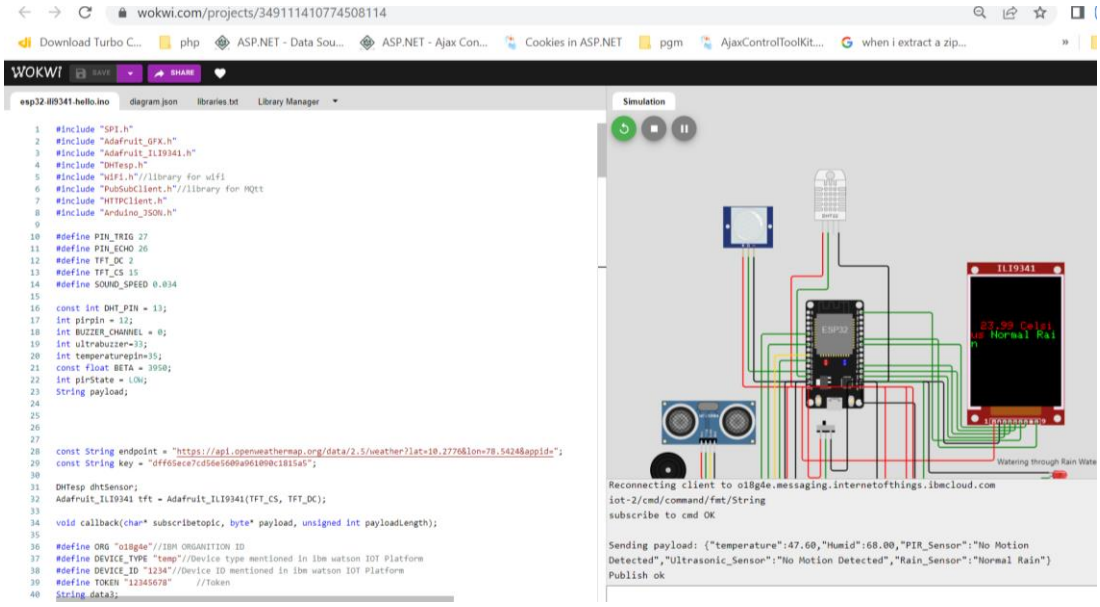
Sprint Code

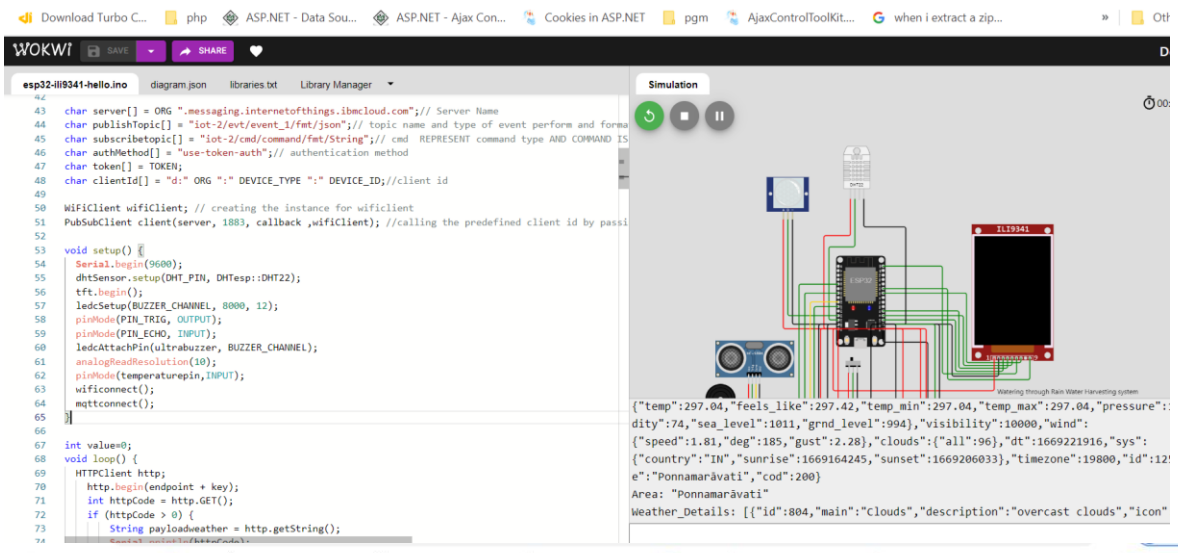
Date	10 Nov 2022
Team ID	PNT2022TMID47541
Project name	IoT Based Smart Crop Protection System for Agriculture

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 IoTF")
        time.sleep(10)


        deviceCli.commandCallback = myCommandCallback

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

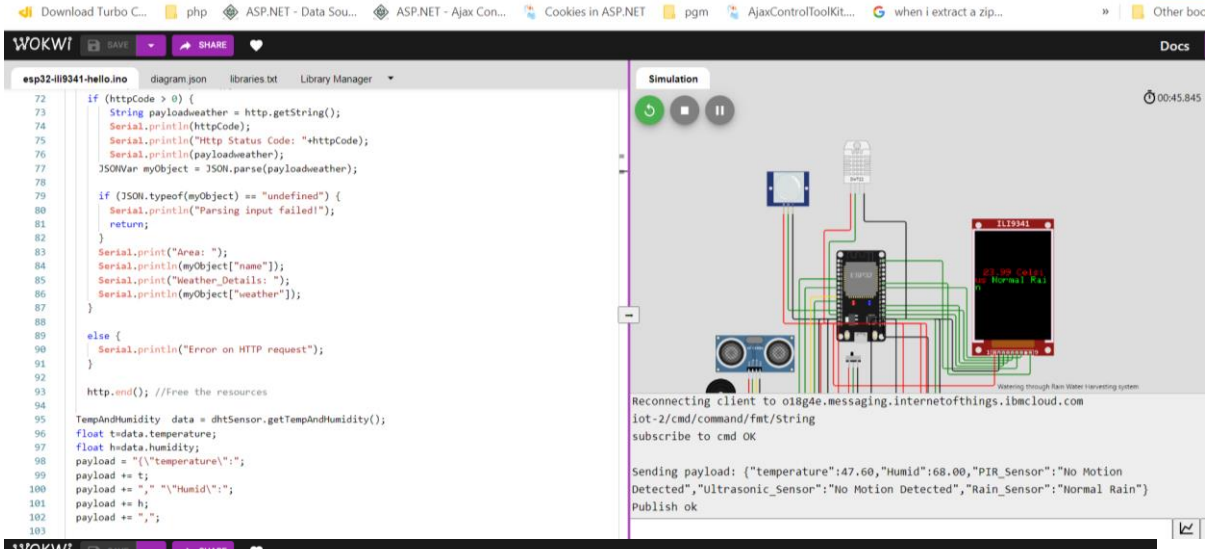
2.Wokwi Code for Monitoring field





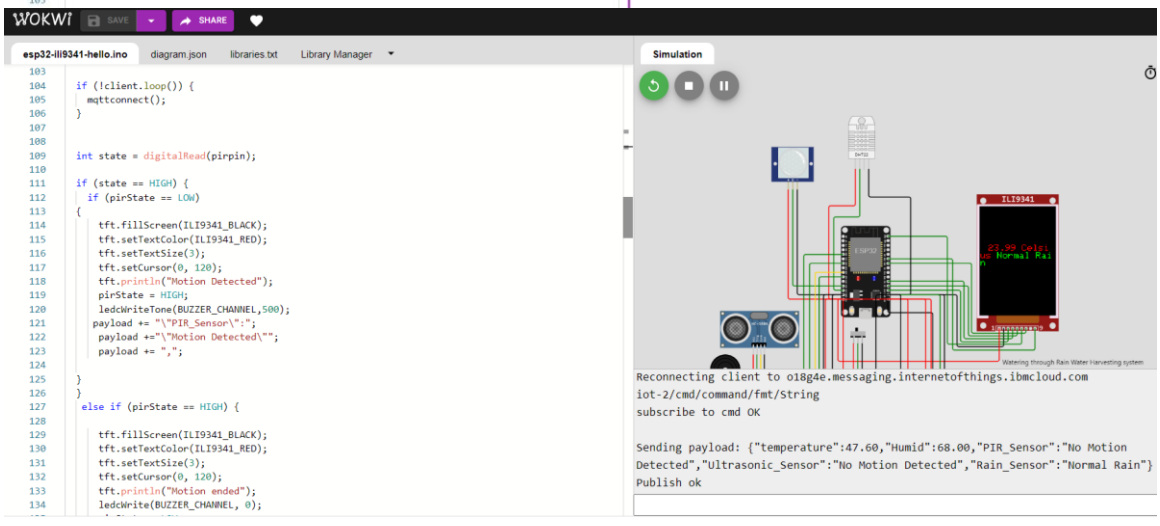
```
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/event_1/fmt/json";// topic name and type of event perform and forma
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND IS
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id

WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passi

void setup() {
  Serial.begin(9600);
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  tft.begin();
  ledcSetup(BUZZER_CHANNEL, 8000, 12);
  pinMode(PIN_TRIG, OUTPUT);
  pinMode(PIN_ECHO, INPUT);
  ledcAttachPin(ultrabuzzer, BUZZER_CHANNEL);
  analogReadResolution(10);
  pinMode(temperaturepin,INPUT);
  wificonnect();
  mqttconnect();
}

int value=0;
void loop() {
  HTTPClient http;
    http.begin(endpoint + key);
    int httpCode = http.GET();
    if (httpCode > 0) {
        String payloadweather = http.getString();
```

```
{"temp":297.04,"feels_like":297.42,"temp_min":297.04,"temp_max":297.04,"pressure":
dity":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":12
e":"Ponnamarāvati","cod":200}
Area: "Ponnamarāvati"
Weather_Details: [{"id":804,"main":"Clouds","description":"overcast clouds","icon"
```



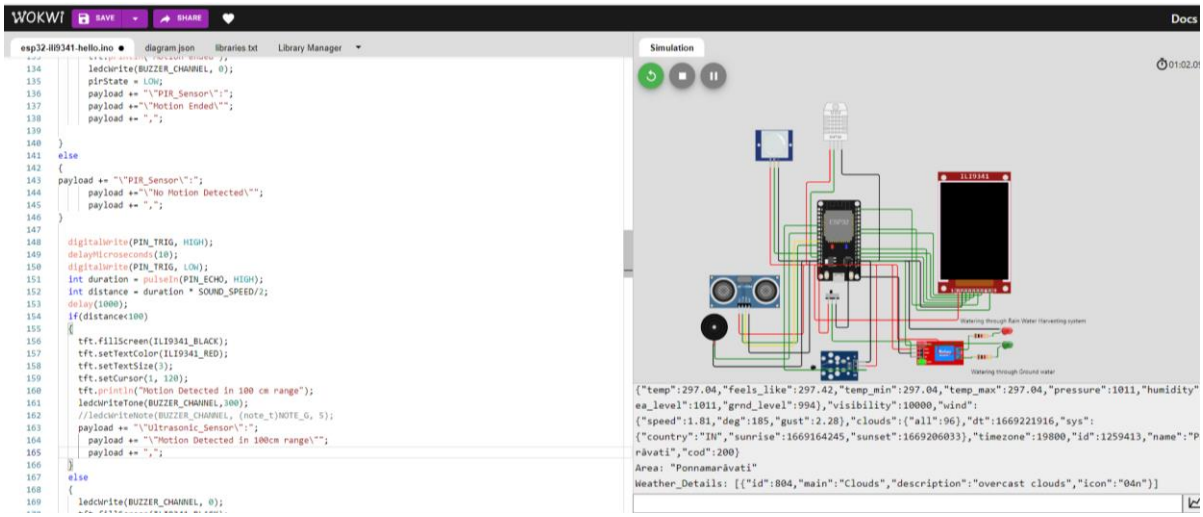
```
    if (httpCode > 0) {
        String payloadweather = http.getString();
        Serial.println(httpCode);
        Serial.println("Http Status Code: "+httpCode);
        Serial.println(payloadweather);
      JSONVar myObject = JSON.parse(payloadweather);

      if (JSON.typeof(myObject) == "undefined") {
        Serial.println("Parsing input failed!");
        return;
      }
      Serial.print("Area: ");
      Serial.println(myObject["name"]);
      Serial.print("Weather_Details: ");
      Serial.println(myObject["weather"]);
    }

    else {
      Serial.println("Error on HTTP request");
    }

    http.end(); //Free the resources

TempAndHumidity  data = dhtSensor.getTempAndHumidity();
float t=data.temperature;
float h=data.humidity;
payload = "{\"temperature\":";
payload += t;
payload += "," "\"Humid\":";
payload += h;
payload += ",";
```

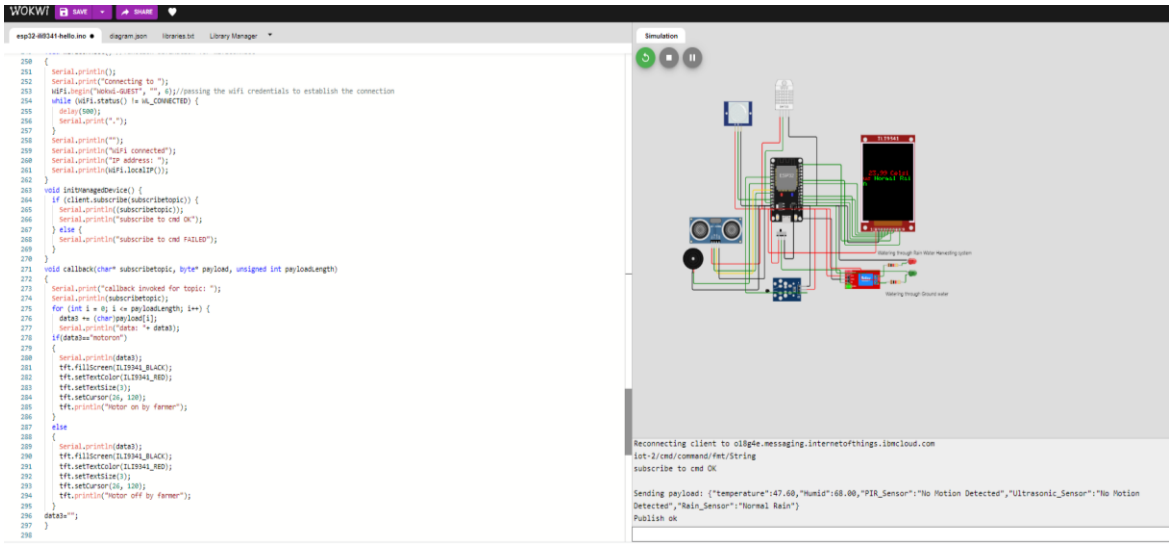
```
Reconnecting client to o18g4e.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK

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



```
  if (!client.loop()) {
    mqttconnect();
  }


  int state = digitalRead(pirpin);

  if (state == HIGH) {
    if (pirState == LOW)
  {
      tft.fillScreen(ILI9341_BLACK);
      tft.setTextColor(ILI9341_RED);
      tft.setTextSize(3);
      tft.setCursor(0, 120);
      tft.println("Motion Detected");
      pirState = HIGH;
      ledcWriteTone(BUZZER_CHANNEL,500);
     payload += "\"PIR_Sensor\":";
      payload +="\"Motion Detected\"";
      payload += ",";

  }
  }
  else if (pirState == HIGH) {

      tft.fillScreen(ILI9341_BLACK);
      tft.setTextColor(ILI9341_RED);
      tft.setTextSize(3);
      tft.setCursor(0, 120);
      tft.println("Motion ended");
      ledcWrite(BUZZER_CHANNEL, 0);
```

```
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-ili9341-hello.ino
diagram.json
libraries.txt
Library Manager
Simulation
ledcWrite(BUZZER_CHANNEL, 0);
pirState = LOW;
payload += "\"PIR_Sensor\":";
payload +="\"Motion Ended\"";
payload += ",";
}
else
{
payload += "\"PIR_Sensor\":";
payload +="\"No Motion Detected\"";
payload += ",";
}
digitalWrite(PIN_TRIG, HIGH);
delayMicroseconds(10);
digitalWrite(PIN_TRIG, LOW);
int duration = pulseIn(PIN_ECHO, HIGH);
int distance = duration * SOUND_SPEED/2;
delay(1000);
if(distance<100)
{
tft.fillScreen(ILI9341_BLACK);
tft.setTextColor(ILI9341_RED);
tft.setTextSize(3);
tft.setCursor(1, 120);
tft.println("Motion Detected in 100 cm range");
ledcWriteTone(BUZZER_CHANNEL,300);
//ledcWriteNote(BUZZER_CHANNEL, (note_t)NOTE_G, 5);
payload += "\"Ultrasonic_Sensor\":";
payload += "\"Motion Detected in 100cm range\"";
payload += ",";
}
else
{
ledcWrite(BUZZER_CHANNEL, 0);
{"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
rāvati","cod":200}
Area: "Ponnamarāvati"
Weather_Details: [{"id":804,"main":"Clouds","description":"overcast clouds","icon":"04n"}]


WOKWI
SAVE
SHARE
esp32-ili9341-hello.ino
diagram.json
libraries.txt
Library Manager
Simulation
{
Serial.println();
Serial.print("Connecting to ");
WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}
void initManagedDevice() {
if (client.subscribe(subscribetopic)) {
Serial.println((subscribetopic));
Serial.println("subscribe to cmd OK");
} else {
Serial.println("subscribe to cmd FAILED");
}
}
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
Serial.print("callback invoked for topic: ");
Serial.println(subscribetopic);
for (int i = 0; i <= payloadLength; i++) {
data3 += (char)payload[i];
Serial.println("data: "+ data3);
if(data3=="motoron")
{
Serial.println(data3);
tft.fillScreen(ILI9341_BLACK);
tft.setTextColor(ILI9341_RED);
tft.setTextSize(3);
tft.setCursor(26, 120);
tft.println("Motor on by farmer");
}
else
{
Serial.println(data3);
tft.fillScreen(ILI9341_BLACK);
tft.setTextColor(ILI9341_RED);
tft.setTextSize(3);
tft.setCursor(26, 120);
tft.println("Motor off by farmer");
}
data3="";
}
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