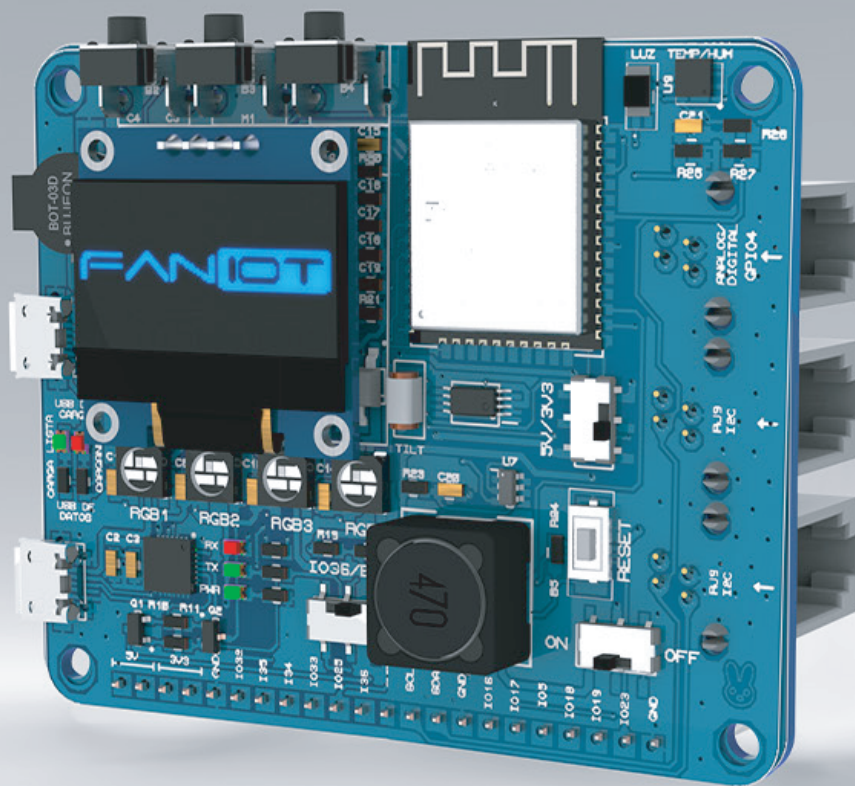


# Guía Rápida

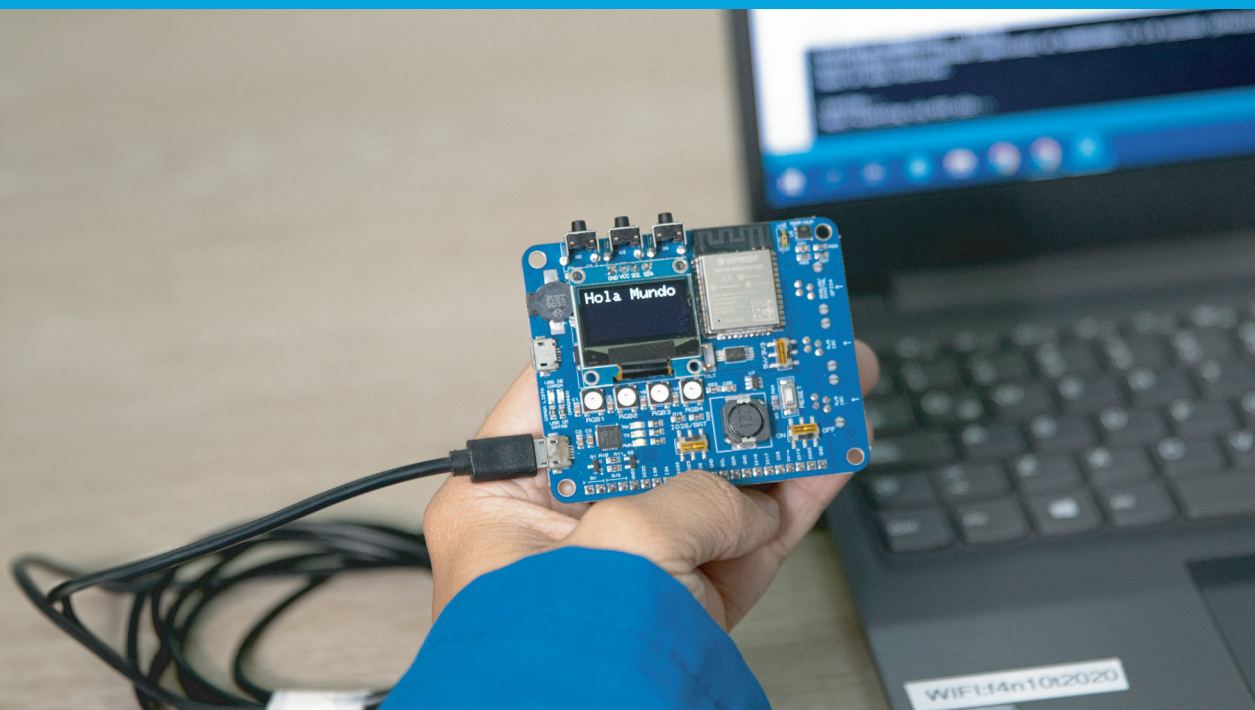
## Kit Maker 2.0 IOT



Lea determinadamente esta guía antes de usar el producto y consévelo para futuras consultas.

FANIOT

3	Descripción general Kit Maker 2.0
4	Componentes
5	Diagrama de pines
6	FANHUB
7	Lenguajes de programación
10	Pasos para iniciar
15	Como iniciar en ArduinoBlocks
20	Como instalar en ArduinoBlocks
24	Instalar Python
28	Como crear un Nuevo Proyecto en Blynk
30	Glosario



# Descripción general

## Kit Maker 2.0

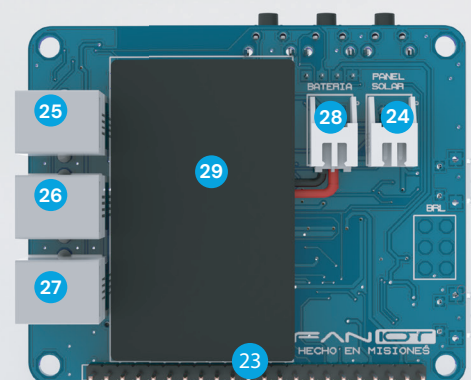
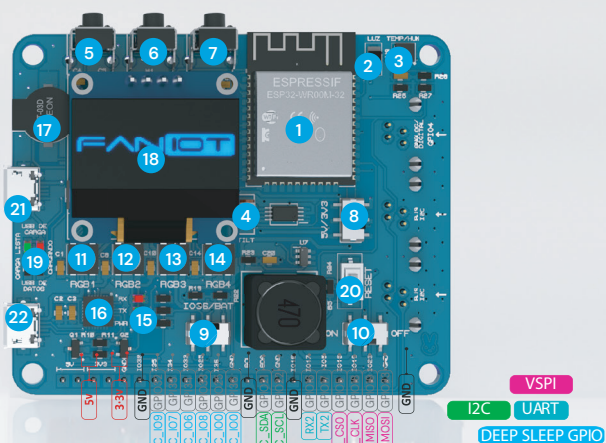
El Módulo Base posee un microcontrolador con conexión WiFi y Bluetooth, 20 entradas/salidas digitales, de las cuales 10 se pueden utilizar como entradas analógicas.

### Especificaciones técnicas

Microcontrolador	—————	ESP32
Reloj	—————	240MHz
Conectividad	—————	WiFi (802.11 b/g/n) Bluetooth v4.2
Alcance WiFi	—————	30m interiores – 90m aire libre
Voltaje de Operaciones	—————	3.3V
Voltaje de Alimentación	—————	5V
Pines E\S Digitales	—————	8
Entradas Analógicas	—————	4
Conectores RJ11	—————	3

# Componetes

- 1 ESP32: Microcontrolador, el cerebro de la placa Kitmaker2.0
- 2 Sensor de luminosidad TEMT6000: GPIO39
- 3 Sensor de temperatura\humedad HYU21d: Protocolo I2C
- 4 Sensor de vibracion TILT: GPIO14
- 5 6 7 Botonos Programables: GPIO0, GPIO15, GPIO13
- 8 Selección del voltaje para el conector RJ9: V5\ 3V3
- 9 Switch para medir el nivel de bateria
- 10 ON\OFF Switches (bateria conectada)
- 11 12 13 14 Neopixels rgb programables: GPIO27
- 15 Programacion de indicadores LED (Rx\Tx), encendido (PWR).
- 16 Convertidor de USB a TTL CP2102
- 17 Buzzer pasivo programable: GPIO12
- 18 Pantalla OLED 128x64 pixels: protocolo I2C
- 19 Indicadores de carga de bateria
- 20 Boton Reset
- 21 Entrad MicroUSB de carga
- 22 Entrada MicroUSB de programacion
- 23 Pines.
- 24 Conector JSP para panel solar.
- 25 GPIO4 Analog\Digi
- 26 I2C
- 27 I2C
- 28 Conector JST para bateria
- 29 Bateria Li-Po 3.7V 700mAh



# Diagrama de pines

Un diagrama de pines es un esquema que representa los pines de una placa de desarrollo, indicando sus tipos y funciones. Cada Pin posee un número identificatorio para utilizarlo dentro del código. Se utilizan las letras para indicar si es una salida digital (D) o analógica (A) respectivamente.

El Kitmaker 2.0 cuenta con 22 pines through-hole en una sola tira, facilitando su uso en protoboard para la conexión de sensores y componentes. También incluye tres puertos RJ9: dos para dispositivos I2C y uno conectado al pin GPIO4.

## Tipos de Pines Disponibles

### Pines GPIO (General Purpose Input/Output)

Son pines que puedes configurar como **entrada** o **salida** según el dispositivo o sensor que conectes.

### Pines GPI

Estos pines, como el I34, solo funcionan como **entradas analógicas**. Son útiles para leer datos de sensores como potenciómetros o detectores de luz.

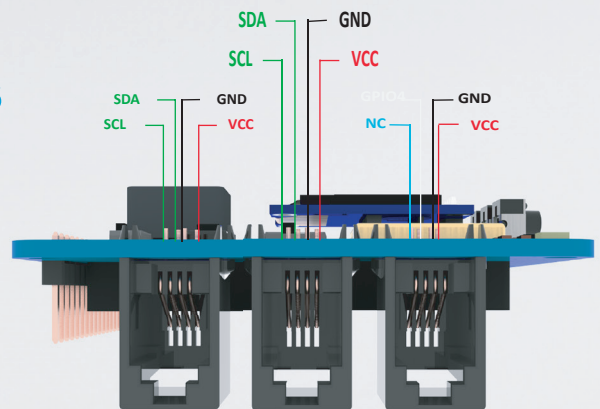
### Pines de Alimentación:

**5V y 3.3V:** Dos pines para cada uno. Se utilizan para alimentar los componentes conectados a la placa.

**GND (Tierra):** Cuatro pines conectados a la tierra del sistema, necesarios para completar el circuito de los dispositivos externos.

### Pines I2C (SDA y SCL)

Son los pines dedicados a dispositivos que utilizan el **protocolo I2C**, como pantallas OLED o sensores de temperatura. Estos pines permiten que varios dispositivos se conecten al ESP32 usando solo dos cables.



## Usos Creativos

Estos pines están disponibles para que los usuarios puedan **interactuar** con diversos componentes, como sensores, actuadores o displays. Al estar dispuestos en una fila compatible con protoboard, es muy fácil experimentar y crear nuevos proyectos con **Kitmaker 2.0**

Es un espacio **gratuito** de **aprendizaje virtual** diseñado para ofrecer a nuestros usuarios actividades y códigos modelos para facilitar el aprendizaje con nuestras **placas IoT programables**. Mejora significativamente los resultados de aprendizaje a partir de **ABP** y metodología **STEAM**.

## Diseño e impresión 3D

¡Personalizá tu KitMaker a partir de modelos **3D** pre diseñados! Descargalos desde FANHUB. [www.fanhub.com.ar](http://www.fanhub.com.ar)



### Lenguaje de programación

Opta por el lenguaje de programación que piensas utilizar.



### Descargar e instalar

Proporciona instrucciones para descargar e instalar el programa en diferentes sistemas operativos.



### Configuración inicial

Ayuda a los usuarios a configurar el programa después de la instalación.



### Recursos de aprendizaje

Ofrece recursos adicionales como tutoriales y documentación para ayudar a los usuarios a empezar.

# Lenguajes de programación

Se programa con el lenguaje “C” de manera gráfica si estás dando tus primeros pasos con ArduBlocks o en código si ya sos un programador avanzado con IDE de Arduino y Microphyton.

## Lee las instrucciones y asegúrate de tener todo listo antes de iniciar.

- Para avanzar con las actividades, es **imprescindible** haber leído la Guía Rápida de la Placa a utilizar.
- **No** se requiere ningún tipo de **componente** extra para estas actividades.
- Es necesario tener la **placa conectada** a su computadora, y haber seleccionado la placa correspondiente (ESP32), en la opción “herramientas” de IDE arduino.
- Los códigos están disponibles para descargar en un archivo de **bloc de notas**. Debes seleccionar todo el texto y pegarlo en el IDE de Arduino. Las instrucciones están ubicadas en las primeras líneas del código.
- Asegúrate de tener los **drivers** necesarios antes de continuar, si no lo tienes puedes descargarlo [AQUI](#), completa la instalación.



# +Actividades Fanhub

Explorando el Mundo de la **Electrónica IoT** y la **Programación**.  
Desarrollado 100% en Misiones–Argentina.

## Nivel 1 ACTIVIDAD N°1: ¡Hola Mundo!

¡Dale la bienvenida a la programación! Este primer ejercicio con nuestras placas IoT te permitirá **imprimir** un “Hola Mundo” en el Monitor Serie el paso inicial para hacer realidad todas tus ideas.

## Nivel 2 ACTIVIDAD N°9: Pantalla OLED

Te dejamos otro ejercicio, para implementarlo que hace lo siguiente: **Imprime textos y parámetros** de cualquier tipo, a través de la pantalla oled que incorpora la placa.

## Nivel 3 ACTIVIDAD N°17: BOT DE TELEGRAM

Esta guía te enseñará a crear un **bot de Telegram** utilizando BotFather. Un bot de Telegram puede automatizar tareas, responder a mensajes y mucho más.

En nuestra página web encontrarás **más ejercicios** disponibles para cada nivel de aprendizaje.



# Entornos de Programación

Un Entorno de Desarrollo Integrado (IDE), es un programa informático para la carga de las instrucciones compiladas, es decir traducidas a un lenguaje máquina, en la memoria del módulo.



# Pasos para iniciar

A

## Descarga e instalación

Accede al enlace, elige el sistema operativo, descarga el archivo e instálalo, el programa se abrirá automáticamente.

B

## Configuración del programa

Ve a Archivo - Preferencias, agrega la URL proporcionada, y descarga la placa ESP32 desde el gestor de placas.

C

## 3. Cargar y ejecutar actividad

Selecciona ESP32 Dev Module, descarga la actividad, copia el código al editor, conecta la placa, selecciona el puerto, carga el código y verifica la salida en el monitor serial.

¿No te gusta leer? No te preocupes, tenemos un **video tutorial** que te lo explica todo paso a paso



# Descarga e instalación

# A

- 1 Accede al enlace <https://www.arduino.cc/en/software>
- 2 Elija el sistema operativo y descargue el archivo correspondiente.
- 3 Siga los pasos de instalación.
- 4 Una vez instalado, el programa se abrirá automáticamente.



## Configuración del programa

# B

- 1** Con el programa abierto, ve a Archivo y luego a Preferencias
- 2** En "URLs adicionales del gestor de placas", pegue el enlace [https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)
- 3** Haga clic en Aceptar
- 4** Dirígete al Gestor de placas y descarga la placa ESP32

## Cargar y ejecutar actividad



- 1** Vaya a Herramientas > Placas y selecciona ESP32 > ESP32 Dev Module .
- 2** En “URLs adicionales del gestor de placas”, pegue el enlace [https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)
- 3** Acceda al enlace <https://faniot.com.ar/actividades-fanhub> y descargue la actividad número 1.
- 4** Copia el código del bloque de notas y pégalo en el editor de código.
- 5** Conecte su placa y seleccione el puerto correspondiente en Herramientas
- 6** Haga clic en Cargar para completar la actividad
- 7** Abrir el monitor serial y ver la impresión “Hola Mundo”.

# Como iniciar en ArduinoBlocks



## Como Iniciar en ArduinoBlocks

El Entorno de Programación en Bloque permite estudiante abordar la programación de manera práctica y visual en la que integra conceptos de robótica educativa con IOT.

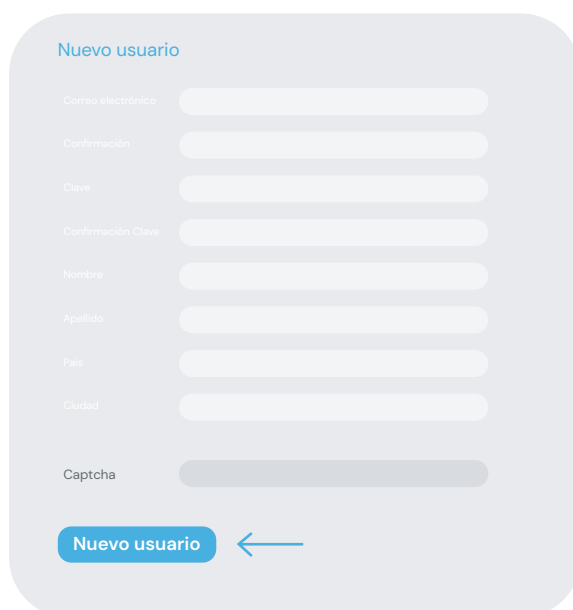
- 1 Ingresar a la página de ArduinoBlocks y hacer **click** en “iniciar sesión”



- 2 Hacer **click** en “Nuevo Usuario”.



- 3 **Completar** los datos, puedes usar una contraseña distinta a la de tu email, siempre y cuando luego la recuerdes!, No te olvides del **Captcha!** y hacer click en “**Nuevo usuario**”.



**4** Abrir la cuenta de e-mail y busca el **correo** que recibiste de ArduinoBlocks, y confirma la creación de la cuenta haciendo **click** en el link en azul.

Confirmar registro Recibidos x

**ArduinoBlocks** <arduinoblocks@gmail.com>  
Para mi -

Pincha en el enlace para completar el proceso de registro:

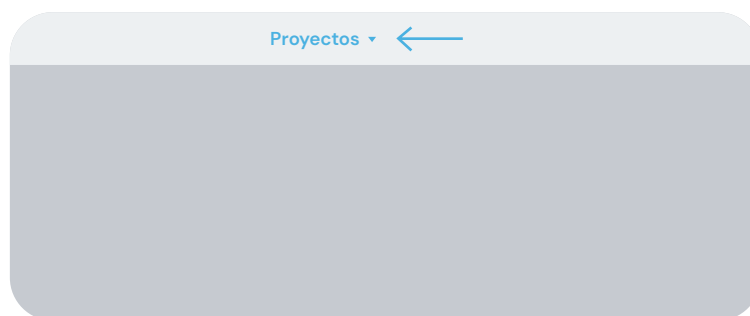
<http://www.arduinoblocks.com/web/site/confirmation?m=blynkfaniot%40gmail.com&t=80406a3b60962fed22fb4d6575e77941>

[www.arduinoblocks.com](http://www.arduinoblocks.com)

**5** Ahora si ya puedes iniciar sesión en ArduinoBlocks, con tu email y tu contraseña de ArduinoBlocks.

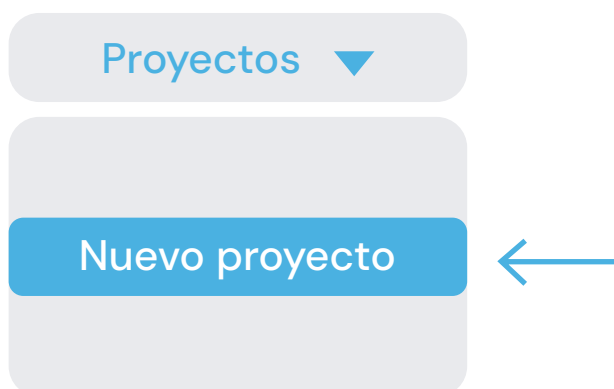


**6** ¡Ahora si!, en esta pantalla podrás ver todos tus proyectos en ArduinoBlocks y puedes crear un nuevo proyecto haciendo click en "Proyectos".





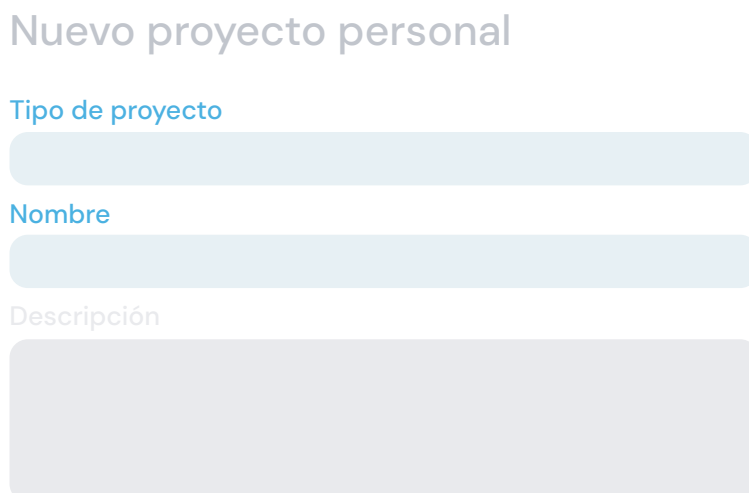
- 7 Para crear un nuevo proyecto debes hacer click en "Proyectos" y luego en "Nuevo proyecto".



- 8 Luego debes hacer click en "Iniciar un proyecto personal".



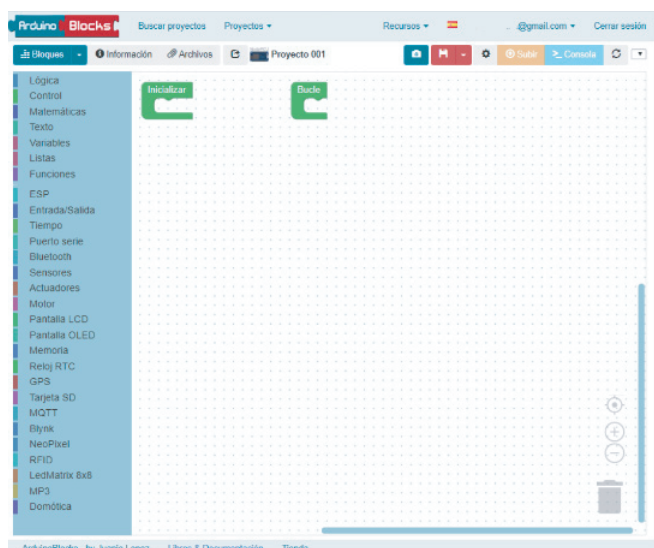
- 9 Completamos los datos "tipo de proyecto" y "nombre", los demás datos son opcionales

A screenshot of a form titled 'Nuevo proyecto personal'. It has three input fields: 'Tipo de proyecto', 'Nombre', and 'Descripción'. The 'Tipo de proyecto' and 'Nombre' fields are highlighted in light blue, and the 'Descripción' field is highlighted in light gray.

**10** Luego al final de la página hacé click en “nuevo proyecto”

Nuevo proyecto ←

**11** ¡Listo! Ya podes empezar a programar en **ArduinoBlocks**.



**¡IMPORTANTE!** Los pasos 1, 2, 3 y 4, los realizarás **una sola vez**, el paso 5 lo realizarás cada vez que ingreses a la plataforma ArduinoBlocks, y el paso 7 lo realizarás cada vez que quieras crear un nuevo proyecto.

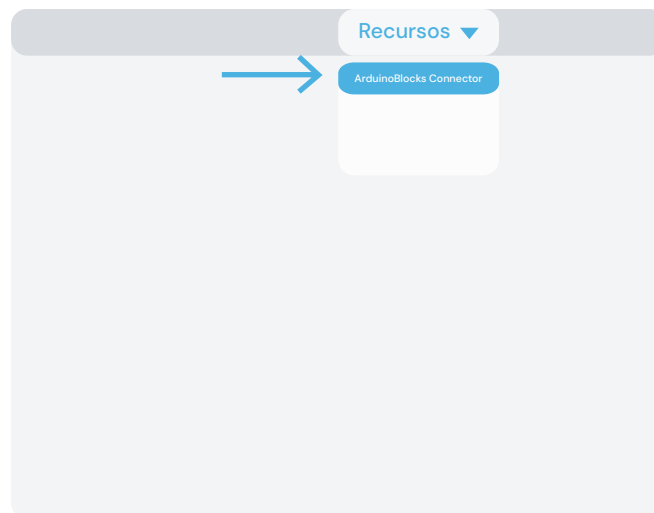
# Como instalar en ArduinoBlocks



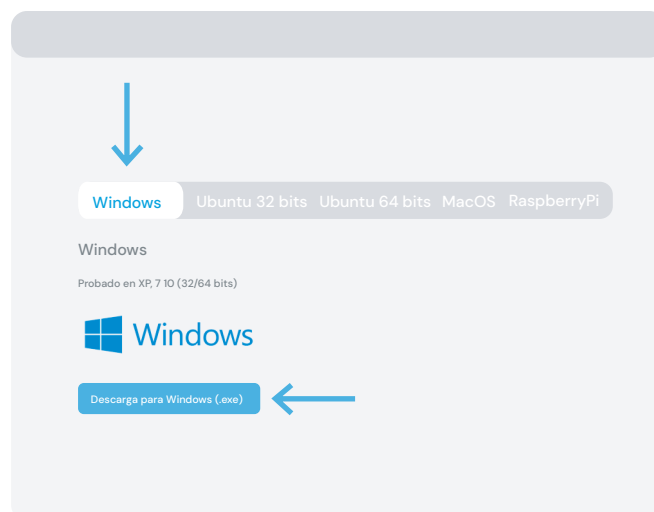
## Como **instalar** en ArduinoBlocks

ArduinoBlocks-Connector nos permite subir el programa hecho en la plataforma directamente a la placa, para instalarlo debemos seguir los siguientes pasos.

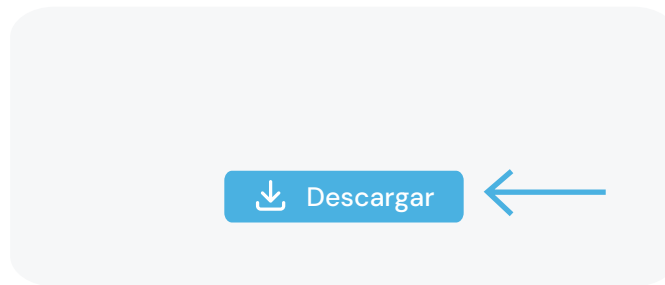
**1** En la pantalla principal de **ArduinoBlocks**, hacemos **click** en “recursos” y luego en “ArduinoBlocks conectar”



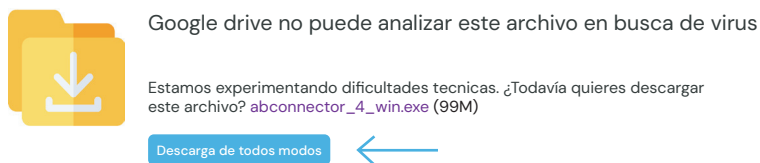
**2** Elegimos el **sistema operativo** en el que trabajamos, seguramente Windows, y hacemos click en “Descarga para Windows (.exe)”



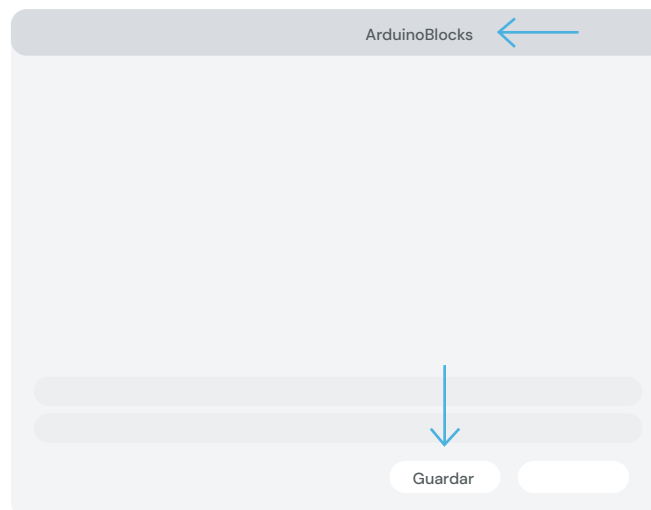
3 Luego hacemos **click** en “descargar”



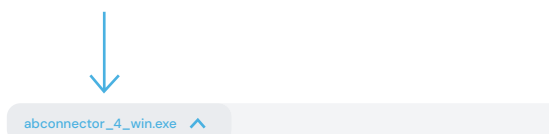
4 Luego hacemos **click** en “Descargar de todos modos”



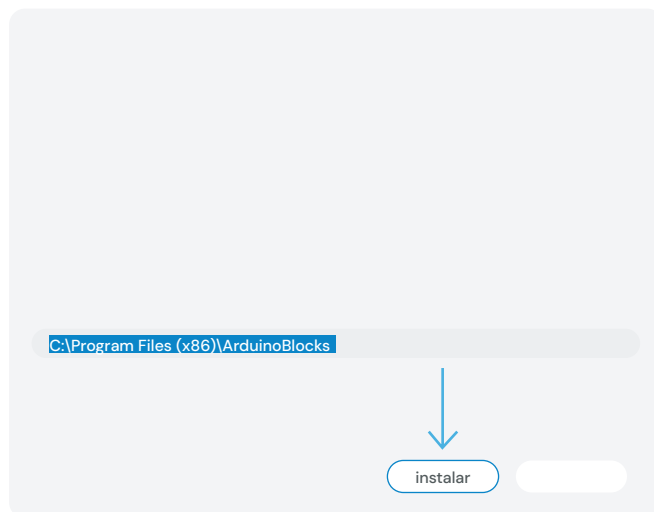
5 Luego elegimos una **carpeta** donde se descargará el archivo, y damos click en “Guardar”



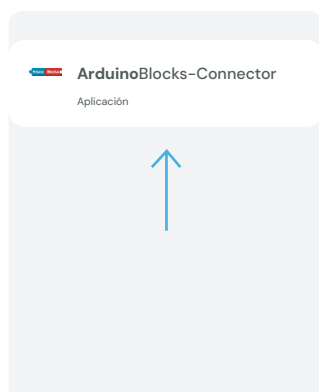
6 **Instalamos** ArduinoBlocks Connector, haciendo click en icono de la descarga.



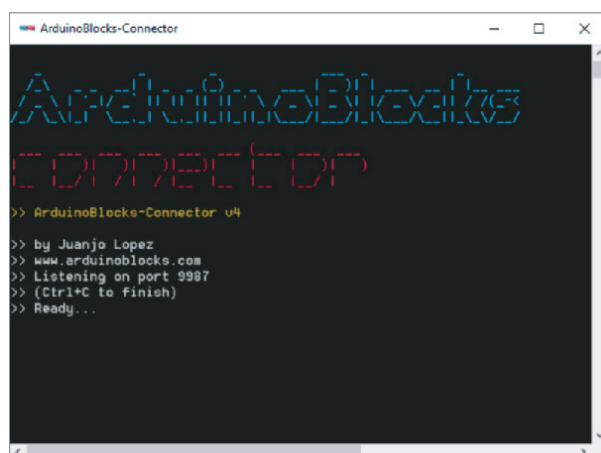
7 Luego hacemos **click** en “Instalar” y luego en “Cerrar”



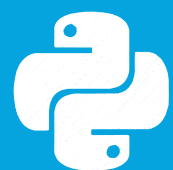
8 Ahora abrimos el programa, haciendo **click** en inicio y luego buscamos “ArduinoBlocks-Connector” y hacemos un click para abrirlo.



8 Luego se abrirá la siguiente ventana, **no cierres** esta ventana, déjala abierta mientras programas, ésta aplicación es la que sube tu programa a la placa!



# instalar Python



## Instalar Python

**1** Instalar Python (última versión)  
from <https://www.python.org/downloads/>



**2** Abra cmd y ejecute: (en caso de que uno no funcione, intente lo siguiente)

```
pip install esptool  
python -m pip install esptool  
pip2 install esptool
```

**3** La memoria flash se borra:

```
esptool.py - chip eps32 - port COMX erase_flash  
(Replace the Xwith the COM port number)
```

```
USB - SERIAL CH340 (COM9)
```

**4** Descargue el último firmware de micropython para ESP32  
desde: <https://micropython.org/download/esp32/>

**5** Copie el comando para cargar el firmware, cambiando el puerto com de linux a COMX, El firmware en este caso se guarda en la carpeta de descarga.



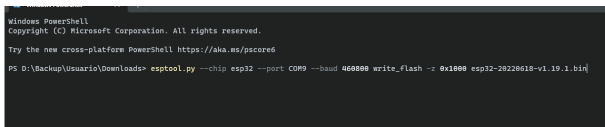
6 Abrir una terminal en descargar

7 Copie el nombre del firmware descargado



8 Coloque el comando copiado, colocando el puerto COM correspondiente y cambiando el nombre del firmware descargado y luego ejecute.

```
esptool.py --chip esp32 --port COM X --baud 4608000 write_flash -z 0x10000 esp 32-20220618-v1.19.1.bin
```



## Programador con MicroPython

1 Ir a la siguiente url: [codewith.mu](https://codewith.mu)

2 Vaya a Descargar y descargue la versión para Windows.

3 Abra el instalador, acepte los términos y presione instalar.

4 Abra el programa Mu, tardará un tiempo en ejecutarse.

5 Seleccionar para programar con ESP MicroPython



6 Seleccionar para programar con ESP MicroPython

7 Seleccionar para programar con ESP MicroPython

8 Conecte la ESP32

9 Haga clic en ejecutar



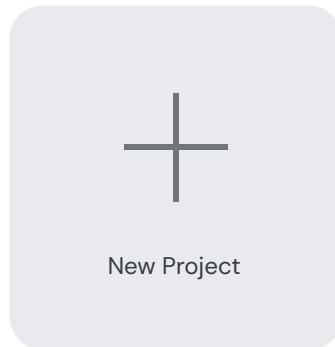
10 Ahora el programa se ejecuta en tiempo real en el ESP32.

# Crear un nuevo proyecto en Blynk



## Como Crear un Nuevo Proyecto en Blynk

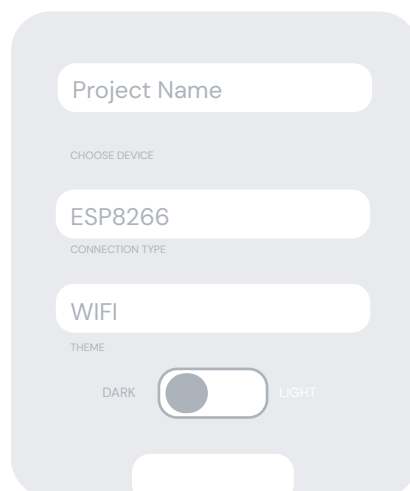
**1** En la siguiente pantalla apreciamos dos opciones, New Project y Community, pinchamos en New Project.



**2** En la opción "Project Name" es donde podemos asignarle un nombre a nuestro proyecto, vamos a ponerle a modo de ejemplo "Mi Proyecto".

Seguidamente en la opción "Choose Device" debemos buscar el nombre de nuestra placa de desarrollo , en este caso vamos a usar el "NodeMCU"

Finalmente en "Connection Type" , podemos escoger si vamos a implementar nuestro proyecto con WiFi ,redes de telefonía móvil ,Ethernet ,Bluetooth , entre otras.



**3** Pinchamos en "Create Project" y nos mostrará la siguiente notificación

**4** Aquí, Blynk nos está notificando que el proyecto se creó correctamente y nos ha enviado el Auth Token al correo con el cual nos hemos registrado. pinchamos en ok y ya estamos listos para trabajar!

## Glosario

**Placa:** Circuito impreso que alberga componentes electrónicos, como microcontroladores, resistencias, entre otros, para crear proyectos.

**Microcontrolador:** Chip programable que ejecuta tareas específicas, como el ESP32 o Arduino.

**Voltaje:** Diferencia de potencial eléctrico entre dos puntos de un circuito, medido en voltios (V).

**Pines:** Conexiones eléctricas en una placa, usadas para enviar o recibir señales o energía.

**Conectores RJ9:** En nuestro caso estos conectores se utiliza para Kit Sensor.

**ESP32:** Microcontrolador con WiFi y Bluetooth integrado, ideal para proyectos IoT y robótica.

**Arduinoblock:** Entorno gráfico para programar placas Arduino sin necesidad de escribir código.

**ESP Micropython:** Versión del lenguaje Python optimizada para microcontroladores como el ESP32.

**Firmware:** Software de bajo nivel programado en el hardware para controlar sus funciones básicas.

**Sensor:** Dispositivo que detecta cambios en el entorno (luz, temperatura, etc.) y los convierte en señales eléctricas.

**Switch:** Interruptor que abre o cierra un circuito para controlar el flujo de electricidad.

**Neopixels:** LEDs RGB direccionables, que permiten controlar color y brillo de cada LED de manera individual.

**TEMT6000 (GPIO39):** Sensor de luz conectado al pin GPIO39 de un microcontrolador para medir la intensidad lumínica.

**URL:** Dirección que se usa para acceder a recursos en Internet, como sitios web o archivos.

**Ethernet:** Tecnología de red cableada para la conexión de dispositivos y transmisión de datos.

**Bluetooth:** Tecnología inalámbrica de corto alcance usada para la comunicación entre dispositivos.

FANIOT