Tech Manual IoT

In this manual I will try to sent weather information to Telegram.

What I used:

- NodeMCU ESP8266

The API I used:

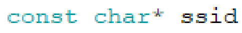
- OpenWeatherMap

Step 1 Getting your key and setting up weather forecast in arduino:

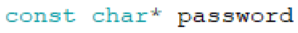
For this step I used: https://randomnerdtutorials.com/esp8266-nodemcu-http-get-open-weather-map-thingspeak-arduino/

I stopped following the tutorial before 'ESP8266 HTTP GET: Update Value (ThingSpeak)'. I wanted to try and send the weather information to Telegram.

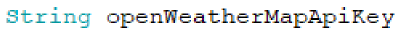
I didn't get any errors following the tutorial. The only things I changed were these:



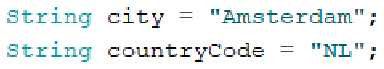
```
const char* ssid
```



```
const char* password
```



```
String openWeatherMapApiKey
```

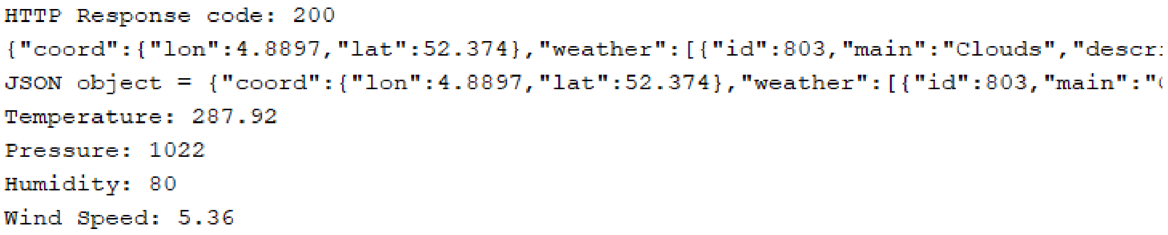


```
String city = "Amsterdam";
String countryCode = "NL";
```

These are things you should change depending on your ssid, password, API key and location.

Note that the API key won't work right away. The site of the API said that it'll take a couple hours before it works. In my case it worked in a little less than an hour.

If everything works correctly you'll see something similar tot this (depending on your location) in the serial monitor:



```
HTTP Response code: 200
{"coord":{"lon":4.8897,"lat":52.374},"weather":[{"id":803,"main":"Clouds","descr
JSON object = {"coord":{"lon":4.8897,"lat":52.374},"weather":[{"id":803,"main":"
Temperature: 287.92
Pressure: 1022
Humidity: 80
Wind Speed: 5.36
```

Note that if you're using the free subscription from OpenWeatherMap you can only call on the API 60 times per minute. Otherwise you can get banned.

Checkpoint: The arduino code for getting weather information works.

Step 2: Creating a bot in Telegram

To install a bot in Telegram you’ll need to have the Telegram app.

Once you’ve installed it you need to search ‘BotFather’.

When you’ve started a chat with BotFather you can just follow the steps to create a new bot.

After you’ve created a bot you’ll get a token. It’s important that you safe the token as you’ll need to use it later on.

Checkpoint: You’ve created a bot in Telegram

Step 3: Get your Telegram ID

To get your Telegram ID, which you’ll use later, you’ll need to search ‘IDBot’ in Telegram.

After that you’ll need to type ‘/start’ and then ‘/getid’. Once you’ve got that ID you’ll need to safe that aswell.

Checkpoint: You’ve got an ID

Step 4: The Code

Below this point is the code I have. I really don’t know how arduino works to be honest. I tried to do something which I don’t know what.

The code itself has no errors, but it won’t send anything to Telegram. After this point I just don’t know what to do.

```
#include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>

#include <WiFiClient.h>

#include <Arduino_JSON.h>

#include <UniversalTelegramBot.h>


const char* ssid = "LSA72";

const char* password = "dlvl5923";


#define CHAT_ID "2013508067"


#define BOTtoken "2081592009:AAFMvTRqRuAjKm0G4tKg4ma4rJDRC4rpoWo"


WiFiClientSecure client;

UniversalTelegramBot bot(BOTtoken, client);


int botRequestDelay = 1000;

unsigned long lastTimeBotRan;


void handleNewMessages(int numNewMessages) {

  Serial.println("handleNewMessages");

  Serial.println(String(numNewMessages));


  for (int i=0; i<numNewMessages; i++) {

    // Chat id of the requester

    String chat_id = String(bot.messages[i].chat_id);

    if (chat_id != CHAT_ID){

      bot.sendMessage(chat_id, "Unauthorized user", "");

      continue;

    }
    // Print the received message
    String text = bot.messages[i].text;
    Serial.println(text);

    String from_name = bot.messages[i].from_name;

    if (text == "/start") {
      String welcome = "Welcome, " + from_name + ".\n";
      welcome += "Use the following command to get current weather.\n\n";
      welcome += "/weather \n";
      bot.sendMessage(chat_id, welcome, "");
    }

    if (text == "/weather") {
      bot.sendMessage(chat_id, "The weather is good", "");
    }
  }
}


// Your Domain name with URL path or IP address with path
String openWeatherMapApiKey = "220b37adf2606cef06541a3f01a50d2e";
// Example:
//String openWeatherMapApiKey = "bd939aa3d23ff33d3c8f5dd1dd4";

// Replace with your country code and city
String city = "Amsterdam";
String countryCode = "NL";

// THE DEFAULT TIMER IS SET TO 10 SECONDS FOR TESTING PURPOSES
// For a final application, check the API call limits per hour/minute to avoid getting blocked/banned
unsigned long lastTime = 0;
// Timer set to 10 minutes (600000)
//unsigned long timerDelay = 600000;
// Set timer to 10 seconds (10000)
unsigned long timerDelay = 10000;

String jsonBuffer;
```

```
void setup() {
  Serial.begin(115200);

  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

  Serial.println("Timer set to 10 seconds (timerDelay variable), it will take 10 seconds before publishing the first reading.");
}

void loop() {
  // Send an HTTP GET request
  if ((millis() - lastTime) > timerDelay) {
    // Check WiFi connection status
    if(WiFi.status()== WL_CONNECTED){
      String serverPath = "http://api.openweathermap.org/data/2.5/weather?q=" + city + "," + countryCode + "&APPID=" + openWeatherMapApiKey;

      jsonBuffer = httpGETRequest(serverPath.c_str());
      Serial.println(jsonBuffer);
      JSONVar myObject = JSON.parse(jsonBuffer);

      // JSON.typeof(jsonVar) can be used to get the type of the var
      if (JSON.typeof(myObject) == "undefined") {
        Serial.println("Parsing input failed!");
        return;
      }

      Serial.print("JSON object = ");
      Serial.println(myObject);
      Serial.print("Temperature: ");
      Serial.println(myObject["main"]["temp"]);
      Serial.print("Pressure: ");
      Serial.println(myObject["main"]["pressure"]);
      Serial.print("Humidity: ");
      Serial.println(myObject["main"]["humidity"]);
      Serial.print("Wind Speed: ");
      Serial.println(myObject["wind"]["speed"]);
    }
    else {
      Serial.println("WiFi Disconnected");
    }
    lastTime = millis();
  }
}

String httpGETRequest(const char* serverName) {
  WiFiClient client;
  HTTPClient http;

  // Your IP address with path or Domain name with URL path
  http.begin(client, serverName);

  // Send HTTP POST request
  int httpResponseCode = http.GET();

  String payload = "{}";

  if (httpResponseCode>0) {
    Serial.print("HTTP Response code: ");
    Serial.println(httpResponseCode);
    payload = http.getString();
  }
  else {
    Serial.print("Error code: ");
    Serial.println(httpResponseCode);
  }
  // Free resources
  http.end();
  return payload;
}
```