Tutorial: Create ESP8266 NodeMCU Web Server using WiFi Station (STA) mode

SCENARIO: We are going to configure our NodeMCU into Station (STA) mode, and create a web server to serve up web pages to any connected client under existing network.

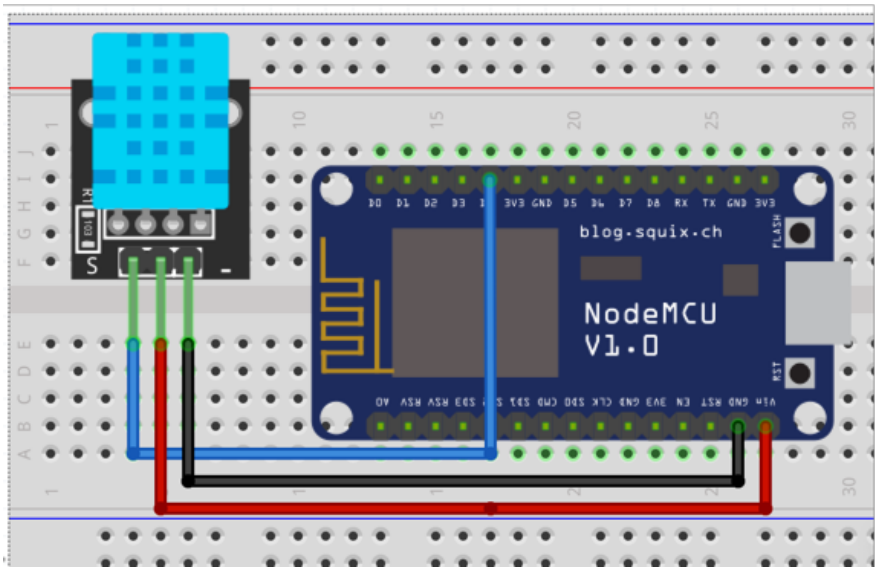

Figure 1: The wiring diagram

We need to modify the following two variables with your network credentials, so that NodeMCU can establish a connection with existing network.

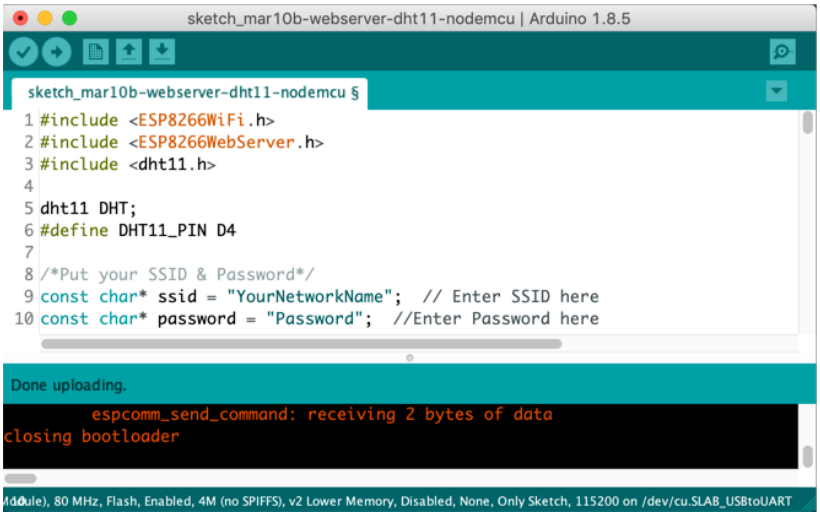

Figure 2: Remember to change *YourNetworkName* & *Password*

Copy and paste the following sketch into your IDE.

```
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <dht11.h>

dht11 DHT;
#define DHT11_PIN D4

/*Put your SSID & Password*/
const char* ssid = "YourNetworkName";  // Enter SSID here
const char* password = "YourPassword";  //Enter Password here

// declare an object of WebServer library
ESP8266WebServer server(80);

int Temperature;
int Humidity;

void setup() {
  Serial.begin(115200);
  delay(100);
  Serial.println("Interface DHT11 NodeMCU Using Web Server");
  Serial.println("Connecting to ");
  Serial.println(ssid);

  //connect to your local wi-fi network
  WiFi.begin(ssid, password);

  //check wi-fi is connected to wi-fi network
  while (WiFi.status() != WL_CONNECTED) {
  delay(1000);
  Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected..!");
  Serial.print("Got IP: ");
  Serial.println(WiFi.localIP());

  server.on("/", handle_OnConnect); //root (/)
  server.onNotFound(handle_NotFound);

  server.begin();
  Serial.println("HTTP server started");

}
void loop() {

  server.handleClient(); //to handle actual HTTP request

}
```

```
void handle_OnConnect() {
int chk = DHT.read(DHT11_PIN);    // READ DATA

 Temperature = DHT.temperature; // Gets the values of the temperature
  Humidity = DHT.humidity; // Gets the values of the humidity
  Serial.print(Temperature);
  Serial.print("&deg;C, ");
  Serial.print(Humidity);
  Serial.println("%");
  server.send(200, "text/html", SendHTML(Temperature,Humidity));  //200 corresponds to the OK response
}

void handle_NotFound(){
  server.send(404, "text/plain", "Not found");
}
//Displaying the HTML Web Page & css
String SendHTML(float Temperaturestat,float Humiditystat){
  String ptr = "<!DOCTYPE html> <html>\n";
  ptr +="<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0, user-scalable=no\">\n";
  ptr +="<title>NodeMCU 8266 Weather Report</title>\n";
  ptr +="<style>html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}\n";
  ptr +="body{margin-top: 50px;} h1 {color: #444444;margin: 50px auto 30px;}\n";
  ptr +="p {font-size: 24px;color: #444444;margin-bottom: 10px;}\n";
  ptr +="</style>\n";
  ptr +="</head>\n";
  ptr +="<body>\n";
  ptr +="<div id=\"webpage\">\n";
  ptr +="<h1>NodeMCU Weather Report at PMJ</h1>\n";

  ptr +="<p>Temperature: ";
  ptr +=(int)Temperaturestat;
  ptr +="°C</p>";
  ptr +="<p>Humidity: ";
  ptr +=(int)Humiditystat;
  ptr +="%</p>";

  ptr +="</div>\n";
  ptr +="</body>\n";
  ptr +="</html>\n";
  return ptr;
}
```

Accessing the Web Server

Once uploaded, open Serial Monitor & observe the setup as in Figure 3. Press Reset, of everything ok it will output the dynamic IP address obtained from your router and show HTTP server started message as in Figure 4.

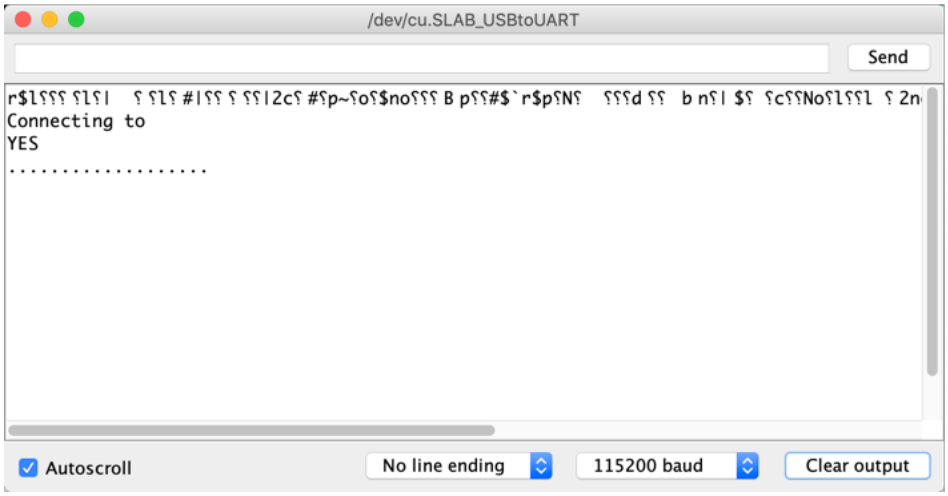

Figure 3: Still not connected – wrong pwd; no network connection

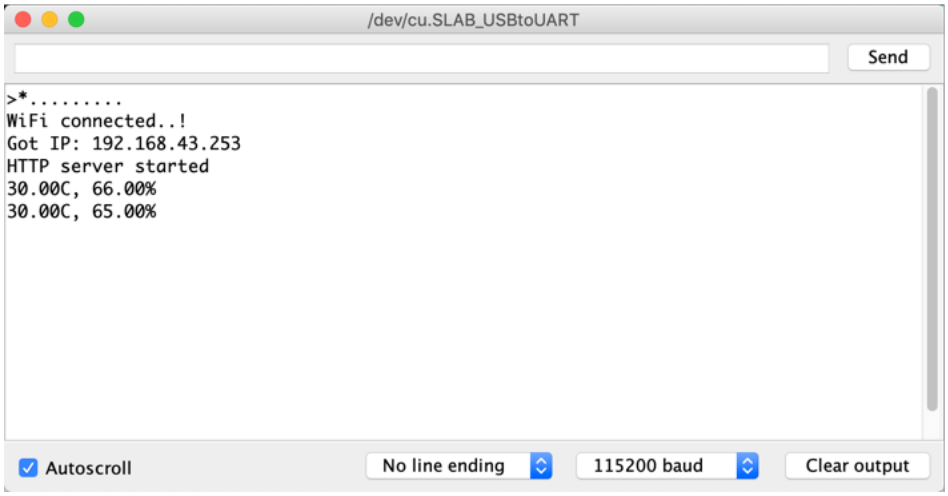

Figure 4: Connected – the temperature & humid will appear every time the page is surfed

The Output

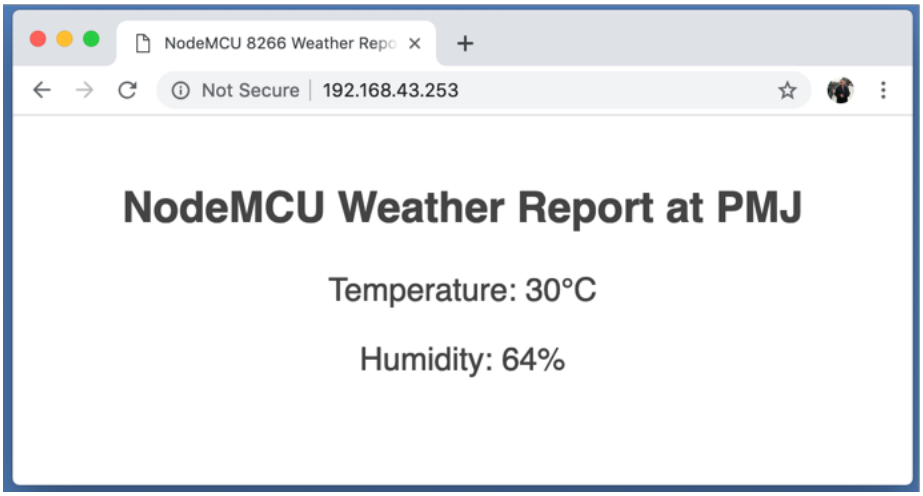

Figure 5: Browsed from laptop

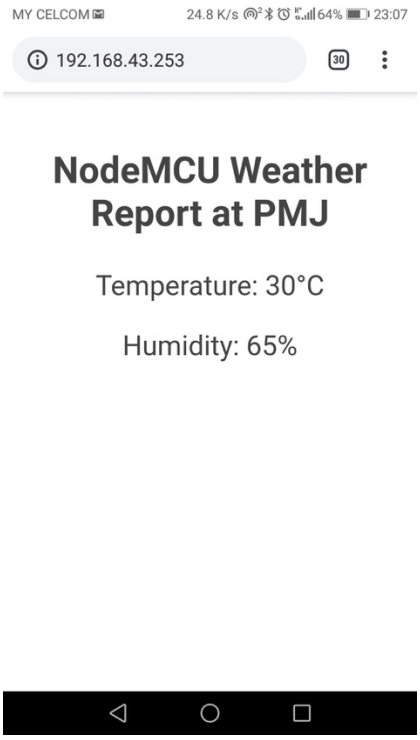

Figure 6: Browsed from smartphone

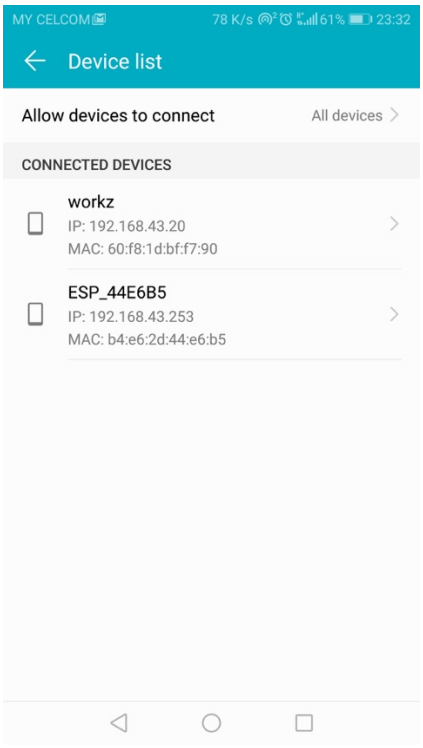

Figure 7: Same subnet

Styling Web Page to Look More Professional

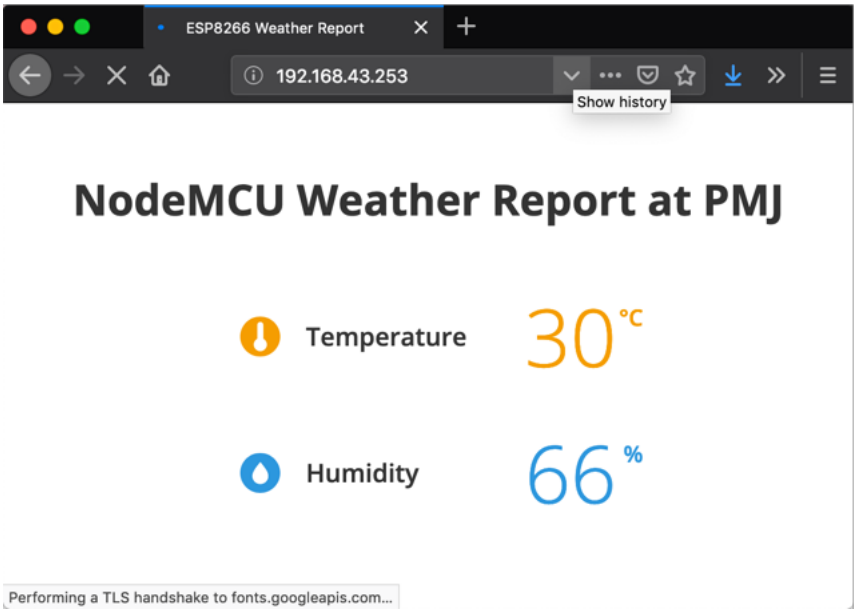

Figure 8: More interactive looks – copy & replace the SendHTML function to the previous sketch and save as different filename

```
String SendHTML(float TempCstat, float Humiditystat){
  String ptr = "<!DOCTYPE html> <html>\n";
  ptr +="<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0, user-scalable=no\">\n";
  ptr +="<link href=\"https://fonts.googleapis.com/css?family=Open+Sans:300,400,600\" rel=\"stylesheet\">\n";
  ptr +="<title>ESP8266 Weather Report</title>\n";
  ptr +="<style>html { font-family: 'Open Sans', sans-serif; display: block; margin: 0px auto; text-align: center;color: #333333;}\n";
  ptr +="body{margin-top: 50px;}\n";
  ptr +="h1 {margin: 50px auto 30px;}\n";
  ptr +=".side-by-side{display: inline-block;vertical-align: middle;position: relative;}\n";
  ptr +=".humidity-icon{background-color: #3498db;width: 30px;height: 30px;border-radius: 50%;line-height: 36px;}\n";
  ptr +=".humidity-text{font-weight: 600;padding-left: 15px;font-size: 19px;width: 160px;text-align: left;}\n";
  ptr +=".humidity{font-weight: 300;font-size: 60px;color: #3498db;}\n";
  ptr +=".temperature-icon{background-color: #f39c12;width: 30px;height: 30px;border-radius: 50%;line-height: 40px;}\n";
  ptr +=".temperature-text{font-weight: 600;padding-left: 15px;font-size: 19px;width: 160px;text-align: left;}\n";
  ptr +=".temperature{font-weight: 300;font-size: 60px;color: #f39c12;}\n";
  ptr +=".superscript{font-size: 17px;font-weight: 600;position: absolute;right: -20px;top: 15px;}\n";
  ptr +=".data{padding: 10px;}\n";
  ptr +="</style>\n";
  ptr +="</head>\n";
  ptr +="<body>\n";

   ptr +="<div id=\"webpage\">\n";

   ptr +="<h1>NodeMCU Weather Report at PMJ</h1>\n";
   ptr +="<div class=\"data\">\n";
   ptr +="<div class=\"side-by-side temperature-icon\">\n";
   ptr +="<svg version=\"1.1\" id=\"Layer_1\" xmlns=\"http://www.w3.org/2000/svg\" xmlns:xlink=\"http://www.w3.org/1999/xlink\" x=\"0px\"
y=\"0px\"\n";
   ptr +="width=\"9.915px\" height=\"22px\" viewBox=\"0 0 9.915 22\" enable-background=\"new 0 0 9.915 22\" xml:space=\"preserve\">\n";
   ptr +="<path fill=\"#FFFFFF\" d=\"M3.498,0.53c0.377-0.331,0.877-0.501,1.374-0.527C5.697-0.04,6.522,0.421,6.924,1.142\n";
   ptr +="c0.237,0.399,0.315,0.871,0.311,1.33C7.229,5.856,7.245,9.24,7.227,12.625c1.019,0.539,1.855,1.424,2.301,2.491\n";
   ptr +="c0.491,1.163,0.518,2.514,0.062,3.693c-0.414,1.102-1.24,2.038-2.276,2.594c-1.056,0.583-2.331,0.743-3.501,0.463\n";
   ptr +="c-1.417-0.323-2.659-1.314-3.3-2.617C0.014,18.26-0.115,17.104,0.1,16.022c0.296-1.443,1.274-2.717,2.58-3.394\n";
   ptr +="c0.013-3.44,0-6.881,0.007-10.322C2.674,1.634,2.974,0.955,3.498,0.53z\"/>\n";
   ptr +="</svg>\n";
   ptr +="</div>\n";
   ptr +="<div class=\"side-by-side temperature-text\">Temperature</div>\n";
   ptr +="<div class=\"side-by-side temperature\">";
   ptr +=(int)TempCstat;
   ptr +="<span class=\"superscript\">&deg;C</span></div>\n";
   ptr +="</div>\n";
   ptr +="<div class=\"data\">\n";
   ptr +="<div class=\"side-by-side humidity-icon\">\n";
   ptr +="<svg version=\"1.1\" id=\"Layer_2\" xmlns=\"http://www.w3.org/2000/svg\" xmlns:xlink=\"http://www.w3.org/1999/xlink\" x=\"0px\"
y=\"0px\"\n\"; width=\"12px\" height=\"17.955px\" viewBox=\"0 0 13 17.955\" enable-background=\"new 0 0 13 17.955\"
xml:space=\"preserve\">\n";
   ptr +="<path fill=\"#FFFFFF\" d=\"M1.819,6.217C3.139,4.064,6.5,0,6.5,0s3.363,4.064,4.681,6.217c1.793,2.926,2.133,5.05,1.571,7.057\n";
   ptr +="c-0.438,1.574-2.264,4.681-6.252,4.681c-3.988,0-5.813-3.107-6.252-4.681C-0.313,11.267,0.026,9.143,1.819,6.217\"></path>\n";
   ptr +="</svg>\n";
   ptr +="</div>\n";
   ptr +="<div class=\"side-by-side humidity-text\">Humidity</div>\n";
   ptr +="<div class=\"side-by-side humidity\">";
   ptr +=(int)Humiditystat;
   ptr +="<span class=\"superscript\">%</span></div>\n";
   ptr +="</div>\n";

  ptr +="</div>\n";
  ptr +="</body>\n";
  ptr +="</html>\n";
  return ptr;
}
```

Improvement to the Code – Auto Page Refresh

One of the improvements you can do with our code is refreshing the page automatically in order to update the sensor value.

With the addition of a single meta tag into your HTML document, you can instruct the browser to automatically reload the page at a provided interval.

```
<meta http-equiv="refresh" content="2" >
```

Place this code in the <head> tag of your document, this meta tag will instruct the browser to refresh every two seconds.