SPRINT - 2

DATE	30 OCTOBER 2022
TEAM ID	PNT2022TMID25296
PROJECT NAME	SMART WASTE MANAGEMENT FOR METROPOLITANCITIES-IOT

CODE :

```
#include <WiFi.h>                    // library

for wifi #include <PubSubClient.h>

                                     // library

for MQTT#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);

//---------------------- credentials of IBM Accounts ----------------------------

#define ORG "wjmfdn"                 // IBM organisation id

#define DEVICE_TYPE "abcd"           // Device type mentioned in ibm watson

iot platform#define DEVICE_ID "1234"                    // Device ID

mentioned in ibm watson iot platform #define TOKEN "12345678"            //

Token

//---------------------- customise above values ---------------------------------------------------

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";          // server name

char publishTopic[] = "iot-2/evt/data/fmt/json";                  // topic name and type of event perform
and format inwhich data to be send

char topic[] = "iot-2/cmd/led/fmt/String";                       // cmd Represent type and command is
test format ofstrings

char authMethod[] = "use-token-auth";                            //

authentication methodchar token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;              //Client id

//-------------------------------------------------------------------------------------------------------------------

WiFiClient wifiClient;                                  // creating instance

for wificlientPubSubClient client(server, 1883, wifiClient);

#define ECHO_PIN 12

#define

TRIG_PI

N 13
```

```
float
dist;
void setup()
{
 Serial.begin(115200);
 pinMode(LED_BUILTIN,
 OUTPUT);
 pinMode(TRIG_PIN,
 OUTPUT);
 pinMode(ECHO_PIN,
 INPUT);
 //pir pin
 pinMode(34, INPUT);
//ledpins
 pinMode(2
 3,
 OUTPUT);
 pinMode(2
 , OUTPUT);
 pinMode(4
 , OUTPUT);
 pinMode(1
 5,
 OUTPUT);
 lcd.init();
 lcd.backlig
 ht();
 lcd.setCurs
 or(1, 0);
 lcd.print(""
 );
 wifiConnec
 t();
```

```
  mqttConne
  ct();
}
float readcmCM()
{
  digitalWrite(TRIG
  _PIN, LOW);
  delayMicrosecon
  ds(2);
  digitalWrite(TRIG
  _PIN, HIGH);
  delayMicrosecon
  ds(10);
  digitalWrite(TRIG
  _PIN, LOW);
  int duration =
  pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
}
void loop()
 {
lc
d.
cl
ea
r()
;
p
u
bli
sh
D
at
a(
```

```
);
de
la
y(
50
0)
;
 if (!client.loop())
  {
    mqttConnect();                          // function call to connect to IBM
  }
}
/* ---------------------------------retrieving to cloud---------------------------------------------- */
void wifiConnect()
{
 Serial.print("Connecting to ");

 Serial.print("Wifi");
 WiFi.begin("Wokwi-
 GUEST", "", 6);
 while (WiFi.status() != WL_CONNECTED)
  {
    delay
    (500)
    ;
    Serial
    .print
    (".");
  }
 Serial.print("WiFi connected, IP
 address: ");
 Serial.println(WiFi.localIP());
}
void mqttConnect()
 {
```

```
  if (!client.connected())
   {
    Serial.print("Reconnecting MQTT
    client to ");Serial.println(server);
    while (!client.connect(clientId, authMethod, token))
     {
      Seri
      al.p
      rint
      ("."
      );
      del
      ay(
      500
      );
     }
    initManagedD
    evice();
    Serial.println()
    ;
   }
 }
void initManagedDevice()
 {
  if (client.subscribe(topic))
   {
    Serial.println("IBM subscribe to cmd OK");
   }
  else
   {
    Serial.println("subscribe to cmd FAILED");
   }
 }
void publishData()
```

```
{
 float cm = readcmCM();
if(digitalRead(34))                    //pir motion detection
 {
  Serial.println("Motion
 Detected");
 Serial.println("Lid
 Opened");
 digitalWrite(15, HIGH);
 if(digitalRead(34)==
 true)
{
 if(cm <= 60)                          //Bin level detection
 {
  digitalWrite(2, HIGH);
  Serial.println("High Alert!!!,Trash bin is about
  to be full");Serial.println("Lid Closed");
  lcd.print("Full!
  Don't use");
  delay(2000);
  lcd.clear();
  digitalWrit
  e(4, LOW);
  digitalWrit
  e(23,
  LOW);
 }
 else if(cm > 60 && cm < 120)
 {
  digitalWrite(4, HIGH);
  Serial.println("Warning!!,Trash is about to cross 50%
  of bin level");digitalWrite(2, LOW);
  digitalWrite(23, LOW);
```

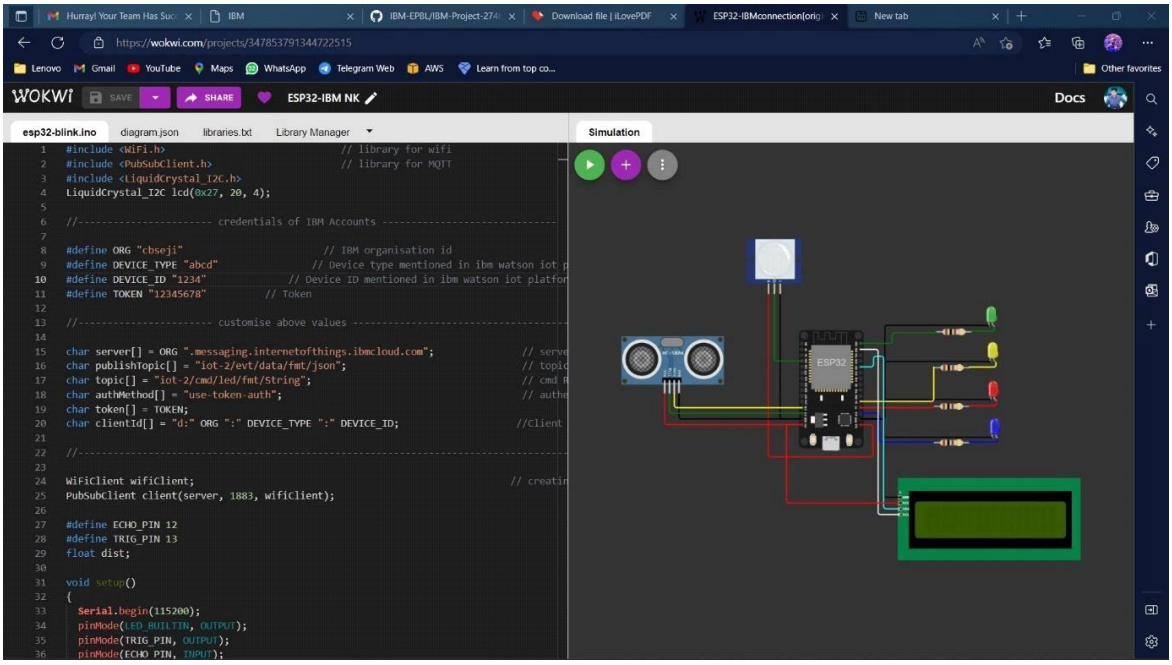
```
 }
 else if(cm > 120)
 {
  digitalWrite(23,
  HIGH);
  Serial.println("Bin is
  available");
  digitalWrite(2,LOW)
  ; digitalWrite(4,
  LOW);
 }
  delay(10000);
  Serial.println("
  Lid Closed");


}
else
{
  Serial.println("No motion
  detected");
  digitalWrite(2, LOW);
  digitalWrite(15, LOW);
  digitalWrite(4, LOW);
  digitalWrite(23, LOW);
}
 }
 else
 {
  digitalWrite(15, LOW);
  }
if(cm <= 60)
{
digitalWrite(21,HIGH);
```

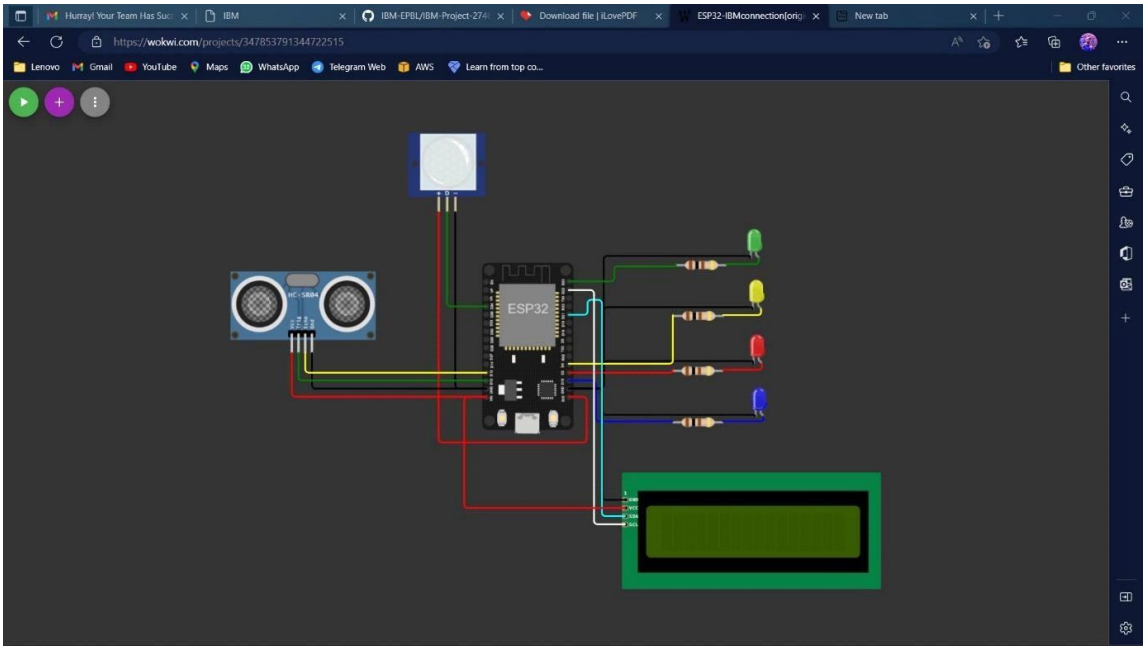
```
String payload =
"{\"High_Alert\":";
payload += cm;
payload += " }";
Serial.print("\n");
Serial.print("Sending
payload: ");
Serial.println(payloa
d);

if (client.publish(publishTopic, (char*) payload.c_str()))        // if data is uploaded to cloud
successfully,prints publish okelse prints publish failed
{
Serial.println("Publish OK");
}
}
else if(cm <= 120)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning\":";
payload += cm ;
payload
+= " }";
Serial.pri
nt("\n");

Serial.print("Sending
payload: ");
Serial.println(payloa
d);
if(client.publish(publishTopic, (char*) payload.c_str()))
{
Serial.println("Publish OK");
```

```
}
else
{
Serial.println("Publish FAILED");
}
}
else
{
Serial.println();
}
 float inches = (cm / 2.54);
                                                    /
  /print on lcdlcd.setCursor(0,0);
  lcd.prin
  t("Inch
  es");
  lcd.set
  Cursor(
  4,0);
  lcd.set
  Cursor(
  12,0);
  lcd.prin
  t("cm")
  ;
  lcd.set
  Cursor(
  1,1);
  lcd.prin
  t(inche
  s, 1);
  lcd.set
  Cursor(
  11,1);
```

```
lcd.prin
t(cm,
1);
lcd.set
Cursor(
14,1);
delay(1
000);
lcd.clea
r();
}
```

CIRCUIT :




Hurray! Your Team Has Suc
IBM
IBM-EPBL/IBM-Project-274
Download file | iLovePDF
ESP32-IBMconnection(orig
New tab
https://wokwi.com/projects/347853791344722515
Lenovo
Gmail
YouTube
Maps
WhatsApp
Telegram Web
AWS
Learn from top co...
Other favorites
ESP32