```cpp
#include <DHT.h>

#include <SoftwareSerial.h>

#include <LiquidCrystal.h>

const int rs = 13, en = 12, d4 = 11, d5 = 10, d6 = 9, d7 = 8;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

String apiKey = "6JBNGHDFZGJHFDKGHJHG";

SoftwareSerial ser(2, 3); // RX, TX

#define DHTPIN A0

#define DHTTYPE DHT11

DHT dht ( DHTPIN, DHTTYPE );

const int S1 = A1;

int humi,temp;

int LEDPin = 4;

int MOT1 = 5;

void setup()

{

  digitalWrite(LEDPin,LOW);

  digitalWrite(MOT1,LOW);

    lcd.begin(16,2);

  lcd.print("WELCOME");

  lcd.setCursor(0,1);

  lcd.print("USE IOT");

  delay(1000);

    unsigned char check_connection=0;

    unsigned char times_check=0;
```

```
  Serial.println("Connecting to Wifi");

 while(check_connection==0)

 {

 if(ser.find("WIFI CONNECTED\r\n")==1 )

 {

  lcd.clear();

  lcd.print("WIFI CONNECTED");

 Serial.println("WIFI CONNECTED");

 delay(1000);

 lcd.clear();

 break;

 }

 times_check++;

 if(times_check>3)

 {

  times_check=0;

  Serial.println("Trying to Reconnect..");

   lcd.clear();

  lcd.print("TRYING TO RECONNECT");

   delay(1000);

 lcd.clear();

  }

 }

 delay(5000);

}
```

```
void loop()

{

  int S,T,O;

  int SPO = digitalRead(S1);

  int MOI=digitalRead(A2);

float humi;

float temp ;

 float temp1 = temp * 0.48828125;

  Serial.print("H:");

  Serial.print(humi);

  Serial.println("%");

  Serial.print("T:");

  Serial.print(temp);

  Serial.println("C");

  lcd.setCursor(0,0);

  lcd.print("H:");

  lcd.setCursor(9,0);

  lcd.print("T:");

  lcd.setCursor(0,1);

  lcd.print("B:");

  lcd.setCursor(7,1);

  lcd.print("O:");

  // TCP connection

  if(ser.find("Error")){

    Serial.println("AT+CIPSTART error");
```

```
  return;
}
// prepare GET string
String getStr = "GET /update?api_key=";
if(humi<75&&temp<40&&SPO==HIGH)
{
digitalWrite(MOT1,LOW);
lcd.setCursor(3,0);
lcd.print(humi);
lcd.setCursor(11,0);
lcd.print(temp);
lcd.setCursor(3,1);
lcd.print(S);
lcd.setCursor(11,1);
lcd.print(S);
getStr += apiKey;
getStr +="&field1=";
getStr += String(humi);
getStr +="&field2=";
getStr += String(temp);
 getStr +="&field3=";
getStr += String(S);
getStr +="&field4=";
getStr += String(S);
getStr += "\r\n\r\n\r\n";
```

```
//if(HEART==LOW)

// {

// lcd.setCursor(11,1);

//lcd.print(s);


//getStr +="&field3=";

//getStr += String(s);

//getStr += "\r\n\r\n\r\n";

//cmd = "AT+CIPSEND=";

//cmd += String(getStr.length());

//ser.println(cmd);

 //}


// send data length

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

ser.println(cmd);

}
if(humi>75)

{


digitalWrite(LEDPin,HIGH);

delay(1500);

digitalWrite(LEDPin,LOW);

digitalWrite(MOT1,LOW);
```

```
lcd.setCursor(3,0);

lcd.print(humi);

lcd.setCursor(11,0);

lcd.print(temp);

lcd.setCursor(3,1);

lcd.print(S);

lcd.setCursor(11,1);

lcd.print(S);

getStr += apiKey;

getStr +="&field1=";

getStr += String(humi);

getStr +="&field2=";

getStr += String(temp);

 getStr +="&field3=";

getStr += String(S);

 getStr +="&field4=";

getStr += String(S);

getStr += "\r\n\r\n\r\n";

//if(HEART==LOW)

// {

// lcd.setCursor(11,1);

//lcd.print(s);


//getStr +="&field3=";

//getStr += String(s);
```

```cpp
//getStr += "\r\n\r\n\r\n";

//cmd = "AT+CIPSEND=";

//cmd += String(getStr.length());

//ser.println(cmd);

 //}


// send data length

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

ser.println(cmd);

}
 if(temp>40)
{
digitalWrite(LEDPin,HIGH);

delay(1500);

digitalWrite(LEDPin,LOW);

digitalWrite(MOT1,LOW);

lcd.setCursor(3,0);

lcd.print(humi);

lcd.setCursor(11,0);

lcd.print(temp);

lcd.setCursor(3,1);

lcd.print(S);

lcd.setCursor(11,1);

lcd.print(S);
```

```
getStr += apiKey;

getStr +="&field1=";

getStr += String(humi);

getStr +="&field2=";

getStr += String(temp);

 getStr +="&field3=";

getStr += String(S);

 getStr +="&field4=";

 getStr += String(S);

getStr += "\r\n\r\n\r\n";

//if(HEART==LOW)

// {

// lcd.setCursor(11,1);

//lcd.print(s);


//getStr +="&field3=";

//getStr += String(s);

//getStr += "\r\n\r\n\r\n";

//cmd = "AT+CIPSEND=";

//cmd += String(getStr.length());

//ser.println(cmd);

 //}


// send data length

cmd = "AT+CIPSEND=";
```

```
cmd += String(getStr.length());

ser.println(cmd);

}

 if(SPO==LOW)

{

digitalWrite(MOT1,HIGH);

lcd.setCursor(3,0);

lcd.print(humi);

lcd.setCursor(11,0);

lcd.print(temp);

lcd.setCursor(3,1);

lcd.print(T);

lcd.setCursor(11,1);

lcd.print(O);

getStr += apiKey;

getStr +="&field1=";

getStr += String(humi);

getStr +="&field2=";

getStr += String(temp);

 getStr +="&field3=";

getStr += String(T);

getStr +="&field4=";

getStr += String(O);

getStr += "\r\n\r\n\r\n";

//if(HEART==LOW)
```

```
// {

// lcd.setCursor(11,1);

//lcd.print(s);


//getStr +="&field3=";

//getStr += String(s);

//getStr += "\r\n\r\n\r\n";

//cmd = "AT+CIPSEND=";

//cmd += String(getStr.length());

//ser.println(cmd);

 //}


// send data length

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

ser.println(cmd);

}


if(ser.find(">")){

 ser.print(getStr);

 Serial.println(getStr);

}
else{

 ser.println("AT+CIPCLOSE");

 Serial.println("CIPCLOSE");
```

```
  }


  // thingspeak needs 10 sec delay between updates

  delay(10000);

  lcd.clear();

}
```