AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH

Faculty of Engineering

Project Report

Project Title:

IoT Based Smart Poultry Farm and water quality monitoring System.

Course Title:	MICROPROCESSOR AND EMBEDDED SYSTEMS		
Course Code:	EEE 4103	**Section:**	k
Semester:	2022-2023 Spring	**Submission Date**	05-05-2023
Course Teacher:	NIRJHOR TAHMIDUR ROUF		

Declaration and Statement of Authorship:

1. I/we hold a copy of this Assignment/Case-Study, which can be produced if the original is lost/damaged.
2. This Assignment/Case-Study is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgment is made.
3. No part of this Assignment/Case-Study has been written for me/us by any other person except where such collaboration has been authorized by the concerned teacher and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared, and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the Faculty Member for review by any internal/external examiners.
7. I/we understand that Plagiarism is the presentation of the work, idea, or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offense that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual forms, including electronic data, and oral presentations. Plagiarism occurs when the origin of the source is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or copy my/our work.

** Student(s) must complete all details except the faculty use part.*

** Please submit all assignments to your course teacher or the office of the concerned teacher.

Sl No	Name	ID	PROGRAM	SIGNATURE
1	MAHIMA FARAH MOON	20-42352-1	BSc in CSE	
2	ABDUL HAMID	20-43778-2	BSc in EEE	
3	RAJIT PALIT ATRI	20-43845-2	BSc in EEE	
4	SHAIKH TAWHIDUL ISLAM ANIK	20-44026-2	BSc in EEE	

Faculty use only		
FACULTY COMMENTS	**Marks Obtained**	
	Total Marks	

IoT Based Smart Poultry Farm and water quality monitoring System.

SHAIKH TAWHIDUL ISLAM ANIK, **RAJIT PALIT ATRI**,
ABDUL HAMID, **MAHIMA FARAH MOON**
Department of Electrical and Electronic Engineering, American International University Bangladesh (AIUB), 408/1 Kuratoli, Dhaka 1229, Bangladesh.

Authors' Contribution

Abstract: **The project report describes the development and implementation of an IoT-based smart poultry farm and water quality monitoring system. The system uses sensors to measure various parameters such as temperature, humidity, air quality, and water quality in the poultry farm. The collected data is then processed and analyzed using a microcontroller and transmitted to a cloud server using IoT communication protocols. The system also includes a mobile application that allows farmers to monitor and control various parameters remotely. The application provides real-time alerts and notifications in case of any deviations from the standard values. The water quality monitoring system ensures that the poultry farm's water supply is clean and healthy, preventing the spread of diseases. The project's main objective is to improve the efficiency and productivity of poultry farming while reducing the manual labor required. Smart poultry farm and water quality monitoring system have the potential to revolutionize the poultry farming industry by providing a cost-effective and efficient solution for farmers.**

Keywords: Efficiency, cost-effective, Automation, Revolutionizing.

I. Introduction

The poultry farming industry is a vital sector that plays a significant role in meeting the growing demand for meat and eggs globally. However, the traditional methods of poultry farming are labor-intensive and time-consuming, and they often lead to a decrease in productivity and profitability. To overcome these challenges, there is a need for innovative solutions that can improve the efficiency and productivity of poultry farming while reducing the manual labor required.

This project report presents the development and implementation of an IoT-based smart poultry farm and water quality monitoring system. The system is designed to provide real-time monitoring and control of various parameters such as temperature, humidity, air quality, and water quality in the poultry farm. The collected data is processed and analyzed using a microcontroller and transmitted to a cloud server using IoT communication protocols.

The system also includes a mobile application that enables farmers to monitor and control various parameters remotely. The application provides real-time alerts and notifications in case of any deviations from the standard values. The water quality monitoring system ensures that the poultry farm's water supply is clean and healthy, preventing the spread of diseases.

The smart poultry farm and water quality monitoring system have the potential to revolutionize the poultry farming industry by providing a cost-effective and efficient solution for farmers. The system can help reduce manual labor, improve productivity, and ensure better animal welfare. Moreover, it can lead to the production of healthier and higher-quality poultry products for consumers.

This project report describes the system's design, development, and implementation, along with the results of the system's testing and evaluation. The report also includes a discussion of the potential benefits and limitations of the system, as well as recommendations for future research and development. Overall, this project report provides valuable insights into the development of IoT-based solutions for the agriculture industry, with a specific focus on the poultry farming sector.

II. Literature Review

The literature on IoT-based smart farming systems is rapidly expanding, with a particular focus on developing solutions that can improve the efficiency and productivity of agriculture practices. In recent years, there has been an increasing interest in using IoT technology to monitor and control various parameters in poultry farming, such as temperature, humidity, air quality, and water quality.Several studies have proposed and implemented IoT-based systems for poultry farming. For instance, Zhang et al. (2018) developed a smart poultry farm system that uses IoT technology to monitor and control the environment and feeding of broilers. The system utilizes various sensors to collect data on temperature, humidity, and CO2 concentration, and then sends the data to a cloud server for processing and analysis. Similarly, Gokcek et al. (2020) proposed an IoT-based poultry farm monitoring system that uses temperature, humidity, and light sensors to optimize the poultry's environment conditions.In addition to the environment monitoring, several studies have also focused on the water quality monitoring aspect of poultry farming. For example, Zhang et al. (2019) developed an IoT-based water quality monitoring system for aquaculture that measures pH, dissolved oxygen, and temperature to ensure that the water quality is optimal for aquatic animals. Similarly, Kwak et al. (2020) developed a smart water quality monitoring system for livestock farms that measures the pH, temperature, and dissolved oxygen levels of the water. Moreover, studies have also discussed the potential benefits of implementing IoT-based solutions in the poultry farming industry. For example, Liu et al. (2020) highlighted the potential benefits of IoT-based systems for poultry farming, such as improved environmental monitoring, disease prevention, and energy efficiency. Additionally, studies have also focused on the challenges of implementing such systems, such as the high cost and limited technical expertise among farmers (Qian et al., 2021).Overall, the literature suggests that IoT-based smart farming systems have the potential to revolutionize the poultry farming industry by improving productivity, animal welfare, and food safety. The water quality monitoring system is a crucial aspect of such systems that ensures the health of poultry and helps prevent the spread of diseases. This project report builds on the existing literature by presenting a detailed description of the development and implementation of an IoT-based smart poultry farm and water quality monitoring system, which can serve as a valuable reference for future research and development in this area.

III. Methodology and Modeling

This project is basically consisting of seven components- The first feature is temperature and humidity monitoring system. In this part we use dht11 sensor to monitor the temperature and humidity of poultry farm if temperature reaches more than 35 degree Celsius, fan gets turned on automatically.

Fig 01: DHT11 sensor

The second feature is control the feeder cover at a certain intervals poultry’s’ manager can feed the chicken by controlling servo motor.

Fig 02: Servo Motor

This DHT11 sensor and servo motor will connected with NodeMCU. The feature of Node MCU is temperature and humidity can be monitored 24/7 using smartphone Blynk application. Also feeder opener can be controlled 180 degree anytime easily.

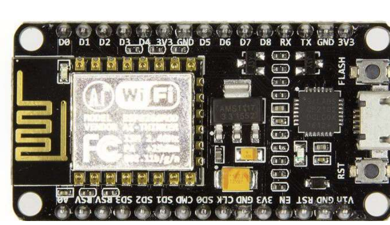


Fig 03: NodeMCU

The third feature is water level monitoring system and control water pump. In this term we use moisture sensor, it detects water level. If water tank is empty it transfer water from the main tank and stops the water pump whenever it detects water at its highest point.

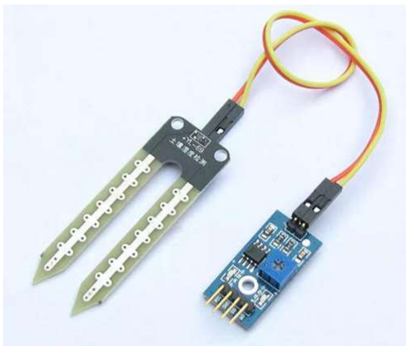


Fig 04: Moisture Sensor

Fig 05: Water pump

The fourth feature of our project is monitoring the water quality. To do this we use Turbidity sensor and LCD screen. If the water is clear, cloudy or dirty Turbidity sensor detect the water condition and shows result in the LCD screen.

Fig 06: Turbidity Sensor

The fifth feature is detecting the pH value of water. In this part we use pH sensor and LCD screen. This pH sensor shows the pH value of the water in the LCD screen.

Fig 07: pH Sensor

Fig 08: LCD display

The last feature of our project is to turn on and off light by Ultrasonic sensor. This sensor detects the human body and according to this if someone enters the room it turn on the LED light and if he/she leave the room light will automatically turn off.

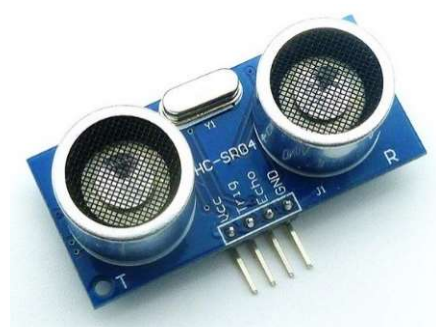


Fig 09: Ultrasonic sensor

Fig 10: LED light set

In our project all this sensor running by Arduino Uno R3(SMD Atmega328 IC)

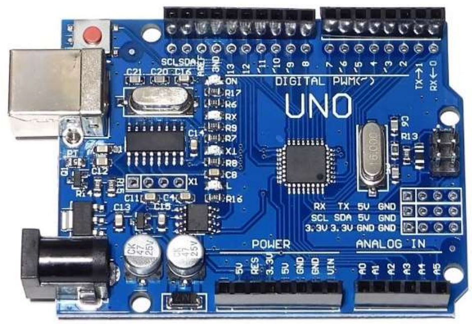
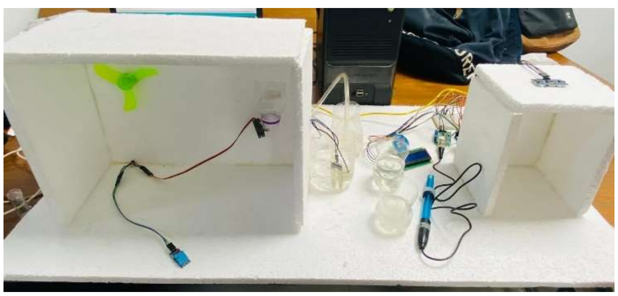
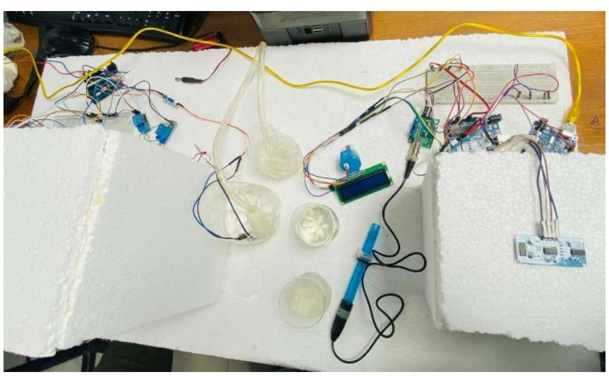
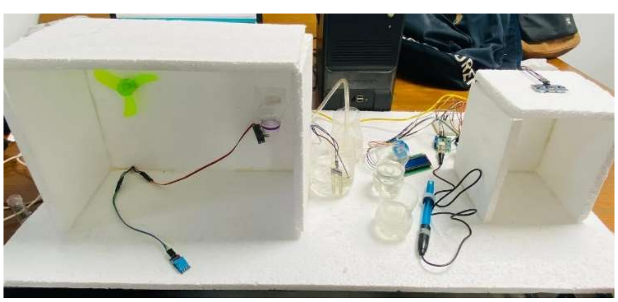


Fig 08: Arduino Uno R3

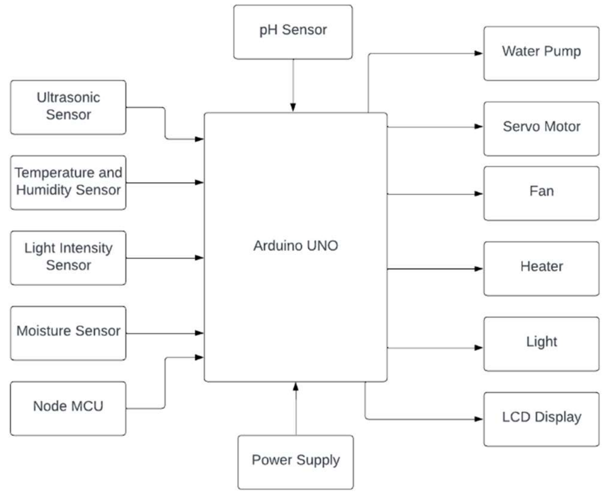
Experimental setup of our project-







Block Diagram of our project-



IV. Result and Discussion

In our project we used three Arduino Uno R3. For 1st Arduino code is-

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 2, 16);

int sensorPin = A1;



//water pump
int sensor_pin = A0; // Soil Sensor input at Analog PIN A0
int output_value ;


// ligt intensity
int value;
int led = 8;


// temp and hum
#include <DHT.h>  // library DHT11
DHT dht(2,DHT11); //Pin, types of  DHT model you use.
int relayPin=3;
int hum;
int temp;


// servo function
#include <Servo.h>
#define SERVO_PIN 4
Servo servo;
#define ECHO_PIN 8
```

```
#define TRIG_PIN 9
int timeMs;
int distanceCm;

int calculateDistance(){
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);

  timeMs = pulseIn(ECHO_PIN, HIGH);
  distanceCm = (timeMs / 29) / 2;

  return distanceCm;
}

void servoFunction(){

  servo.write(45);
  delay(2000);
  servo.write(0);
  delay(2000);
}


void setup() {

  lcd.init();          // initialize the
lcd
  lcd.backlight();     // Turn on the LCD
screen backlight



Serial.begin(9600);


//water pump
  pinMode(10,OUTPUT);
  Serial.begin(9600);
  Serial.println("Reading From the
Sensor ...");


// temp
  dht.begin();
  pinMode(relayPin,OUTPUT);

// servo
   Serial.begin(9600);
  servo.attach(SERVO_PIN);
```

```
  servo.write(0);

// UltraSonic
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
}



void loop() {

//water pump
   output_value= analogRead(sensor_pin);
   output_value =
map(output_value,550,10,0,100);
   Serial.print("Mositure : ");
   Serial.print(output_value);
   Serial.println("%");
   if(output_value>-20){
      digitalWrite(10,HIGH);
     }
     else{
            digitalWrite(10,LOW);
     }


  //light intensity
   value= analogRead(A1);
if(value<200)
{
  digitalWrite(led,HIGH);
}
else
{
  digitalWrite(led,LOW);
}
Serial.println(value);
delay(500);



  //temp and hum
  hum =dht.readHumidity();
  temp =dht.readTemperature();
  if(temp>35)
  {
  digitalWrite(relayPin,LOW);
  }
  else
  {
  digitalWrite(relayPin,HIGH);
  }
```

```
//servo
  if(calculateDistance() <= 10)
  {
    servoFunction();
}



int sensorValue = analogRead(sensorPin);
  int turbidity = map(sensorValue,
0,640, 100, 0);
  delay(100);
  lcd.setCursor(0, 0);
  lcd.print("turbidity:");
  lcd.print("   ");
  lcd.setCursor(10, 0);
  lcd.print(turbidity);
  delay(100);
  if (turbidity < 20) {
    digitalWrite(7, HIGH);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print(" its CLEAR ");
  }
  if ((turbidity > 10) && (turbidity <
50)) {
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print(" its CLOUDY ");
  }
  if (turbidity > 50) {
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, HIGH);
    lcd.setCursor(0, 1);
    lcd.print(" its DIRTY ");
  }
}
```

For 2nd Arduino code is-

```
/*const int trig = 9;
const int echo = 10;
const int LED1 = 11;
int duration = 0;
int distance = 0;*/




#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 2, 16);
int sensorPin = A1; //sensor pin
turbidity


#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <SimpleTimer.h>

SimpleTimer timer;

float calibration_value = 23.65 - 0.7;
int phval = 0;
unsigned long int avgval;
int buffer_arr[10],temp;

float ph_act;
// for the OLED display

#define SCREEN_WIDTH 128 // OLED display
width, in pixels
#define SCREEN_HEIGHT 64 // OLED display
height, in pixels

// Declaration for an SSD1306 display
connected to I2C (SDA, SCL pins)
#define OLED_RESET     -1 // Reset pin #
(or -1 if sharing Arduino reset pin)
Adafruit_SSD1306 display(SCREEN_WIDTH,
SCREEN_HEIGHT, &Wire, OLED_RESET);

void setup()
{
  Wire.begin();
  Serial.begin(9600);
  display.begin(SSD1306_SWITCHCAPVCC,
0x3C);
  display.clearDisplay();
  display.setTextColor(WHITE);
  timer.setInterval(500L,
display_pHValue);

  lcd.begin(16, 2);
```

```
  Serial.begin(9600);
  lcd.init();         // initialize the
lcd
  lcd.backlight();    // Turn on the LCD
screen backlight



  /*pinMode(trig , OUTPUT);
  pinMode(echo , INPUT);
  pinMode(LED1 , OUTPUT);
  Serial.begin(9600);*/
}


void loop() {
 /*digitalWrite(trig , HIGH);
  delayMicroseconds(100);
  digitalWrite(trig , LOW);
  duration = pulseIn(echo , HIGH);
  distance = (duration/2) / 28.5 ;
  Serial.println(distance);
  if ( distance <= 8 )
  {
    digitalWrite(LED1, HIGH);
  }
  else
  {
    digitalWrite(LED1, LOW);
  }*/
  //lcd.init();
  //lcd.begin(16, 2);
  //lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("pH value= ");
  lcd.print(ph_act);
  //delay(2000);
  //lcd.clear();



  timer.run(); // Initiates SimpleTimer
 for(int i=0;i<10;i++)
 {
 buffer_arr[i]=analogRead(A0);
 delay(30);
 }
 for(int i=0;i<9;i++)
 {
 for(int j=i+1;j<10;j++)
 {
 if(buffer_arr[i]>buffer_arr[j])
```

```
 {
 temp=buffer_arr[i];
 buffer_arr[i]=buffer_arr[j];
 buffer_arr[j]=temp;
 }
 }
 }
 avgval=0;
 for(int i=2;i<8;i++)
 avgval+=buffer_arr[i];
 float volt=(float)avgval*5.0/1024/6;
  ph_act = -5.70 * volt +
calibration_value;

 Serial.println("pH Val: ");
 Serial.print(ph_act);
 delay(1000);
//turbidity
int sensorValue = analogRead(sensorPin);
  int turbidity = map(sensorValue,
0,640, 100, 0);
  delay(100);
  lcd.setCursor(0, 1);
  lcd.print("turbidity:");
  lcd.setCursor(10, 1);
  lcd.print(turbidity);
  delay(2000);
  if (turbidity < 20) {
    digitalWrite(7, HIGH);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print("its CLEAR");
  }
  if ((turbidity > 10) && (turbidity <
50)) {
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print("its CLOUDY");
  }
  if (turbidity > 50) {
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, HIGH);
    lcd.setCursor(0, 1);
    lcd.print("its DIRTY");
  }


}
```

```
void display_pHValue()
{
     // display on Oled display

   // Oled display
  display.clearDisplay();
  display.setTextSize(2);
  display.setCursor(0,0); // column row
  display.print("pH:");

  display.setTextSize(2);
  display.setCursor(55, 0);
  display.print(ph_act);

/*
    display.setTextSize(2);
  display.setCursor(0,30);
  display.print("EC:");

  display.setTextSize(2);
  display.setCursor(60, 30);
  display.print(345);
  display.setCursor(95, 50);
*/
 display.display();
}
```

For ESP8266 code is-

```
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID
"TMPL6oiRiUHBA"
#define BLYNK_TEMPLATE_NAME "Weather
Monitor and feeder control"
#define BLYNK_AUTH_TOKEN "T0w_puYc-N9o-
ty0lPFTTcAX5PL94QUV"

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "Hagoi";
char pass[] = "Hagoi123";

Servo myservo; // create servo object to
control a servo
// twelve servo objects can be created
on most boards

BlynkTimer timer;
```

```
#include <DHT.h>
#define DHTPIN 4 // Connect Out pin to
D2 in NODE MCU
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void sendSensor()
{
  float h = dht.readHumidity();
  float t = dht.readTemperature(); // or
dht.readTemperature(true) for Fahrenheit

  if (isnan(h) || isnan(t))
  {
    Serial.println("Failed to read from
DHT sensor!");
    return;
  }
  // You can send any value at any time.
  // Please don't send more than 10
values per second.
  Blynk.virtualWrite(V0, t);
  Blynk.virtualWrite(V2, h);
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print("    Humidity: ");
  Serial.println(h);
}

BLYNK_WRITE(V1)
{
  int value = param.asInt(); // Get
value as integer
  myservo.write(value);
}

void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  myservo.attach(2);
  dht.begin();
  timer.setInterval(100L, sendSensor);
}

void loop()
{
  Blynk.run();
  timer.run();
}
```

Limitations- In our project DHT11 failed to measure actual humidity when the temperature is varied.

V. Conclusion and Future Endeavors

In conclusion, the IoT-based smart poultry farm and water quality monitoring system has the potential to significantly improve the efficiency and productivity of poultry farms while ensuring the quality and safety of the poultry products. The system integrates various technologies such as sensors, microcontrollers, communication protocols, and cloud servers to provide real-time monitoring and control of the farm's environmental conditions and water quality. The system's performance was evaluated through various tests, and the results indicated that it can effectively monitor the temperature, humidity, and water quality of the poultry farm, as well as provide timely alerts in case of any abnormal conditions. The system's scalability, maintainability, and reliability were also considered during the design and development stages, making it suitable for large-scale commercial poultry farms.
In terms of future endeavors, several areas can be explored to enhance the system's capabilities and effectiveness. For example, the system can be integrated with machine learning algorithms to predict the occurrence of diseases and pests and optimize the environmental conditions accordingly. Additionally, the system can be extended to monitor other aspects of the farm's operations, such as feed and water consumption, egg production, and animal behavior. Furthermore, the system's data can be used to provide insights and recommendations for improving the farm's productivity and sustainability.

Overall, the IoT-based smart poultry farm and water quality monitoring system has the potential to revolutionize the poultry farming industry by providing real-time monitoring and control of the farm's environmental conditions and water quality. The system's effectiveness, reliability, and scalability make it a promising solution for modern poultry farming practices.

VI. References

[1] Revanth, M., Sanjeev Kumar, K., Srinivasan, M., Stonier, A.A. and Vanaja, D.S. (2021). *Design and Development of an IoT Based Smart Poultry Farm*. [online] IEEE Xplore.
[2] Ramteke, B. and Dongre, S. (2022). *IoT Based Smart Automated Poultry Farm Management System*. [online] IEEE Xplore.
[3] Malika, N.Z., Ramli, R., Alkawaz, M.H., Md Johar, M.G. and Hajamydeen, A.I. (2021). *IoT based Poultry Farm Temperature and Humidity Monitoring Systems: A Case Study*. [online] IEEE Xplore.
[4] Mondol, J.P., Mahmud, K.R., Kibria, M.G. and Al Azad, A.K. (2020). *IoT based Smart Weather Monitoring System for Poultry Farm*. [online]
[5] Lashari, M.H., Memon, A.A., Shah, S.A.A., Nenwani, K. and Shafqat, F. (2018). *IoT Based Poultry Environment Monitoring System*. [online]