IoT mérőállomás megvalósítása ESP-12 WiFi modul segítségével

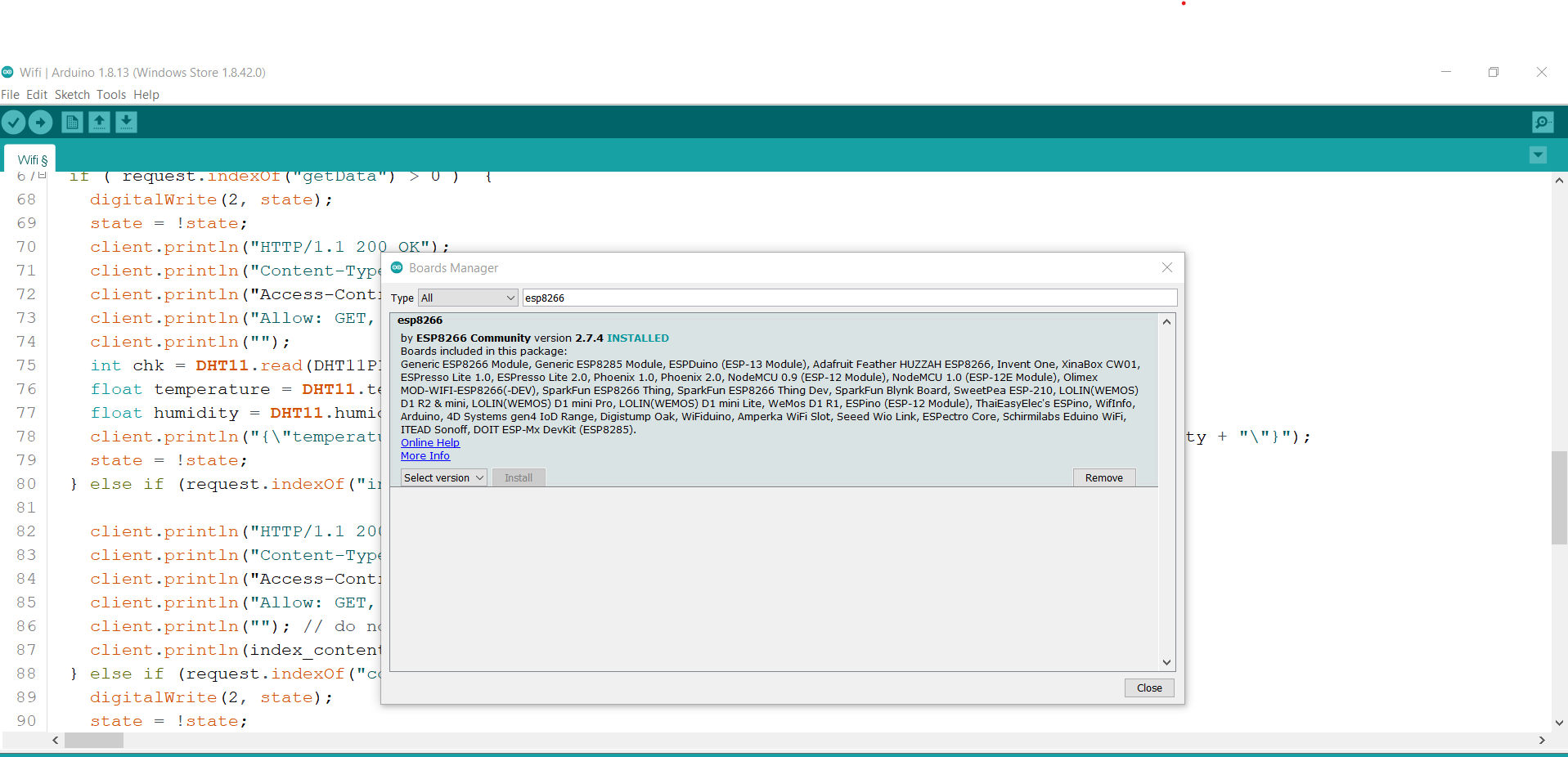
Ezt a projektet az alábbi linken megtalálható dokumentum inspirálta [1]

**Alkatrészlista:**

* ESP-12F wifi modul;
* 3,3V-os USB to UART konverter,
* DH11 hőmérséklet és páratartalom szenzor,
* próba panel,
* vezetékek,
* ellenállások: 3 db 10kOhm,
* 2 db mikrokapcsoló,
* próba panel táp.

**Arduino IDE konfigurálása:**

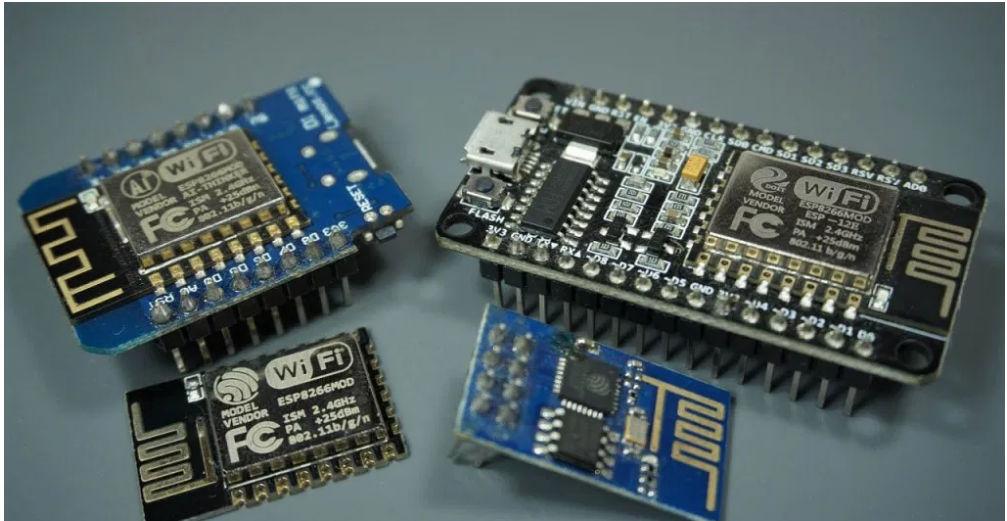
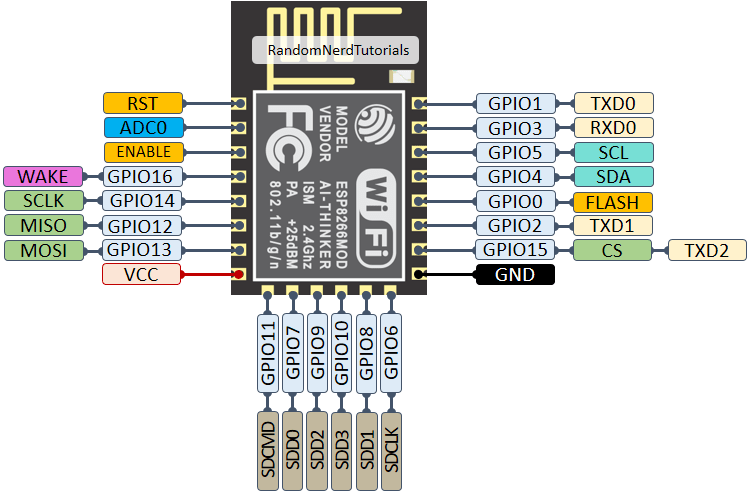
1. Arduino IDE telepítése,
2. Board manager beállítás: File-> Preferences -> Additional Board Manager URLs: <http://arduino.esp8266.com/stable/package_esp8266com_index.json> -> OK,
3. Következő lépésben kiválasztjuk az eszközt: Tools-> Board -> Board Manager,
4. Telepítsük az alábbi elemet: „esp8266 by ESP8266 Community”,



1. Eszköz kiválasztása: Board -> ESP8266 -> Node MCU 0.9(ESP-12 Module).

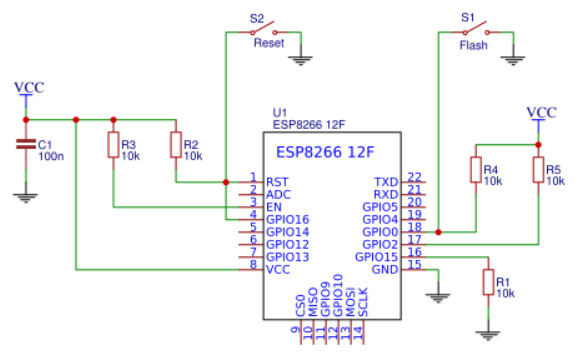
**ESP-12 Programozása:**

ESP-12 modul kicsi, de annál több lehetőséget hordoz magában. Az eszközt az AI-Thinker[2]. kínai vállalat fejleszti, sajnos korlátozott a hozzá elérhető angol nyelvű leírás[3]. Kapható hozzá adapter, mely segítéségével könnyebben lehet csatlakozni az eszközhöz, de készen is lehet vásárolni, de magunk is készíthetünk hozzá adaptert.



Mivel az eszköz nem tartalmaz USB to UART, ezért szükségünk lesz valamilyen eszközre, amely képes erre (az én esetembe: USB 2.0 UART Module Serial Converter[4]).

Az eszköznek két módja van, amelyek futtatásához az eszköz kivezetéseit a megfelelő feszültségszintre kell kapcsolni:



1. UART (programozáshoz):

* GPIO0 – LOW,
* GPIO2 – HIGH,
* GPIO15 – LOW,
* VCC – 3,3V,
* GND - GND,
* CH\_PD – HIGH.

1. Flash boot (rajta lévő program futtatása):

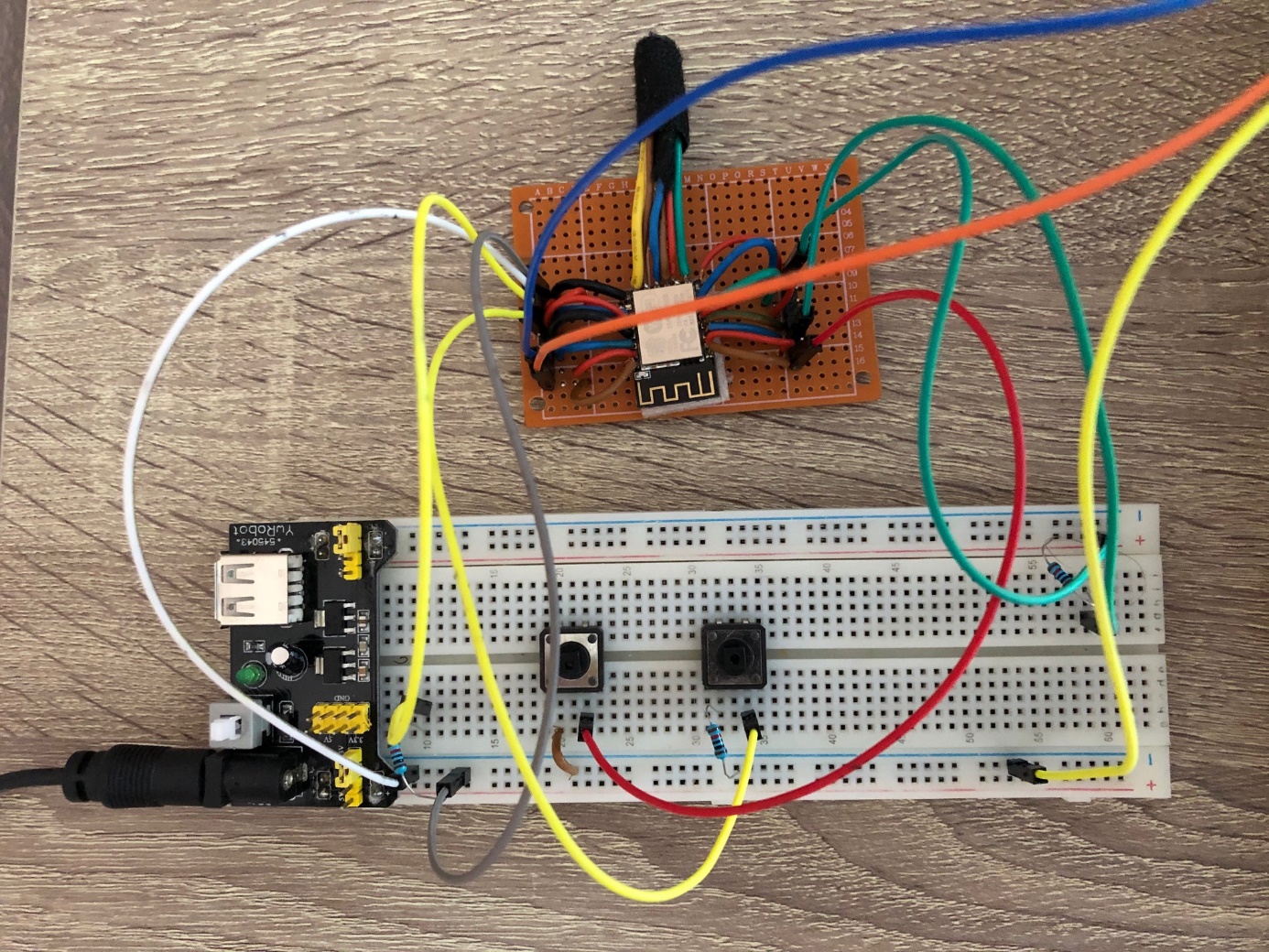
* GPIO2 – HIGH,
* GPIO15 – LOW,
* VCC – 3,3V,
* GND - GND,
* CH\_PD – HIGH.

**FONTOS!** A VCC és GND kivezetések kivételével, a többi pinhez egy-egy 10kOhm-os pull-up vagy pull-down ellenállást használjunk.

Az eszköz egyszerű programozásához az alábbi áramkört hozzuk létre:

Bekötés:

* GPIO0 – Flash gomb,
* GPIO2 – HIGH,
* GPIO15 – LOW,
* VCC – 3,3V,
* GND - GND,
* CH\_PD – HIGH,
* Reset – Reset gomb,
* TX – RX,
* RX – TX.



Próbapanel táp

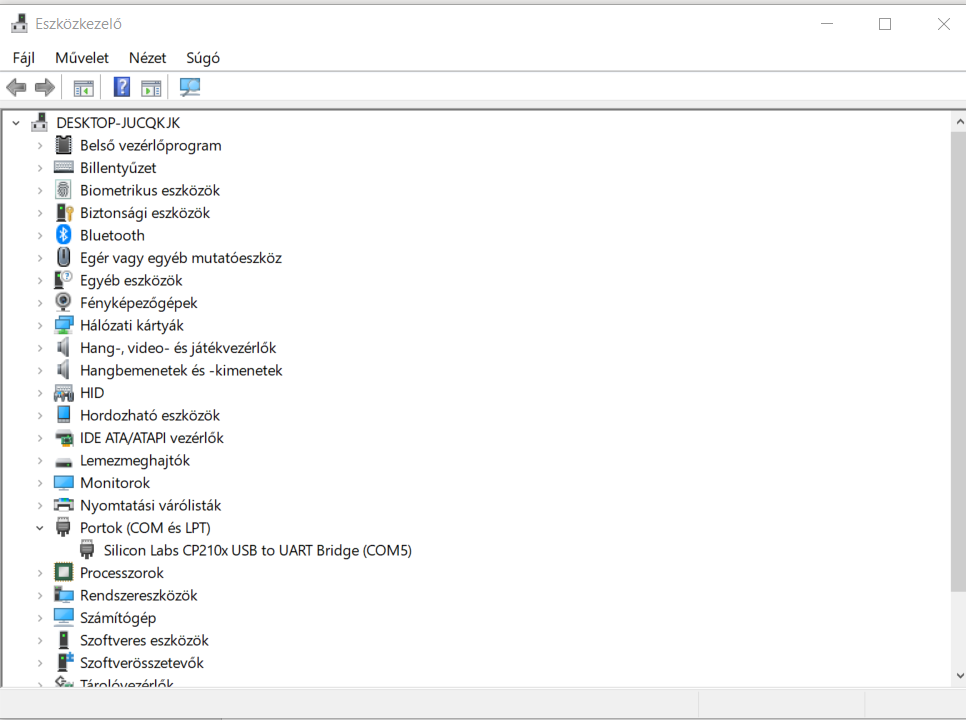
ESP-12F

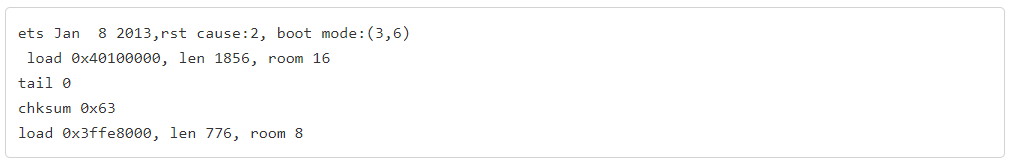
Reset gomb

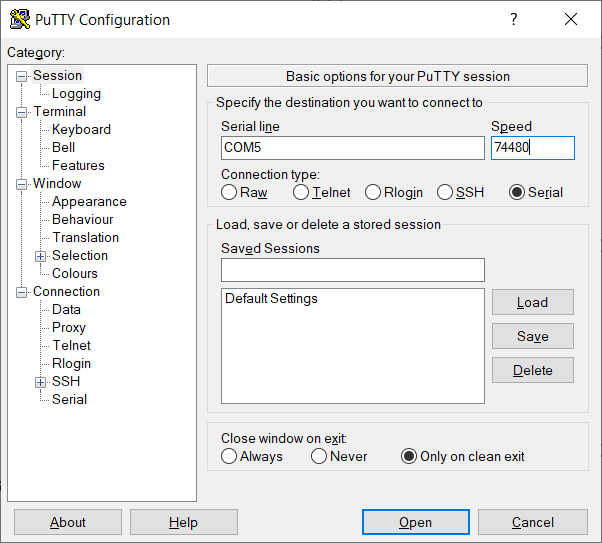
Flash gomb

**FONTOS!** Első bekapcsolás:

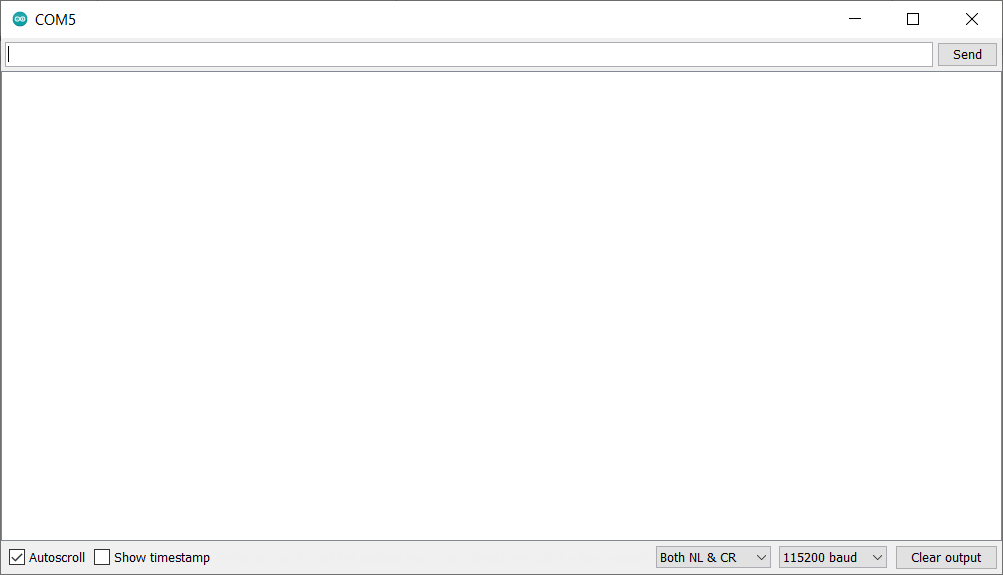
1. Csatlakoztassuk az eszközt,
2. Ellenőrizzük, hogy melyik COM porton keresztül csatlakozik: Eszközkezelő -> Portok

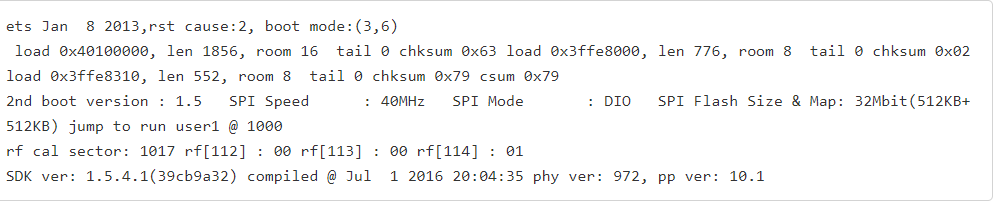


1. Indítsuk el a putty[6] alkalmazást, csatlakoztassuk COM portot 74880-as baud rate-el,
2. Kapcsoljuk be az eszközt, valami hasonlót kell látnunk:  
   
3. Állítsuk át a baud rate-t 115200-ra, majd indítsuk újra az eszközt, lépjünk ki a programból,



1. Indítsuk el az Arduino IDE-t, majd Tools -> Port -> állítsuk be a megfelelő portot,
2. Nyissuk meg a Serial portot, 74480-as baud rate-el és a jobb alsó sarokban állítsuk be (ismét valami „szemét” üzenetet kapunk,





1. Állítsuk át a baud rate-t 115200-ra

Jelen pillanatban az eszközön lévő firmware-hez férünk hozzá, itt[7] található részletesebb leírása.

**Eszköz programozása**

Eszköz írásához flash módba kell kapcsolni az eszközt:

1. flash mikrokapcsolót lenyomjuk és nyomva tartjuk,
2. majd lenyomjuk a reset kapcsolót,
3. reset kapcsolót felengedjük,
4. flash kapcsolót felengedjük,
5. az eszköz flash módban van.
6. nyissuk meg az arduino IDE-t, majd a kiválasztott kódot töltsük fel az eszközre,
7. ha végzett a feltöltéssel, nyomjuk le a reset gombot,
8. az eszköz kész a futtatásra.

**Projekt**

DHT 11 szenzor adatait veszi az eszköz és ehhez ad hozzáférést az interneten keresztül.

Szenzor bekötése:

* DHT 11 szenzor 3 kimenettel rendelkezik (GND, OUT, VCC),
  + GND -> GND,
  + VCC -> 3,3V,
  + OUT -> GPIO5.
* Az arduino IDE-be töltsük le a a DHT11-es libaryt, amelynek segítségével könnyen hozzáférünk az eszköz adataihoz

**ESP-12 program:**

A mellékelt fájl tartalmazza a programot, amelyet szükséges feltölteni az eszközre.

Funkciót:

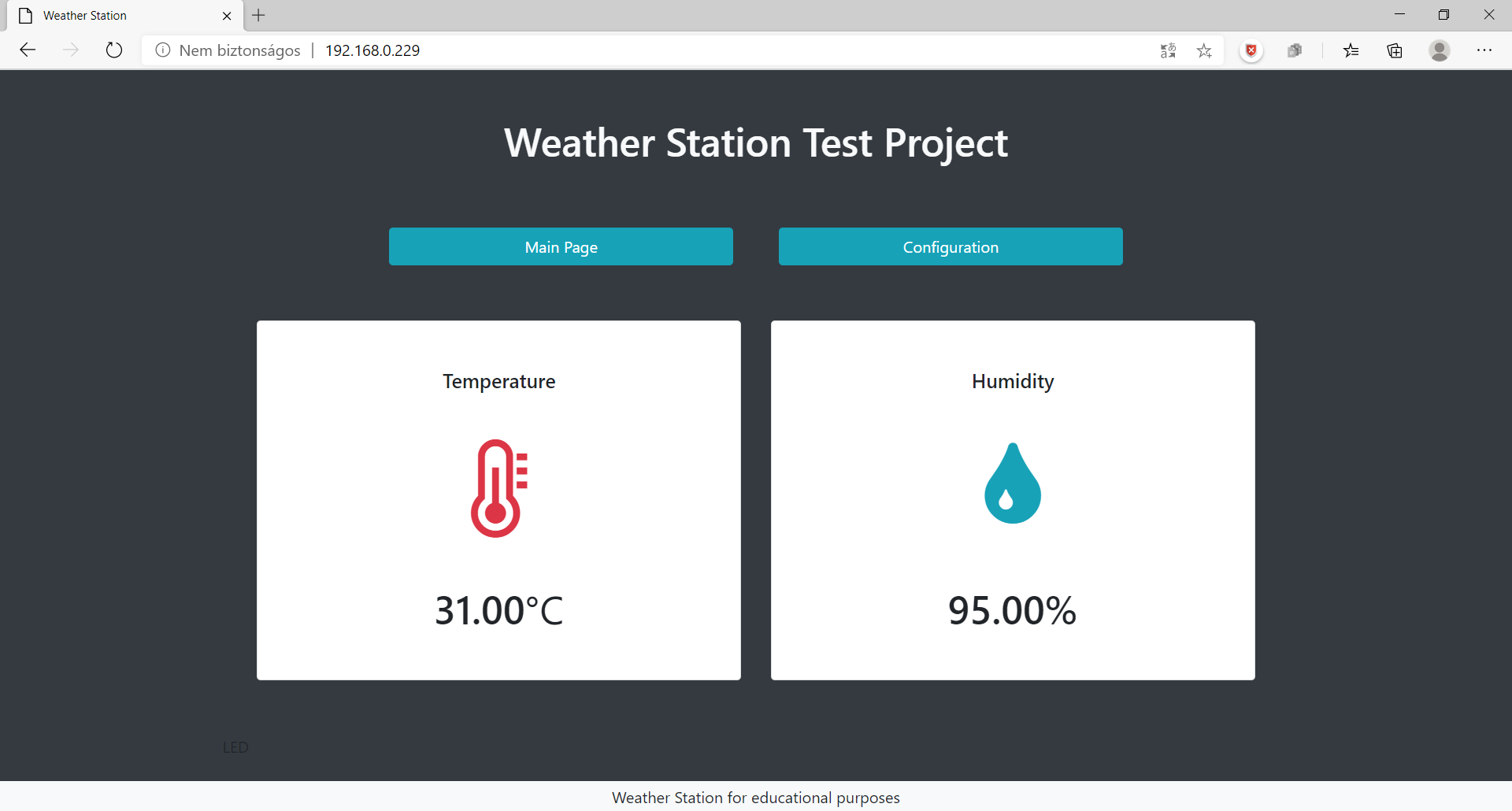
* DHT11 szenzor adatainak olvasása,
* ESP-12F Wifi hálózatra csatlakoztatása,
* Kérések kiszolgálása:
  + index.html betöltése,
  + adatok küldése getData kérésre (Ajax request),
  + főoldal és a beállítások közti váltás (Ajax request), tartalom cseréje (index,config kérésre).

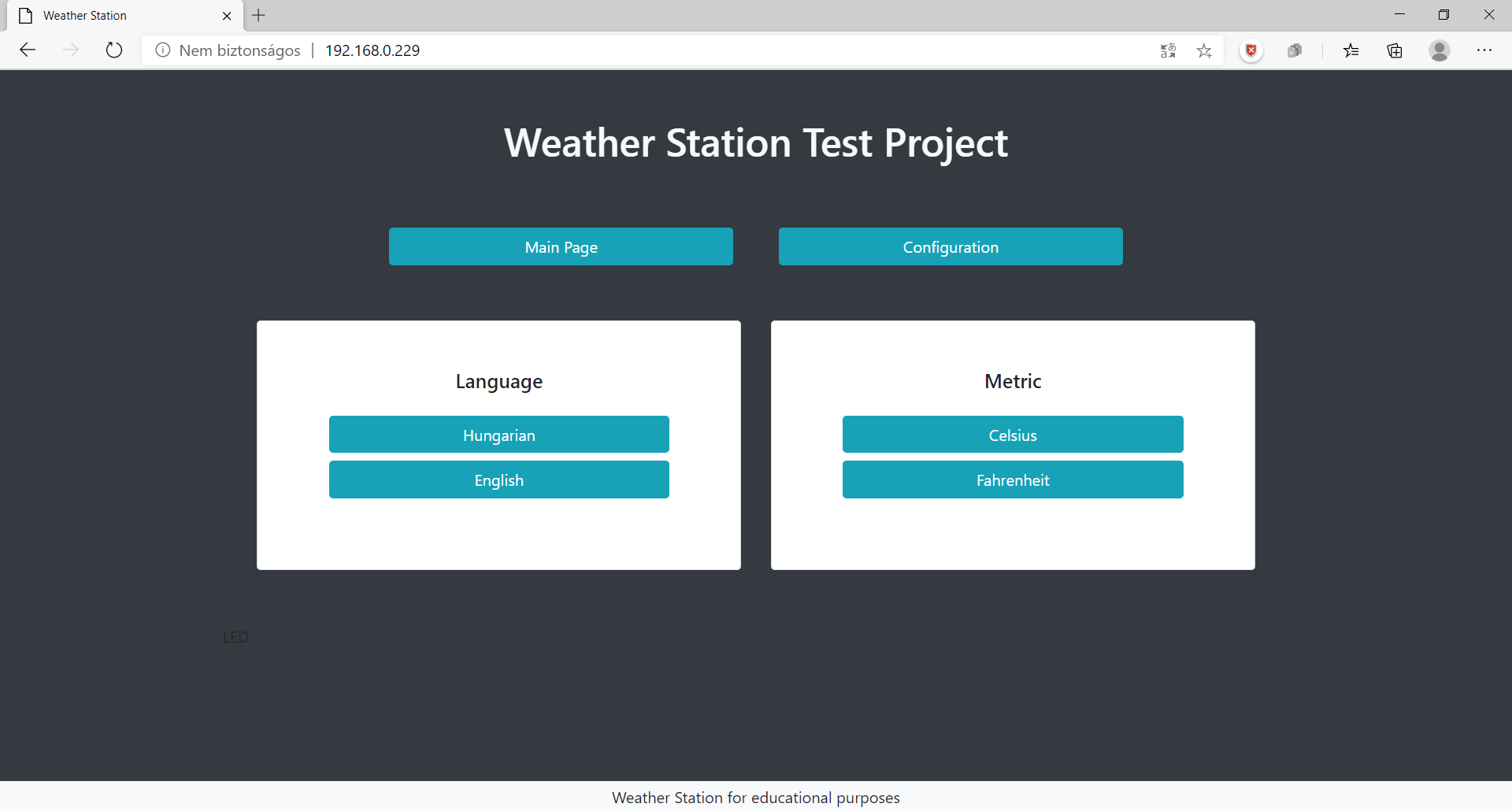
**GUI**

GUI elkészítéséhez HTML, CSS (külső könyvtár: Bootsrap, FontAwesome), Js használunk

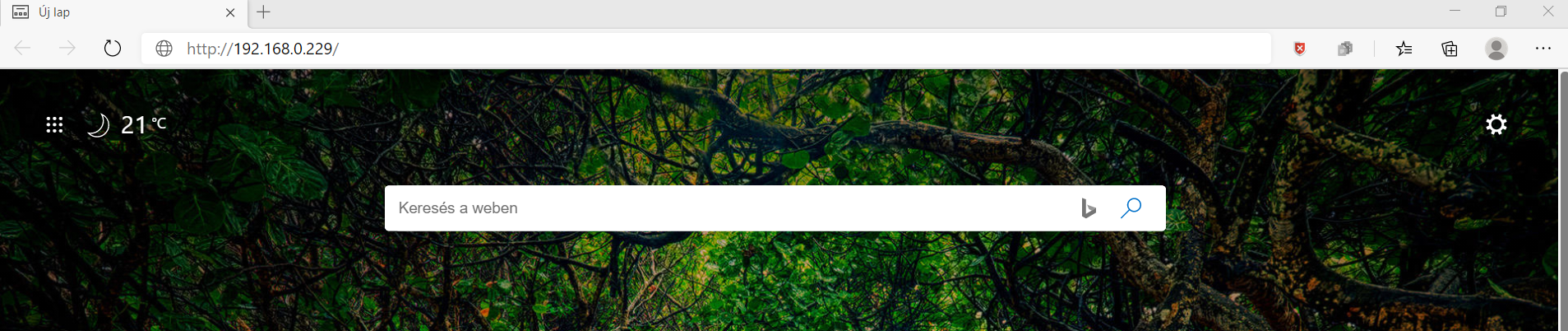
Két fület találunk:

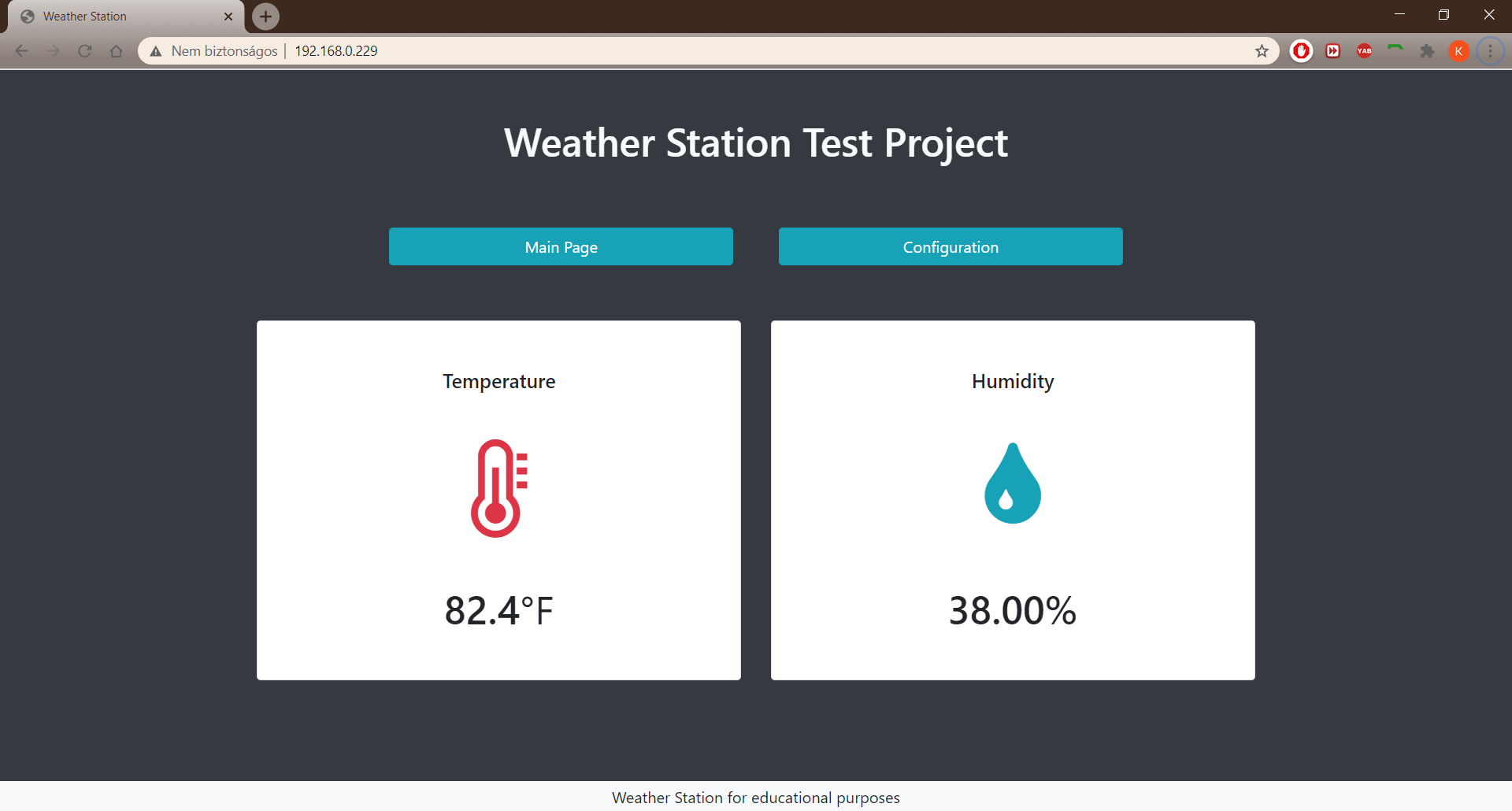
1. szenzor adatait olvashatjuk és beléphetünk a beállításokba,
2. beállíthatjuk a kívánt nyelvet, illetve a mértékegységet.





**Csatlakozás az eszközre:**

1. ESP-12F indításakor a Serial monitoron kiírja, a részére kiosztott IP címet (pl.: <http://192.168.0.229/> ),
2. Böngészőbe gépeljük be az adott IP címet
3. Már csatlakoztunk is az eszközhöz

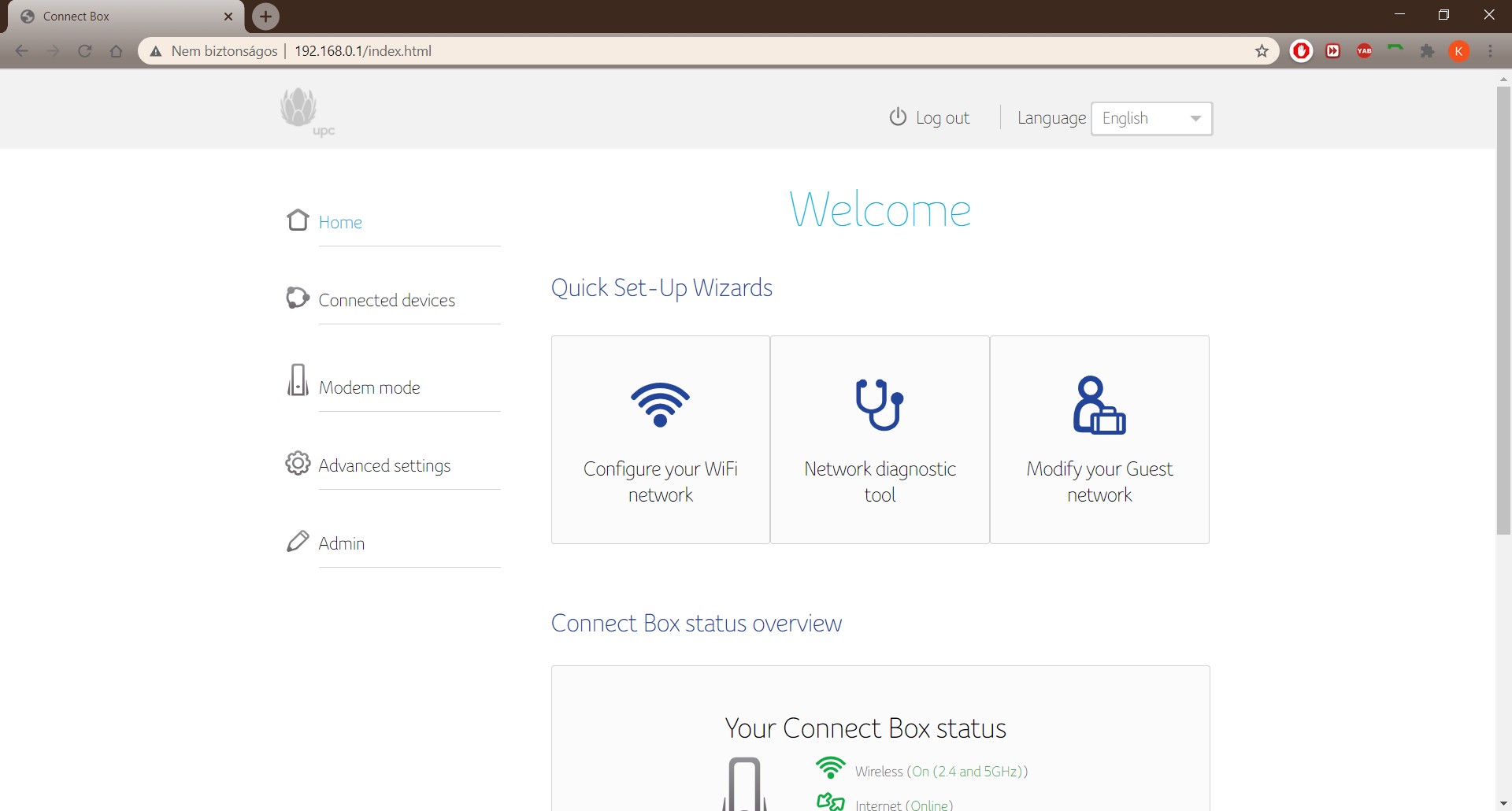


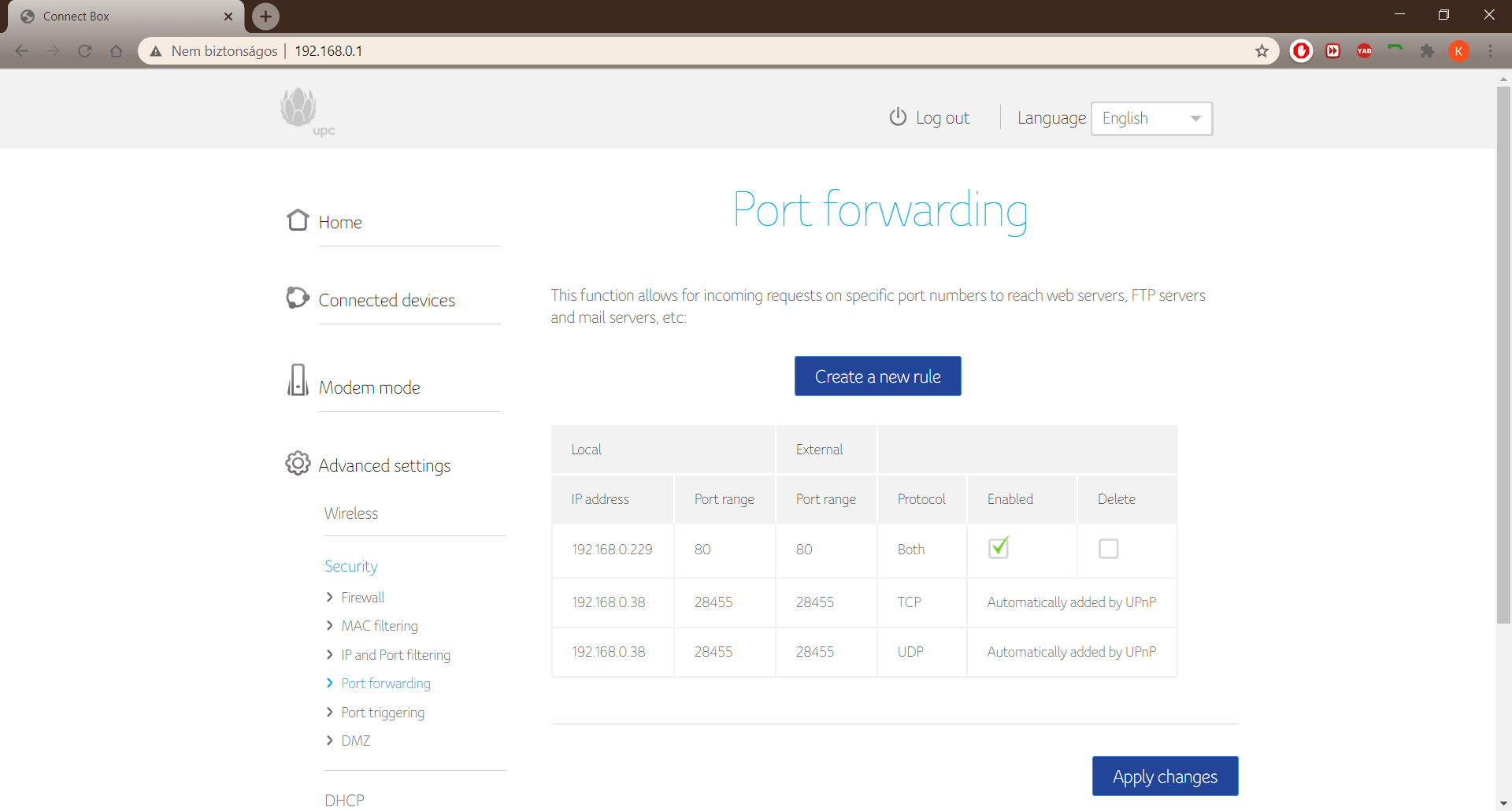
Ezen pillanatban csak akkor tudunk csatlakozni az eszközhöz, ha egy Acces Pointra csatlakozunk. Amennyiben interneten keresztül is el szeretnénk érni az adott eszköz, routeren szükséges a pont forwarding-ot beállítani

**Port Forwarding**

Különböző eszközökön a kezelőfelület más és más lehet, itt egy digi routeren végzem a bemutatást.

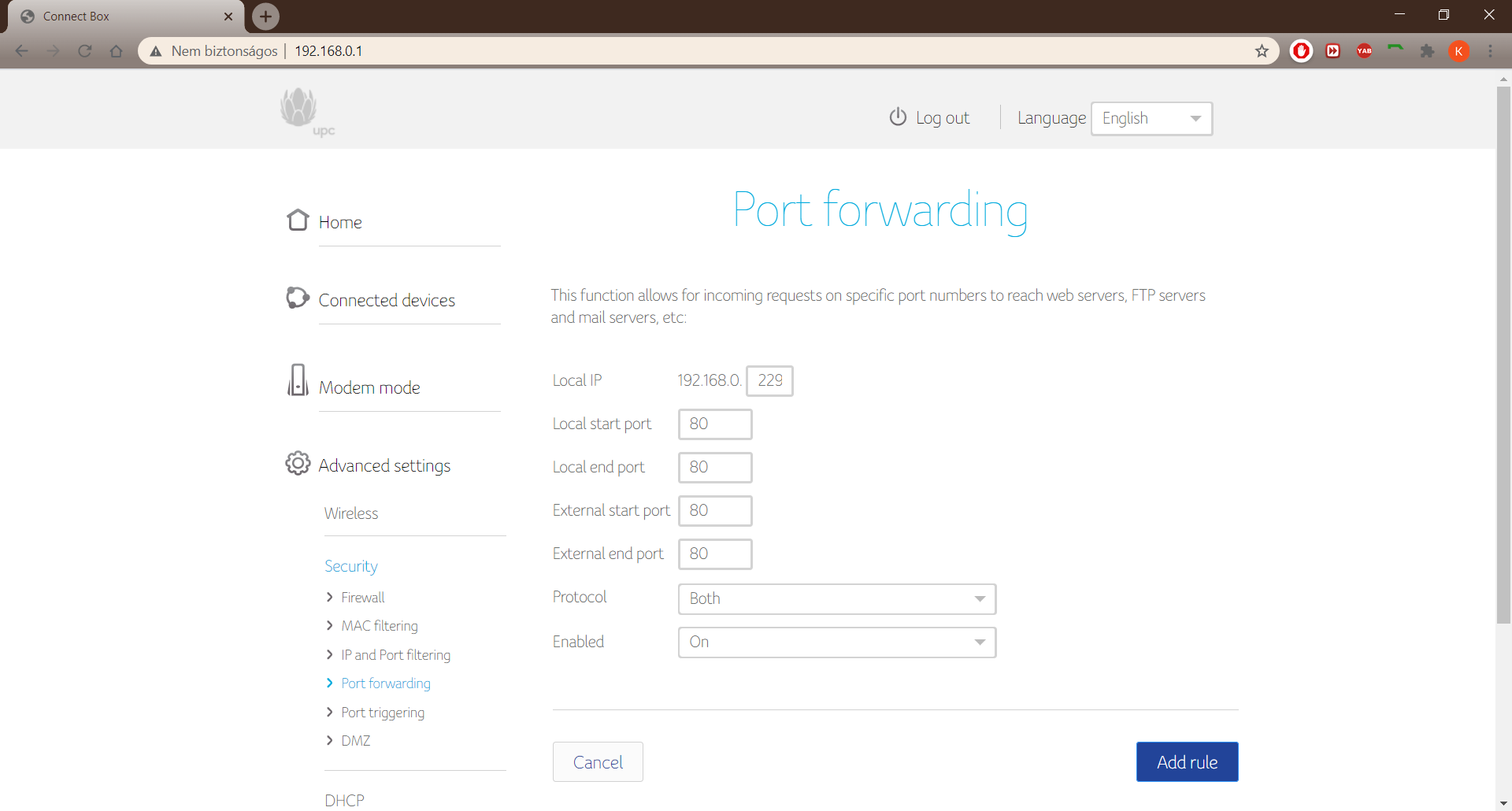
1. Router megnyitása
2. Advanced settings -> Security -> Port forwarding -> Create new rule





1. Port forwarding beállítás:

* Local IP: Az eszköz IP címe,
* Local start port: állítsuk be az eszközre konfigurált port számot,
* Local end port: állítsuk be az eszközre konfigurált port számot,
* External start port: melyik porton keresztül érjük el a routert kívülről,
* External end port : melyik porton keresztül érjük el a routert kívülről,
* Protocol: TCP+UDP,
* Enable: On.



**Források:**

[1] <http://www.inf.u-szeged.hu/miszak/wp-content/uploads/2019/01/Arduinowifi.pdf>

[2] <http://en.ai-thinker.com/>

[3] <https://www.elecrow.com/download/ESP-12F.pdf>

[4] <https://www.ebay.com/itm/WeMos-D1-CH340-ESP8266-ESP-12E-CP2102-WiFi-Development-Board-Shield-Adapter-New/202004310865?hash=item2f08652b51:g:HLsAAOSw4IlZeBNy>

[5] <https://www.ebay.com/itm/USB-2-0-to-TTL-UART-Module-Serial-Converter-CP2102-STC-PRGMR-Free-cable-UK/192586527957?ssPageName=STRK%3AMEBIDX%3AIT&_trksid=p2057872.m2749.l2649>

[6] <https://www.putty.org/>

[7] <https://www.espressif.com/sites/default/files/documentation/4a-esp8266_at_instruction_set_en.pdf>

[8] <https://tomeko.net/online_tools/cpp_text_escape.php?lang=en>

[9] <https://www.instructables.com/id/ESP-12F-ESP8266-Module-Connection-Test/>

[10] <https://cameradepo.hu/termektamogatas/port-forwarding-a-gyakorlatban>

[11] <https://www.instructables.com/id/Intro-Esp-8266-firmware-update/>

[12] <https://babaawesam.com/2015/11/26/control-input-output-of-esp8266-with-ajax/>