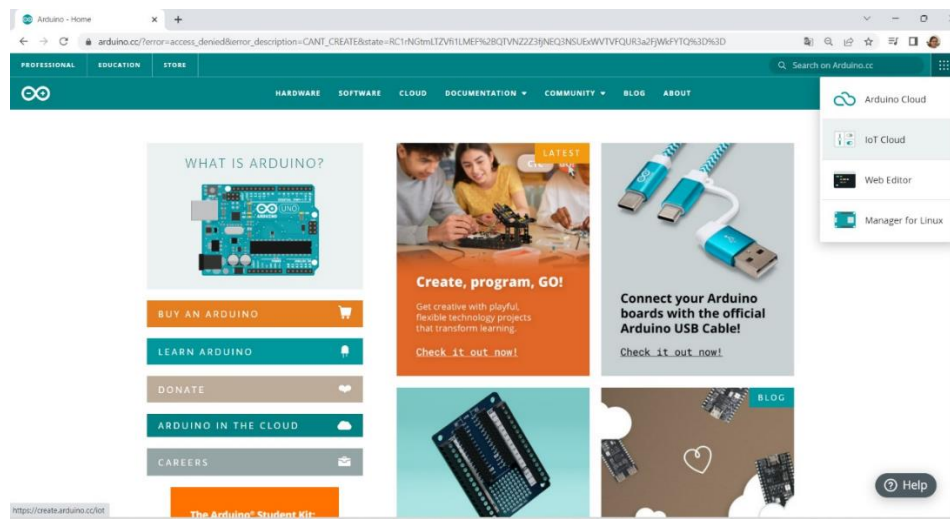


EJERCICIO 1-D:

IMPLEMENTACION DE UN SENSOR INTELIGENTE DE TEMPERATURA:

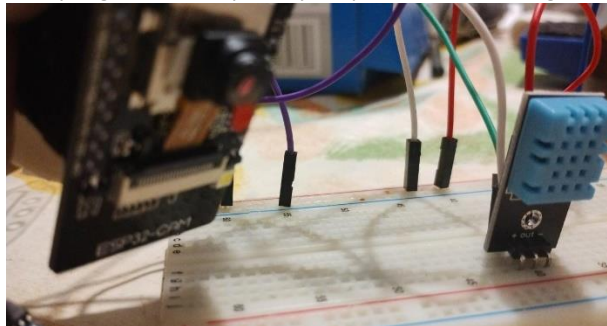
Un sensor inteligente de temperatura, se basa en el uso de un sensor de temperatura y el valor de lectura lo procesa un dispositivo como un microprocesador el cual almacena en su memoria el valor leído, a partir de un programa proceso la información y puede enviarlo a la nube o una aplicación de manera tal de acceder a ese valor y tomar decisiones, utilizo tablero para visualizar sus valores de la forma más apropiada o representativa. Como propuesta de como usaría un dispositivo se desarrolla la siguiente propuesta:

Para el desarrollo de este punto, se procedió a registrarse en la plataforma de arduino.cc, para poder crear y acceder a la cuenta de Arduino IoT Cloud.



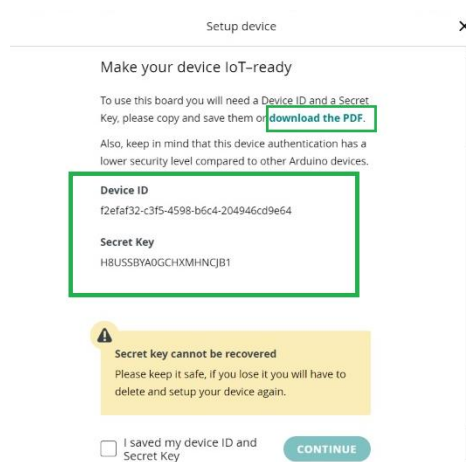
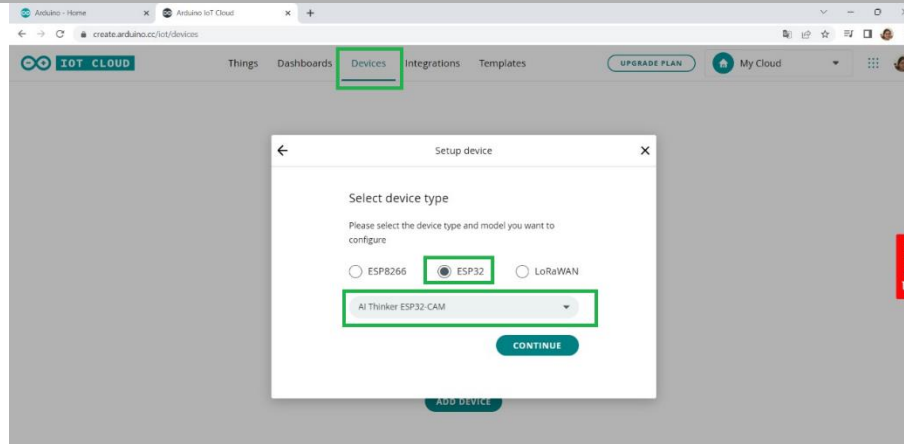
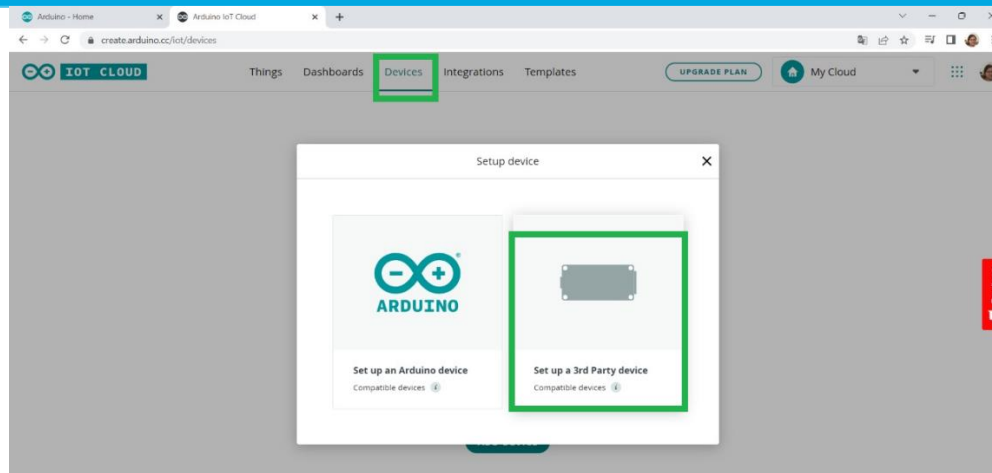
En este ejemplo se usarán los siguientes componentes, se aclara que su selección es debido a que eh decidido verificar su funcionamiento usando los componentes con lo que cuento.

- ESP32 CAM.-
- DHT11 , se usara el sensor de temperatura . Se aclara que este sensor no es solo para temperatura sino también para humedad, pero en esta explicación solo se tomara su funcionalidad como sensor de temperatura ya que es el que cuento para luego bajar el programa a mi placa y responder a la consigna solicitada.

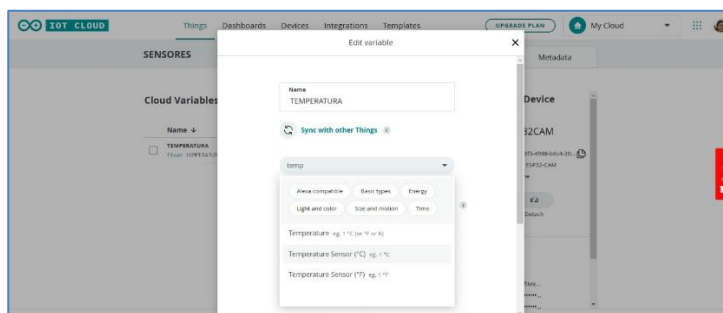
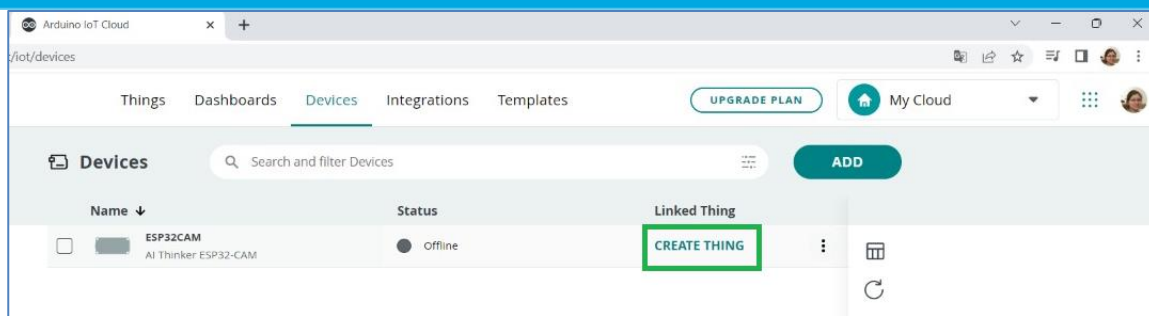


Al ingresar a la cuenta de Arduino IoT Cloud, se realiza siguen los siguientes pasos:

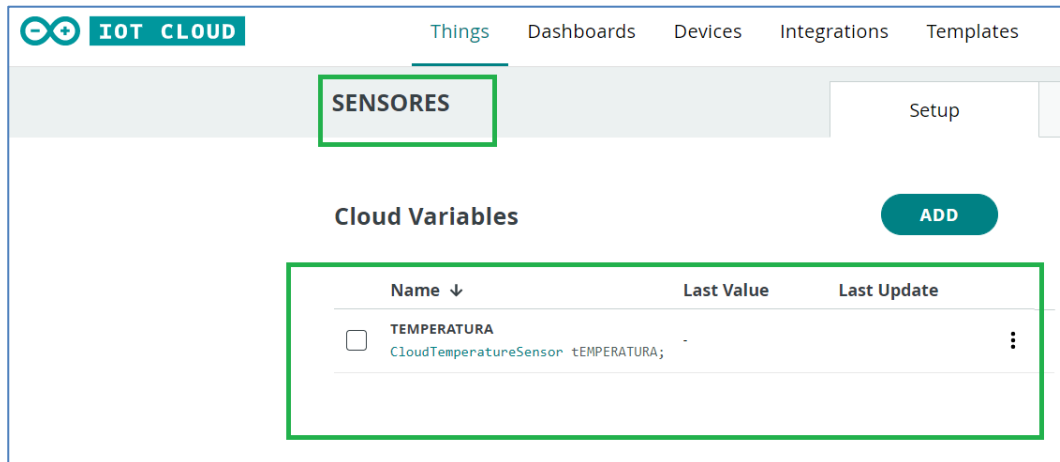
- 1- En Devices se selecciona el modelo de placa ESP32 CAM , al seleccionar se registra el mismo y se genera una clave secreta y el ID del dispositivo, en el archivo pdf se almacena esta información.



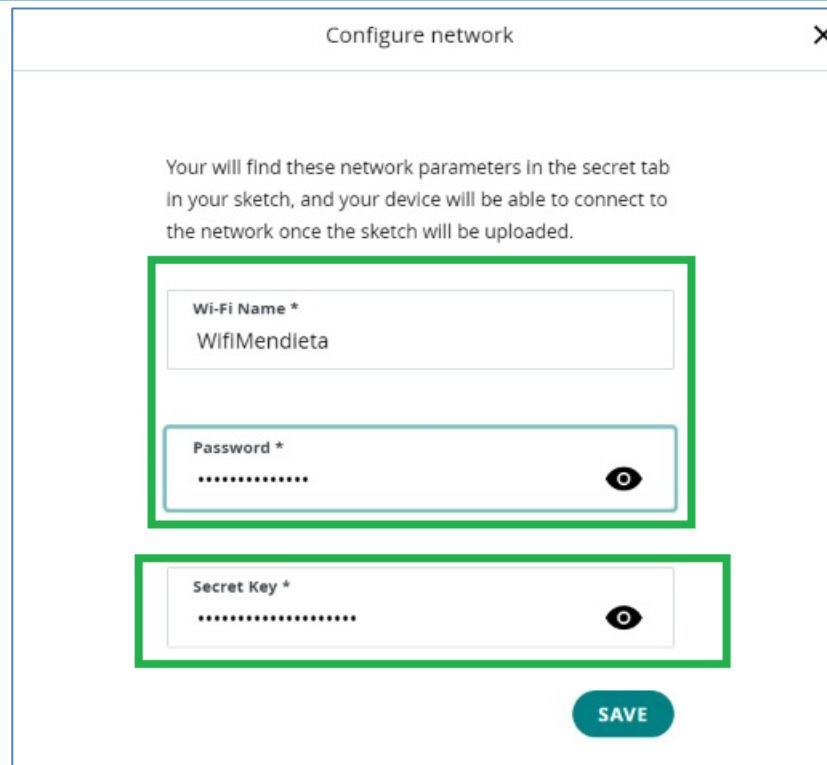
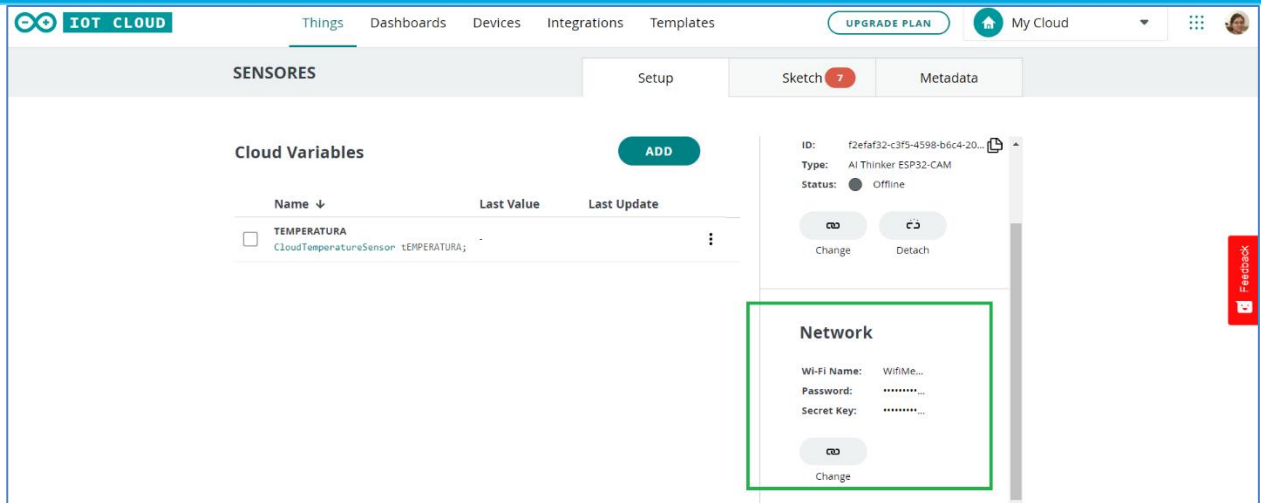
- 2- Asociada a la placa se crea el sensor de temperatura, para eso se selecciona la opción de crear una cosa. En este caso se llamo **sensores** y se le asigno una variable con el nombre de **Temperatura**. Se edita el tipo de variable para asociarla con el cambio en el valor de la temperatura y con la unidad de medida en °C., ya que en un principio seleccione como una variable del tipo float .



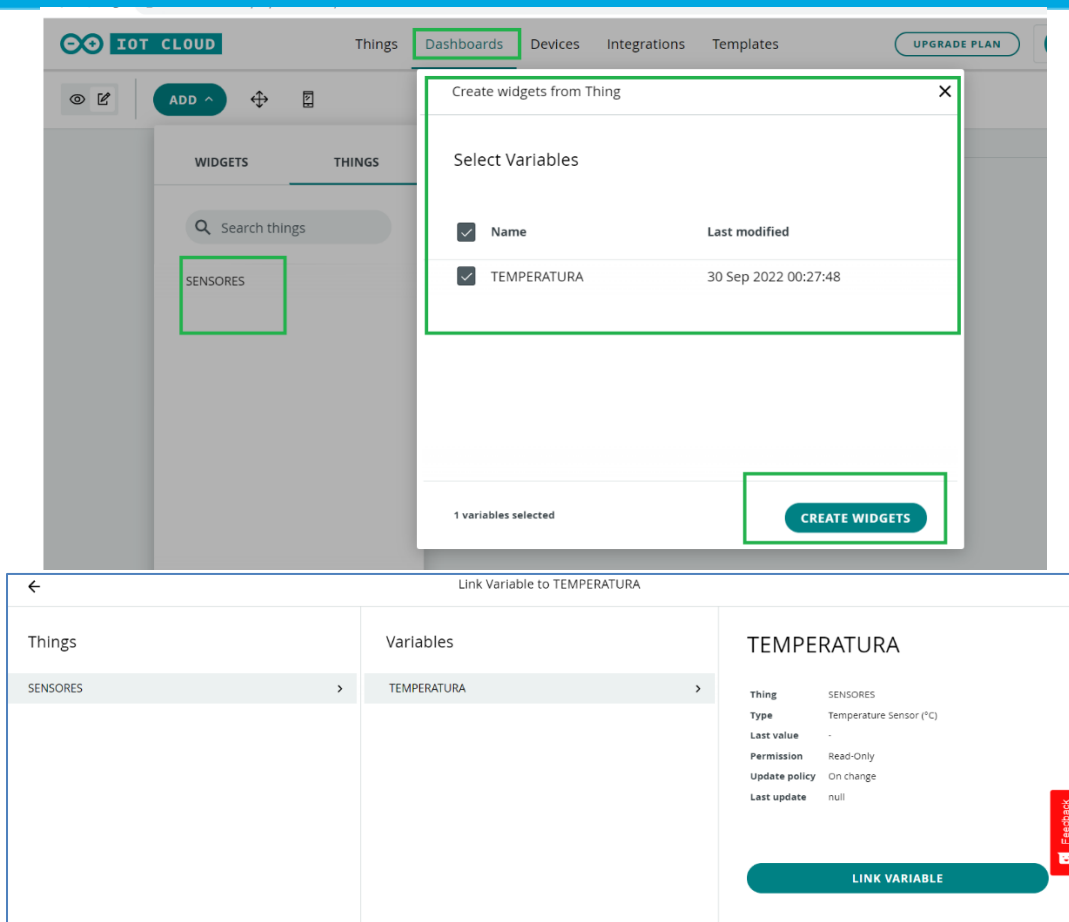
Se edita el tipo de variable TEMPERATURA, para que sea del tipo sensor de temperatura con unidad de medida de °C, del tipo solo lectura en un rango de 0-50 °C.



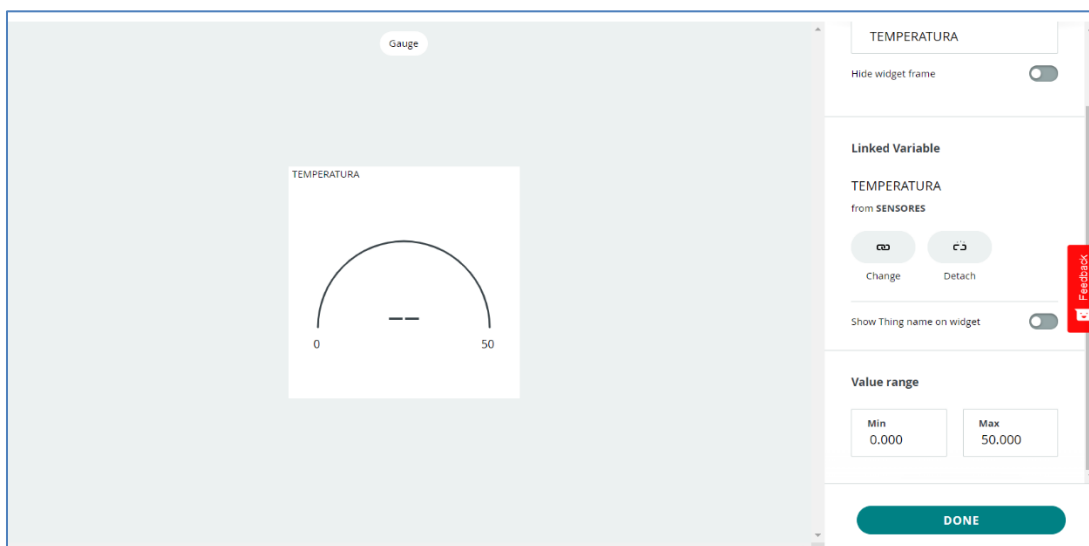
- 3- Configurar la red a la que se va a conectar la placa ESP32 CAM, con los datos de la red y la clave secreta del dispositivo guardada en el archivo PDF anteriormente descargado.

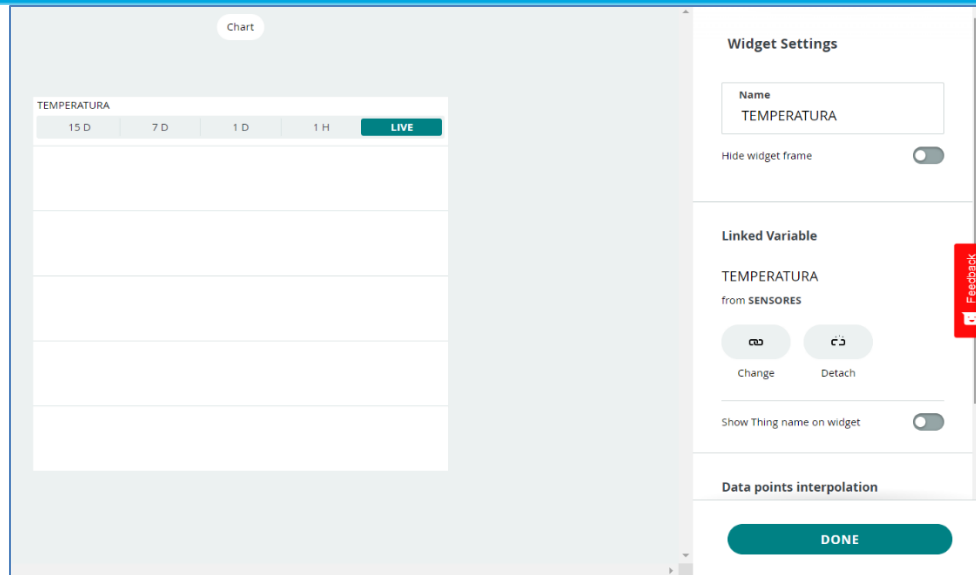


- 4- Para visualizar los datos de temperatura desde ESP32 CAM y tener un registro de estos se va a utilizar los widgets de gráficos en el Dashboard de Arduino. Se crea un dashborad y se la nombra SENSORES INTELIGENTES, para ello se selecciona la cosa creada anteriormente, en este caso SENSORES y nos muestra la variable TEMPERATURA asociada a la cosa. Posteriormente se agrega nuevos widgets que se asocia a la variable TEMPERATURA para tener una gráfica de las mediciones:



Se agrega para visualizar de otra forma la lectura de la temperatura otros tipos de widgets:

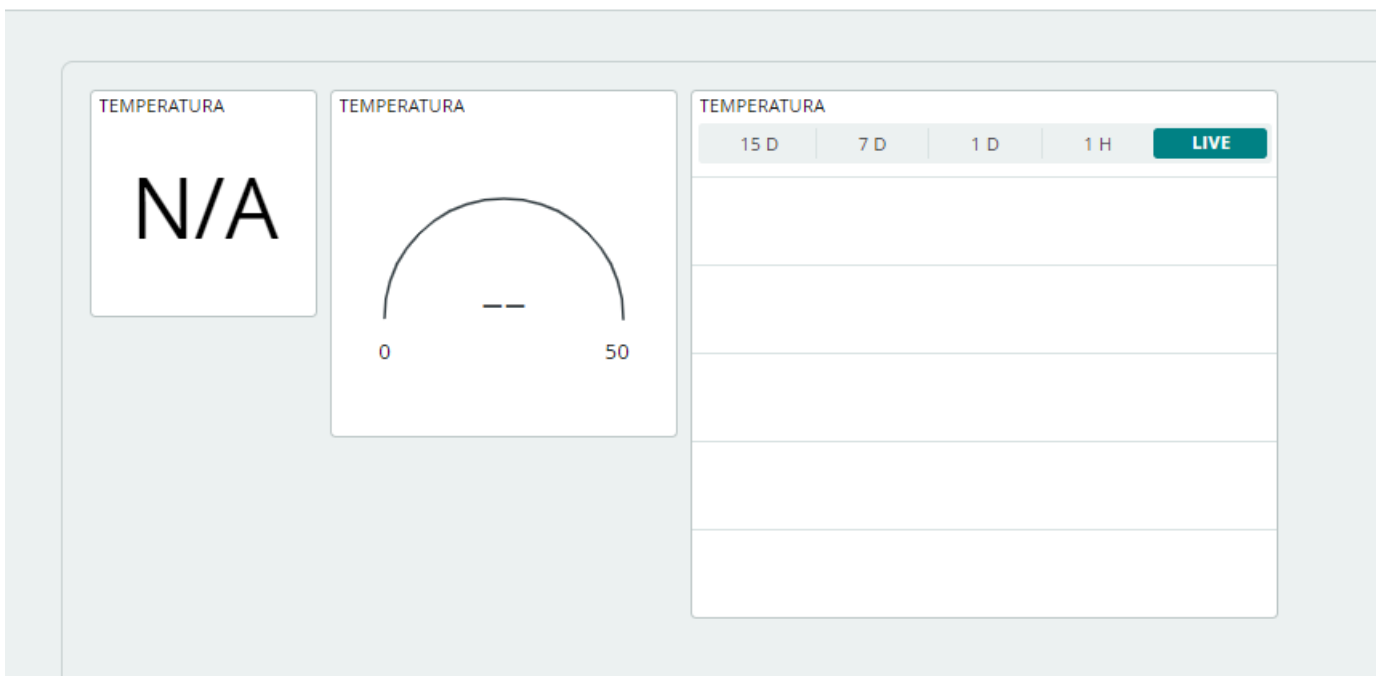




Tablero:

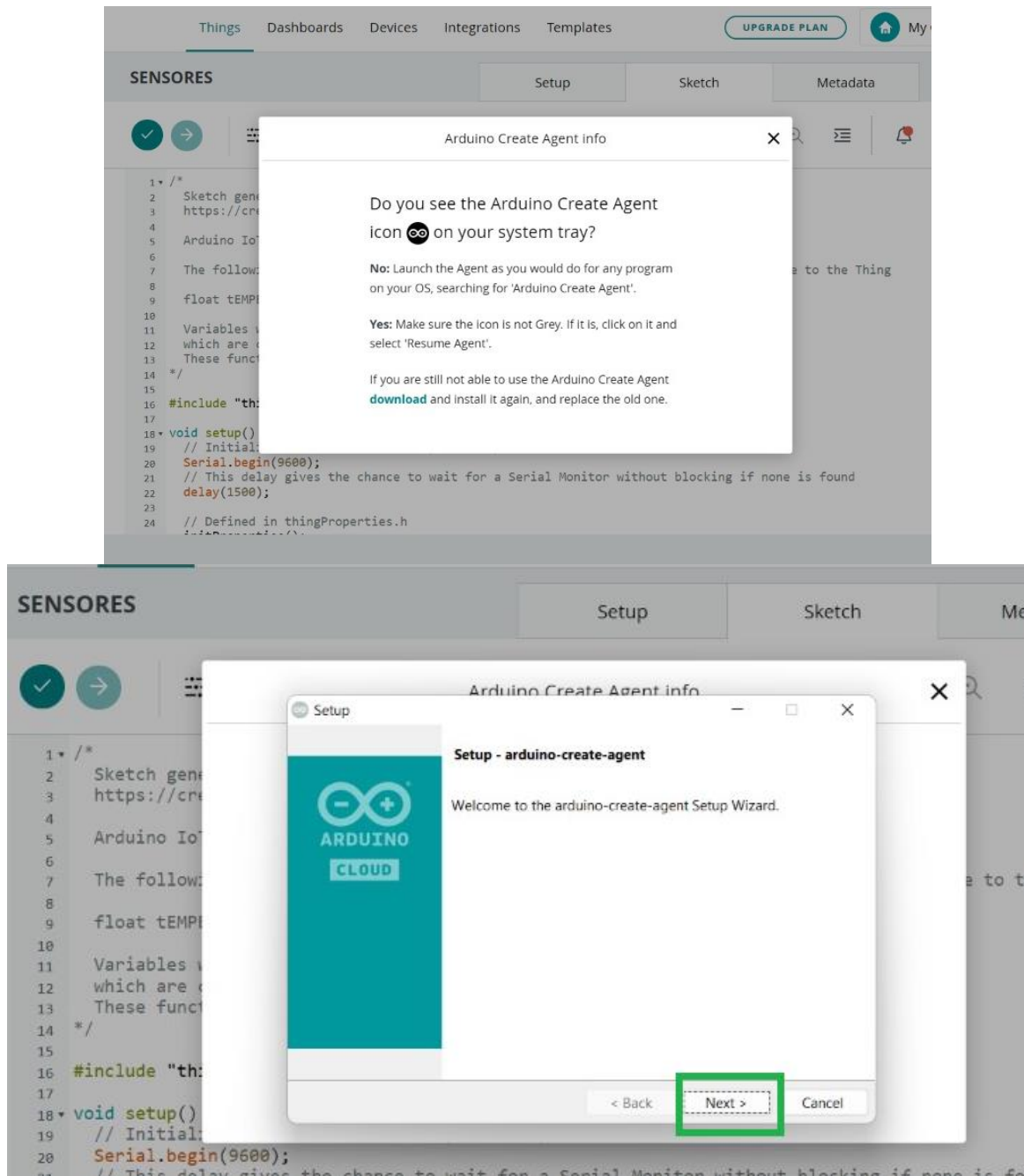
Things Dashboards Devices Integrations Templates

SENSOR INTELIGENTE



- Al ingresar a la solapa Sketch, dentro de Things, la primera vez aparece un mensaje el cual nos avisa que se debe descargar el instalador del agente de creación de Arduino para cargar el código a la placa y ejecutar dando permiso al

firewall de Windows .



Se procede a instalar.

- 6- Se edita el programa para dar lectura al sensor de temperatura y usar el monitor serie para visualizar los mismos, como otra opción de visualizar el valor medido de temperatura.

SENSORES

Setup

Sketch

Metadata



No associated device found

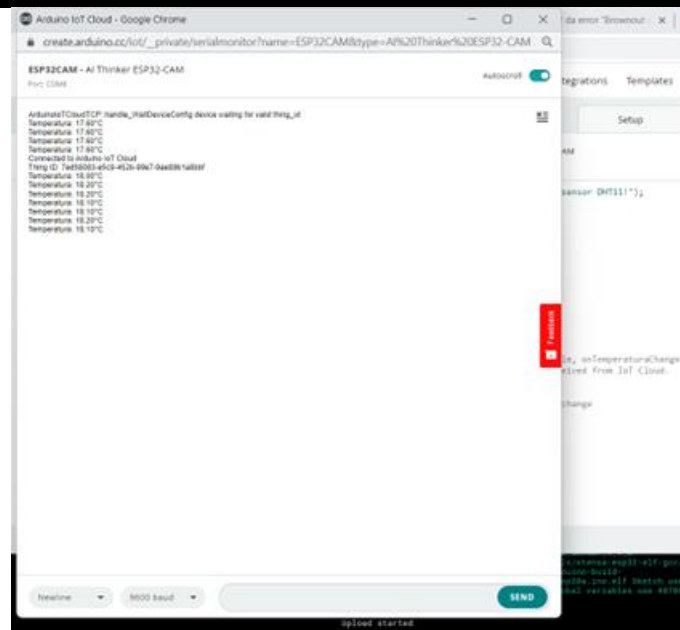
Open full editor



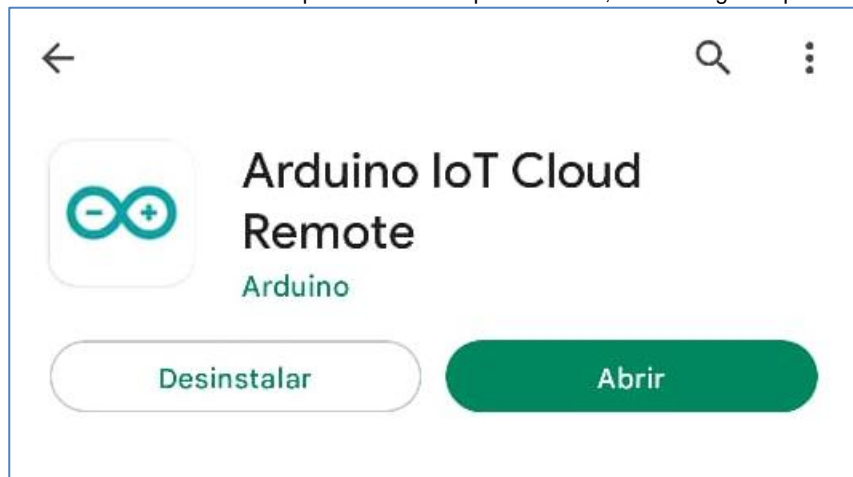
```

7 The following variables are automatically generated and updated when changes are made to the Thing
8
9 CloudTemperatureSensor TEMPERATURA;
10
11 Variables which are marked as READ/WRITE in the Cloud Thing will also have functions
12 which are called when their values are changed from the Dashboard.
13 These functions are generated with the Thing and added at the end of this sketch.
14
15 */
16
17 #include "thingProperties.h"
18
19 #include "DHT.h"
20 #define DHTpin 2 //
21 #define DHTTYPE DHT11
22 DHT dht(DHTpin,DHTTYPE);
23
24
25 void setup() {
26 // Initialize serial and wait for port to open:
27 Serial.begin(9600);
28 // This delay gives the chance to wait for a Serial Monitor without blocking if none is found
29 delay(1500);
30 dht.begin();
31
32 // Defined in thingProperties.h
33 initProperties();
34
35 // Connect to Arduino IoT Cloud
36 ArduinoCloud.begin(ArduinoIoTPreferredConnection);
37
38
39 /*
40 The following function allows you to obtain more information
41 related to the state of network and IoT Cloud connection and errors
42 the higher number the more granular information you'll get.
43 The default is 0 (only errors)
44 */

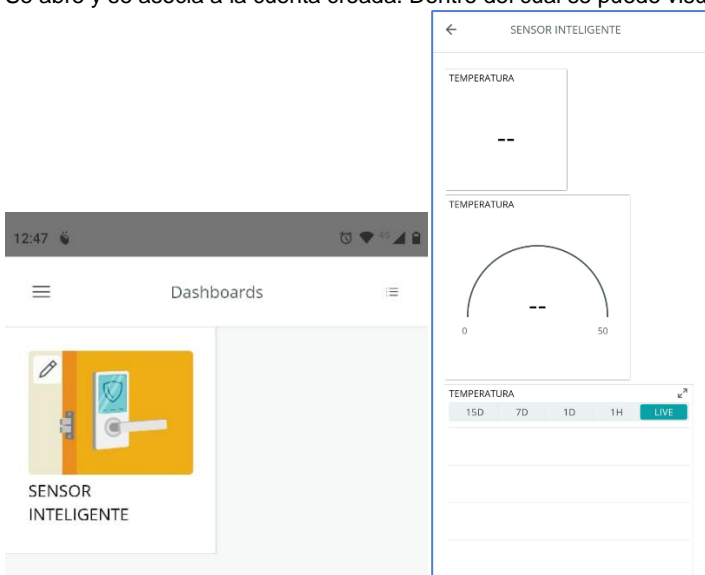
```



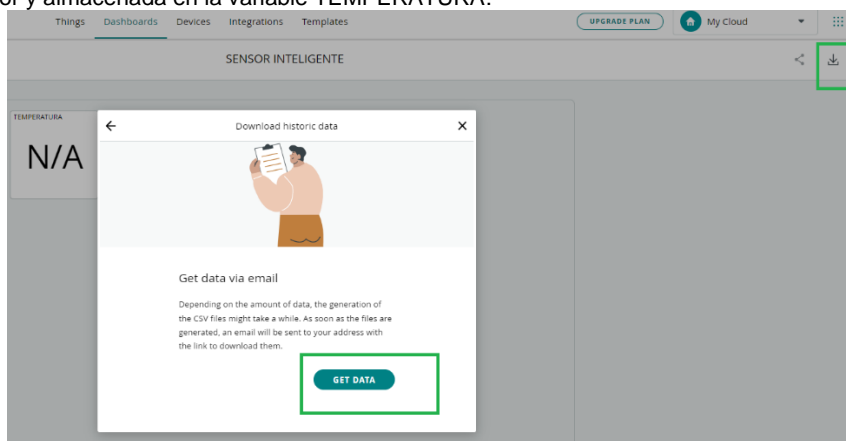
- 7- Como otra alternativa de visualizar los datos de la temperatura medida por el sensor, se descarga la aplicación en el celular:



- 8- Se abre y se asocia a la cuenta creada. Dentro del cual se puede visualizar:

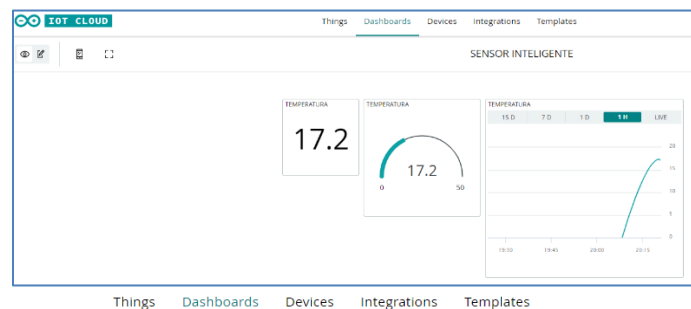
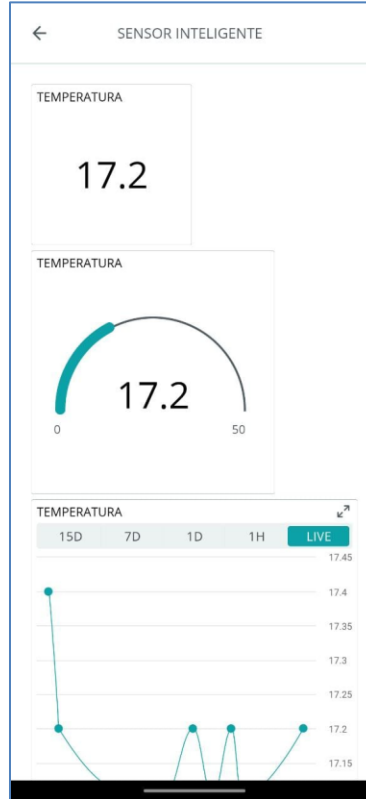


- 9- La plataforma de Arduino brinda la opción de envía un email un archivo .zip con una hoja de cálculo con los datos de las mediciones realizadas por el sensor y almacenada en la variable TEMPERATURA:

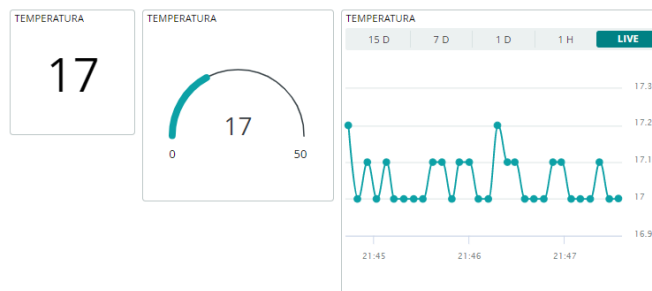


RESULTADOS OBTENIDOS:

Se baja el programa a la placa y se verifica su funcionalidad. Se observa como se el tablero desde la aplicación del celular y desde la computadora.



SENSOR INTELIGENTE



Autoguardado ☐ **SENSORES-temperatura**

Archivo Inicio Insertar Disposición de página Fórmulas Datos Revisar

Calibri 11 A⁺ A⁻ N K S Fuente

A196 2022-10-01T00:17:48.518Z,17.200000762939453

	A
1	time, value
2	2022-09-30T23:08:15.396Z,0
3	2022-09-30T23:20:27.762Z, 17.399999618530273
4	2022-09-30T23:20:33.597Z,17.200000762939453
5	2022-09-30T23:59:07.338Z,17
6	2022-09-30T23:59:13.174Z,17.299999237060547
7	2022-09-30T23:59:19.011Z,17.299999237060547
8	2022-09-30T23:59:24.858Z,17.200000762939453
9	2022-09-30T23:59:30.686Z,17.200000762939453
0	2022-09-30T23:59:36.522Z,17.299999237060547
1	2022-09-30T23:59:42.355Z,17.299999237060547
2	2022-09-30T23:59:48.202Z,17.200000762939453
3	2022-09-30T23:59:54.03Z,17.299999237060547
4	2022-09-30T23:59:59.866Z,17.200000762939453
5	2022-10-01T00:00:05.702Z,17.200000762939453
6	2022-10-01T00:00:11.542Z,17.200000762939453
7	2022-10-01T00:00:17.379Z,17.200000762939453
8	2022-10-01T00:00:23.218Z,17.200000762939453
9	2022-10-01T00:00:29.051Z,17.200000762939453
0	2022-10-01T00:00:34.891Z,17.200000762939453
1	2022-10-01T00:00:40.726Z,17.200000762939453
2	2022-10-01T00:00:46.562Z,17.200000762939453
3	2022-10-01T00:00:52.402Z,17.200000762939453
4	2022-10-01T00:00:58.239Z,17.200000762939453
5	2022-10-01T00:01:04.071Z,17.200000762939453
6	2022-10-01T00:01:09.914Z,17.299999237060547
7	2022-10-01T00:01:15.746Z,17.200000762939453
8	2022-10-01T00:01:21.583Z,17.200000762939453

SENSORES-temperatura

listo Accesibilidad: No disponible

Valores obtenidos de las lecturas realizadas por el sensor.

Se adjunta el archivo que se envía por correo electrónico, el cual se puede abrir con un Excel y procesar dicha información. El programa también se puede bajar desde la plataforma de Arduino.cc