

## Az időjárás állomás fő funkciói:



- **Külső hőmérséklet kijelzése**  
Színátmenetes betűszín meleg piros hideg kék
- **Külső páratartalom kijelzése**
- **Légnyomás kijelzése**
- **Szélerősség kijelzése**  
Színátmenetes betűszín erős piros gyenge szürke
- **Ég-kép kijelzése (20 féle)**
  1. derült nappal
  2. derült éjjel
  3. gyengén felhős nappal
  4. gyengén felhős éjjel
  5. mérsékelten felhős nappal
  6. mérsékelten felhős este
  7. erősen felhős nappal
  8. erősen felhős éjjel
  9. borult
  10. gyenge eső
  11. eső
  12. zivatar
  13. száraz zivatar
  14. ónos eső
  15. havaseső
  16. havazás
  17. köd/pára
  18. zápor nappal
  19. zápor este
  20. erős szél
  21. (jégeső)

- **Aktuális időjárás szöveges megjelenítése (23 féle)**

1. derült
2. gyengén felhős
3. közepesen felhős
4. erősen felhős
5. borult
6. viharos szél
7. pára
8. köd
9. sűrű köd
10. havaseső
11. hózápor
12. erős hózápor(intenzív)
13. havazás
14. erős havazás(intenzív)
15. hószállingózás
16. zápor
17. ónos eső
18. felhőszakadás
19. erős zivatar
20. zivatar
21. szitálás
22. gyenge eső
23. száraz zivatar

- **Belső hőmérséklet páratartalom kijelzése**

Színátmenetes betűszín meleg piros hideg kék

- **Pontos idő kijelzése**
- **Dátum és a hét napjának kijelzése**
- **3óránkénti előrejelzés, órák kiírásával, éghéppel, hőmérséklettel választható 2.5 napos előrejelzés is, ennek módja a kódban le van írva**
- **Napfelkelte/lemente idők kijelzése**
- **Holdfelkelte/lemente idők kijelzése**
- **Holdciklusnak megfelelő kép megjelenítése (26 lépésben)**

- **Adatok cseréje Blynk telefonos alkalmazással**
  - Hőmérőtől Blynk-nek
  - Ég-kép (V0)
  - Külső hő és pára (V2, V4)
  - Belső hő és pára (V1, V3)
  - Légnyomás(V6)
  - Szél(V7)
  - Teszt (V8) *(csak teszteléskor használt)*
  - Hőérzet(V9)
  - Harmatpont(V10)
- **Blynk-től hőmérőnek**
  - Üzemmódválasztó(V5) Fényerő:  
gyenge/automata/erős/

### Egyéb funkciók:

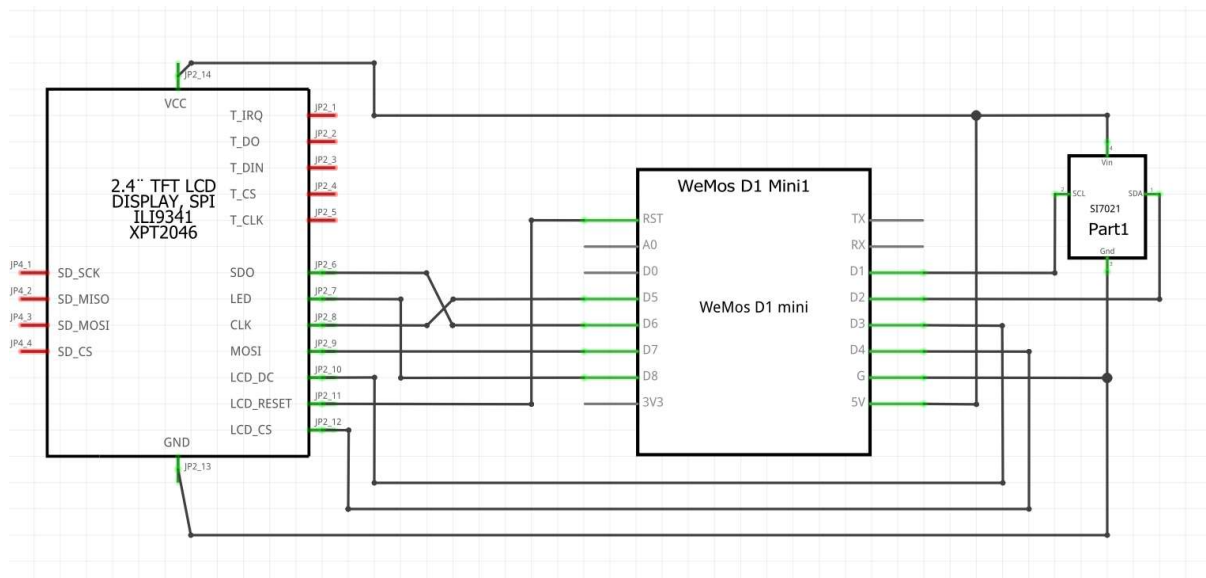
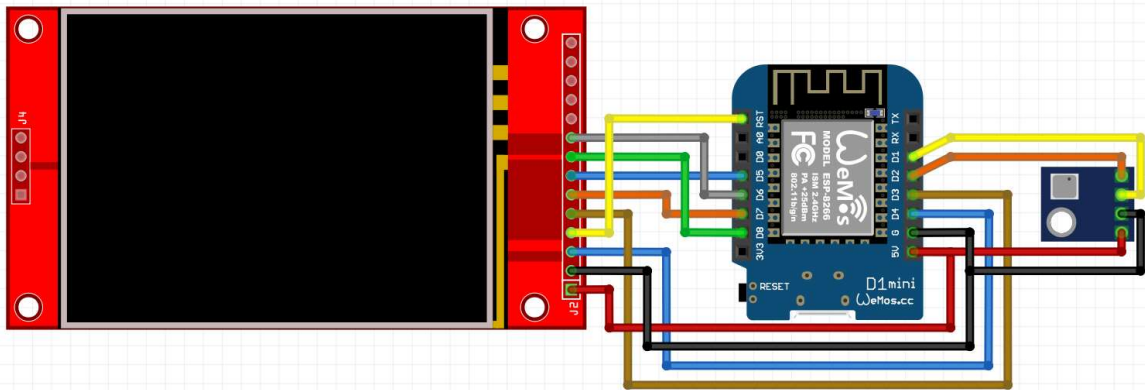
- **Fényerő automata módban a napfelkeltéhez, lementéhez alkalmazkodik**
- **Pontos Dátum Idő automatikusan beállításra kerül az adatokat ~30mp-ként szinkronizálja**

# Megvalósításhoz szükséges dolgok:

## 1. Hardver elemek

- Wemos D1 mini Pro (Alliexpress áratat írok: ~1000ft)
- ILI9341 TFT 2.4, vagy esetemben 3.2" Kijelző ~3000ft
- SHT21 hő és páratartalom mérő ~500-800ft (kék nyáklapos verzióval tesztelve)
- Breadboard+Jumperek vagy NYÁK (nyákterv még nincs de bármilyen raszteres nyákra könnyen össze lehet forrasztgatni) ~500-2000ft
- raszteres osztású tűkesor+hüvelysor ~250ft
- 5V min 1A DC táp microUSB csatlakozóval, pl telefontöltő ~1000ft

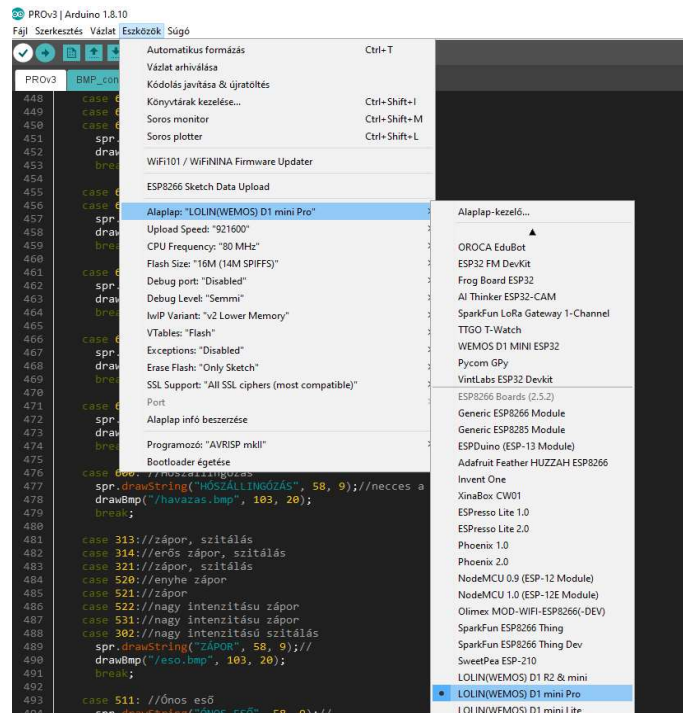
## 2. A kapcsolás



# Arduino IDE(fejlesztő szoftver) + ESP telepítése

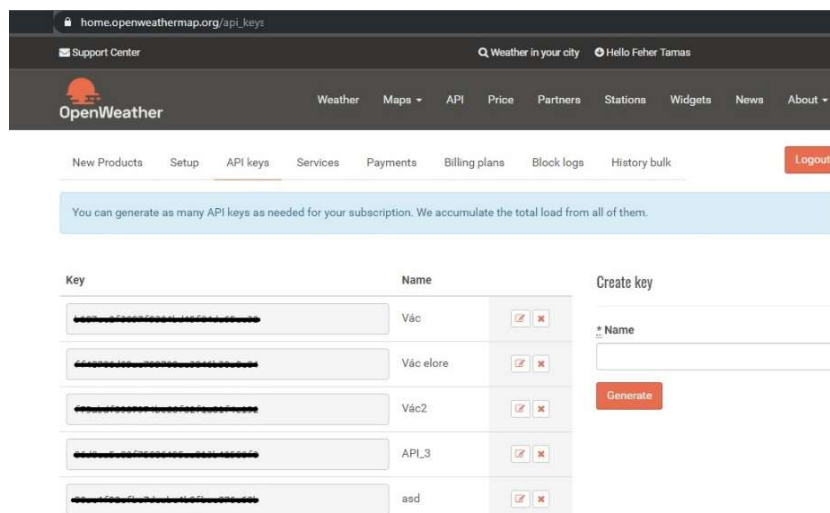
- Magyar részletes leírás a menetről <http://www.tavir.hu/iot-3-esp8266-standalone>

Ha telepítettünk mindent dugjuk rá a boardot a gépre



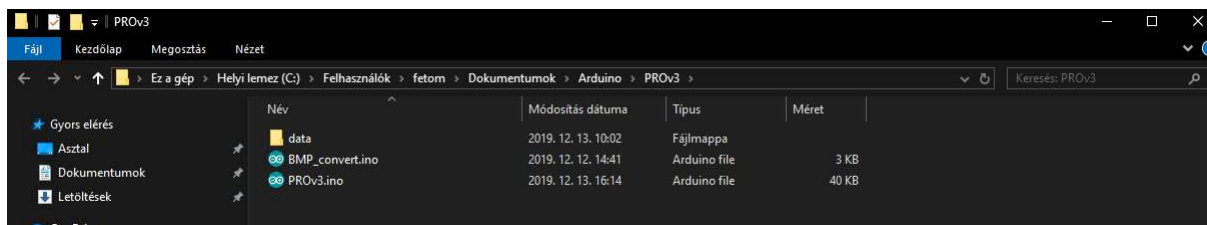
## 3. Openweather fiók

- Openweather accountot kell csinálnunk, hogy kérhessünk API-kulcsot.
- Kattints a linkre <https://openweathermap.org>
- Majd a Sign up-ra ha megadtad az adatokat
- Jelentkezz be
- A kezdő oldalon az API keys-re kell kattintani
- Névadás után generálni kell egy kulcsot
- Ezt kell majd bemásolni a megfelelő helyre (lejjebb írok róla).



## 4. Képek, betűtípusok feltöltése

- Telepíteni kell egy ESP8266 fájlrendszer feltöltőt. Én ezzel sokat szenvedtem mert valamiért mindig hibát dobott telepítésnél, ezért a Zipbe berakom azt a verziót, ami nálam jól működik Itt van az angolnyelvű leírás a „telepítésről” nem nagy kunszt. [https://www.instructables.com/id/Using-ESP8266-SPIFFS/?fbclid=IwAR373TR\\_1aGOkcLnrmG6gogZlrpp1ncJothRHfByMp1k8DWHhAcGWT4pRc](https://www.instructables.com/id/Using-ESP8266-SPIFFS/?fbclid=IwAR373TR_1aGOkcLnrmG6gogZlrpp1ncJothRHfByMp1k8DWHhAcGWT4pRc)
- Python 3 telepítése is szükséges ezt is csatoltam, elég kényes a programverzióra így ha jól akarsz akkor ezt használd.
- Fontos, hogy a képfájlok kiterjesztése csak .bmp lehet, és a nevük ne tartalmazzon ékezetes karaktereket és lehetőleg space-t sem.
- A fájlok neve módosítható, de azt akkor a kódban is át kell írni
- A mappa struktúrát is fontos tartani különben nem fog betöltődni a kép vagy a betűkészlet.
- A képek (képpont)mérete szintén kulcsfontosságú én arra törekedtem, hogy a lehető legnagyobb legyen minden. Így lettek a nagy ikonok 114x104, a hold ikonok 64x64 és az előrejelzés ikonok 53x48 méretűek. Ezeket igény szerint módosíthatod, de ezt akkor le kell kezelni a kódban is.
- A betűtípusokat és a hold képeket berakom a Zipbe, de sajnos a többi képet nem adhatom oda szerzői jogok miatt. Egyébként is, kis keresgéléssel lehet találsz olyan ikonkészletet, ami jobban teszik. Kis átméretezéssel képszerkesztéssel bárki megoldja ezt a dolgot 😊 Ez egy jó forrás lehet ingyenes vektoros ikonok beszerzésére: <https://www.flaticon.com/>
- A következő képeken látható, hogy hogyan néz ki az egész.  
(Tipp: Ha az arduinoban meg van nyitva a projekt akkor Ctrl+K billentyűkombinációval egyből megnyílik a projekt mappa)





## 5. Betűtípusok létrehozása

Ezt a részt kihagyhatod, ha megfelel az általam használt típus.

- Ha módosítanád akkor szükséged lesz a Processing 3 nevű ingyenes alkalmazásra, ebbe van építve a nekünk szükséges font generáló eszköz
- Nincs más dolgunk, mint elindítani és rákattintani a Tools/Create font-ra
- Ki kell választani a nekünk tetsző típust és méretet
- Majd a characters-re kattintani
- Utána a Specific Unicode Blockra
- És be kell pipálni az első hármat Basic Latin, Latin-1, Latin extended-A-t ezek kellene ahhoz, hogy a magyar karakterekkel dolgozhassunk.
- Nyomunk 2 ok-ot és ha vissza kerültünk az üres projekthez akkor nyomunk egy Ctrl+K-t ez megnyitja a betűkészletet tartalmazó mappát, amit onnan már átmásolhatunk az ESP-re.
- Fontos, hogy a létrehozott fájl nevét át kell vezetnünk a kódba is a #define FONT\_X -nál

## 6. Könyvtárak

Egy Zip-be berakom mindet. (nem érdemes letölteni őket a netről, mert van, ami módosított, valamint nem garantálható, hogy ha később frissítik valamelyiket akkor az nem okoz majd gondot)

- `#include <ESPHTTPClient.h> //Tartalmazza az arduino_626543 mappa (Gyári)`
- `#include "SunMoonCalc.h" // Tartalmazza az arduino_626543 mappa (Gyári)`
- `#include <time.h> // Tartalmazza Timemaster mappa (Gyári)`
- `#include <Arduino.h> //Gyári`
- `#include <ESP8266WiFi.h> //Ez települ automatikusan.`
- `#include <JsonListener.h> //Gyári`
- `#include <OpenWeatherMapCurrent.h> // Tartalmazza az arduino_626543 mappa módosítva van.`
- `#include <OpenWeatherMapForecast.h> // Tartalmazza a arduino_626543 mappa módosítva van.`
- `#include <BlynkSimpleEsp8266.h> //Gyári`
- `#include <Wire.h> //Gyári`
- `#include "SHT21.h" //Gyári de érdemes a csatoltat használni mert ebből is van több verzió.`
- `#include <UnixEpochToDate.h> //Saját módosított a zipben van.`
- `#include <FS.h> // Gyári`
- `#include <TFT_eSPI.h> // Gyári (User_Setup.h-t be kell állítani)`
- `#include <SPI.h> //Gyári`

Ezek közül mindet a C:/Felhasználók/"Neved"/Dokumentumok/Arduino/ libraries mappába kell másolni. A gyári azt jelenti, hogy nem nyúltam bele letölthető és frissíthető is.

A TFT\_eSPI könyvtárat be kell állítani az adott bekötésnek és kijelző típusnak megfelelően ezt a lib példái közt megtalálod. De abban, amit mellékeltem módosítva van. User\_Setup.h (a jelenlegi hardvernek és bekötésnek megfelelően)



## 7. A szoftver személyre szabása

Először is javaslom az Arduinon belül a Fájl/Beállítások/Sor számok megjelenítése négyzet bepipálását így tudok a sorszámmra hivatkozni.

- Blynk egy alkalmazás, amivel adatokat cserélhetünk a telefonunk és a mikrovezérlő között a 8. pontban írok róla. A **10.** sorba kell bemásolni a Blynk által adott API-key-t ilyesmit kell majd látnunk:

```
char auth[] = "123a4567b8c1234d5678e91f234ghij5";
```

- A WiFi beállítások **35, 36, 37** sor. Ide kell beírni annak a wifihálózatnak az adatait, amelyet használni fogunk.

```
#define WIFI_SSID "TP-Link_WiFi"  
#define WIFI_PASS "*****"  
#define WIFI_HOSTNAME "Weather-station" //Ide bármit írhatsz ez lesz a hálózaton  
látható neve az állomásnak.
```

- **38.** sor ide kell bemásolni az Openweathermap által adott API-key-t ilyesmit kell majd látnunk:

```
String OPEN_WEATHER_MAP_APP_ID="a123bc2d1234e1234fg12h12ij12kl12";
```

- **40.** sor ide a location ID számot kell bemásolnunk ez fogja megadni, hogy melyik város adatait szeretnénk látni. Ezt a számot az openweathermap-ról tudjuk meg. Ehhez a kezdőoldalon a lap tetején, középen a Weather in your city keresőmezőbe kell beírni a városunkat, majd entert nyomva a találatok közül ki kell választani a miénket, ha erre rákattintunk a böngészőnk URL-sávjában valami ilyesmit kell látnunk: <https://openweathermap.org/city/3043293> ebből nekünk a végén lévő számsor kell.

```
String OPEN_WEATHER_MAP_LOCATION_ID = "3043293"; //Vác
```

- A **41.** sorba a városunk nevét kell beírni ékezet nélkül.

```
String DISPLAYED_CITY_NAME = "Vac"; (elvileg nincs hatással semmire)
```

- Az **5,6.** sorban kell megadni annak a helynek a koordinátáit, aminek látni szeretnénk a nap és hold adatait

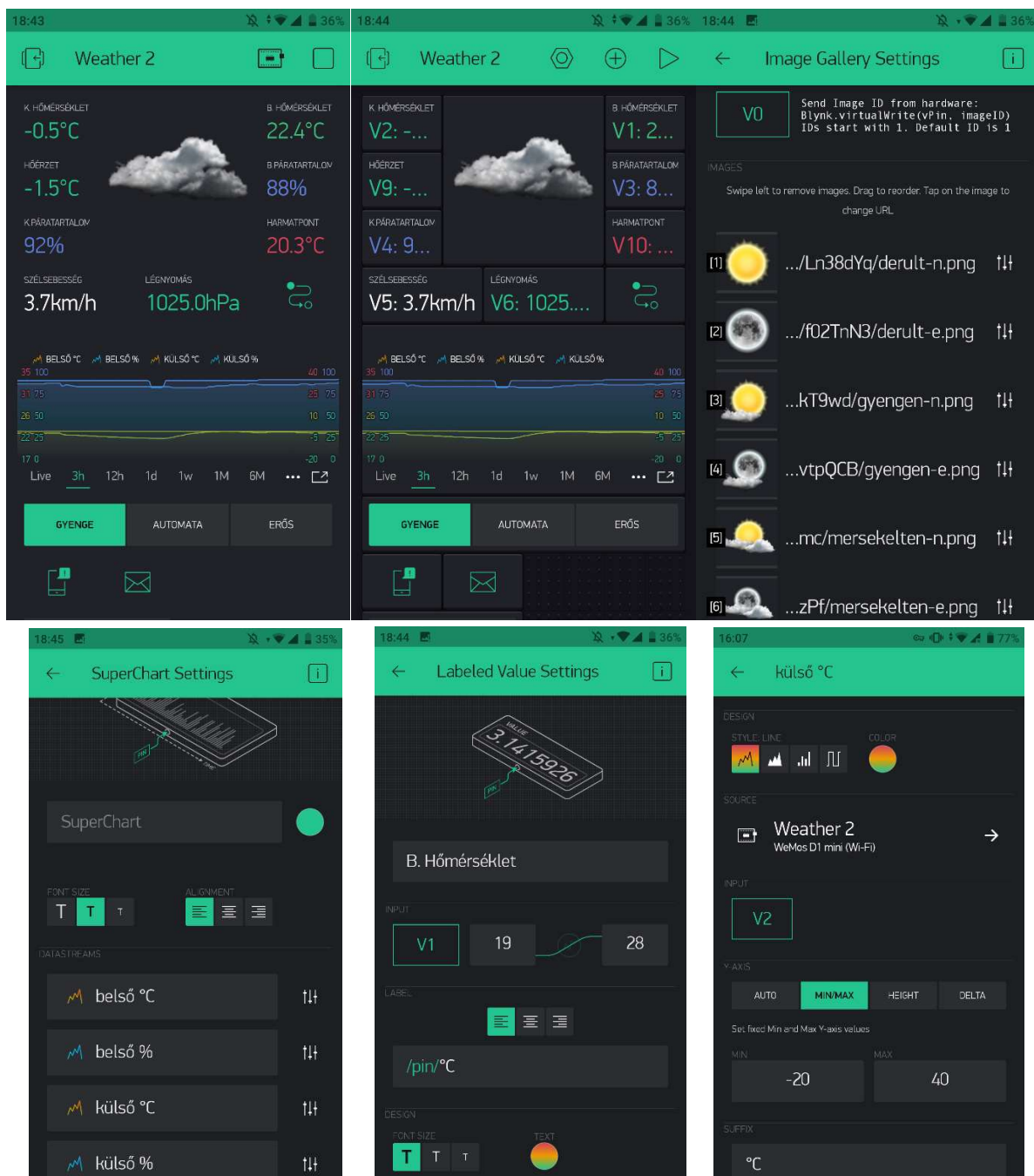
```
const float eszaki_szelesseg = 47.000000;
```

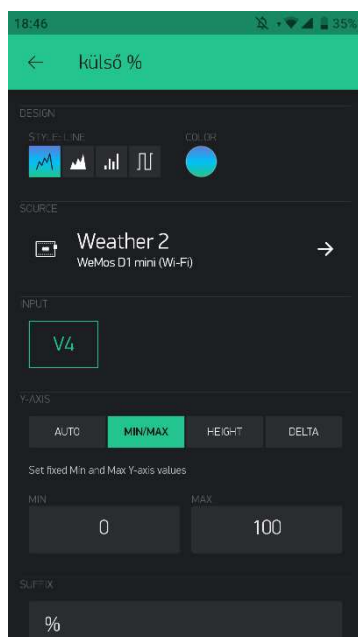
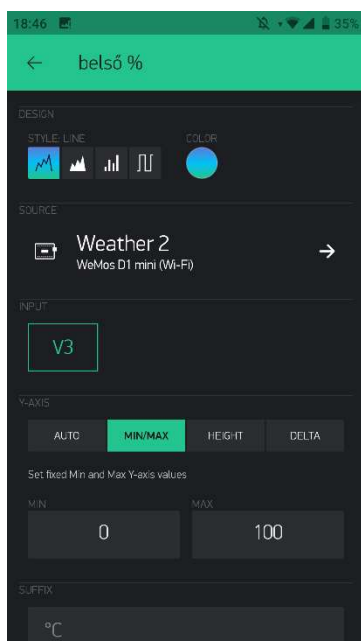
```
const float keleti_hosszusag = 19.000000;
```

## 2. Blynk

- Ha nem szeretnénk az adatokat telefonon megjeleníteni akkor gyakorlatilag ezt a részt át is ugorhatjuk
- Amit fontos megemlítenem, hogy a Blynk ingyenes, viszont korlátozva van a kijelzőre kipakolható gombok modulok száma, ez azt jelenti, hogy ami itt lentebb látható, grafikonos, képes, gombos, elrendezés, **ingyen nem teljesen valósítható meg**. Itt kompromisszumot kell kötni, vagy nem rakunk be ennyi csilivili grafikont, képet, vagy fizetünk egy keveset a Blynk fejlesztőinek (az egyébként remek munkájukért) én utóbbit választottam.

Az alkalmazás telepítésekor alaphoz 2000 kreditet ad ingyen. Minden modulnak van ára, például: a labeled value ahol a °C és % látható 400 kreditbe kerül a Super chart grafikon 900ba. Tehát ebből a 2000ből is össze lehet rakni használható elrendezést. De ha pontosan ugyan így szeretnéd megcsinálni ahogy én csináltam, akkor az 4400 kreditbe kerül. Tehát 2400at kell venni, ami (1399ft az összefoglaló írásakor)





### 3. Ismert hibák

- Ha nagyon lassú az internet vagy, épp egyáltalán nincs akkor fekete lesz a kijelző amíg helyre nem áll a kapcsolat. (ezt próbálom majd javítani, hogy legalább egy hiba üzenetet ki írjon.) **V9 től javítva**
- Adatok frissítésekor a Blynkről leszakad, érdemes kikapcsolni az idevonatkozó értesítőt a Blynkben.
- A nyári és téli időzóna nem áll át magától, ezt szándékosan hagytam ki, mert tudomásom szerint ez el lesz törölve. Az időzónát a **1104.** sorban tudod kiválasztani.
- 2038.01.19-től hibás lesz a dátum/idő kijelzés, ugyanis UNIX időben többet nem lehet kifejezni, ez nem az én hibám, ebből globális problémák is lesznek.
- Ez nem hiba, de figyelni kell rá: a mikrovezérlő üzemszerűen fel fog melegedni, így zárt dobozban nem szabad a hőmérő modult a közelébe rakni mert csalni fog. Érdemes legalább 10centire ki vezetékeznii.
- A belső páratartalom extrém esetben tud 100%-nál több is lenni.

## 4. Jövőbeli tervek

- Tervezek egy második oldalt, ahol grafikonon lehetne látni értékeket. Egyéb adatokat lehetne kiírni esetleg előrejelzést óránkénti felbontásban.
- Analóg A0 pin szabadon van még oda lehetne egy fényérzékelőt kötni így a fényerőt a környezetihez lehetne igazítani, mint a telefonoknál.
- Beállítások képernyőt kellene létrehozni, hogy PC-nélkül lehessen beállítani wifihálózatot és egyéb adatokat. Ez nagy munka nem biztos, hogy lesz rá időm.
- Még jó ötletnek tarok egy figyelmeztető rendszert is, amivel értesítést lehetne küldeni akár telefonra akár kijelzőre, ha pl 1-3 órán belül szélsőséges időjárási esemény következhet be (Zivatar, Erős szél, szélsőséges hőmérséklet, szélsőséges mennyiségű csapadék)
- Esetleg lehetne pár easter egget bele rejteni, pl karcsonykor karácsonfát kirakni ikonnak, egyéb ünnepek születésnapok, akár névnapok kiírása.
- Ha van jó ötleted vagy kész megoldásod szívesen fogadom.

Készítette: Fehér Tamás 2020.03.30.