T2.Using the BlueLab IoT System

IARIA - ICONS 2020 – Lisboa, Portugal
Vitor Vaz da Silva
vsilva@deetc.isel.ipl.pt
(ISEL/IPL CTS-UNINOVA)
Instituto Superior de Engenharia de Lisboa/ Instituto Politécnico de Lisboa
Centre of Technology and Systems – Uninova – Caparica
NexComm 2020 - 23-27 February 2020

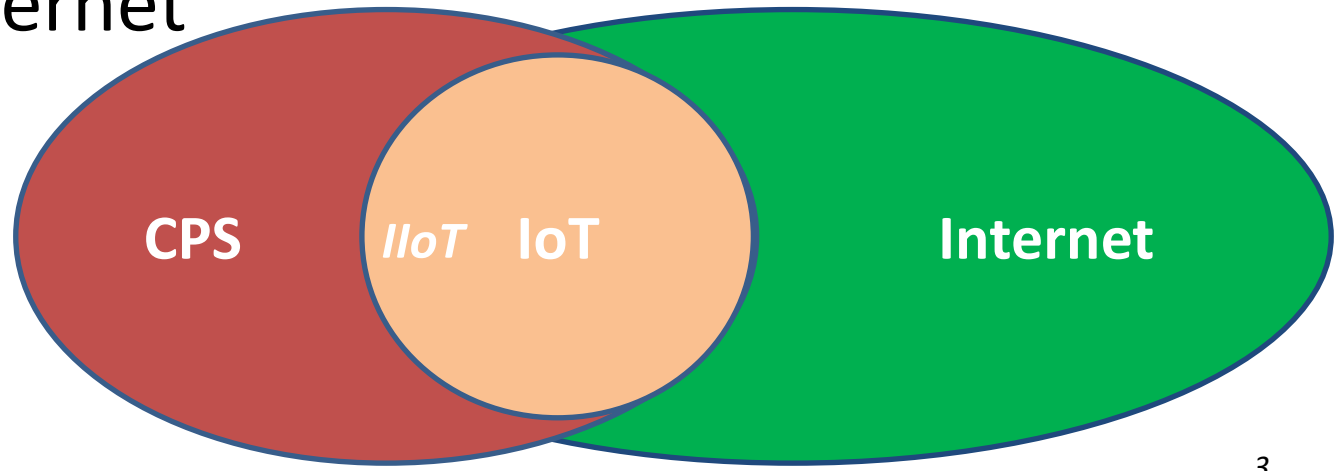
- IoT / CPS
- Other Systems
  - AskSensors
  - uBeac
  - Temboo – Kosmos
  - Thinger
- BlueLab IoT System
  - Demonstration
    - Android smartphone
    - Physical stations – ESP8266, ESP32

IoT - CPS

- IoT – Internet of Things *(Popular term)*
- CPS – Cyber Physical Systems *(Academic)*

IoT – needs Internet

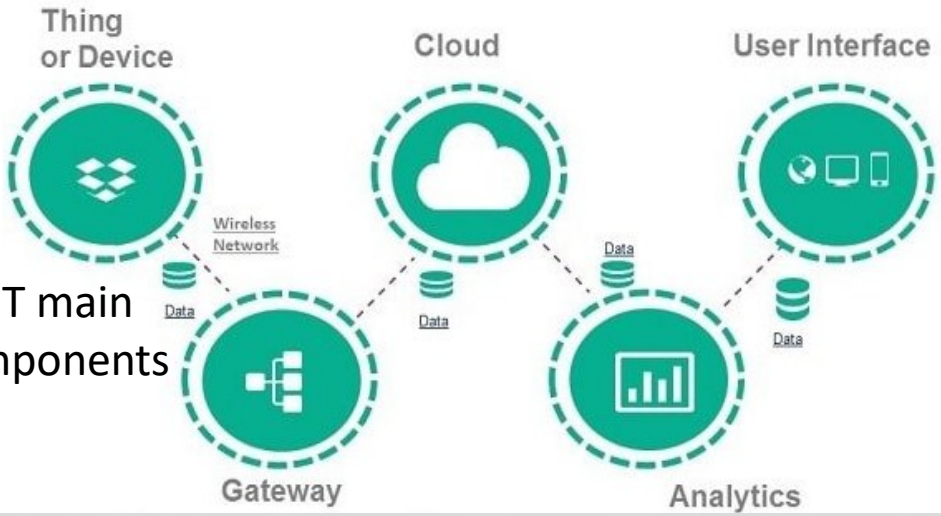
CPS – uses communications but not necessarily the Internet

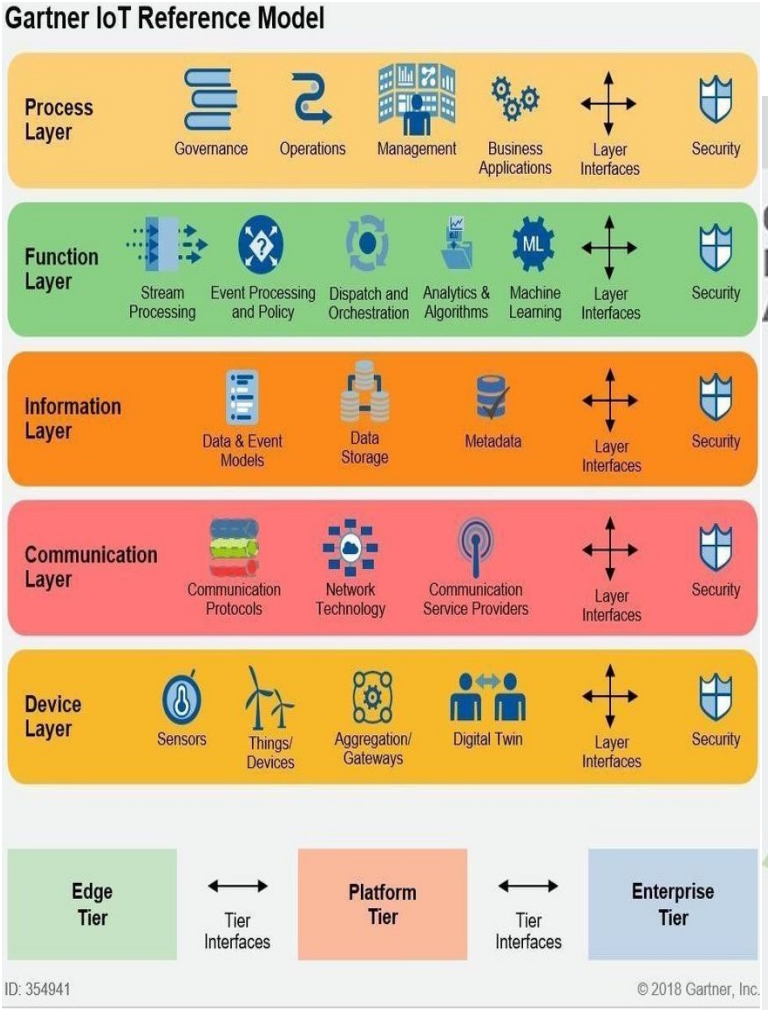


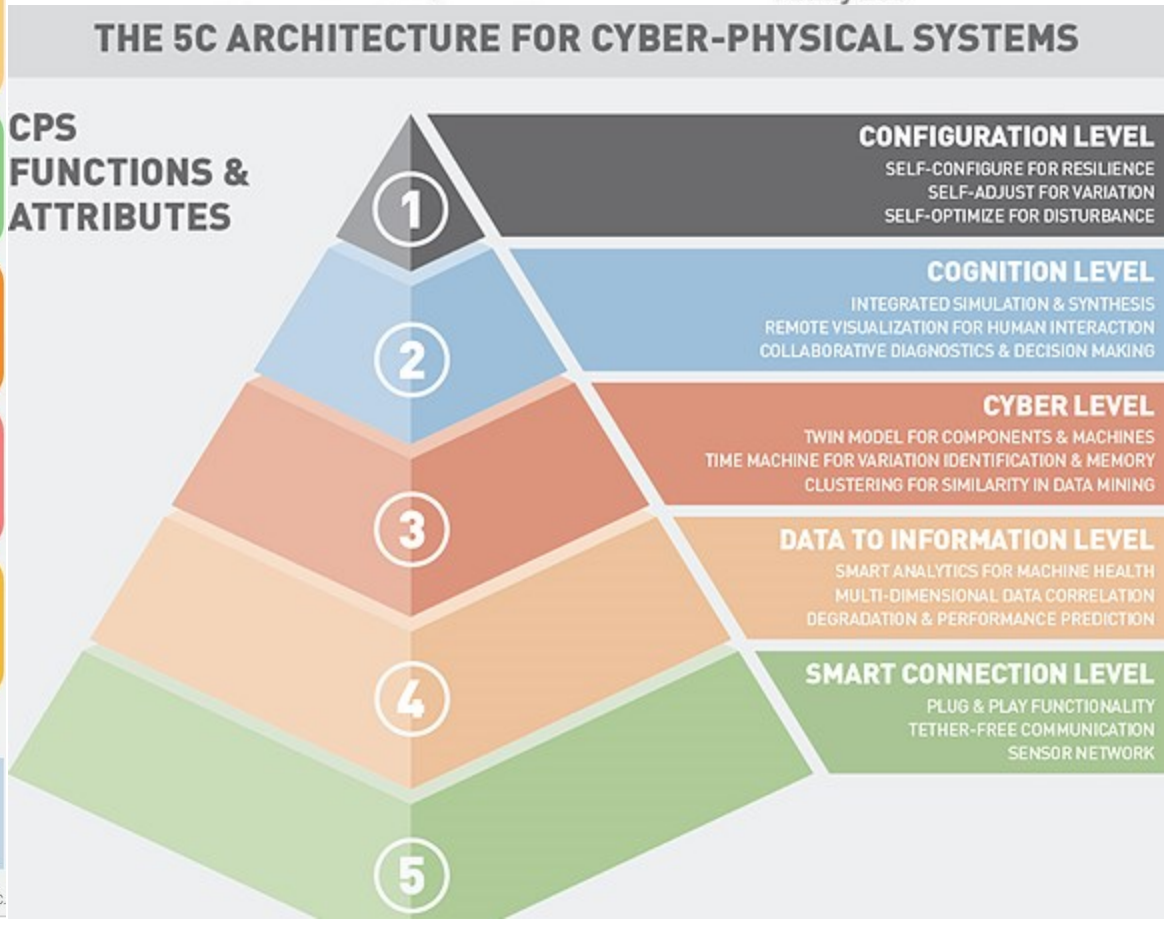
IIoT – Industrial IoT

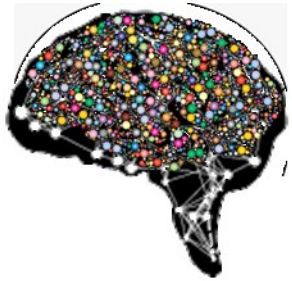
Models

IoT main components









Reality – Sensing – Remembering

Recalling - Acting

Tools and Methods Changed

Platforms that support Expressif chips

https://www.espressif.com/en/ecosystem/cloud-platform

Compare Systems

AskSensors

Kosmos

BlueLab IoT

AskSensors

- Register by e-mail
- Create Sensor and Modules
- Each Sensor has an API key ex: FALOAPPKH17ZR4Q23A8U9W0XPJL0F6OG
- Alarm – min-max value and time interval (E-mail)
- Displays module values (several graphic types) auto-refreshed
- Exports CSV data

AskSensors

- Actuators *(each has a API Key out)*
- An actuator may have modules
- Dashboard
- 15-90 day free trial, 2-60 devices (sensor/actuator)
- Protocol - *Http/Https & Mqtt API*
- *JSON data*
- Several plans *(unlimited data storage @ 30 month data retention)*

AskSensors

```
********** connecting to HOST : api.asksensors.com
********** requesting URL: /write/RS6EZyqRDlo71vn73zPWjEObgz3vR4gc?module1=26
> Request sent to ASKSENSORS
********** ASKSENSORS replay:
1
********** closing connection
********** requesting URL: /read/gaHIGUwyfuc3N8ZyhxBAZOd4xvR3fjSE?module=module1&maxResults=1
HTTP/1.1 200 OK

********** connecting to HOST : api.asksensors.com
********** requesting URL: /write/RS6EZyqRDlo71vn73zPWjEObgz3vR4gc?module1=10
> Request sent to ASKSENSORS
********** ASKSENSORS replay:
1
********** closing connection
********** requesting URL: /read/gaHIGUwyfuc3N8ZyhxBAZOd4xvR3fjSE?module=module1&maxResults=1
HTTP/1.1 200 OK
```

AskSensors Alert

Hi victorix

The data read on your sensor has exceeded the alert threshold:

Sensor : humi
Module : Module 1
minThreshold : 40
maxThreshold : 60
Value : 70
Date : 12 Feb 2020 23:09:38 UTC

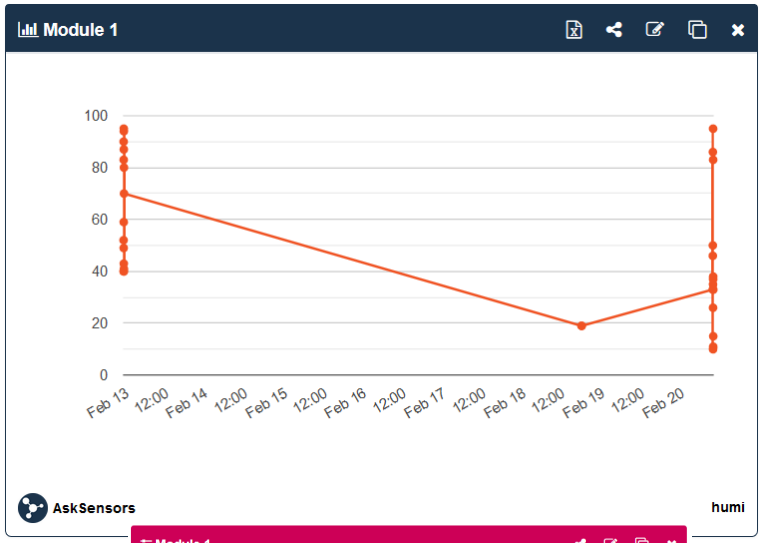
You receive this email because you have enabled an AskSensor email Alert.
Login to your AskSensors account

Thanks for trying AskSensors!

All the best,

The AskSensors team

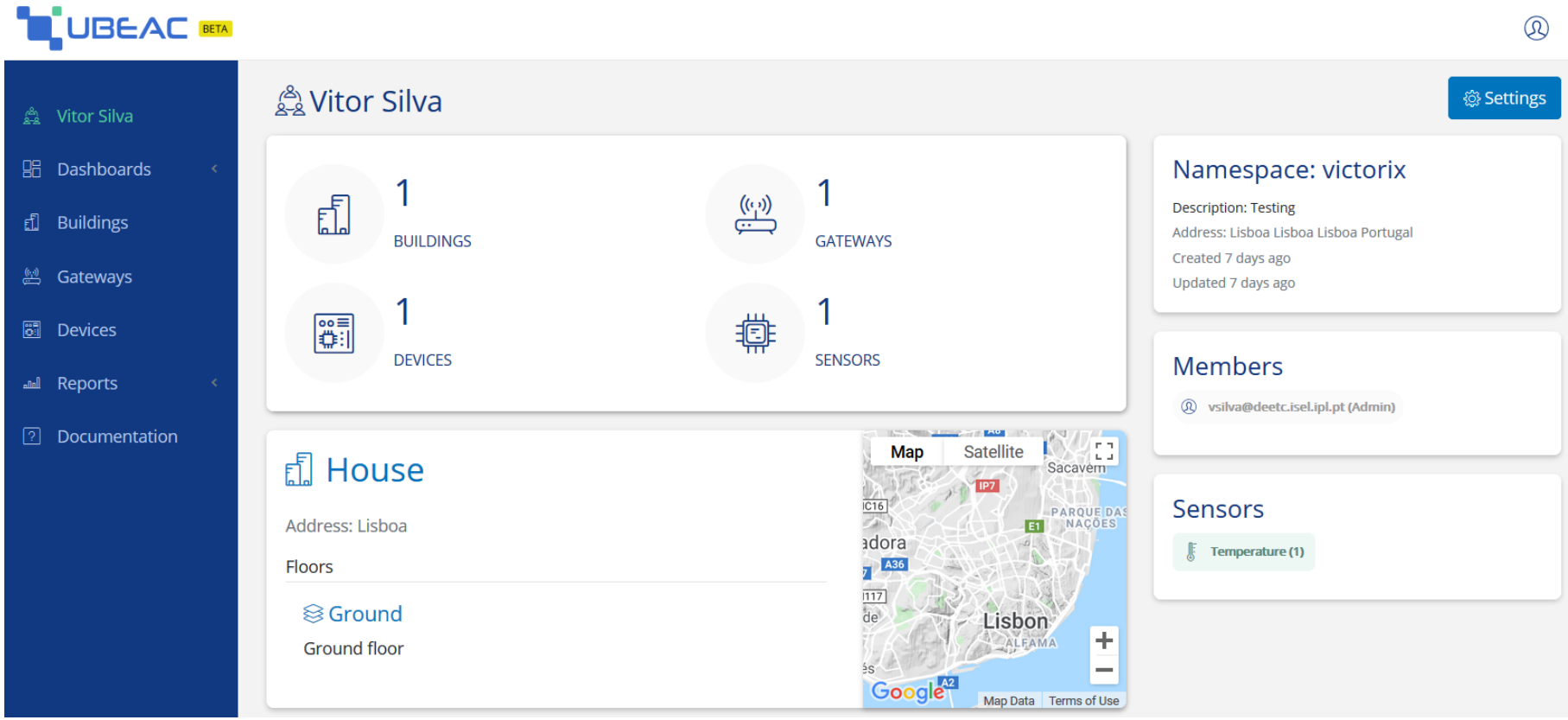
This email was sent to vsilva@deetc.isel.ipl.pt by AskSensors.



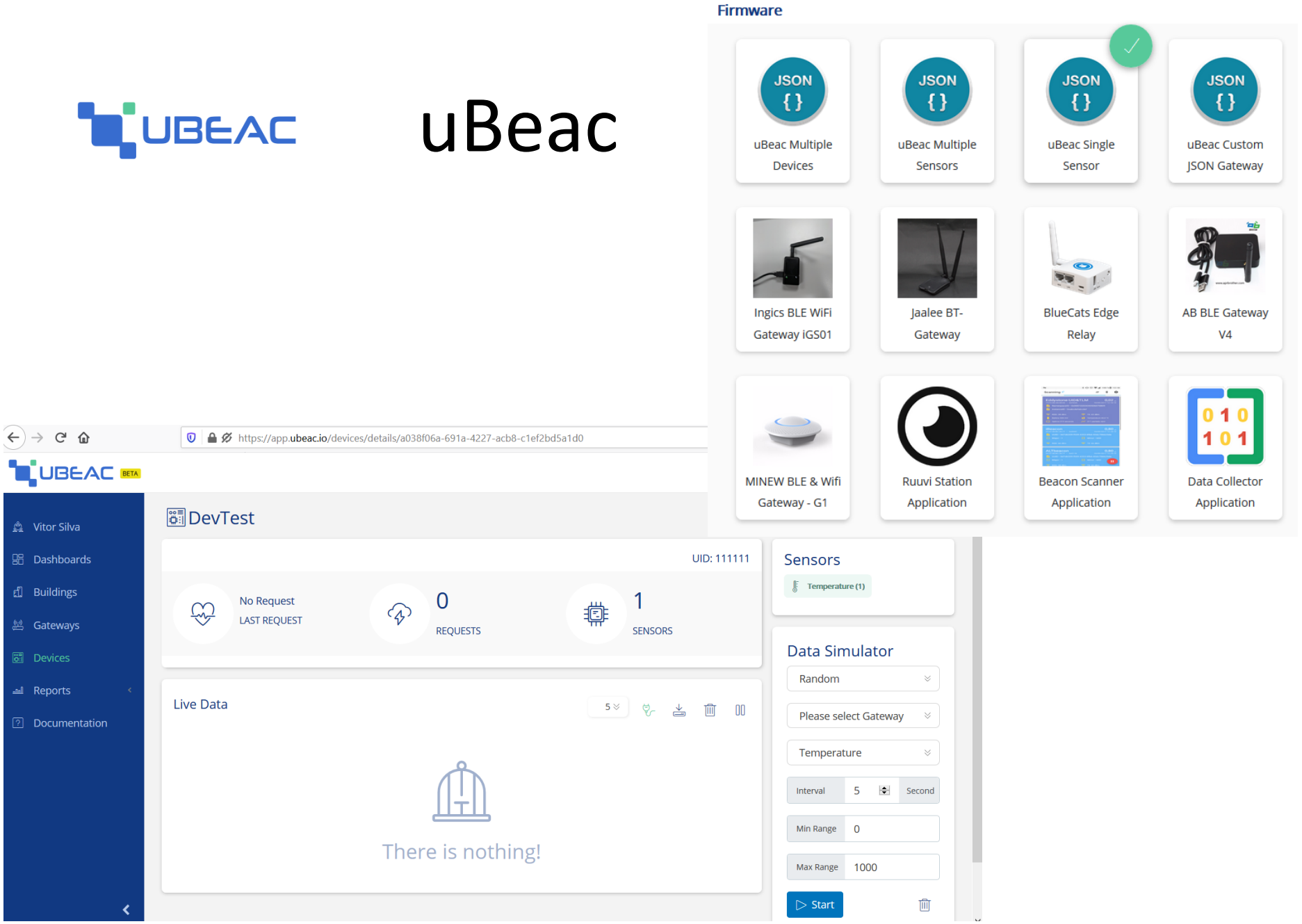
uBeac

- Register by e-mail
- Teams, Buildings, Floor
- Create Device, Sensor
- Create Gateway – several types
- Create an EndPoint
- Protocols HTTP(s) MQTT(s)

uBeac


UBEAC
BETA
Vitor Silva
Dashboards
Buildings
Gateways
Devices
Reports
Documentation
Vitor Silva
Settings
1
BUILDINGS
1
GATEWAYS
1
DEVICES
1
SENSORS
House
Address: Lisboa
Floors
Ground
Ground floor
Map
Satellite
Lisbon
Google
Map Data
Terms of Use
Namespace: victorix
Description: Testing
Address: Lisboa Lisboa Lisboa Portugal
Created 7 days ago
Updated 7 days ago
Members
vsilva@deetc.isel.ipl.pt (Admin)
Sensors
Temperature (1)

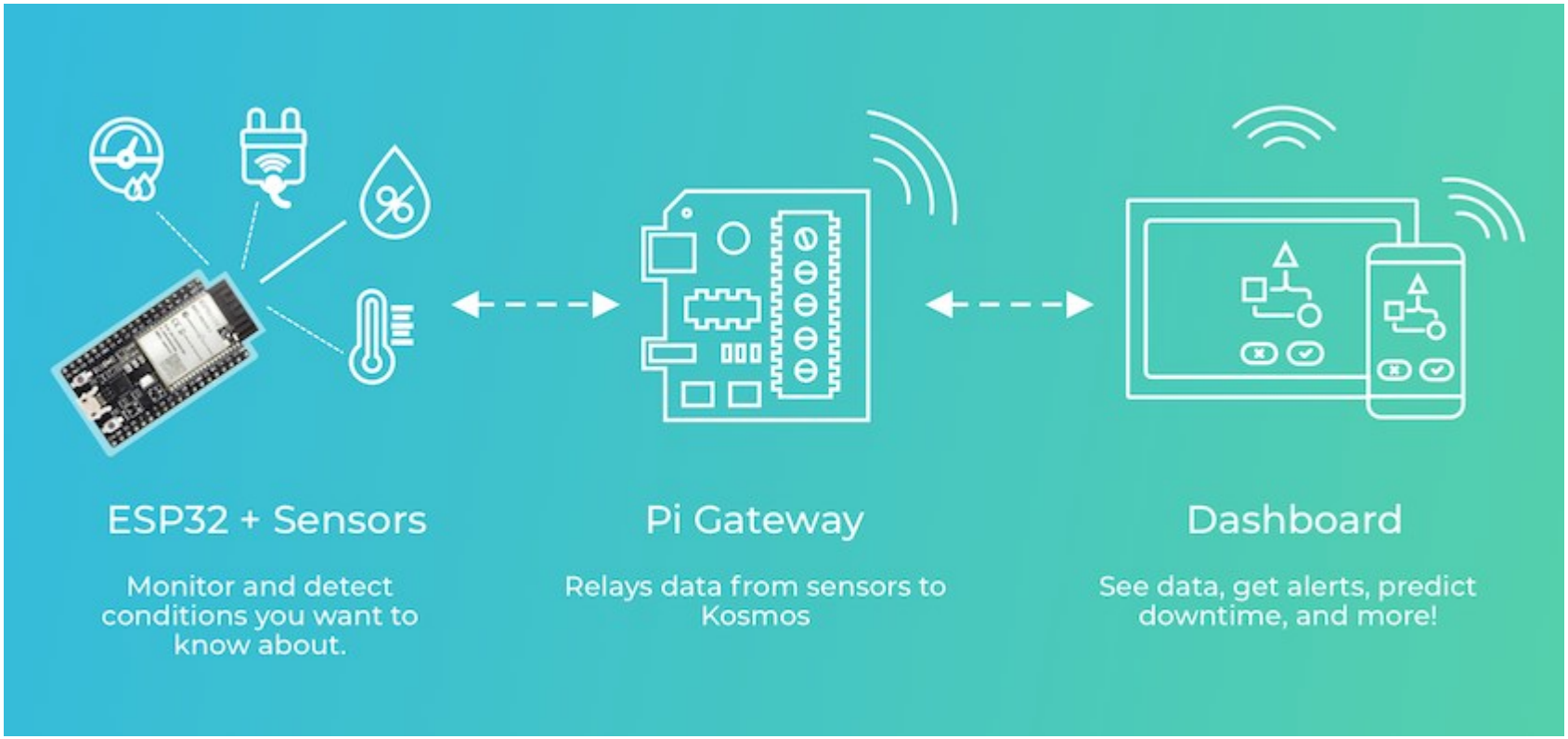
uBeac


Firmware
uBeac Multiple Devices
uBeac Multiple Sensors
uBeac Single Sensor
uBeac Custom JSON Gateway
Ingics BLE WiFi Gateway iGS01
Jaalee BT-Gateway
BlueCats Edge Relay
AB BLE Gateway V4
MINEW BLE & Wifi Gateway - G1
Ruuvi Station Application
Beacon Scanner Application
Data Collector Application
https://app.ubeac.io/devices/details/a038f06a-691a-4227-acb8-c1ef2bd5a1d0
UBEAC BETA
Vitor Silva
Dashboards
Buildings
Gateways
Devices
Reports
Documentation
DevTest
UID: 111111
No Request
LAST REQUEST
0
REQUESTS
1
SENSORS
Live Data
There is nothing!
Sensors
Temperature (1)
Data Simulator
Random
Please select Gateway
Temperature
Interval
5
Second
Min Range
0
Max Range
1000
Start

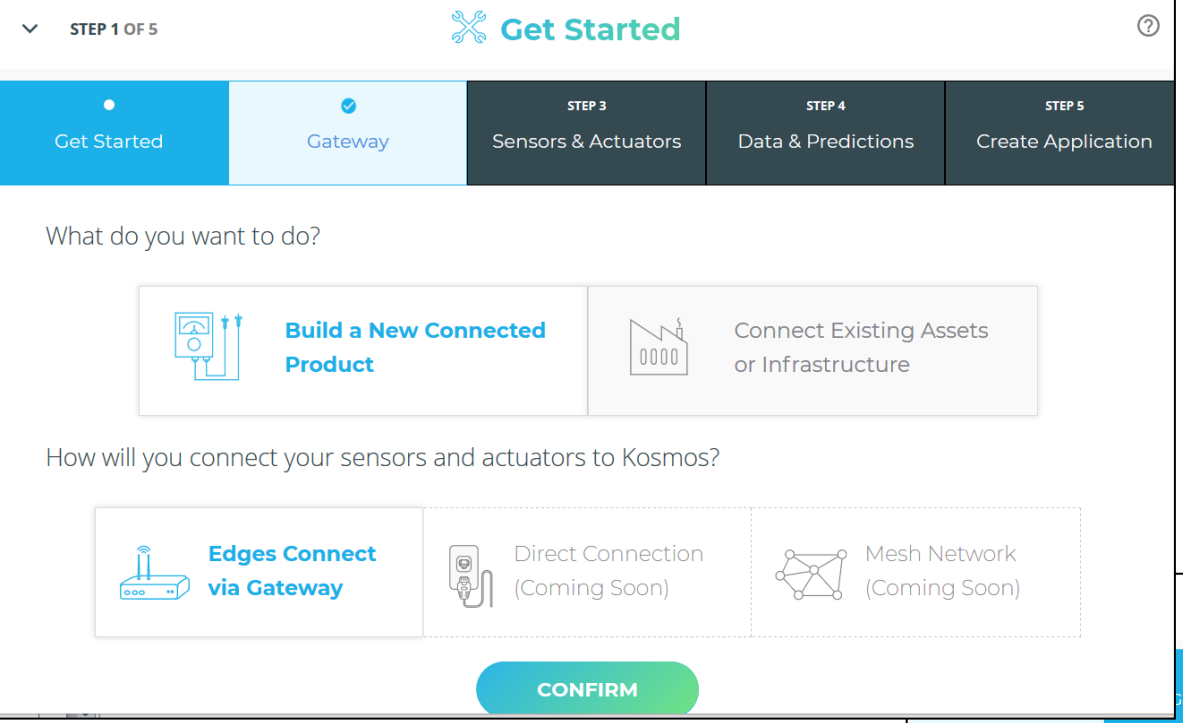
Temboo - Kosmos

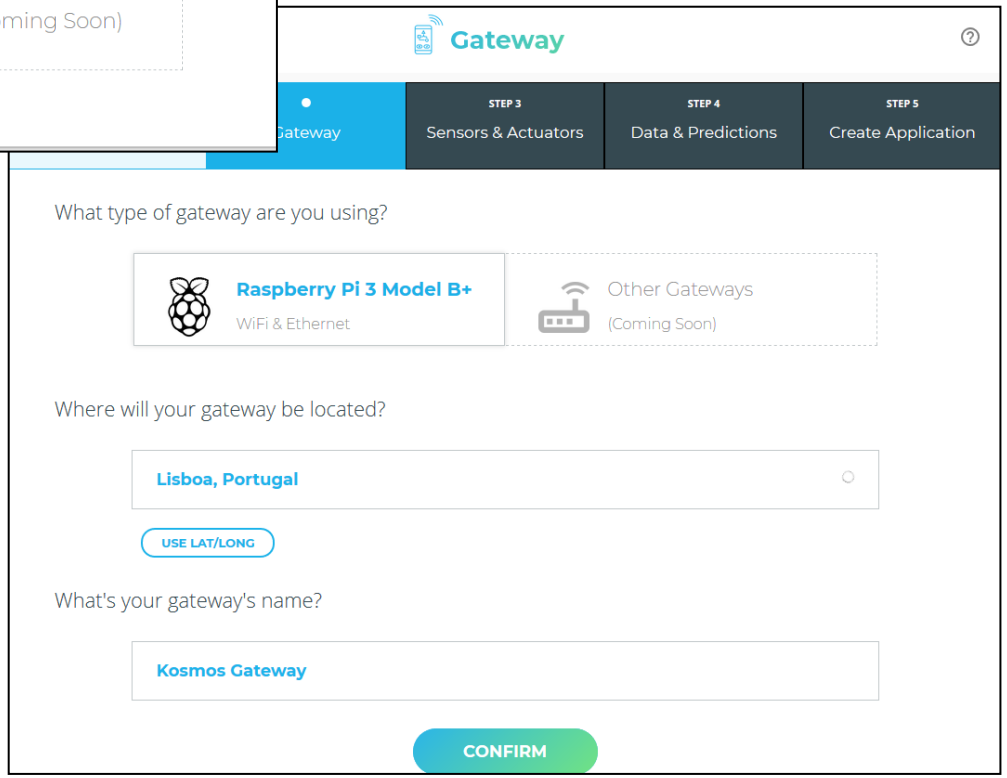
- Registration by email
- 5 steps to generate and download
  - gateway and device code
- Sensors and Actuators
- Alarms (and predictive)

Temboo - Kosmos

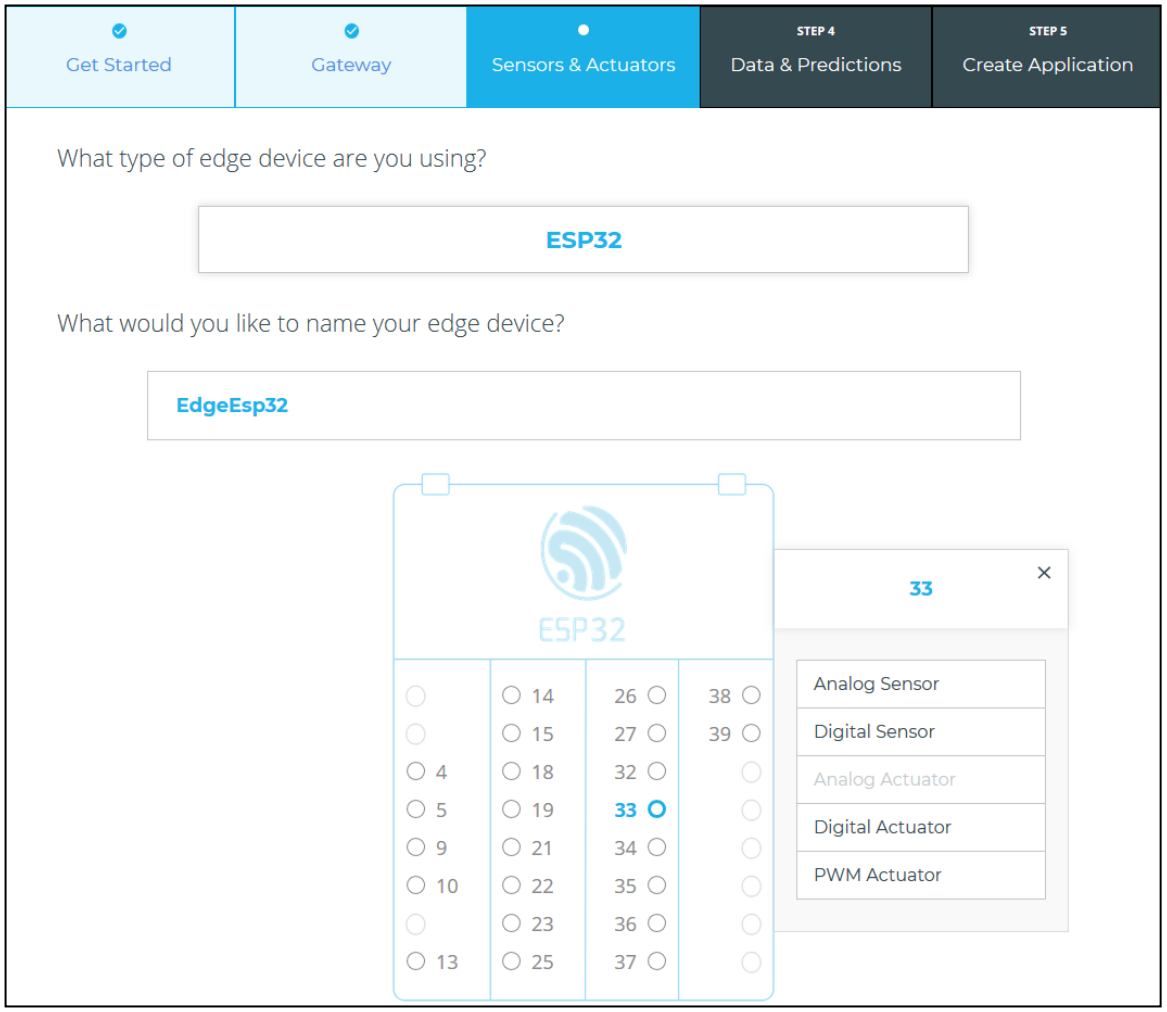


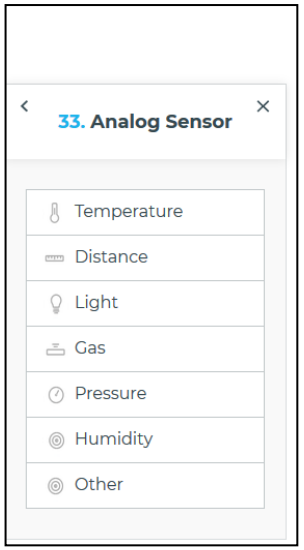
Temboo - Kosmos

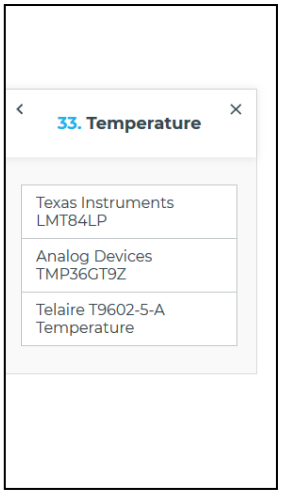


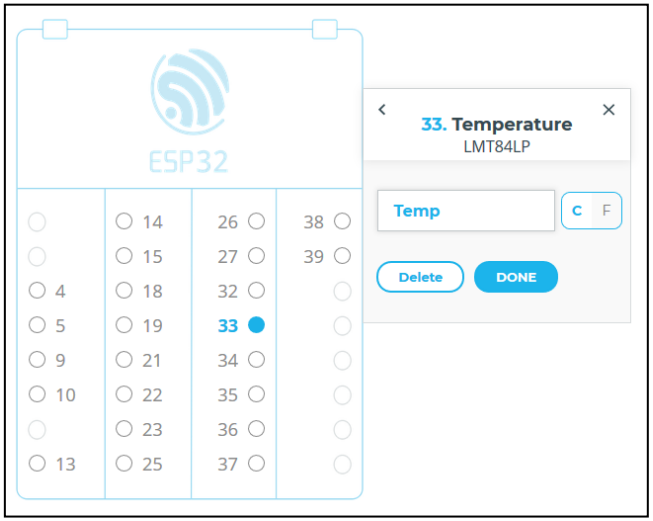


Temboo - Kosmos

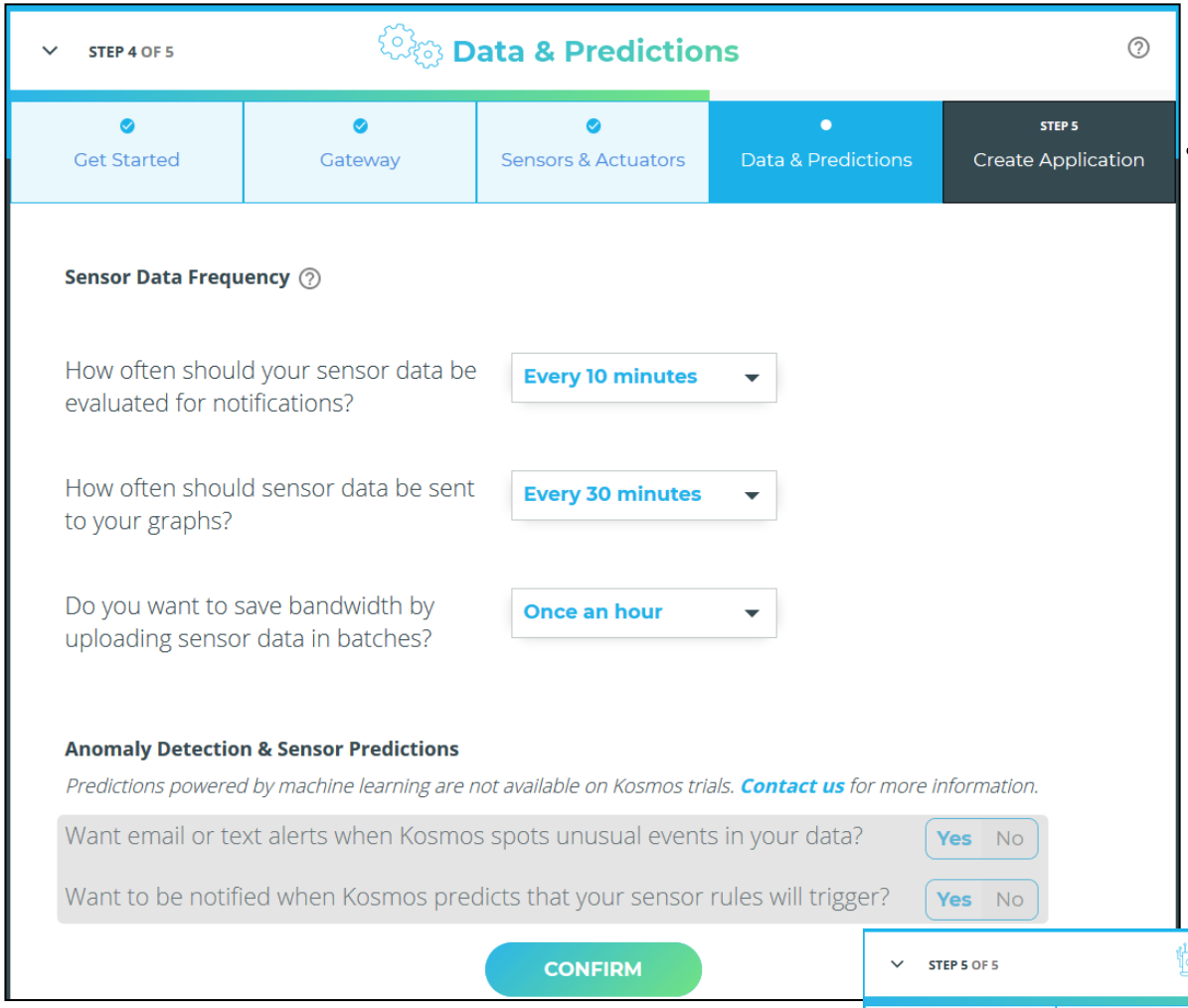


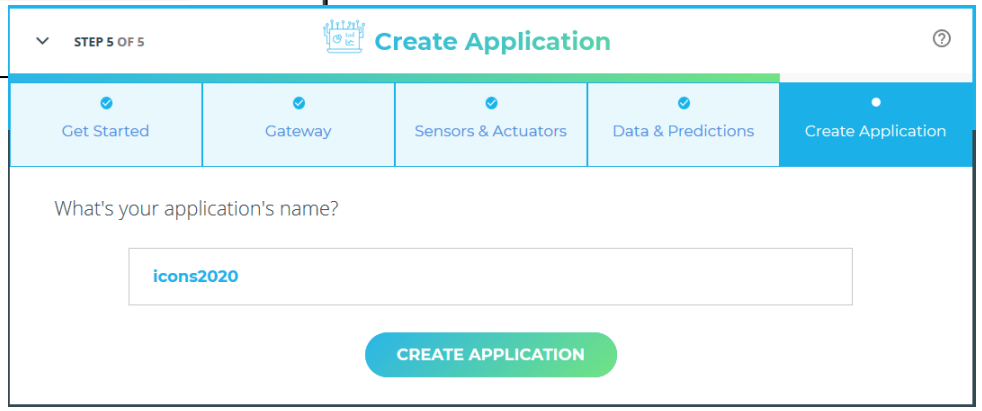




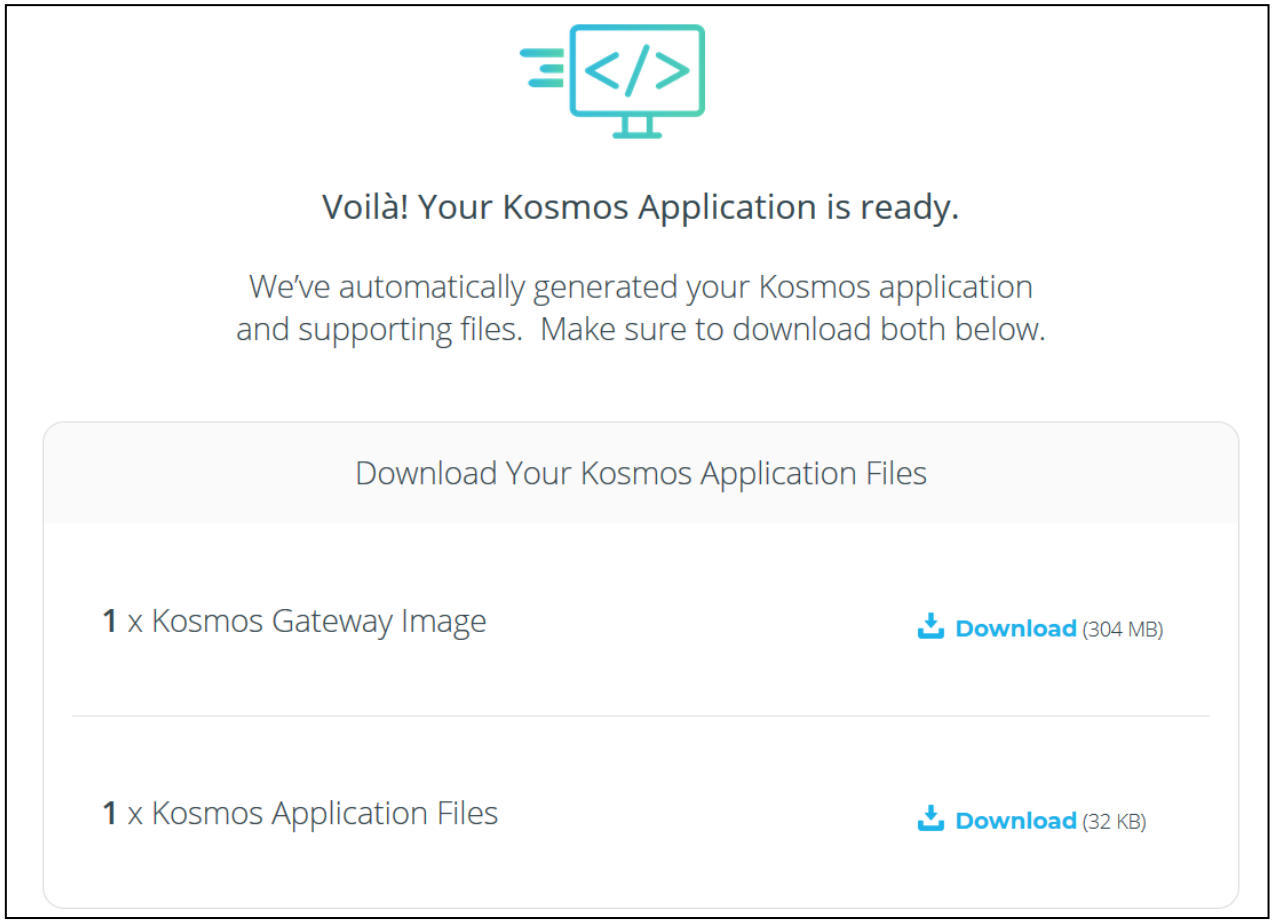


Temboo - Kosmos

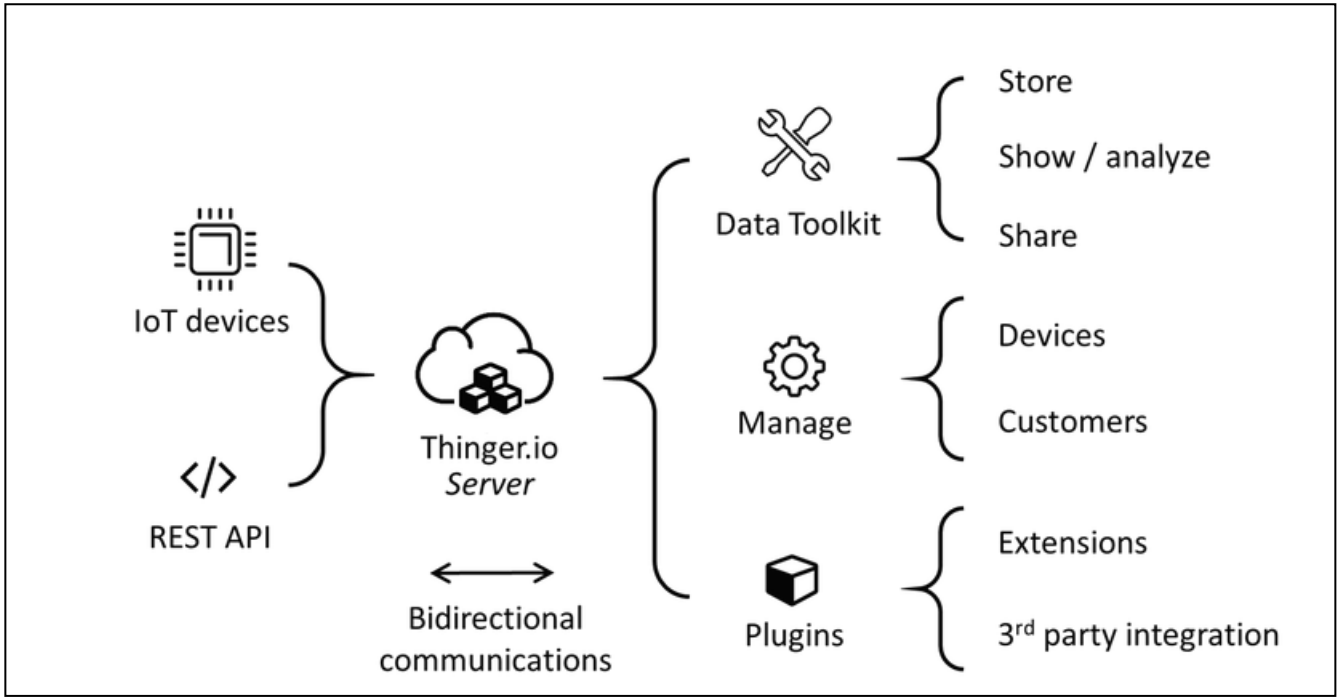




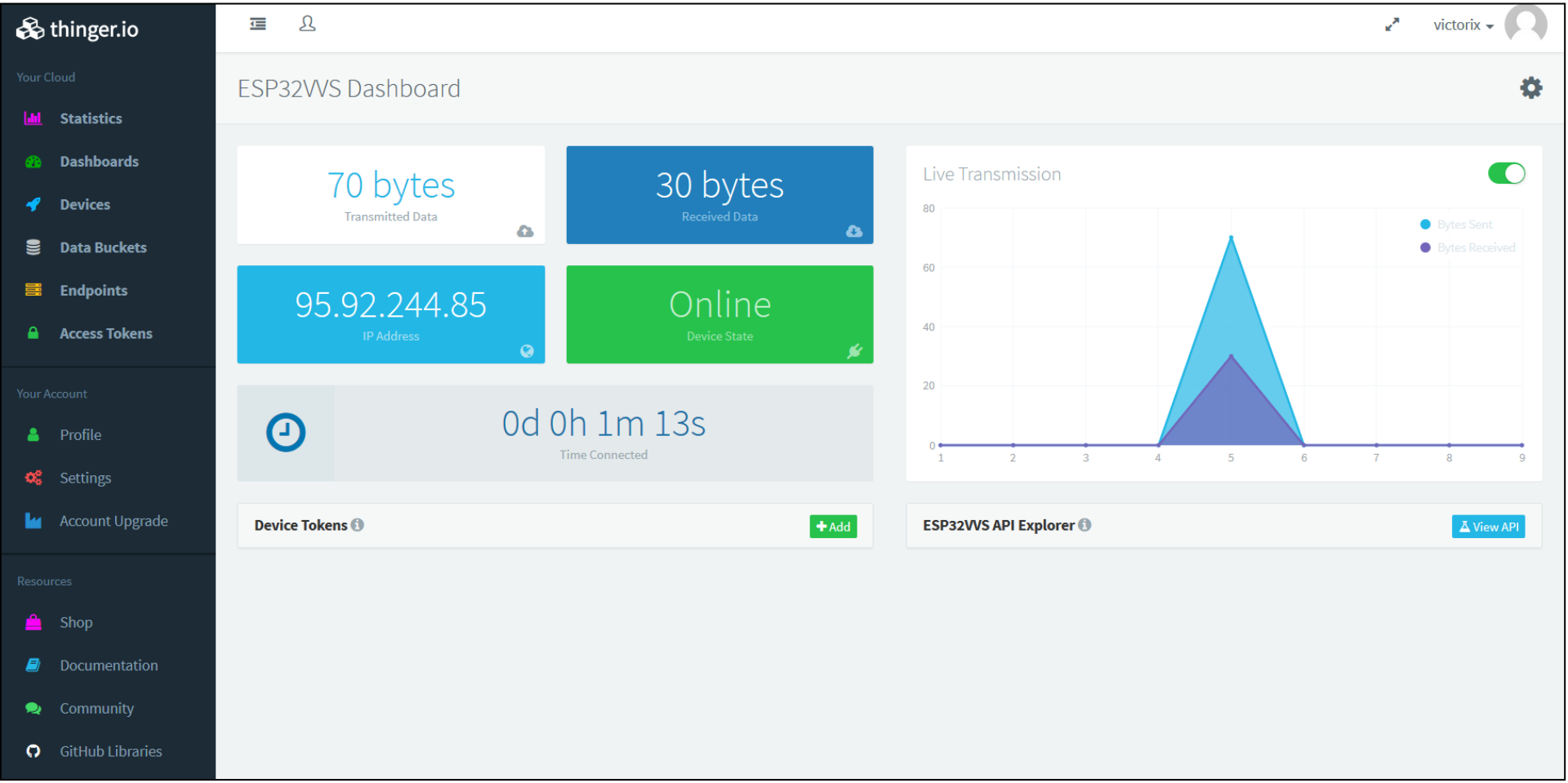
Temboo - Kosmos



Thinger

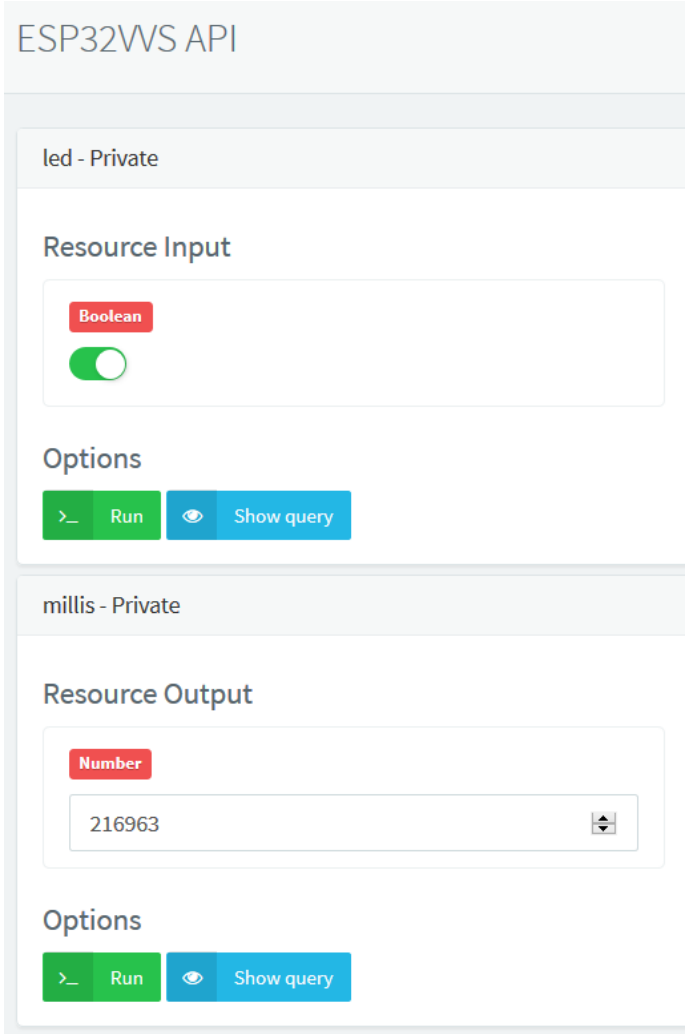

IoT devices
REST API
Thinger.io
Server
Bidirectional
communications
Data Toolkit
Store
Show / analyze
Share
Manage
Devices
Customers
Plugins
Extensions
3rd party integration

Thinger

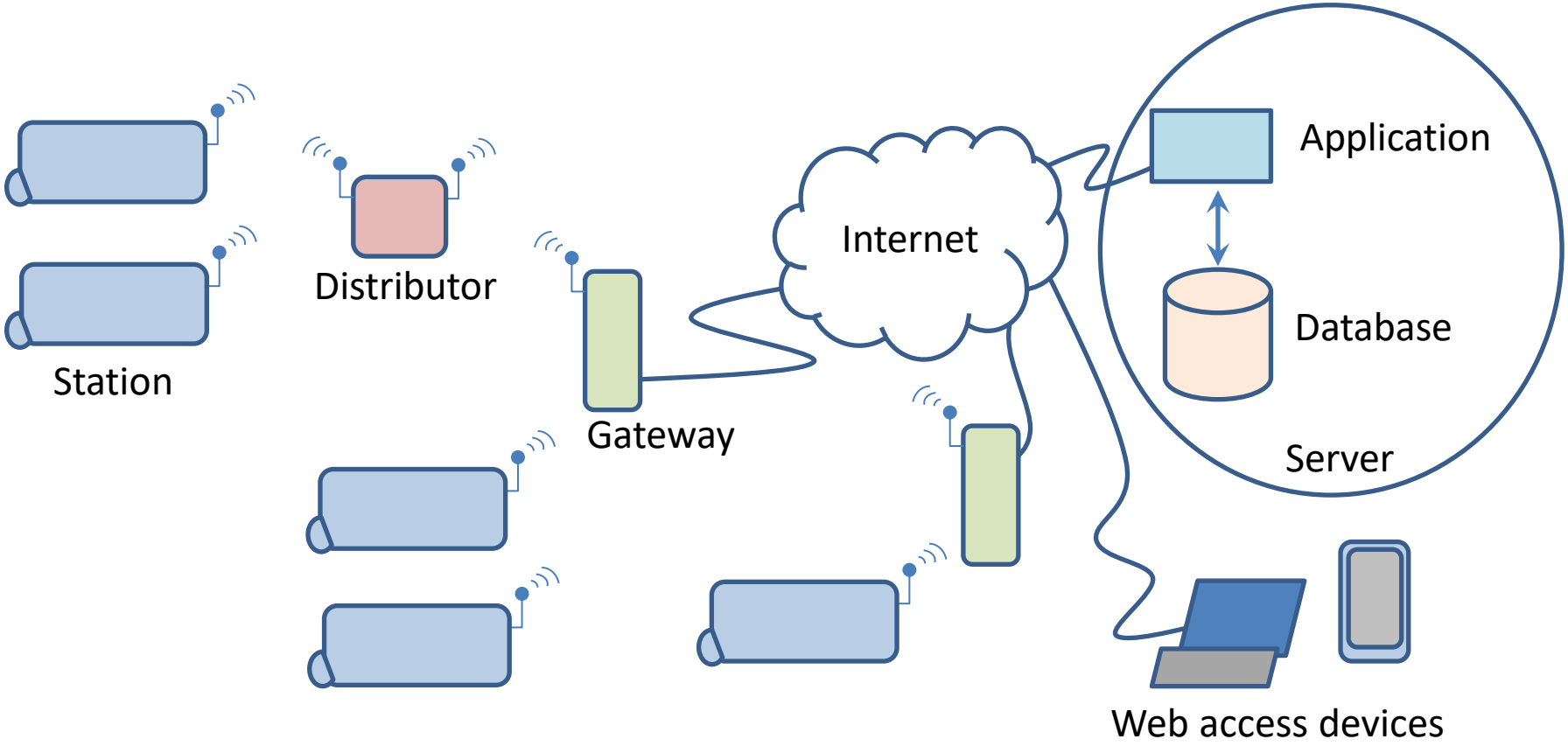

thinger.io
Your Cloud
Statistics
Dashboards
Devices
Data Buckets
Endpoints
Access Tokens
Your Account
Profile
Settings
Account Upgrade
Resources
Shop
Documentation
Community
GitHub Libraries
victorix
ESP32VVS Dashboard
70 bytes
Transmitted Data
30 bytes
Received Data
95.92.244.85
IP Address
Online
Device State
0d 0h 1m 13s
Time Connected
Live Transmission
Bytes Sent
Bytes Received
Device Tokens
Add
ESP32VVS API Explorer
View API

Thinger

```
#include <ThingerESP32.h>

#define USERNAME "victorix"
#define DEVICE_ID "esp32vvs"
#define DEVICE_CREDENTIAL "wxO0bNb0Wv&%"

#define SSID "MySSID"
#define SSID_PASSWORD "MyPassword"

#define LED_BUILTIN 2
ThingerESP32 thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);

void setup() {
  Serial.begin(115200);
  delay(4000);   //Delay needed ....

  pinMode(LED_BUILTIN, OUTPUT);

  thing.add_wifi(SSID, SSID_PASSWORD);

  // digital pin control example (i.e. turning on/off a light,
  // a relay, configuring a parameter, etc)
  thing["led"] << digitalPin(LED_BUILTIN);

  // resource output example (i.e. reading a sensor value)
  thing["millis"] >> outputValue(millis());

  // more details at http://docs.thinger.io/arduino/
  Serial.print("Setup Done");
}

void loop() {
  //Serial.print("."+String(millis()));
  thing.handle();
}
```



BlueLab IoT


Station
Distributor
Gateway
Internet
Application
Database
Server
Web access devices

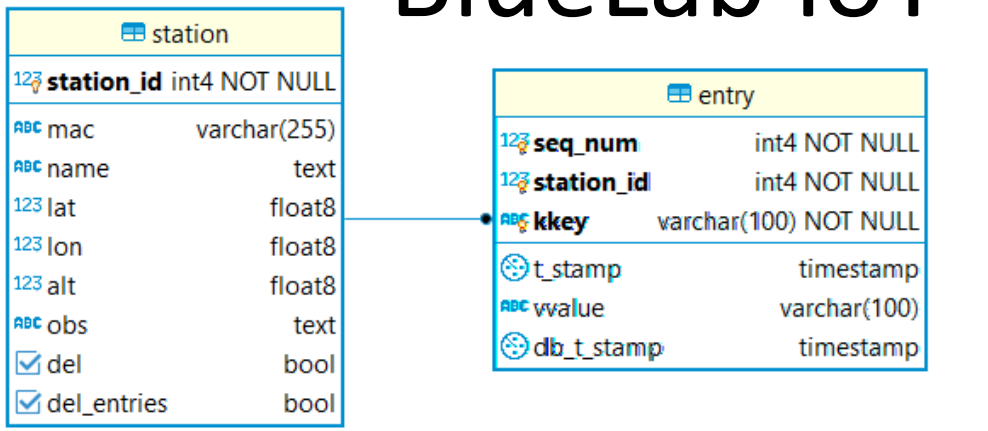
BlueLab IoT

- Register by e-mail or phone
- A user can have
  - stations
    - sensors
    - actuators
    - data is sent from the station to the system
    - the system does not *(yet)* act upon the actuators
  - profiles
    - for station authentication in the system
    - different stations can have the same profile

BlueLab IoT

- Station
  - communicates by WiFi
  - protocol https
  - can send alarms (e-mail or sms, … profile dependent)
  - unique id given by the system *(function of a physical unique id ex: MAC address)*
  - *Hardware used ESP8266, ESP32 (C++ code)*

BlueLab IoT



- Data
  - stored as a pair (key, value) that belongs to a station
  - has a sequence number *(starts with 1)*
  - has a device/sensor timestamp
  - has a database timestamp
  - *Errors and other information can also be sent as a key, value pair*
  - *Users can delete data.*
  - *Data cannot be edited*
  - *All data are Strings*
    - *waists space*
    - *easy to write/read – the context is with the sensor*

BlueLab IoT

Two different sensors in the same station with **different sample rates**, and their **values are buffered and then sent together**

Seq num	s1	s2	Database timestamp
N	Val1_1 Stime1_1	Val2_1 Stime2_1	Tstamp_N
N+1	Val1_2 Stime1_2	Val2_2 Stime2_2	Tstamp_N
N+2		Val2_3 Stime2_3	Tstamp_N
N+3		Val2_4 Stime2_4	Tstamp_N
N+4			

Two different sensors in the same station with **different sample rates**, and their **values are sent after sampling**

Seq num	s1	s2	Database timestamp
N		Val2_1 Stime2_1	Tstamp_N
N+1		Val2_2 Stime2_2	Tstamp_N_1
N+2	Val1_1 Stime1_1		Tstamp_N_2
N+3		Val2_3 Stime2_3	Tstamp_N_3
N+4		Val2_4 Stime2_4	Tstamp_N_4

BlueLab IoT

Two different sensors in the same station with **equal sample rates**, and their **values are buffered and then sent together**

Seq num	s1	s2	Database timestamp
N	Val1_1 Stime1_1	Val2_1 Stime2_1	Tstamp_N
N+1	Val1_2 Stime1_2	Val2_2 Stime2_2	Tstamp_N
N+2			

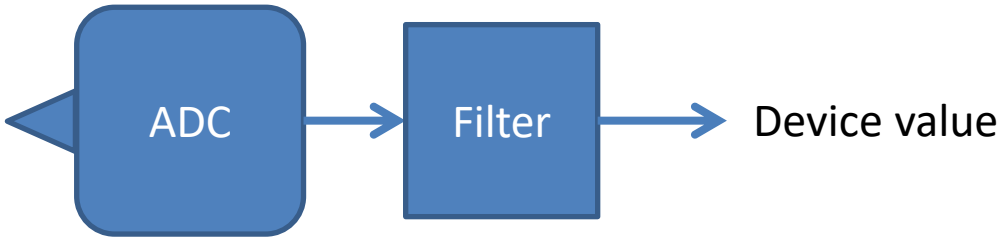
Two different sensors in the same station with **equal sample rates**, and their values are **sent together after sampling**

Seq num	s1	s2	Database timestamp
N	Val1_1 Stime1_1	Val2_1 Stime2_1	Tstamp_N
N+1	Val1_2 Stime1_2	Val2_2 Stime2_2	Tstamp_N_1
N+2			

BlueLab IoT

- Data - Parameters
  - stored as a pair (key, value) that belongs to a station
  - has a sequence number *always 0*
  - has a device/sensor timestamp (if the station stored it)
  - has a database timestamp
  - used for additional information of the sensor (key)
  - Special parameters: **#key_** *followed by* **0:n** | **U** | **O** | **S**
    - 0:n – $x^{0:n}$, U – units, O – offset, S – scale
    - >float, <float – clamp values

BlueLab IoT – Data parameters



- Sensor value x = Offset + Scale * device value
- *(this transformation may be necessary for example due to the ADC range, in volts and in bits)*

- Sensor parameters $a^n x^n, n \in R$
- *(these parameters may be necessary due to the non linear response of the sensor)*

- Data parameters do not change the stored data values

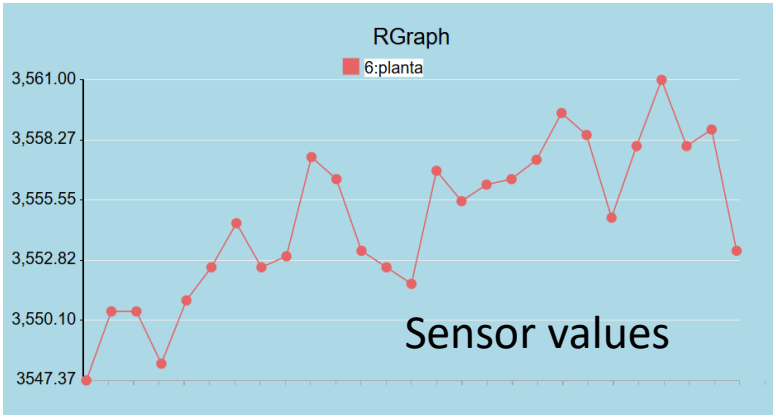
BlueLab IoT – Data parameters

- Moisture sensor SEN0193 *(Reference 16)*
- $\theta = 13.248 - 2.576 \cdot 10^{-3}x + 1.726 \cdot 10^{-7}x^2 - 3.839 \cdot 10^{-12}x^3$

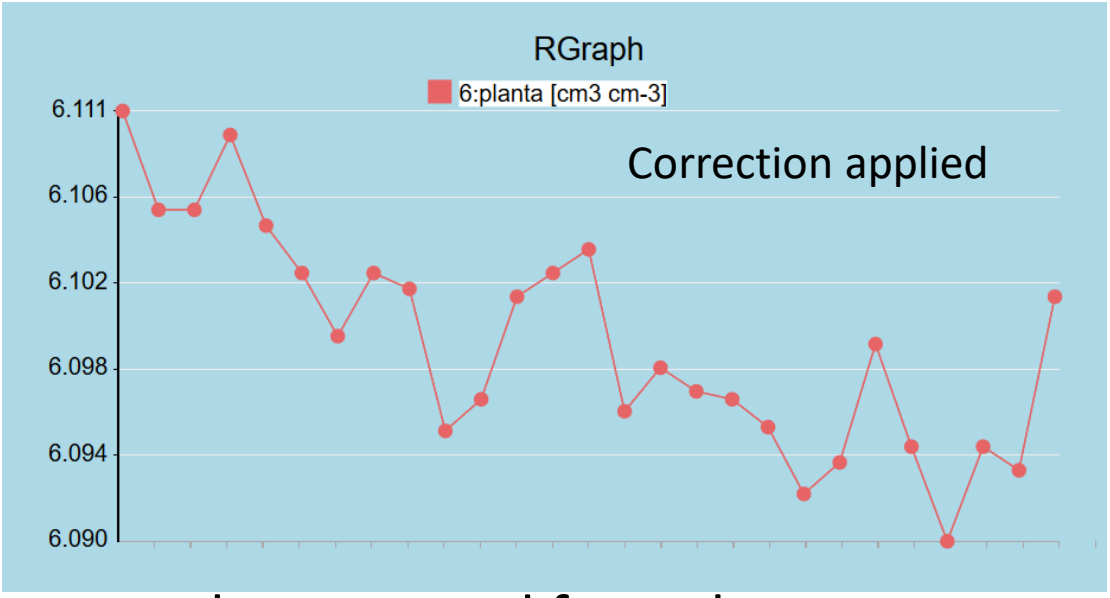
Station id	6
Seq number	0
timeStamp	Mon Feb 17 2020 09:29:14 GMT+0000 (Western European Stai
key	#planta_3
value	-3.839e-12
	Insert

Station:Parameter	Value
1:#bat_1	0.00645
2:#bat_1	0.00645
2:#bat_U	V
2:#humi_-1	1024
2:#humi_U	rH%
4:#bat_1	0.006445
4:#humi_-1	1024
5:#bat_1	0.006445
5:#humi_-1	1024
6:#planta_0	13.248
6:#planta_1	-2.576e-3
6:#planta_2	1.726e-7
6:#planta_3	-3.839e-12

- x - Sensor value [raw counts]
- θ - Volumetric water content [cm^3 cm^{-3}]



Air – Sensor is surrounded by air



Values are stored in the database as they are read from the sensors;

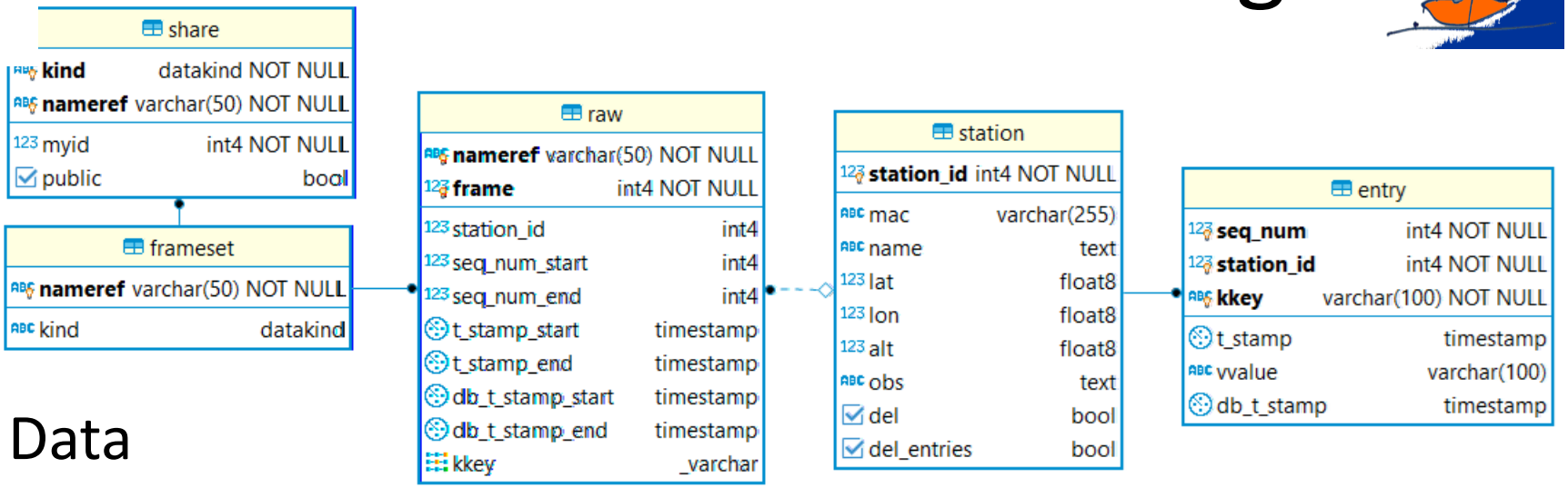
can be viewed and downloaded

Sensor parameters can be viewed, downloaded and changed

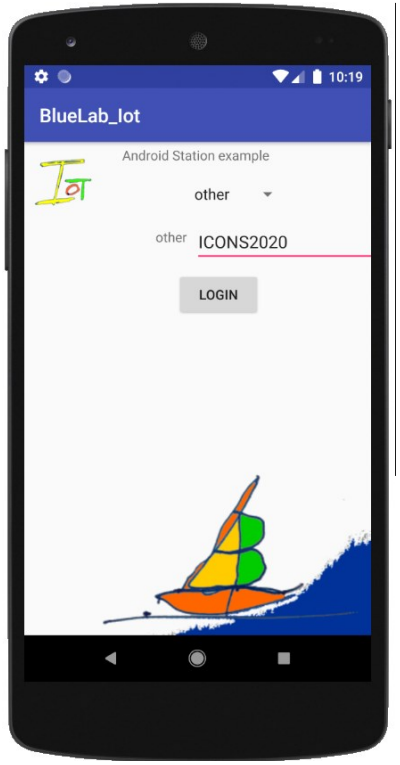
(presently they have the timestamp of the calibration; in future, calibration history will be added to account for sensor decay and correctly apply the parameters to the data)

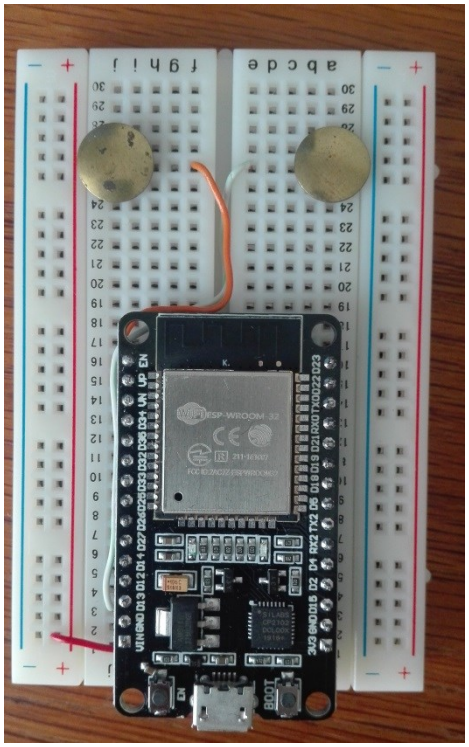
Calibration corrections are applied for viewing, and processing

BlueLab IoT – Data Sharing



- Data
  - Can be classified as raw (frames)
  - A frameset is a set of raw data (frames)
  - Framesets can be shared *(among projects of the user – not yet done)*
  - Shared framesets can be public – shared with all users
  - There is no data duplication *(only database table views)*
  - *Data parameters are shared as well*



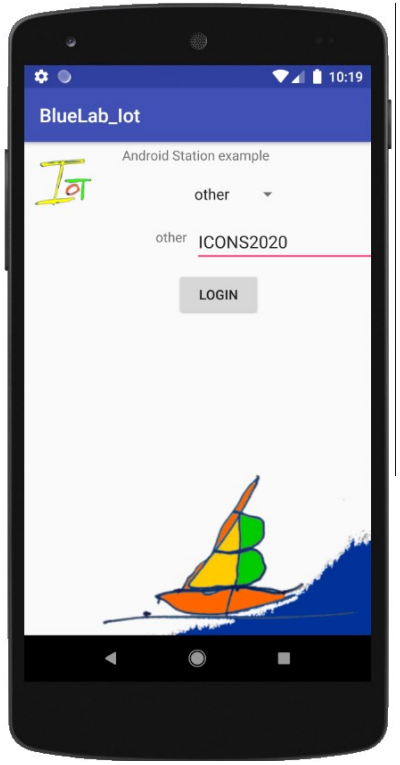


https://bluelab.pt/iot

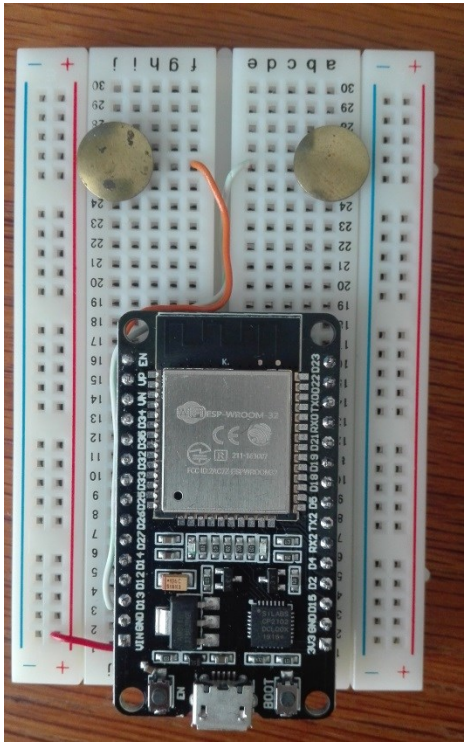
Live examples

github.com/tektonia/bluelab_iot

Web interface for the user

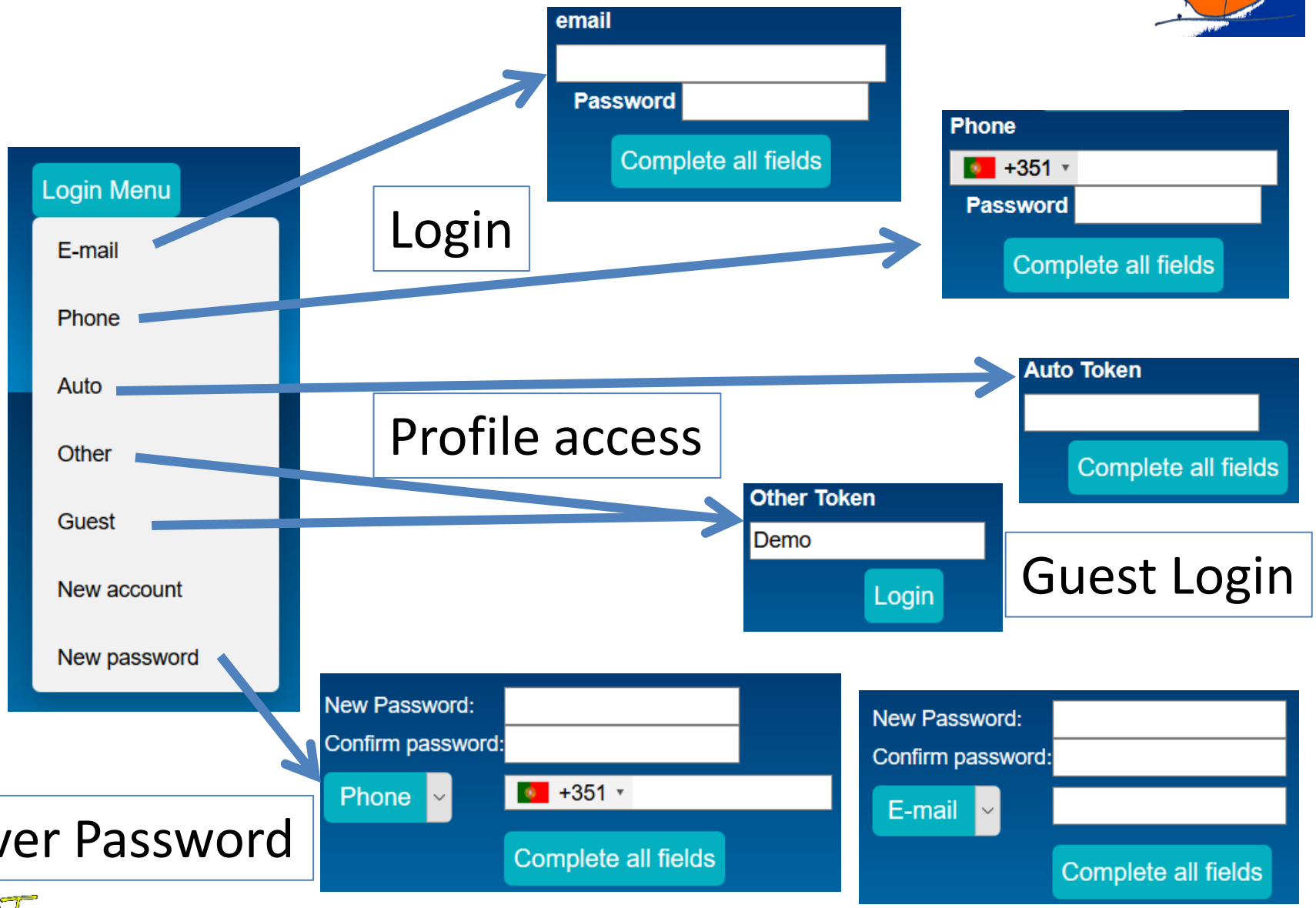


Example of a station – Android App
- Light sensor
- GPS sensor



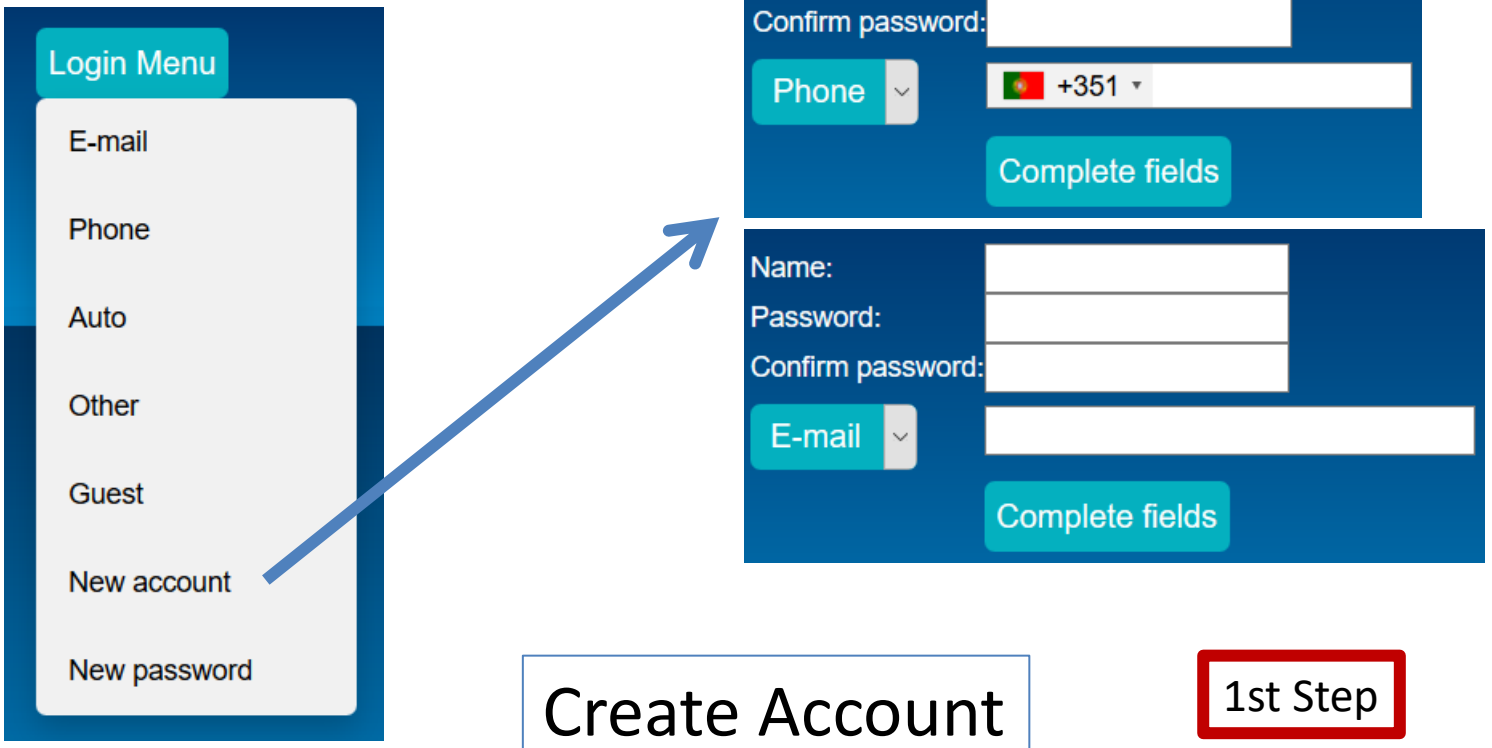
Example of a station – ESP32
- ADC sensor

BlueLab IoT


email
Password
Complete all fields
Login
Login Menu
E-mail
Phone
Auto
Other
Guest
New account
New password
Phone
+351
Password
Complete all fields
Auto Token
Complete all fields
Profile access
Other Token
Demo
Login
Guest Login
New Password:
Confirm password:
Phone
+351
Complete all fields
New Password:
Confirm password:
E-mail
Complete all fields
Recover Password

BlueLab IoT

https://bluelab.pt/iot



Create Account

1st Step

BlueLab IoT

Profile Access – Is a direct login token that does not need password

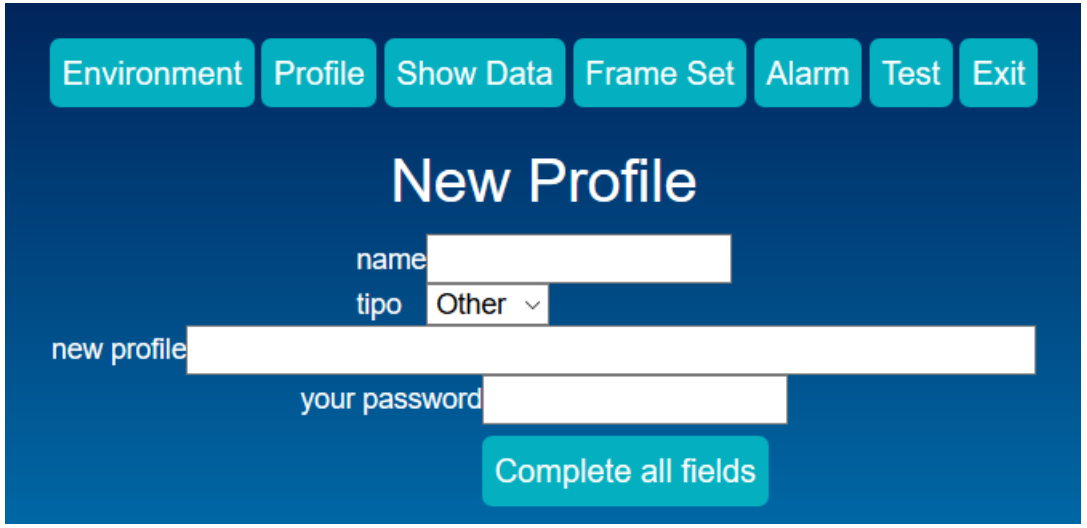
- Preferably used in each station for accessing the system
- More than one station can have the same profile
- Each station has a unique identifier (ex: MAC address)
- Reduced privileges; used for storing data
  does not allow deletion or data sharing
- Created by a login (email or phone) with full privileges

Unique in the system

Other – is for ease of use

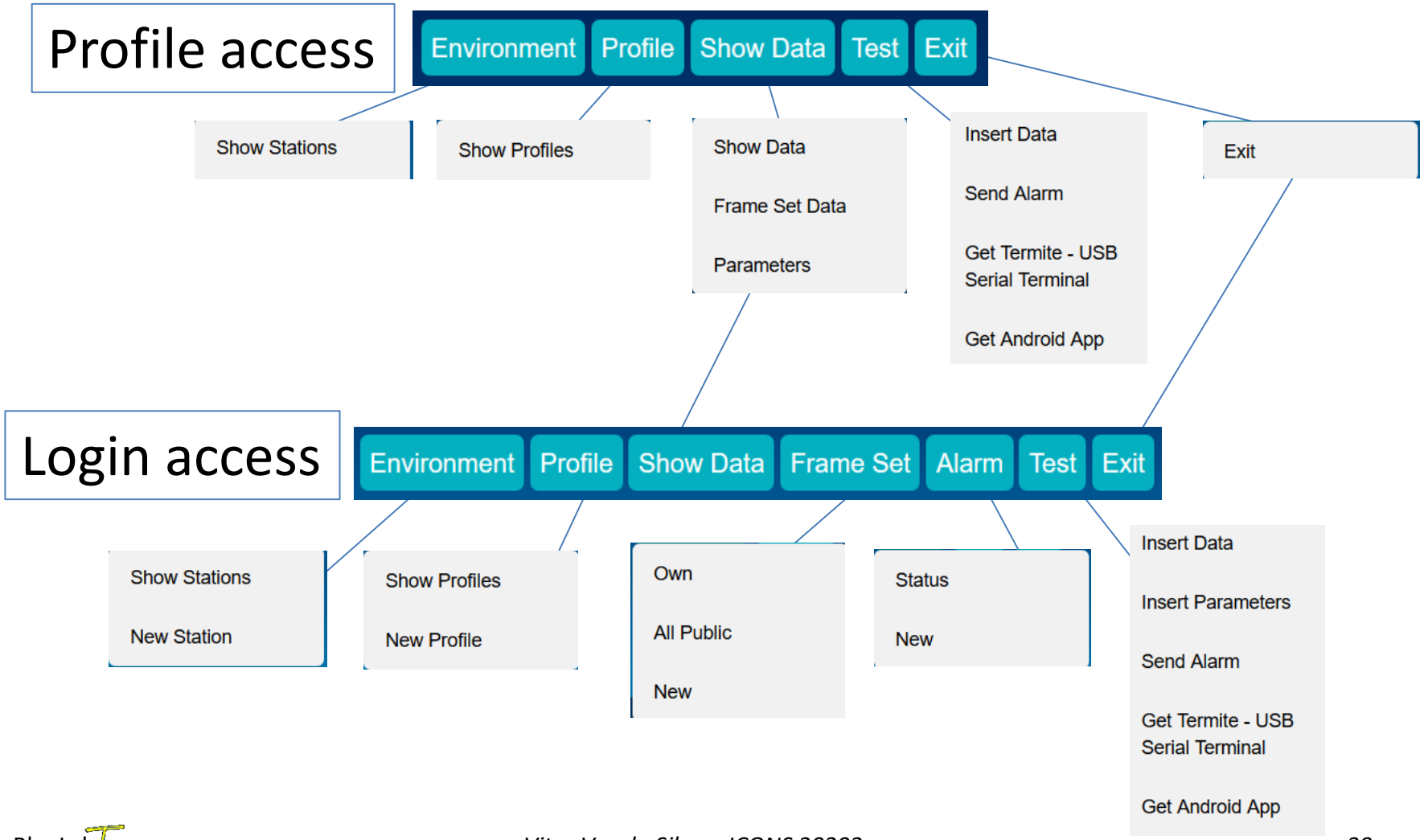
New Profile

2nd Step

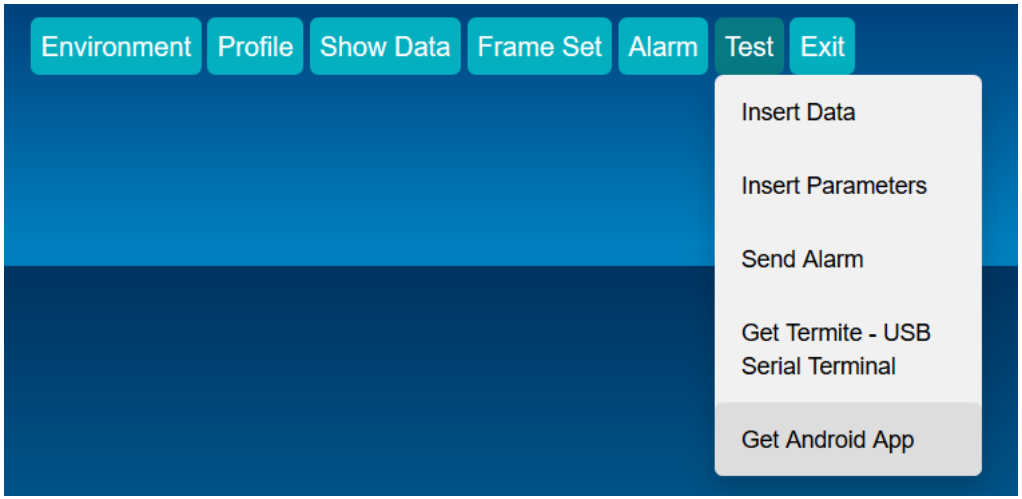


BlueLab IoT

Menu differences



BlueLab IoT

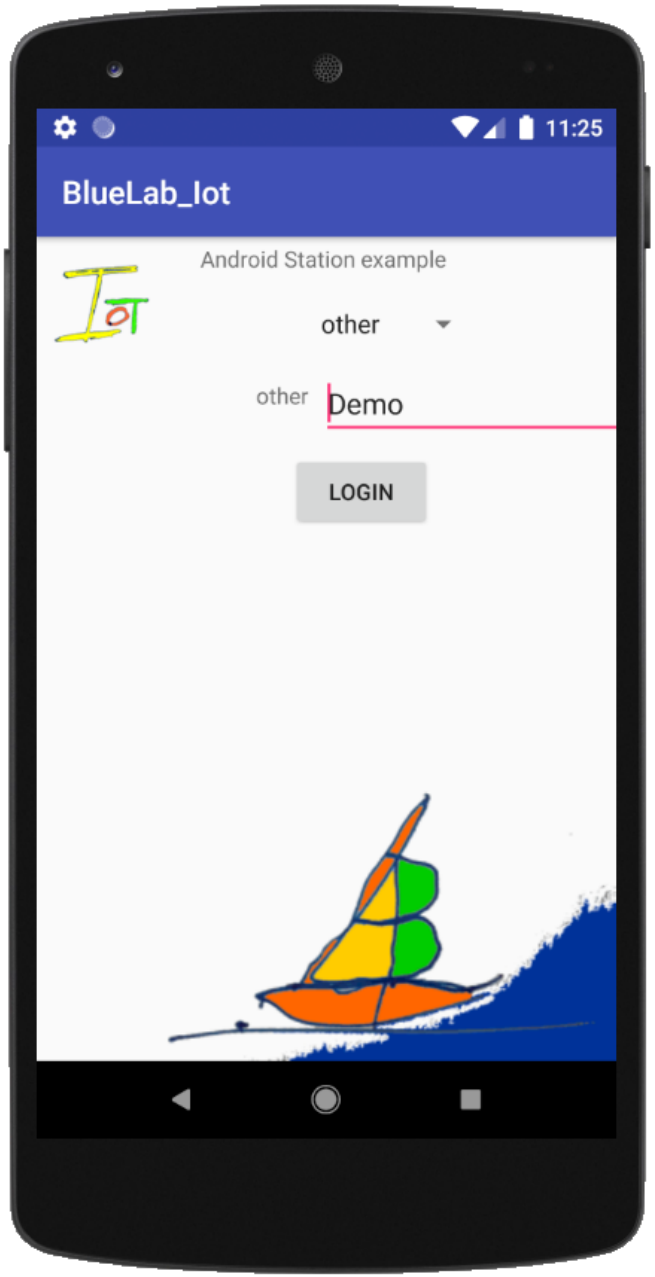


Get Android App

3rd Step

The *Android App* is a Station – it uses the light sensor, and the GPS sensor

Demo – is a (*Other*) Profile



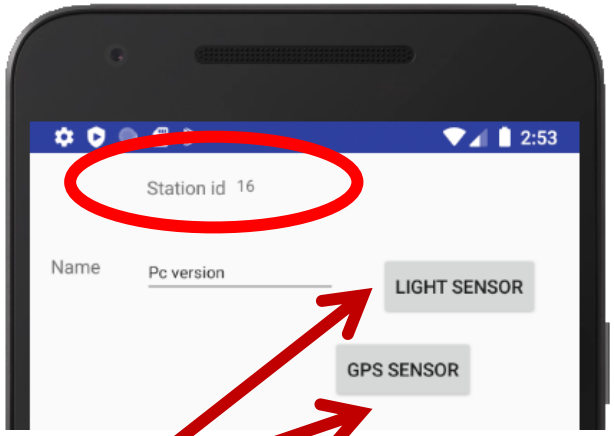
1 - First use *other* Demo

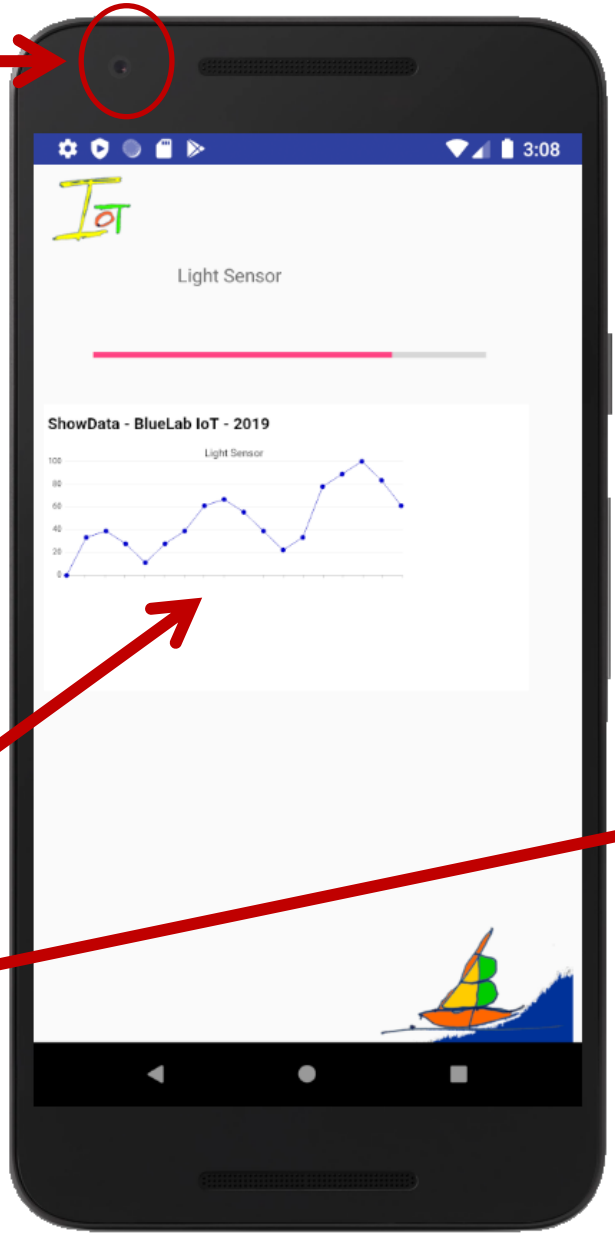
2 - Then use the *other* Profile you created

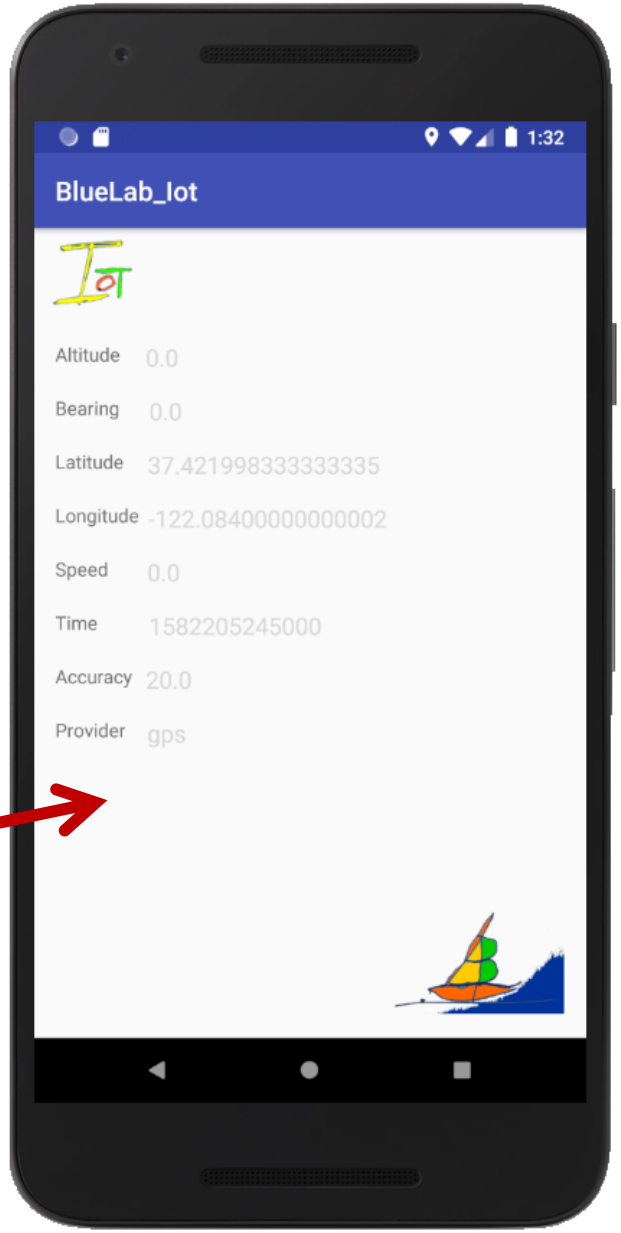
Station Name

Creates/activates the station in the system
(if the station exists the button will not show up)

Light sensor



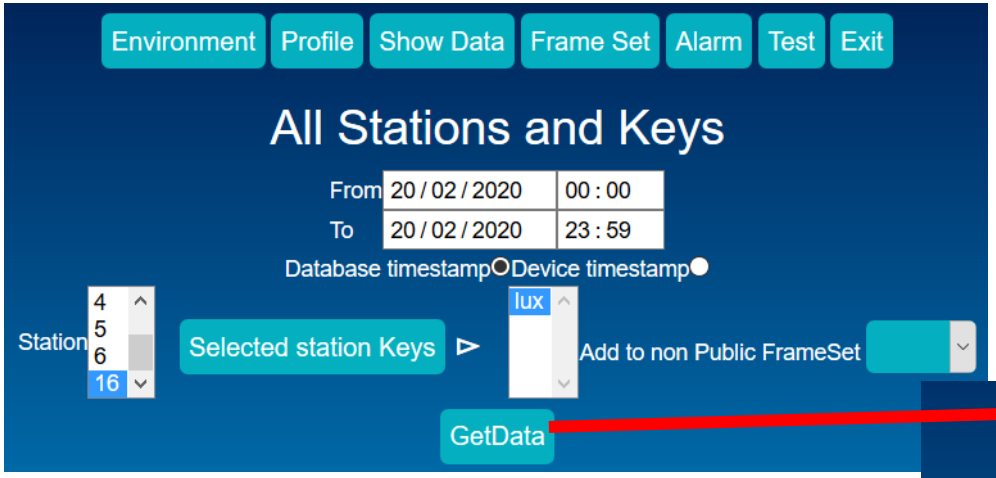




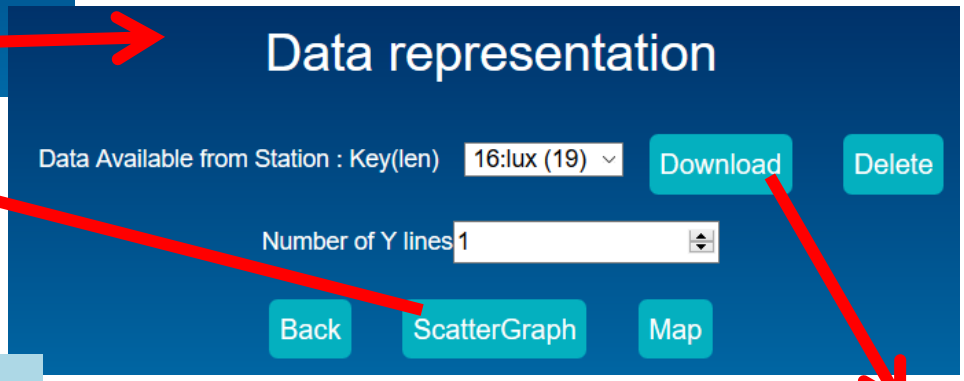
Two exclusive sensors

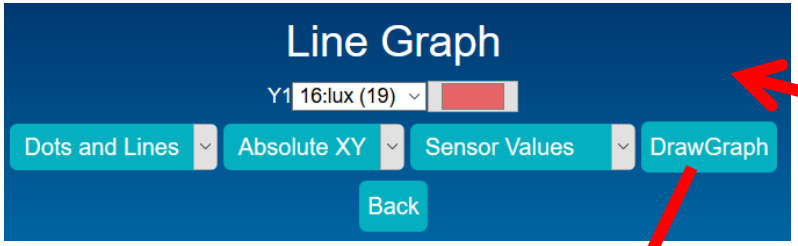
The graph is refreshed every 10 samples

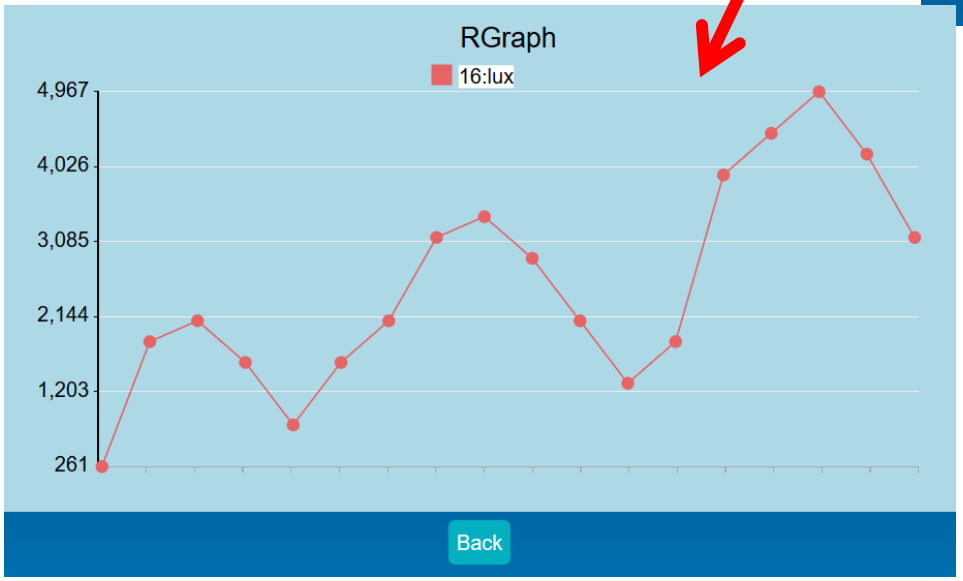
BlueLab application must have location permit and GPS on to work as a GPS sensor



Accessing the light sensor data through the user interface







Station;	Sequence;	DB Timestamp;	Timestamp;	key;	value
16;	1;	2020-02-20 14:55:44.053171;	2020-02-20 14:55:42.466442;	lux;	261.4
16;	2;	2020-02-20 14:55:44.256685;	2020-02-20 14:55:42.667047;	lux;	1830.1
16;	3;	2020-02-20 14:55:44.655435;	2020-02-20 14:55:43.069886;	lux;	2091.5
16;	4;	2020-02-20 14:55:44.926323;	2020-02-20 14:55:43.270425;	lux;	1568.6
16;	5;	2020-02-20 14:55:45.851666;	2020-02-20 14:55:44.27353;	lux;	784.3
16;	6;	2020-02-20 14:55:46.863138;	2020-02-20 14:55:45.275244;	lux;	1568.6
16;	7;	2020-02-20 14:55:47.055202;	2020-02-20 14:55:45.476049;	lux;	2091.5
16;	8;	2020-02-20 14:55:47.264;	2020-02-20 14:55:45.677337;	lux;	3137.3
16;	9;	2020-02-20 14:55:47.495075;	2020-02-20 14:55:45.875548;	lux;	3398.7
16;	10;	2020-02-20 14:55:48.468083;	2020-02-20 14:55:46.878141;	lux;	2875.8
16;	11;	2020-02-20 14:55:48.671224;	2020-02-20 14:55:47.079157;	lux;	2091.5
16;	12;	2020-02-20 14:55:48.87234;	2020-02-20 14:55:47.279901;	lux;	1307.2
16;	13;	2020-02-20 14:55:50.082462;	2020-02-20 14:55:48.083158;	lux;	1830.1
16;	14;	2020-02-20 14:55:50.267729;	2020-02-20 14:55:48.688466;	lux;	3921.6
16;	15;	2020-02-20 14:55:50.473015;	2020-02-20 14:55:48.883658;	lux;	4444.4
16;	16;	2020-02-20 14:55:50.665643;	2020-02-20 14:55:49.080783;	lux;	4967.3
16;	17;	2020-02-20 14:55:52.063593;	2020-02-20 14:55:50.481718;	lux;	4183.0
16;	18;	2020-02-20 14:55:52.258649;	2020-02-20 14:55:50.683075;	lux;	3137.3
16;	19;	2020-02-20 15:08:13.982742;	2020-02-20 15:08:12.381435;	lux;	3137.3

CSV format

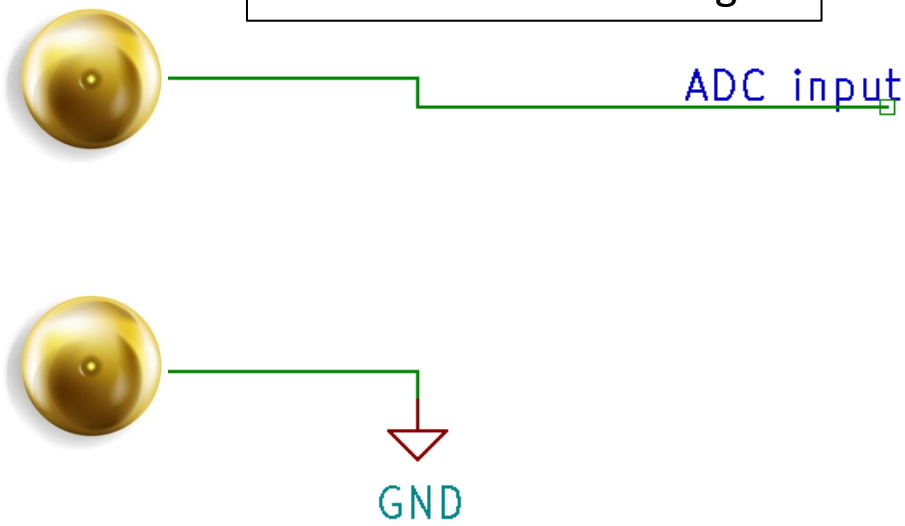
BlueLab IoT

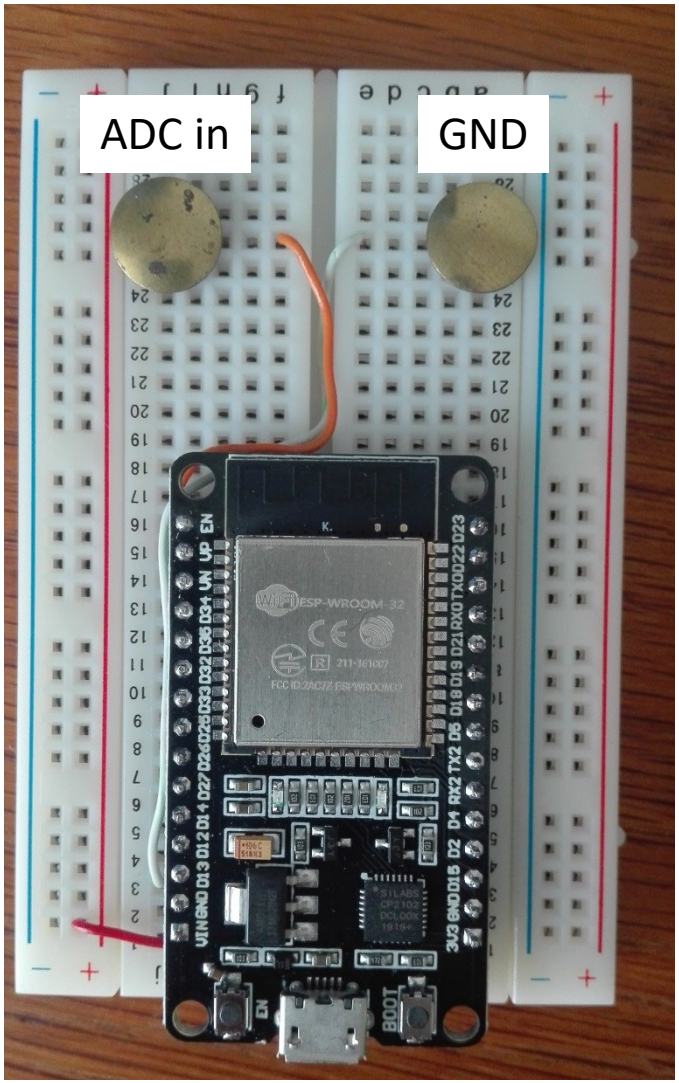
Connect USB cable to Station
Use terminal Termite *(or equivalent)*

Current Configuration

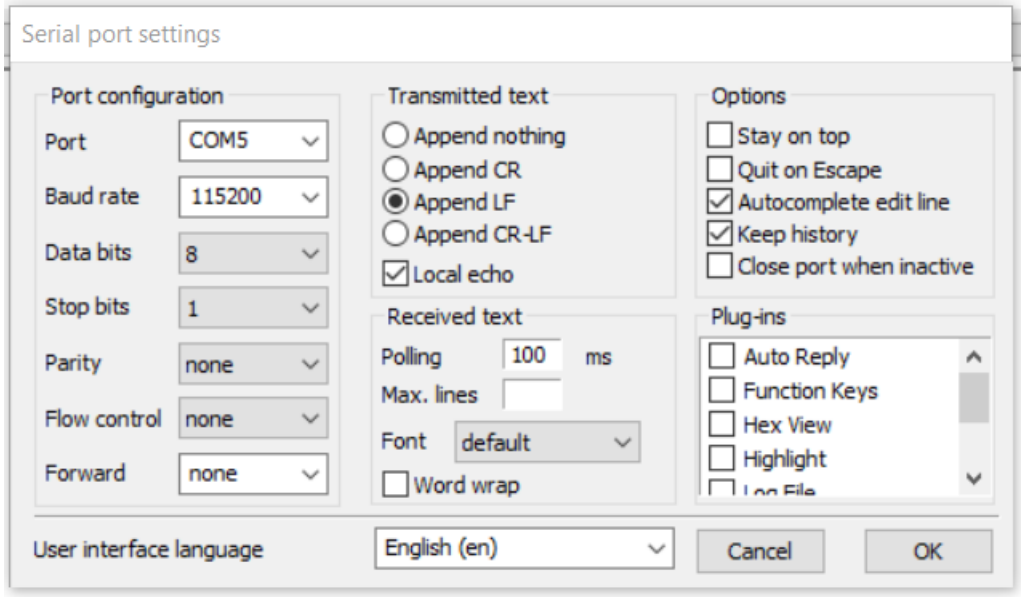
```
> s
Module Satus
 UserContactType: #O#
 UserContact: #ICONS2020#
 UserPassword: ##
 SSID: #my_ssid#
 SSID_PWD: #my_ssid_password#
 Module Name: #ESP32-6#
```

Touch contacts with fingers





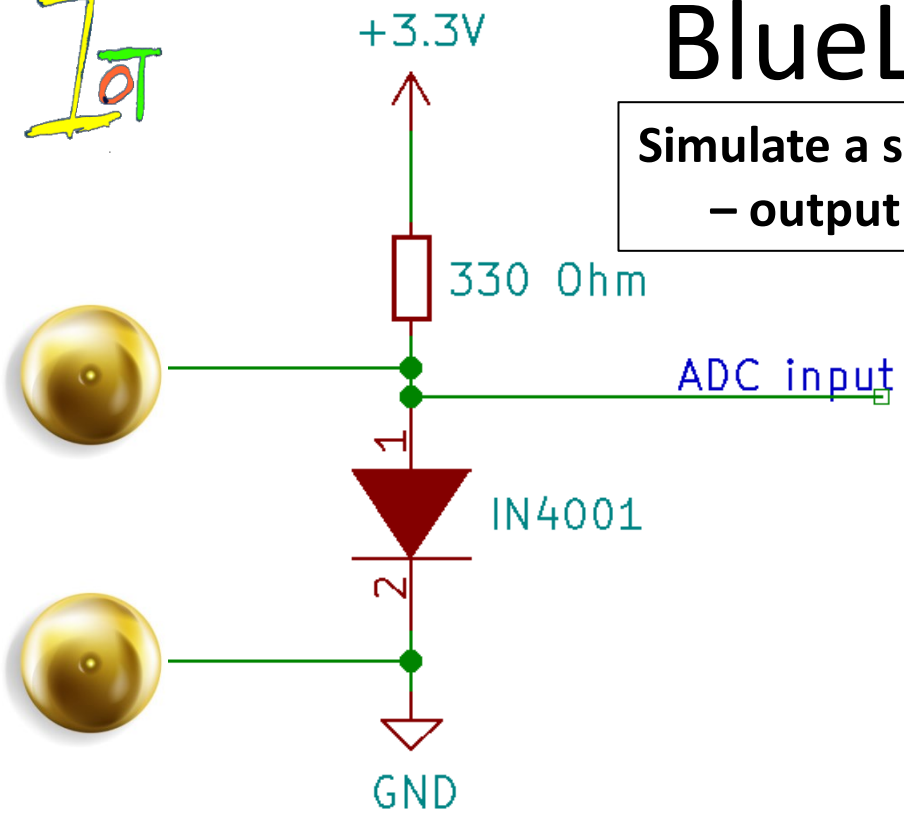
Termite port configuration

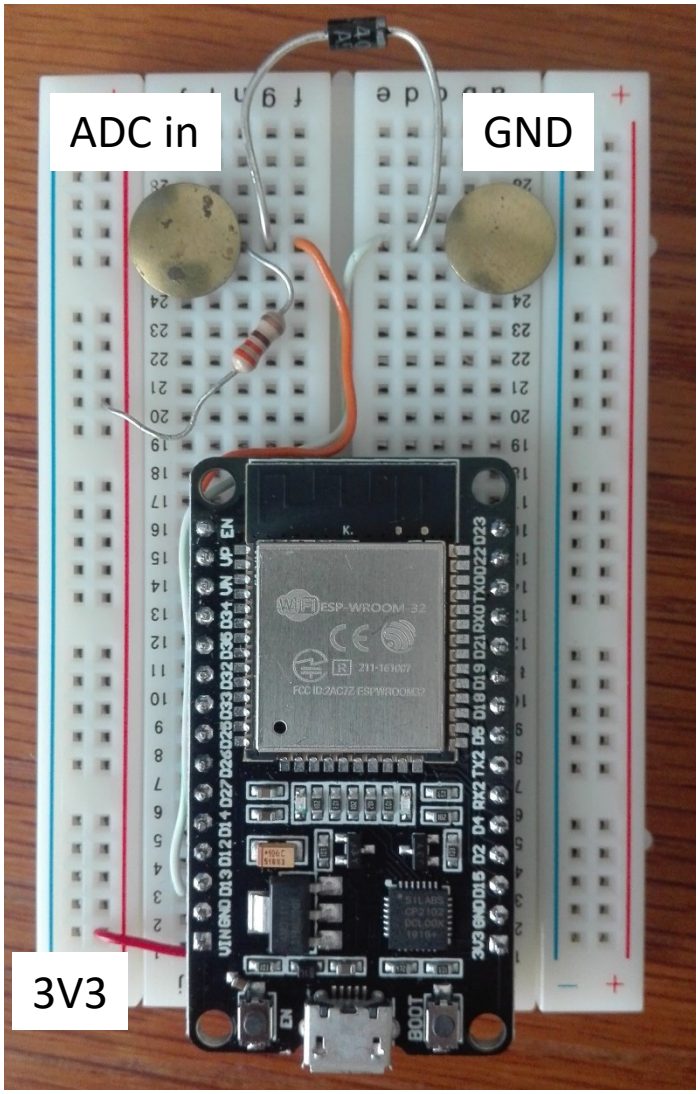


To access text written by the station program on the serial interface

BlueLab IoT

Simulate a sensor
– output = 0.68 V





ESP8266 dev kit – (ADC – 10 bits -> 0-1023)

ESP32 dev kit – (ADC – 12 bits -> 0-4095)

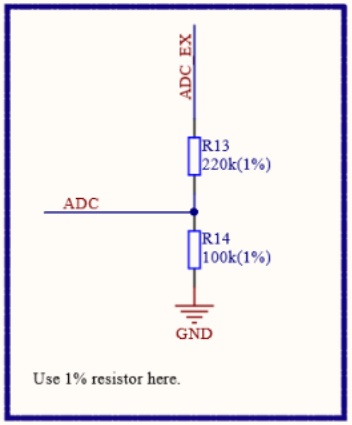
1 - Add data parameters to correct sensor values

2 - Create frameset and share to public

3 - All public data graphs of the sensor
should show values close to 0.68V

BlueLab IoT

Voltage divider of the development kit ESP8266



ESP8266 – voltage Ref 1V

- Resolution 10 bits
- Voltage divider 0.3125 attenuation of 3.2
- Value = (3.2 * 1) * adc_in /1023 = 3.13e-3 * adc_in

ESP32 – voltage ref 1.1V

- Default Resolution 12 bits
- Default attenuation of 11dB (3.548)
- Value = (3.548 * 1.1) * adc_in / 4095 = 0.967e-3 * adc_in

BlueLab IoT

Parameters

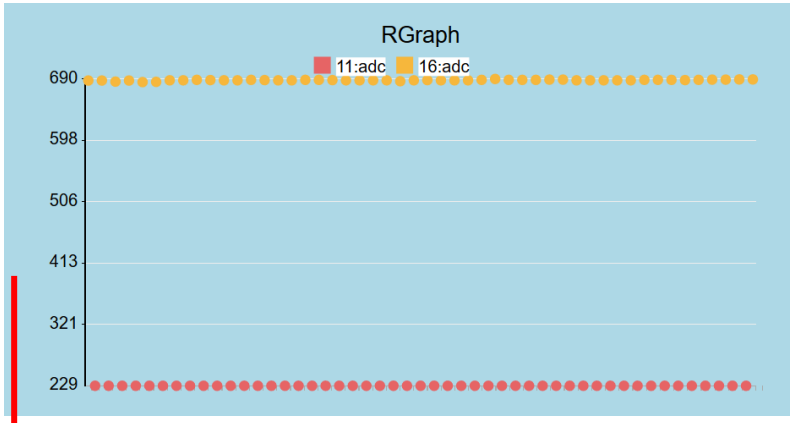
Delete	Station:Parameter	Value	t_stamp	db_t_stamp
Del	11:#adc_1	3.13e-3	2020-02-21 13:48:39	2020-02-21 13:49:09
Del	11:#adc_U	V	2020-02-21 14:45:58	2020-02-21 14:46:12
Del	16:#adc_1	0.967e-3	2020-02-21 13:49:10	2020-02-21 13:49:38
Del	16:#adc_U	V	2020-02-21 14:46:12	2020-02-21 14:46:17

Insert Parameter

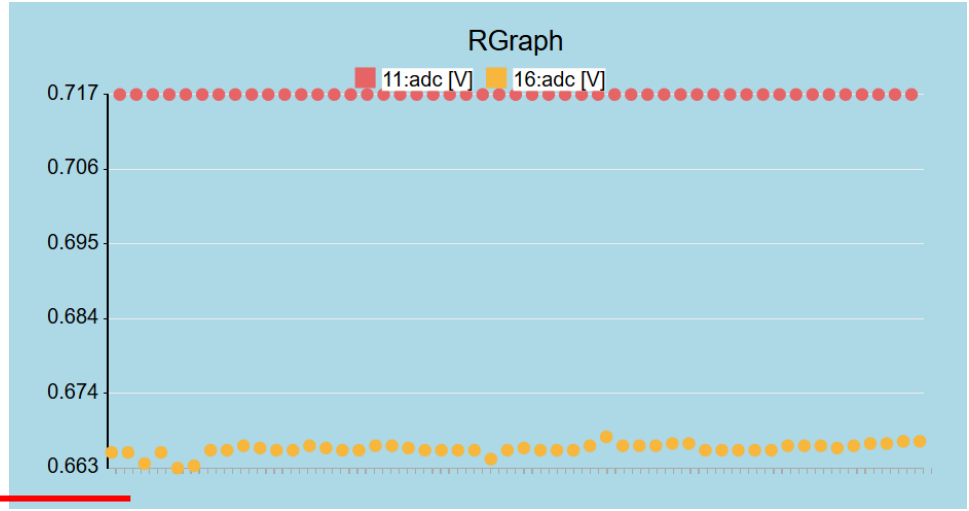
Station id: 16
Seq number: 0
timeStamp: Fri Feb 21 2020 14:43:25 GMT+0000 (Western European Stanc
key: #adc_1
value: 0.967e-3
Insert

ESP8266

ESP32



ADC reading



Corrected for voltage

BlueLab IoT - FrameSets

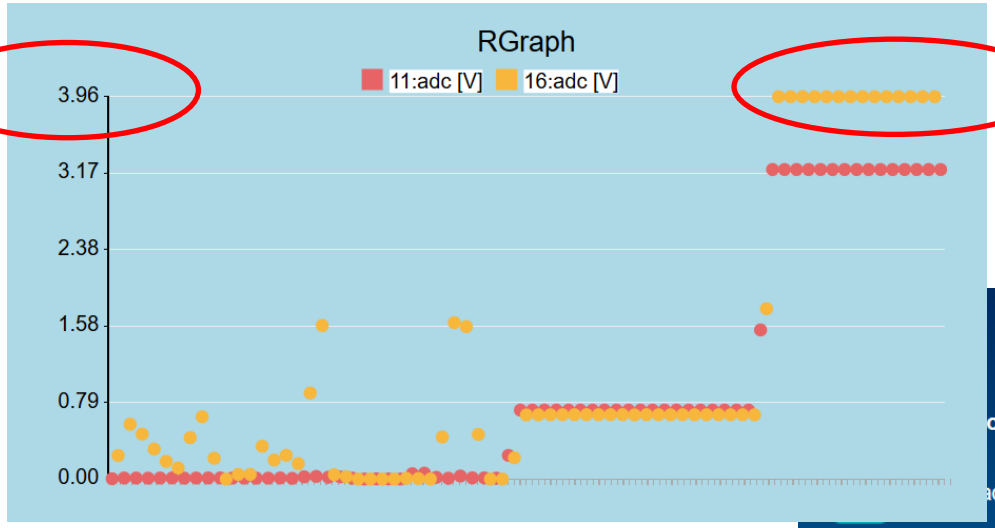
ESP32

When VDD_A is 3.3 V:

- 0 dB attenuation gives full-scale voltage 1.1 V (100 and 950 mV)
- 2.5 dB attenuation gives full-scale voltage 1.5 V (100 and 1250 mV)
- 6 dB attenuation gives full-scale voltage 2.2 V (150 to 1750 mV)
- 11 dB attenuation gives full-scale voltage 3.9 V (150 to 2450 mV)

At 11 dB attenuation the maximum voltage is limited by VDD_A, not the full scale voltage.

(see literature; readings are not linear above 2450mV)

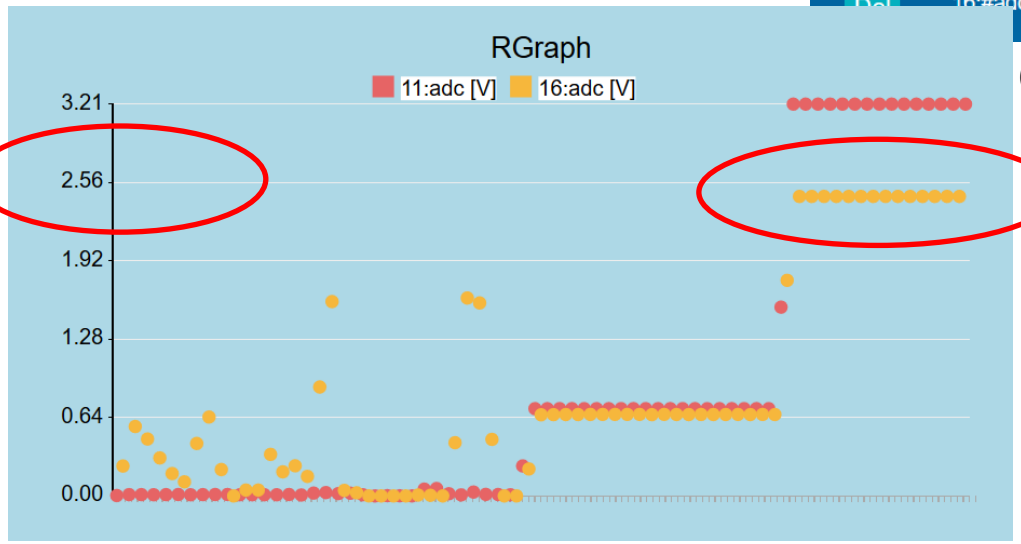


Does not make sense as the maximum is 2.45 V

Parameters

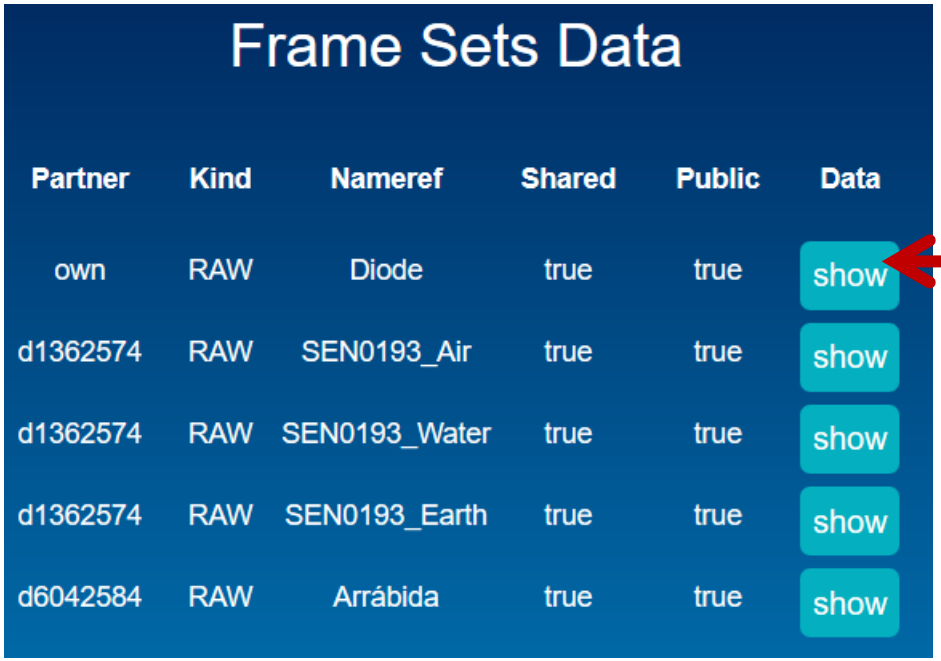
	on:Parameter	Value	t_stamp	db_t_stamp
	adc_1	3.13e-3	2020-02-21 13:48:39	2020-02-21 13:49:09
Del	11:#adc_U	V	2020-02-21 14:45:58	2020-02-21 14:46:12
Del	16:#adc_1	0.967e-3	2020-02-21 13:49:10	2020-02-21 13:49:38
Del	16:#adc_>2450e-3	2.45	2020-02-22 11:20:33	2020-02-22 11:20:39
Del	16:#adc_U	V	2020-02-21 14:46:12	2020-02-21 14:46:17

Correct



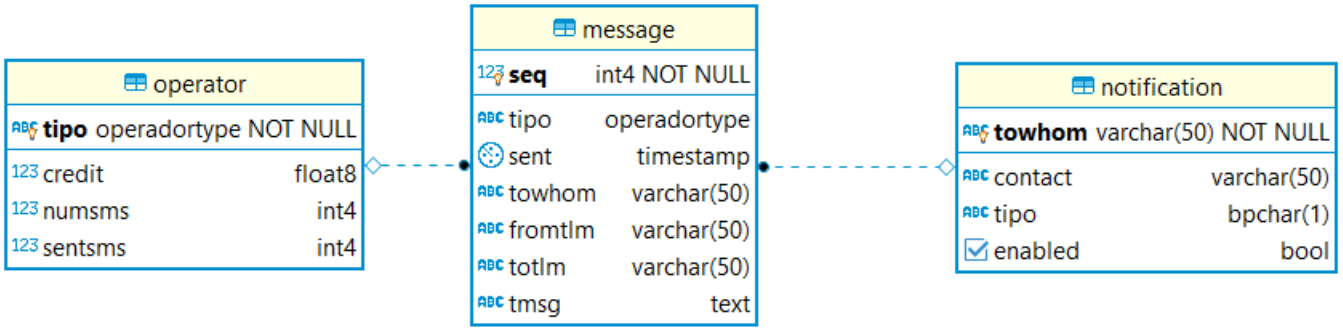
Key	Value
0…n	A_n of x^n
U	Units
O	Offset
S	Scale
>value	Max value
<value	Min value

BlueLab IoT – Shared FrameSets



Similar to
Previous example

BlueLab IoT



Alarm service structure

A station may send alarms that are sent by e-mail or phone to the user

There is a minimum time between consecutive alarms

The user can enable or disable the sending of alarms

ESP32

Tensilica Xtensa 32-bit LX6 microprocessor

2 cores (ESP32-S0WD only one) 240 MHz

Ultra low power co-processor**:** Allows ADC conversions, computation while in deep sleep.

Wi-Fi: 802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s)

Bluetooth: v4.2 BR/EDR and Bluetooth Low Energy (BLE)

Internal memory: ROM: 448 KB, SRAM: 520 KB, RTC fast SRAM: 8 KB, slow : 8 KB, eFuse: 1Kbit

External flash & SRAM: up to 4 x 16MB

Rich peripheral interface with DMA, capacitive touch, ADCs, DACs, I^2C , CAN 2.0, SPI, I^2S, RMII, PWM

IEEE 802.11 standard security

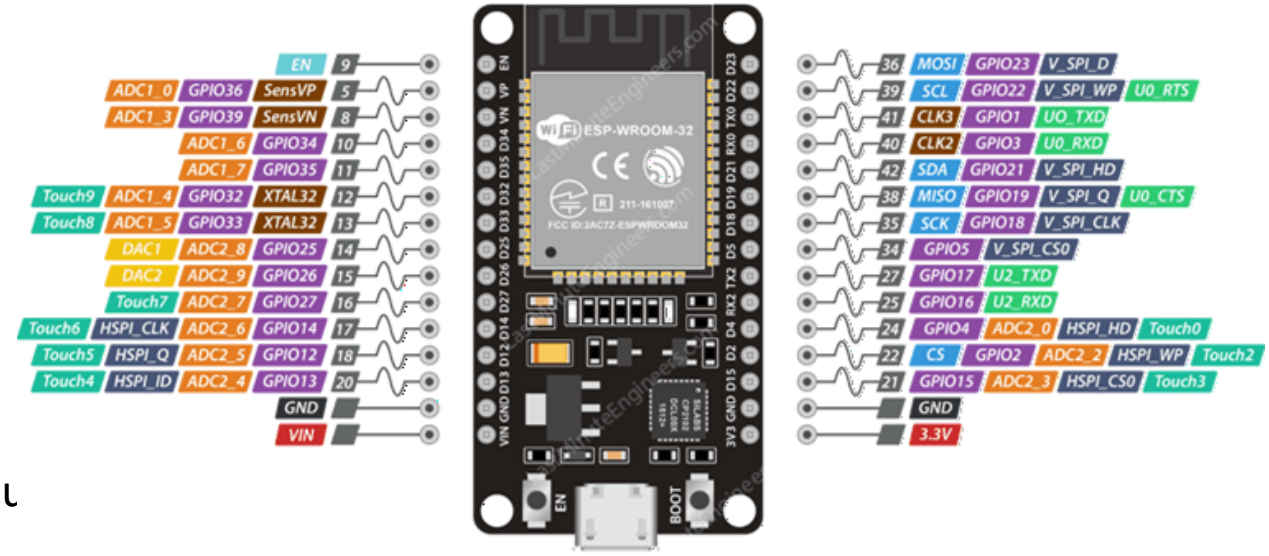
WFA, WPA/WPA2 and WAPI

Secure boot

Flash encryption

1024-bit OTP, up to 768-bit for cu

Cryptographic hardware acceleration: AES, SHA-2, RSA, ECC, RNG



Built-in low power 32-bit MCU @ 80MHz

512kB Flash Memory

Power Supply: +3.3V only

Current Consumption: 100mA

I/O Voltage: 3.6V (max)

I/O source current: 12mA (max)

Supports Deep sleep (<10uA)

UART TX / UART RX 1200-115200

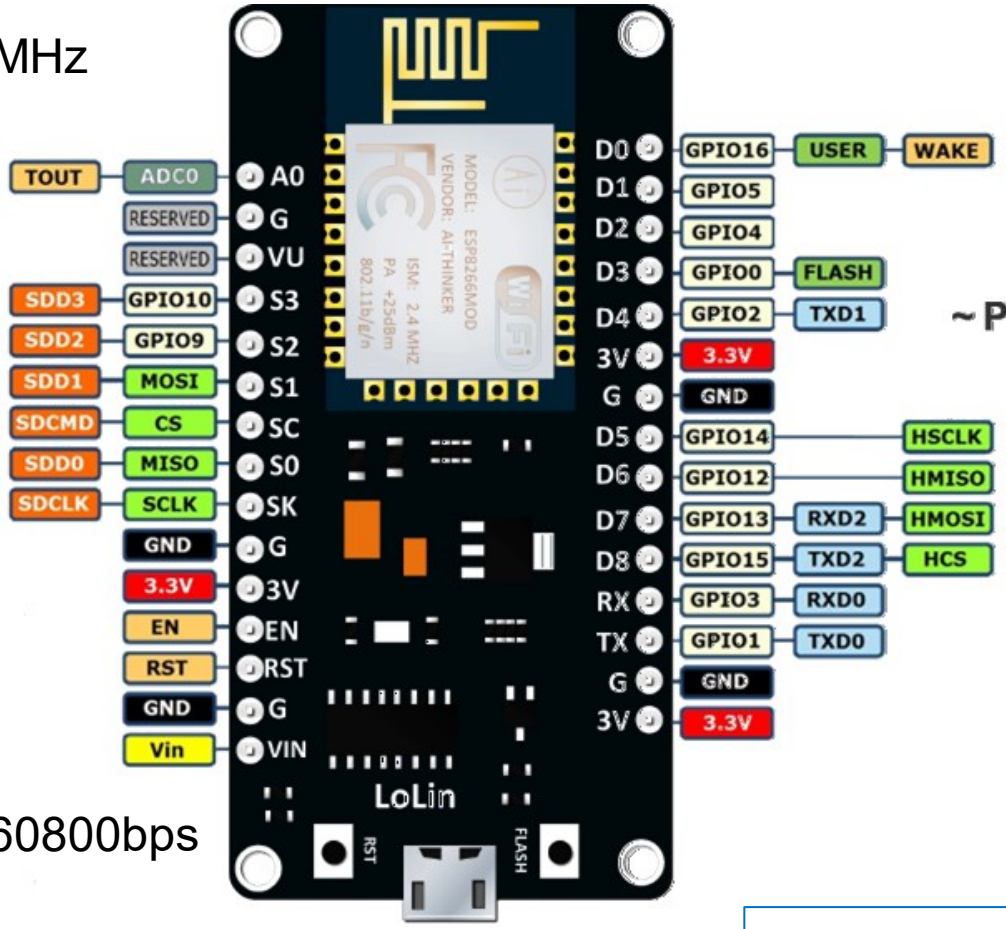
802.11 b / g / n wireless standards;

Serial WiFi transmission rate: 110-460800bps

WiFi operation current:

continuous transmission operation: ≈70mA (200mA MAX)

idle mode: <200uA;



ESP8266

Can be used as Station or Access Point or both combined

TCP / IP protocol stack, One socket - Standard TCP / UDP Server and Client;

Future

- Cron job for data maintenance, clean delete, compression of old unshared data
- Improve data parameter for sensor calibration history
- Add data processing for level P1 (processed data)
- Allow sharing to restricted group of users
- Allow user to use its own database
- Use other Hardware ex:Thingy:52
- Improve Site
- Different graph types
- Develop Distributor
- Clean code
- Add BlueLab IoT to Arduino Library Manager

References

1. https://randomnerdtutorials.com/esp8266-pinout-reference-gpios/
2. http://esp8266.net/
3. https://randomnerdtutorials.com/esp32-pinout-reference-gpios/
4. https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/
5. http://esp32.net/
6. https://asksensors.com/doc/control-esp32-https.html
7. https://github.com/asksensors
8. https://www.ubeac.io
9. https://hook.ubeac.io/
10. https://temboo.com
11. https://thinger.io/
12. https://docs.thinger.io/devices/arduino
13. https://github.com/NordicSemiconductor/Nordic-Thingy52-FW
14. https://www.nordicsemi.com/Software-and-tools/Prototyping-platforms/Nordic-Thingy-52
15. https://techbeacon.com/app-dev-testing/67-open-source-tools-resources-iot
16. Radi, R., Murtiningrum, M., Ngadisih, N., Muzdrikah, F. S., Nuha, M. S., & Rizqi, F. A. (2018). Calibration of Capacitive Soil Moisture Sensor (SKU:SEN0193). In *2018 4th International Conference on Science and Technology (ICST)* (pp. 1–6). IEEE. https://doi.org/10.1109/ICSTC.2018.8528624
17. https://www.w3.org/WoT/IG/wiki/images/9/9a/10N0536_CD_text_of_ISO_IEC_30141.pdf
18. https://www.postscapes.com/internet-of-things-protocols/

BlueLab IoT