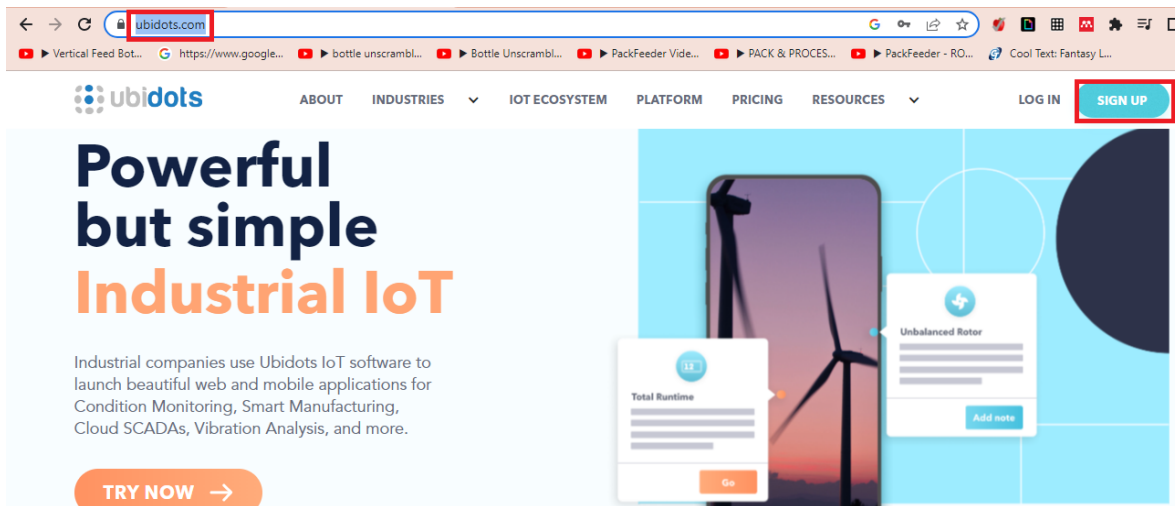


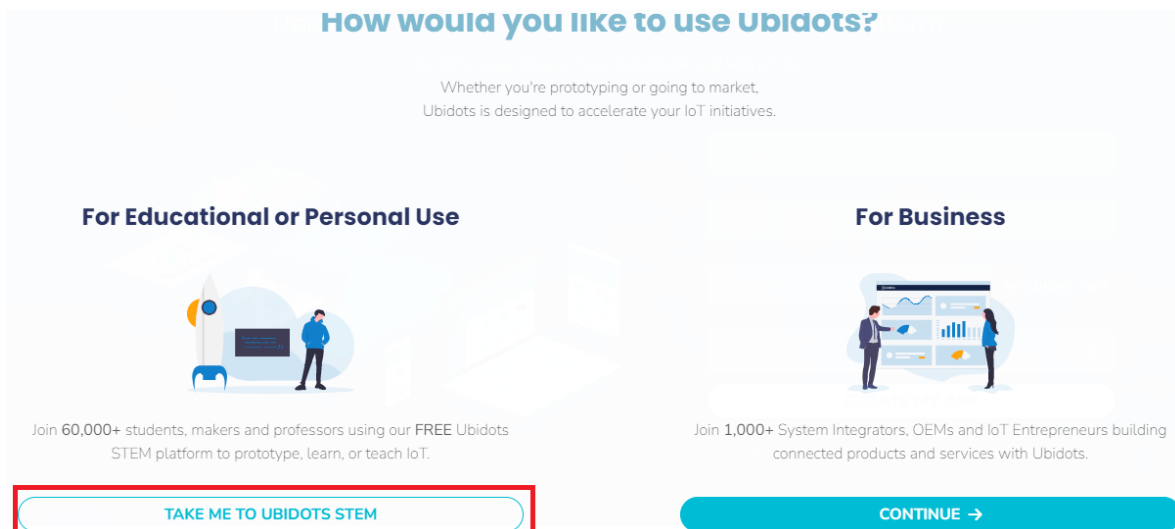
Ing. Alexander Vasco Orozco

EJEMPLO START – STOP CON UBIDOTS

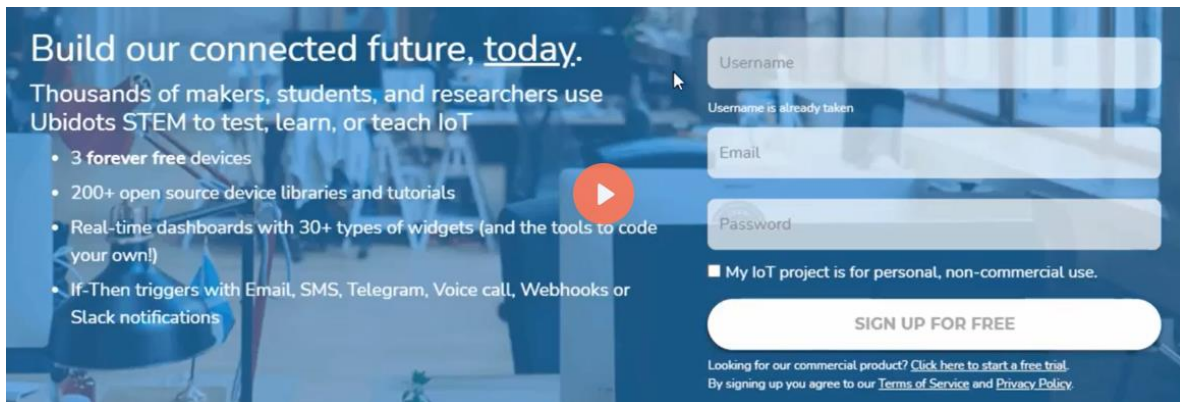
Cree una cuenta en la plataforma de IoT (Internet de las cosas) **UBIDOTS** que habilita la toma de decisiones a empresas de integración de sistemas a nivel global. Este producto permite enviar datos de sensores a la nube, configurar tableros y alertas, conectarse con otras plataformas, usar herramientas de analítica y arrojar mapas de datos en tiempo real. Para ello ingrese a la página <https://ubidots.com/> y seleccione la opción SIGN UP:



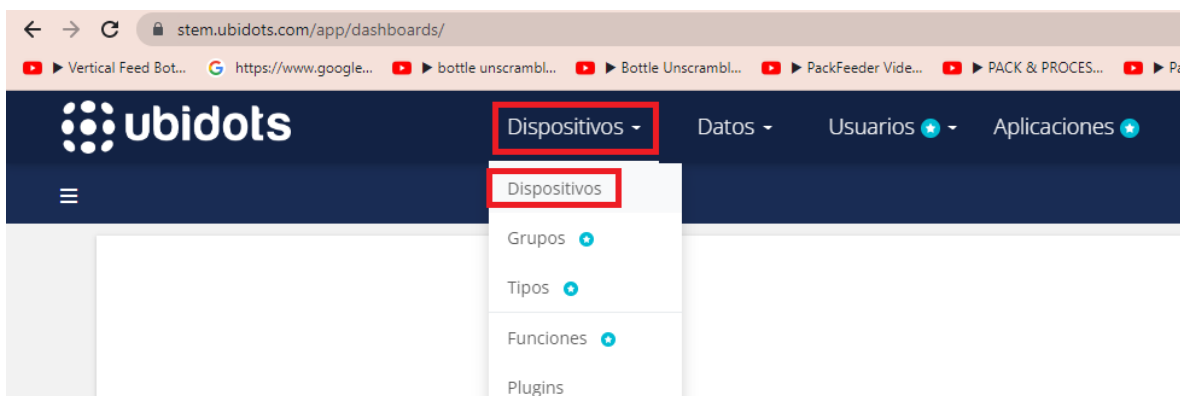
Seleccione el tipo de cuenta “Educational”:



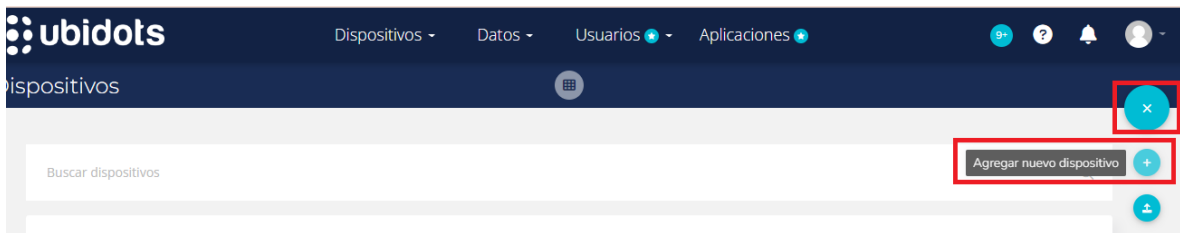
Diligencie los datos requeridos y (de ser necesario) confirme la cuenta desde el correo registrado:



Ahora agregue un nuevo dispositivo



Seleccione el símbolo “+” de la parte derecha y luego la opción agregar nuevo dispositivo



Del menú desplegable seleccione la opción “Dispositivo en blanco”:

Agregar Nuevo Dispositivo

Crea un nuevo Dispositivo en blanco, o crea Dispositivos automáticamente al usar las librerías y documentación que encuentras abajo. La primera vez que envías un dato a Ubidots, un nuevo Dispositivo se crea automáticamente. [CERRAR](#)

CONNECTIVITY

Ethernet Cellular LoRaWAN

LTE-M NB-IoT Sigfox

WiFi

HARDWARE TYPE

Chips & Modules Dev Kits

Gateways Production Ready

INTEGRATION TYPE

Library Plug-n-Play Tutorial



Dispositivo en blanco



Adafruit



Alorium Technology



Ambient Weather









Asigne el nombre que desee, en este ejemplo “Mi primer tablero” y presione la flecha verde para finalizar:


[← ATRÁS](#)

Nombre del dispositivo

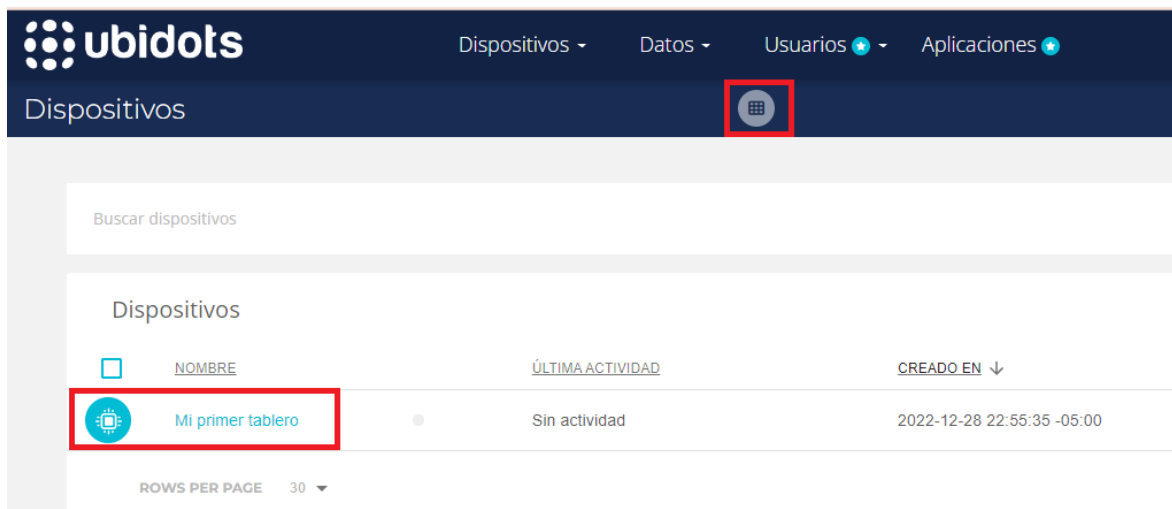
Mi primer tablero

Label del dispositivo

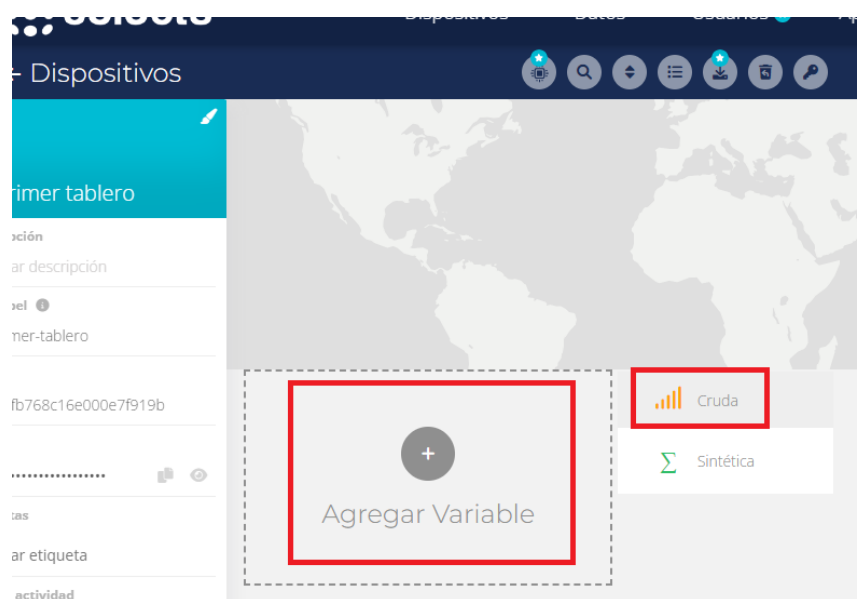
mi-primer-tablero

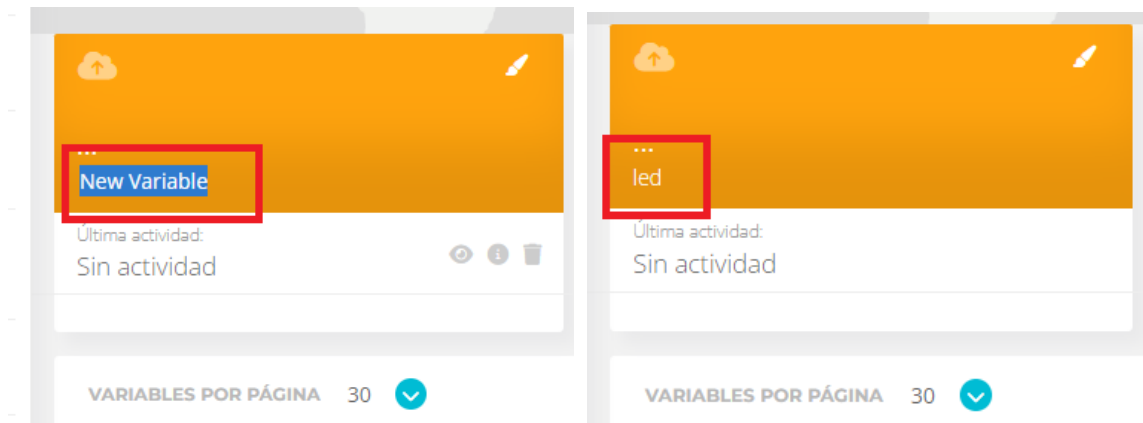


Todos los dispositivos creados se enlistarán y podrán ser filtrados por nombre, inclusive puede cambiar su apariencia con el recuadro dentro del círculo en la parte superior. Seleccione el elemento creado

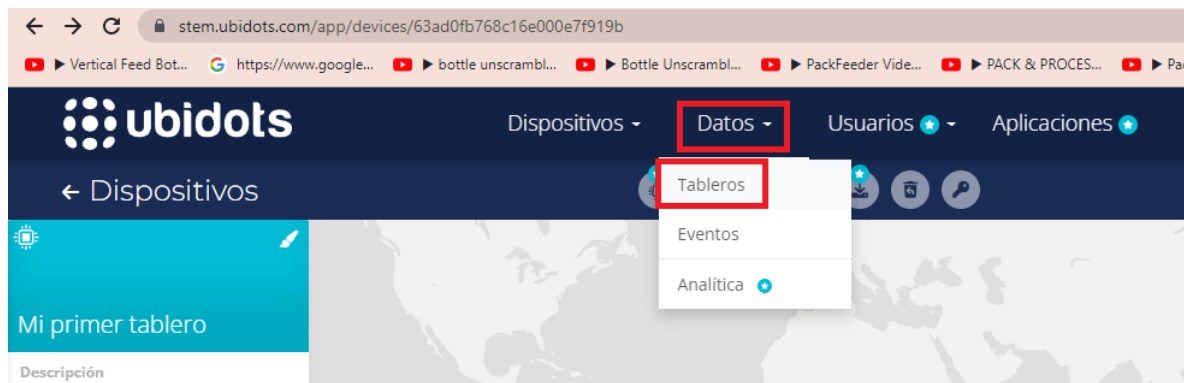


El primer ejemplo será encender un led. Para ello adicione presione clic en la opción “Agregar variable” y luego la opción “Cruda” y finalmente asigne el nombre led a dicha variable:

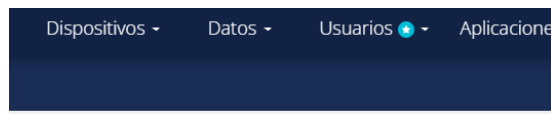




Ahora configure la interfaz del tablero, para ello seleccione el menú “Datos / Tableros” en la parte superior de la página:



Cree un nuevo tablero presionando en dicha opción:



Aún no se han creado Tableros

Crea Tableros para visualizar los datos en tiempo real



Por el momento solo cambie el nombre, puede asignar el mismo del dispositivo “mi primer tablero” y clic en GUARDAR:

Crear un nuevo tablero

CONFIGURACIONES APARIENCIA

Nombre mi primer tablero

Rango de tiempo por defecto Últimas 24 horas

Tablero Dinámico Actualiza los widgets con base en un dispositivo seleccionado Estático

Ancho Auto

Alineación Centro

CANCELAR GUARDAR

Ahora puede seleccionar entre multiples “Widgets”, para ver las diferentes opciones presione clic en “Agregar un nuevo Widget”:



En el menú desplegable seleccione el “Interruptor”, lastimosamente no existen los pulsadores en esta aplicación, por lo tanto se deben enclavar y desenclavar de manera programada para darles el funcionamiento de pulsador:



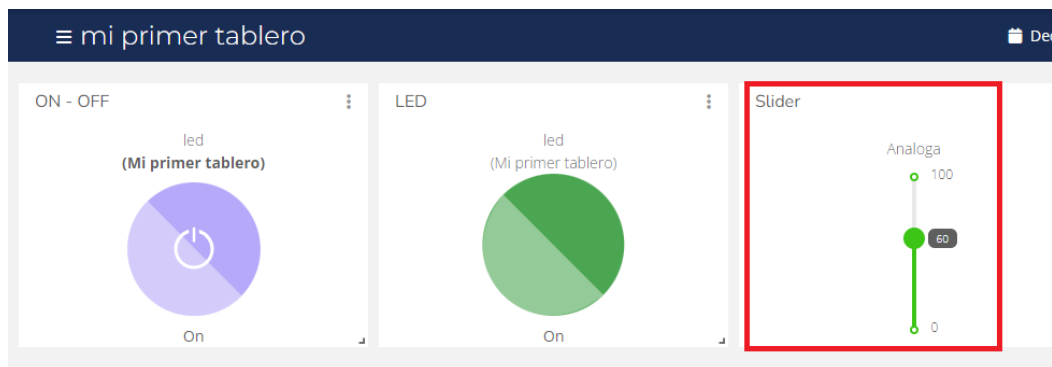
Asocie la variable que creo con anterioridad en la opción “AGREGAR VARIABLE” de la pestaña “CONFIGURACIONES”:



Ahora en la pestaña “APARIENCIA” asigne el nombre del interruptor, en este caso “ON - OFF”:



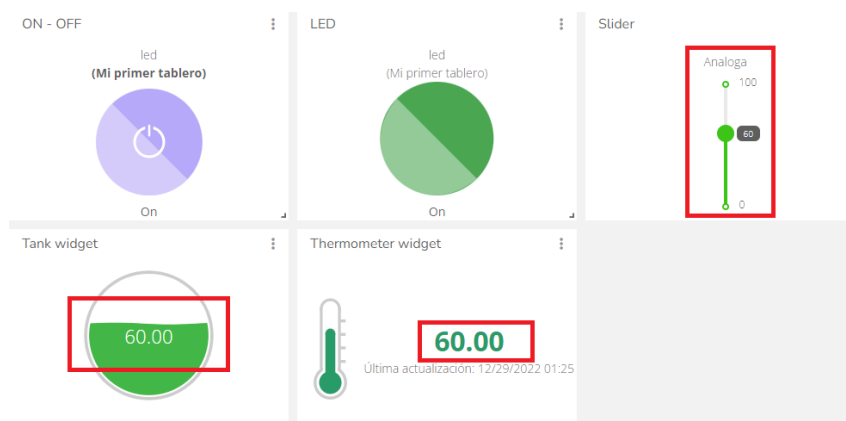
Asocie la variable “Analoga” y si desea cambie los valores mínimo y máximo del deslizador:



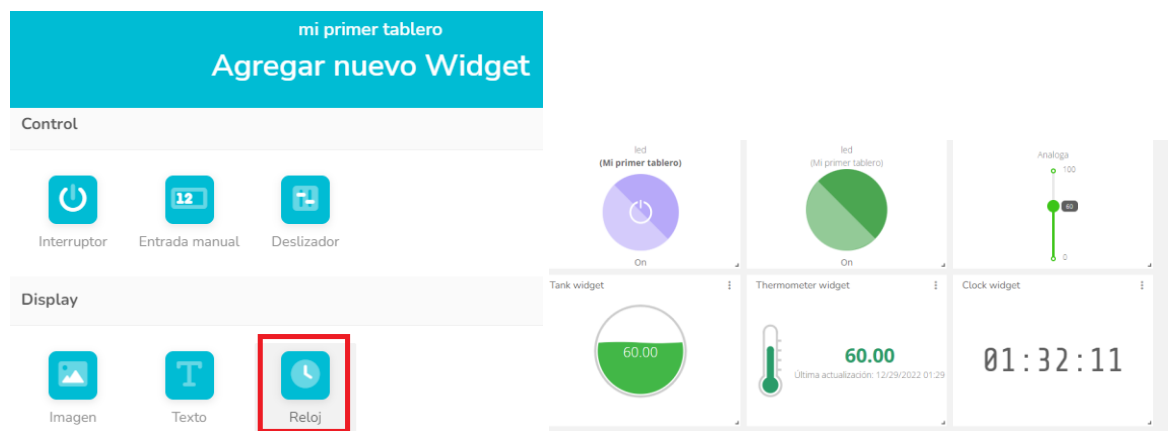
Ahora seleccione visualizadores de salida para esta variable, como son el tanque y el termómetro:



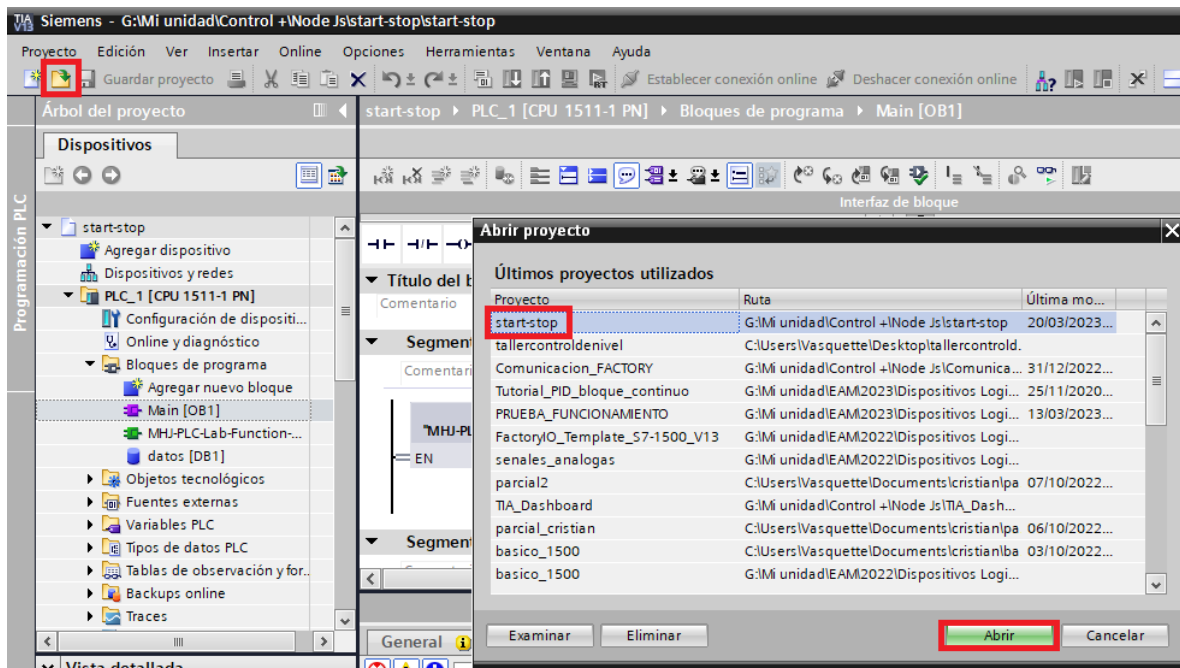
Asigne la variable y compruebe el funcionamiento



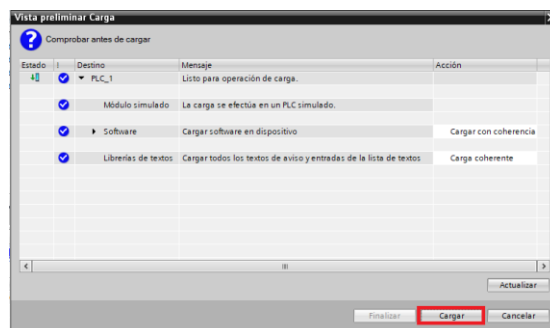
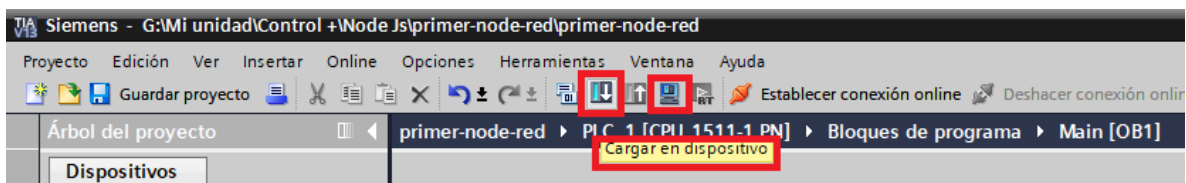
Agregue un Widget que no requiere variable como es el reloj (modifique los parámetros a su gusto), en este caso se seleccionó por defecto:



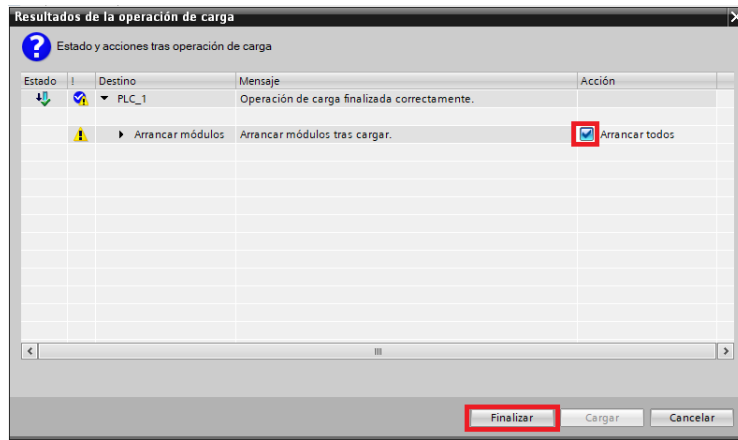
Ahora realice la comunicación con TIA Portal. Abra el archivo “start-stop” realizado en el tutorial anterior:



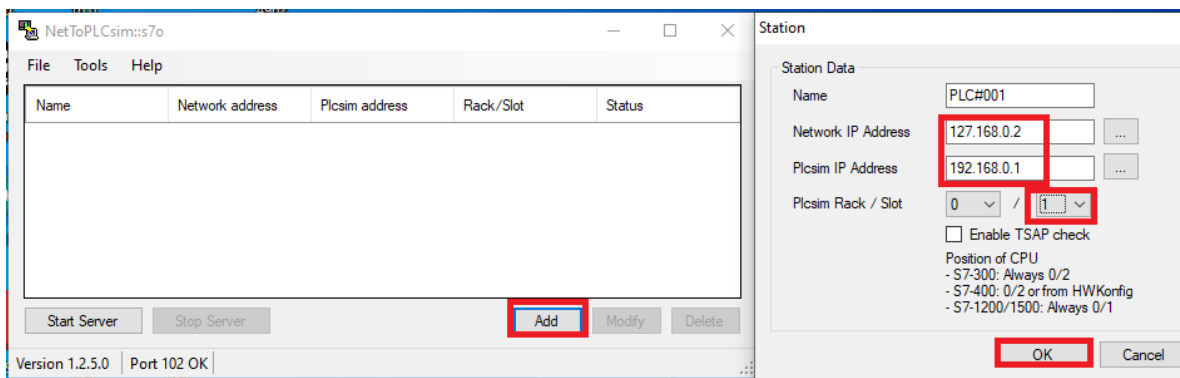
Cargue el programa al simulador S7-PLCSIM... si no lo tiene abierto haga clic en el símbolo de un “computador”, por el contrario (si lo tiene abierto) solo presione en el símbolo “flecha abajo” (cargar en dispositivo) y en la ventana emergente clic en “Cargar”:



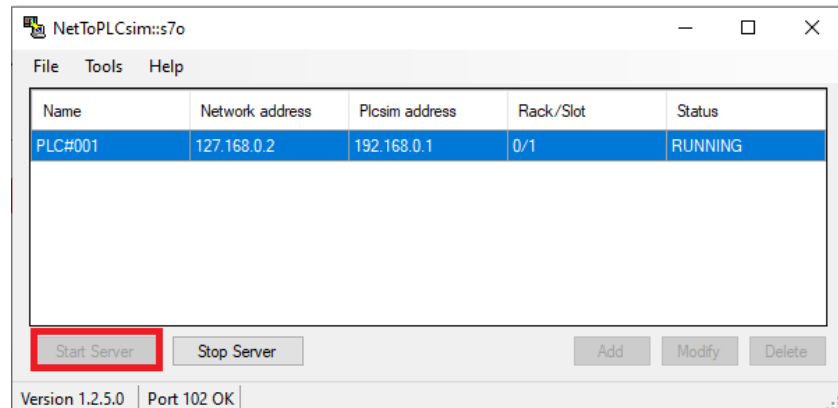
En la última ventana asegúrese de tener el check box “Arrancar todos” activo y clic en “Finalizar”:



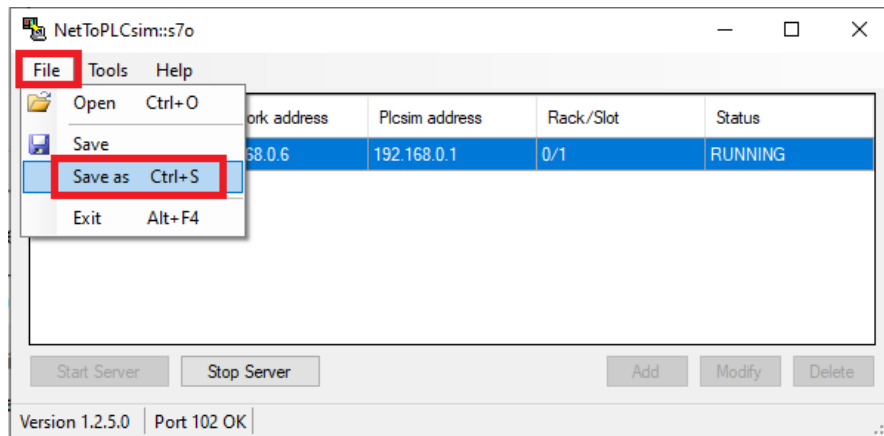
Ejecute el programa NetToPLCsim como administrador, abra una estación creada con anterioridad o de lo contrario cree una nueva estación en el botón “Add” y asigne cualquier dirección para el servidor, en este caso “Network IP Address: 127.168.0.2” y la dirección del simulador “Plcsim IP Address: 192.168.0.1”, y cambie el Slot al número “1”:



Inicie el servidor en el botón “Start Server”:



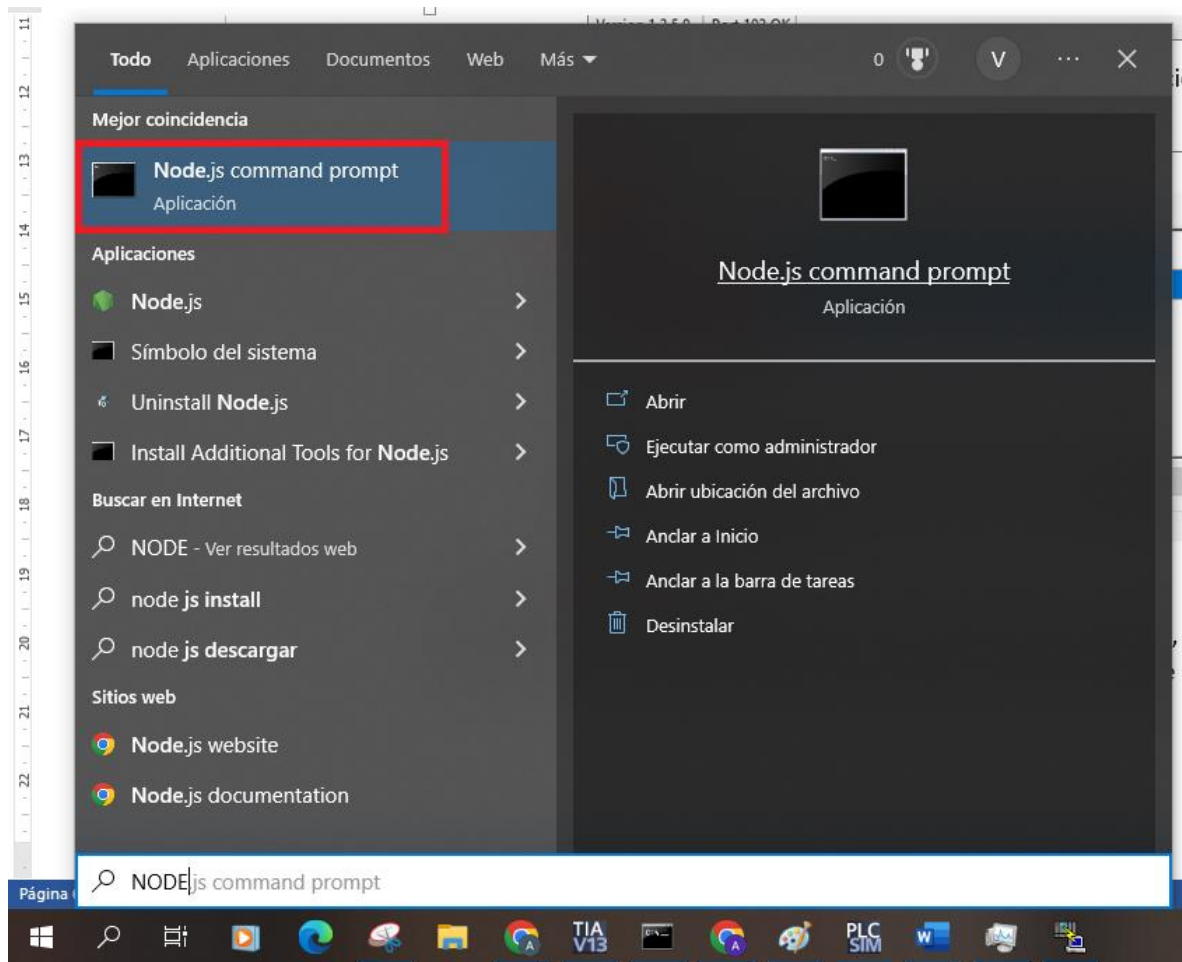
Guarde el servidor para no tener que volverlo a configurar, en la opción “File /Save as” y asígnele un nombre:



Abra el montaje del ejemplo “start-stop” en el software Factory IO que contiene una banda, un emisor de cajas y el tablero de control y realice la conexión con el PLCSim:



Ahora ejecute Node-red mediante el comando Node.js command prompt en el inicio de windowa:



Ejecute el comando `node-red` y copie la dirección del servidor, en este caso es: `http://127.0.0.1:1880/` y luego abra un navegador web como Google Chrome.

```
cmd Selecccionar node-red
25 Nov 09:22:39 - [info] Context store : 'default' [module=memory]
25 Nov 09:22:39 - [info] User directory : C:\Users\Vasquette\.node-red
25 Nov 09:22:39 - [warn] Projects disabled : editorTheme.projects.enabled=false
25 Nov 09:22:39 - [info] Flows file : C:\Users\Vasquette\.node-red\flows.json
25 Nov 09:22:39 - [info] Creating new flow file
25 Nov 09:22:39 - [warn]

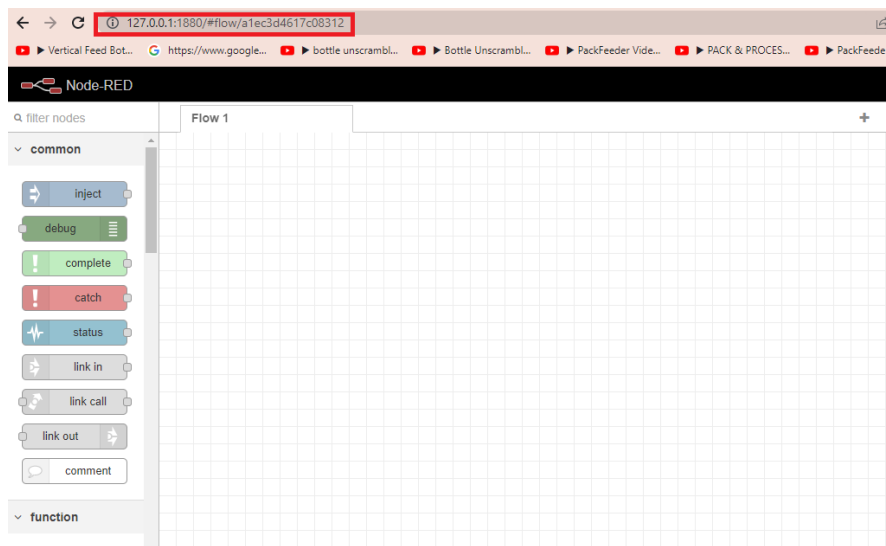
-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

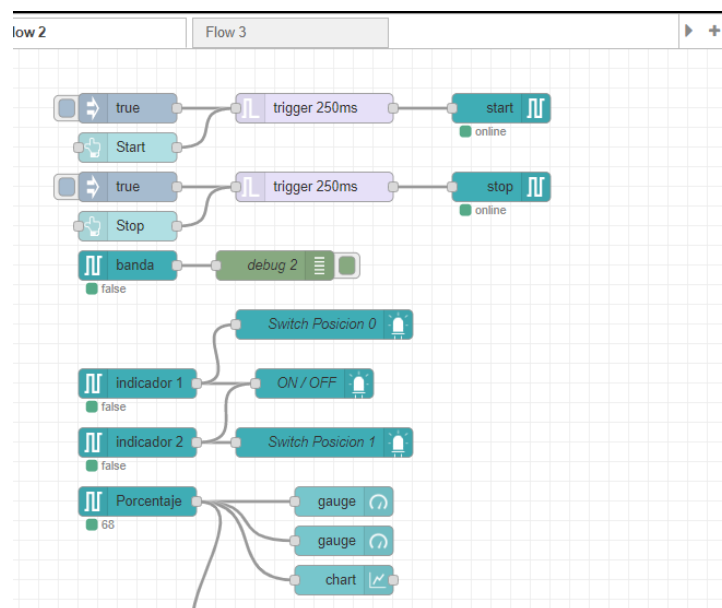
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

25 Nov 09:22:39 - [info] Server now running at http://127.0.0.1:1880/
25 Nov 09:22:39 - [warn] Encrypted credentials not found
25 Nov 09:22:39 - [info] Starting flows
25 Nov 09:22:39 - [info] Started flows
```

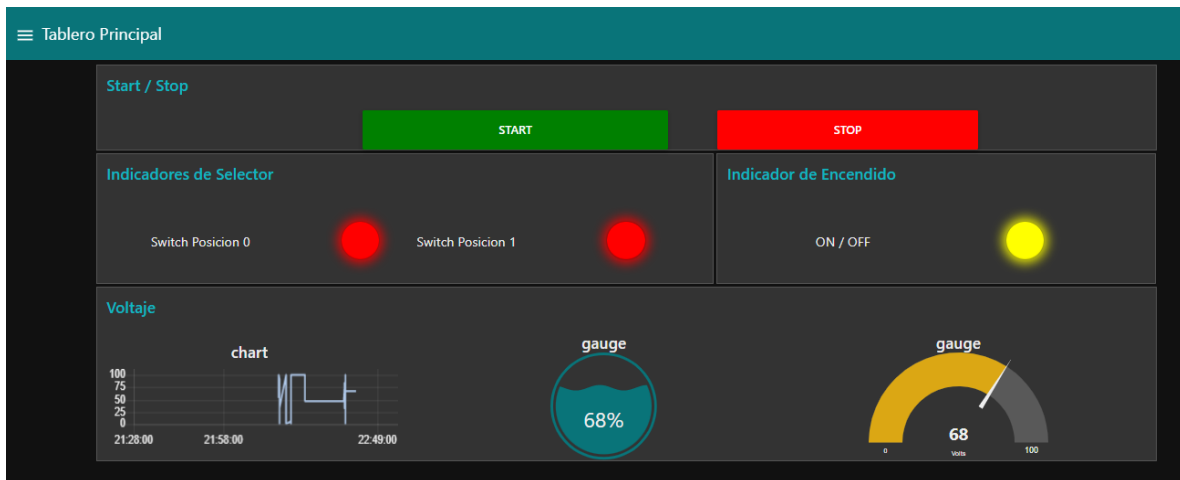
Pegue la dirección para ingresar al entorno de programación:



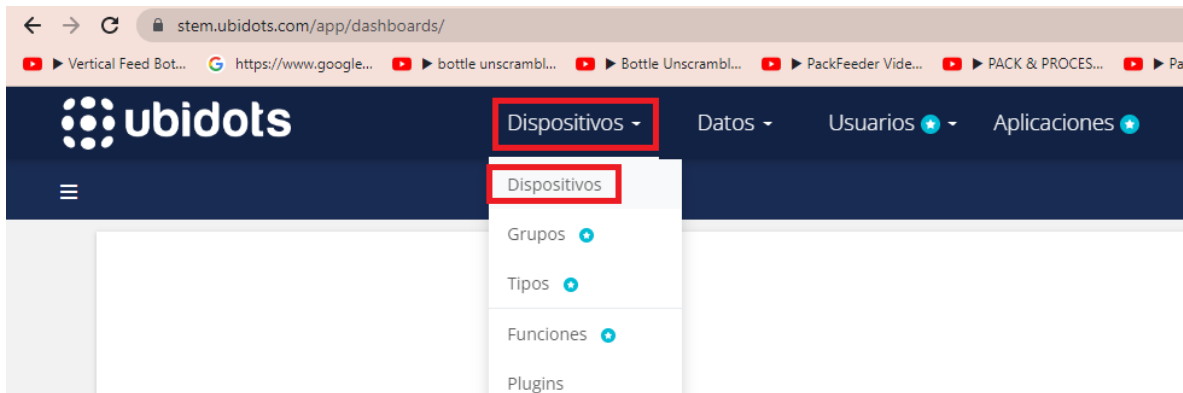
Seleccione el flujo que contiene la programación para el ejemplo “start - stop”



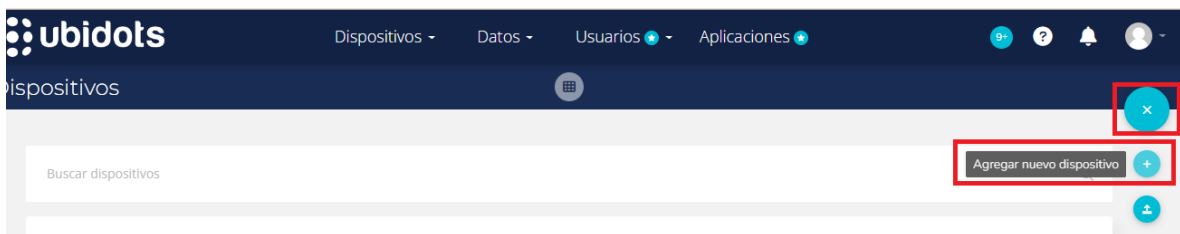
Verifique el adecuado funcionamiento del dashboard:



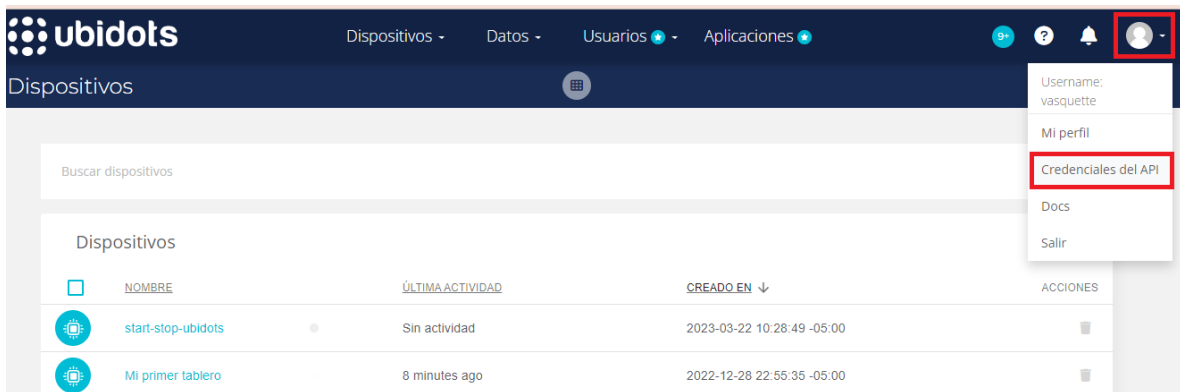
Abra la plataforma de UBIDOTS (<https://ubidots.com/>) y agregue un nuevo dispositivo:



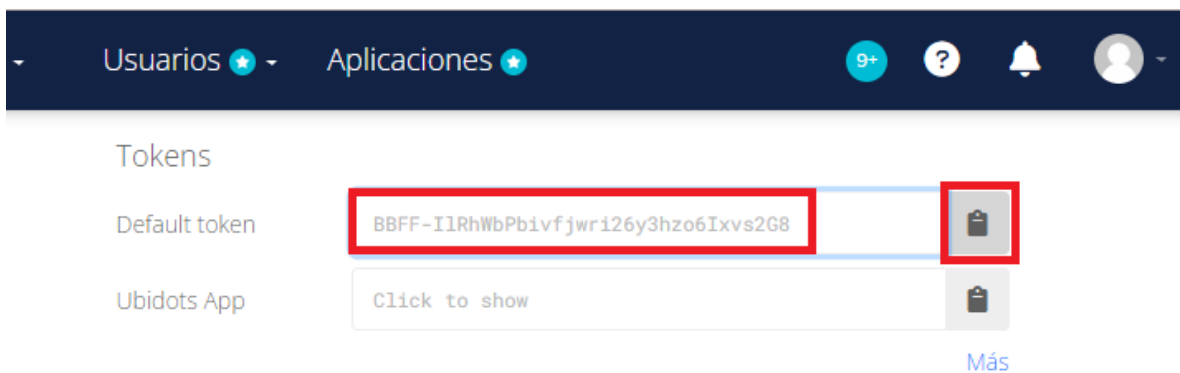
Seleccione el símbolo “+” de la parte derecha y luego la opción agregar nuevo dispositivo



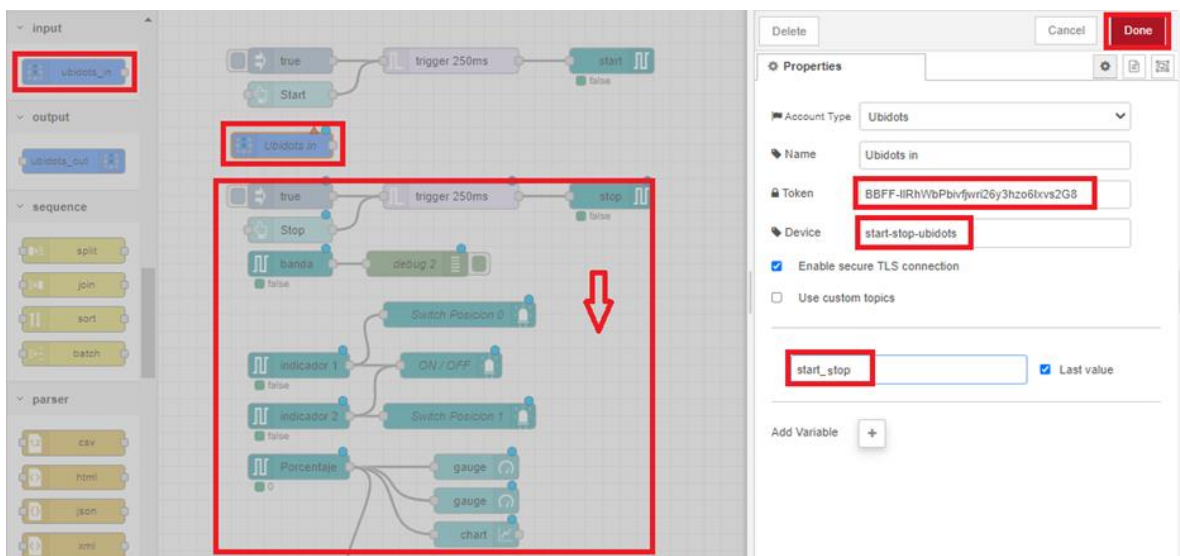
Del menú desplegable seleccione la opción “Dispositivo en blanco”:



Presione clic sobre la barra “Default token” para mostrar la credencial y luego presione en el símbolo “📋” para copiarlo (NO funciona ctrl+c):



Regrese a node-red, abra espacio y posicione el bloque “ubidots_in”; configúrelo pegando (ctrl+v) la información copiada desde ubidots en la opción “Token”, y en la opción “Device” coloque el nombre del dispositivo creado en ubidots “start-stop-ubidots”, finalmente asigne la variable “start_stop” como muestra la figura:



Para enviar datos a Ubidots tenga en cuenta el siguiente concepto:

Concepto básico de los objetos JavaScript

La creación de un objeto a menudo comienza con la definición e iniciación de una variable.

```
var nombreObjeto = {};
```

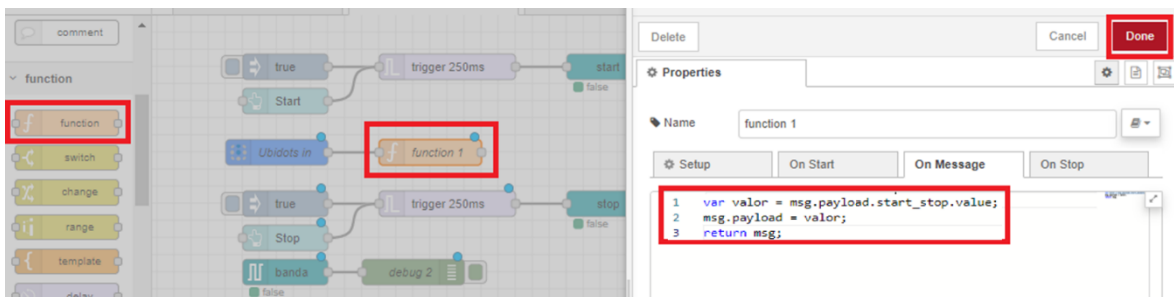
Un objeto se compone de varios miembros, cada uno de los cuales tiene un nombre (por ejemplo, nombre y edad) y un valor (por ejemplo, ['David', 'Smith'] y 32). Cada par nombre/valor debe estar separado por una coma, y el nombre y el valor en cada caso están separados por dos puntos. La sintaxis siempre sigue este patrón:

```
var nombreObjeto = { miembro1Nombre: miembro1Valor, miembro2Nombre: miembro2Valor }
```

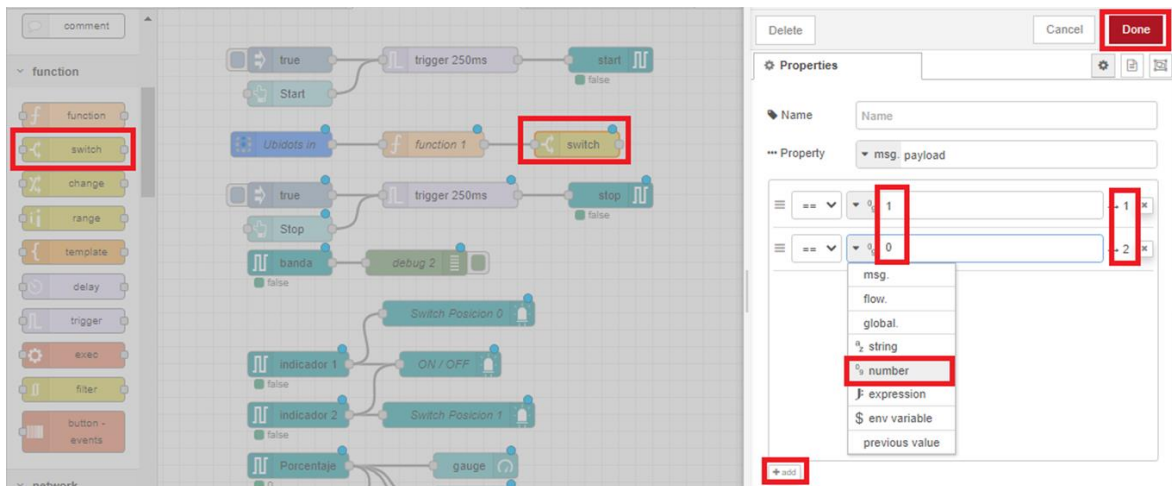
```
var nombreObjeto = {  
    miembro1Nombre: miembro1Valor,  
    miembro2Nombre: miembro2Valor  
}
```

Agregue un bloque "Function" donde se debe transformar la información que llega desde ubidots en forma de objeto JavaScript a un formato "true or false" que maneja node-red:

```
var valor = msg.payload.start_stop.value;  
msg.payload = valor;  
return msg;
```

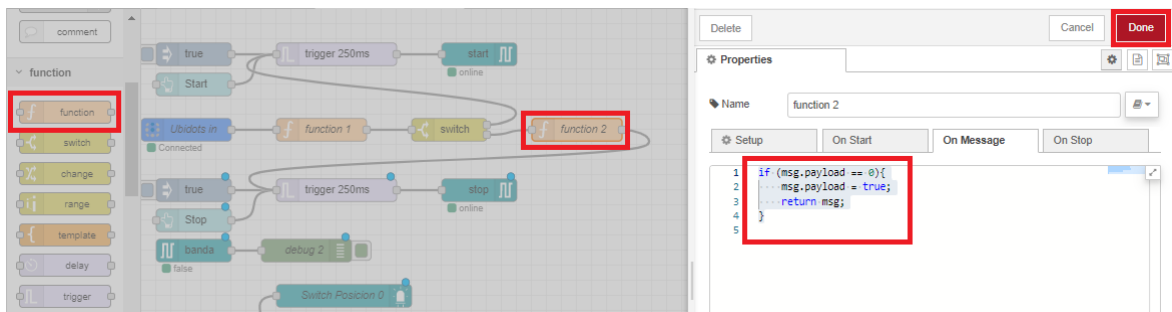


Como ya se había mencionado, Ubidots no tiene pulsadores, solo interruptores (estados enclavados), por lo tanto, con uno solo se va a comandar el start y el stop del node red con cada uno de sus estados, para ello posicione un bloque "Switch" y configúrelo para separar ambos estados y redireccionarlos al trigger correspondiente:

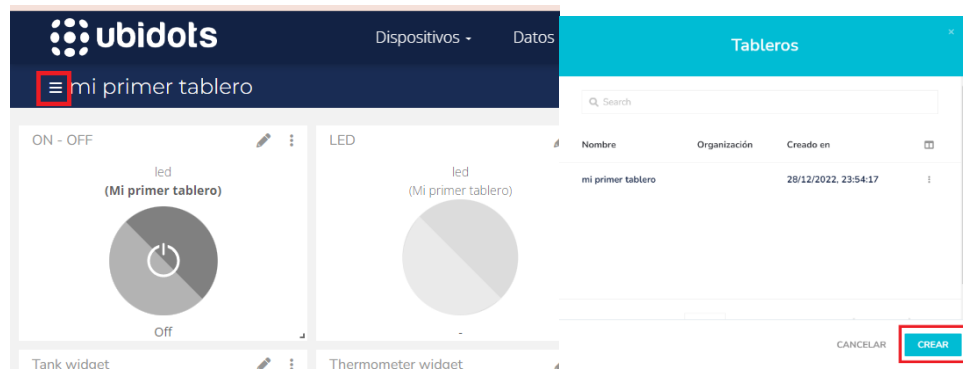


Es de notar que ambos “trigger” se activan con un true (1 lógico) y del switch por la salida 1, se genera un 1 que se dirige hacia el trigger del start, pero por la salida 2 se genera un 0, por lo tanto, es necesario cambiarlo a un estado true (1) mediante un bloque “function”. Finalmente, una los bloques como muestra la figura:

```
if (msg.payload == 0){
  msg.payload = true;
  return msg;
}
```

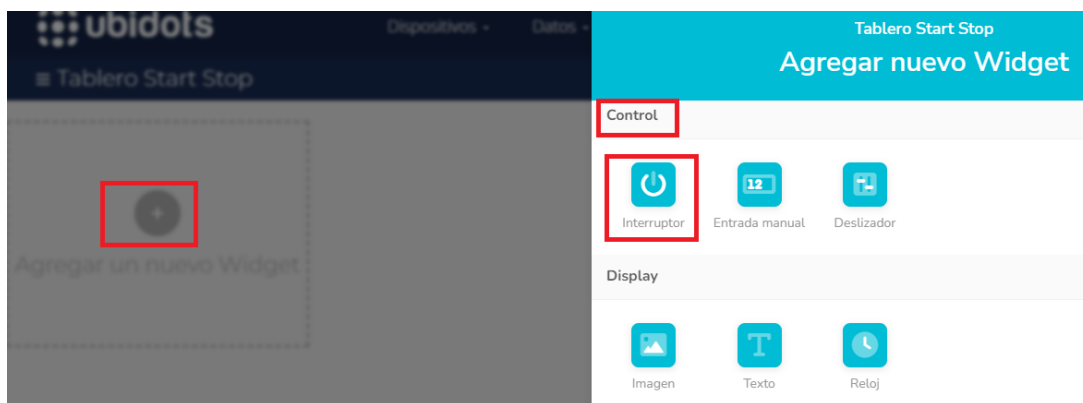


Retorne a Ubidots e ingrese al dispositivo creado “start-stop-ubidots”:

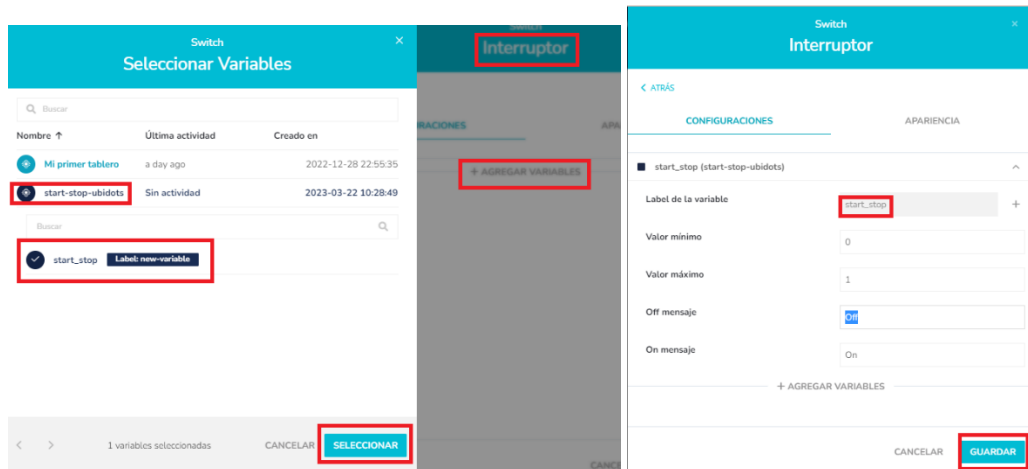


Asigne el nombre “Tablero Start Stop” y luego “GUARDAR”:

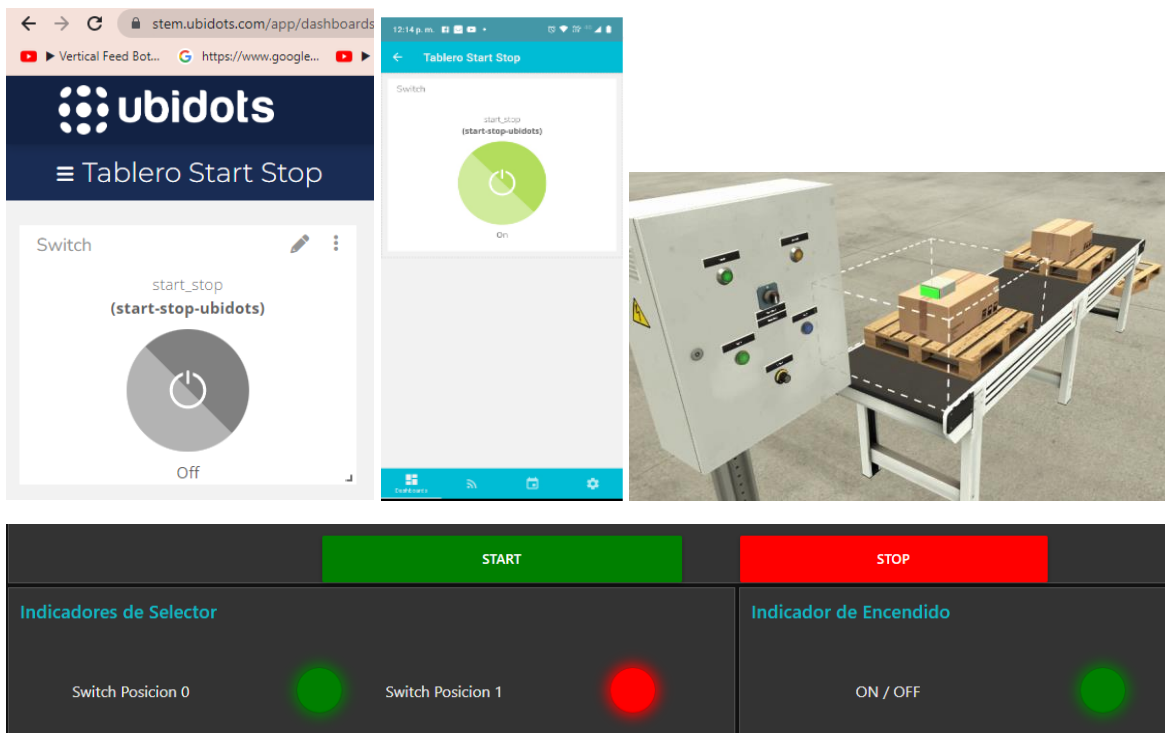
Agregue un nuevo widget, de la librería control seleccione el “Interrupor”



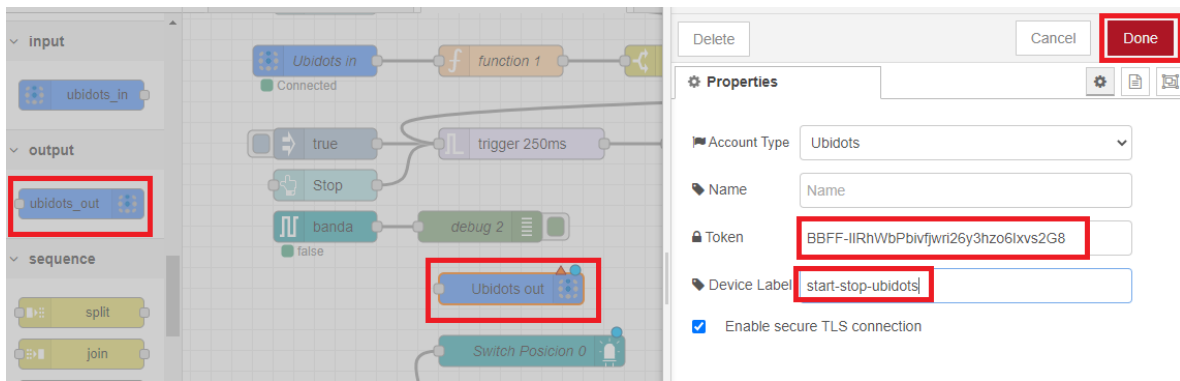
Presione en “AGREGAR VARIABLES”, y en la ventana emergente seleccione el tablero y la variable correspondiente y clic en seleccionar. La configuración final debe quedar como la figura de la derecha:



Compruebe el funcionamiento del interruptor desde el pc o la aplicación móvil, pues ya está conectado a la nube y puede ser modificado desde cualquier parte del mundo. Ubidots envía y recibe datos del node-red y por si solo no puede controlar nuestros dispositivos, por lo tanto cada vez que interactúe con esta plataforma, esta modifica el node-red al igual que su dashboard e igualmente el Factory (o el equipo físico real):

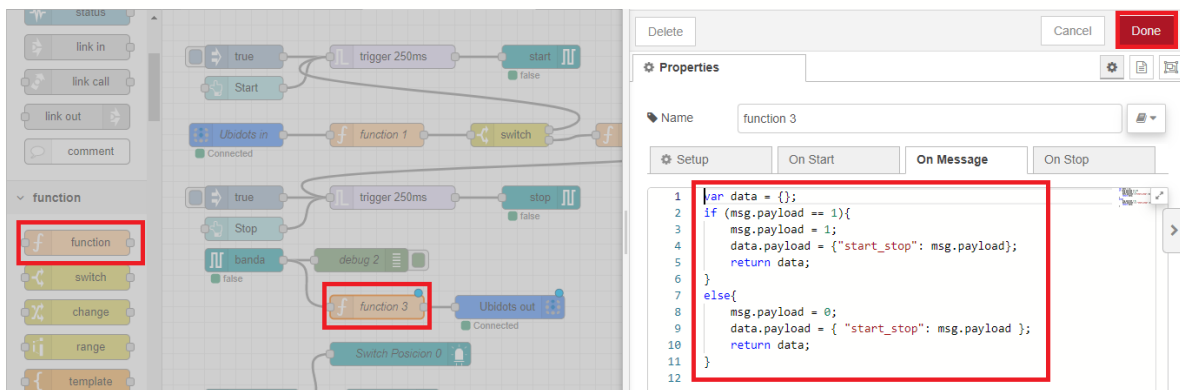


Por el momento funciona bien desde ubidots hacia node-red, pero no en viceversa, para ello es necesario cambiar el estado del interruptor desde un bloque del node-red que corresponda a esta dinámica (para este caso se usara el bloque s7-out “banda”). Agregue un bloque “Ubidots_out” y coloque el mismo “Token” y “Device Label” que en el bloque “Ubidots_in”:

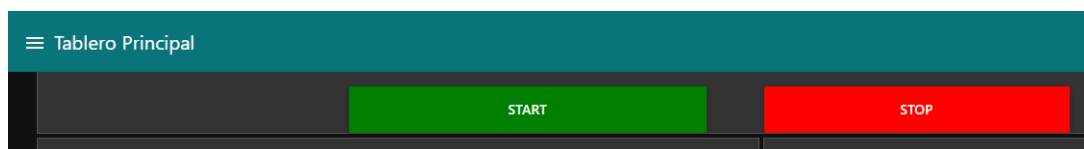


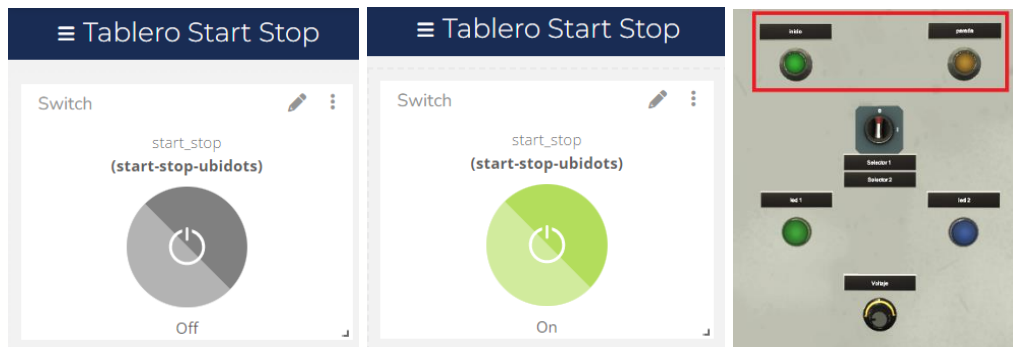
Posicione un bloque "Function" para transformar el estado lógico (0 o 1) del bloque "banda" en un objeto JavaScript y modificar el estado del interruptor en Ubidots, escribiendo el siguiente código:

```
var data = {};
if (msg.payload == 1){
    msg.payload = 1;
    data.payload = {"start_stop": msg.payload};
    return data;
}
else{
    msg.payload = 0;
    data.payload = { "start_stop": msg.payload };
    return data;
}
```

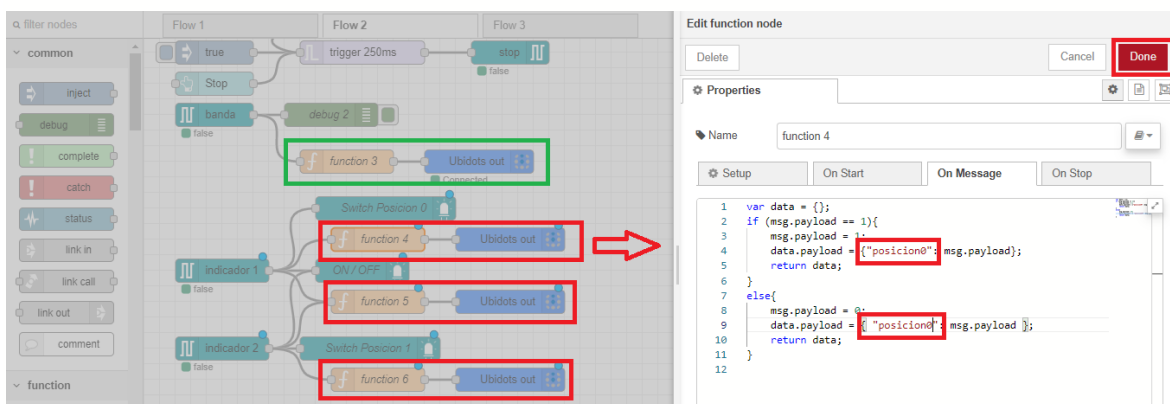


Verifique el funcionamiento desde el dashboard de node-red o desde el Factory, cada vez que presione los pulsadores start o stop, se debe actualizar el interruptor de Ubidots:

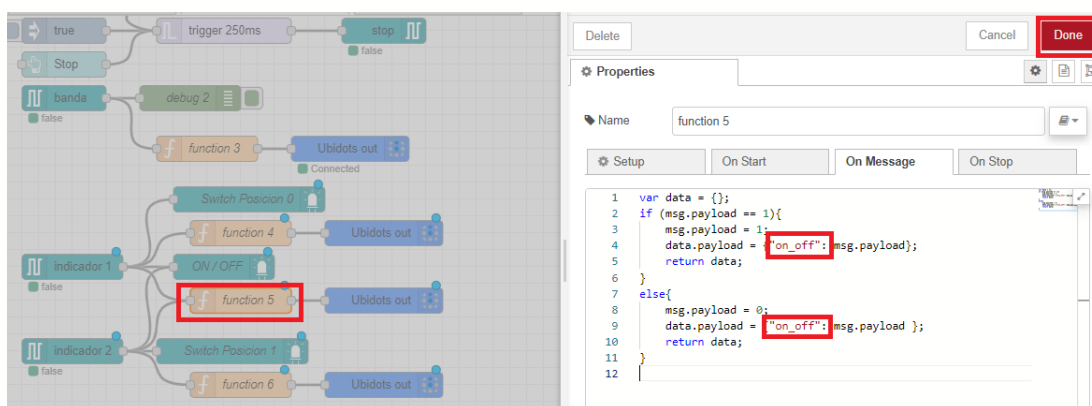




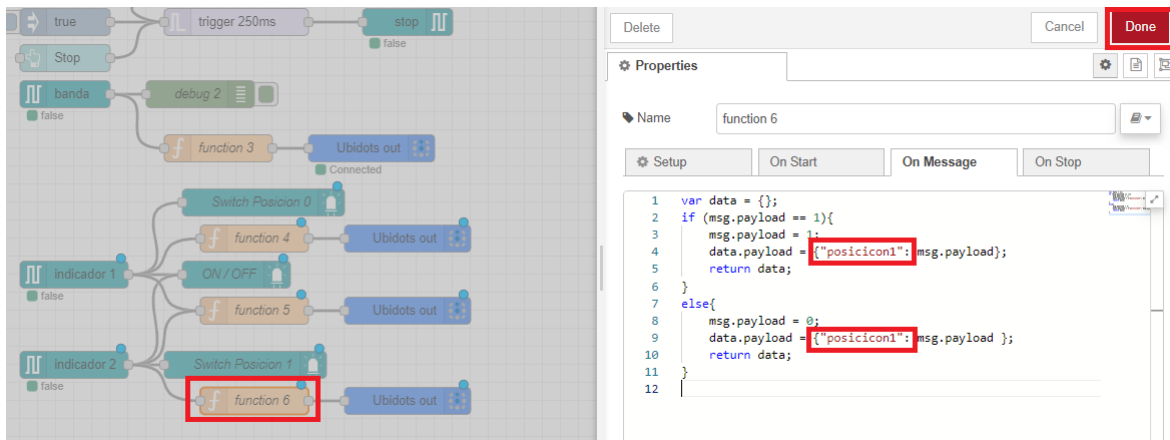
Copie los bloques “function” y “ubidots_out” y genere 3 copias que trabajaran en paralelo a cada uno de los bloques “LED”. A cada bloque “Function” le deben llegar las mismas líneas de conexión que a su respectivo bloque “LED”. Configure el bloque “Function 4” cambiando la variable a escribir por “posicion0” que se activará cuando se seleccione el “Switch Posición 0”:



Para el bloque “Function 5” cambie la variable a escribir por “on_off” que se activará cuando se seleccione el “ON/OFF”



Para el bloque “Function 6” cambie la variable a escribir por “posicion1” que se activará cuando se seleccione el “Switch Posición 1”

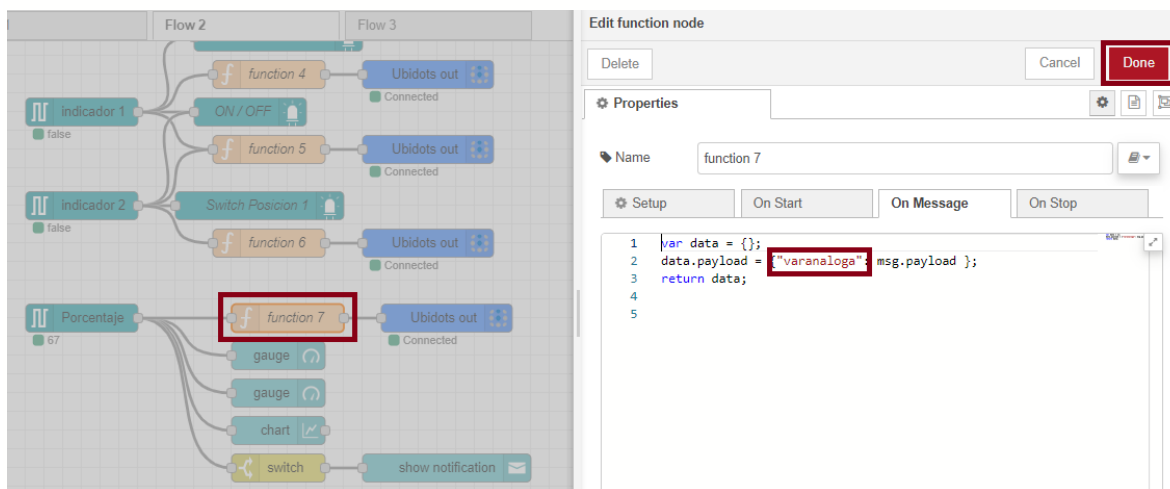


Realice el mismo procedimiento para la variable “porcentaje”, con la diferencia que esta variable no es de 2 estados, por lo tanto, se enviará de manera continua en todo momento, asígnele el nombre “varanaloga”:

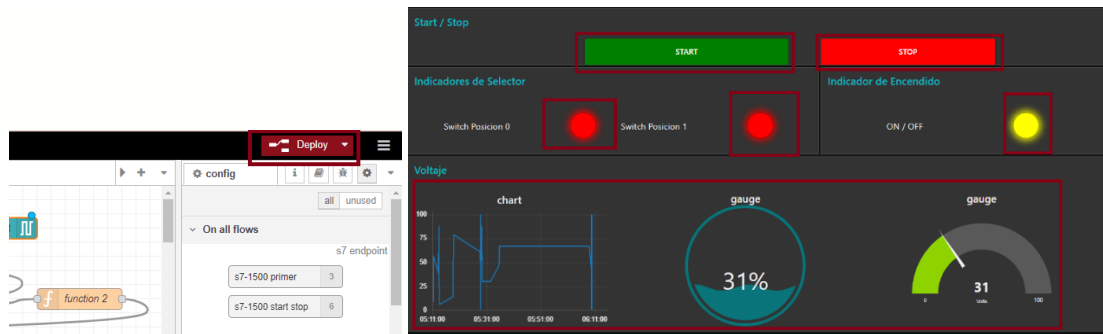
```

var data = {};
data.payload = {"varanaloga": msg.payload };
return data;

```



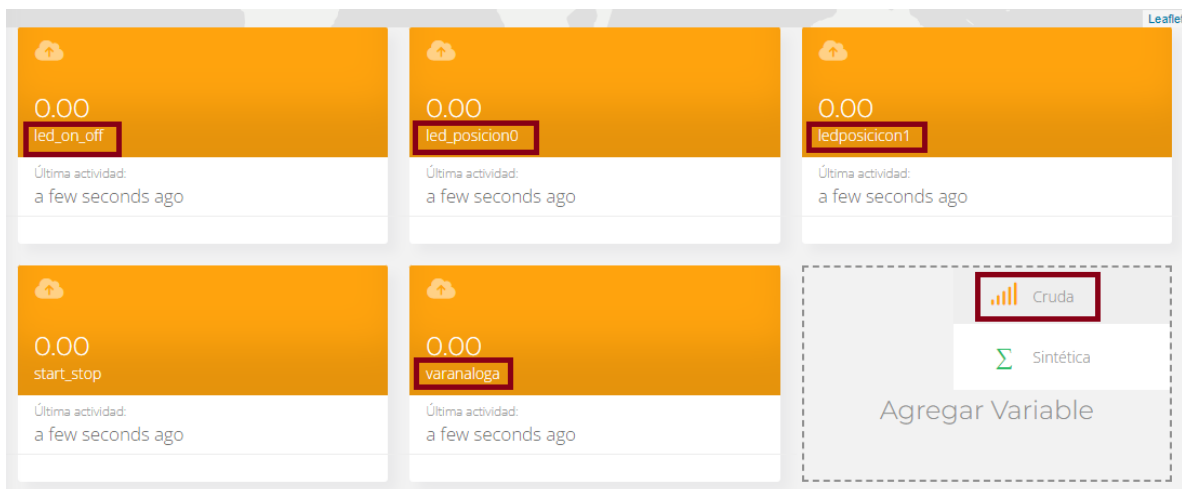
Presione en el botón “Deploy” para actualizar todos los cambios y utilice una vez más la interfaz en dashboard de node-red asegurándose de utilizar todos los elementos (incluyendo el switch y el potenciómetro desde Factory), puesto que cada vez que una de las variables de los nuevos bloques de Ubidsots sea utilizado, esto generará la creación de esta variable de manera automática en dicha plataforma:



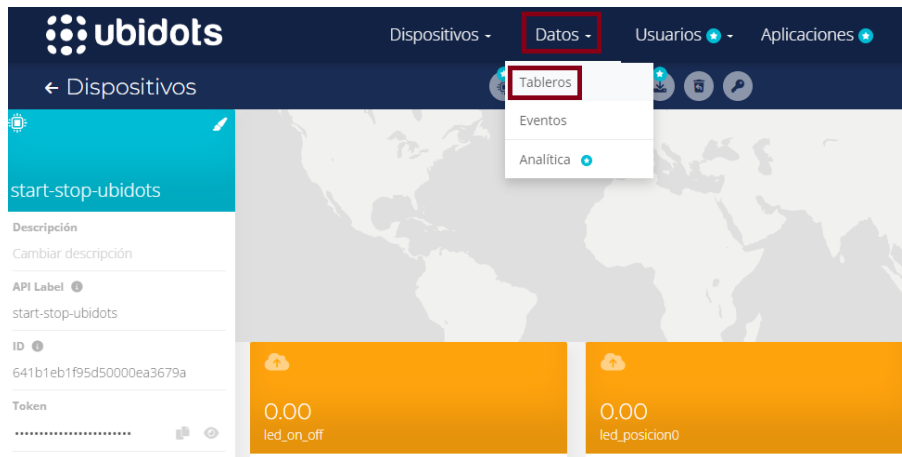
Retorne a Ubidots e ingrese al dispositivo creado “start-stop-ubidots”:

Dispositivos			
Buscar dispositivos			
Dispositivos			
<input type="checkbox"/>	NOMBRE	ÚLTIMA ACTIVIDAD	CREADO EN ↓
<input checked="" type="checkbox"/>	start-stop-ubidots	Sin actividad	2023-03-22 10:28:49 -05:00
<input type="checkbox"/>	Mi primer tablero	11 hours ago	2022-12-28 22:55:35 -05:00

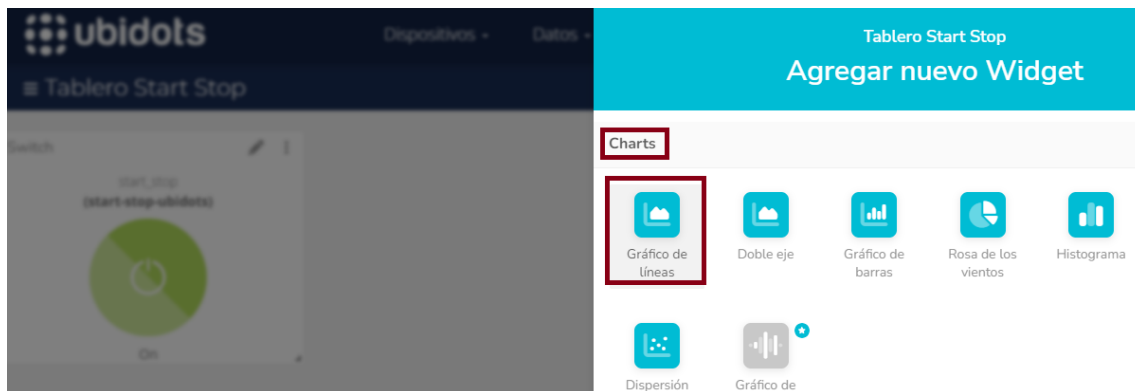
En este punto, las variables generadas en el node-red deben haberse cargado de manera automática en las variables de Ubidots, de no ser así (o si falta alguna) créelas de manera manual, seleccione la opción “Cruda” cuando agregue nuevas variables:



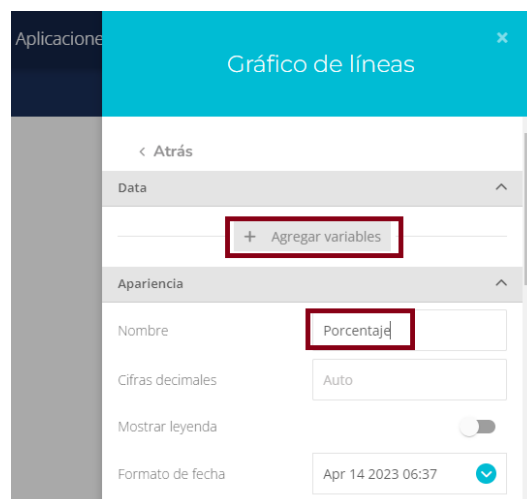
En la parte superior despliegue el menú “Datos” y seleccione la opción “Tableros”:



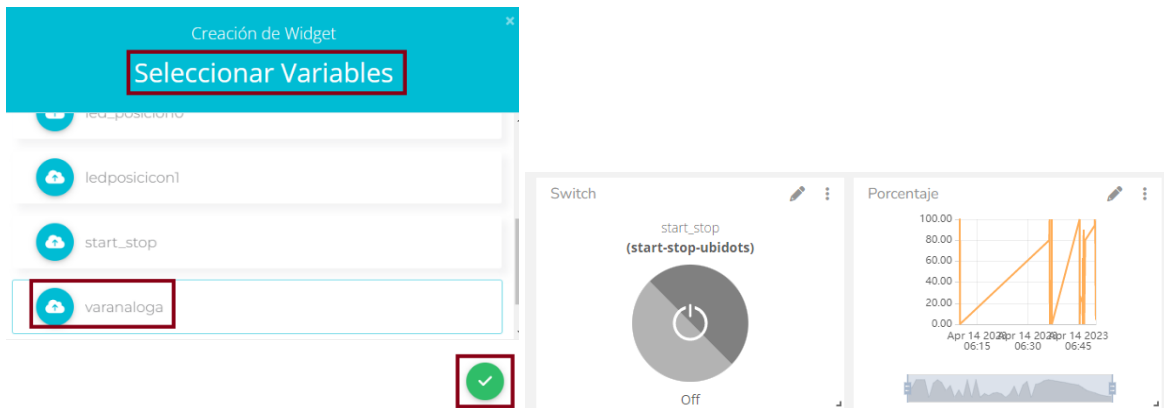
Agregue un nuevo “Widget” tipo “Gráfico de líneas” de la librería “Charts”:



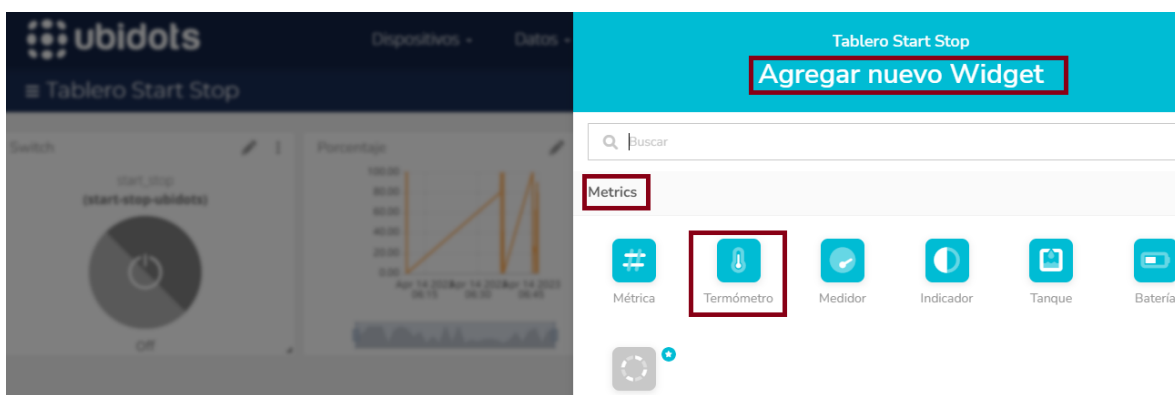
En la ventana emergente, asigne el nombre “porcentaje” y luego presione sobre el botón “Agregar variables”:



En la nueva ventana emergente, seleccione la variable continua “varanaloga” y finalice las ventanas desde el símbolo “✓”:



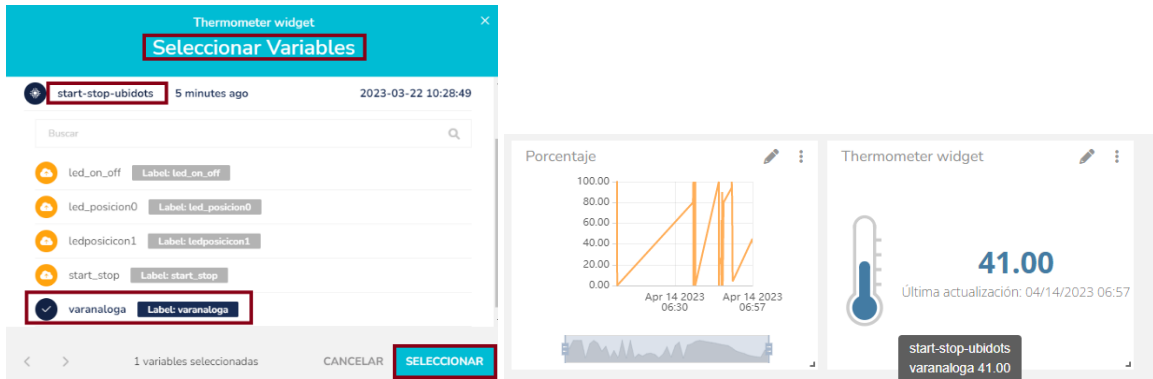
Agregue un nuevo “Widget” tipo “termómetro” de la librería “Metrics”:



En la ventana emergente, presione sobre el botón “Agregar variables”:



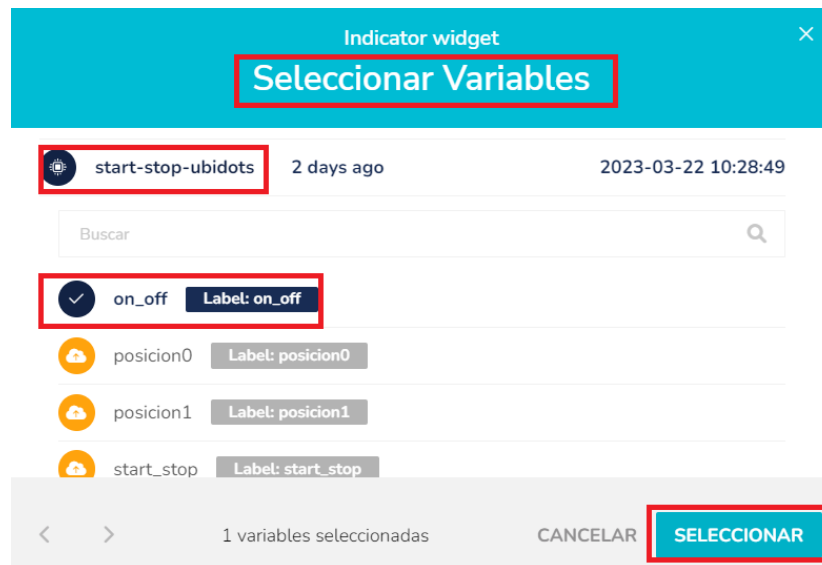
En la nueva ventana emergente, ubique y seleccione la variable continua “varanalog” y finalice desde el botón “SELECCIONAR”:



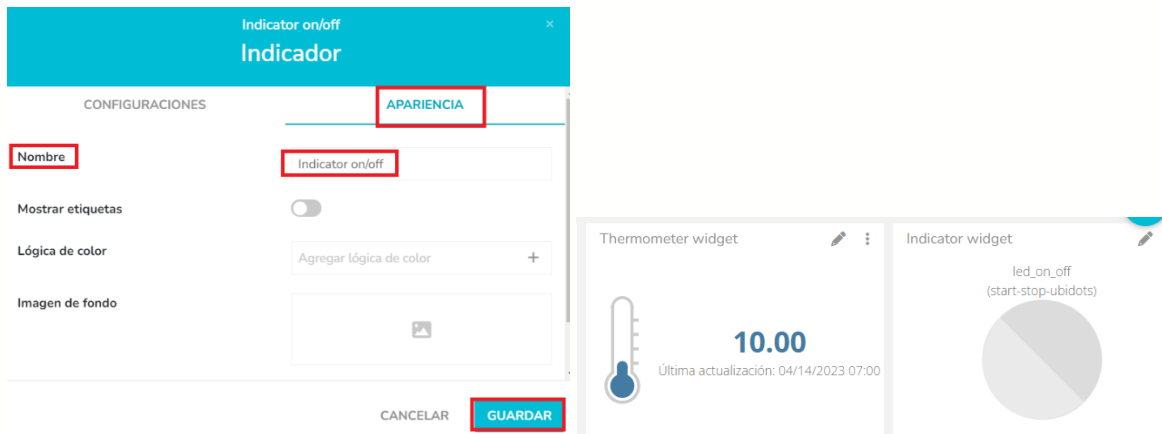
Agregue un nuevo “Widget” tipo indicador de la librería “Metrics”:



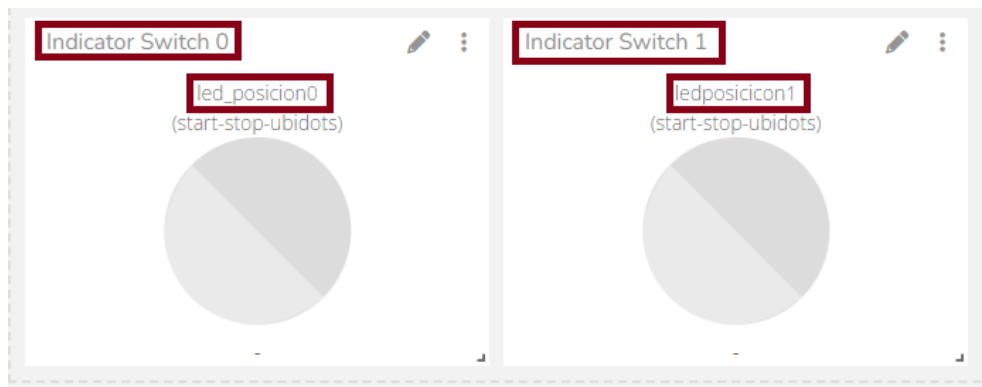
Seleccione la variable on_off:



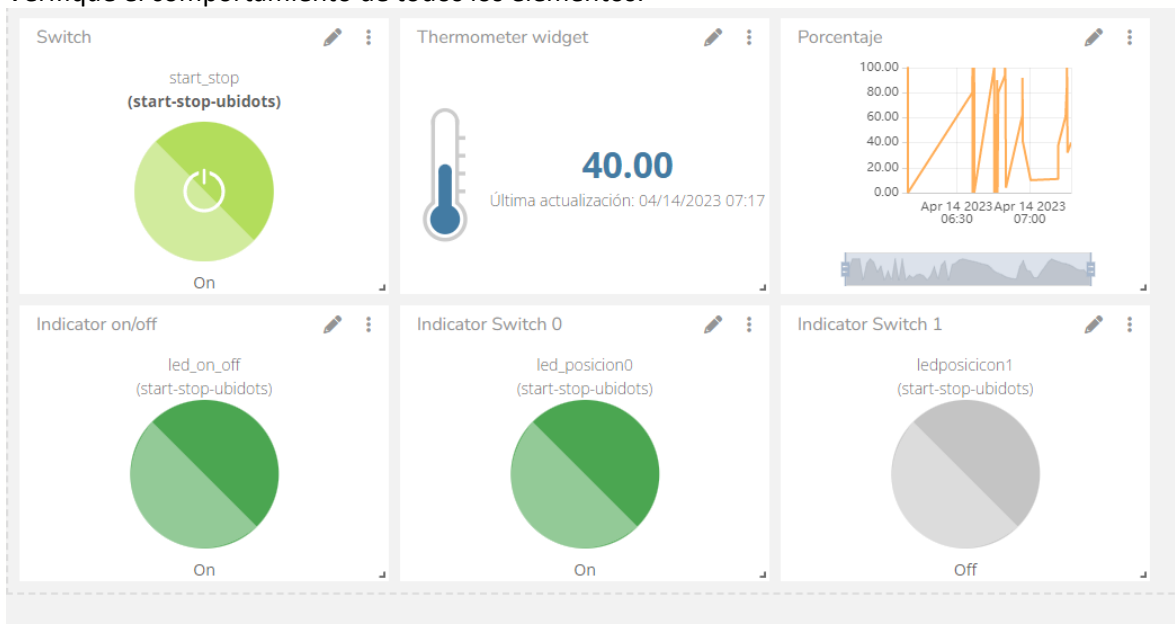
En la pestaña “APARIENCIA” cambie el nombre por “indicador on/off”:



Realice el mismo procedimiento para agregar 2 indicadores mas para las variables “led_posicion0” y “led_posicion1” y asigne los nombres “Indicador Switch 0” e “Indicador Switch 1” respectivamente:



Verifique el comportamiento de todos los elementos:



Instale la aplicación móvil desde Play Store o Apple Store, ingrese con la cuenta creada y verifique el funcionamiento del tablero generado desde cualquier parte del mundo:

