Source Code
Of
Waste Management System

Course Code:	CSE299
Course Name:	Junior Design
Section:	06

Submitted To: Md. Mahfuzur Rahman

Submission date : 09/10/2020

Group Members:

Student ID:	**Student Name:**
1712314042	Md. Mahir Hasan
1631761042	Tahmina Afroz Prima
1712118042	Anika Tahsin

Source Code

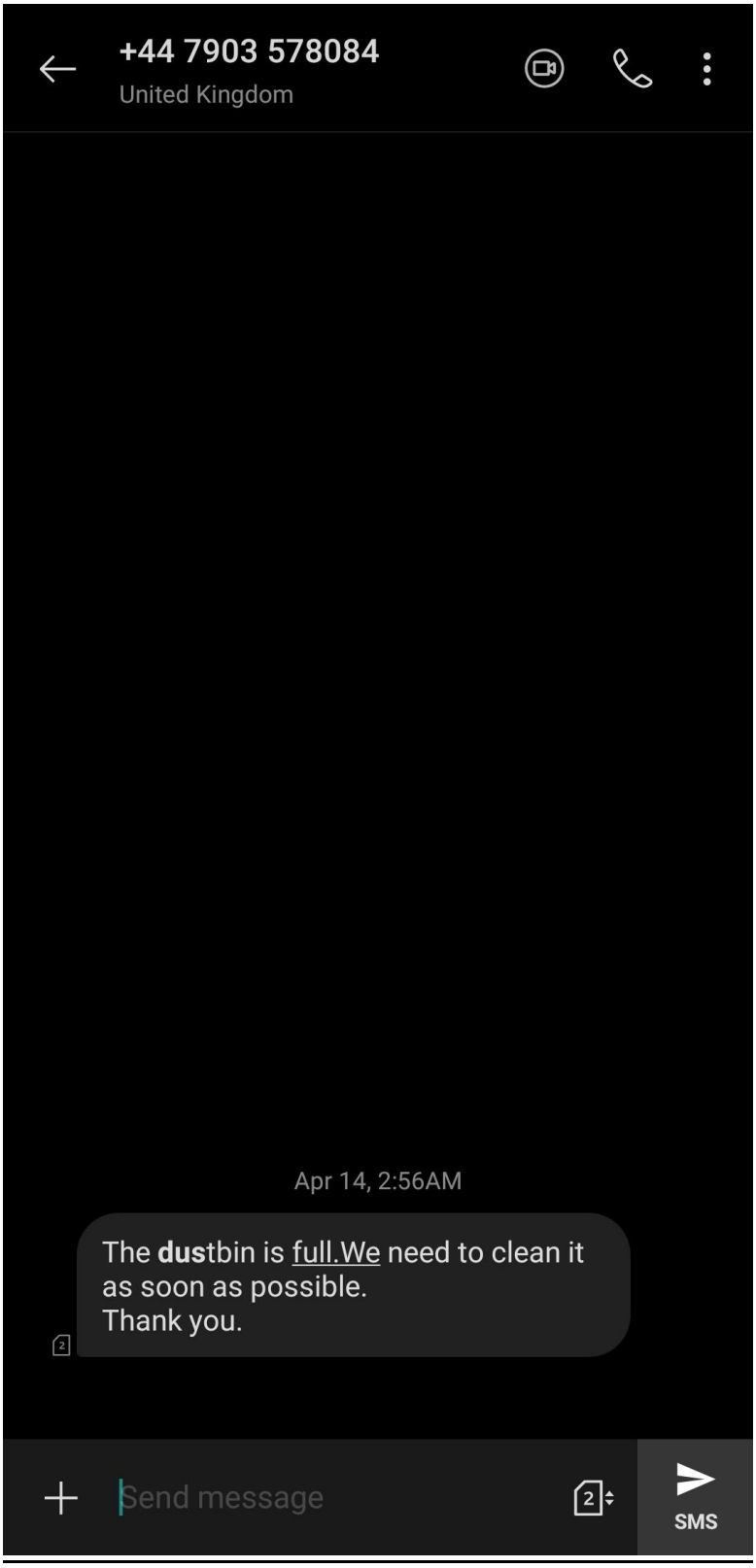
```
#include <ESP8266WiFi.h>
#include "Ubidots.h"
#include <LiquidCrystal_I2C.h>

String apiKey = "2HTEAEQRMPH1CM5U";     //  Enter your Write API key from ThingSpeak

const char *ssid =  "Invincible";     // replace with your wifi ssid and wpa2 key
const char *pass =  "01680098798n";
const char* server = "api.thingspeak.com";

const char* DEVICE_TYPE = "Arduino";
const char* WIFI_SSID = "Invincible";
const char* WIFI_PASS = "01680098798n";
const char* UBIDOTS_TOKEN = "BBFF-YPl79vvXrmaC8NM1z6tDgKD5f9TUEe";
Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);

LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27, 16, 2);

#define led D0
#define buzzer D3
#define echoPin D5
#define trigPin D6

WiFiClient client;

void setup() {

  lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
  lcd.print("Welcome to Waste");
  lcd.setCursor(0,1);
  lcd.print("Management System");
  Serial.begin(9600);
  ubidots.wifiConnect(WIFI_SSID, WIFI_PASS);
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
    }
  Serial.println("");
  Serial.println("WiFi connected");
  pinMode(led, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  lcd.clear();
}
```

```
void loop() {
  long duration, distance;
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(1000);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2)/29.1;
  Serial.print(distance);
  Serial.println("cm");
  delay(500);

  if((distance < 15 && distance  > 7))
  {
    digitalWrite(led, HIGH);
    noTone(buzzer);
    lcd.setCursor(1,0);
    lcd.print("Distance:");
    lcd.print(distance);
    lcd.print("cm");
    lcd.setCursor(1,1);
    lcd.print("Almost full!");

    if (isnan(distance))
                {
                    Serial.println("Failed to read from ultasonic sensor!");
                    return;
                }

    if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
                     {

                            String postStr = apiKey;
                            postStr +="&field1=";
                            postStr += String(distance);
                            postStr += "\r\n\r\n";

                            client.print("POST /update HTTP/1.1\n");
                            client.print("Host: api.thingspeak.com\n");
                            client.print("Connection: close\n");
                            client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
                            client.print("Content-Type: application/x-www-form-
urlencoded\n");
                            client.print("Content-Length: ");
                            client.print(postStr.length());
                            client.print("\n\n");
                            client.print(postStr);
                            Serial.println(". Send to Thingspeak.");
                       }
          client.stop();

          Serial.println("Waiting...");

  // thingspeak needs minimum 15 sec delay between updates
  delay(500);
```

```
  lcd.clear();

  }
  else if(distance < 7){
    digitalWrite(led, LOW);
    tone(buzzer,2000,200);
    delay(500);
    lcd.setCursor(1,0);
    lcd.print("Distance:");
    lcd.print(distance);
    lcd.print("cm");
    lcd.setCursor(1,1);
    lcd.print("Totally full");

    ubidots.add("Dustbin message", distance);

    bool bufferSent = false;
    bufferSent = ubidots.send();
    if (bufferSent) {
    Serial.println("Values sent by the device");
    }

    delay(2000);


    if (isnan(distance))
                {
                    Serial.println("Failed to read from ultasonic sensor!");
                    return;
                }

    if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
                      {

                            String postStr = apiKey;
                            postStr +="&field1=";
                            postStr += String(distance);
                            postStr += "\r\n\r\n";

                            client.print("POST /update HTTP/1.1\n");
                            client.print("Host: api.thingspeak.com\n");
                            client.print("Connection: close\n");
                            client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
                            client.print("Content-Type: application/x-www-form-
urlencoded\n");
                            client.print("Content-Length: ");
                            client.print(postStr.length());
                            client.print("\n\n");
                            client.print(postStr);
                            Serial.println(". Send to Thingspeak.");
                       }
          client.stop();
          Serial.println("Waiting...");

  // thingspeak needs minimum 15 sec delay between updates
```

```
  delay(500);
  lcd.clear();
  }
  else
  {
    digitalWrite(led, LOW);
    noTone(buzzer);
    lcd.setCursor(1,0);
    lcd.print("Distance:");
    lcd.print(distance);
    lcd.print("cm");
    lcd.setCursor(1,1);
    lcd.print("Use me!");

    if (isnan(distance))
                {
                    Serial.println("Failed to read from ultasonic sensor!");
                    return;
                }
    if (client.connect(server,80))   //   "184.106.153.149" or api.thingspeak.com
                     {

                            String postStr = apiKey;
                            postStr +="&field1=";
                            postStr += String(distance);
                            postStr += "\r\n\r\n";

                            client.print("POST /update HTTP/1.1\n");
                            client.print("Host: api.thingspeak.com\n");
                            client.print("Connection: close\n");
                            client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
                            client.print("Content-Type: application/x-www-form-
urlencoded\n");
                            client.print("Content-Length: ");
                            client.print(postStr.length());
                            client.print("\n\n");
                            client.print(postStr);
                            Serial.println(". Send to Thingspeak.");
                       }
          client.stop();

          Serial.println("Waiting...");

  // thingspeak needs minimum 15 sec delay between updates
  delay(500);
  lcd.clear();
  }
}
```

Output SMS


+44 7903 578084
United Kingdom
Apr 14, 2:56AM
The dustbin is full.We need to clean it as soon as possible.
Thank you.
Send message
SMS