# Automated Irrigation system Applied in Crop Farming (ITC's Green House)


Lecturer : HEL CHANTHON

Student : PROEUNG BUNRONG (e20191346)

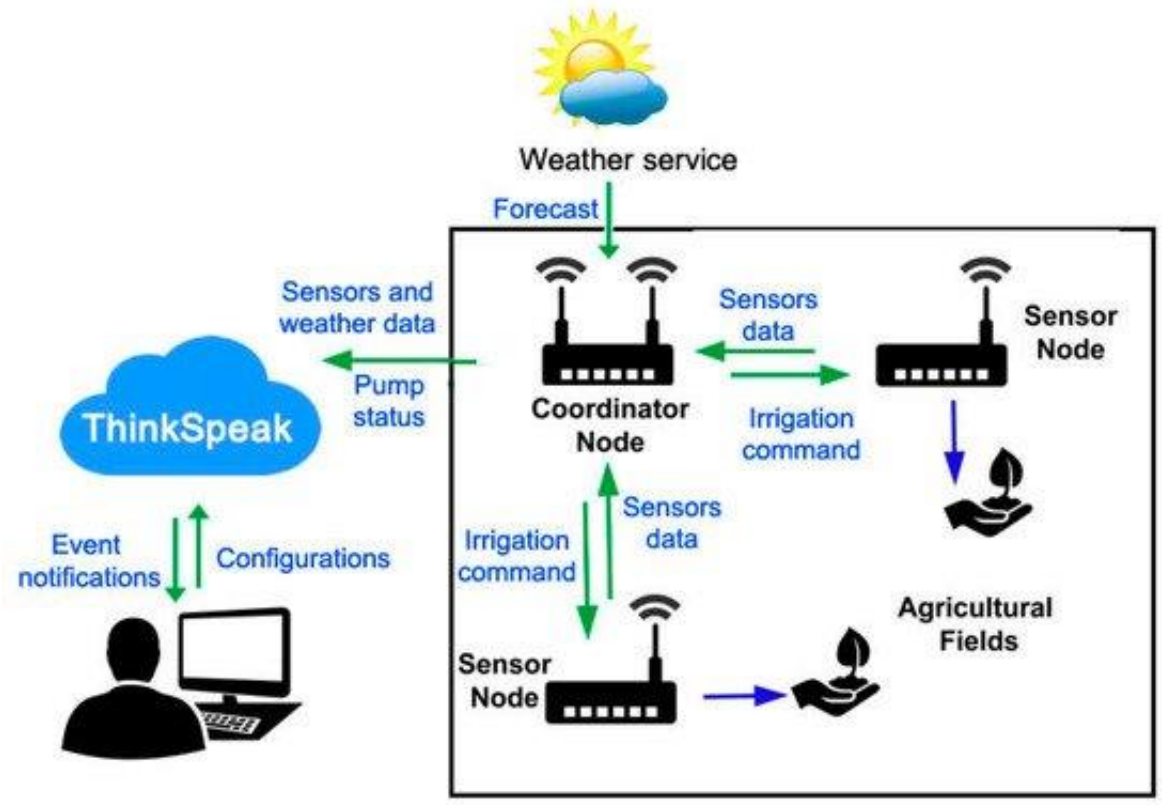
Department : I3 GTR

# Project Analysis

## ❖ The Composition of the system

The system is composed mainly of three parts:

Cloud Platform, Coordinator Node, and Sensor Nodes …



# Project Analysis

## ❖ Planning for first month

| Week 1<br>6 Aug – 13 Aug | Week2<br>14 Aug – 20 Aug | Week3<br>21 Aug – 27 Aug | Week4<br>28 Aug – 3 Sep |
|---|---|---|---|
| Researching<br>Code and test with dht11<br>Create charnel Thingspeak and testing. | To understand problem of code, Using Wi-Fi with username<br>Combine code with coordinator and testing then send data to Thingspeak. | Collect data from Greenhouse send to Thingspeak<br>And show all the result of Temperature (T1+T2)<br>Or another Sensor on thingspeak. | Need to Complete planning of first month and continue working for second month to use Camera in greenhouse. |
| Missing: testing some error, problem to understand something in code | Not ready combine code with coordinator yet. | | |

# Project Analysis

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

coordinator_with_ESP8266

```
#include <SoftwareSerial.h>
#include <LiquidCrystal_I2C.h>
#include <XBee.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#define RX 11 // set rx pin
#define TX 12 // set tx pin
String AP = "GTR_LAB";       // AP NAME
String PASS = "@gtrlab@"; // AP PASSWORD
String API = "88OOFGWCTW33Q0I6";   // Write API KEY
String HOST = "api.thingspeak.com";
String PORT = "80";
String field1 = "field1";
String field2 = "field2";
String field3 = "field3";
String field4 = "field4";
String field5 = "field5";
String field6 = "field6";
String field7 = "field7";

int temp2;
int hum2;
int rad1;
int temp1;
int hum1;
int countTrueCommand;
int countTimeCommand;
boolean found = false;
SoftwareSerial esp8266(TX,RX);

//Define XBee
SoftwareSerial XBee(0,1);

//Define package structure
/* | header | node_id | humidity | temperature | moisture | rdiation |
   |   2    |    2    |     4    |      4      |     4    |     4    | */

typedef struct
{
  uint8_t header;
  uint8_t node_id;
```

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

coordinator_with_ESP8266

```

  float humidity;
  float temperature;
  float moisture;
  float radiation;
} data_struct_t;

typedef union
{
  data_struct_t data_struct;
  uint8_t data_byte[20];
} packet_t;

//Define LCD
LiquidCrystal_I2C lcd(0x3F, 20, 4);

//Define relay pin
#define relayPin 5

//Define mois1 and mois2, mois1 and 2 should more than 50 in order to prevent relay activate in case one node is not available
float mois1;
float mois2;

void setup() {
  XBee.begin(9600);
  Serial.begin(9600);
  lcd.begin();
  lcd.backlight();

  //Set relay pin
  pinMode(relayPin, OUTPUT);

  //Set title
  lcd.setCursor(4,1);
  lcd.print("NEVER GIVE UP");
  delay(1000);
  lcd.clear();

  esp8266.begin(115200);
  sendCommand("AT",5,"OK");  //send AT command
  sendCommand("AT+CWMODE=3",5,"OK"); // set MODE 1
```

# Project Analysis

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

coordinator_with_ESP8266

```
  sendCommand("AT+CWJAP=\""+ AP +"\",\""+ PASS +"\"",20,"OK"); // configure esp8266 to WiFi

}
void display();


void loop() {
  //Check if XBee available
  packet_t coord;
  if (XBee.available() > 0)
  {
    //Set seperate display
    lcd.setCursor(9,0);
    lcd.print("|");
    lcd.setCursor(9,1);
    lcd.print("|");
    lcd.setCursor(9,2);
    lcd.print("|");
    lcd.setCursor(9,3);
    lcd.print("|");

    coord.data_byte[0] = XBee.read();
    if (coord.data_struct.header == 0x55)
    {
      Serial.print(coord.data_byte[0], HEX);
      Serial.print(";");
      for (int i = 1; i <= 20; i++)
      {
        coord.data_byte[i] = XBee.read();
        Serial.print(coord.data_byte[i], HEX);
        Serial.print(";");
      }
    }

    //Print data from Node 1 on LCD
    if (coord.data_struct.node_id == 1)
    {
      lcd.setCursor(0,0);
      lcd.print("H1:");
      lcd.setCursor(3,0);
      lcd.print(coord.data_struct.humidity ,2);
```

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

coordinator_with_ESP8266

```
      lcd.print(coord.data_struct.humidity ,2);
      lcd.setCursor(0,1);
      lcd.print("T1:");
      lcd.setCursor(3,1);
      lcd.print(coord.data_struct.temperature, 2);
      lcd.setCursor(0,2);
      lcd.print("M1:");
      lcd.setCursor(3,2);
      lcd.print(coord.data_struct.moisture, 2);
      huml = coord.data_struct.humidity;
      templ = coord.data_struct.temperature;
      moisl = coord.data_struct.moisture;
    }

    //Print data from Node 2 on LCD
    if (coord.data_struct.node_id == 2)
    {
      lcd.setCursor(11,0);
      lcd.print("H2:");
      lcd.setCursor(14,0);
      lcd.print(coord.data_struct.humidity, 2);
      lcd.setCursor(11,1);
      lcd.print("T2:");
      lcd.setCursor(14,1);
      lcd.print(coord.data_struct.temperature, 2);
      lcd.setCursor(11,2);
      lcd.print("M2:");
      lcd.setCursor(14,2);
      lcd.print(coord.data_struct.moisture, 2);
      lcd.setCursor(11,3);
      lcd.print("R1:");
      lcd.setCursor(14,3);
      lcd.print(coord.data_struct.radiation, 1);

      hum2 = coord.data_struct.humidity;
      temp2 = coord.data_struct.temperature;
      mois2 = coord.data_struct.moisture;
      radl = coord.data_struct.radiation;
    }
    //Define when to start the Valve
    if (moisl <= 32.00 or mois2 <= 32.00)
    {
```

# Project Analysis

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

coordinator_with_ESP8266

```
    {
      digitalWrite(relayPin, HIGH);
    }

    //Define when to stop the Valve
    if (mois1 >= 34.00 and mois2 >= 34.00)
    {
      digitalWrite(relayPin, LOW);
    }
    /*
    Serial.println("Mois1 is: ");
    Serial.print(mois1);
    Serial.println("Mois2 is: ");
    Serial.print(mois2);
    */
  }
      String getData = "GET /update?api_key="+ API +"&field1="+String(temp1)+"&field2="+String(hum1)+"&field3="+String(mois1)+"&field4="+String(temp2)+"&field5="+String(hum2)+"&field6="+String(mois2)+"&field7="+String(rad1);
      sendCommand("AT+CIPMUX=1",5,"OK"); // set to single connection
      sendCommand("AT+CIPSTART=0,\"TCP\",\""+ HOST +"\","+ PORT,15,"OK"); // to connect to the Thingspeak API using TCP protocol
      sendCommand("AT+CIPSEND=0," +String(getData.length()+4),4,">");// for read data and start sending data
      esp8266.println(getData); // send data to Thingspeak
      //delay(1000);
      countTrueCommand++;
      sendCommand("AT+CIPCLOSE=0",5,"OK");// for end and close transmission
      //delay(10000
}

void sendCommand(String command, int maxTime, char readReplay[]) {
  Serial.print(countTrueCommand);
  Serial.print(". at command => ");
  Serial.print(command);
  Serial.print(" ");
  while(countTimeCommand < (maxTime*1))
  {
    esp8266.println(command);//at+cipsend
    if(esp8266.find(readReplay))//ok
    {
      found = true;
      break;
    }

    countTimeCommand++;
```

Done compiling.

# Project Analysis

coordinator_with_ESP8266 | Arduino 1.8.13 Hourly Build 2020/06/03 04:33

File Edit Sketch Tools Help

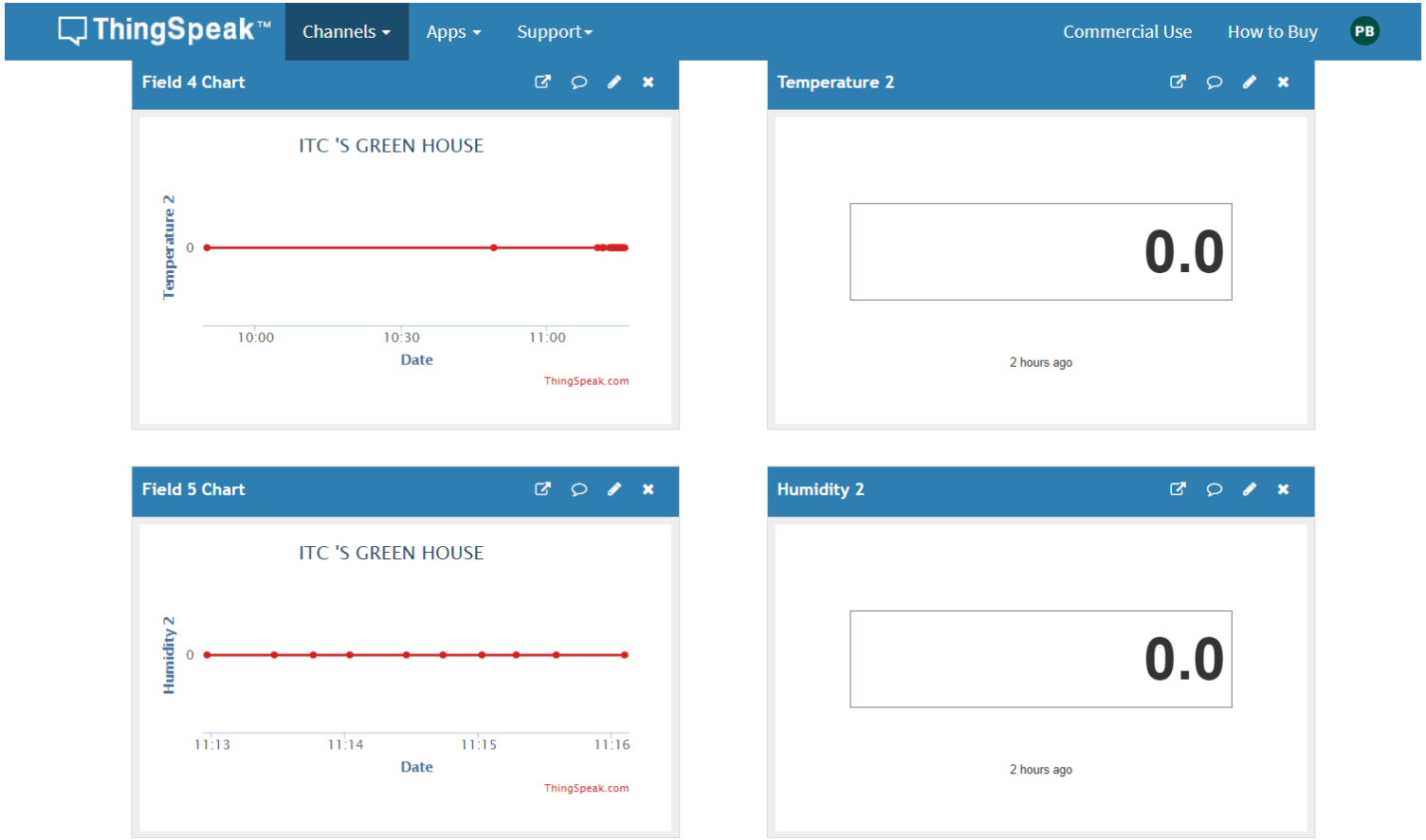
coordinator_with_ESP8266

```
180       String getData = "GET /update?api_key="+ API +"&field1="+String(temp1)+"&field2="+String(humi)+"&field3="+String(mois1)+"&field4="+Stri
181       sendCommand("AT+CIPMUX=1",5,"OK"); // set to single connection
182       sendCommand("AT+CIPSTART=0,\"TCP\",\""+ HOST +"\","+ PORT,15,"OK"); // to connect to the Thingspeak API using TCP protocol
183       sendCommand("AT+CIPSEND=0," +String(getData.length()+4),4,">");// for read data and start sending data
184       esp8266.println(getData); // send data to Thingspeak
185       //delay(1000);
186       countTrueCommand++;
187       sendCommand("AT+CIPCLOSE=0",5,"OK");// for end and close transmission
188       //delay(10000
189 }
190
191 void sendCommand(String command, int maxTime, char readReplay[]) {
192   Serial.print(countTrueCommand);
193   Serial.print(". at command => ");
194   Serial.print(command);
195   Serial.print(" ");
196   while(countTimeCommand < (maxTime*1))
197   {
198     esp8266.println(command);//at+cipsend
199     if(esp8266.find(readReplay))//ok
200     {
201       found = true;
202       break;
203     }
204
205     countTimeCommand++;
206   }
207
208   if(found == true)
209   {
210     Serial.println("Yes");
211     countTrueCommand++;
212     countTimeCommand = 0;
213   }
214   if(found == false)
215   {
216     Serial.println("Failed");
217     countTrueCommand = 0;
218     countTimeCommand = 0;
219   }
220   found = false;
221 }
```

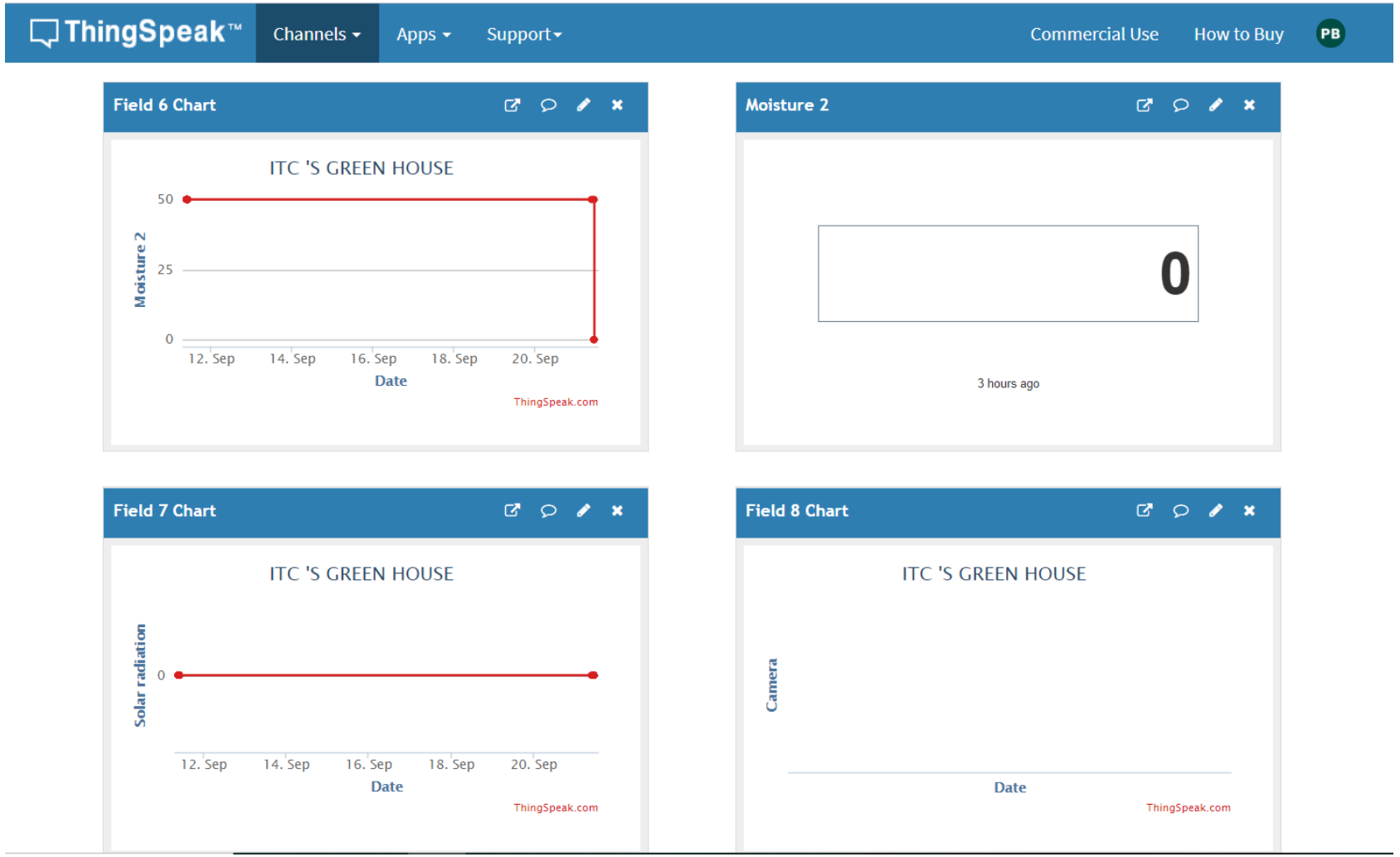
# Project Analysis

## ❖ Problems

Combine code not ready correct yet because send data to thingspeak value of dht not correct and lcd not print.



# Project Analysis



# Thank You