);
  delay(100);
}
void loop() {
  int sensor_data = analogRead(sensor_pin);
  Serial.print("Sensor_data: MOISTURE = ");
  Serial.print(sensor_data);
  Serial.print("\t | ");
//dht sensor
float humi =dht.readHumidity();
float tempC=dht.readTemperature();
Serial.print("Humidity is : ");
Serial.print(humi);
Serial.print("%");
Serial.print("   |   ");
Serial.print("Temperature is : ");
Serial.print(tempC);
Serial.print(" Degree Celcius\n");
delay(1000);
//moisture level &pumping of water
if(sensor_data > 580)
{
    Serial.println("No moisture, Soil is dry");
     // Update LCD display
    lcd.setCursor(0,0);
    lcd.print("Soil Dry, M ON");
    lcd.setCursor(0,1);
    lcd.print("Tem:");
    lcd.print(tempC);
    lcd.print(" hum:");
```

```
    lcd.print(humi);
    digitalWrite(Buzzer,HIGH);
    delay(100);
    digitalWrite(Buzzer,LOW);
    digitalWrite(LED_pin,HIGH);
    // Turn the pump on
    digitalWrite(relay_pin, LOW);
    delay(800);
    digitalWrite(relay_pin, HIGH);
    delay(3000);
    digitalWrite(LED_pin,LOW);
}
  else if(sensor_data >= 400 && sensor_data <= 580)
  {
    Serial.println("There is some moisture, Soil is medium");
    digitalWrite(relay_pin, HIGH);
    lcd.setCursor(0,0);
    lcd.print("Soil Medium,M OFF");
    lcd.setCursor(0,1);
    lcd.print("Tem:");
    lcd.print(tempC);
    lcd.print(" hum:");
    lcd.print(humi);
      }
  else if(sensor_data < 400)
  {
    Serial.println("Soil is wet");
    digitalWrite(relay_pin, HIGH);
    lcd.setCursor(0,0);
    lcd.print("Soil Wet,M OFF  ");
```

```
   lcd.setCursor(0,1);
   lcd.print("Tem:");
   lcd.print(tempC);
   lcd.print(" hum:");
   lcd.print(humi);
 }
 delay(5000);
 //ultrasonic sensor
digitalWrite(trigPin,HIGH);
delayMicroseconds(10);
digitalWrite(trigPin,LOW);
int duration_us=pulseIn(echoPin,HIGH);
int distance_cm=0.017*duration_us;
Serial.print("distance from base of water tank : ");
Serial.print(distance_cm);
Serial.print(" cm");
Serial.print("\n");
delay(500);
if(distance_cm>13)
{
 Serial.print("Water tank has less water !");
 Serial.print("\n");
 Serial.print("\n");
}
else{
 Serial.print("Water tank has sufficient water");
 Serial.print("\n");
 Serial.print("\n");
}
}
```

Code for Pest Detection

```
#include "esp_camera.h"
#include <WiFi.h>

// Camera Model Selection
#define CAMERA_MODEL_AI_THINKER // Has PSRAM
#include "camera_pins.h"

// WiFi Credentials
const char *ssid = "MINIPROJECT";
const char *password = "Mp@2024";

// IR Sensor and Buzzer Pins
#define IR_SENSOR_PIN 15
#define BUZZER 14

void startCameraServer();
void setupLedFlash(int pin);

void setup() {
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  Serial.println();

  // Initialize IR Sensor and Buzzer
  pinMode(IR_SENSOR_PIN, INPUT);
  pinMode(BUZZER, OUTPUT);
  digitalWrite(BUZZER, LOW);

  // Camera Configuration
```

```
camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sccb_sda = SIOD_GPIO_NUM;
config.pin_sccb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
config.frame_size = FRAMESIZE_SVGA;
config.jpeg_quality = 12;
config.fb_count = 1;

if (psramFound()) {
  config.jpeg_quality = 10;
  config.fb_count = 2;
  config.grab_mode = CAMERA_GRAB_LATEST;
} else {
```

```
  config.frame_size = FRAMESIZE_QVGA;
  config.fb_location = CAMERA_FB_IN_DRAM;
}

esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
  Serial.printf("Camera init failed with error 0x%x\n", err);
  return;
}

sensor_t *s = esp_camera_sensor_get();
if (s->id.PID == OV3660_PID) {
  s->set_vflip(s, 1);
  s->set_brightness(s, 1);
  s->set_saturation(s, -2);
}
s->set_framesize(s, FRAMESIZE_QVGA);

WiFi.begin(ssid, password);
WiFi.setSleep(false);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("\nWiFi connected");

startCameraServer();

Serial.print("Camera Ready! Use 'http://");
Serial.print(WiFi.localIP());
```

```
  Serial.println("' to connect");
}

void loop() {
  int irValue = digitalRead(IR_SENSOR_PIN);

  if (irValue == LOW) {
    Serial.println("Pest Detected!");
    digitalWrite(BUZZER, HIGH);
  } else {
    Serial.println("No Pest Detected.");
    digitalWrite(BUZZER, LOW);
  }

  delay(5000); // Check every 5 seconds
}
```

Code for Plant Disease Classifier

```
import os
import json
from PIL import Image

import numpy as np
import tensorflow as tf
import streamlit as st


working_dir = os.path.dirname(os.path.abspath(__file__))
model_path = f"{working_dir}/trained_model/plant_disease_prediction_model.h5"
# Load the pre-trained model
model = tf.keras.models.load_model(model_path)

# loading the class names
class_indices = json.load(open(f"{working_dir}/class_indices.json"))
```

```
# Function to Load and Preprocess the Image using Pillow
def load_and_preprocess_image(image_path, target_size=(224, 224)):
    # Load the image
    img = Image.open(image_path)
    # Resize the image
    img = img.resize(target_size)
    # Convert the image to a numpy array
    img_array = np.array(img)
    # Add batch dimension
    img_array = np.expand_dims(img_array, axis=0)
    # Scale the image values to [0, 1]
    img_array = img_array.astype('float32') / 255.
    return img_array


# Function to Predict the Class of an Image
def predict_image_class(model, image_path, class_indices):
    preprocessed_img = load_and_preprocess_image(image_path)
    predictions = model.predict(preprocessed_img)
    predicted_class_index = np.argmax(predictions, axis=1)[0]
    predicted_class_name = class_indices[str(predicted_class_index)]
    return predicted_class_name


# Streamlit App
st.title('Plant Disease Classifier')

uploaded_image = st.file_uploader("Upload an image...", type=["jpg", "jpeg", "png"])

if uploaded_image is not None:
    image = Image.open(uploaded_image)
    col1, col2 = st.columns(2)

    with col1:
        resized_img = image.resize((150, 150))
        st.image(resized_img)

    with col2:
        if st.button('Classify'):
            # Preprocess the uploaded image and predict the class
            prediction = predict_image_class(model, uploaded_image, class_indices)
            st.success(f'Prediction:  {str(prediction)}')
```

AgriGenie - A Farmers Feed:

AgriGenie is an AI-powered chatbot designed specifically to support farmers by providing precise and relevant information related to agriculture. Developed using BotPress, AgriGenie serves as a virtual assistant that can answer a wide range of agricultural questions and provide detailed, context-specific advice.

PURPOSE AND FEATURES:
To assist farmers with accurate and immediate information to improve their farming practices and decision-making processes.

1.Real-time Consultation:

- Farmers can ask questions and receive instant answers related to their query.
- Precise answers are given out as we have fed certain authenticated agricultural books.

2.Crop Recommendations:

- Suggests best practices for growing and maintaining different crops based on the latest agricultural research and current conditions.

3. Pest and Disease Management:

- Offers advice on identifying and managing pests and diseases that affect crops, including recommended treatments and preventive measures.

4.Irrigation and Soil Management:

- Provides guidelines on efficient irrigation practices and soil management techniques to enhance crop productivity.

DEVELOPMENT:

BotPress:

This platform was chosen for its user-friendly interface and strong features for developing conversational AI. BotPress allows for the integration of various functionalities and provides a great user experience for farmers.

TYPES OF QUESTIONS ASKED:

1.Crop Management:

- "How do I improve the yield of my wheat crop?"

- "What is the best time to plant maize in my region?"

2. Irrigation Practices:

- "What is the best irrigation method for my sandy soil?"

3.Soil Health:

- "What is the ph of the black soil?"

HOW IS IT DIFFERENT FROM OTHER AI:

Precise Information directly from Agricultural Books.

To ensure the chatbot provides precise and accurate information, we uploaded a collection of agriculture-related books and resources into its knowledge base. This approach ensures that the answers are detailed and specific, unlike the generalized responses you might get from other AI chatbots like ChatGPT.

BENEFITS:

- Accuracy: Answers are based on the latest and most reliable agricultural research and publications.
- Specific: Responses are related to the particular crops, pests, and conditions described in the questions.
- Practical Advice: The chatbot offers actionable recommendations that farmers can implement directly in their fields.

AgriGenie is designed to be a reliable companion for farmers, providing them with the knowledge and support they need to enhance their farming practices and achieve better yields. With its basic AI capabilities and agricultural knowledge-based knowledge fed into it, AgriGenie can be a good help for the farmers.

If interested here is a link for AgriGenie:

https://mediafiles.botpress.cloud/f5ccd271-d423-4799-ab9e-15a01973cc8d/webchat/bot.html

RESULT- WORK DONE

- The automatic irrigation control using Arduino UNO has been experimentally proven to work satisfactorily and we could successfully set the timer and managed to control the motor over time.
- This process not only records values of temperature and humidity it also controls the motor accordingly.
- Analyzing the weather condition motor will automatically maintain water supply making it possible to maintain greenery without human intervention.
- Additionally, by storing and analyzing the pest detection data over time, farmers can gain insights into trends and patterns in pest activity, allowing them to make informed decisions about pest control strategies.
- Providing an AI Chatbot which gives precise information of any question related to agriculture.
- We also built Deep Learning based Plant-Disease Pediction with image classification using CNN (Convolution Nueral Networks).

REFERENCES

- Dr . Narayan G. Hegde, “Water Scarcity and Security in India”, BAIF Development Research Foundation, Pune.
- Marvin T. Batte, “Changing computer use in agriculture: evidence from Ohio”, Computers and Electronics in Agriculture, Elsevier science publishers, vol. 47, 1–13, 2005
- Embedded System for Irrigation”, International Journal of Computer Applications Vol. 47 – No.13, June 2012
- "RanaVision System for Motion-Based Detection of Small Invertebrates"
- Wikipedia Project Summary, [Online].
- https://youtu.be/L-Tqf1w5d0I?si=MxS1xJir3AKHg4Uk