

DSTIKE DEAUTHER WATCH CLONE

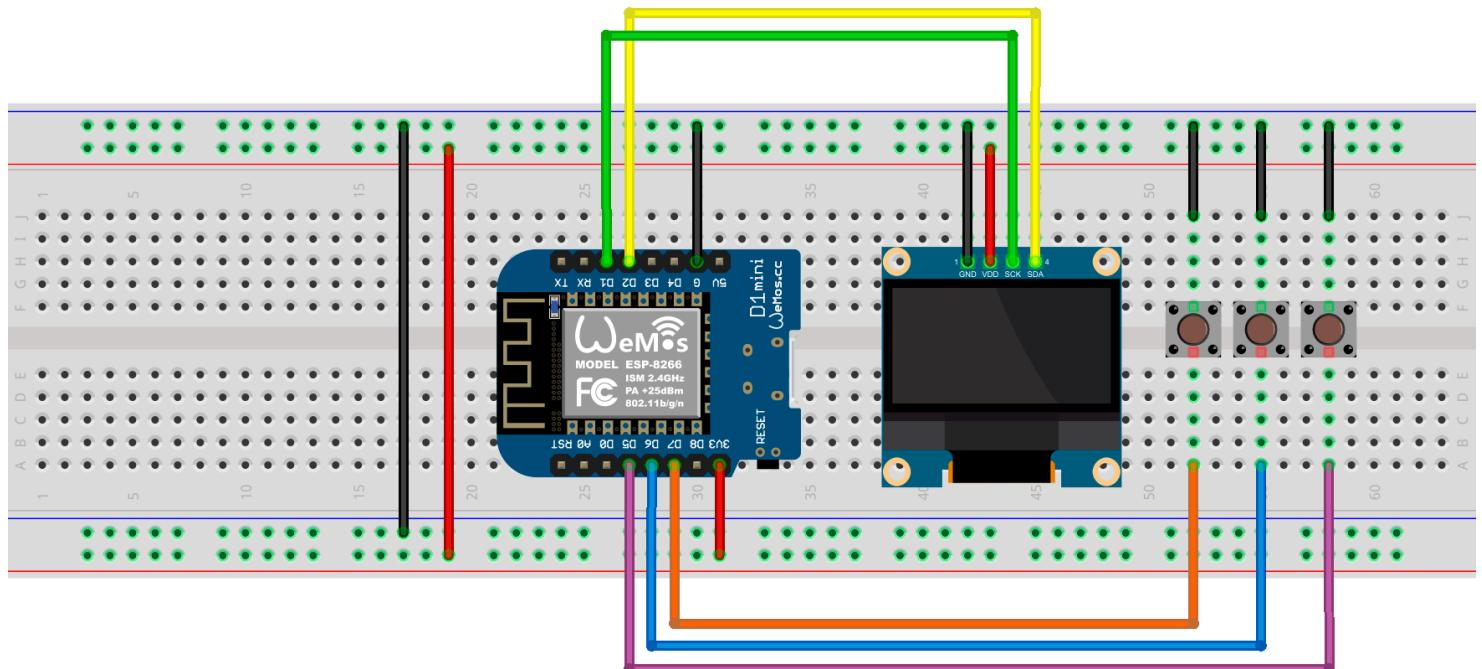
Les traigo un muy breve manual de como crear su propio dstike deauther watch



---COMPONENTES---

- WEMOS mini
- Display Oled 0.96 I2c 128x64 SSD1306
- 3 Pulsadores o Push Button Switch

El diagrama de conexión es sencillo, cuenta con 3 botones para poder desplazarse por el menú. Cada uno corresponde a UP, DOWN, INTRO.



La conexión al display seria

Display	WEMOS
GND	GND

Display	WEMOS
VCC	3.3V
SLC	GPIO5 D1
SDA	GPIO4 D2

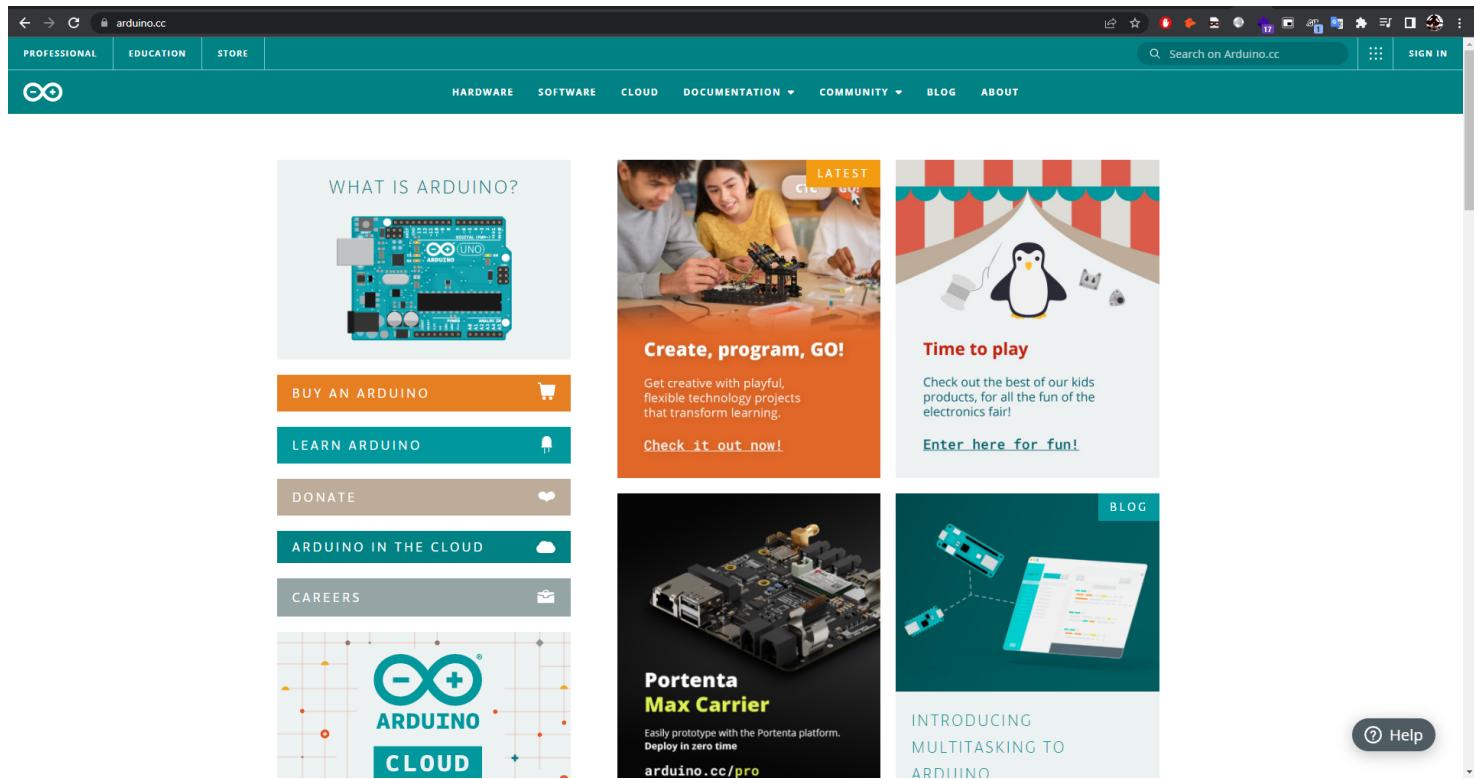
mientras que los botones irán a

BOTON	WEMOS
UP	GPIO13 D7
DOWN	GPIO12 D9
INTRO	GPIO14 D5

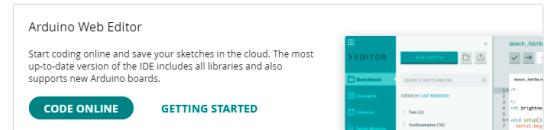
Ahora vamos a programar el cacharrito.

Para esto usaremos la plataforma de Arduino para hacerlo lo más sencillo posible

Nos dirigimos a <https://arduino.cc>



vamos a la pestaña software y pinchamos en "Windows Win 7 and newer"



Downloads

Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE
Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this gpg key](#).

DOWNLOAD OPTIONS

- Windows** Win 7 and newer
- Windows ZIP file**
- Windows app** Win 8.1 or 10 [Get](#)
- Linux** 32 bits
- Linux** 64 bits
- Linux** ARM 32 bits
- Linux** ARM 64 bits
- Mac OS X** 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

Hourly Builds
Download a [preview of the incoming release](#) with the most updated features and bugfixes.

Previous Releases
Download the previous version of the current release, the classic 1.0.x, or old beta releases.

Alice from Arduino

Have you ever tried Arduino Cloud? It allows you to connect devices, visualize data and control your projects from anywhere in the world.

[DISCOVER HOW IT WORKS](#)

"JUST DOWNLOAD" lo guardamos o instalamos al mejor estilo siguiente siguiente siguiente ...

Support the Arduino IDE

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **63.855.601** times — impressive! Help its development with a donation.

[\\$3](#) [\\$5](#) [\\$10](#) [\\$25](#) [\\$50](#) [Other](#)

JUST DOWNLOAD **CONTRIBUTE & DOWNLOAD**

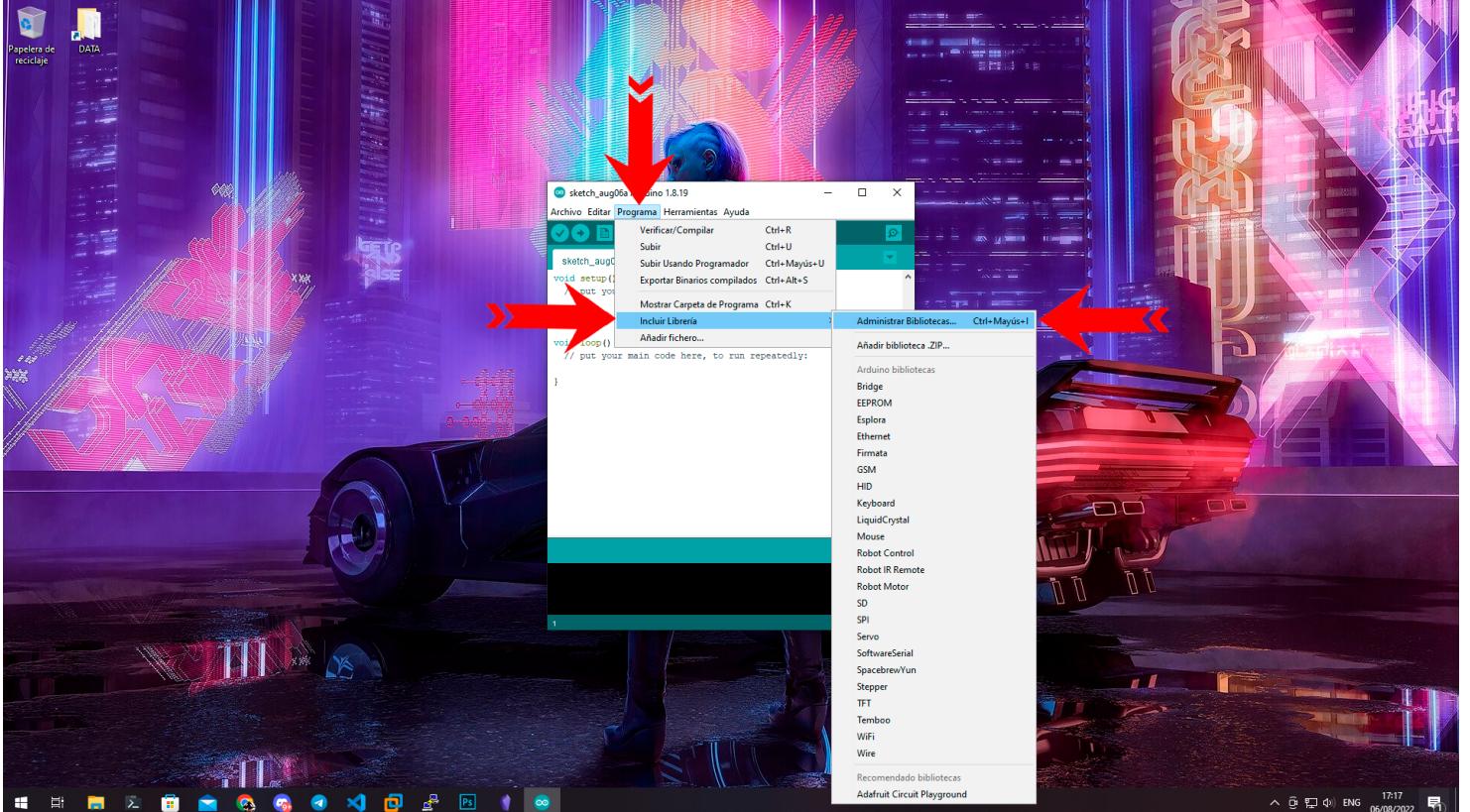
Learn more about [donating to Arduino](#).

Help

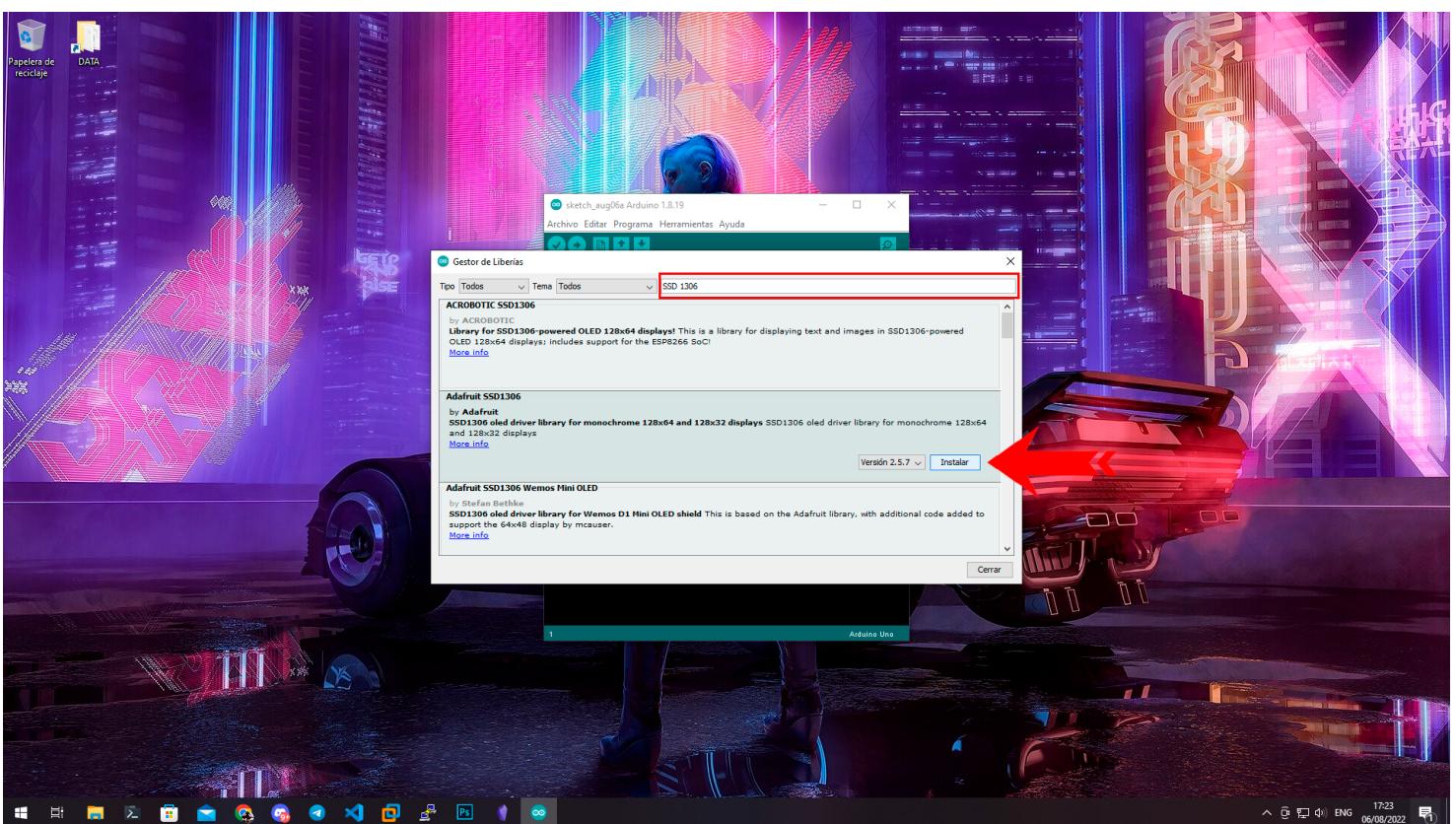
una vez instalado

necesitamos agregar la librería para el display Oled

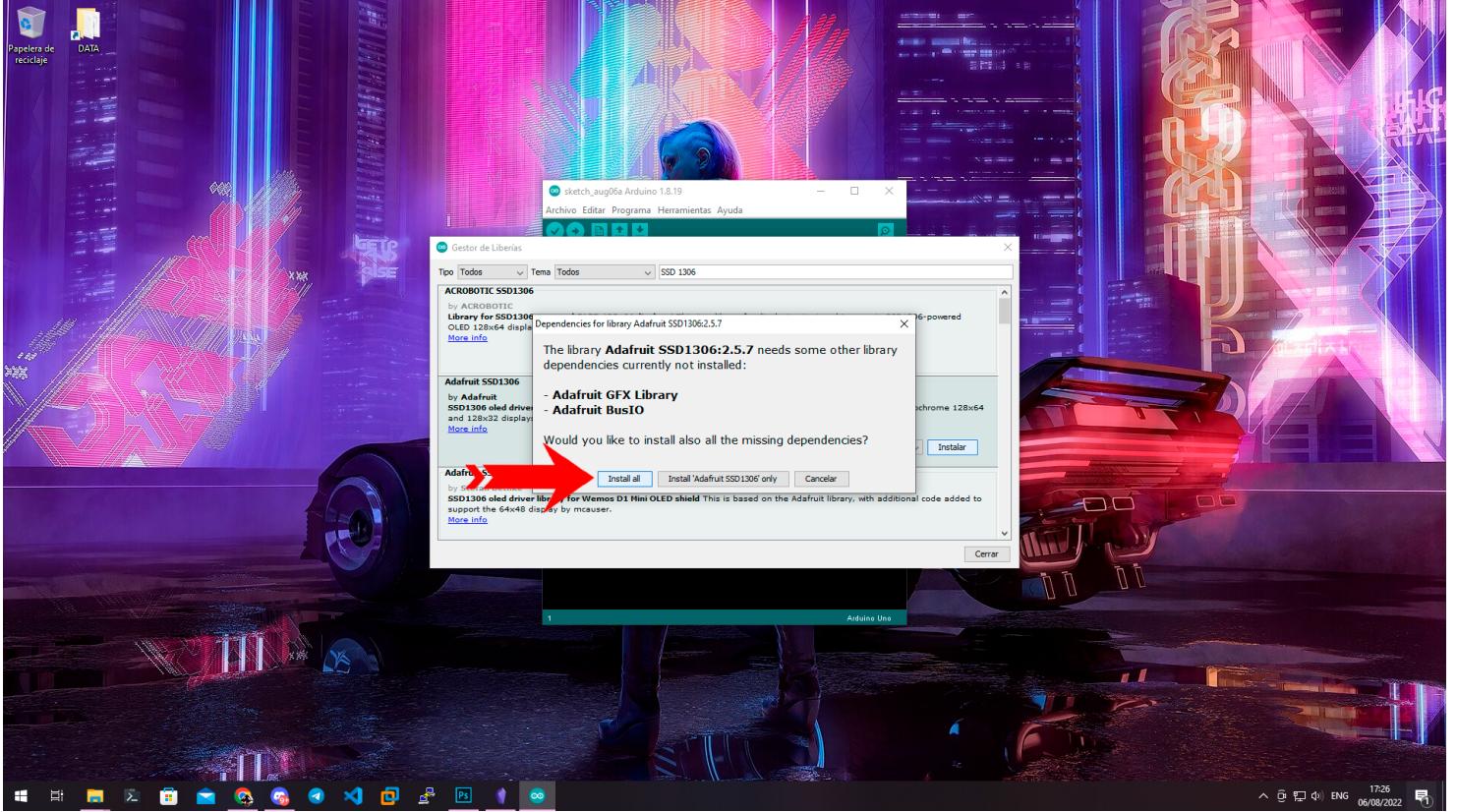
para eso nos dirigimos a Programa Incluir Librería Administrar Bibliotecas



En el cuadro de búsqueda buscamos "SSD 1306" e instalamos el de Adafruit SSD 1306



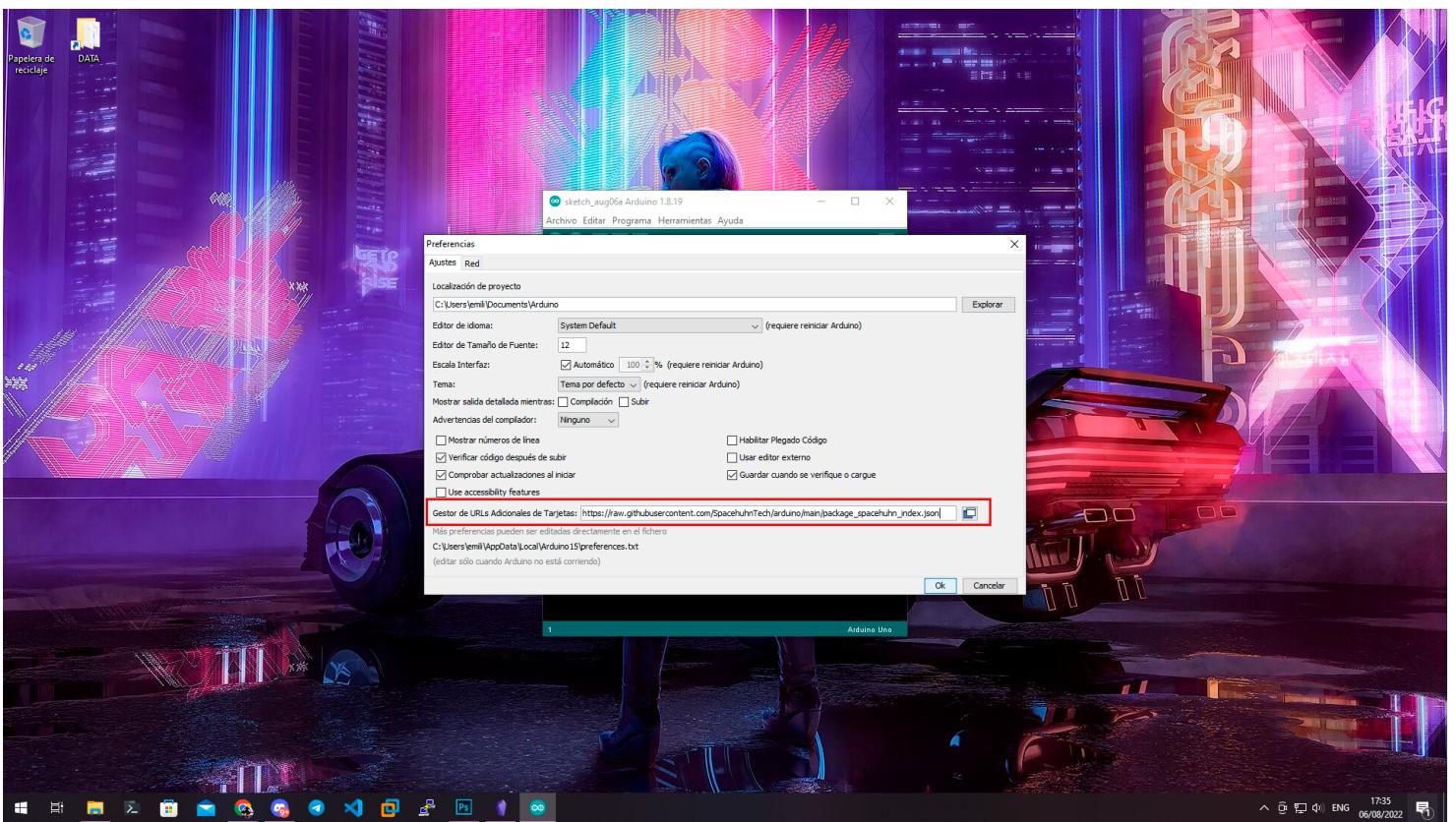
Install all



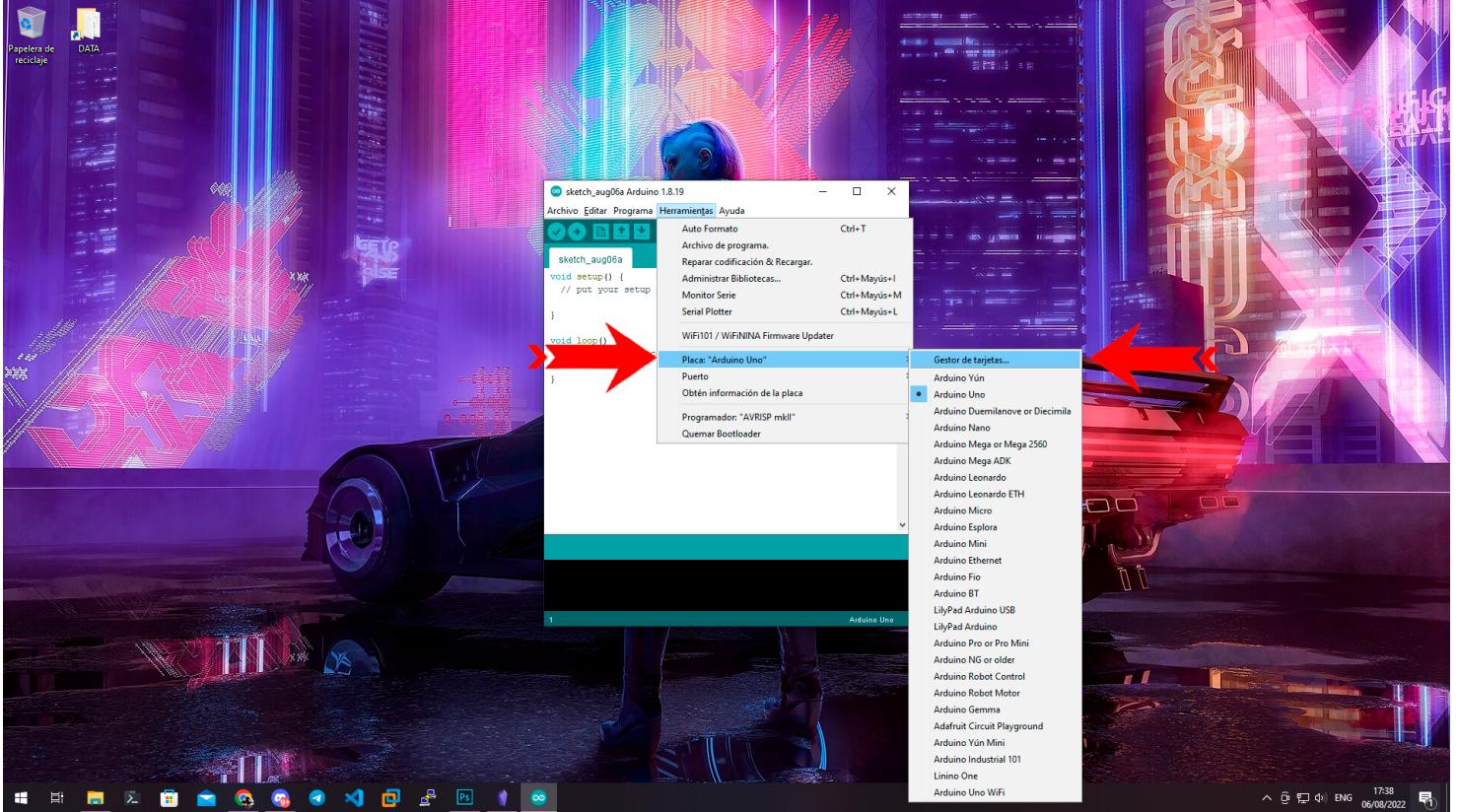
Una vez hecho esto tenemos que agregar la placa con la que vamos a trabajar en Archivo Preferencia

y en el cuadro de "Gestor de URLs Adicionales...." Agregar el siguiente JSON

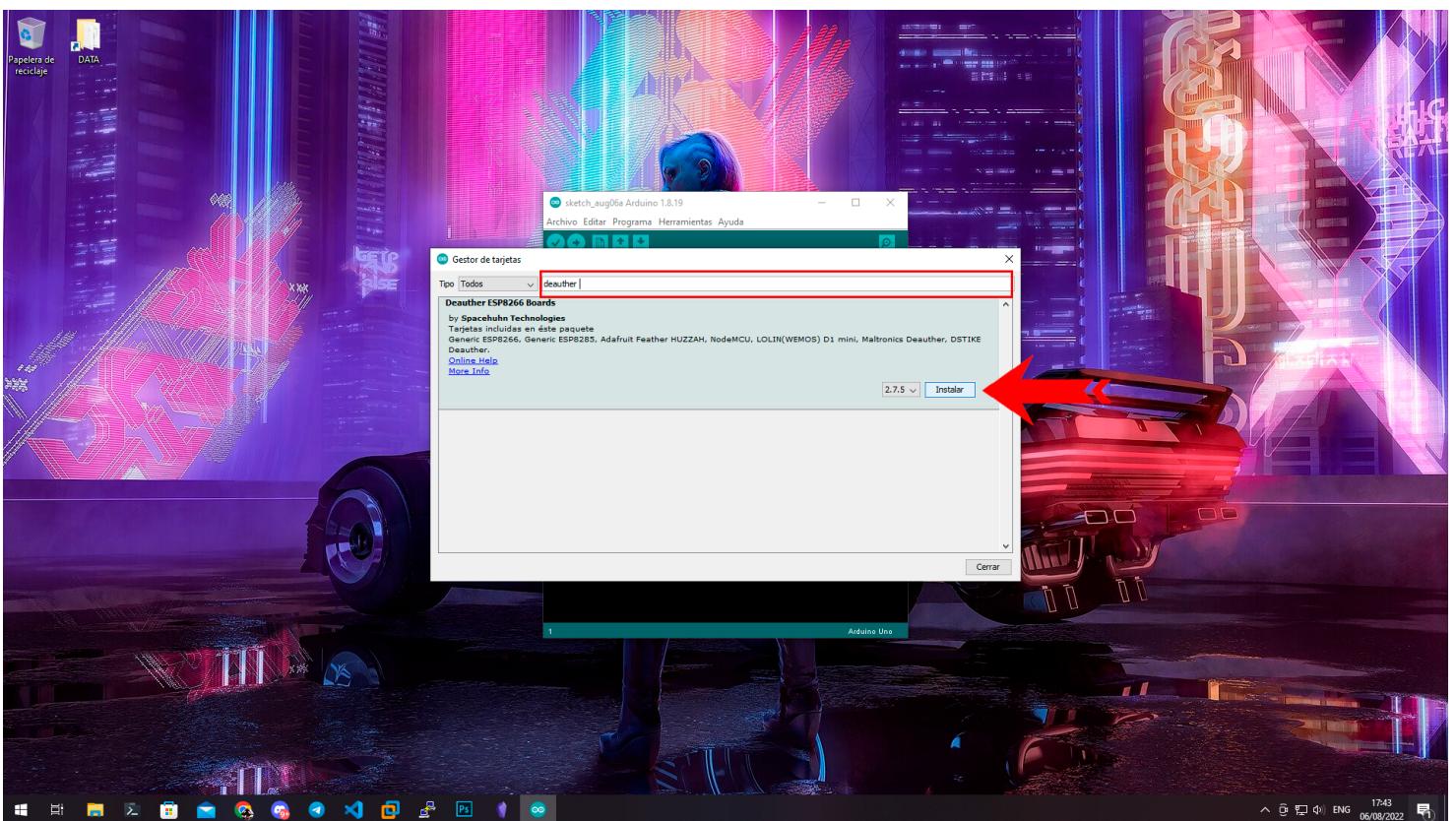
https://raw.githubusercontent.com/SpacehuhnTech/arduino/main/package_spacehuhn_index.json



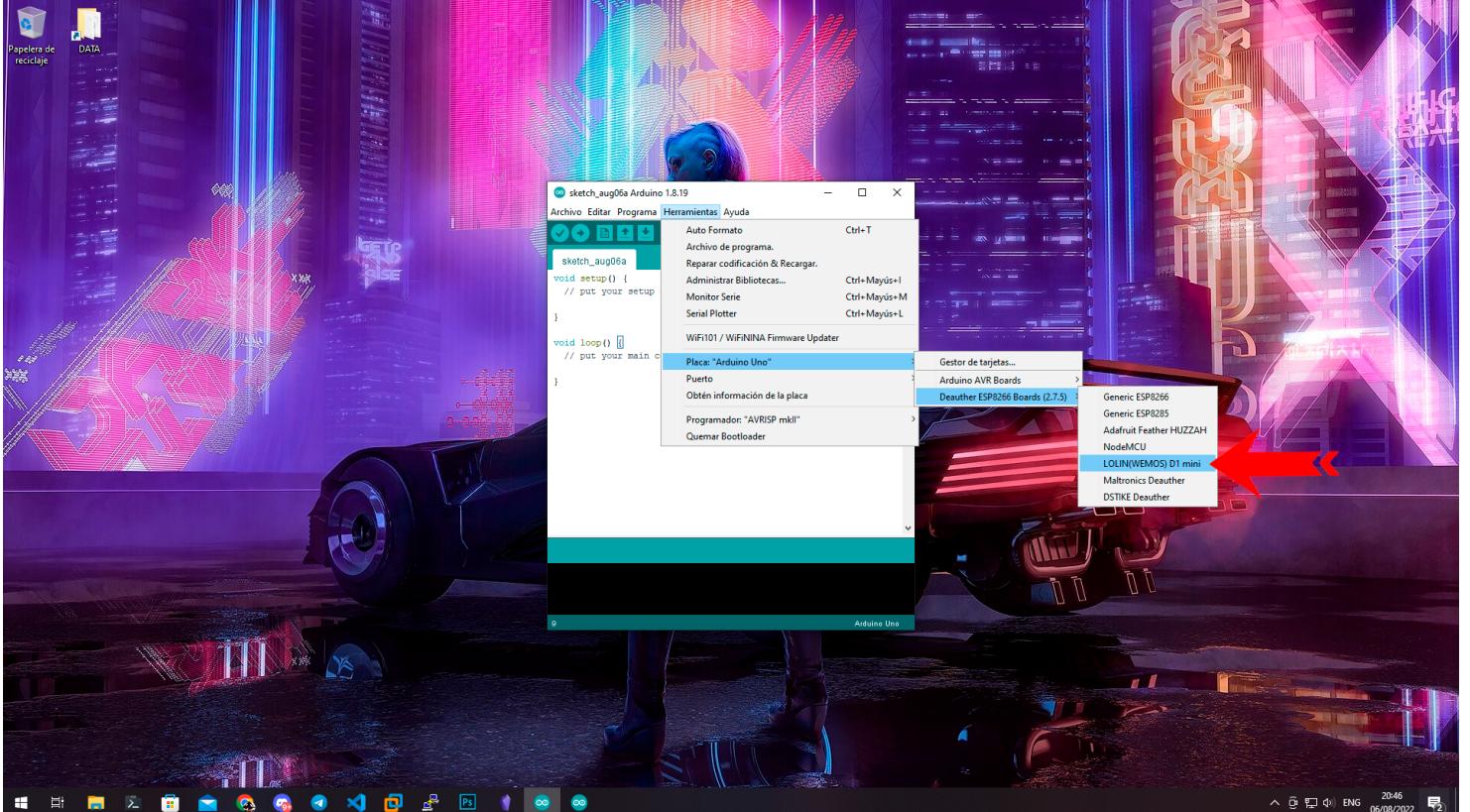
Luego en Herramientas Placa Gestor de tarjeta



En el cuadro de búsqueda buscamos "deauther" y le damos a Instalar



una vez instalado volvemos a Herramientas Placa Deauther ESP8266 Boards
y seleccionamos LOLIN(WEMOS) D1 mini



ya con eso tendríamos listo el entorno para programar el micro.

Ahora nos dirigimos al siguiente repositorio de Git Hub

https://github.com/SpacehuhnTech/esp8266_deauther

yaremos click el Releases

A screenshot of a GitHub repository page for 'SpacehuhnTech / esp8266_deauther'. The page shows the repository's structure, including branches, tags, and recent commits. On the right side, there is an 'About' section with project details like 'Affordable WiFi hacking platform for testing and learning' and a 'Releases' section. A red arrow points to the 'Version 2.6.1 (Latest)' link under the releases section.

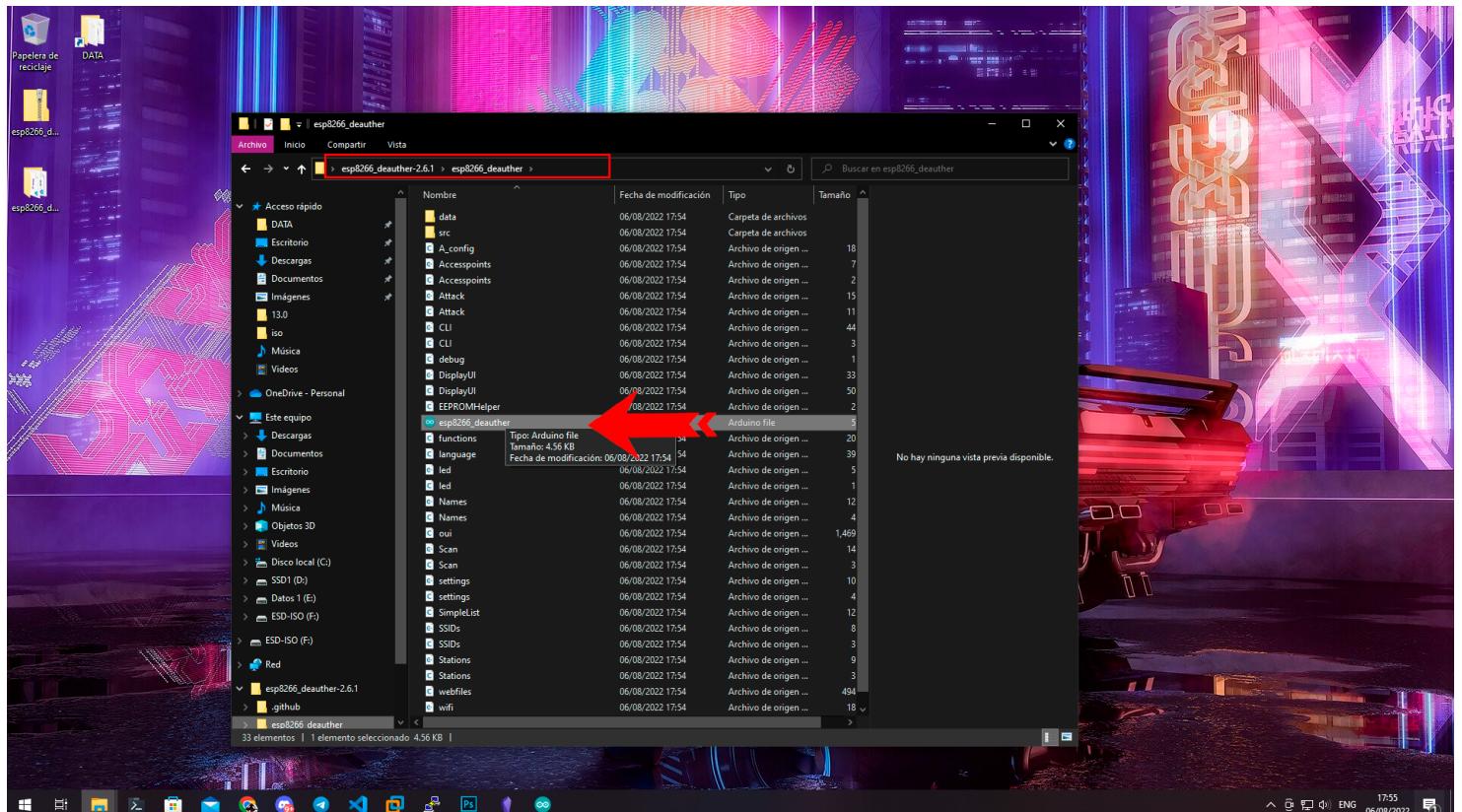
Nos descargamos el Source code de la última versión

esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V2.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V2.5.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V3.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V3.5.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V4.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V5.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_OLED_V6.bin	817 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_V1.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_V2.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_V3.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_V3.5.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_WATCH.bin	816 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DEAUTHER_WATCH_V2.bin	817 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_DUINO_B_V5_LED_RING.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_NODEMCU_07.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_NODEMCU_07_V2.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_USB_DEAUTHER.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_DSTIKE_USB_DEAUTHER_V2.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_HACKHELD_VEGA.bin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_IYASI_7W_E27_LAMPbin	815 KB	07 Aug 2021
esp8266_deauther_2.6.1_MALTRONICS.bin	817 KB	07 Aug 2021
esp8266_deauther_2.6.1_NODEMCU.bin	814 KB	07 Aug 2021
esp8266_deauther_2.6.1_WIFIOS_D1_MINLbin	814 KB	07 Aug 2021
Source code (zip)		
Source code (tar.gz)		

(57) (13) (23) (26) (12) (23) 79 people reacted

Una vez descomprimido nos iremos al directorio /esp8266_deauther-2.6.1/esp8266_deauther y abrimos el archivo

esp8266_deauther.ino



vamos a la pestaña A_config.h y descomentamos la siguiente línea borrando "://"

```

esp8266_deauther - A_config.h | Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda
A_config.h | Accesspoints.cpp | Accesspoints.h | Attack.cpp | Attach | CLICpp | CLIBs | DisplayUI.cpp | DisplayUI.h | EEPROMHelper.h | Names.cpp | Names.h | SSIDs.cpp | SSIDs.h | Scan.cpp | Search | SimpleList.h | Stations.cpp | Stations.h | debug.h | fonts.h

#define ENABLE_DEBUG
#define DEBUG_PORT Serial
#define DEBUG_BAUD 115200
#define DEFAULT_ESP8266

// #define NODEMCU
// #define WEMOS_D1_MINI
// #define HACKFIELD_VEGA
// #define DISPLAY_EXAMPLE_I2C
// #define DISPLAY_EXAMPLE_SPI

// #define MALTRONICS
// #define DSTIKE_DEAUTHER_V1
// #define DSTIKE_DEAUTHER_V2
// #define DSTIKE_DEAUTHER_V3
// #define DSTIKE_DEAUTHER_V5
// #define DSTIKE_D_DUNIO_B_V5_LED_RING
// #define DSTIKE_NODEMCU_07
// #define DSTIKE_NODEMCU_07_V2
// #define DSTIKE_DEAUTHER_OLED
// #define DSTIKE_DEAUTHER_OLED_V1_5_S
// #define DSTIKE_DEAUTHER_OLED_V1_5
// #define DSTIKE_DEAUTHER_OLED_V2
// #define DSTIKE_DEAUTHER_OLED_V2_S
// #define DSTIKE_DEAUTHER_OLED_V3
#define DSTIKE_DEAUTHER_OLED_V3_S
// #define DSTIKE_DEAUTHER_OLED_V4
// #define DSTIKE_DEAUTHER_OLED_V5
// #define DSTIKE_DEAUTHER_OLED_V6
// #define DSTIKE_DEAUTHER_MOSTER
// #define DSTIKE_DEAUTHER_MOSTER_V2
// #define DSTIKE_DEAUTHER_MOSTER_V3
// #define DSTIKE_DEAUTHER_MOSTER_V4
// #define DSTIKE_DEAUTHER_MOSTER_V5
// #define DSTIKE_USB_DEAUTHER
// #define DSTIKE_USB_DEAUTHER_V2
// #define DSTIKE_DEAUTHER_WATCH
// #define DSTIKE_DEAUTHER_WATCH_V2
// #define DSTIKE_DEAUTHER_MINI
// #define DSTIKE_DEAUTHER_MINI_EVO

// #define LYASI_7W_E27_LAMP
// #define AVATAR_SW_E14_LAMP

```

Compilamos el programa para ver que este todo bien y no tengamos ningún programa con alguna librería antes de subir el código...

```

esp8266_deauther - A_config.h | Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda
A_config.h | Accesspoints.cpp | Accesspoints.h | Attack.cpp | Attach | CLICpp | CLIBs | DisplayUI.cpp | DisplayUI.h | EEPROMHelper.h | Names.cpp | Names.h | SSIDs.cpp | SSIDs.h | Scan.cpp | Search | SimpleList.h | Stations.cpp | Stations.h | debug.h | fonts.h

// #define NODEMCU
// #define WEMOS_D1_MINI
// #define HACKFIELD_VEGA
// #define DISPLAY_EXAMPLE_I2C
// #define DISPLAY_EXAMPLE_SPI

// #define MALTRONICS
// #define DSTIKE_DEAUTHER_V1
// #define DSTIKE_DEAUTHER_V2
// #define DSTIKE_DEAUTHER_V3
// #define DSTIKE_DEAUTHER_V5
// #define DSTIKE_D_DUNIO_B_V5_LED_RING
// #define DSTIKE_NODEMCU_07
// #define DSTIKE_NODEMCU_07_V2
// #define DSTIKE_DEAUTHER_OLED
// #define DSTIKE_DEAUTHER_OLED_V1_5_S
// #define DSTIKE_DEAUTHER_OLED_V1_5
// #define DSTIKE_DEAUTHER_OLED_V2
// #define DSTIKE_DEAUTHER_OLED_V2_S
// #define DSTIKE_DEAUTHER_OLED_V3
#define DSTIKE_DEAUTHER_OLED_V3_S
// #define DSTIKE_DEAUTHER_OLED_V4
// #define DSTIKE_DEAUTHER_OLED_V5
// #define DSTIKE_DEAUTHER_OLED_V6
// #define DSTIKE_DEAUTHER_MOSTER
// #define DSTIKE_DEAUTHER_MOSTER_V2
// #define DSTIKE_DEAUTHER_MOSTER_V3
// #define DSTIKE_DEAUTHER_MOSTER_V4
// #define DSTIKE_DEAUTHER_MOSTER_V5
// #define DSTIKE_USB_DEAUTHER
// #define DSTIKE_USB_DEAUTHER_V2

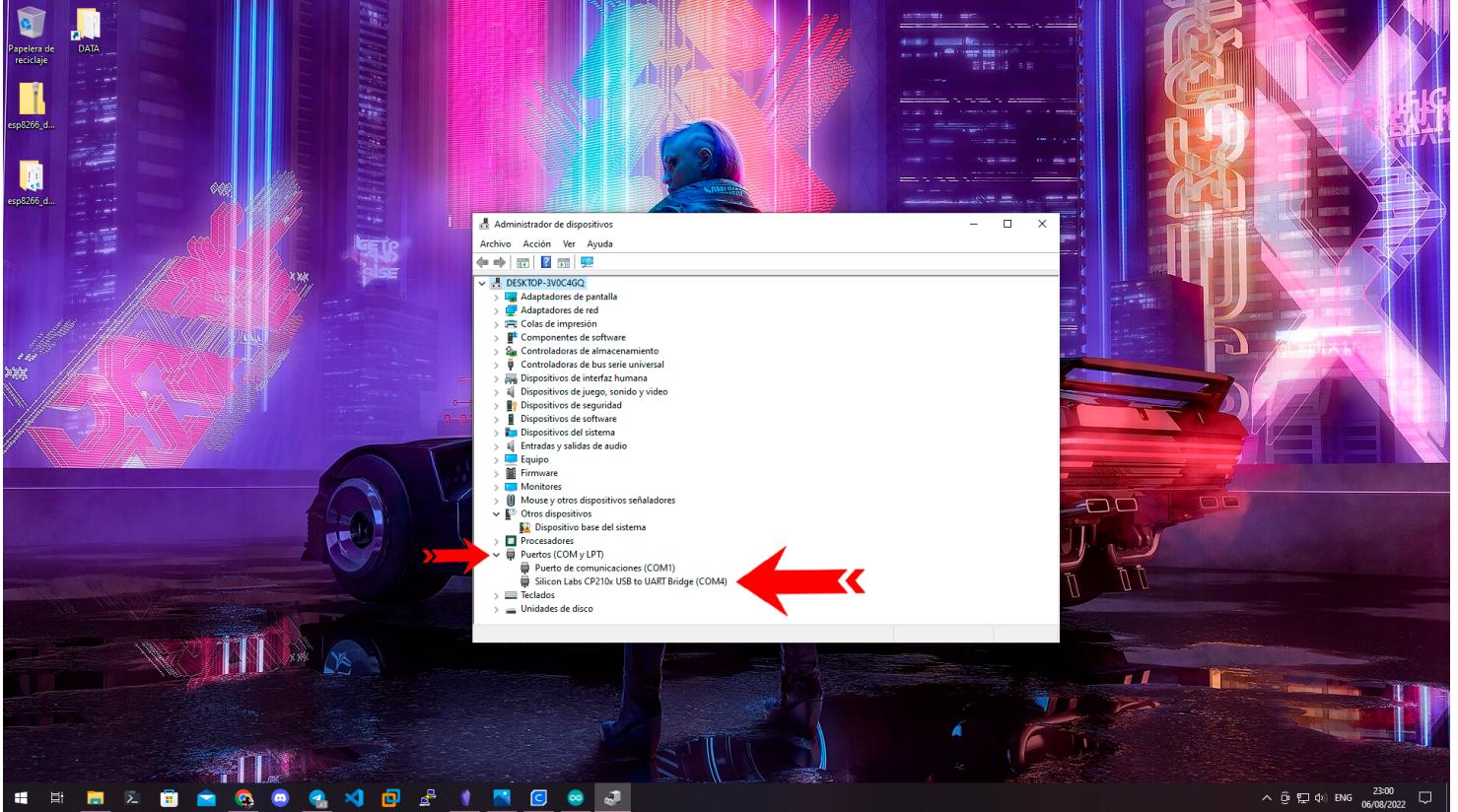
Compilando

Executable segment sizes:
IRAM : 797312 - code in flash (default or ICACHE_FLASH_ATTR)
IRAM : 28912 / 32768 - code in IRAM (ICACHE_RAM_ATTR, ISRs...)
DATA : 1348 ) - initialized variables (global, static) in RAM/HEAP
RODATA : 3360 ) / 1920 - constants (global, static) in RAM/HEAP
BSS : 27480 ) - zeroed variables (global, static) in RAM/HEAP
El Sketch usa 830932 bytes (86%) del espacio de almacenamiento de programa. El máximo es 958448 bytes.
Las variables Globales usan 32188 bytes (3%) de la memoria dinámica, dejando 4972 bytes para las variables locales. El máximo es 81920 bytes.

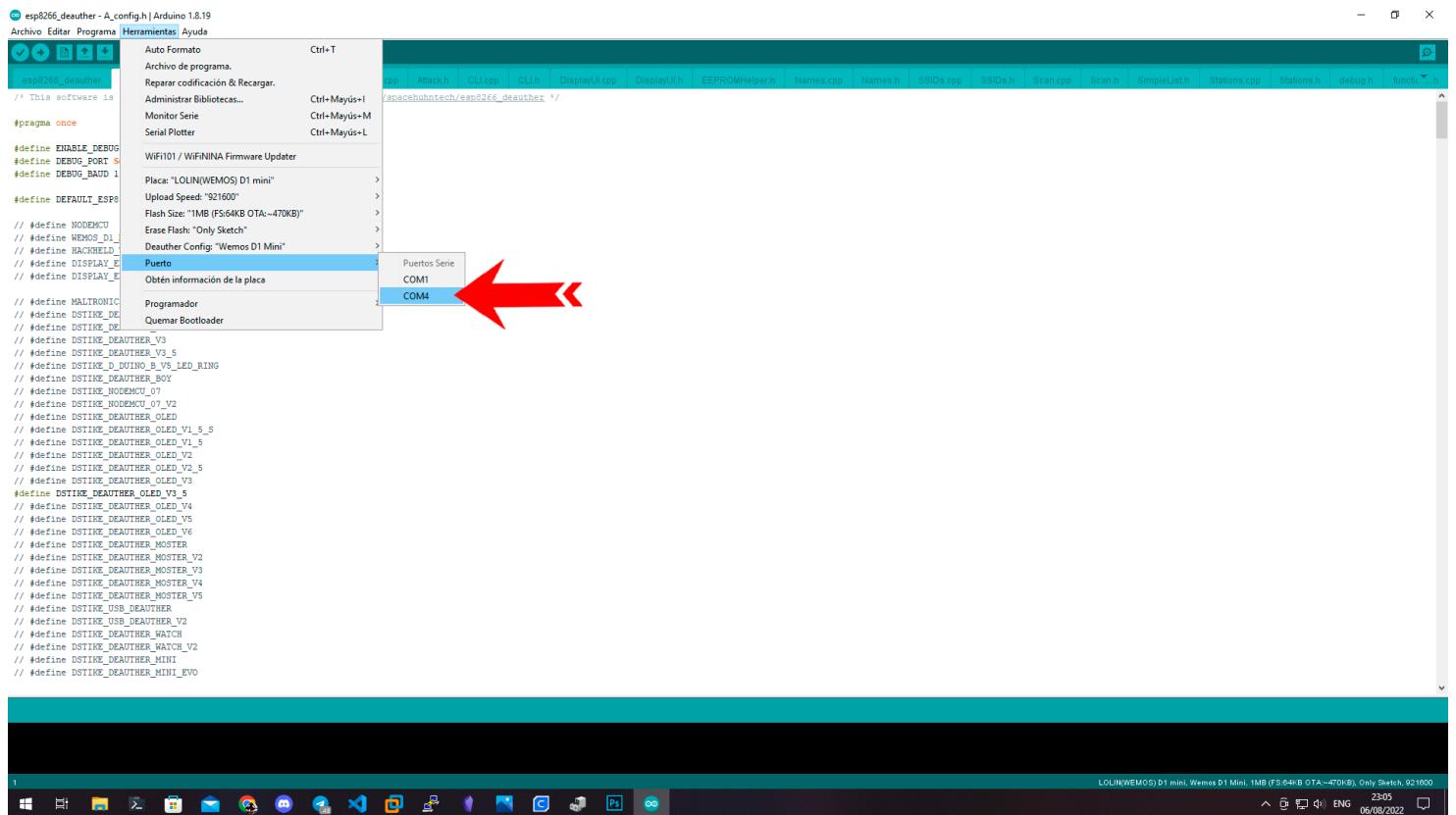
22
LOLINK(WEMOS) D1 mini, Wemos D1 Mini, 1MB (FS:64KB OTA~470kB), Only Sketch, 921600

```

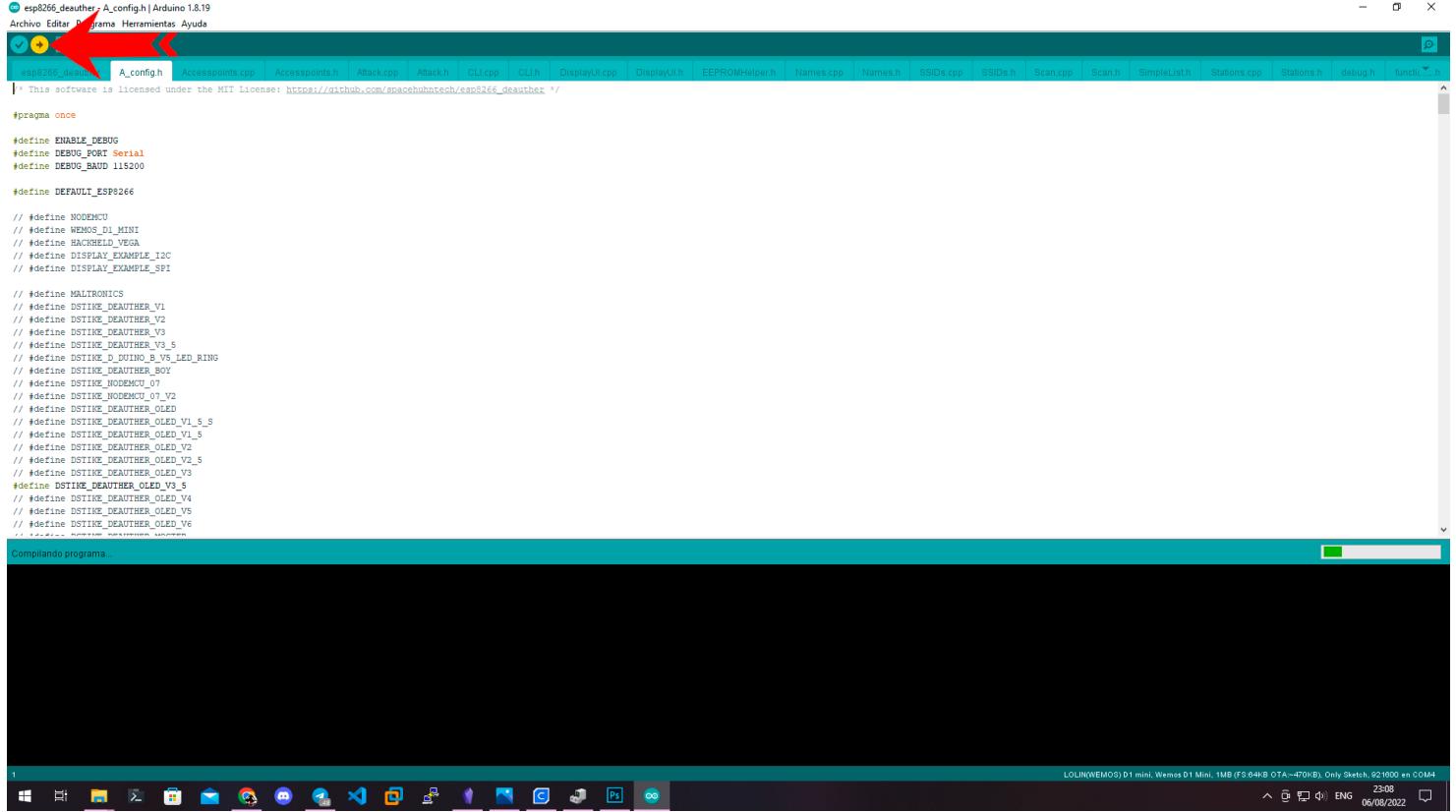
Conectamos nuestra placa por el puerto usb y verificamos que puerto COM nos dio esto lo podemos ver en el Administrador de dispositivos Puerto (COM y LPT) en mi caso seria COM4



viviendo a Arduino IDE vamos a dirigirnos a Herramientas Puerto COM4 (El puerto asignado)



y le damos al botón de subir...



```
esp8266_deauther | A.config.h | Arduino 1.8.19
Archivo Editor Programa Herramientas Ayuda
/* This software is licensed under the MIT License: https://github.com/spacehubnitech/esp8266_deauther */
#pragma once

#define ENABLE_DEBUG
#define DEBUG_PORT Serial
#define DEBUG_BAUD 115200

#define DEFAULT_ESP8266

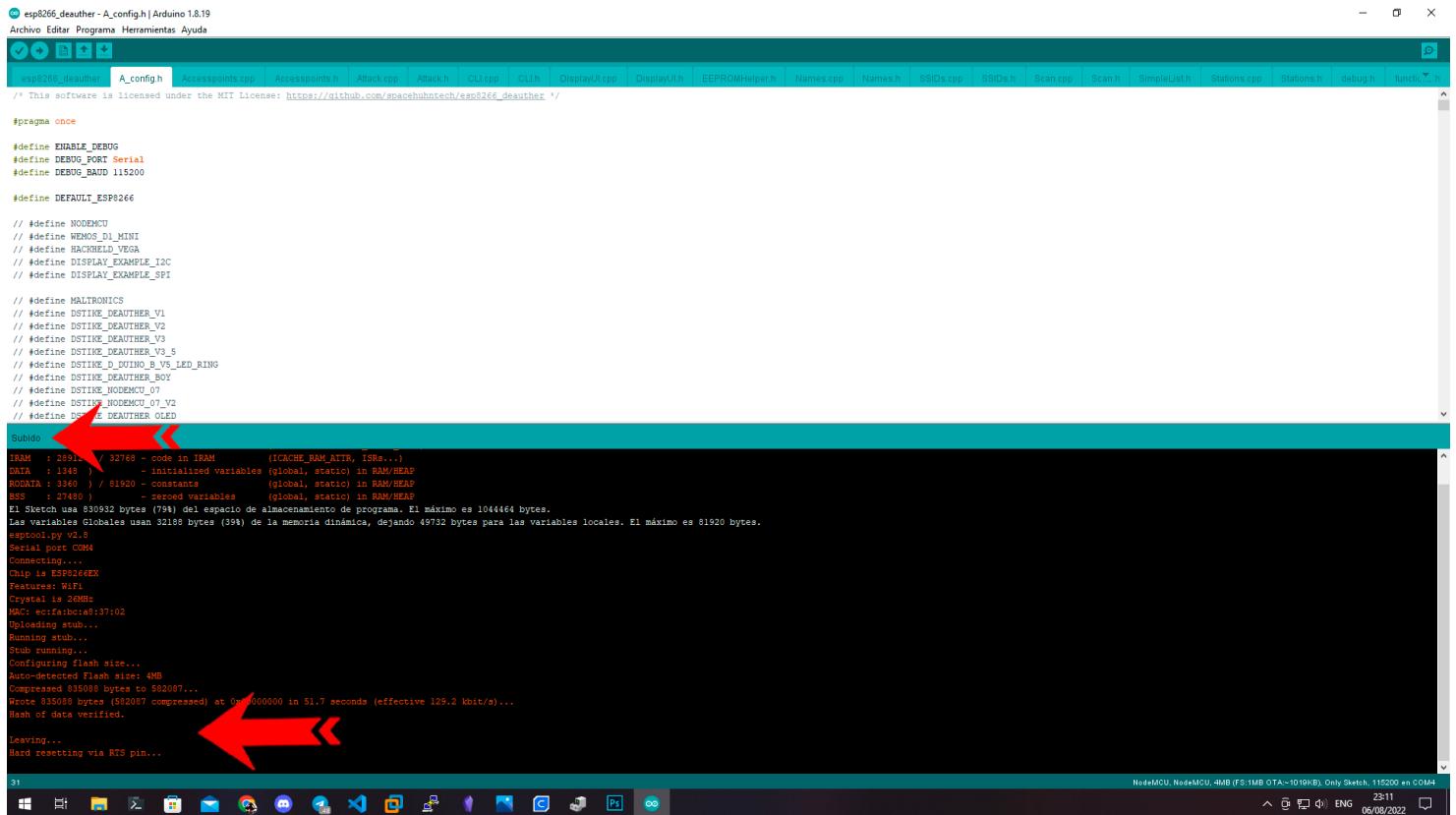
// #define NODEMCU
// #define WEMOS_D1_MINI
// #define HACKRELD_VEGA
// #define DISPLAY_EXAMPLE_I2C
// #define DISPLAY_EXAMPLE_SPI

// #define MALTRONICS
// #define DSTIKE_DEAUTHER_V1
// #define DSTIKE_DEAUTHER_V2
// #define DSTIKE_DEAUTHER_V3
// #define DSTIKE_DEAUTHER_V3_5
// #define DSTIKE_D_DUINO_B_V5_LED_RING
// #define DSTIKE_DEAUTHER_BOY
// #define DSTIKE_NODEMCU_07
// #define DSTIKE_NODEMCU_07_V2
// #define DSTIKE_DEAUTHER_OLED
// #define DSTIKE_DEAUTHER_OLED_V1_5_S
// #define DSTIKE_DEAUTHER_OLED_V1_5
// #define DSTIKE_DEAUTHER_OLED_V2
// #define DSTIKE_DEAUTHER_OLED_V2_5
// #define DSTIKE_DEAUTHER_OLED_V3
// #define DSTIKE_DEAUTHER_OLED_V3_5
// #define DSTIKE_DEAUTHER_OLED_V4
// #define DSTIKE_DEAUTHER_OLED_V4_V
// #define DSTIKE_DEAUTHER_OLED_V5
// #define DSTIKE_DEAUTHER_OLED_V6
// ...various definitions for different boards and components...
```

Compliance programs...

LOLIN(WEMOS) D1 mini, Wemos D1 Mini, 1MB (FS 64KB UTA~470KB), Only Sketch, 92100 en COM4

si todo está bien veremos el siguiente mensaje y se reiniciara nuestro WEMOS



```
esp8266_deauther - A.config.h | Arduino 1.8.19
Archivo Editor Programa Herramientas Ayuda
esp8266_deauther | A.config.h | Arduino 1.8.19
Archivo Editor Programa Herramientas Ayuda
/* This software is licensed under the MIT License: https://github.com/spacehubnitech/esp8266_deauther */
#pragma once

#define ENABLE_DEBUG
#define DEBUG_PORT Serial
#define DEBUG_BAUD 115200

#define DEFAULT_ESP8266

// #define NODEMCU
// #define WEMOS_D1_MINI
// #define HACKRELD_VEGA
// #define DISPLAY_EXAMPLE_I2C
// #define DISPLAY_EXAMPLE_SPI

// #define MALTRONICS
// #define DSTIKE_DEAUTHER_V1
// #define DSTIKE_DEAUTHER_V2
// #define DSTIKE_DEAUTHER_V3
// #define DSTIKE_DEAUTHER_V3_5
// #define DSTIKE_D_DUINO_B_V5_LED_RING
// #define DSTIKE_DEAUTHER_BOY
// #define DSTIKE_NODEMCU_07
// #define DSTIKE_NODEMCU_07_V2
// #define DSTIKE_DEAUTHER_OLED
Subido


```
IRAM : 28912 / 32768 - code in IRAM (ICACHE_RAM_ATTR, ISRs...)
DATA : 1348) - initialized variables (global, static) in RAM/HEAD
RODATA : 3360) / 81920 - constants (global, static) in RAM/HEAD
BSS : 27680) - zeroed variables (global, static) in RAM/HEAD
El Sketch tiene 330932 bytes (7%) del espacio de almacenamiento de programa. El máximo es 104444 bytes.
Las variables Globales usan 32188 bytes (9%) de la memoria dinámica, dejando 49704 bytes para las variables locales. El máximo es 81920 bytes.
Sketchbook: v2.3.1
Serial port: COM4
Connecting...
Chip is ESP8266EX
Features: WiFi
Crystal is 26MHz
MAC: ee:fb:eb:c8:37:02
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 655088 bytes to 582087...
Wrote 655088 bytes (582087 compressed) at 0x00000000 in 51.7 seconds (effective 129.2 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

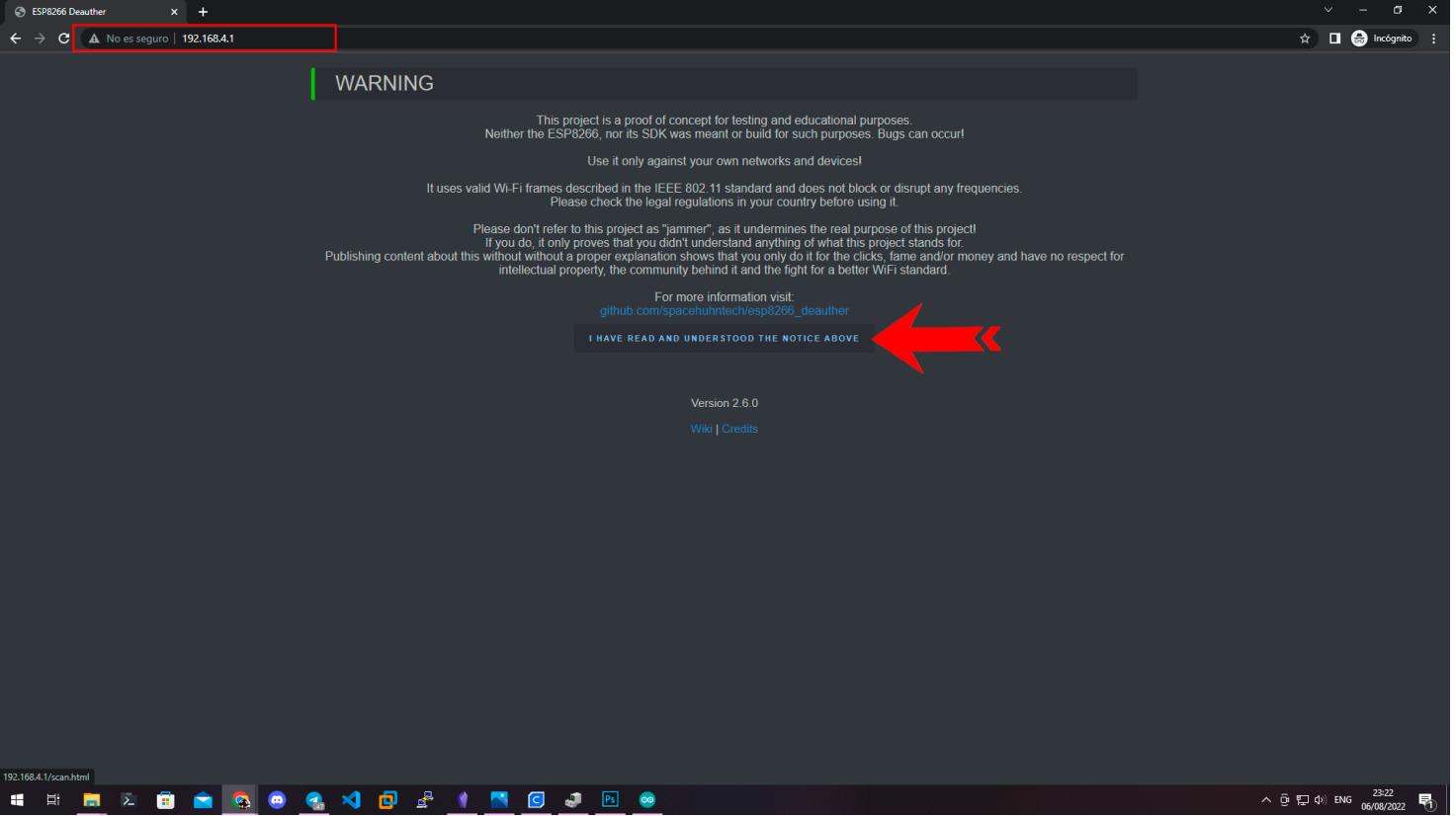
31
NodeMCU_NodemCU_RMB (FS:1MB OTA~1019KB), Only Sketch, 115200 en COM4
```

Una vez iniciado nos mostrara el menú en display, además nos levantara un AP wifi con el BSSID: "pwned" al cual nos podremos conectar para administrarlo.

BSSID: pwned

PASS: deauther

una vez conectado si nos dirigimos a la ip 192.168.4.1 o deauth.me nos mostrara el dashboard



This screenshot shows the "Scan" tab of the ESP8266 Deauther web interface. The top navigation bar includes tabs for "Scan", "SSIDs", "Attack", and "Settings". The main area is titled "Scan" and shows a table of "Access Points: 4". The table columns are SSID, Name, Ch, RSSI, Enc, MAC, and Vendor. The rows show four access points with their respective details. Below the table are buttons for "SELECT ALL" and "DESELECT ALL". A second section titled "Stations: 5" is partially visible below it. The Windows taskbar at the bottom shows various icons and the date/time as 23-23 06/08/2022.

Lo bueno es que podemos omitir ponerle el display y los botones y quedarnos con el WEMOS solo u ocultarlo dentro de un cargador usb para poder usar su alimentación y controlarlo de manera remota con un celular.

De aquí en adelante que a su imaginación...

Happy Hacking!!!

ak4m3_x

EOF

twitter: https://twitter.com/ak4m3_x

Créditos del Proyecto a SpacehuhnTech

<https://github.com/SpacehuhnTech>