**Nyíregyházi Egyetem**

**Programtervező Informatikus Alapképzési szak**

**Hálózati operációs rendszerek és IOT technológia**

**Otthonunk ideális páratartalmának beállítása. Az adatok mérése és rögzítése, a család egészségének megőrzése céljából**

Készítette:

Név: Győri László

Neptun kód: WBHQXM

Tanév: 2022-23

A levegő hőmérséklete mellett, a [páratartalom](https://www.webbeteg.hu/cikkek/legzoszervi_betegseg/1579/a-paratartalom-hatasai-szervezetunkre) is nagyon lényeges összetevője a minőségnek. Az egészséges felnőtt ember számára a 40-60 százalékos páratartalom az optimális, mind a komfortérzet, mind az egészségmegőrzés céljából. Kisgyermekek szobájában ennél magasabb is lehet a páratartalom, náluk a 60-70 százalékos az ideális. A túl alacsony páratartalom kiszáradásra hajlamosít. Ilyen környezetben a nyálkahártyák irritációja könnyebben kialakul. Ennek jelei: a [szemek szárazsága](https://www.webbeteg.hu/cikkek/szem_betegsegei/1410/a-szarazszemuseg), égő érzése; torokkaparás, szájszárazság; a légutak nyálkahártyájának irritációját száraz köhögés jelzi.

A levegő túl magas páratartalma sem egészséges, ugyanis kedvez számos kórokozó, atka és gomba elszaporodásának. A túl párás levegő növeli az allergiák kialakulásának kockázatát, elsősorban a házipor (poratka) esetében. Emellett a magas páratartalom káros a házak szerkezetére is: könnyebben penészedik (penészgombák), a falak dohosá válnak.

Otthonom páratartalmának mérésére egy Wemos D1 mini fejlesztői kártyát, hozzá pedig egy DHT11-es hő és páratartalom érzékelőt vásároltam

A DHT11 egy egyszerű, alacsony költségű digitális hőmérséklet- és páratartalom-érzékelő. Kapacitív páratartalom-érzékelőt és termisztort használ a környező levegő mérésére, és digitális jelet ad ki a kimeneten. Ez a kisméretű és kis fogyasztású szenzor nagyobb távolságokba is képes eljuttatni az adatokat.



3 lába van (GMD, adat láb, VCC).

Méréshatárok:

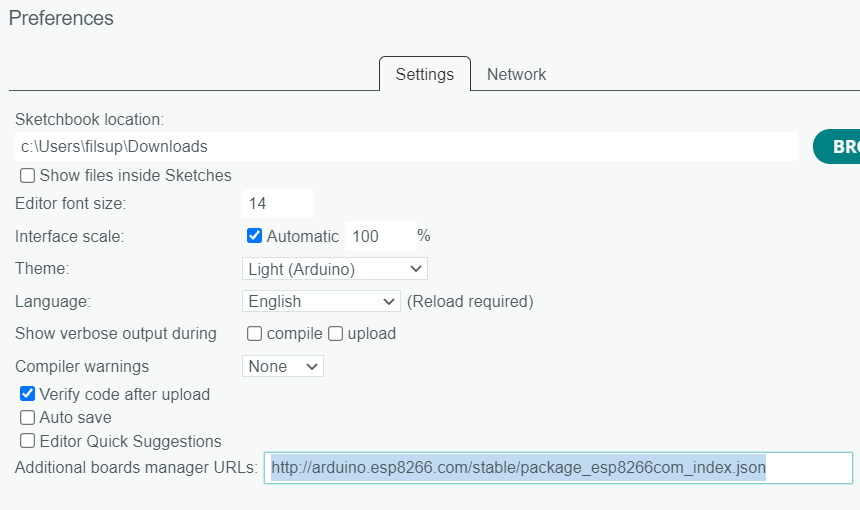
Hőmérsékletet 0-50 celsius fok között +-2 C°-, páratartalmat pedig 20-90% között +-5% eltéréssel mérhetünk vele. Működéséhez 3,5-5 V feszültség szükséges.

Működése:

A kontroller küld egy indító jelet (start signal). A szenzor ilyenkor átvált kis fogyasztású módról működési módra és várja, hogy a kontroller befejezze a startszignált. Amikor befejeződött akkor a DHT11 küld egy 40 bites válaszjelet, mely tartalmazza a hőmérsékletet és a relatív páratartalom információt a vezérlő számára. Ezután már képesek vagyunk kiolvasni az adatokat. Startszignál nélkül nincs válaszjel. Az adatok összegyűjtése után a DHT11 visszaáll kisfogyasztású üzemmódba, egészen a köv. indítójelig.

Én az Arduino IDE-t telepítettem fel a számítógépemre, majd az ESP 8266 magot:

File menü ->Preferences -> Additional boards manager URLs sorban hozzáadjuk a következő fájlt az IDE-hez: <http://arduino.esp8266.com/stable/package_esp8266com_index.json>'



Ezután a Tools menü -> Boards almenü -> Boards Manager. Itt a listából kikeressük az ESP 8266 by ESP8266 Community-t és az is feltelepítjük.

Ezután kezdhetünk neki a programunk elkészítéséhez:

#include "ESP8266WiFi.h"

#include "WiFiClientSecure.h"

#include "SimpleDHT.h"

const char\* wifi\_ssid = "UPC1427586";             // SSID

const char\* wifi\_password = "\*\*\*\*\*\*\*\*\*\*\*\*";         // WIFI kód

const char\* apiKeyIn = "nbuZndzsAmSOjQvaXVpDgEDV6sPagRAl";      // API KEY

const unsigned int writeInterval = 25000;

const char\* https\_host = "api.asksensors.com";         // ASKSENSORS host név

const int https\_port = 443;                        // https port

const char\* https\_fingerprint =  "E0 B3 5B 92 2B F3 48 FA 1E E3 07 6F 01 06 0D D1 AC B0 9B 85";

int pinDHT11 = 2;

SimpleDHT11 dht11(pinDHT11);

WiFiClientSecure client;

void getTem()

{

  byte temperature = 0;

  byte humidity = 0;

  int err = SimpleDHTErrSuccess;

  if ((err = dht11.read(&temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {

    delay(100);

    return;

  }

  if (!client.connect(https\_host, https\_port)) {

  delay(200);

  return;

  }

  String url = "/write/";

  url += apiKeyIn;

  url += "?module1=";

  url += ((int)temperature);

  url += "&module2=";

  url += ((int)humidity);

  client.print(String("GET ") + url + " HTTP/1.1\r\n" +

               "Host: " + https\_host + "\r\n" +

               "Connection: close\r\n\r\n");

  client.connected();

}

void setup() {

  WiFi.begin(wifi\_ssid, wifi\_password);

  while (WiFi.status() != WL\_CONNECTED) {

    delay(500);

  }

  client.setInsecure();

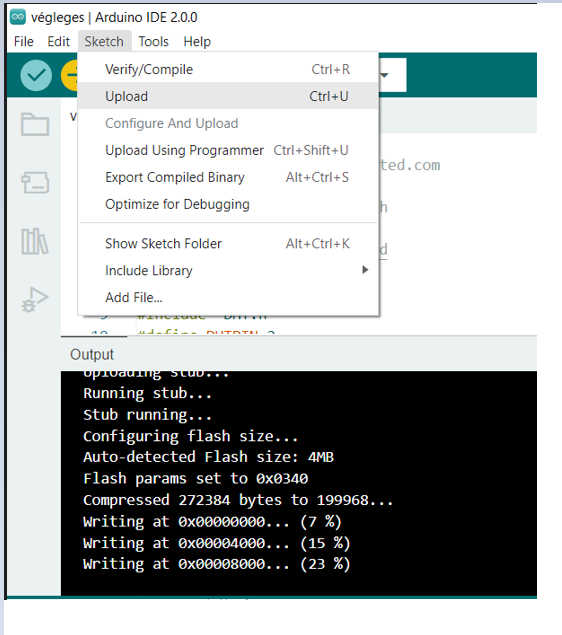
}

void loop() {

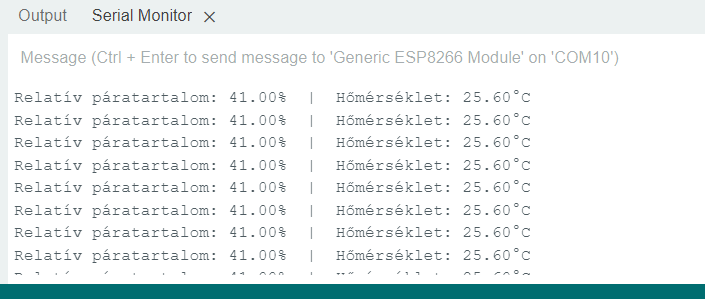
         getTem();

  delay(1000); }

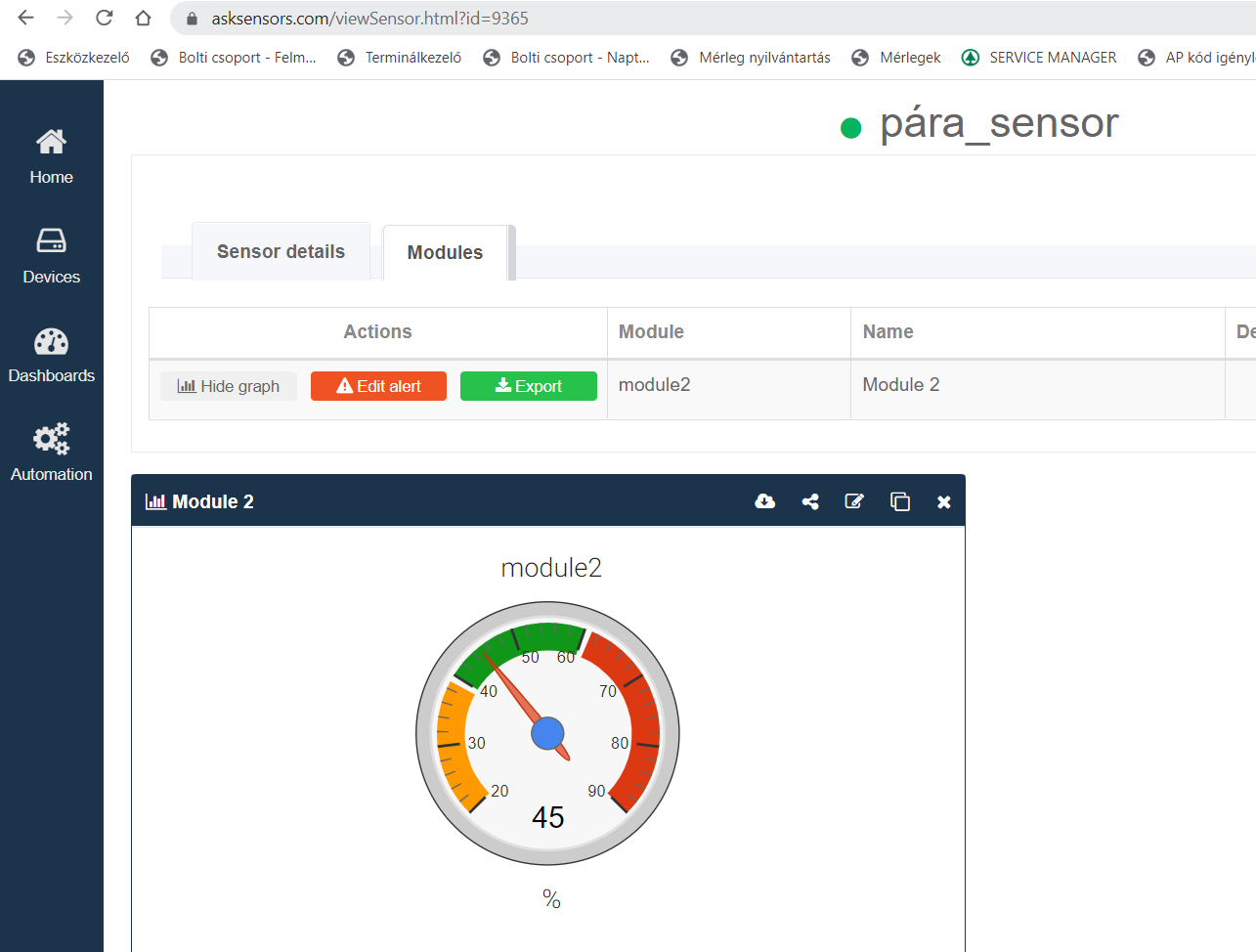
A program elkészülése után fel kell töltenünk. A Arduino IDE Tools -> Board (Generic ESP8266 Module), és Port beállítások után a Sketch -> Upload-ra kattintva.



Amennyiben a Tools -> Serial Monitor almenüt választjuk, akkor a soros monitoron máris megjelennek az általunk mért adatok:



Ezután már csak annyi dolgunk van, hogy kitöltünk egy ingyenes regisztrációs űrlapot a Asksensor.com oldalán. Itt létrehoztam egy új érzékelőt Pára-sensor néven. Beállítottam a számomra ideális értéknek megfelelő 40-60%-os páratartalmat és a szélsőértékek átlépésekor figyelmeztető e-mail küldést.



Így akár távolról is figyelemmel tudom követni az otthonunk páratartalmának változásait. Alább egy ilyen figyelmeztetés miatt küldött levélben olvasható, hogy a páratartalom 36%-ra esett vissza.

