



# Paneles IoT

MQTT Server: wss://broker.hivemq.com/mqtt Port: 8884 Estado: Conectado

**COMPONENTE DESLIZADOR (R) - RGB**

Muestra el valor de la variable asociada al componente R del RGB

/smb/88/887/ 07/04/2021 21:22:07

255

**COMPONENTE DESLIZADOR (G) - RGB**

Muestra el valor de la variable asociada al componente G del RGB

/smb/88/874/ 07/04/2021 21:22:07

0

**COMPONENTE DESLIZADOR (B) - RGB**

Muestra el valor de la variable asociada al componente B del RGB

/smb/88/879/ 07/04/2021 21:22:07

0

**COMPONENTE ENVIAR TEXTO**

Este componente nos permite enviar un texto desde el panel hasta la placa

/smb/88/484/ 07/04/2021 21:22:07

**COMPONENTE GRAFICA**

Este componente nos permitirá mostrar una gráfica con los cambios de una variable.

/smb/88/381/ 07/04/2021 21:22:07

21:22:07 21:22:15 21:22:23 21:22:31 21:22:39 21:22:47

Seguir en vivo

**COMPONENTE INTERRUPTOR**

Al actuar sobre el elemento, cambiaremos el estado de una variable tipo booleana.

/smb/88/648/ 07/04/2021 21:22:07

ON

**Ejemplo de componente tipo indicador**

Este componente nos mostrará el estado de una variable digital.

/smb/88/675/ 07/04/2021 21:22:07

ON

**COMPONENTE SEMICIRCULO**

Muestra el valor de una variable en un semicírculo

/smb/88/628/ 07/04/2021 21:22:07

23.33

**COMPONENTE HORIZONTAL**

Este componente nos muestra el valor de una variable en una línea en formato horizontal.

/smb/88/487/ 07/04/2021 21:22:07

23.33

**COMPONENTE MAPA**

Este componente nos permite mostrar sobre un mapa un posición GPS

/smb/88/672/ 07/04/2021 21:22:07

**COMPONENTE TIPO TEXTO**

Este componente muestra el valor de por ejemplo una variable texto almacenada en nuestra placa

/smb/88/256/ 07/04/2021 21:22:07

...

**COMPONENTE NÚMERO**

Este componente nos permite mostrar en MQTT un valor numérico que nos envíe la placa. Irá incrementando en una unidad cada 1 segundo.

/smb/88/125/ 07/04/2021 21:22:07

...

Editado por [www.franciscojose.es](http://www.franciscojose.es) y [www.ardutaller.es](http://www.ardutaller.es)  
 en colaboración con [steamakersblocks](http://steamakersblocks.com) e [InnovaDidactic](http://InnovaDidactic.com).

**Este manual está en constante actualización**

Si quieres este documento en .pdf puedes descargarlo [aquí](#). Es posible que no esté tan actualizado como el documento online.

Si ahora mismo estás usando una versión en pdf o impresa, te recomendamos que visites la versión [online](#) escanees el código QR, para tener siempre disponible el documento actualizado. Este manual ha sido desarrollado por ardutaller, basándose en documentación publicada en Catalán por Miguel Pérez para Robolot y otros documentos referenciados en la bibliografía.



**Programación realizada con:**

**Basado en kit distribuido por:**

**Paneles IoT**

## Paneles IoT

---

---

### [1. Paneles IoT](#)

#### [1.1. Crear un panel IoT](#)

#### [1.2. Configuración panel IoT](#)

#### [1.3. Crear nuevo componente](#)

##### [1.3.1. Configuración de componentes en panel](#)

##### [1.3.1. Configuración de componentes en Esp32steamakers](#)

##### [1.3.2. Inicializar servidor MQTT](#)

##### [1.3.2. Incluir componentes en proyecto](#)

#### [1.4. Tipo de componentes](#)

##### [1.4.1. Número](#)

##### [1.4.2. Indicador](#)

##### [1.4.3. Gráfico](#)

##### [1.4.4. Tipo semicírculo](#)

##### [1.4.5. Nivel horizontal](#)

##### [1.4.6. Interruptor](#)

##### [1.4.7. Deslizador](#)

##### [1.4.8. Texto](#)

##### [1.4.9. Enviar texto](#)

##### [1.4.10. Mapa](#)

##### [1.4.11. Organizar elementos del panel](#)

### [2. Bibliografía](#)

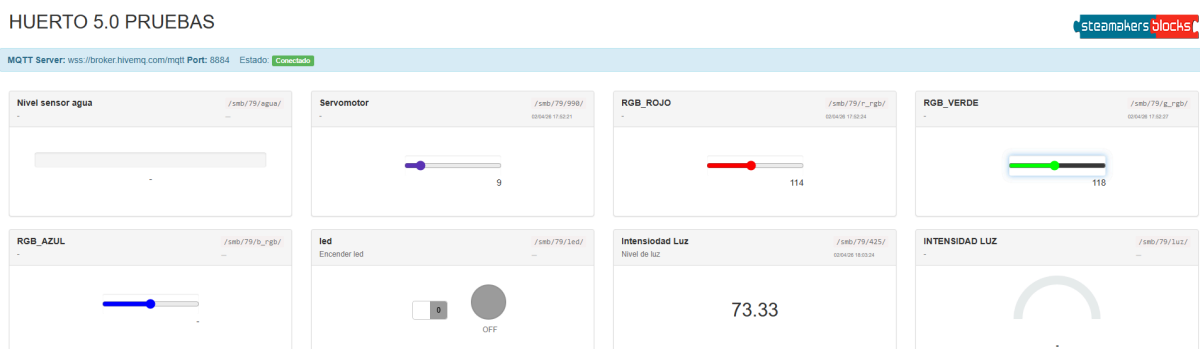
# Paneles IoT

## 1 Paneles IoT

Los paneles IoT, son una funcionalidad recientemente añadida a la plataforma STEAMakersBlocks. Dicha funcionalidad fue presentada en la última edición de Simo celebrada en IFEMA-Madrid en 2025, y actualmente sólo está disponible para usuarios premium de la plataforma.

Nos permitirá interactuar con la placa desde internet, ya sea para enviar información a la placa, como para que la placa nos devuelva información. Este último aspecto, ya lo vimos con Thingspeak, pero con esta nueva plataforma damos un paso más, ya que podremos controlar el funcionamiento de un motor o el encendido de un led, a través de un pc o terminal móvil. No será necesario que placa y dispositivo estén en la misma red, sólo se requiere del inicio del usuario en la plataforma STEAMakersBlocks, y que la placa esté correctamente conectada a internet.

El aspecto del panel será algo similar a esto:



### 1.1 Crear un panel IoT

Como ya se ha indicado anteriormente, para poder utilizar esta funcionalidad hay que ser usuario premium de STEAMakerblocks.

El acceso se hará desde un menú que aparece en la barra principal de la web (Utilidades -> Paneles IoT):

## Paneles IoT



Nos aparecerá una pantalla como la siguiente, en la que tendremos disponibles todos los paneles que vayamos creando.

Para crear uno nuevo, pulsamos arriba a la derecha sobre “Nuevo panel”.

Panel ID	Título	Subtítulo	Público	Server (websocket)	Port (WS)	
	HUERTO 5.0 PRUEBAS		Privado	wss://broker.hivemq.com/mqtt	8884	  
	Panel Educacont		Privado	wss://broker.hivemq.com/mqtt	8884	  
	pruebas		Privado	wss://broker.hivemq.com/mqtt	8884	  

### 1.2 Configuración panel IoT

Una vez creado el panel, procedemos a configurarlo, rellenando los campos siguientes:

## Paneles IoT

### Nuevo panel

**Título**

  
**Subtítulo**  

Use HiveMQ (for testing only)

The public server [broker.hivemq.com] is not secure — for testing only.

MQTT-Websocket (panel):

**Server (websocket)**  **Port (WS)**

MQTT-TCP (ESP32 STEAMakers):

**Server (TCP)**  **Port (TCP)**

**Usuario**  **Clave**

Título y Subtítulo, pondremos nombres que identifiquen al proyecto.

En los siguientes campos, configuraremos el servidor MQTT, con el que intercambiaremos información. Puesto que la mayoría de servicios son de pago, para las pruebas, vamos a hacer uso del servicio gratuito HiveMQ, que aunque no es seguro en cuanto al modo de compartir la información, para fines didácticos es muy adecuado. No está recomendado su uso profesional o comercial.

Si pulsamos sobre Use HiveMQ, nos muestra una texto de advertencia:

**www.steamakersblocks.com dice**

Warning: The public HiveMQ server (broker.hivemq.com) does not require authentication and is not secure. Use for educational purposes only. Do not use in production. Do you want to fill in the fields with HiveMQ values for testing?

*Advertencia: El servidor público de HiveMQ (broker.hivemq.com) no requiere autenticación y no es seguro. Úselo solo con fines educativos. No lo use en producción. ¿Desea completar los campos con valores de HiveMQ para realizar pruebas?*

## Paneles IoT

Una vez hecha esta advertencia, pulsamos sobre aceptar.

Se cumplimentan de manera automática, los campos de conexión requeridos. El usuario y contraseña lo dejaremos en blanco. Ya podemos pulsar en guardar para crear el panel.

MQTT-Websocket (panel):

**Server (websocket)**  **Port (WS)**

MQTT-TCP (ESP32 STEAMakers):

**Server (TCP)**  **Port (TCP)**

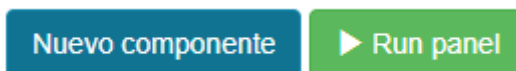
**Usuario**  **Clave**

### 1.3 Crear nuevo componente

Para la creación de un nuevo componente, arriba a la derecha pulsamos sobre crear componente.

Panel > PRUEBA TUTORIAL ARDUTALLER Nuevo componente ▶ Run panel

Type	Nombre	Descripción	Topic MQTT	End row	Actions	Bloques
------	--------	-------------	------------	---------	---------	---------



Nuevo Componente: Crear nuevo componente  
Run panel: ejecutar panel

A continuación veremos todos los componentes que podremos crear.

Antes de entrar en detalle con todos los componentes, vamos a ver la configuración que es común a todos los componentes

#### 1.3.1 Configuración de componentes en panel

En la opción "Type", indicaremos el tipo del componente que vamos a configurar.

## Paneles IoT

---

En los siguientes apartados, vemos la configuración de todos y cada uno los tipos de componentes así como ejemplos de programación en stemakersblocks.

Estos campos que se muestran a continuación, son comunes a todos los componentes:

**Type**

Número #

**Nombre**

**Descripción**

**Topic MQTT**

/smb/80/426/

End row

En nombre y descripción indicaremos información sobre el componente.

En topic MQTT, nos generará automáticamente un número, que será con el que después lo identificaremos en stemakersblocks, o bien podremos ponerle un número. Tan sólo tendremos que cambiar el número que nos pone, por un nombre que nos sea más fácil de identificar.

/smb/80/426/ ----> /smb/80/426/comp/

El primer elemento smb, es común y hace referencia a steamakersblocks  
El segundo elemento, en este caso, 80, es el número de panel

El tercer elemento, en este caso el 426 es un número aleatorio que nos genera el sistema y sirve para identificar el componente. Si queremos podemos añadir después un texto que nos ayude a identificarlo, utilizando un texto que nos parezca más familiar.

Los campos que aparecen más abajo, variarán en función del módulo elegido. Una vez guardemos el componente, se accederá al menú de componentes:



## Paneles IoT


Panel > PRUEBA TUTORIAL ARDUTALLER Nuevo componente ▶ Run panel

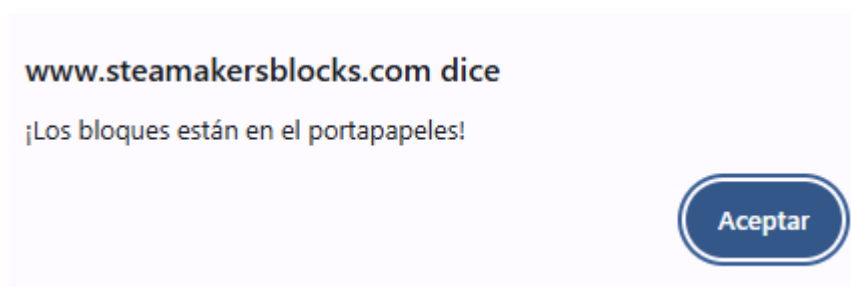
Type	Nombre	Descripción	Topic MQTT	End row	Actions	Bloques
number	Componente prueba 1		/sub/50/426/	No	 	


Guardar orden MQTT Init block (ESP32) MQTT Pub blocks (ESP32)


A continuación analizaremos los botones que nos aparecen en esta pantalla:

A lado de cada componente nos aparecen estos botones, que serán para editor o para borrar el componente:  

También nos aparece este botón: . Si lo pulsamos, nos copiará una información en el portapapeles, que como veremos a continuación, nos servirá para la programación en STEAMakersblocks.



: nos copiará en el portapapeles la configuración del servidor MQTT que como veremos a continuación en la configuración de componentes, tendremos que poner en el bloque inicializar.

: copia en el portapapeles, la información de todos los bloques que tengamos en el panel.

A la hora de trabajar, es muy importante, que tengamos en cuenta, que cada vez que incluyamos algo en el portapapeles, eliminará lo anterior. Es por ello que tendremos que hacer este traspaso de información de manera ordenada.

### 1.3.1 Configuración de componentes en Esp32steamakers

## Paneles IoT

Dentro de cualquier proyecto que tengamos en la plataforma, podremos incluir los módulos para interactuar con el panel.

Lo que a continuación se muestra es una explicación genérica, mas adelante entraremos en el detalle de la configuración de los diferentes componentes.

### 1.3.2 Inicializar servidor MQTT

En primer lugar, como en todos los proyectos de IoT, tendremos que conectar placa a internet, para ello, conectaremos a una wifi en el bloque inicializar.

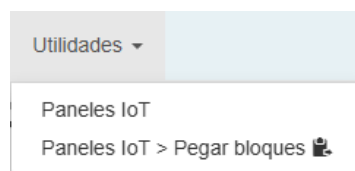
A continuación crearemos una función que será inicializar\_MQTT, en la que incluiremos unos bloque que obtendremos del modo siguiente:

Ya indicamos anteriormente, que pulsando sobre



, se nos copia en el portapapeles una información, por lo tanto, si no lo hemos hecho, lo hacemos ahora, vamos al panel, pulsamos sobre ese botón y volvemos nuevamente al proyecto.

En el menú superior, sobre utilidades, nos aparece una opción "Pegar bloque". pulsamos sobre ella para que nos pegue lo que guardamos en el portapapeles:

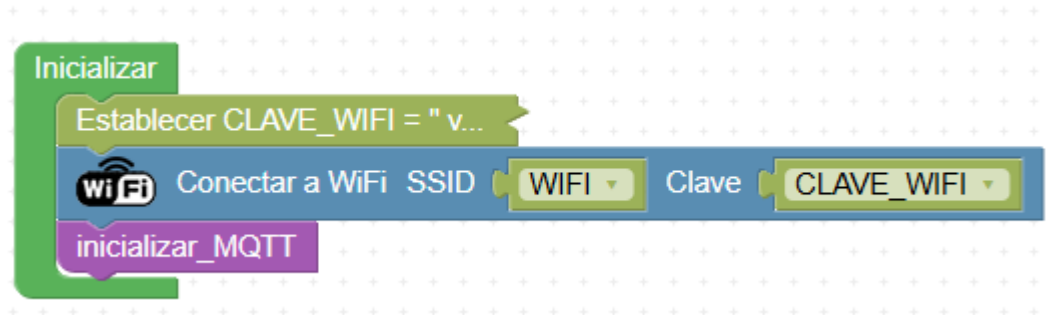


Nos pega este bloque que incluiremos en la función.



Con esto ya tendremos configurada la conexión al servidor MQTT, quedando el bloque inicializar de este modo:

## Paneles IoT




### 1.3.2 Incluir componentes en proyecto

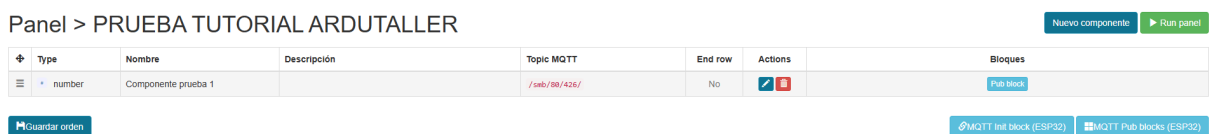
En los siguientes apartados, veremos cómo configurar la publicación de los diferentes temas o topic, en función del tipo de componente. En este apartado, a modo general, vamos a ver cómo llevar la información de los bloques del panel al proyecto.


Como ya indicamos, dentro del menú principal del panel, hay varios botones que nos llevan la información al portapapeles.


Al igual que hicimos para pegar la información de inicialización, podremos hacer ahora lo mismo con los componentes.

Dentro del menú de paneles, junto al panel en cuestión, pulsaremos

sobre  , y entraremos en la sección de paneles:



Si pulsamos sobre  , que está al lado de cada uno de los componentes, copiaremos la información de ese componente en el portapapeles.

Si pulsamos sobre  , copiamos al portapapeles, la información de todos los bloque de tipo “publicar”. Si

## Paneles IoT



pulsamos sobre  , nos copiará todos los bloques de tipo Suscripción.

Nos pegaría algo así



Si pulsamos sobre “?” nos muestra un comentario, con la información sobre el componente.

En el campo valor, como veremos a continuación, podremos poner el valor que se enviará al servidor MQTT, pudiéndose poner ahí, por ejemplo una variable. Si pulsamos sobre Retain, mantendrá almacenado el servidor, el último valor recibido.

Además junto a a cada componente, podremos llevarlo al portapapeles de manera individual, pulsando sobre un botón u otro de los que nos



aparecen junto al componente.

### 1.4 Tipo de componentes

Podremos configurar un total de 10 componentes diferentes. A continuación veremos cómo configurarlo, no sólo en el panel, sino también en el proyecto que enviaremos a la placa.

Puesto que vamos a trabajar sobre un mismo ejemplo de proyecto, la publicación de los topics, la haremos haciendo uso de unas funciones, que iremos incluyendo en el bucle.



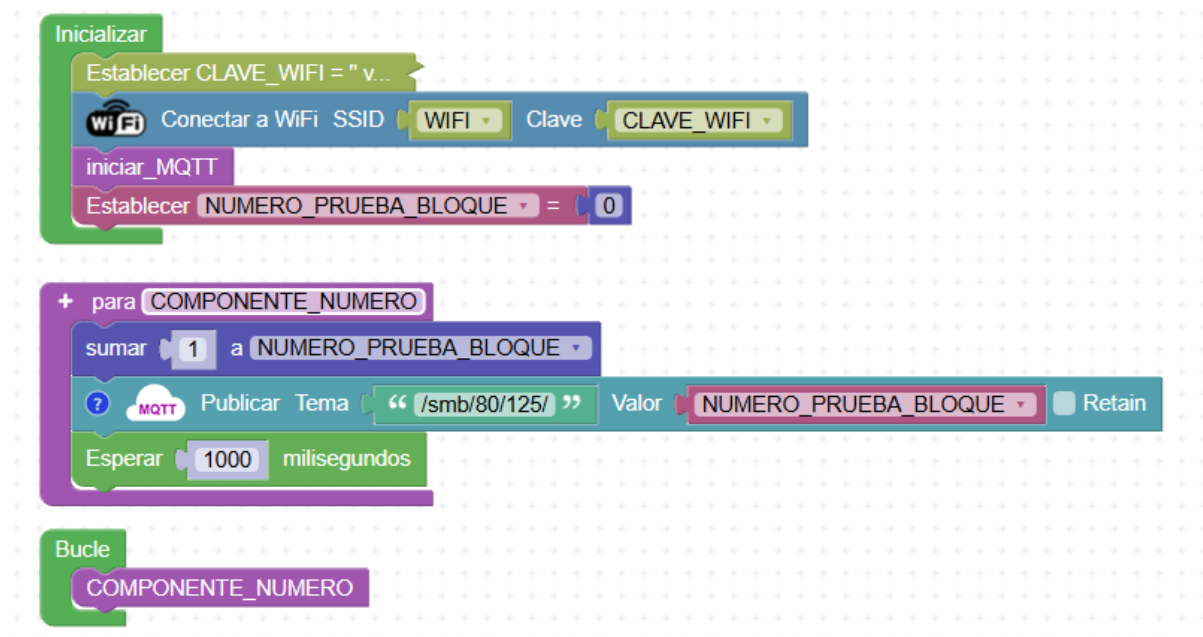
## Paneles IoT

Trabajaremos con valores que nos darán los sensores y otros que se generarán incrementando según indiquemos en el programa.

### 1.4.1 Número

Este componente nos permitirá mostrar en nuestro panel un valor de tipo numérico que hayamos enviado desde nuestra placa al servidor MQTT.

Mediante la opción pega bloques IoT, que indicamos anteriormente nos llevamos la información del nuevo componente al proyecto, y quedaría algo así. Sólo tendremos que de ponerlo en el bucle. Para facilitar la vista del resultado,, lo único que hemos hecho ha sido, crear una variable numérica, con un valor una valor inicial fijo, y que hemos mostrado. En el programa, vamos a hacer que dicho valor se incremente en "1", en intervalos de 1 segundo.



Subimos el programa a la placa, y una vez subido, abrimos el panel. El resultado será el siguiente:

## Paneles IoT

**COMPONENTE NÚMERO**  
Este componente nos permite mostrar en MQTT un valor numérico que nos envíe la placa. Irá incrementando en una unidad cada 1 segundo.

/smb/80/125/  
03/04/26 11:52:49

64.00

Nuestro panel, sólo será accesible con nuestro usuario de steamakerblocks, y accediendo un enlace similar a este:

<https://www.steamakersblocks.com/web/panel/run/xx> , donde xx, será el número de panel. Esto será válido para todos los componentes que vayamos añadiendo, de manera que nos irán saliendo en el panel.

### 1.4.2 Indicador

Este componente nos indicará el estado de una variable digital.

Podremos configurar además el color que queremos que muestre cuando esté en "on", y en "off"

En el panel quedaría del modo siguiente.

**Ejemplo de componente tipo indicador**  
Este componente nos mostrará el estado de una variable diigital.

/smb/80/675/  
—

  
OFF



## Paneles IoT

Para incluir el ejemplo en el proyecto, vamos a crear un variable de tipo Boolean (0-1) que iremos cambiando el valor el estado de la misma en intervalo de 2 segundos.

Pegaremos el bloque, lo incluiremos en una función y quedará de este modo.

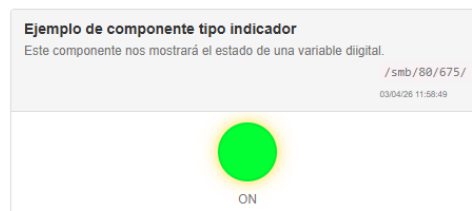
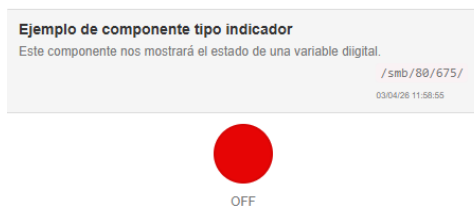
```

Inicializar
  Establecer CLAVE_WIFI = " v...
  Conectar a WiFi SSID WIFI Clave CLAVE_WIFI
  iniciar_MQTT
  Establecer NUMERO_PRUEBA_BLOQUE = 0
  Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = verdadero

Bucle
  COMPONENTE_INDICADOR

+ para COMPONENTE_INDICADOR
  Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = verdadero
  MQTT Publicar Tema "/smb/80/675/" Valor BOOLENA_PRUEBA_BLOQUE_INDICADOR Retain
  Esperar 2000 milisegundos
  Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = falso
  MQTT Publicar Tema "/smb/80/675/" Valor BOOLENA_PRUEBA_BLOQUE_INDICADOR Retain
  Esperar 2000 milisegundos
  
```

El resultado será que el componente irá cambiando de color en intervalos de 2 segundos.



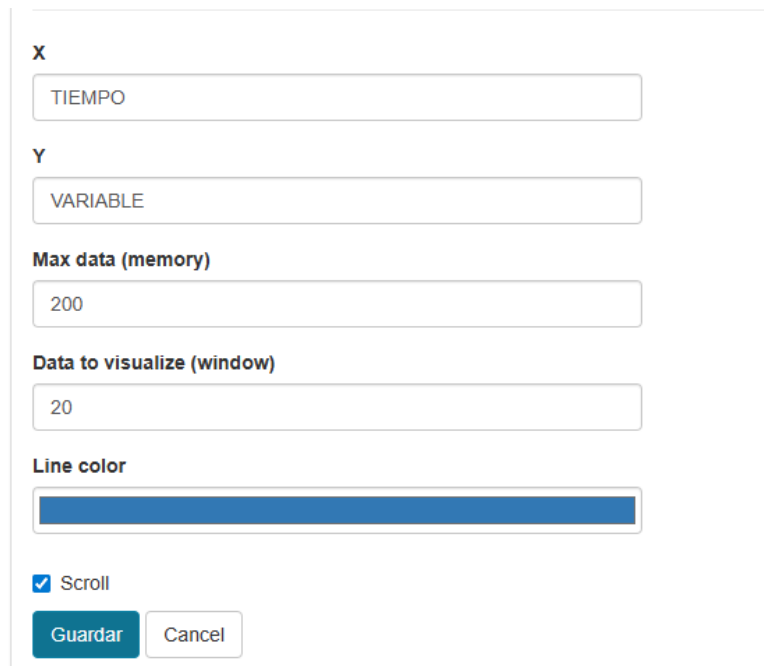
### 1.4.3 Gráfico

Este componente, será similar a lo que vimos por ejemplo en Thingspeak, mostrándonos una gráfica dentro de nuestro panel.

## Paneles IoT

Podremos poner valor al eje X y al eje Y, así como definir un máximo de datos en memoria y un máximo de datos a visualizar en la ventana. También podremos definir el color de la línea.

En este caso, vamos a mostrar el valor instantáneo que nos está proporcionando un sensor de luz, a través de una variable determinada.




X  
TIEMPO

Y  
VARIABLE

Max data (memory)  
200

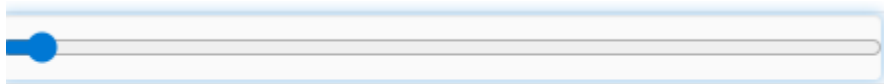
Data to visualize (window)  
20

Line color  


Scroll

Guardar Cancel

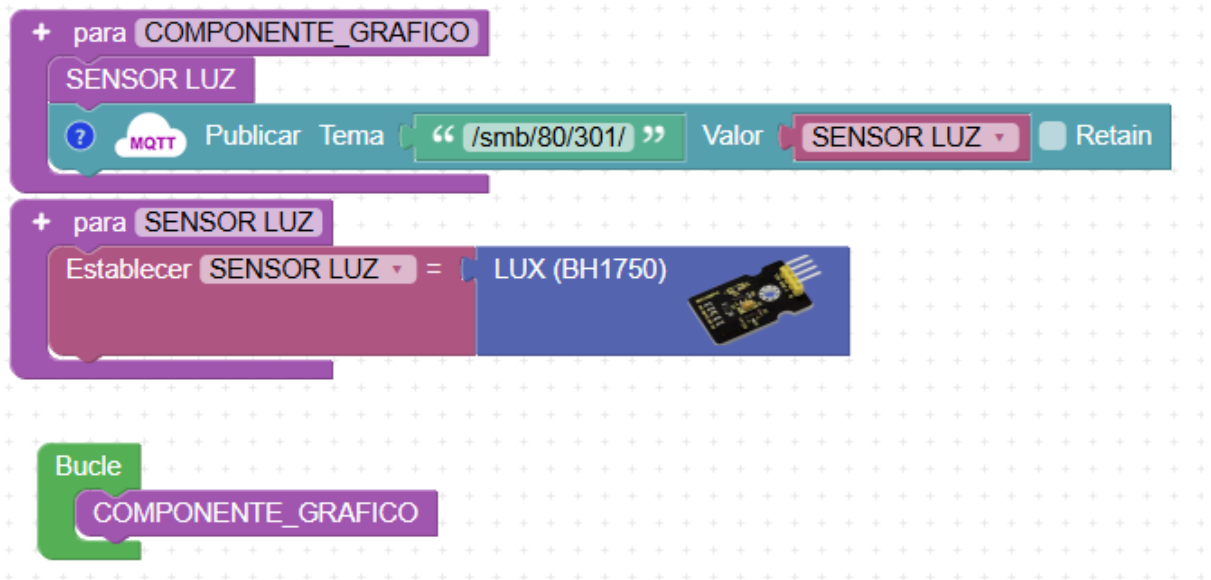
Si habilitamos scroll, nos aparecerá una línea debajo que nos permitirá desplazarnos en la gráfica.



Utilizaremos un sensor de luz, y asignaremos el valor del mismo a una variable.



## Paneles IoT



El resultado en el panel será algo similar a esto:



Si pulsamos sobre los puntos que se generan, nos indicarán la hora de la medida y el valor.


Además podremos descargar un archivo en formato csv.

## Paneles IoT

### 1.4.4 Tipo semicírculo

Nos muestra el valor de la variable, en una gráfica, a modo de semicírculo. Además podemos definir el rango en el que se muestran los valores.

Lo configuraremos así:

Type  
Semicírculo 

Nombre  
COMPONENTE SEMICIRCULO

Descripción  
Muestra el valor de una variable en un semicírculo


Topic MQTT  
/smb/80/628/

End row

---

Min value  
0

Max value  
255

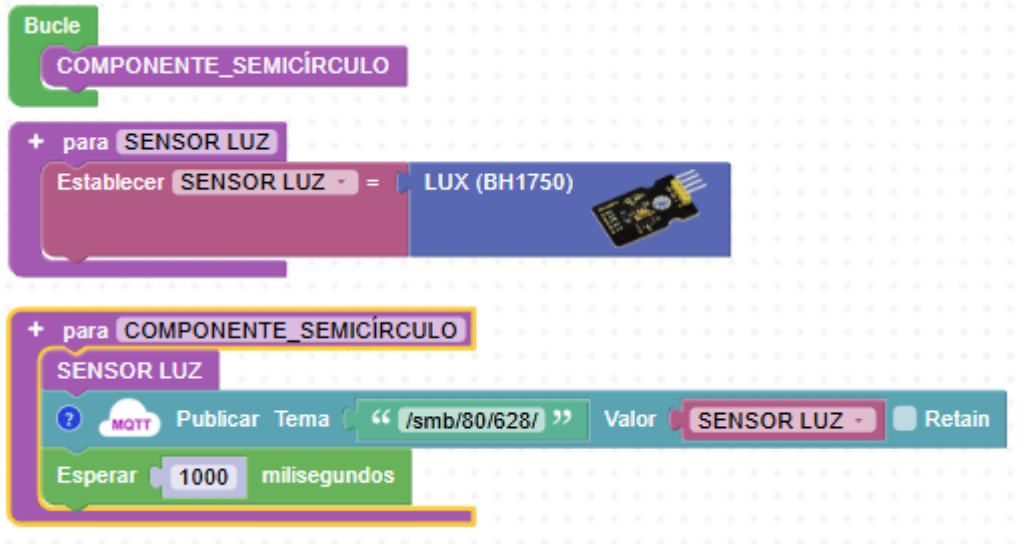
Color  


En este caso mostraremos el valor de la variable luz con este modo de visualización.

Los bloques en el proyecto serán los siguientes:



## Paneles IoT



Si queremos que el valor se muestre de manera más instantánea, podremos quitar la temporización.



### 1.4.5 Nivel horizontal

En este caso, la visualización del valor se hace en una gráfica horizontal.



## Paneles IoT

La configuración será así:

Type

Nombre

Descripción

Topic MQTT

End row

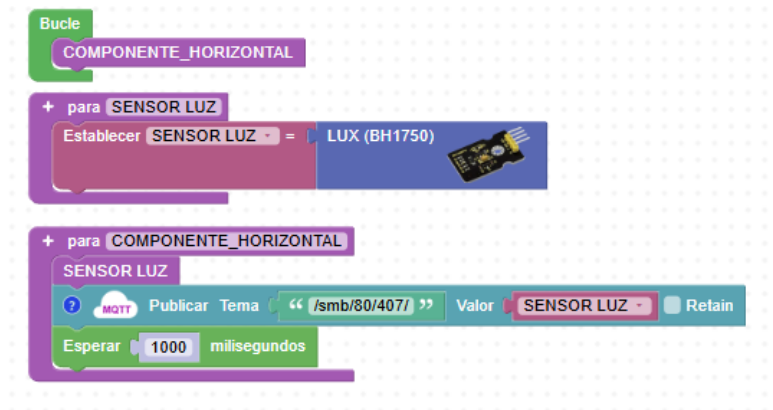
---

Min value

Max value

Color

Los bloques se configurarán de esta forma:



## Paneles IoT

**COMPONENTE HORIZONTAL**

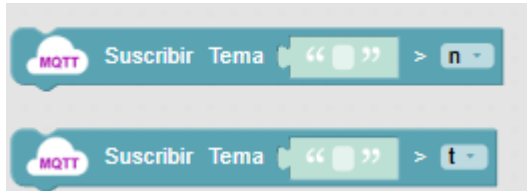
Este componente nos muestra el valor de una variable en una línea en formato horizontal.

/smb/80/407/  
03/04/26 13:19:06



### 1.4.6 Interruptor

Con este bloque comenzamos con los componentes, que envían información desde el navegador hasta la placa. Veremos por tanto que varía la forma en que enviamos la información MQTT, En lugar de “Publicar tema”, lo que haremos será “Suscribir tema”.



Cuando pulsemos sobre Pub block, nos copiará un bloque de este tipo con el nombre de la variable asignada.

Definimos una variable de tipo número, que recibirá el estado del valor del topic del interruptor del panel. Esta variable a su vez, cambiará el valor de la variable booleana que definimos anteriormente para que cambiar de color el componente indicador.

Después haremos un ejemplo en que controlaremos el encendido de un led rgb en un dos estados: apagado - encendido rojo que estará conectado a la placa.

Cogeremos el bloque “suscribir tema” utilizando la opción “pegar bloques paneles” y no aparecerá ya configurado de esa manera.

Cogemos este bloque, ya que previamente habremos definido la variable INTERRUPTOR como de tipo numérico.



## Paneles IoT

```
MQTT Suscribir Tema " " > n
```

```
Inicializar
  Establecer CLAVE_WIFI = " v...
  WiFi Conectar a WiFi SSID WIFI Clave CLAVE_WIFI
  iniciar_MQTT
  Establecer NUMERO_PRUEBA_BLOQUE = 0
  Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = verdadero
  Establecer ESTADO_INTERRUPTOR = 0
```

Crearemos una función denominada “componente interruptor” en la que mediante un condicional analizaremos el valor de la variable. Si es a 1, sera que el interruptor está encendido, y si es otro número será apagado. En este caso el interruptor enviaría un “0”,

Este será el bloque para hacer la igualdad:

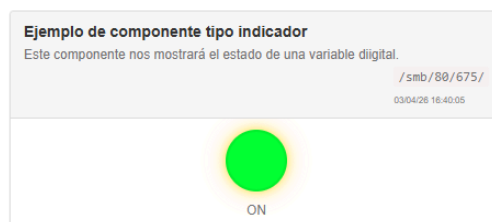
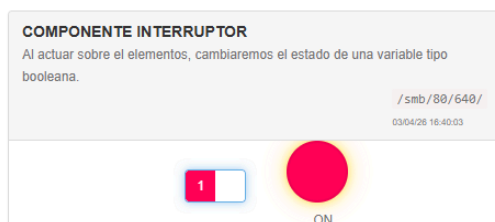
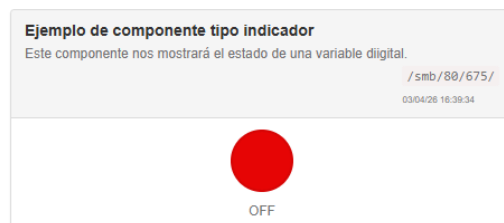
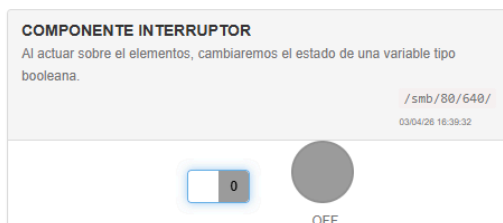
```
= Devuelve verdadero si ambas entradas son iguales. (NUMERIC)
```

Los bloques quedarían así:

```
Bucle
  COMPONENTE INTERRUPTOR
  + para COMPONENTE INTERRUPTOR
    MQTT Suscribir Tema "/smb/80/640/" > ESTADO_INTERRUPTOR
    + si ESTADO_INTERRUPTOR = 1
      hacer
        Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = verdadero
        MQTT Publicar Tema "/smb/80/675/" Valor BOOLENA_PRUEBA_BLOQUE_INDICADOR Retain
      sino
        Establecer BOOLENA_PRUEBA_BLOQUE_INDICADOR = falso
        MQTT Publicar Tema "/smb/80/675/" Valor BOOLENA_PRUEBA_BLOQUE_INDICADOR Retain
```

## Paneles IoT

Si actuamos sobre el interruptor, cambiamos el estado del indicador de rojo a verde.



### 1.4.7 Deslizador

Este módulo nos permitirá enviar a placa un valor numérico, el cual vendrá definido en función de la posición del deslizador.

Previamente deberemos haber definido cual es el rango de funcionamiento del deslizador.

En este caso, para hacer un caso práctico, lo que vamos hacer es colocar 3 deslizadores, los cuales, irán asociados a 3 variables de tipo numérico, que definirán la cantidad de R, de G, y de B, que tiene una led RGB conectado a nuestra placa. Los deslizadores se definirán con un intervalo de 0 a 255.

Sólo mostramos la creación del deslizador para el componente R del RGB. Dejamos el resto como ejercicio para el lector.



## Paneles IoT

[PRUEBA TUTORIAL ARDUTALLER] Nuevo componente

Type: Deslizador

Nombre: COMPONENTE DESLIZADOR (R) - RGB

Descripción: Muestra el valor de la variable asociada al componente R del RGB

Topic MQTT: /smb/80/087/

End row

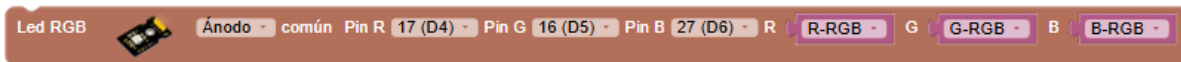
Min value: 0

Max value: 255

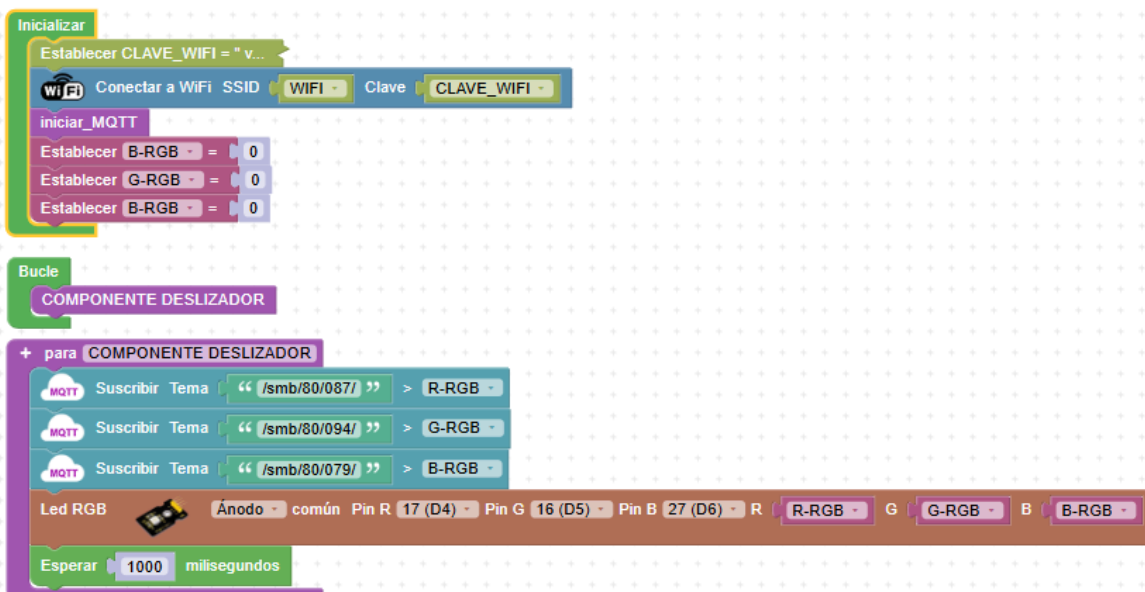
Color:

Guardar Cancel

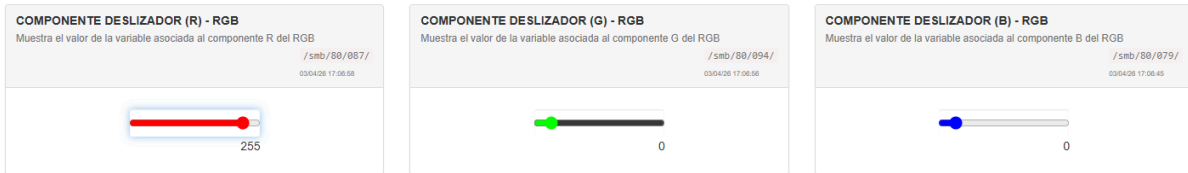
Previamente hemos conectado el led a los pines que se indican en el siguiente bloque:



Los bloques quedarían del modo siguiente:



## Paneles IoT



### 1.4.8 Texto

Este componente, muestra un valor de tipo texto que sea enviado desde la placa. Puede por ejemplo cualquier variable de tipo texto. Utilizaremos un bloque de tipo "Publicar", por lo que podremos generar pulsando Pub block.

En este caso lo que vamos a mostrar es un valor que enviaremos desde la consola serie, y que se mostrará en el panel.

#### Consola serie

Baudrate: 9600 Conectar Desconectar Limpiar

Enviar

☐xã™"/-n^..pÒQò%-☐☐Áü

#### COMPONENTE TIPO TEXTO

Este componente muestra el valor de por ejemplo una variable texto almacenada en nuestra placa

/smb/80/256/

03/04/26 17:25:04

HOLA ESTO ES UNA PRUEBA

### 1.4.9 Enviar texto

Con este componente, haremos el proceso contrario, es decir enviaremos un texto desde el panel y lo guardaremos en una variable de la placa. Para ver el resultado, mostraremos el texto recibido en la consola serie. Para ellos haremos refresco de escritura en la consola cada 1 segundo.

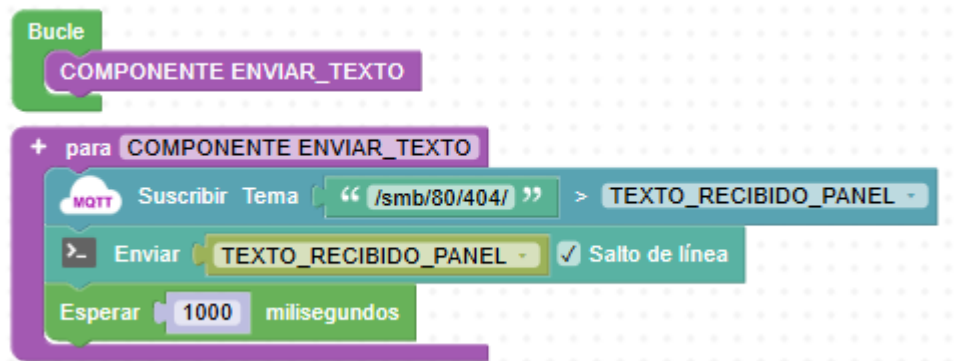


## Paneles IoT

Este será un módulo de tipo Suscribirse y de tipo texto:



Los bloques quedarían así:



El resultado sería este:



### 1.4.10 Mapa

Este módulo nos permitirá mostrar un mapa en el panel de manera que indique la posición en que se encuentra por ejemplo la placa. Si disponemos de un GPS conectado a la placa, tan sólo tendríamos que definir este campo, en función de lo que nos diga el localizador GPS.

En este caso no dispongo de localizador GPS, por lo que voy a poner la ubicación de manera manual. La placa le dará dicha información al panel.

Vamos a poner por ejemplo la ubicación de Alcoy, que hemos localizado en la web: <https://www.123coordenadas.com/coordinates/677552-alcoy-espana>

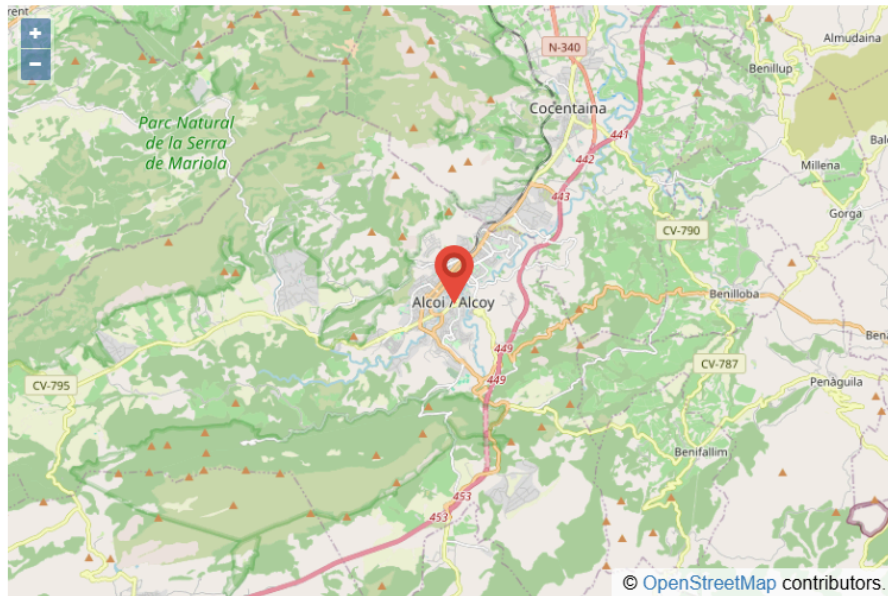


## Paneles IoT

Esta es la información que no proporciona la web:

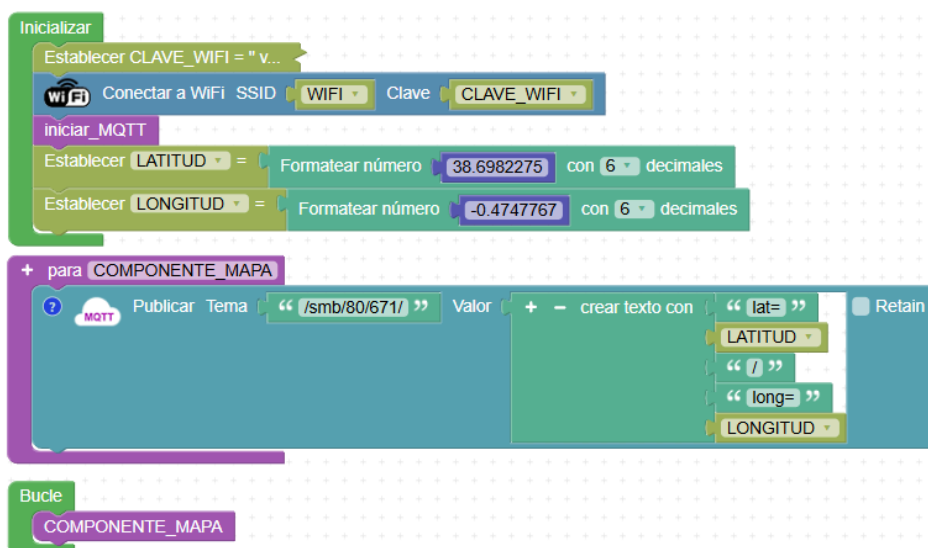
### Las coordenadas GPS para Alcoy españa

Latitud: 38.6982275  
 Longitud: -0.4747767



Tendemos que incluir la coordenadas en el formato que nos indica el panel, es decir este formato, dentro del bloque inicializar:

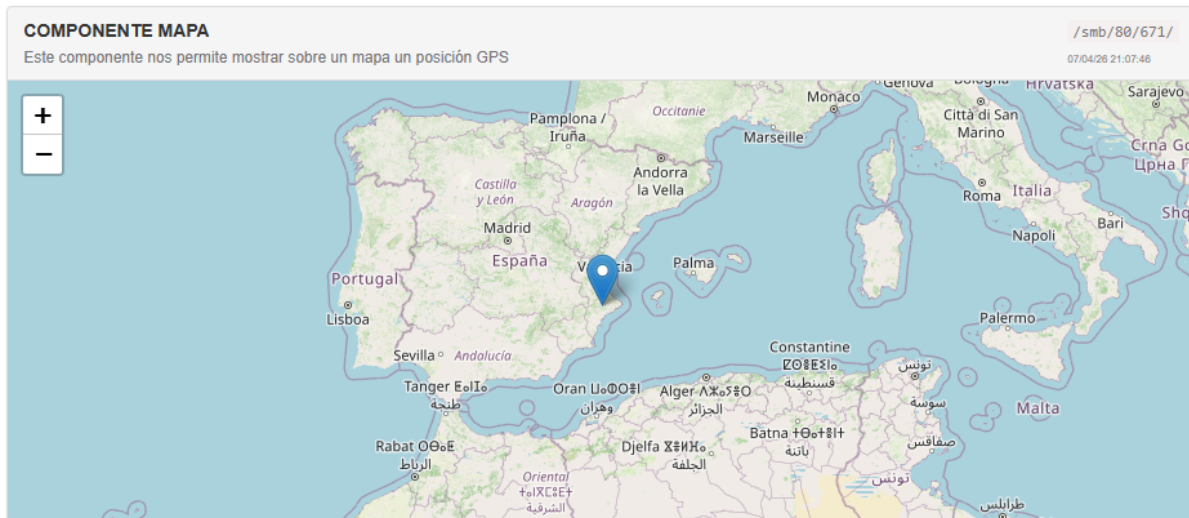
Tendremos que configurar los bloques de este modo:



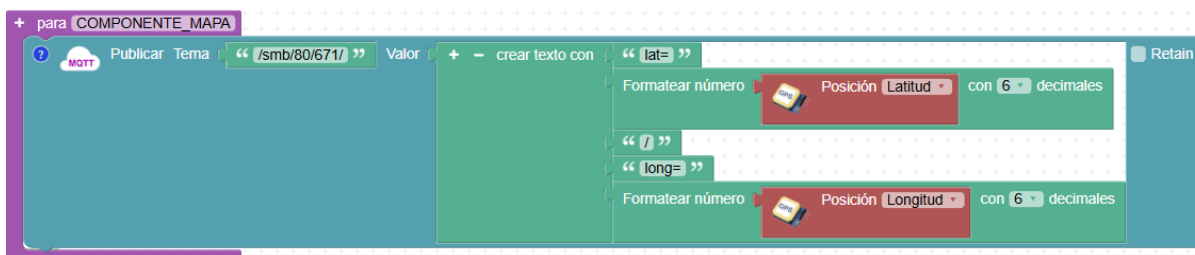
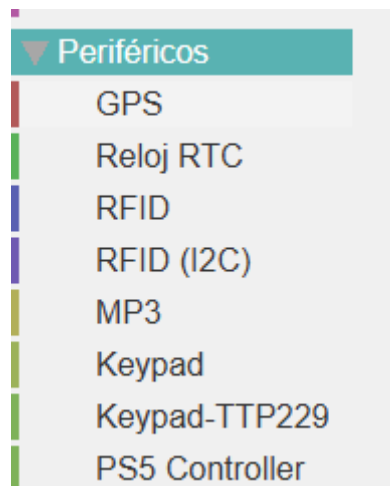
Y el resultado sería éste:



## Paneles IoT



Si dispusiéramos de un localizador GPS, tendríamos que localizar los bloques correspondientes en Periféricos → GPS





# Paneles IoT

## 1.4.11 Organizar elementos del panel

Una panel con todos los componentes vistos podría ser así:

PRUEBA TUTORIAL ARDUTALLER

MQTT Server: wss://broker.hivemq.com/mqtt Port: 8884 Estado: **Conectado**

Si queremos organizar los componentes de otra manera, podemos hacerlo en el menú de configurar los paneles.

Podemos mover los componentes, pulsando sobre ellos, y llevándolos a una nueva posición:

+	Type	Nombre	Descripción	Topic MQTT	End row	Actions	Bloques
☰	text_send	COMPONENTE ENVIAR TEXTO	Este componente nos permite enviar un texto desde el panel hasta la placa	/smb/90/404/	No	📄 🗑️	Pub block
☰	map	COMPONENTE MAPA COMPONENTE MAPA	Este componente nos permite mostrar sobre un mapa un posición GPS Este componente nos permite mostrar sobre un mapa un posición GPS	/smb/90/671/ /smb/90/671/	No	📄 🗑️	Pub block
☰	text	COMPONENTE TIPO TEXTO	Este componente muestra el valor de por ejemplo una variable texto almacenada en nuestra placa	/smb/90/256/	No	📄 🗑️	Pub block

Si pulsamos la columna más a la izquierda del panel, la que tiene un símbolo como este , nos permitirá mover el elemento.

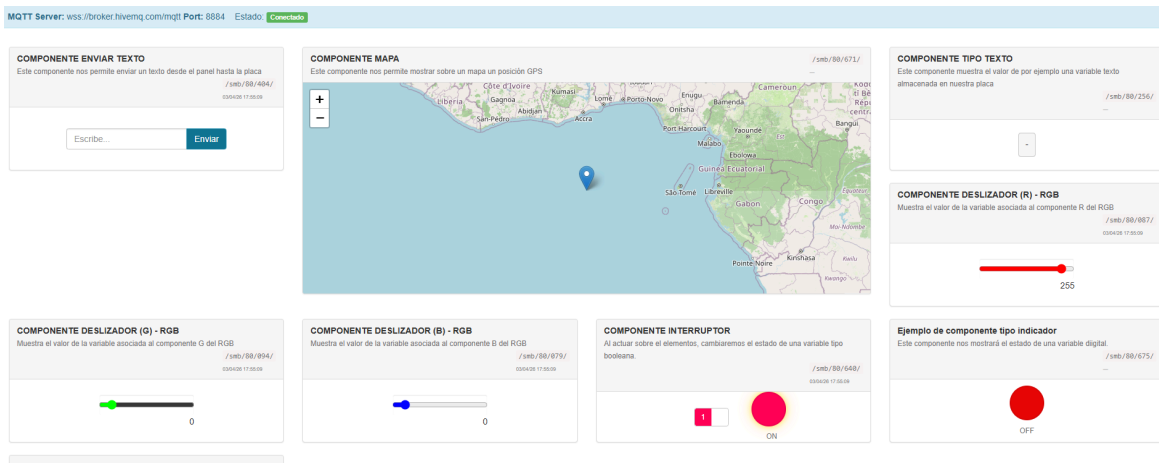
Una vez movido es muy importante que pulsemos abajo en



, y ya quedaría el panel con la nueva organización.



## Paneles IoT



## 2. Bibliografía

Para la realización de esta formación, se ha consultado la bibliografía que a continuación se expone. Agradecemos a sus autores, por la aportación que hacen a la comunidad con sus proyectos.

Paneles iot - publicado por Miguel Pérez para Robolot

[STEAMakersBlocks - FreeBook - ES](#) Manual oficial de STEAMakers blocks publicado por Juanjo López

[Guías fundamentales de Fede Coca](#) - GitHub oficial de Federico Coca

[Documentación de Proyecto Educacont 5.0](#) - Manuales publicados en el proyecto participado por Club Robótica Granada, IES Pinos Puente, Colegio Salesianos Juan XXIII

[Manuales varios de Ardutaller - SteamakersTaller](#) Manuales online editados por Francisco Soldado.

[Programación Bloques JSon con Arduinoblocks](#) Manual editado por Juanjo López

## Paneles IoT

---

[Telegram en Esp32STEMAakers](#) Manual editado por Juanjo López

[Taller Comunicaciones LoRa con ESP32 STEAMakers y Arduinoblocks-](#)  
Club Robótica Granada

[Análisis de datos en formatos Json](#) Juanjo Lopez y Arduinoblocks

[ESP32STEAMakers + Telegram bot](#) Juanjo Lopez y Arduinoblocks