```
/*****************************************
 * Include Libraries
*****************************************/
#include "UbidotsESPMQTT.h"

#include <PubSubClient.h>



#include <ESP8266WiFi.h>
/*****************************************
 * Define Constants
*****************************************/
#define TOKEN
"BBFF-t7UzUescnmOGq0ol5mZCREVUV2oonh"    //
Ubidots TOKEN
#define WIFINAME
"smartfarm"                          // SSID
#define WIFIPASS
"smartfarm"                          // Wifi
Pass

const int G =14;          //RGB pins
const int R =12;
const int B =13;

const int in1=5;          //motor driver pins
const int in2=4;
const int in3=0;
const int in4=2;

char inputr[10];          //char array to
store LED inputs
char inputg[10];
char inputb[10];

Ubidots client(TOKEN);    //create client
object

String hum = "";          //initialize
sensors variable
String temp = "";
String water = "";
String air = "";
String light = "";


void callback(char* topic, byte* payload,
unsigned int length) {        //call back
function when message received
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i=0;i<length;i++) {
    Serial.print((char)payload[i]);
  }

  if (strcmp(topic,"/v1.
6/devices/esp8266/fanio/lv")==0){
//if fan turned on/off

  if ((char)payload[0]=='1'){
  Serial.print("Fan On");
  digitalWrite(in3,HIGH);
  digitalWrite(in4,LOW);

  }
  else{
  Serial.print("Fan Off");
  digitalWrite(in3,LOW);
  digitalWrite(in4,LOW);

  }
  }

  if (strcmp(topic,"/v1.
6/devices/esp8266/waterio/lv")==0){
//if water pump turned on/off

  if ((char)payload[0]=='1'){
  Serial.print("Water On");
  digitalWrite(in1,HIGH);
  digitalWrite(in2,LOW);

  }
  else{
  Serial.print("Water Off");
  digitalWrite(in1,LOW);
  digitalWrite(in2,LOW);

  }
  }

  if (strcmp(topic,"/v1.
6/devices/esp8266/lightior/lv")==0){
//if red LED turned on/off

  for (int i=0;i<length;i++) {
   inputr[i]=(char)payload[i];
  }
  int r =(int)(256-atoi(inputr));
  analogWrite(R,r*4);

  }



  if (strcmp(topic,"/v1.
6/devices/esp8266/lightiog/lv")==0){
//if green LED turned on/off
```

```
  for (int i=0;i<length;i++) {
   inputg[i]=(char)payload[i];
  }
  int g =(int)(256-atoi(inputg));
  analogWrite(G,g*4);

  }

  if (strcmp(topic,"/v1.
6/devices/esp8266/lightiob/lv")==0){
//if blue LED turned on/off

  for (int i=0;i<length;i++) {
   inputb[i]=(char)payload[i];
  }
  int b = (int)(256 -atoi(inputb));
  analogWrite(B,b*4);

  }



  Serial.println();
}
String getValue(String data, char separator,
int index)             //function to split
received serial data
```

```
{
    int found = 0;
    int strIndex[] = { 0, -1 };
    int maxIndex = data.length() - 1;

    for (int i = 0; i <= maxIndex && found
<= index; i++) {
        if (data.charAt(i) == separator || i
== maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ?
i+1 : i;
        }
    }
    return found > index ? data.
substring(strIndex[0], strIndex[1]) : "";
}

void setup() {
  // put your setup code here, to run once:
  Serial.begin(74880);
    // put your setup code here, to run once:
  pinMode(R, OUTPUT);           //initialize
all output pins
  pinMode(G, OUTPUT);
  pinMode(B, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4,OUTPUT);

  analogWrite(R,1024);          //
initialize all led to be turn off
  analogWrite(G,1024);
  analogWrite(B,1024);


  digitalWrite(in1,LOW);
  digitalWrite(in2,LOW);
  digitalWrite(in3,LOW);
  digitalWrite(in4,LOW);


client.ubidotsSetBroker("business.api.ubidots
.com");    //ubidots mqtt broker
  client.setDebug(true);       // Pass a
true or false bool value to activate debug
messages
 client.wifiConnection(WIFINAME,
WIFIPASS);  //connect to AP
  client.
begin(callback);                     //begin
```

```
listen to related topics


 client.ubidotsSubscribe("esp8266",
"fanio");       //subscribe to related topics
 client.ubidotsSubscribe("esp8266",
"waterio");
 client.ubidotsSubscribe("esp8266",
"lightior");
 client.ubidotsSubscribe("esp8266",
"lightiog");
 client.ubidotsSubscribe("esp8266",
"lightiob");

}

void loop() {

if(!client.connected()){          //if
client disconnect, reconnect
  client.reconnect();
 client.ubidotsSubscribe("esp8266","fanio");
 client.ubidotsSubscribe("esp8266",
"waterio");
 client.ubidotsSubscribe("esp8266",
"lightior");
 client.ubidotsSubscribe("esp8266",
"lightiog");
 client.ubidotsSubscribe("esp8266",
"lightiob");

      }
if (Serial.available()) {            //if
serial message received

    String messageTemp = Serial.
readString();       //read string
    hum = getValue(messageTemp, ',',
0);            //split and store related
sensors data
    temp = getValue(messageTemp, ',', 1);
    water = getValue(messageTemp, ',', 2);
    air = getValue(messageTemp, ',', 3);
    light = getValue(messageTemp, ',', 4);



  }
    client.add("hum", hum.
toFloat());           //publish all sensors
data to ubidots
   client.ubidotsPublish("esp8266");
    delay(500);
   client.add("temp", temp.toFloat());
   client.ubidotsPublish("esp8266");
```

```
     delay(500);
    client.add("air", air.toFloat());
    client.ubidotsPublish("esp8266");
    delay(500);
    client.add("water", water.toFloat());
    client.ubidotsPublish("esp8266");
    delay(500);
    client.add("light", light.toFloat());
    client.ubidotsPublish("esp8266");
    delay(500);

    client.loop();
    delay(1000);

}
```