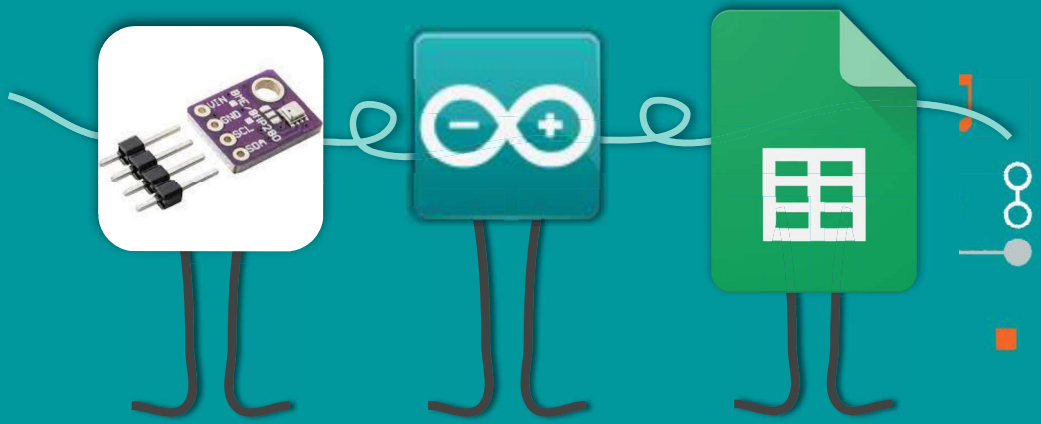
ARDUINO DAYS 2024

Sensor talking to Google Sheets

A love story for a talkative sensor



How to log data to a server?

Set up a server myself

ThingSpeak

How to log data to a server?

Set up a s ver myself

- Too much of an effort (future task)

ThingSpeak

How to log data to a server?

Set up a server myself

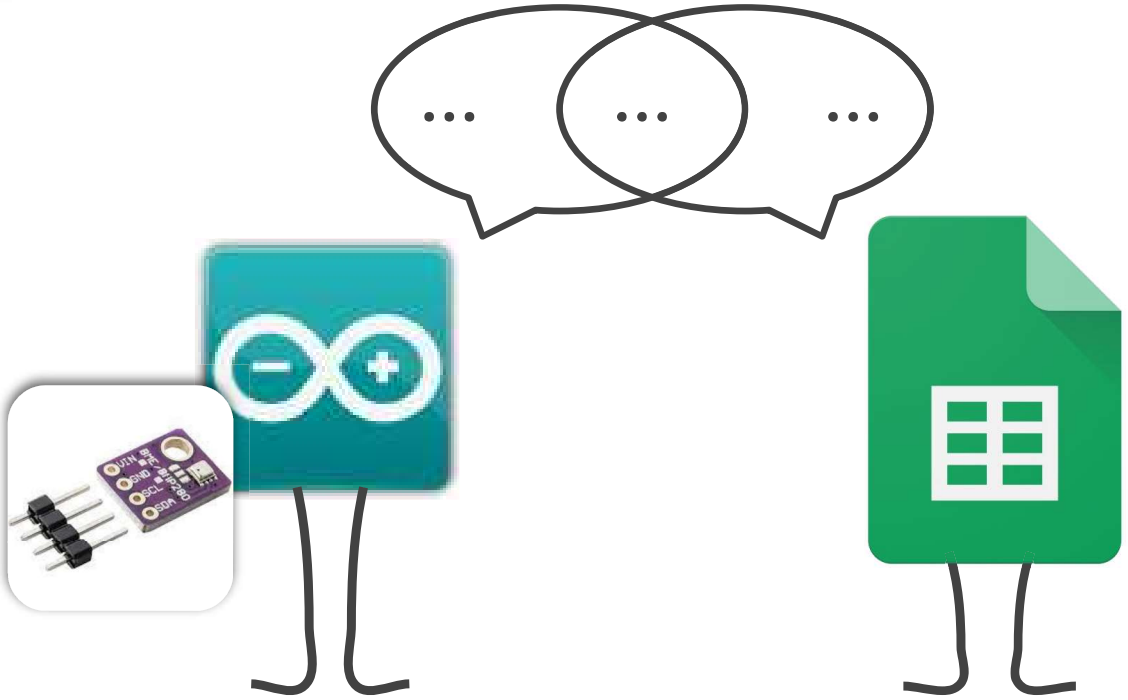
- Too much of an effort (future task)

ThingSpeak

Free web service for displaying the data online:

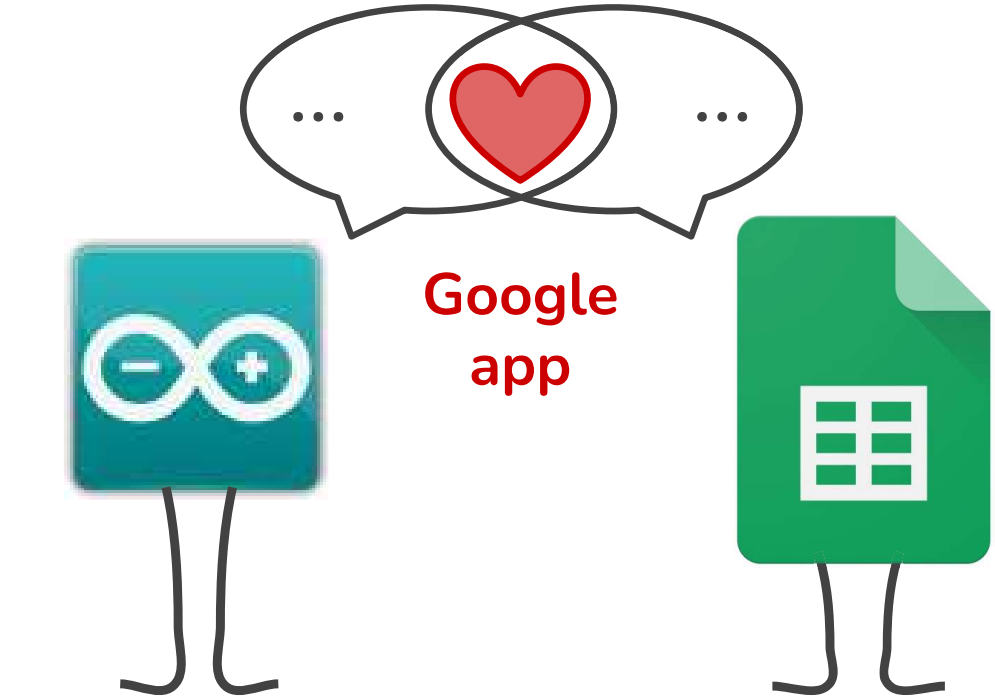
- Free link for limited time and limited number of sensors
- access only with a MathWorks account (Matlab)

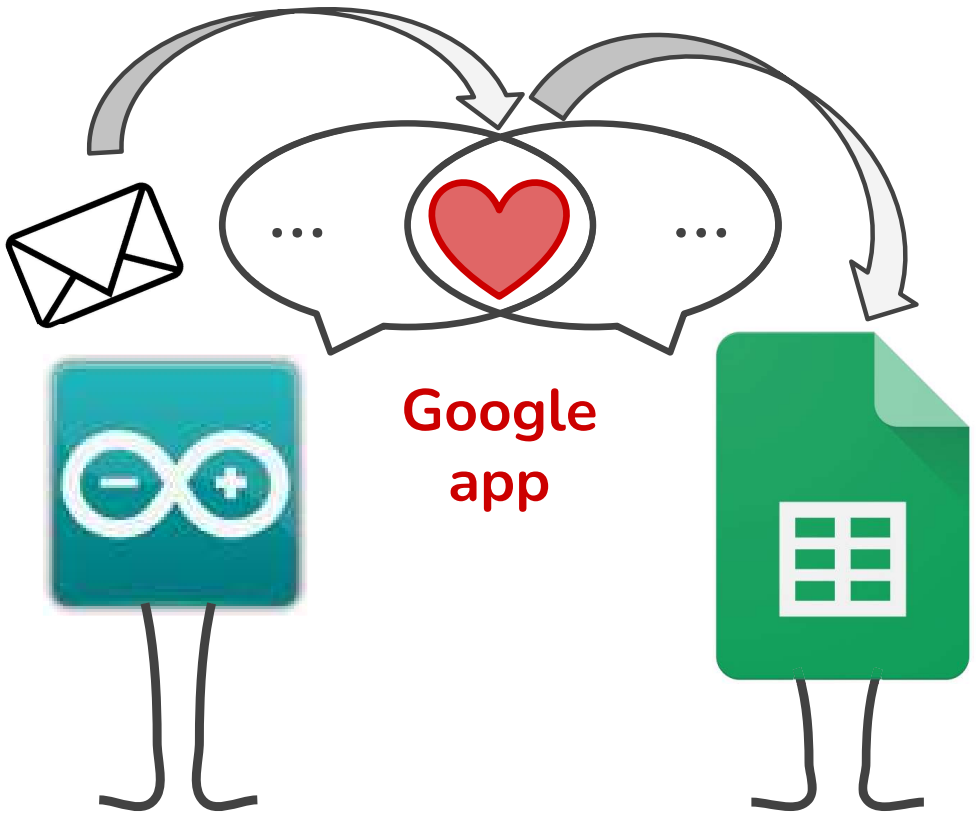
ARDUINO DAYS 2024



Sensor and microcontroller

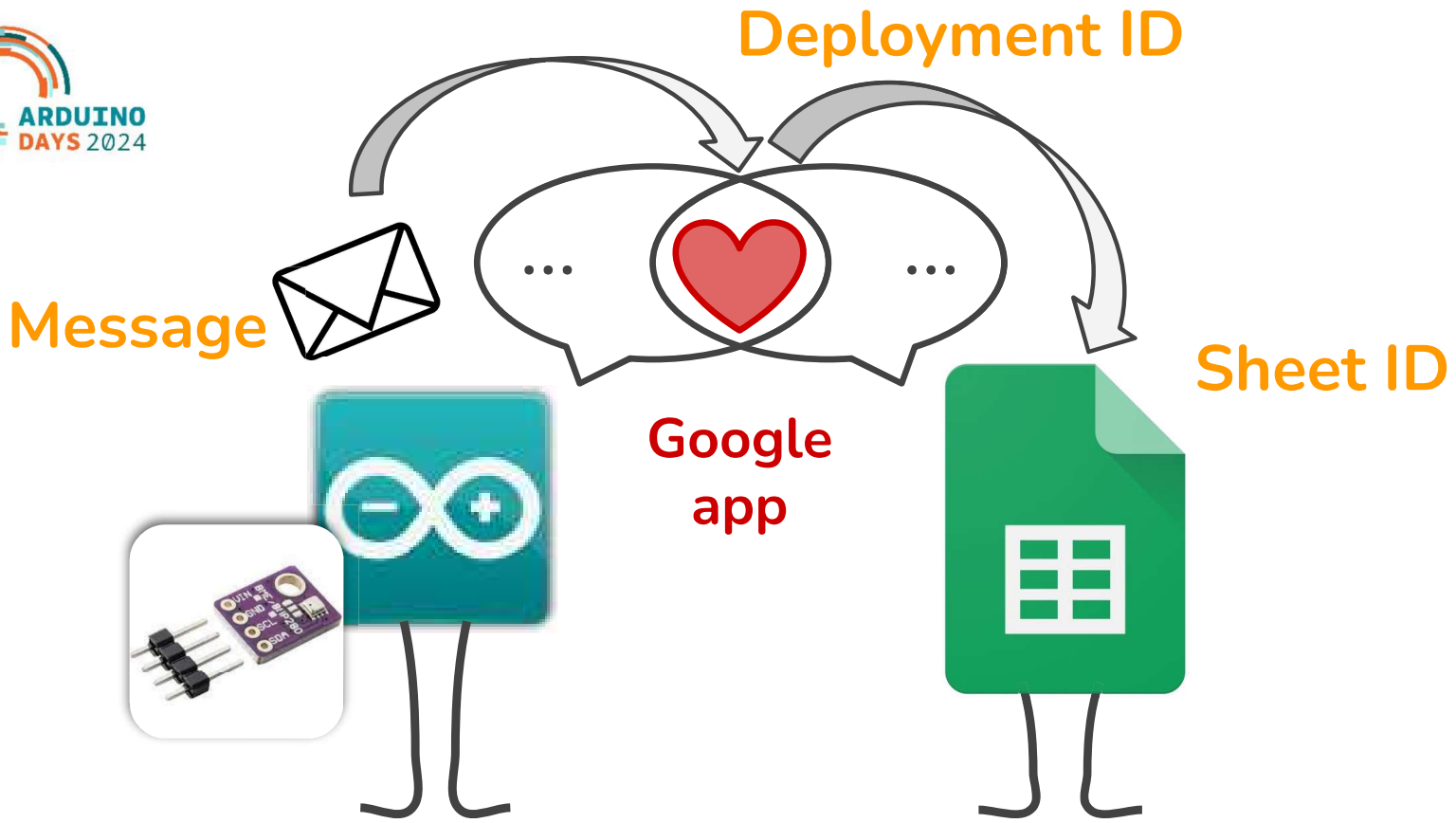
Online spreadsheet editor


Google
app
Sensor and
microcontroller
Online
spreadsheet
editor



Sensor and microcontroller

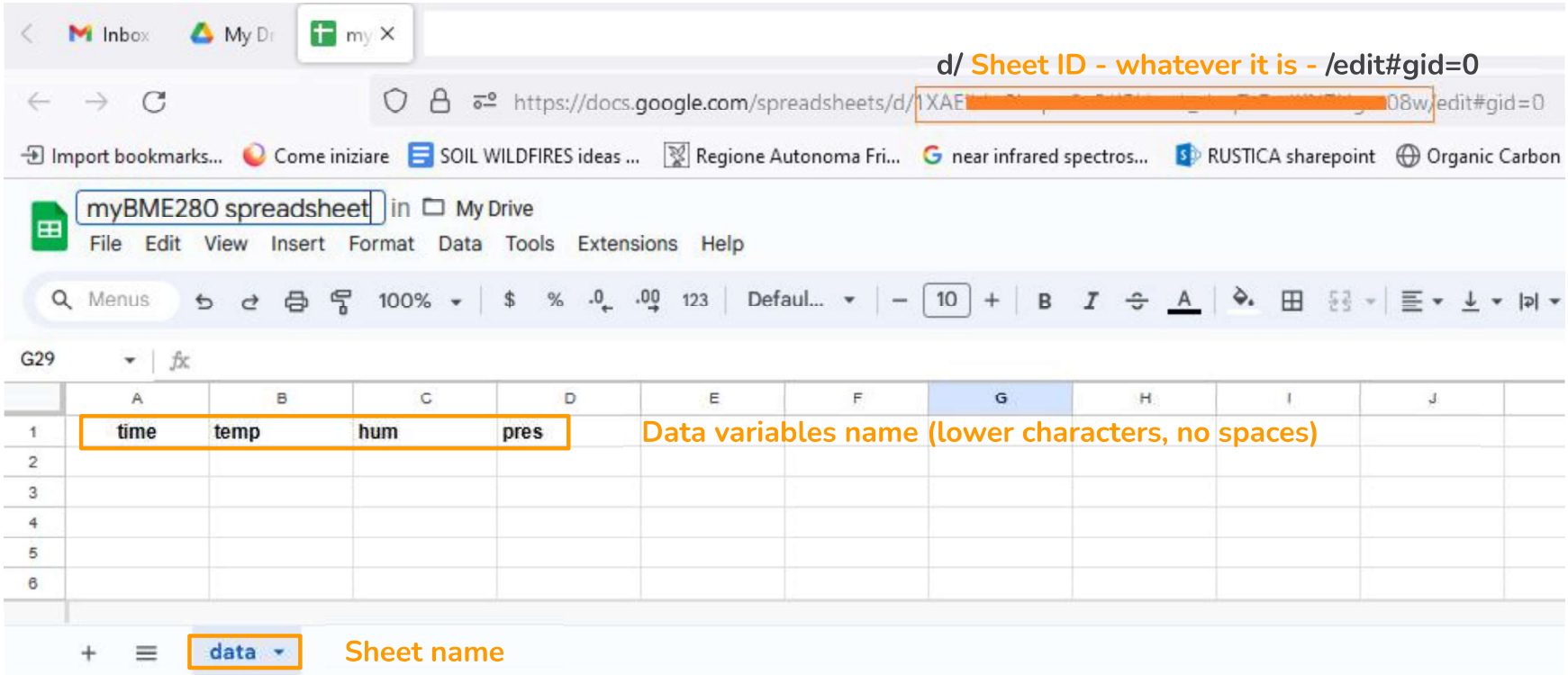
Online spreadsheet editor

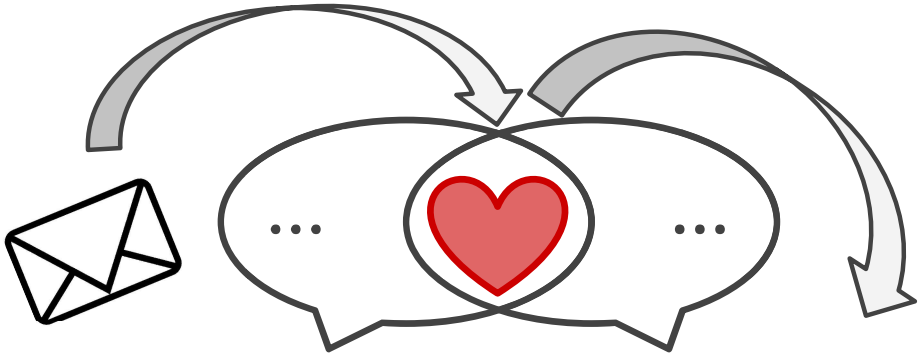

Deployment ID
Message
Google app
Sheet ID
Sensor and microcontroller
Online spreadsheet editor

ARDUINO DAYS 2024

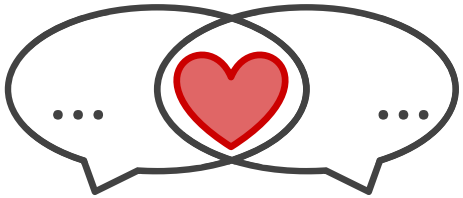
Online spreadsheet editor

Set up the online spreadsheet editor





Google app



Create a Google app script

Extensions Help

Add-ons

Macros

Apps Script

Apps Script Untitled project

Deploy

Files

myBME280_Appscript.gs

Libraries

Services

Run Debug doGet Execution log

Sheet ID
Sheet name

Variables collected

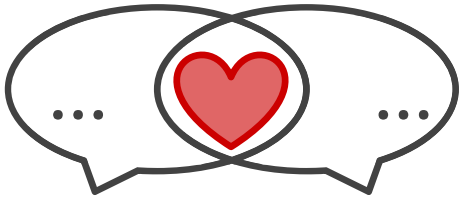
```
/*!
The sheet address is (from browser bar):
//https://docs.google.com/spreadsheets/d/1Rd                                    cgM/edit#gid=1427696656
The sheet_id is the number between "d/" and "/edit..."
*/

// Variables definition
var sheet_id = "1Rd                                    cgM";
var sheet_name = "data";  // watever name you gave the data sheet where you want to collect the data
var ss = SpreadsheetApp.openById(sheet_id);
var sheet = ss.getSheetByName(sheet_name);

// Function to get the data
function doGet(e){
  //get data from ESP32
  if (e.parameter == 'undefined') {
    return ContentService.createTextOutput("Received data is undefined");
  }
  // variables collected. In order of collection (same as in the header of the Google sheet) ----------
  var time = e.parameter.time;
  var temp = e.parameter.temp;
  var hum = e.parameter.hum;
  var pres = e.parameter.pres;

  // append new row of data ------------------------------------------------------
  sheet.appendRow([time,temp,hum,pres]);

  //returns response back to ESP32
  return ContentService.createTextOutput("Status Updated in Google Sheet");
  //-------------------------------------------------------------------------------
}
```



Deploy & get the deployment ID

Deploy

New deployment

New deployment

Select type

Configuration

Web app

Description

New description

data logger (to googlesheets)

Web app

Execute as

Me (mtsoilsensors@gmail.com)

The web app will be authorized to run using your account data.

Who has access

Anyone

This can also be used as a library. Learn more

Cancel

Deploy

New deployment

Deployment successfully updated.

Version 1 on Mar 20, 2024, 12:35 PM

Deployment ID

AKfycbwR-A7zaZCmWVVsXdEvhIQlWdSzP6q9IsGRZaZSRGyPzwLFBrs-O5saGummpheTMJsr

Copy

Web app

URL

https://script.google.com/macros/s/

Copy

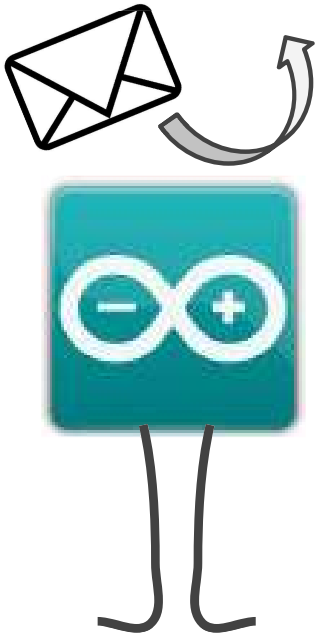
+

/exec?time=300&temp=20&hum=45&pres=101074

Done

ARDUINO DAYS 2024

Send data to a server via HTTP



Sensor and microcontroller

Parts that go BEFORE the void setup(){}

```
#include <WiFi.h>
#include <HTTPClient.h>       // to send data to Google Sheets
#include <Wire.h>             // For the BME280 sensor
#include <BME280I2C.h>        // For the BME280 sensor by Tyler Glen

#define SERIAL_BAUD 115200

// BME280
  const uint8_t sda = 21; // SDA (Serial Data) ->  21 on TTGO-T1, A4 on Uno/Pro-Mini, 20 on Mega2560/Due, 2 Leonardo/Pro-Micro
  const uint8_t scl = 22; // SCK (Serial Clock)  ->  22 on TTGO-T1, A5 on Uno/Pro-Mini, 21 on Mega2560/Due, 3 Leonardo/Pro-Micro
  BME280I2C bme;    // Default : forced mode, standby time = 1000 ms
  float temp(NAN), hum(NAN), pres(NAN);

// Network credentials (STATION)
  const char* ssid = "[redacted]";
  const char* password = "[redacted]";

// Google Deployment script ID and more
  String GOOGLE_SCRIPT_ID = "[redacted]jkH";    // change Deployment Gscript ID with yours
  String urlFinal;  // char for the url with data to log
```

Parts that can go BEFORE the void setup(){}

```
// FUNCTIONS ---------------------------------------------------------------
  // f9) send data to google sheet
    void sendtogoogle(String urlFinal){
        int httpCode;
        String payload;  // response data received from the server.

        HTTPClient http;
        http.begin(urlFinal.c_str()); //  Initializes the HTTP client to make a request to the URL and converts the String to a C-style
        http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
        httpCode = http.GET();  // stores the response code from a HTTP GET request to the URL
        Serial.printf("HTTP Status Code: %d\n",httpCode);
        payload = http.getString();
        http.end();
        //http.clear();
        if (httpCode > 0) {                                    // getting response from google sheet
          Serial.println("Payload--> "+payload);
        } else {
          Serial.println("Not 0 - Payload--> "+payload);
        }
    }

// -------------------------------------------------------------------------
```

void setup(){} and void loop(){}

```
62  // SETUP ------------------------------------------------------------
63  void setup(){
64    delay(1000);
65    Serial.begin(SERIAL_BAUD);
66
67    // Connect to WIFI and print the accessed WIFI
68    WiFi.begin(ssid, password);
69      Serial.println("\nConnected to the WiFi network");
70      Serial.println("IP Address: " + WiFi.localIP().toString());
71      Serial.print("RSSI: ");
72      Serial.println(WiFi.RSSI());
73
74    // BME280 check connection
75    while(!Serial) {} // Wait
76    Wire.begin(sda, scl);
77    while(!bme.begin())
78    {
79      Serial.println("Could not find BME280 sensor!");
80      delay(1000);
81    }
82  }
83
84  void loop() {
85
86    // 1) Read BME280 sensor
87    BME280::TempUnit tempUnit(BME280::TempUnit_Celsius);
88    BME280::PresUnit presUnit(BME280::PresUnit_Pa);
89    bme.read(pres, temp, hum, tempUnit, presUnit);
90    Serial.println("Humidity (%) = " + String(hum) + " , Temperature (°C) = " + String(temp) + " , Pressure (Pa) = " + String(pres));
91
92    // 2) log data to Google sheet
93    unsigned long seconds = millis() / 1000;
94    urlFinal = "https://script.google.com/macros/s/"+GOOGLE_SCRIPT_ID+"/exec?time="+seconds+"&temp="+temp+"&hum="+hum+"&pres="+pres;
95    Serial.println("POST data to spreadsheet: "+urlFinal);
96    sendtogoogle(urlFinal);
97
98    delay(5000);
99  }
```

void setup(){} and void loop(){}

```
62  // SETUP ------------------------------------------------------------
63  void setup(){
64    delay(1000);
65    Serial.begin(SERIAL_BAUD);
66
67    // Connect to WIFI and print the accessed WIFI
68    WiFi.begin(ssid, password);
69      Serial.println("\nConnected to the WiFi network");
70      Serial.println("IP Address: " + WiFi.localIP().toString());
71      Serial.print("RSSI: ");
72      Serial.println(WiFi.RSSI());
73
74    // BME280 check connection
75    while(!Serial) {} // Wait
76    Wire.begin(sda, scl);
77    while(!bme.begin())
```

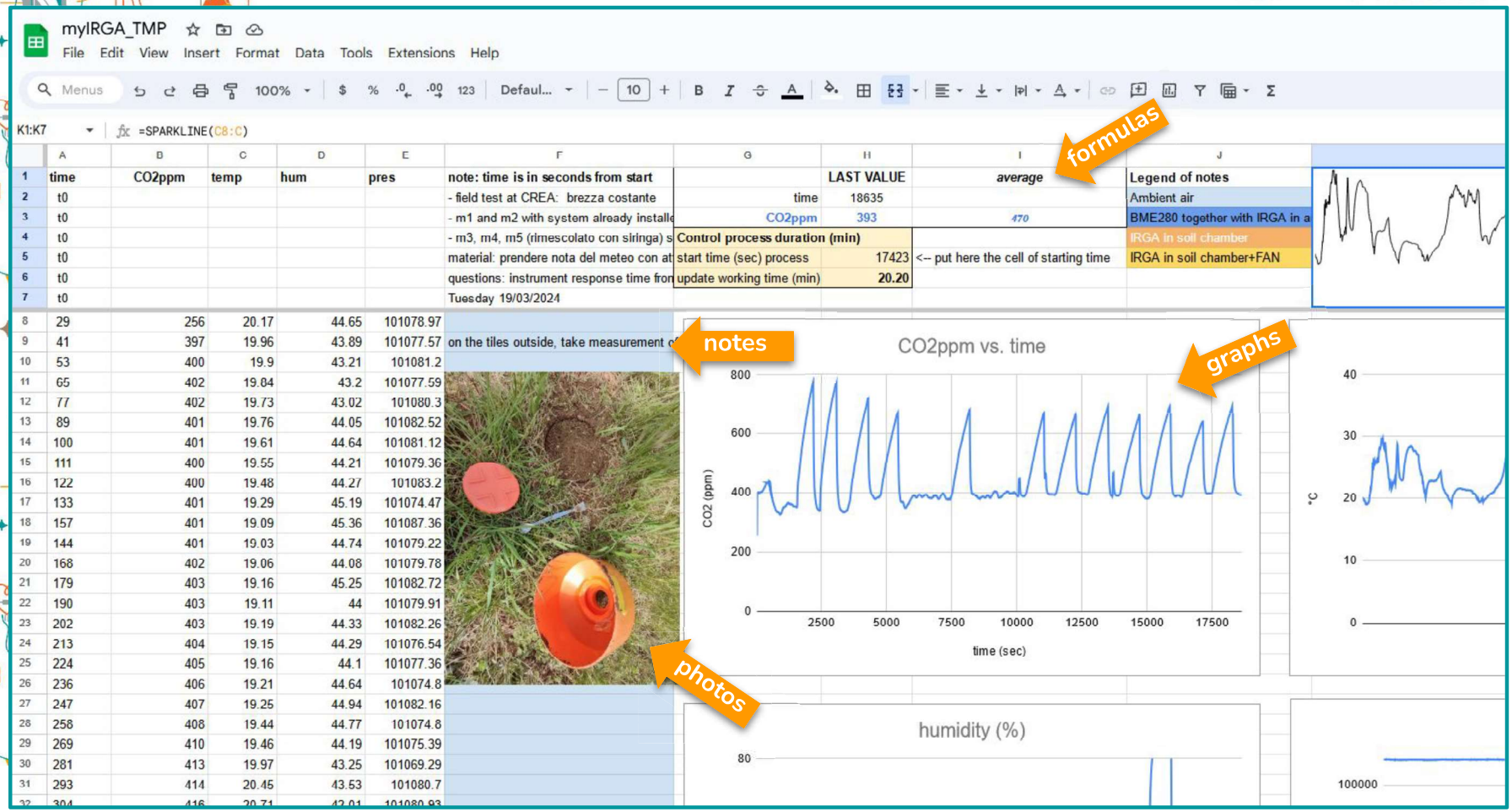
Output Serial Monitor ×

Message (Enter to send message to 'TTGO T1' on 'COM3')

```
16:42:14.818 -> Humidity (%) = 39.64 , Temperature (°C) = 19.64 , Pressure (Pa) = 98245.22
16:42:14.818 -> POST data to spreadsheet: https://script.google.com/macros/s/AKfycbzNVxlsJiYqz65VEkMevTMQoXNnJ-R7m02dgmzKtd9YWW9HeGyVYaSXnbZK3b89OjkH/exec?time=173&temp=19.64&hum=39.64&pres=98245.22
16:42:19.220 -> HTTP Status Code: 200
16:42:19.220 -> Payload--> Status Updated in Google Sheet
```

```
88    BME280::PresUnit presUnit(BME280::PresUnit_Pa);
89    bme.read(pres, temp, hum, tempUnit, presUnit);
90    Serial.println("Humidity (%) = " + String(hum) + " , Temperature (°C) = " + String(temp) + " , Pressure (Pa) = " + String(pres));
91
92    // 2) log data to Google sheet
93    unsigned long seconds = millis() / 1000;
94    urlFinal = "https://script.google.com/macros/s/"+GOOGLE_SCRIPT_ID+"/exec?time="+seconds+"&temp="+temp+"&hum="+hum+"&pres="+pres;
95    Serial.println("POST data to spreadsheet: "+urlFinal);
96    sendtogoogle(urlFinal);
97
98    delay(5000);
99  }
```

My desktop


myIRGA_TMP
File Edit View Insert Format Data Tools Extensions Help
K1:K7
=SPARKLINE(C8:C)
formulas
notes
graphs
photos
CO2ppm vs. time
CO2 (ppm)
time (sec)
humidity (%)
°C
LAST VALUE
average
Legend of notes
Ambient air
BME280 together with IRGA in a
IRGA in soil chamber
IRGA in soil chamber+FAN
Control process duration (min)
start time (sec) process
update working time (min)
<-- put here the cell of starting time
Tuesday 19/03/2024

Benefits of using Google sheets

- Simplified Data Logging
- Versatile access
- Easy to use custom functions and google apps integration
- Seamless Collaboration
- Efficient Note-taking

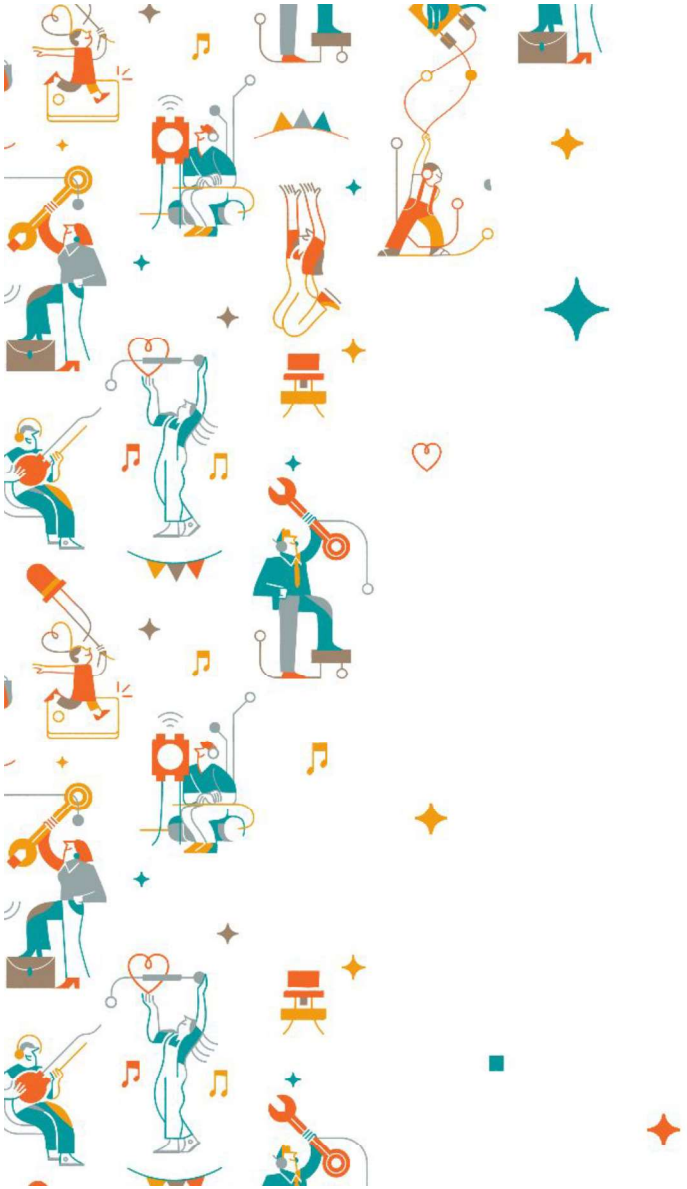
Data Logging is straightforward and robust and doesn't need any third-party services

Supports both desktop and mobile access: easy to use it also in the field

- Easy manipulation and analysis of collected data with functions.
- Simplify data monitoring with conditional formatting

Enable simultaneous collaboration with others, fostering teamwork and enhancing productivity.

Easy to record notes during data acquisition, beneficial during the setup phase of sensor deployment.



Thank you!