

Sprint Code

Date	10 Nov 2022
Team ID	PNT2022TMID47541
Project name	IoT Based Smart Crop Protection System for Agriculture

1. Python code for Monitoring field

```
while True:
    motorstatus = ""
    waterlevel=random.randint(1,20)
    Waterusing=""
    if waterlevel < 2:
        waterusing =str(waterlevel) + " feet Water Level Low"
    else:
        waterusing =str(waterlevel) + " feet Using rain water"

    soilmoisture=random.randint(0,872)
    if soilmoisture <200:
        motorstatus="Motor on automatically"
    else:
        motorstatus="Motor off automatically"

    response = requests.get(complete_url)
    x = response.json()
    if x["cod"] != "404":
        y=x["name"]
        z = x["weather"]
        weather_description = z[0]["description"]
        data = { 'Waterlevel': waterusing, 'SoilMoisture': soilmoisture, 'MotorStatus':motorstatus, 'Area':str(y), 'WeatherDescription':str(weather_descripti
        #print data
        def myOnPublishCallback():
            print ("Published Waterlevel = %s " % waterusing, "SoilMoisture = %s" % soilmoisture, "MotorStatus = %s" % motorstatus, " Area="+str(y), " Weath
        success = deviceCli.publishEvent("event_1", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoT")
            time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
|
```

2. Wokwi Code for Monitoring field

wokwi.com/projects/349111410774508114

Download Turbo C... php ASP.NET - Data Sou... ASP.NET - Ajax Con... Cookies in ASP.NET pgm AjaxControlToolKit... when i extract a zip...

WOKWI

esp32 409341 hello.ino diagram.json libraries.txt Library Manager

```
1 #include "SPI.h"
2 #include "Adafruit_GFX.h"
3 #include "Adafruit_IL19341.h"
4 #include "DHT.h"
5 #include "WiFi.h" //library for wifi
6 #include "PubSubClient.h" //library for MQTT
7 #include "HTTPClient.h"
8 #include "Arduino_JSON.h"
9
10 #define PIN_TRIGGER 27
11 #define PIN_ECHO 26
12 #define TTF_DC 2
13 #define TTF_CS 15
14 #define SOUND_SPEED 0.034
15
16 const int DHT_PIN = 13;
17 int pirpin = 12;
18 int BUZZER_CHANNEL = 0;
19 int ultrabuzzer=37;
20 int temperaturepin=35;
21 const float BETA = 3950;
22 int pirState = LOW;
23 String payload;
24
25
26
27
28 const String endpoint = "https://api.openweathermap.org/data/2.5/weather?lat=18.2779&lon=78.5424&appid=";
29 const String key = "d6f65eca7cd6e509a901090c1815a5";
30
31 DHTesp dhtSensor;
32 Adafruit_IL19341 tft = Adafruit_IL19341(TTF_CS, TTF_DC);
33
34 void callback(char* topic, byte* payload, unsigned int payloadlength);
35
36 #define ORG "o18g4e" //IBM ORGANITION ID
37 #define DEVICE_TYPE "temp" //Device type mentioned in the Watson IoT Platform
38 #define DEVICE_ID "1234" //Device ID mentioned in the Watson IoT Platform
39 #define TOKEN "12345678" //Token
40 String data;
```

Simulation

Reconnecting client to o18g4e.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK

Sending payload: {"temperature":47.60,"Humid":68.00,"PIR_Sensor":"No Motion Detected","Ultrasonic_Sensor":"No Motion Detected","Rain_Sensor":"Normal Rain"}
Publish ok

WOKWI

SAVE SHARE

esp32-ili9341-hello.ino

diagram.json libraries.txt Library Manager

```

43 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
44 char publishTopic[] = "iot-2/evt/event_1/fmt/json";// topic name and type of event perform and format
45 char subscribeTopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT command type AND COMMAND IS
46 char authMethod[] = "use-token-auth";// authentication method
47 char token[] = TOKEN;
48 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
49
50 WiFiClient wifiClient; // creating the instance for wifiClient
51 PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing
52
53 void setup() {
54   Serial.begin(9600);
55   dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
56   tft.begin();
57   ledSetup(BUZZER_CHANNEL, 8000, 12);
58   pinMode(PIN_TRIG, OUTPUT);
59   pinMode(PIN_ECHO, INPUT);
60   ledAttachPin(ultrabuzzer, BUZZER_CHANNEL);
61   analogReadResolution(10);
62   pinMode(temperaturepin, INPUT);
63   wifiConnect();
64   mqttConnect();
65 }
66
67 int value=0;
68 void loop() {
69   HTTPClient http;
70   http.begin(endpoint + key);
71   int httpCode = http.GET();
72   if (httpCode > 0) {
73     String payloadweather = http.getString();
74     //Serial.println(payloadweather);

```

Simulation

```

{"temp":297.04,"feels_like":297.42,"temp_min":297.04,"temp_max":297.04,"pressure":1013.25,"humidity":74,"sea_level":1011,"grnd_level":994},"visibility":10000,"wind":{"speed":1.81,"deg":185,"gust":2.28},"clouds":{"all":96},"dt":1669221916,"sys":{"country":"IN","sunrise":1669164245,"sunset":1669206033},"timezone":19800,"id":1234567890,"e":"Ponnamarāvati","cod":200}
Area: "Ponnamarāvati"
Weather_Details: [{"id":804,"main":"Clouds","description":"overcast clouds","icon":

```

WOKWI

SAVE SHARE

esp32-ili9341-hello.ino

diagram.json libraries.txt Library Manager

```

72 if (httpCode > 0) {
73   String payloadweather = http.getString();
74   Serial.println(httpCode);
75   Serial.println("Http Status Code: "+httpCode);
76   Serial.println(payloadweather);
77   JSONVar myJsonObject = JSON.parse(payloadweather);
78
79   if (JSON.typeof(myJsonObject) == "undefined") {
80     Serial.println("Parsing input failed!");
81     return;
82   }
83   Serial.print("Area: ");
84   Serial.println(myJsonObject["name"]);
85   Serial.print("Weather_Details: ");
86   Serial.println(myJsonObject["weather"]);
87 }
88
89 else {
90   Serial.println("Error on HTTP request");
91 }
92
93 http.end(); //Free the resources
94
95 TempAndHumidity data = dhtSensor.getTempAndHumidity();
96 float t=data.temperature;
97 float h=data.humidity;
98 payload = "{"+"temperature:"+t;
99 payload += ","+"Humidity:"+h;
100 payload += ","+"PIR_Sensor\":\"No Motion Detected\"";
101 payload += ","+"Ultrasonic_Sensor\":\"No Motion Detected\"";
102 payload += ","+"Rain_Sensor\":\"Normal Rain\"";
103 payload += ",";

```

Simulation

```

Reconnecting client to o18g4e.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK

Sending payload: {"temperature":47.60,"Humid":68.00,"PIR_Sensor":"No Motion Detected","Ultrasonic_Sensor":"No Motion Detected","Rain_Sensor":"Normal Rain"}
Publish ok

```

WOKWI

SAVE SHARE

esp32-ili9341-hello.ino

diagram.json libraries.txt Library Manager

```

103
104 if (!client.loop()) {
105   mqttConnect();
106 }
107
108
109 int state = digitalRead(pirpin);
110
111 if (state == HIGH) {
112   if (pirState == LOW)
113   {
114     tft.fillScreen(ILI9341_BLACK);
115     tft.setTextColor(ILI9341_RED);
116     tft.setTextSize(3);
117     tft.setCursor(0, 120);
118     tft.println("Motion Detected");
119     pirState = HIGH;
120     ledWriteTone(BUZZER_CHANNEL, 500);
121     payload += "{"+"PIR_Sensor\":\"";
122     payload += "\"Motion Detected\"";
123     payload += ",";
124   }
125 }
126
127 else if (pirState == HIGH) {
128   tft.fillScreen(ILI9341_BLACK);
129   tft.setTextColor(ILI9341_RED);
130   tft.setTextSize(3);
131   tft.setCursor(0, 120);
132   tft.println("Motion ended");
133   ledWrite(BUZZER_CHANNEL, 0);

```

Simulation

```

Reconnecting client to o18g4e.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK

Sending payload: {"temperature":47.60,"Humid":68.00,"PIR_Sensor":"No Motion Detected","Ultrasonic_Sensor":"No Motion Detected","Rain_Sensor":"Normal Rain"}
Publish ok

```

WOKWI SAVE SHARE Docs

esp32-40341-hello.ino diagram.json libraries.txt Library Manager

```

134 ledwrite(BUZZER_CHANNEL, 0);
135 pirState = LOW;
136 payload += "\\"PIR_Sensor\\"";
137 payload += "\\"Motion Ended\\"";
138 payload += ",";
139
140 }
141 else
142 {
143   payload += "\\"PIR_Sensor\\"";
144   payload += "\\"No Motion Detected\\"";
145   payload += ",";
146 }
147
148 digitalWrite(PIN_TRIG, HIGH);
149 delayMicroseconds(10);
150 digitalWrite(PIN_TRIG, LOW);
151 int duration = pulseIn(PIN_ECHO, HIGH);
152 int distance = duration * SOUND_SPEED/2;
153 delay(1000);
154 if(distance<100)
155 {
156   tft.fillScreen(IL19341_BLACK);
157   tft.setTextColor(IL19341_RED);
158   tft.setTextSize(3);
159   tft.setCursor(1, 120);
160   tft.println("Motion Detected in 100 cm range");
161   ledwrite(BUZZER_CHANNEL, 100);
162   //ledwriteNote(BUZZER_CHANNEL, (note_t)NOTE_G, 5);
163   payload += "\\"Ultrasonic_Sensor\\"";
164   payload += "\\"Motion Detected in 100cm range\\"";
165   payload += ",";
166 }
167 else
168 {
169   ledwrite(BUZZER_CHANNEL, 0);
170

```

Simulation

```

{"temp":297.04,"feels_like":297.42,"temp_min":297.04,"temp_max":297.04,"pressure":1011,"humidity"
ea_level":1011,"grnd_level":994,"visibility":10000,"wind":
{"speed":1.81,"deg":185,"gust":2.28},"clouds":{"all":96},"dt":1669221916,"sys":
{"country":"IN","sunrise":1669164245,"sunset":1669206033},"timezone":19800,"id":1259413,"name":"P
rivate","cod":200}
Area: "Ponnamaravati"
Weather_Details: [{"id":804,"main":"Clouds","description":"overcast clouds","icon":"04n"}]

```

WOKWI SAVE SHARE Docs

esp32-40341-hello.ino diagram.json libraries.txt Library Manager

```

250 {
251   Serial.println();
252   Serial.print("Connecting to ");
253   WiFi.begin("Wokwi-ESP32", ""); //passing the wifi credentials to establish the connection
254   while (WiFi.status() != WL_CONNECTED) {
255     delay(500);
256     Serial.print(".");
257   }
258   Serial.println("");
259   Serial.println("WiFi connected");
260   Serial.println("IP address: ");
261   Serial.println(WiFi.localIP());
262 }
263 void intramangedevice() {
264   if (client.subscribe(subscribetopic)) {
265     Serial.println(subscribetopic);
266     Serial.println("subscribe to cmd OK");
267   } else {
268     Serial.println("subscribe to cmd FAILED");
269   }
270 }
271 void callback(char* subscribetopic, byte* payload, unsigned int payloadlength)
272 {
273   Serial.print("callback invoked for topic: ");
274   Serial.println(subscribetopic);
275   for (int i = 0; i < payloadlength; i++) {
276     detail += (char)payload[i];
277     Serial.println(detail);
278     if(detail=="tempor")
279     {
280       Serial.println(detail);
281       tft.fillScreen(IL19341_BLACK);
282       tft.setTextColor(IL19341_RED);
283       tft.setTextSize(3);
284       tft.setCursor(1, 120);
285       tft.println("temp on by farmer");
286     }
287   }
288   else
289   {
290     Serial.println(detail);
291     tft.fillScreen(IL19341_BLACK);
292     tft.setTextColor(IL19341_RED);
293     tft.setTextSize(3);
294     tft.setCursor(1, 120);
295     tft.println("temp off by farmer");
296   }
297   detail="";
298 }

```

Simulation

```

Reconnecting client to o8g4e.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/ret/String
subscribe to cmd OK

Sending payload: {"temperature":47.60,"Humid":68.00,"PIR_Sensor":"No Motion Detected","Ultrasonic_Sensor":"No Motion
Detected","Rain_Sensor":"Normal Rain"}
Publish ok

```