EXPERIENTIAL ENGINEERING EDUCATION FABRICATION / MODEL DEVELOPMENT

REPORT ON

# SMART IRRIGATION SYSTEM USING IOT

*A Report submitted*

***by***

S.SHREYA SRI - 20951A04J2

S.USHA - 21955A0423

B.HARIKA - 20951A04E8

C.SATYA SREE - 20951A04G9

IARE-New-Logo_Black.png

Name of the Department

# INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad – 500 043, Telangana

June, 2022

EXPERIENTIAL ENGINEERING EDUCATION FABRICATION / MODEL DEVELOPMENT

**Title of your Idea : Smart Irrigation System using IOT**

**Thrust Area / Sector : Internet of Things**

**Branch & Section : ECE - D**

**Year / Semester : 2nd YEAR / 4th SEM**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Name of the Student** | **Roll Number** | **Mobile Number** |
| **1** | S. SHREYA SRI | 20951A04J2 | 9347442961 |
| **2** | S. USHA | 21955A0423 | 9676302515 |
| **3** | B. HARIKA | 20951A04E8 | 9121320406 |
| **4** | C. SATYA SREE | 20951A04G9 | 7729812450 |

1. **Abstract of your Idea:**

As water supply is becoming scarce in today’s world there is an urgency of adopting smart ways of irrigation. The project describes how irrigation can be handled smartly using IOT. This project aims at saving time and avoiding problems like constant vigilance. It also helps in conserving water by automatically providing water to the plants/field depending on the water requirements. This system can also prove to be helpful in agriculture, parks and lawns. The objective of this system is to detect the moisture content of the soil and depending on it sprinkle water .This entire information will be sent to the user’s mobile phone.

1. **Objectives & Significance:**

* The objectives of this paper is to investigate the concept of smart irrigation system using IoT, to develop a system using an Arduino NODEMCU - ESP8266
* It processes the data from the soil sensor which automatically water the plant and to analyse the real time condition of soil of the plants via the smart phone that is connected to the internet.
* To know the status of atmospheric conditions of the farm from far away place.
* To receive alert message to the farmer if any animal is detected in the farm.

1. **Background of the Idea:**

Agriculture is the strength of Indian Economy. However, for agriculture water consumption is more than rainfall every year. Improving farm yield is essential to meet the rapidly growing demand of food for population growth across the world. By considering and predicting environmental circumstances, farm yield can be increased. Crop quality is based on data collected from field such as soil moisture, ambient temperature and humidity etc. Advanced tools and technology can be used to increase farm production. Developing IoT technologies can help to collect large amount of environmental and crop recital data.Agriculture remains the sector which contributes the highest to India’s GDP. But, when considering technology that is deployed in this field, we find that the development is not tremendous. Now a day’s there is huge enhancement in technologies which have a significant impact on various fields like agriculture, healthcare etc. Agriculture is the primary occupation in our country. India’s major income source is depending on agriculture therefore the development of agriculture is important. In today also most of the irrigation system are operated manually. The available traditional techniques are like drip irrigation, sprinkler irrigation etc. These techniques are need to be combined with IoT so that we can make use of water vary efficiently. IoT helps to access information and make major decision making process by getting different values from sensors like soil moisture, water level sensors, water quality etc. This paper focuses primarily on reducing the wastage of water and minimizing the manual labor on field for irrigation so that you can saving time, cash and power of the farmer.

“IoT encompasses many new intelligent concepts for using in the near future such as smart home, smart city, smart transportation, and smart farming”. The technique can be used for application of accurate amount of fertilizer, water, pesticide etc. to enhance productivity and excellence. Sensors are hopeful device for smart agriculture. The retime environmental parameters like soil moisture level, temperature and tank water level have continuous influence on the crop lifecycle. By forming sensor network, good monitoring of water regulation in the agriculture field can be achieved. The system was developed to monitor the environmental conditions such as temperature, soil moisture content, humidity of the air and water level of agriculture land for controlling the irrigation. The real time conditions sensed data is send to the cloud server for storing and decision making and controlling actions for future also.

1. **Detailed Problem Description:**

* An automated irrigation system is needed to optimize water use for agricultural crops.
* By forming sensor network, good monitoring of water regulation in the agriculture field can be achieved. Advanced tools and technology can be used to increase farm yield.
* It is important to monitor temperature , humidity and soil moisture and also the animal activity

of the crop field.

* An irrigation system is critical to any homeowner’s landscape because it keeps the landscape properly irrigated on a consistent schedule so that a homeowner does not have to try to remember when they last watered the backyard lawn or the rose bushes out front.
* But now irrigation has gotten even more sophisticated with smart irrigation.
* The smart irrigation system will help you have better control of your landscape and irrigation needs as well as peace of mind that the smart system can make decisions independently if you are away.
* You will save a significant amount of money on your water bills because through intelligent control and automation, your smart irrigation system will optimize resources so that everything gets what it needs without needless waste.
* Additionally, we have all seen many places in the country that have experienced droughts and we know that our water resources are precious. With smart irrigation systems we can be better stewards of our resources which is better for the environment.

1. **Proposed Innovative Solution and Methodology:**

The [DHT11 humidity and temperature sensor](https://www.adafruit.com/product/386) make it really easy to add humidity and temperature data to your DIY electronics projects. It’s perfect for remote weather stations, home environmental control systems, and farm or garden monitoring systems.

**DHT11 Sensor**

* The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analog input pins needed). Its fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds, the sensor readings can be up to 2 seconds old.

### What is Relative Humidity?

* The DHT11 measures relative humidity. The relative humidity is the amount of water vapor in air vs. the saturation point of water vapor in the air. At the saturation point, water vapor starts to condense and accumulate on surfaces forming dew.
* VCC - red wire Connect to 3.3 - 5V power. Sometime 3.3V power isn't enough in which case try 5V power.
* Data out - white or yellow wire
* Not connected
* Ground - black wire

### SOIL MOISTURE SENSOR Operation

* The Soil Moisture Sensor measures soil moisture grace to the changes in electrical conductivity of the earth ( soil resistance increases with drought ).

3V --> VCC

GND --> GND

A0 --> A0

* **Soil moisture sensors** measure the volumetric [water content I](https://en.wikipedia.org/wiki/Water_content)n [soil.[1] S](https://en.wikipedia.org/wiki/Soil)ince the direct [gravimetric measurement o](https://en.wikipedia.org/wiki/Gravimetric_analysis)f free soil moisture requires removing, drying, and weighing of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with [neutrons,](https://en.wikipedia.org/wiki/Neutron)as a proxy for the moisture content.

ULTRASONIC SENSOR

* Ultrasonic Sensor HC-SR04 is a sensor that can measure distance. It emits an ultrasound at 40 000 Hz (40kHz) which travels through the air and if there is an object or obstacle on its path It will bounce back to the module. Considering the travel time and the speed of the sound you can calculate the distance.
* The configuration pin of HC-SR04 is VCC (1), TRIG (2), ECHO (3), and GND (4). The supply voltage of VCC is +5V and you can attach TRIG and ECHO pin to any Digital I/O in your Arduino Board.

What is a Relay?

* A Relay is a simple electromechanical switch. While we use normal switches to close or open a circuit manually, a [Relay](https://www.electronicshub.org/relay-wiring-diagram/) is also a switch that connects or disconnects two circuits. But instead of a manual operation, a relay uses an electrical signal to control an electromagnet, which in turn connects or disconnects another circuit.
* Cathod is connected to GND . Anode is connected to D2.
* Relay is connected motor and it automativally switch ON and OFF according to soil moisture level.

1. **Design and Modeling (Software or Hardware) of the proposed solution:**

#include "DHT.h" // Including library for dht

#include <ESP8266WiFi.h>

String apiKey = "1GIRLVD3J64LG132"; // Enter your Write API key from ThingSpeak

const char \*ssid = "SREEPADA"; // replace with your wifi ssid and wpa2 key

const char \*pass = "aurora#444";

const char\* server = "api.thingspeak.com";

const int sensor\_pin = A0;

const int trigPin = 12;

const int echoPin = 14;

const int pump = D2;

int pump\_status = 0;

//define sound velocity in cm/uS

#define SOUND\_VELOCITY 0.034

#define CM\_TO\_INCH 0.393701

long duration;

float distanceCm;

float distanceInch;

DHT dht;

WiFiClient client;

void setup()

{

dht.setup(D0); // data pin D1

Serial.begin(9600);

delay(10);

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

pinMode(pump, OUTPUT); // acts as pump connection pin

digitalWrite(pump, LOW); // Initially make pump off

// dht.begin();

Serial.println("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, pass);

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

}

void loop()

{

float h = dht.getHumidity();

float t = dht.getTemperature();

float moisture\_percentage = ( 100.00 - ( (analogRead(sensor\_pin)/1023.00) \* 100.00 ) );

if (moisture\_percentage < 0)

{

moisture\_percentage = 0;

}

if (moisture\_percentage == 0)

{

digitalWrite(pump, HIGH);

pump\_status = 1;

}

else

{

digitalWrite(pump, LOW);

pump\_status = 0;

}

if (isnan(h) || isnan(t))

{

Serial.println("Failed to read from DHT sensor!");

return;

}

if (client.connect(server,80)) // "184.106.153.149" or api.thingspeak.com

{

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculate the distance

distanceCm = duration \* SOUND\_VELOCITY/2;

// Convert to inches

distanceInch = distanceCm \* CM\_TO\_INCH;

// Prints the distance on the Serial Monitor

Serial.print("Distance (cm): ");

Serial.println(distanceCm);

Serial.print("Distance (inch): ");

Serial.println(distanceInch);

Serial.print("Pump Status: ");

Serial.println(pump\_status);

delay(100);

String postStr = apiKey;

postStr +="&field1=";

postStr += String(t);

postStr +="&field2=";

postStr += String(h);

postStr +="&field3=";

postStr += String(moisture\_percentage);

postStr +="&field4=";

postStr += String(distanceInch);

postStr +="&field5=";

postStr += String(pump\_status);

postStr += "\r\n\r\n";

client.print("POST /update HTTP/1.1\n");

client.print("Host: api.thingspeak.com\n");

client.print("Connection: close\n");

client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");

client.print("Content-Type: application/x-www-form-urlencoded\n");

client.print("Content-Length: ");

client.print(postStr.length());

client.print("\n\n");

client.print(postStr);

Serial.print("Temperature: ");

Serial.print(t);

Serial.print(" degrees Celcius, Humidity: ");

Serial.print(h);

Serial.println("%. Send to Thingspeak.");

Serial.print("Soil Moisture(in Percentage) = ");

Serial.print(moisture\_percentage);

Serial.println("%");

delay(1000);

}

client.stop();

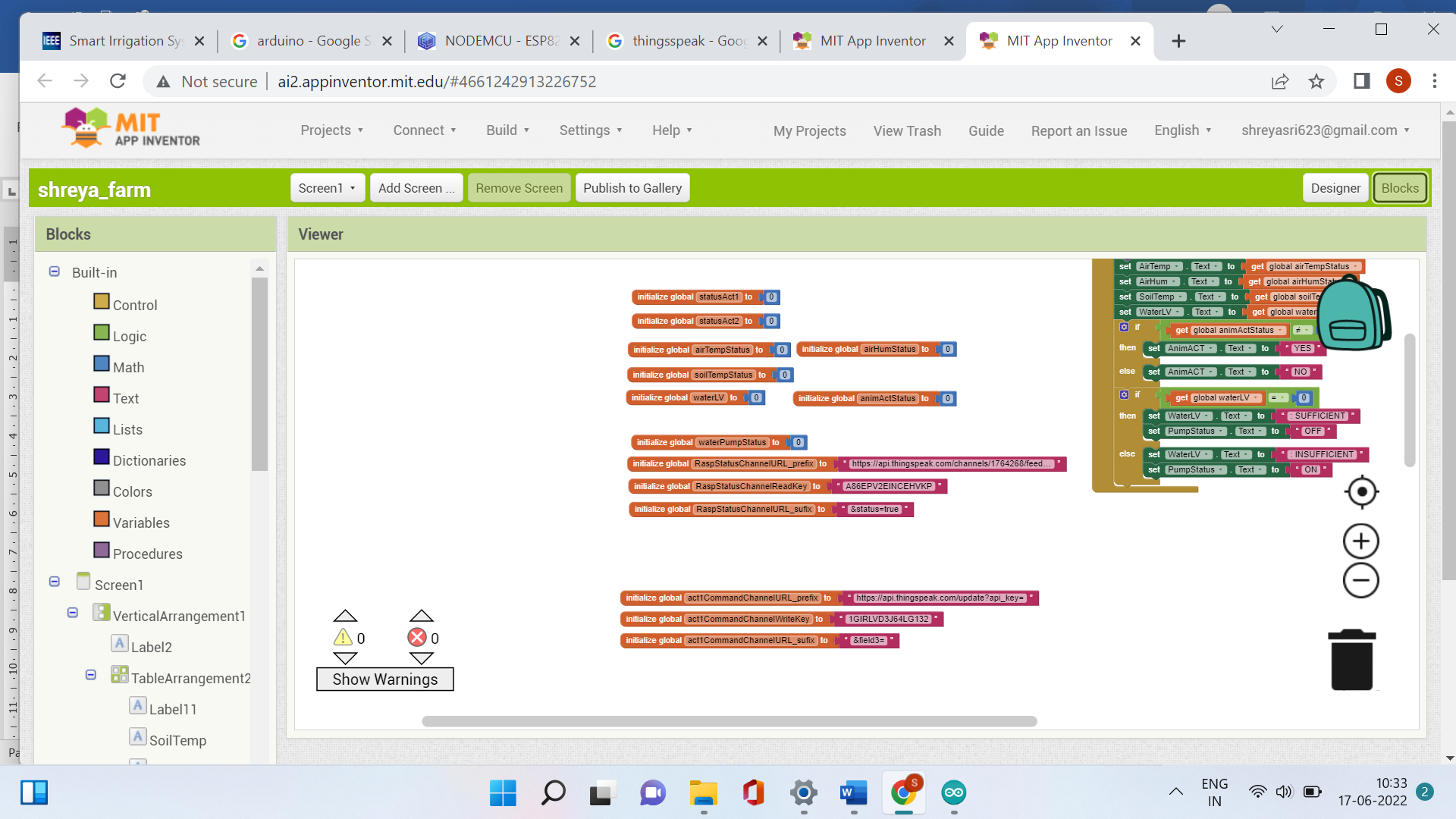
Serial.println("Waiting...");

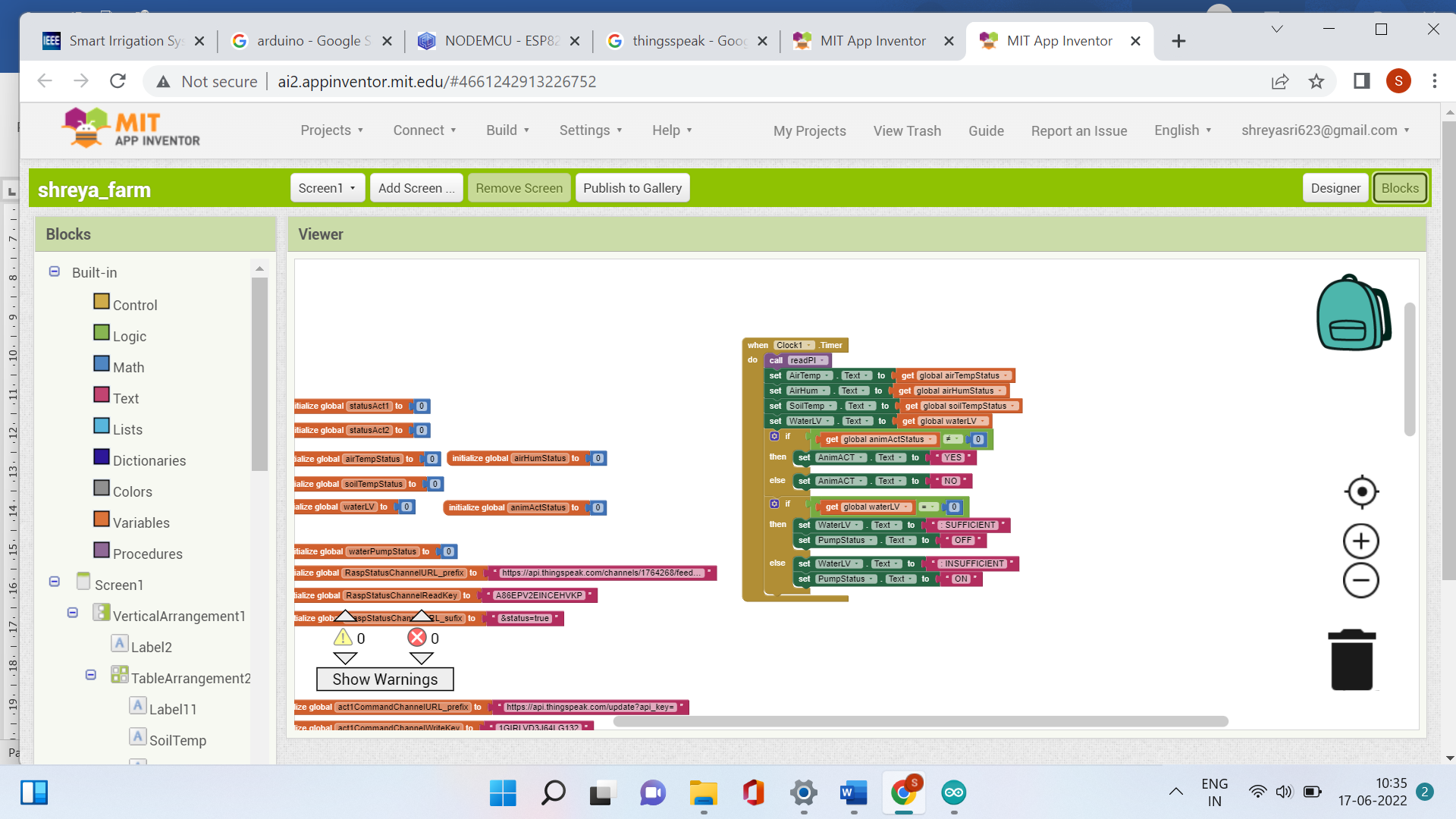
// thingspeak needs minimum 15 sec delay between updates

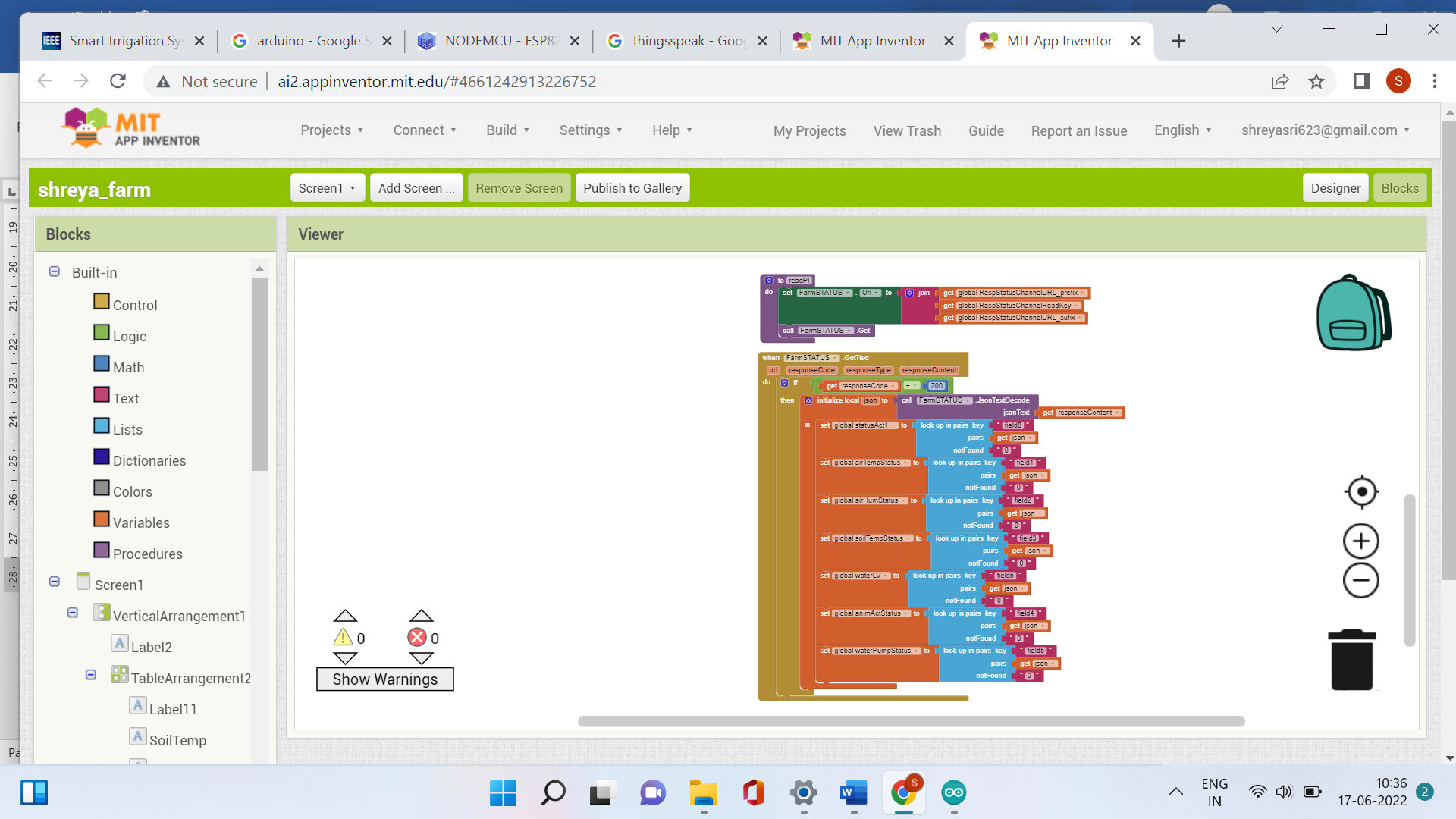
delay(1000);

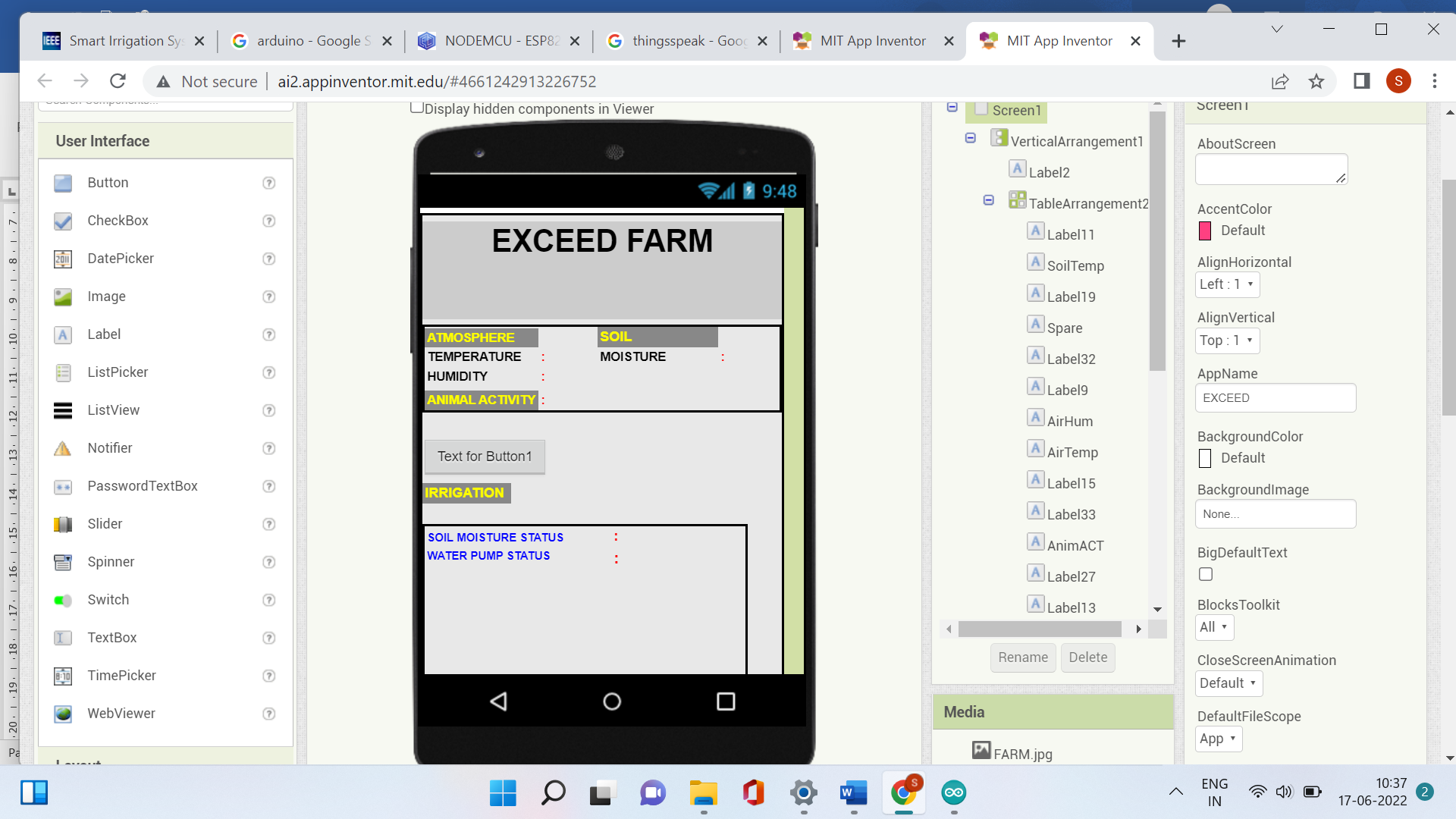
}

* This is the code used to receive the data from Arduino ESP 8266 to ThingSpeak (cloud).
* we developed the app by using MIT app inventor tool , following is the design blocks



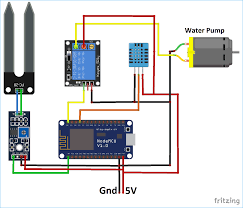






1. **Detailed description of the Prototype or Product**

(Including block diagrams, working principles, explanation of each and every components, Technology description and demonstration):



* Finally this is how the data looks on ThingSpeak Dashboard

