

IoT-Wifi-Blynk

Legacy

- [Conseguir la App BLYNK LEGACY](#)
- [Crear cuentas](#)
- [Entrando en el Blynk Legacy: El panel de control](#)
- [En la APP](#)
- [Poner el TOKEN en ARDUINOBLOCKS](#)
- [Primer programa: Encender los pines rojo y verde](#)
- [Segundo programa RGB y el potenciómetro](#)
- [Tercer programa Medir la luz del LDR](#)
- [Cuarto programa: Medir Temperatura y Humedad con el DHT11](#)
- [Quinto programa: Lectura sensor LM35 y receptor infrarrojos](#)
- [Sexto programa Leer eventos del TDR Steam](#)
- [Séptimo programa: El timbre](#)
- [¿Todo junto?](#)

Conseguir la App BLYNK LEGACY

La APP de **BLYNK LEGACY** no la encontramos en el buscador de Google Play de forma tradicional, pues está sin mantenimiento, tenemos varias opciones para descargar la APP.

OJO no confundir Blynk (legacy) que es el que vamos a trabajar



con Blynk IoT que **sí** que lo encuentras en el buscador de Google Play



OPCIÓN A Desde Google Play

En este enlace aún tiene la APP antigua, pero es de propiedad de Blynk Inc luego **no es de extrañar que desaparezca.**

<https://play.google.com/store/apps/details?id=cc.blynk>



Blynk (legacy)

Blynk Inc.

In-app purchases

Uninstall

Open

What's new •

Last updated 3 May 2022



Fixed issues with gps widget, home screen widgets on Android 11 and later

Rate this app

Tell others what you think



[Write a review](#)

Developer contact



Join the beta

Try new features before they're officially released and give your feedback to the developer. Certain data on your use of the app will be collected and shared with the developer to help improve the app.



[Join](#) [Learn more](#)



OPCION B Fuera de Google Play

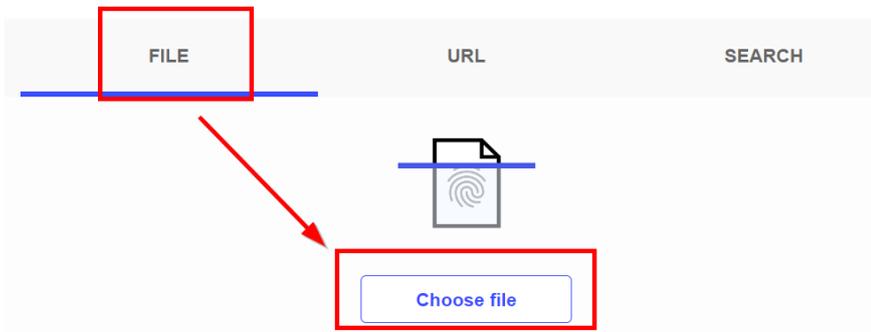
OPCIÓN B: ANTES DE LA INSTALACIÓN

No nos tenemos que fiar de descargar APK de repositorios fuera de Google Play, luego una vez descargado, pasamos el fichero por una página de Antivirus, por ejemplo

<https://www.virustotal.com/gui/home/upload>



Analyze suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community



No nos tiene que dar ningún virus

1d0293785d280f922bed9e0bf99866b9b431de1c24a50cc1104594efdb1bc30e

0 / 61

✓ No security vendors and no sandboxes flagged this file as malicious

1d0293785d280f922bed9e0bf99866b9b431de1c24a50cc1104594efdb1bc30e
blynk-2-27-32.apk

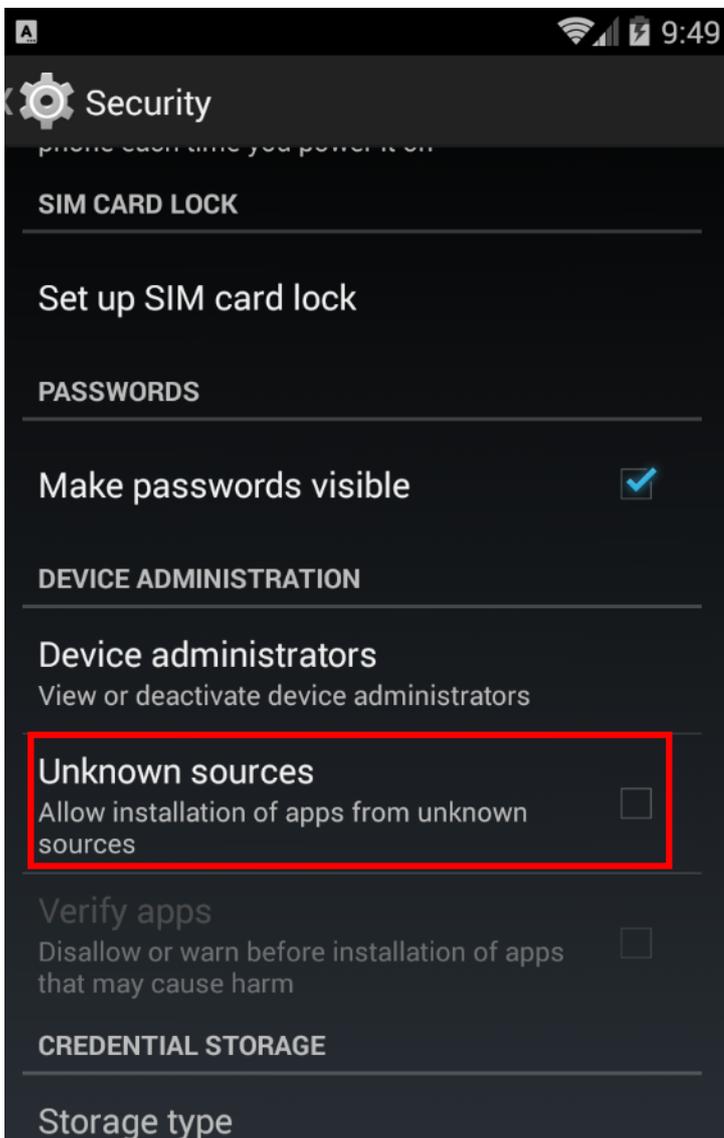
android apk contains-elf reflection runtime-modules telephony

Community Score

Para instalarlo en tu móvil:

Primero, tenemos que llevar el fichero .apk a tu dispositivo, recomendamos usar email, por ejemplo enviándote a ti mismo un correo electrónico y lo consultas desde el móvil.

Segundo, tenemos que instalarlo y para ello hay que decirle que confiamos con apps instaladas fuera del repositorio oficial (Google Play), como depende las pantallas del modelo de tu móvil, puedes encontrar cualquier tutorial en internet cómo se hace pero prácticamente es llegar a la misma pantalla en seguridad/privacidad- permitir apps de orígenes desconocidos.



Tercero instalarlo

Cuarto, deshacer el segundo paso

OPCIÓN B DESCARGA

Puedes buscar en un navegador esta APP, encontraras varios sitios para descargarla, por ejemplo aquí:

- <https://blynk.uptodown.com/android/descargar>
- <https://blynk.en.uptodown.com/android/download/3921223>
- Github: <https://github.com/BlynkMobile>
- Desde este repositorio nuestro



Blynk Client

BlynkMobile

Follow

Since Blynk.io decided to shut down the Blynk app and local server, we decided to clone the mobile app and develop it with the community. Help us keep the legacy

👤 3 followers · 2 following

<https://www.reddit.com/r/blynkmobileapp/>

🐦 @MobileBlynk

Block or Report

Crear cuentas

Antes se creaban cuentas en Blynk usando el correo electrónico: Cada usuario en la APP de Blynk podía crearse una cuenta, enviando un email. El servidor local Blynk puede enviar correos electrónicos si tiene configurado un fichero que se llama mail.properties y se envía el correo con las credenciales. No obstante esta opción de crear cuenta **ha desaparecido** de la APP (En CATEDU ya *barruntábamos* esto, que Blynk Inc al quitar crear nuevos usuarios es que pensaba quitar este servicio Blynk legacy).

No obstante, la creación de usuarios mediante emails era un problema pues menores de 16 años no pueden tener emails ni redes sociales, ni whatsapps (¿oigo risas?).

En el curso vas a utilizar los servidores de CATEDU luego tendrás asignado 12 cuentas de alumnos + un profesor

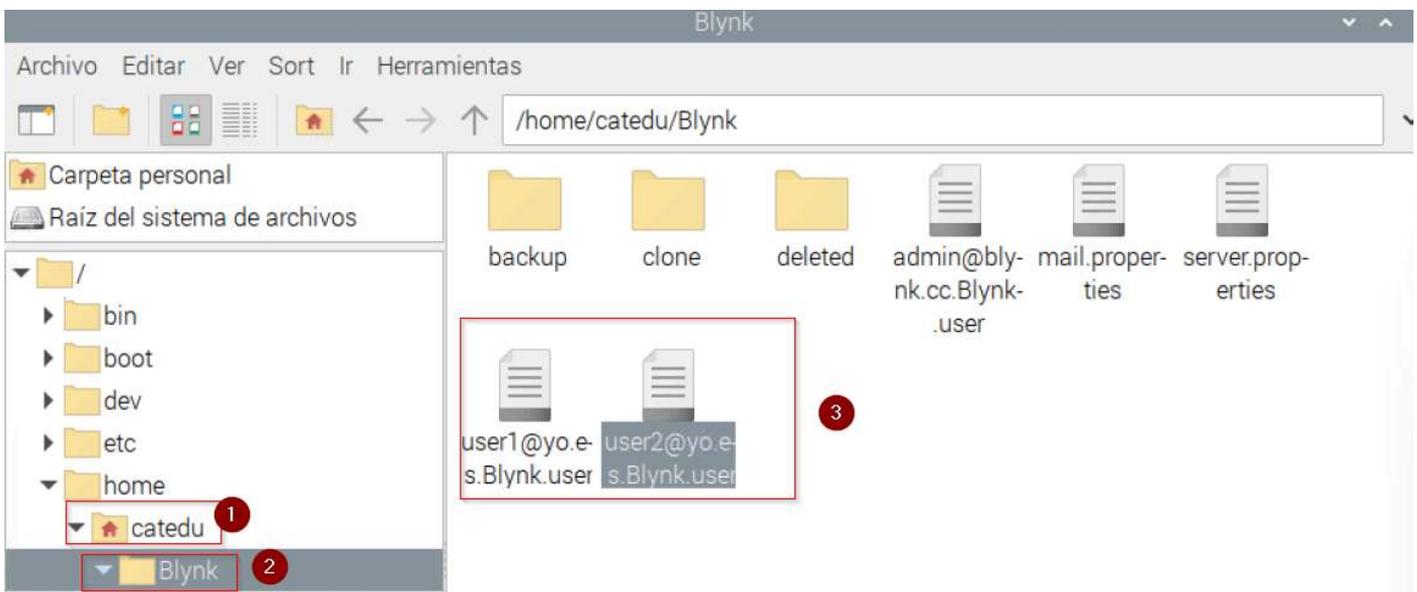
Recibirás un email con los nombres de usuario y las contraseñas

¿Y fuera del curso?

En este caso necesitarás tener un servidor Blynk legacy local que tal y como lo explicamos en los siguiente capítulos, montado en una Raspberry.

Entramos en la **carpeta de Blynk en nuestra Raspberry**, ya sea por VNC, o con una pantalla, teclado y ratón o por SSH y lo que hay que hacer es crear unos ficheros texto que se llamen **emailusuario.Blynk.user** los emails de los usuarios pueden ser inventados, por ejemplo en la figura puedes ver que en la raspberry hay dos usuarios [user1@yo.es](#) [user2@yo.es](#) luego los ficheros textos son **user1@yo.es.Blynk.user** y el otro es **user2@yo.es.Blynk.user**.

Entrando por VNC o pantalla, teclado y ratón:



Entrando por SSH

```
login as: catedu
catedu@192.168.245.80's password:
Linux raspberrypi 5.15.61-v7+ #1579 SMP Fri Aug 26 11:10:59 BST 2022 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 13 15:14:43 2022
catedu@raspberrypi:~$ cd Blynk
catedu@raspberrypi:~/Blynk$ ls
admin@blynk.cc.Blynk.user  clone  mail.properties  user1@yo.es.Blynk.user
backup                    deleted  server.properties  user2@yo.es.Blynk.user
catedu@raspberrypi:~/Blynk$ sudo nano user1@yo.es.Blynk.user
GNU nano 5.4
{"name":"user1@yo.es","email":"user1@yo.es","appName":"Blynk","region":"local","ip
```

El contenido por ejemplo del fichero **user2@yo.es.Blynk.user** es:

```
{"name":"user2@yo.es","email":"user2@yo.es","appName":"Blynk","region":"local","ip":"ip_local_raspberry","pass":"uffzlHwvej
aeZZz2PzqmKuB22OACUDRLWvKEI7RCDQI=","lastModifiedTs":1663010486621,"lastLoggedIP":"192.168.1.73","lastLoggedAt":1635057921978,"profile":{},"isFacebookUser":false,"isSuperAdmin":false,"energy":99600,"id":"user2@yo.es-Blynk"}
```

Luego lo único que tienes que hacer para crear nuevos usuarios es **crear ficheros texto que se llamen `userx@yo.es.Blynk.user`** y con contenido igual que el anterior pero cambiando el

numero 2 por x (si quieres respetar userx@yo.es pero puedes inventarte cualquier email ficticio)

¿Por qué puede ser que el contenido de un fichero pueda tener más información o menos que otros?

Porque en esos ficheros txt tiene la información de los proyectos creados por el usuario

¿Entonces si el fichero user1@yo.es.Blynk.user tiene toda la información del usuario y todas sus proyectos... si ese usuario quiere llevarse sus proyectos a otro servidor Blynk sólo tiene que copiar y pegar es fichero texto?

Respuesta : SI , así de sencillo, ese fichero texto (por lo tanto libre de virus) tiene toda la información.

¿En la APP no se almacena ningún proyecto?

Ninguno, todos los proyectos están almacenados en esos ficheros txt si no hay conexión con el servidor, en la APP no aparece NADA

Vale, ya sé crear los nombres de usuarios ¿Y las contraseñas?

En teoría están encriptadas en el fichero texto, como puedes ver, sale
uffzIHwvejaeZZz2PzqmKuB22OACUDRLWvKEI7RCDQI= eso significa *alcorisa*

Pero tranquilo, luego veremos en el panel de control cómo se pueden poner las contraseñas sin encriptar

+información [aquí](#)

Entrando en el Blynk Legacy: El panel de control

Como entro en el panel de control

- **En el curso de CATEDU**
 - Habrás recibido un email con la URL para entrar en la administración web del Blynk Legacy, con usuario y contraseña
- **Fuera del curso de CATEDU usando tu servidor local**
 - Desde cualquier ordenador conectado en la misma red local que la raspberry, ejecutas en un navegador la siguiente dirección:
 - Si la IP de la Raspberry es 192.168.1.112 entonces entramos en:
 - <https://192.168.1.112:9443/admin>

Vale, ya estoy ¿y ahora qué?

Seguramente te saldrá la siguiente advertencia por el certificado SSL, dale a **Configuración avanzada** y luego a **Acceder a (la IP del servidor Blynk Legacy) sitio no seguro**



La conexión no es privada

Es posible que los atacantes estén intentando robar tu información de **192.168.43.111** (por ejemplo, contraseñas, mensajes o tarjetas de crédito). [Más información](#)

NET::ERR_CERT_AUTHORITY_INVALID

💡 Para disfrutar del máximo nivel de seguridad en Chrome, [activa la protección mejorada](#).

Ocultar configuración avanzada

1

Volver para estar a salvo

Este servidor no ha podido probar que su dominio es **192.168.43.111**, el sistema operativo de tu ordenador no confía en su certificado de seguridad. Este problema puede deberse a una configuración incorrecta o a que un atacante haya interceptado la conexión.

Acceder a 192.168.43.111 (sitio no seguro)

2

Luego entramos con el usuario y contraseña que nos han proporcionado

Cambiando las contraseñas de los alumnos

Y podemos ver los usuarios creados con los ficheros textos mencionados anteriormente

Blynk Administration

- Users
- Stats >
- Hardware Info >
- Config

Users list

<input type="checkbox"/>	Email
<input type="checkbox"/>	user1@yo.es
<input type="checkbox"/>	user2@yo.es
<input type="checkbox"/>	admin@blynk.cc

Entrando en un usuario PODEMOS GENERARLE UNA CONTRASEÑA, por supuesto la pones normal, sin encriptar

También puedes borrar el usuario si no nos interesa este usuario.

Blynk Administration

- Users
- Stats >
- Hardware Info >
- Config

Edit user "user1@yo.es"

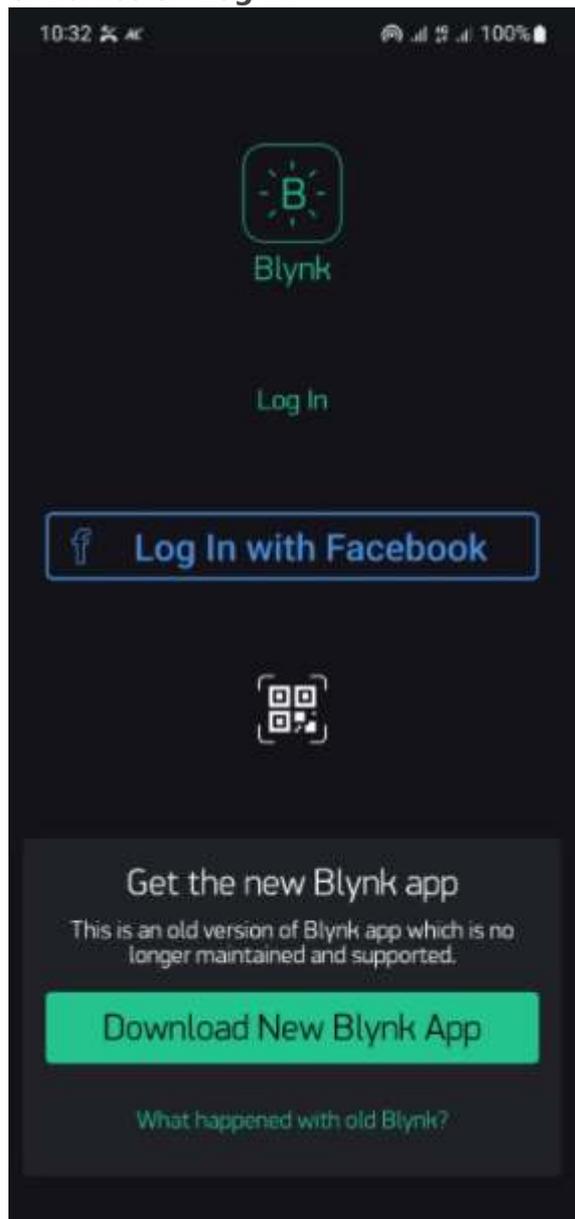
List Delete

Email	user1@yo.es
Name	user1@yo.es
Pass

En la APP

Conectar con el nuevo usuario

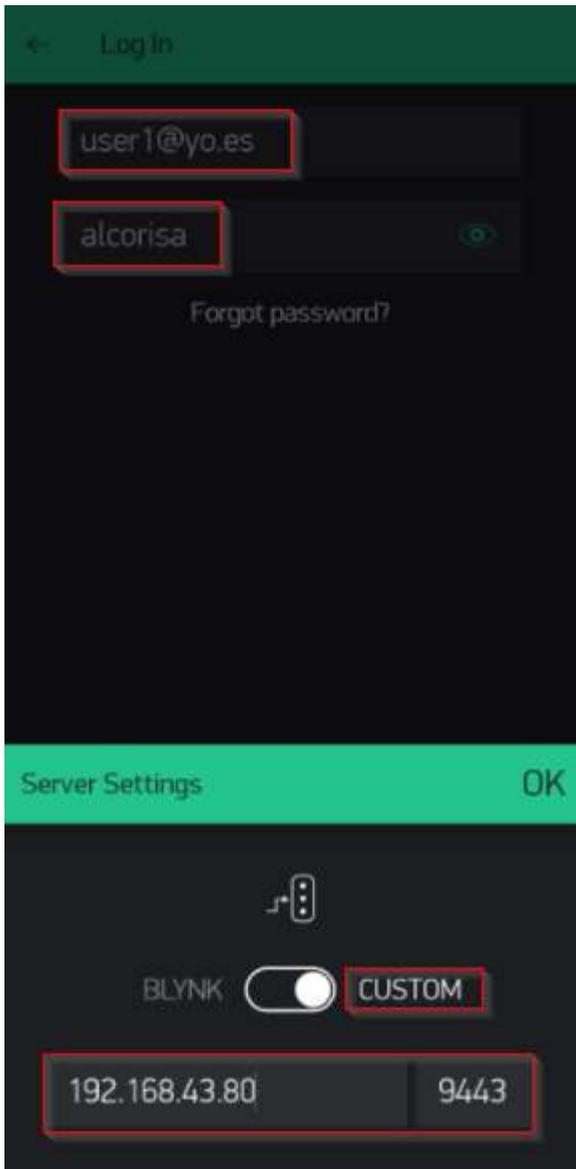
Entramos en la APP de Blynk que hemos visto anteriormente como descargarla e instalara y entramos en **Log In**



Y ponemos el nombre de usuario que hemos creado anteriormente, y la contraseña, la que hayamos definido en el panel de control

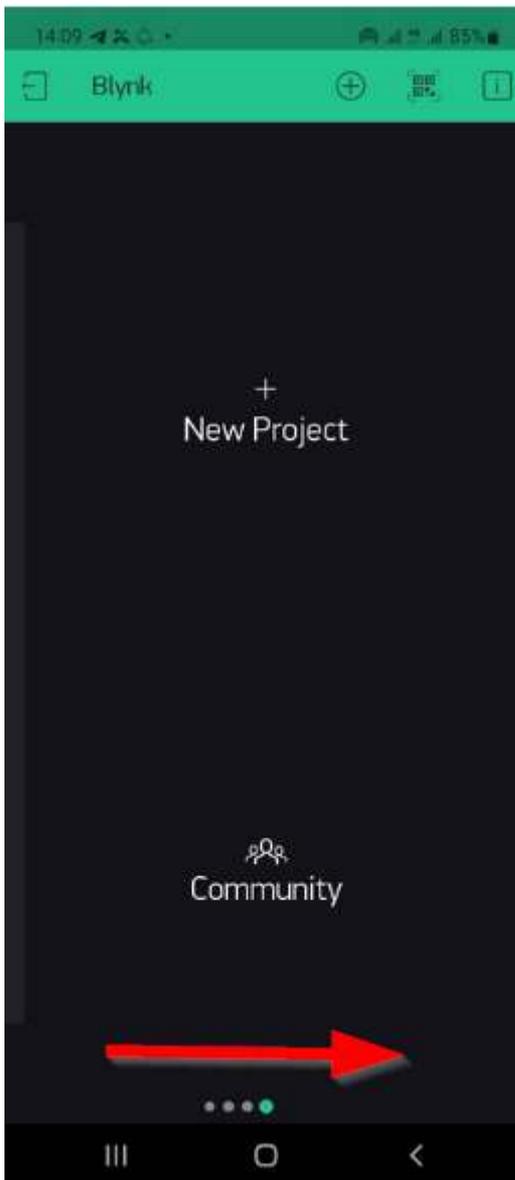
IMPORTANTE: Entra en **CUSTOM** y pon **LA DIRECCIÓN IP DEL SERVIDOR BLYNK LEGACY**

- Si es dentro del curso de CATEDU habrás recibido un email de las IPs y de los puertos que hay que poner en esta APP de Blynk Legacy
- Si es fuera del curso, la IP DE LA RASPBERRY, O SEA DEL SERVIDOR LOCAL BLYNK y puerto 9443

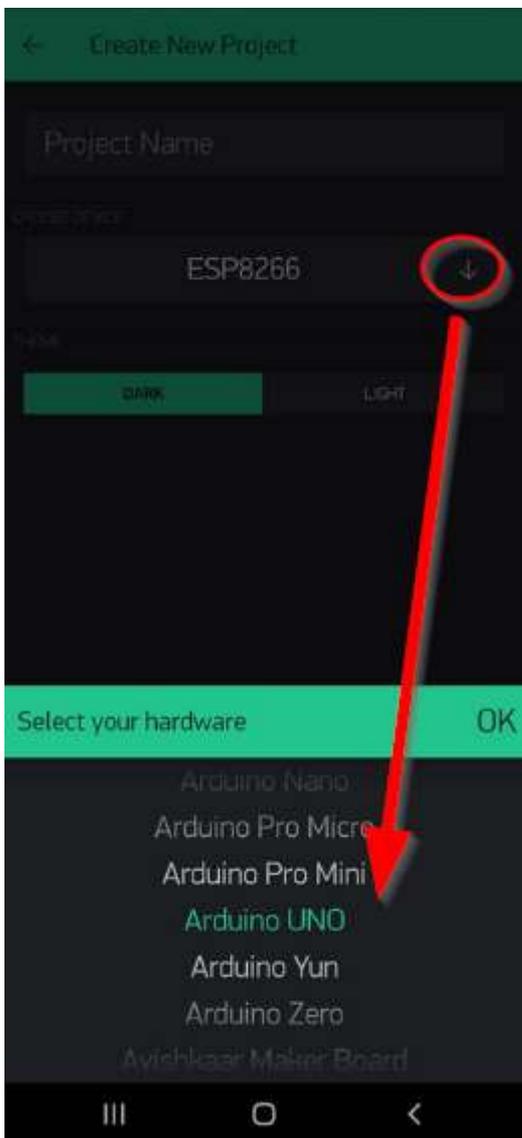


Crear un nuevo proyecto

Podemos ver los proyectos que hay ya almacenados, vamos a crear un nuevo proyecto:

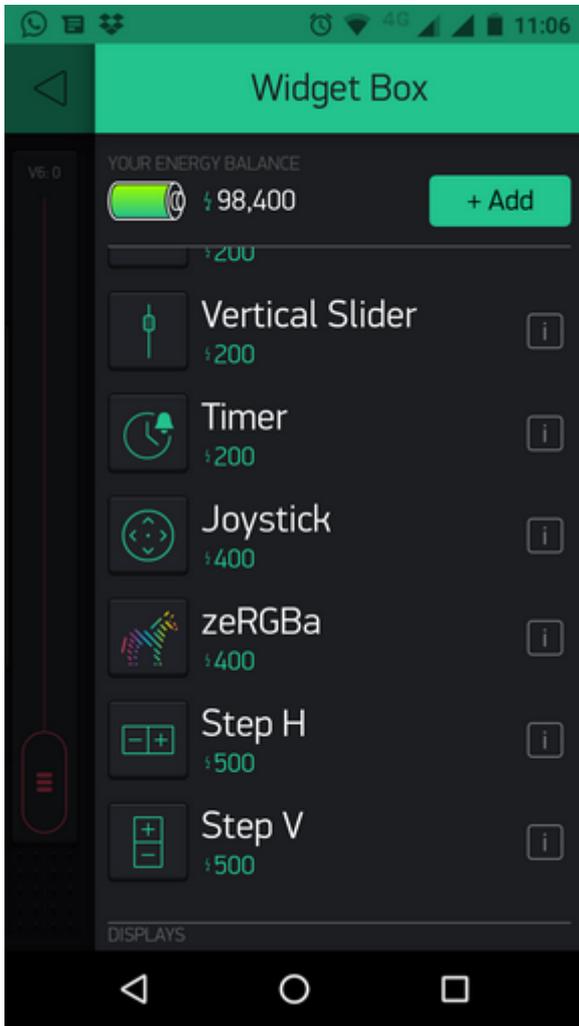


Nos preguntará por el **DEVICE**:



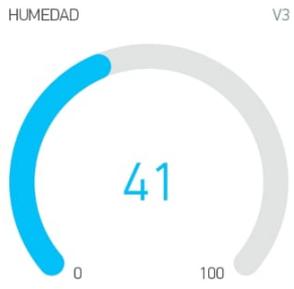
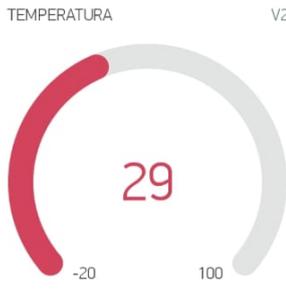
- En **Arduinoblocks en el Aula**, la placa es el TDR STEAM colocado en un Arduino con el ESP01 conectado, luego **DEVICE Arduino Uno CONNECTION TYPE Wifi**
- En **Rover con Arduino**, la placa es un NodeMCU v2, luego **DEVICE NodeMCU CONNECTION TYPE Wifi**
- En **ESP32 en el Aula**, la placa es un ESP32 con el Imagina TDR STEAM, luego **DEVICE ESP32 DevBoard CONNECTION TYPE Wifi**

Empieza un nuevo proyecto, tienes un montón de energía y widgets para hacerlo:



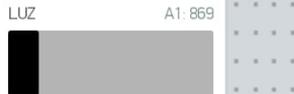
Crea un nuevo proyecto, y así se crea en el servidor Blynk local un **TOKEN** que ahora veremos qué es

Aquí tienes la captura de un proyecto o dashboard creado para el kit Imagina TDR Steam



TEMPERATURA
V7: 31...

POTENCIOMETRO
A0: 326



V8: 02...

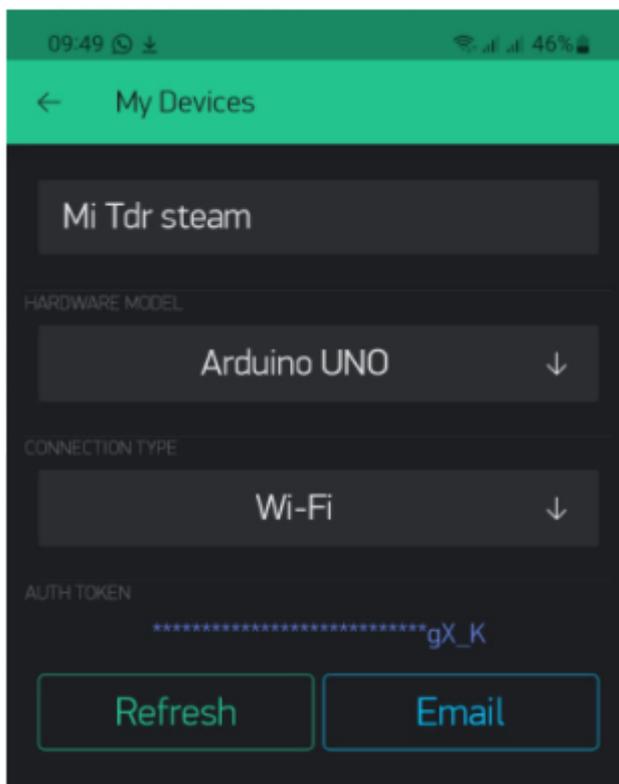


Poner el TOKEN en ARDUINOBLOCKS

¿Cómo conseguir el TOKEN?

OPCION A: QUE LO HAGA EL ALUMNO En la APP

Es la opción más sencilla, entrar en Devices **pulsar en el Token y se copiará en el portapapeles** del móvil, luego ir a cualquier aplicación para poder enviarla al PC (email, whatsapp...) y pegarla en Arduinoblocks en la instrucción que luego veremos.



OPCION B: QUE LO HAGA EL PROFESOR En el panel de control

Entramos en el panel de control del servidor Blynk y podemos ver en los usuarios creados los tokens

Blynk Administration

- Users
- Stats
- Hardware Info
- Config

Users list

<input type="checkbox"/>	Email
<input type="checkbox"/>	user1@yo.es
<input type="checkbox"/>	user2@yo.es
<input type="checkbox"/>	admin@blynk.cc

Entrando en un usuario PODEMOS VER SUS PROYECTOS Y LOS TOKENS

Devices

Id

Name

BoardTy

Token JGnbIYNVLfbdgqNvHe3xK1LyJV0QDiE9

LastLoggedIP 192.168.1.131

Connect

RECUERDA: UN PROYECTO UN TOKEN

¿Qué se hace con ese TOKEN? Se pone en ARDUINOBLOCKS:

Entramos en ArduinoBlocks y vamos al bloque de comunicaciones IoT - Blynk legacy

ATENCIÓN: ARDUINOBLOCKS CONNECTOR tiene que estar actualizado, al menos v5 (octubre 2022)

Tenemos que poner los datos de la **IP servidor BLYNK LOCAL**, y el **TOKEN** del proyecto

- **DENTRO DEL CURSO CATEDU** Habrás recibido un email con las IPs y los puertos a poner en Arduinoblocks
- **SI ES EN TU SERVIDOR BLYNK LOCAL** Hay que poner la IP de la Raspberry o tu servidor local y **EL PUERTO 8080**

En el caso de que es un proyecto con **ARDUINO** o **TDR-STEAM- ARDUINO** con **ESP01** sale :



En el caso de sea un proyecto con **NodeMCU** o **ESP32** como en el curso del Rover con Arduino o IoT en el aula sale :



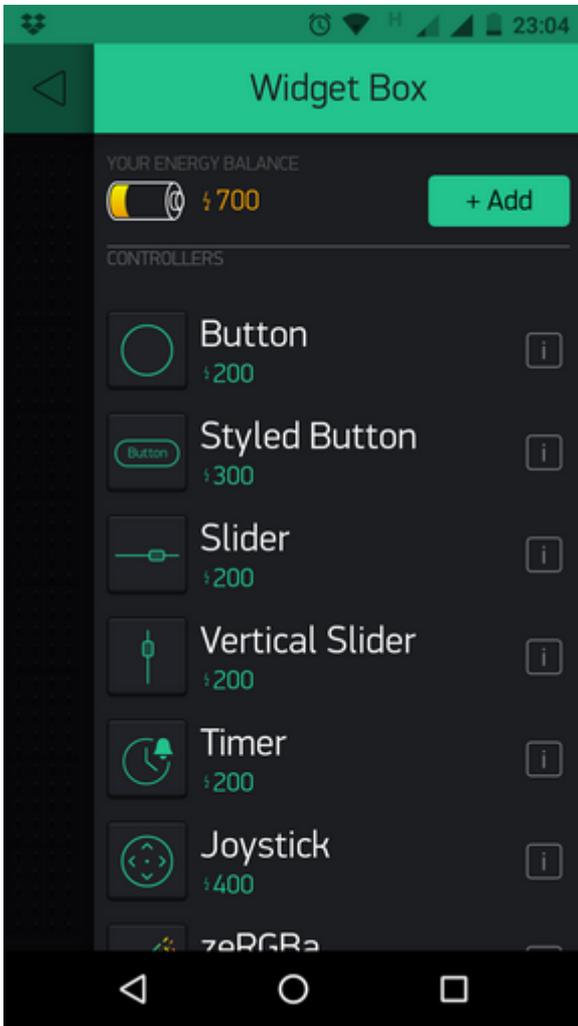
Primer programa: Encender los pines rojo y verde

Vamos a ver un primer programa de la APP al TDR STEAM

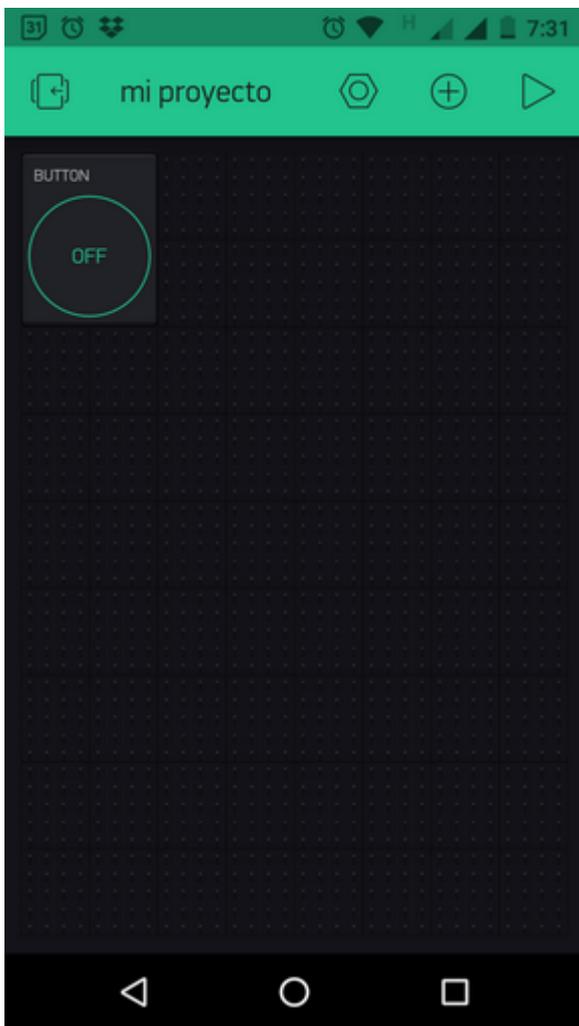
Una vez dentro del proyecto de la App aparece todo un panel si nada, añadimos controles apretando al + que hay en la parte superior



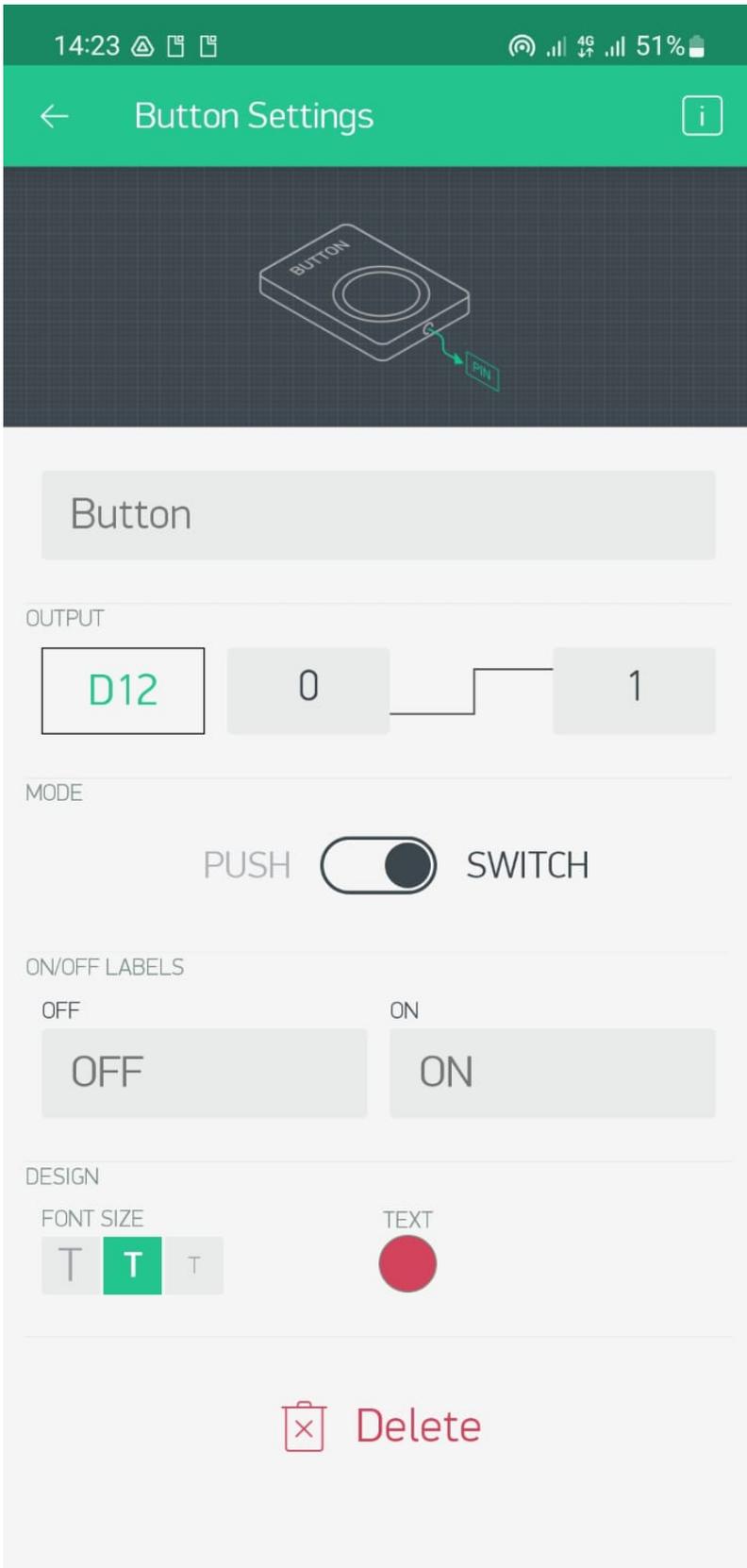
Seleccionamos el botón



Nos aparece el botón



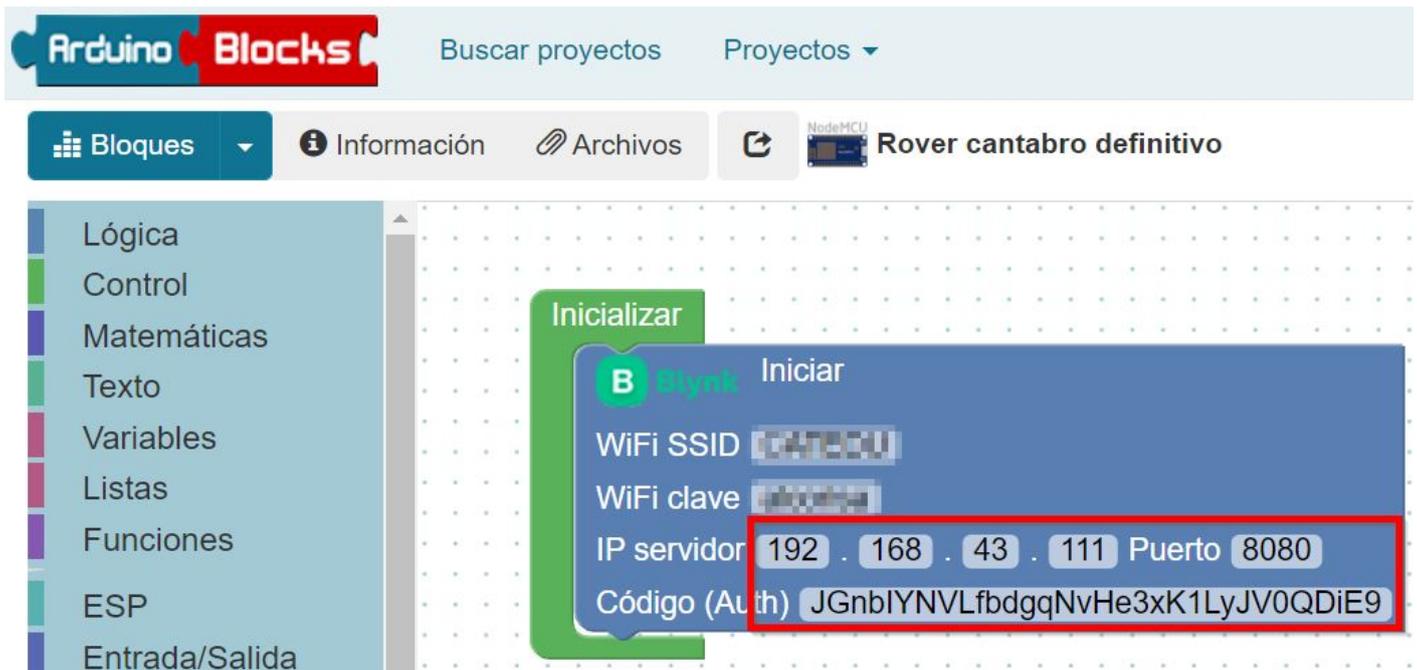
Pulsamos sobre él para entrar en sus propiedades, le decimos que vaya de 0 a 1 (es decir que cuando se apriete, que envíe un 1) que sea tipo **switch** (no pulsador, sino interruptor), le ponemos un color **rojo** para indicar que es el led rojo, y como el TDR-STEAM el led rojo lo tiene en el D12 le decimos que el **pin** es **DIGITAL** y **D12**



Hacemos lo mismo con el azul pero **D13**

En ARDUINO BLOCKS

En Arduinoblocks dejamos el token tal y como hemos aprendido anteriormente



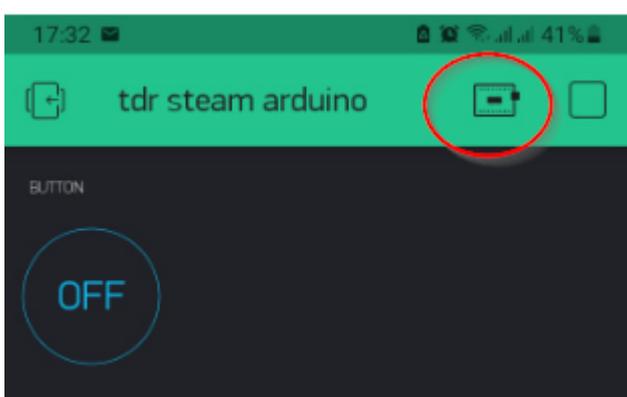
¿Nada más?



Nada más !! así de sencillo. Le damos a **subir** (teniendo el programa Arduinoblocks conector minimizado, eso lo podemos ver enseguida pues detecta en que COM está conectado, en la figura sale COM5)



Vamos a la APP de BLYNK, Pulsamos al botón de **play** ▶ de arriba arriba y vemos el estado NO TIENE QUE APARECER UN PUNTO ROJO si aparece es que nuestra placa no está conectada



Hay que tener algo de paciencia hasta que se comunica con la Raspberry, cuando no aparece el punto rojo, pulsamos en los botones y tienen que encenderse y apagarse correctamente los leds rojo y azul de nuestro TDR Steam

Para usuarios del servidor Blynk de CATEDU y la placa ARDUINO CON ESP32 WIFI

No sabemos por qué pero al darle al Play, el icono de la placa de arriba sale en rojo, como si no estuviera conectado, pero el proyecto funciona perfectamente. Si sabes por qué por favor ponte en contacto con nosotros <https://catedu.es/informacion/>



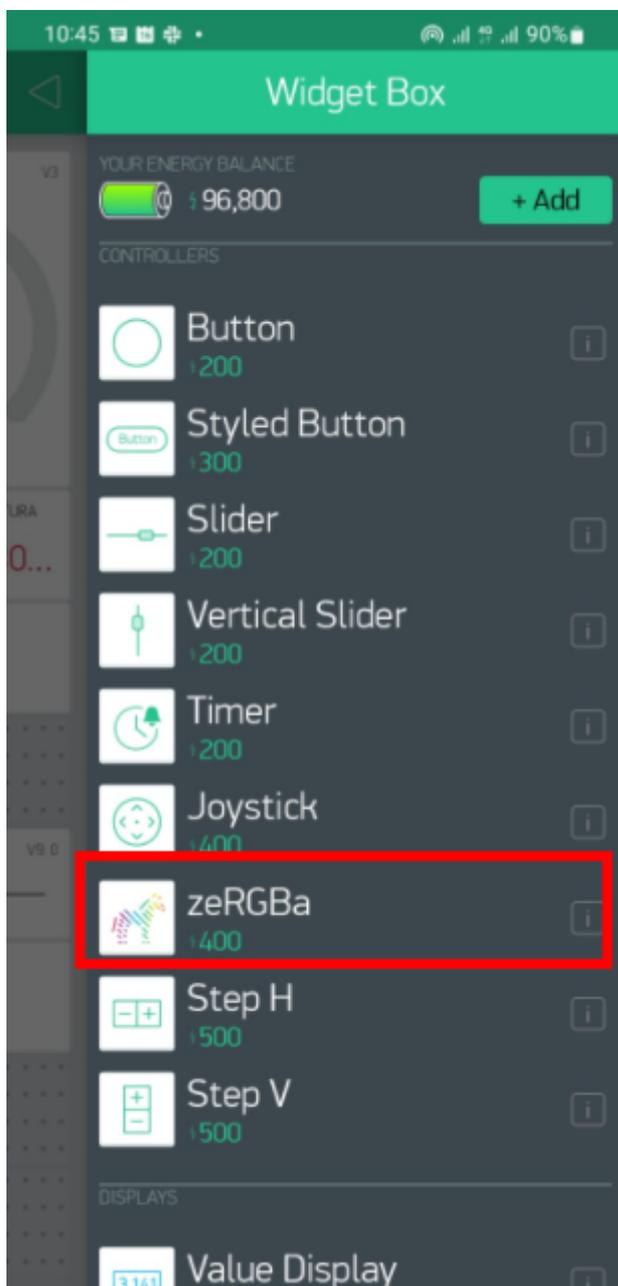
Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



Segundo programa RGB y el potenciómetro

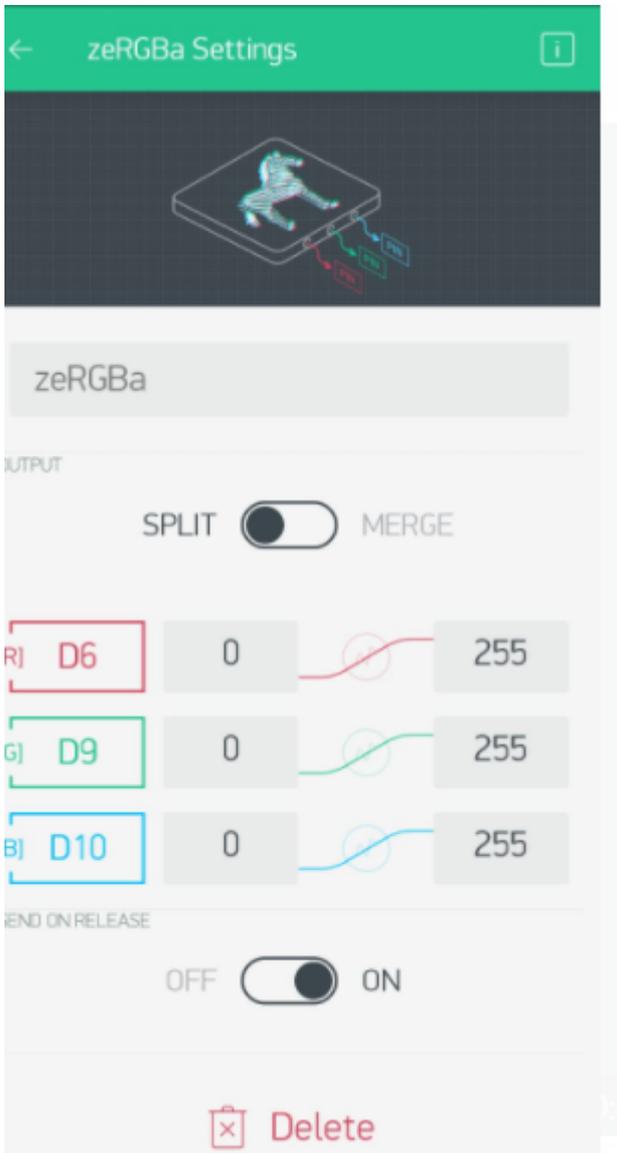
LED RGB

En la APP de Blynk ponemos el gadget "Cebra"



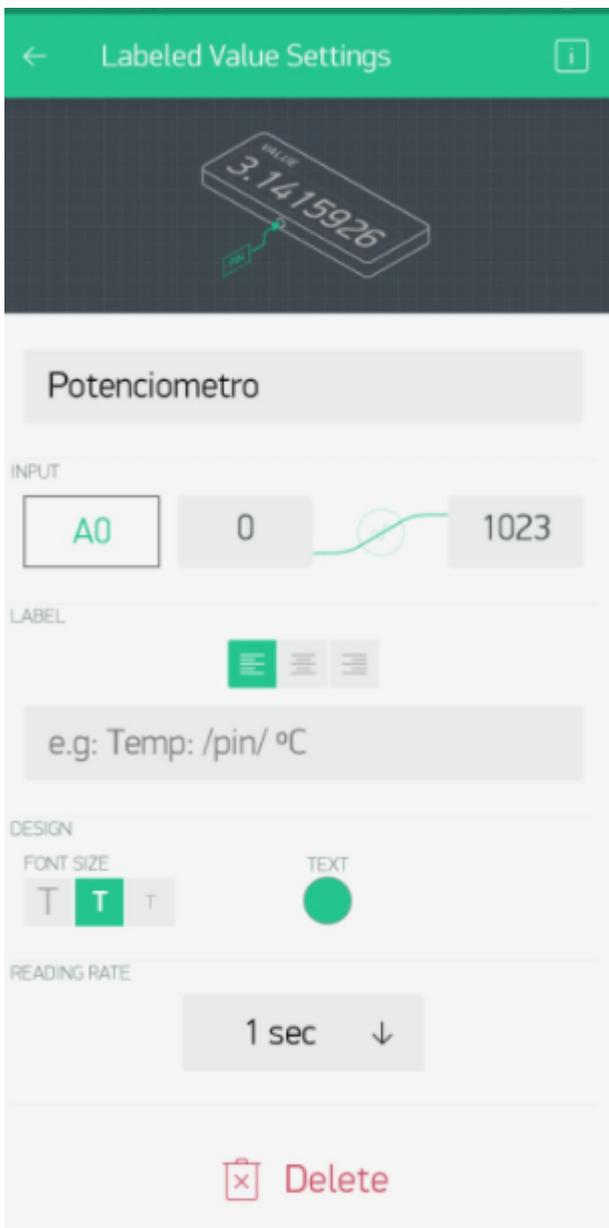
Y lo configuramos según sus pines. El led RGB en el TDR Steam está conectado a los siguientes pines

- Red D6
- Green D9
- Blue D10



Potenciómetro

El potenciómetro está conectado en A0 podemos insertar un gadget Label y lo configuramos como tal:



¿En Arduinoblocks no hay que añadir nada?

Nada !!! cuando son gadgets que leen directamente de los pines (digitales o analógicos) NO HACE FALTA MÁS CÓDIGO que el de conectar el servidor Blynk

Arduino Blocks Buscar proyectos Proyectos ▾

Bloques ▾ Información Archivos  Rover cantabro definitivo

- Lógica
- Control
- Matemáticas
- Texto
- Variables
- Listas
- Funciones
- ESP
- Entrada/Salida

Inicializar

B Blynk Iniciar

WiFi SSID GATECOU

WiFi clave 12345678

IP servidor 192 . 168 . 43 . 111 Puerto 8080

Código (Auth) JGnbiYNVLFbdgqNvHe3xK1LyJV0QDiE9

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU

 **Financiado por la Unión Europea**
NextGenerationEU

 **Plan de Recuperación, Transformación y Resiliencia**

 **GOBIERNO DE ESPAÑA**

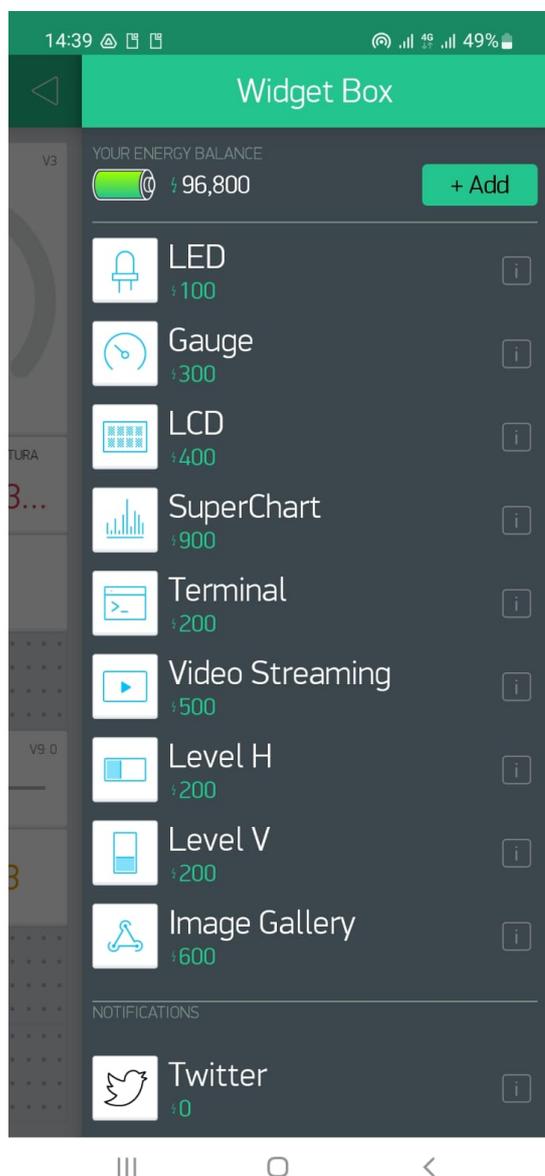
 **MINISTERIO DE EDUCACIÓN Y FORMACIÓN PROFESIONAL**

 **GOBIERNO DE ARAGON**

Tercer programa Medir la luz del LDR

Nuestro segundo programa será al revés del TDR Steam a la APP

Ahora añadimos otro control **level H**



Elegimos de PIN el analógico, el LDR está conectado al pin analógico A1 y como el LDR aumenta según la *oscuridad* vamos a poner que vaya de 1023 en formato negro.



Sin modificar nada del programa de ARDUINOBLOCKS vamos a darle al PLAY y ¡¡ Y FUNCIONA !!!

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



Cuarto programa: Medir Temperatura y Humedad con el DHT11

Este caso es distinto, pues

- En el **Rover marciano con Arduino** el DHT22 está conectado al pin D5 digital del NodeMCU
- En el **Arduinobloks en el aula**, el DHT11 de la placa Imagina TDR Steam está conectado al pin D4 digital del Arduino
- En el **IoT en el Aula** el DHT11 de la placa Imagina TDR Steam está conectado al pin D4 digital del ESP32

Pero los dos miden dos variables **Temperatura Y Humedad**

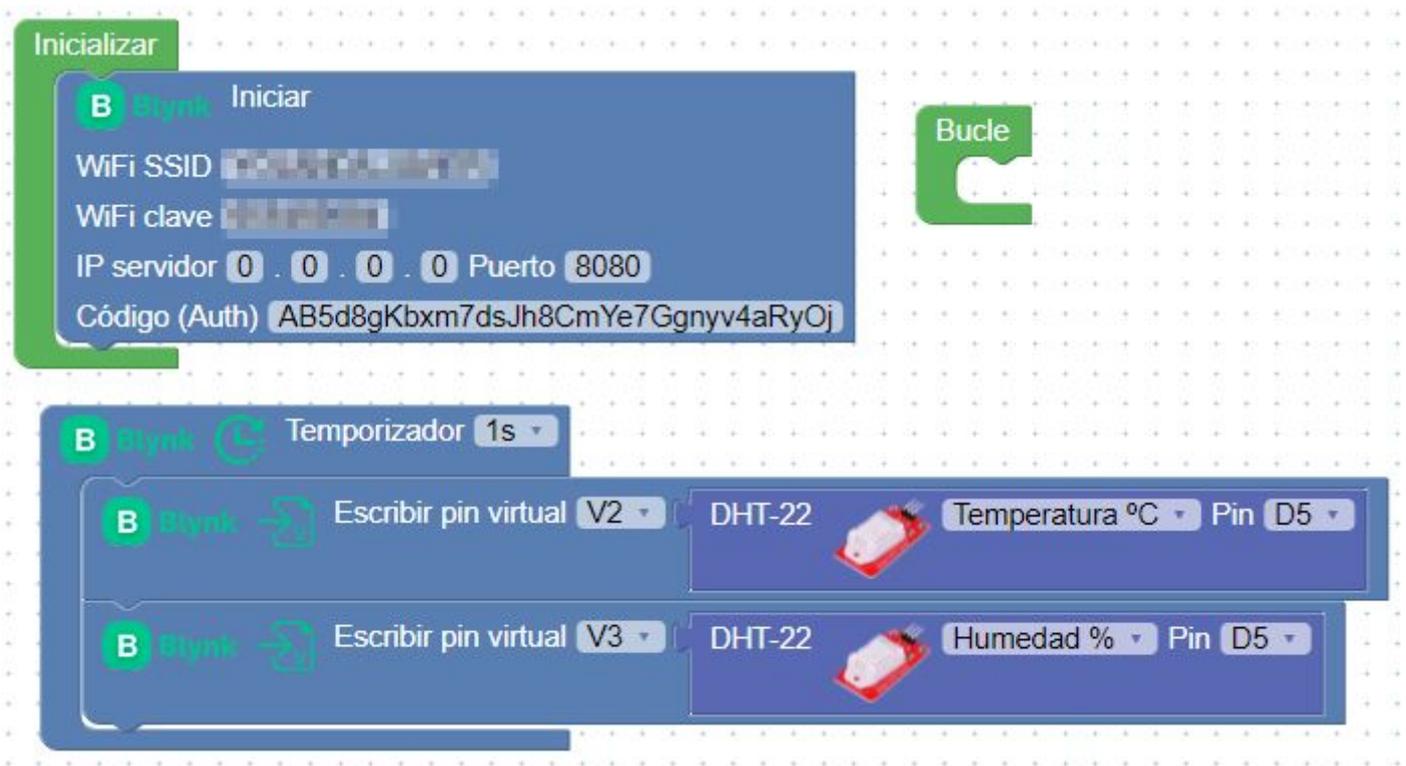
□□□□¿Cómo hacemos para medir dos variables en un sensor que está conectado a un sólo PIN?□□□□

Solución PINES VIRTUALES: si, has leído bien, vamos a utilizar **pinos virtuales**.

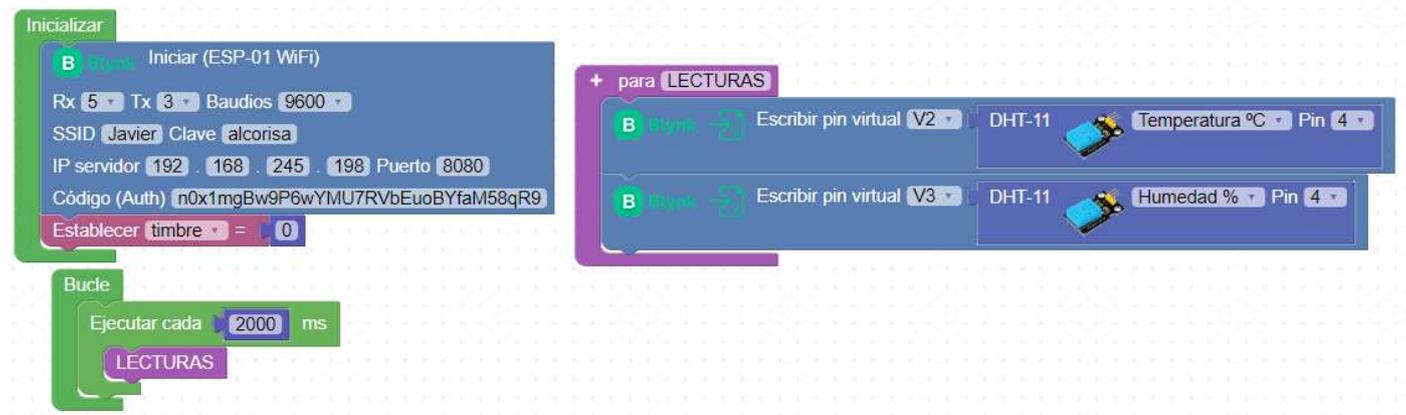
En ARDUINOBLOCKS

Vamos a ARDUINOBLOCKS y establecemos dos pines virtuales, uno para la temperatura y otro para la humedad, arbitrariamente he puesto V2 y V3 pero puede ser cualquiera.

- Para el kit de **ROVER MARCIANO CON ARDUINO** poner pin **D5**
 - Si el sensor es AZUL entonces DHT11 si el sensor es BLANCO entonces DHT22



- Para el kit que tenga la placa **TDR STEAM IMAGINA** es decir los cursos **ARDUINO BLOCKS EN EL AULA** y **ESP32 EN EL AULA** poner pin **D4**
 - Si el sensor es AZUL entonces DHT11 si el sensor es BLANCO entonces DHT22



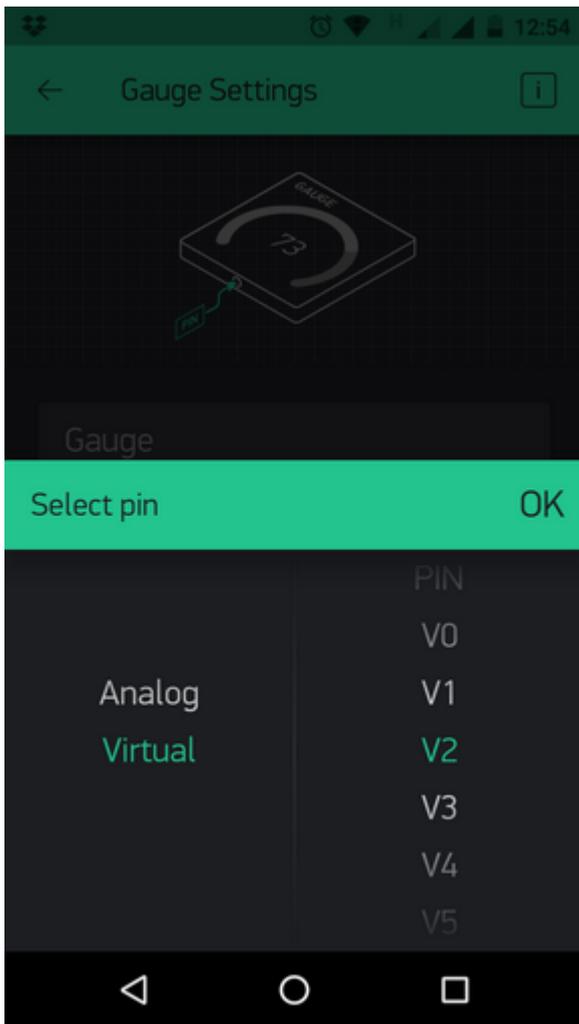
Fíjate que hay dos maneras de programar estos eventos:

- en el primero, utilizamos un bucle propio de Blynk que simplemente cada segundo lee el sensor y los almacenan los pines virtuales V2 y V3
- en el segundo no se ha utilizado ese bucle propio de Blynk sino simplemente dentro de Bucle hemos puesto otro bucle que se ejecuta cada 2 segundos, y dentro una función que se llama LECTURAS. dentro de lecturas esta la lectura de los pines virtuales V2 y V3

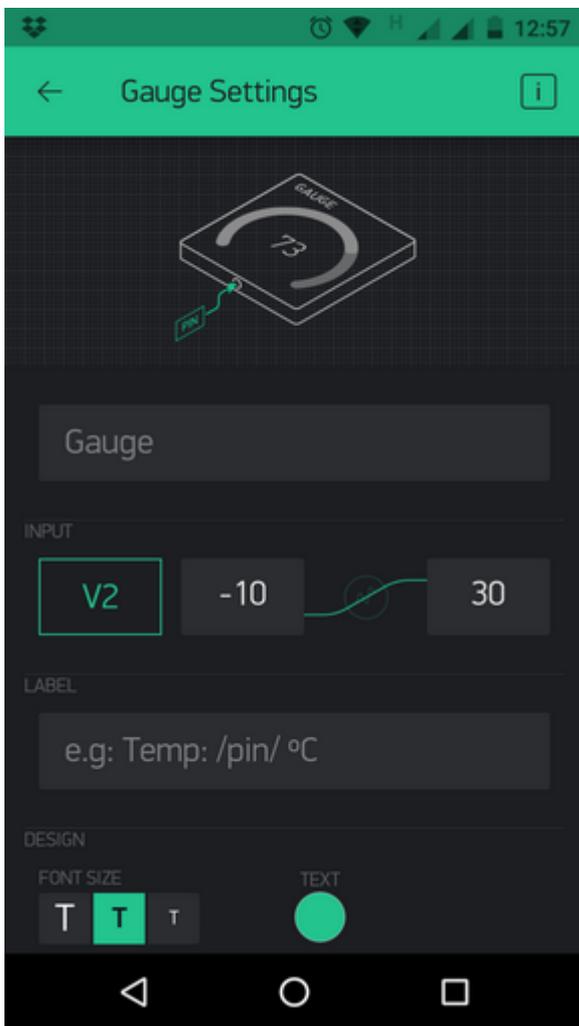
Da igual. Nosotros tenemos preferencia por el segundo método, nos parece más elegante y más controlable.

En la APP de Blynk

Y en Blynk incorporamos un Gauge que sea al PIN VIRTUAL V2.



Modificamos también los límites, pues por defecto sale 0 a 1023 y se vería muy bajo la temperatura, ponemos -10 a 30. Si es para el Rover marciano de Arduino, allí se puede llegar a -50°C y si es para el TDR Steam será en Aragón, el récord lo tiene [Fuentes Claras con -30°C](#).



Para la humedad hacemos lo mismo:

- Pin virtual V3
- Límites 0% a 100%

Resultado :

HUMEDAD



TEMPERATURA



Quinto programa: Lectura sensor LM35 y receptor infrarrojos

Podríamos hacer una lectura de estos sensores, sin necesidad de tratamiento ninguno, tal y como hemos hecho en la lectura de la luz con LDR, simplemente se añade un gadget a la APP de Blynk que :

- Lea el pin D11 que es donde está el sensor de infrarrojos
- Lea el pin A2 que es donde está el sensor de temperaturas LM35

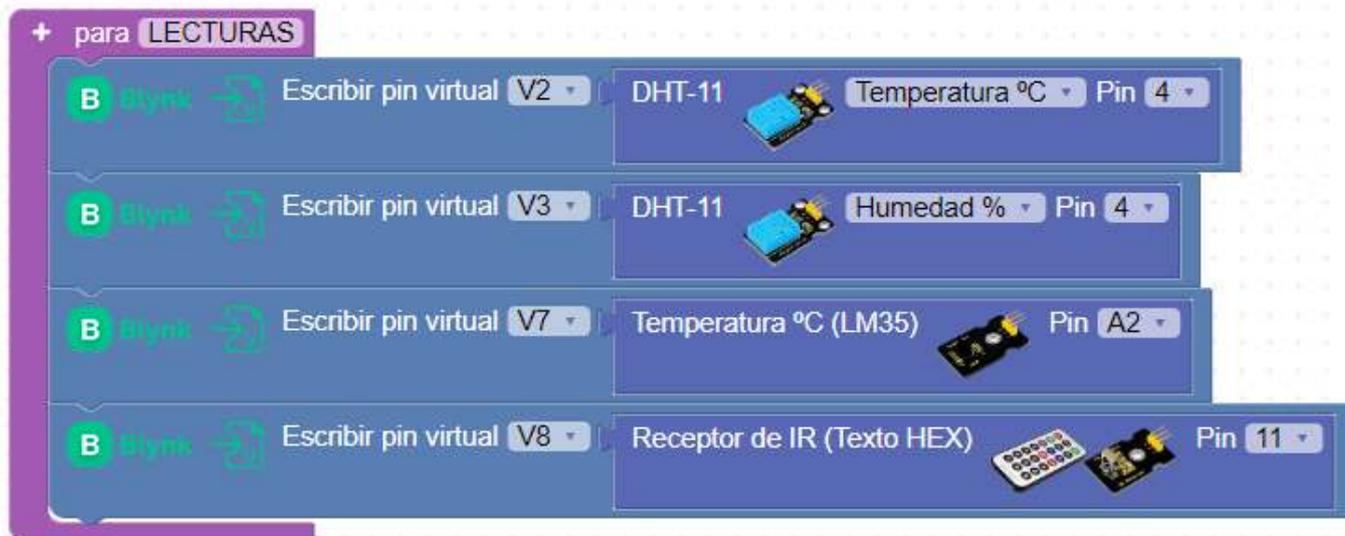
Los resultados serían totalmente incorrectos !!!

- El sensor de infrarrojos lanza un código que al leerlo el pin D11 en la APP aparecería encendidos y apagados sin poder leer qué código es lo que dice
- En el sensor de temperaturas, es un sensor conectado al A2 por lo que en la APP mediría desde 0 hasta 1023, y esto no sólo habría que mapearlo a temperaturas, sino tratarlo correctamente pues el LM35 da lecturas de dos decimales.

En **Arduinoblocks** tenemos dos bloques específicos para tratar estas lecturas, las llevamos a **pinos virtuales** y que los gadgets de la APP visualicen los valores de estos pines virtuales. Otro uso de los pines virtuales.

En Arduinoblocks

Ponemos dentro de la función LECTURAS los siguientes dos bloques :



Como vemos, Arduinoblocks procesa la lectura de estos dos sensores, y simplemente se almacenan en los pines virtuales V7 y V8

En la APP de Blynk

Ponemos para la lectura de la temperatura, un display que visualice el número V7



Para el sensor de infrarrojos igual pero que visualice V8



Lo **subimos** el programa Arduinoblocks al Arduino TDR Steam, pulsamos el **play** en la APP, esperamos a que se conecte y el sensor de temperatura muestra su valor perfectamente

Curiosamente dan unas lecturas algo diferentes el DHT11 y el LM35, el correcto es el LM35 pues el DHT11 no es un sensor muy preciso.

Para ver los códigos que se leen en Infrarrojos, hay que pulsar los números del mando a distancia.

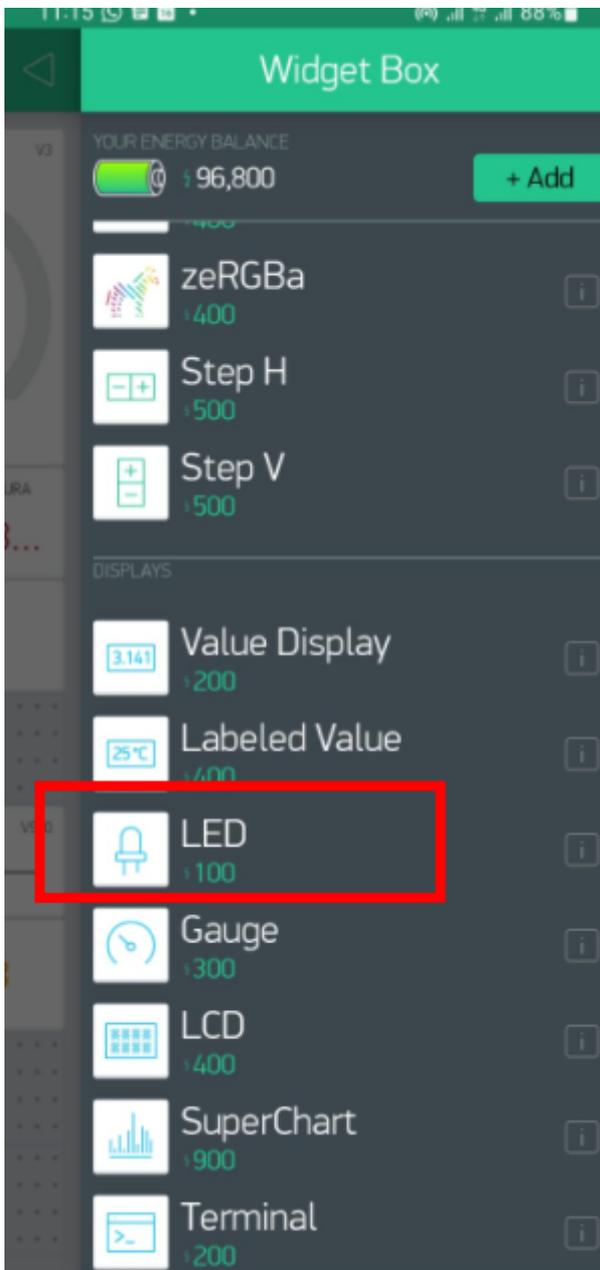


Sexto programa Leer eventos del TDR Steam

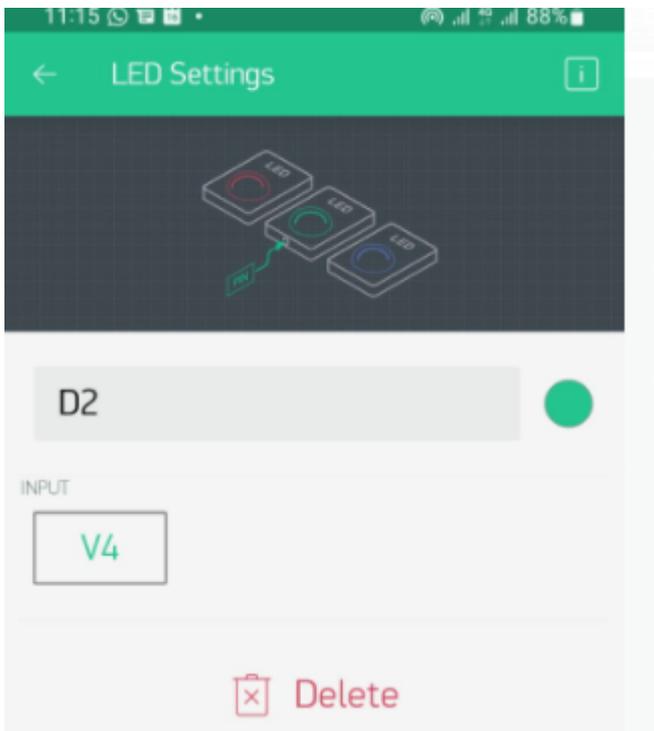
Hasta ahora hemos enviado eventos al TDR Steam, y leer sensores del TDR Steam pero ¿y leer un evento? por ejemplo leer si se pulsa los pulsadores D2 y D7

En la APP de Blynk

Añadimos un gadget que se llama LED



y leerá de un pin virtual pin V4 y otro gadget para V5



En ARDUINOBLOCKS

Pondremos el siguiente código, simplemente lee que si se pulsa pues que encienda el led en el APP

The image shows a Scratch script for a Blynk IoT project. The script is organized into two main sections:

- Top Section (Loop):** A purple loop block labeled "para LECTURAS" contains four "Escribir pin virtual" blocks:
 - Block 1: "Escribir pin virtual V2" with "DHT-11" sensor, "Temperatura °C" unit, and "Pin 4".
 - Block 2: "Escribir pin virtual V3" with "DHT-11" sensor, "Humedad %" unit, and "Pin 4".
 - Block 3: "Escribir pin virtual V7" with "Temperatura °C (LM35)" sensor and "Pin A2".
 - Block 4: "Escribir pin virtual V8" with "Receptor de IR (Texto HEX)" sensor and "Pin 11".
- Bottom Section (Conditional Logic):** A "si" block with two digital input checks and corresponding LED actions:
 - Condition 1: "Leer digital Pin 2". If true, "Led V4" is turned "ON". If false, "Led V4" is turned "OFF".
 - Condition 2: "Leer digital Pin 7". If true, "Led V5" is turned "ON". If false, "Led V5" is turned "OFF".

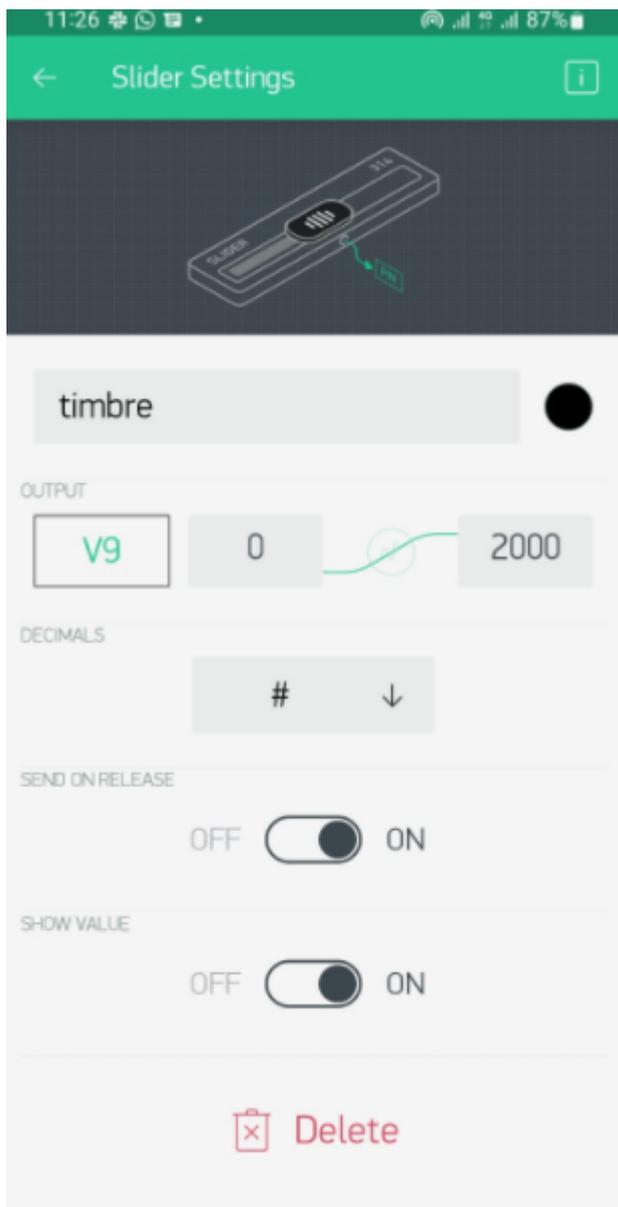
Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



Séptimo programa: El timbre

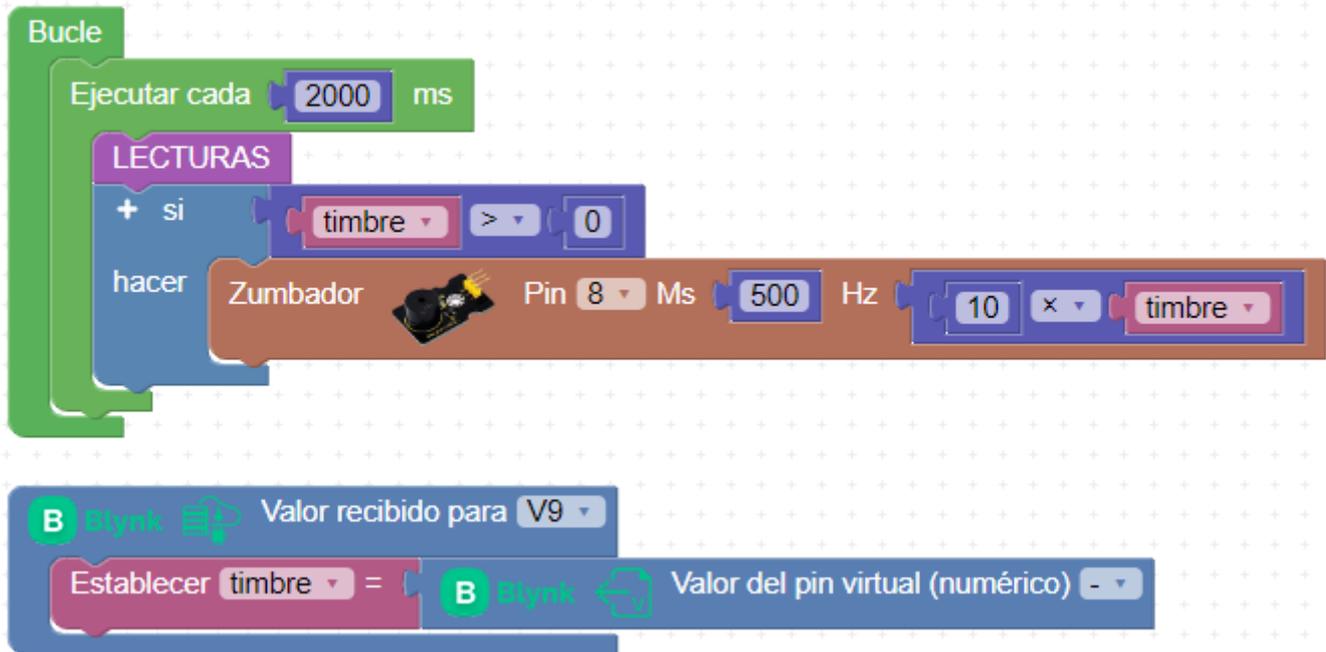
El timbre es algo especial, si ponemos una música, el Arduino está tan ocupado y el código engorda que da muchas desconexiones, si es el ESP32 lo aguanta bien.

En la APP ponemos un slider asociado a un pin virtual, por ejemplo a V9



Y en Arduinoblocks añadimos un bucle que lea ese pin virtual V9 y que lo almacene en una variable numérica, que lo hemos llamado *timbre*

Si esta variable, ha cambiado el valor, que haga un tono. si te fijas el valor máximo del slider se ha puesto a 2000, el máximo tono audible es 20.000, pero el slider no lo permite, por lo tanto, el truco es multiplicarlo por 10



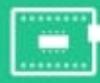
Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU



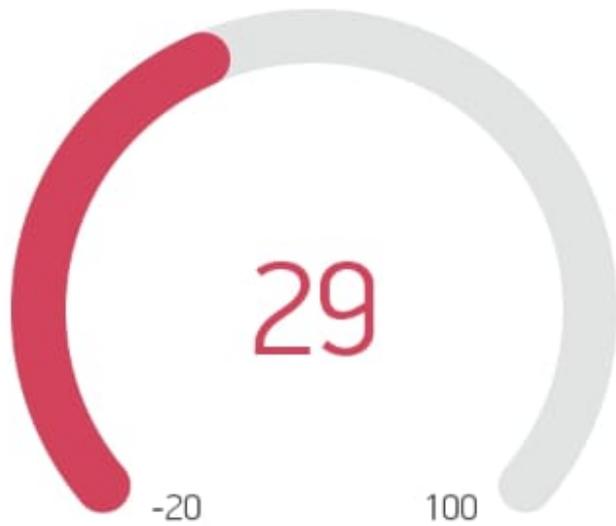
¿Todo junto?

Si que se puede pero con la placa Arduino alguna vez se desconecta, pues no tiene un microprocesador muy potente. Con ESP32 sí que lo aguanta bien.

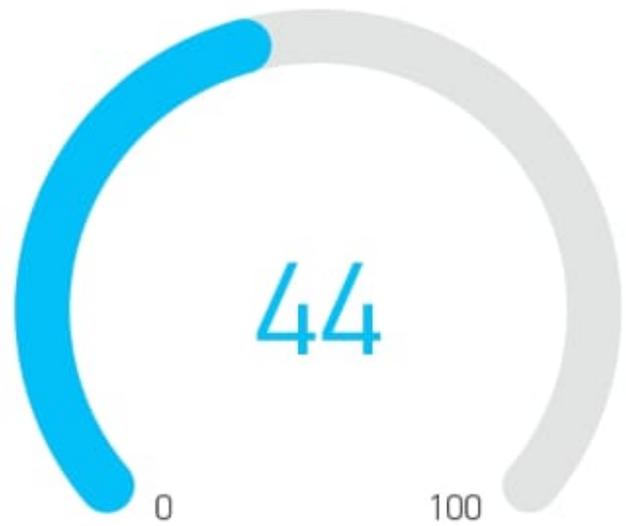
En la APP el dashboard o proyecto queda:



TEMPERATURA



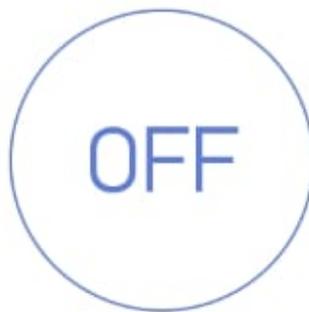
HUMEDAD



BUTTON



BUTTON



D2



D7



TEMPERATURA

32.26

POTENCIOMETRO

327

ZERGBA

251 0 0



LUZ

821



TIMBRE

0



Y en ARDUINOBLOCKS

The image displays an Arduino Blocks program with the following structure:

- Inicializar:** A block to start the ESP-01 WiFi module with the following configuration:
 - Rx: 5, Tx: 3, Baudios: 9600
 - SSID: Javier, Clave: alcorisa
 - IP servidor: 192.168.245.198, Puerto: 8080
 - Código (Auth): n0x1mgBw9P6wYMU7RVbEuoBYfaM58qR9
 - Establecer timbre = 0
- Bucle:** A loop that executes every 2000 ms.
 - Inside the loop, there is a 'LECTURAS' block.
 - A 'si' (if) block checks if 'timbre' is greater than 0.
 - If true, a 'hacer' (do) block triggers a buzzer on Pin 8 with a frequency of 500 Hz and a duration of 10 ms.
 - Below the loop, a block reads the value received for V9 and sets 'timbre' to the value of the virtual pin (numerical).
- LECTURAS (for loop):** A series of blocks to read and act on virtual pins:
 - Escribir pin virtual V2: DHT-11 sensor, Temperatura °C, Pin 4.
 - Escribir pin virtual V3: DHT-11 sensor, Humedad %, Pin 4.
 - Escribir pin virtual V7: Temperatura °C (LM35), Pin A2.
 - Escribir pin virtual V6: Receptor de IR (Texto HEX), Pin 11.
 - si Leer digital Pin 2: If ON, Led V4 ON; If OFF, Led V4 OFF.
 - si Leer digital Pin 7: If ON, Led V5 ON; If OFF, Led V5 OFF.

Financiado por el Ministerio de Educación y Formación Profesional y por la Unión Europea - NextGenerationEU

