

IoT - ESP8266

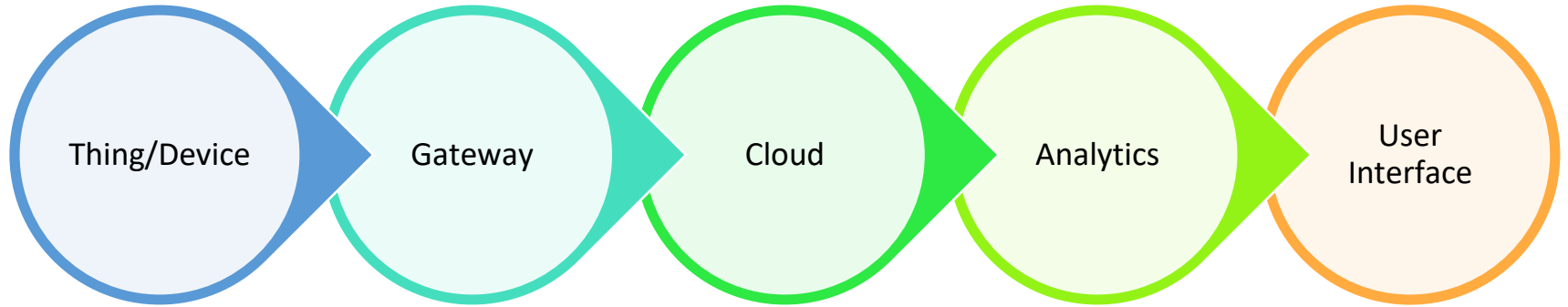
What is **IoT**?



Is it a
Device / System ?

In the simplest terms, the Internet of Things (IoT) is how we describe the digitally connected universe of everyday physical devices. These devices are embedded with internet connectivity, sensors and other hardware that allows to monitor physical factors

Components in IoT



Thing/Device



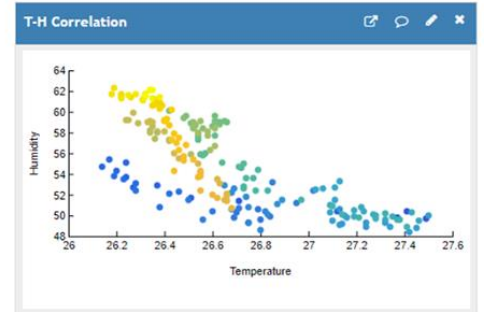
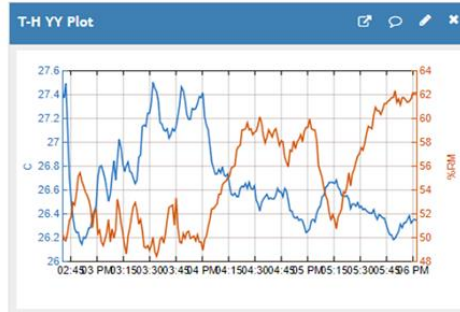
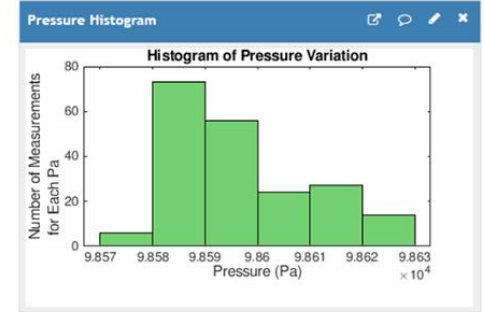
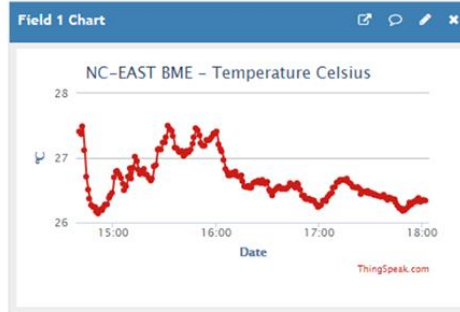
GateWay/Connectivity



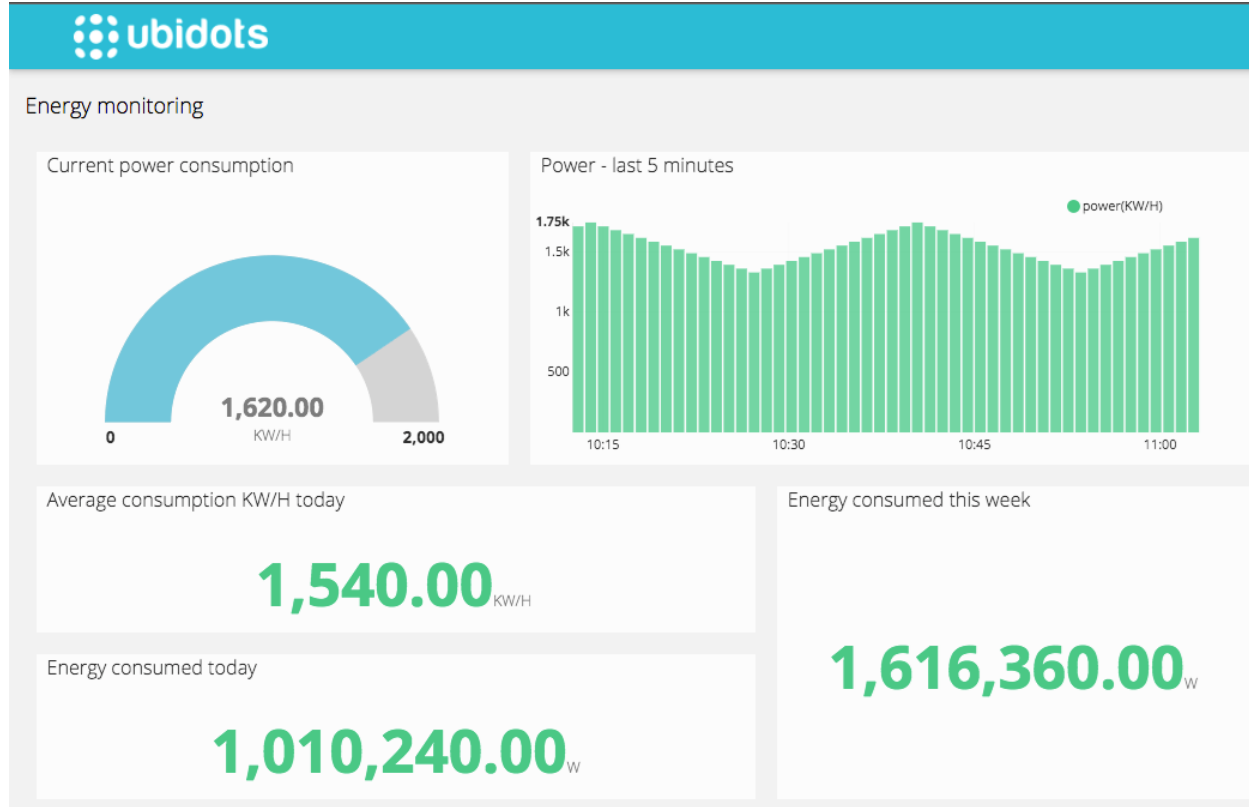
Cloud



Analytics



User Interface



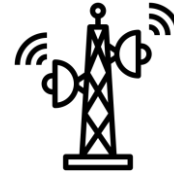
Different Connectivities



WiFi - Wireless Fidelity



Bluetooth



Cellular

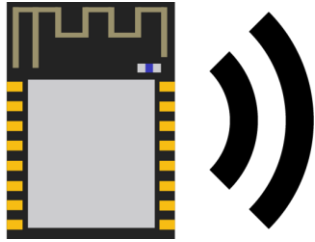


LPWAN
(Low Power Wide Area Network)



Satellite Coms - Iridium sat,
Lacuna Space, etc.

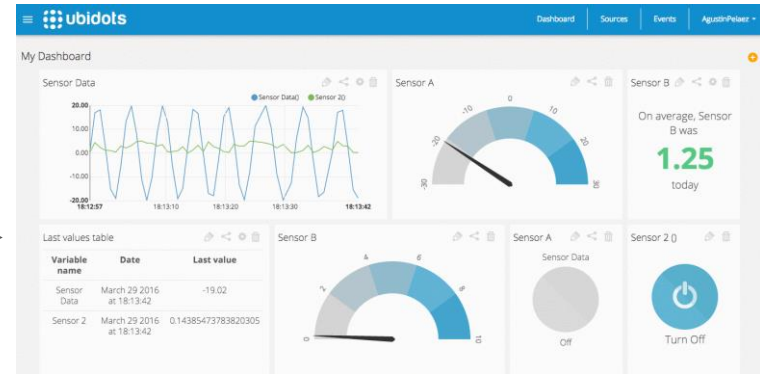
We'll be learning



How to use an
ESP8266 with Arduino Platform,
Connecting it with WiFi, Data
Acquisition & Sending data to
cloud.

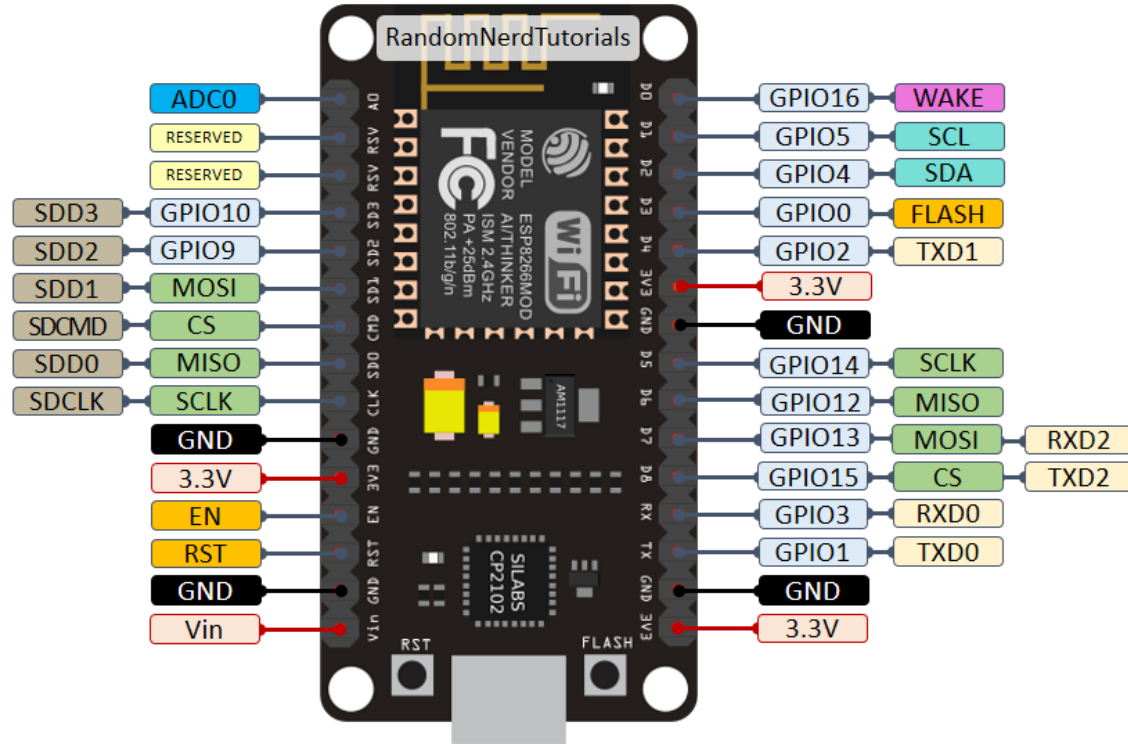


Creating account, Device
management

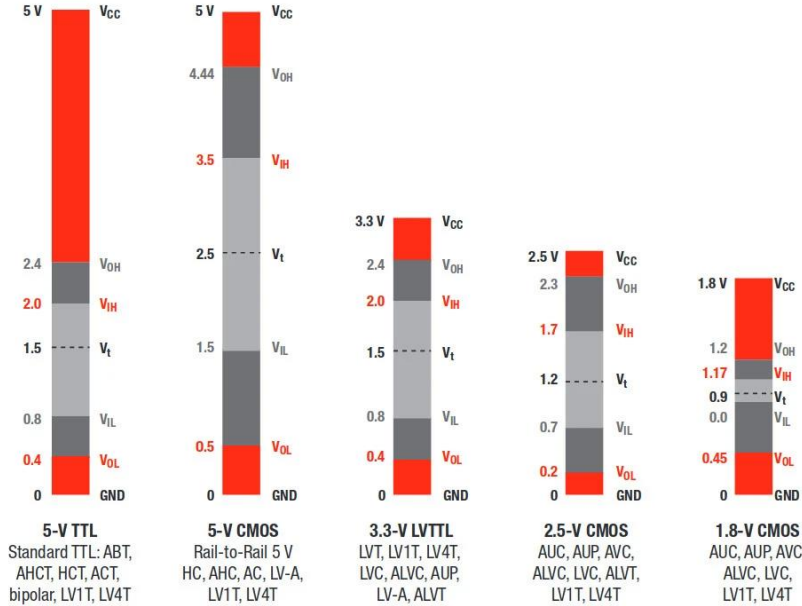


Creating dashboard, adding
widgets, customizing widgets.

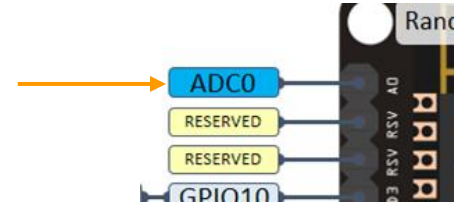
ESP8266



Things to keep in mind



Logic Level is 3.3v



Only one Analog Pin

Things to keep in mind.....

Strapping Pins:

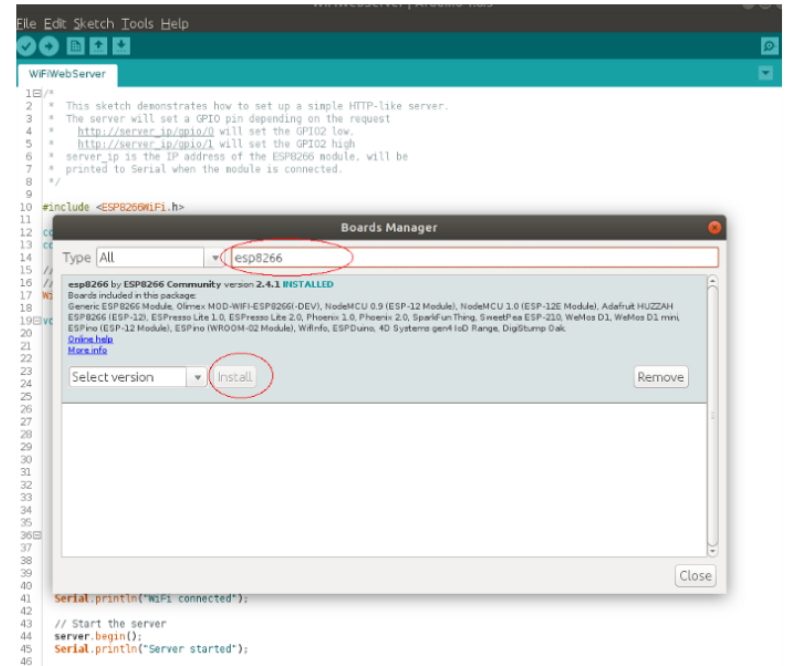
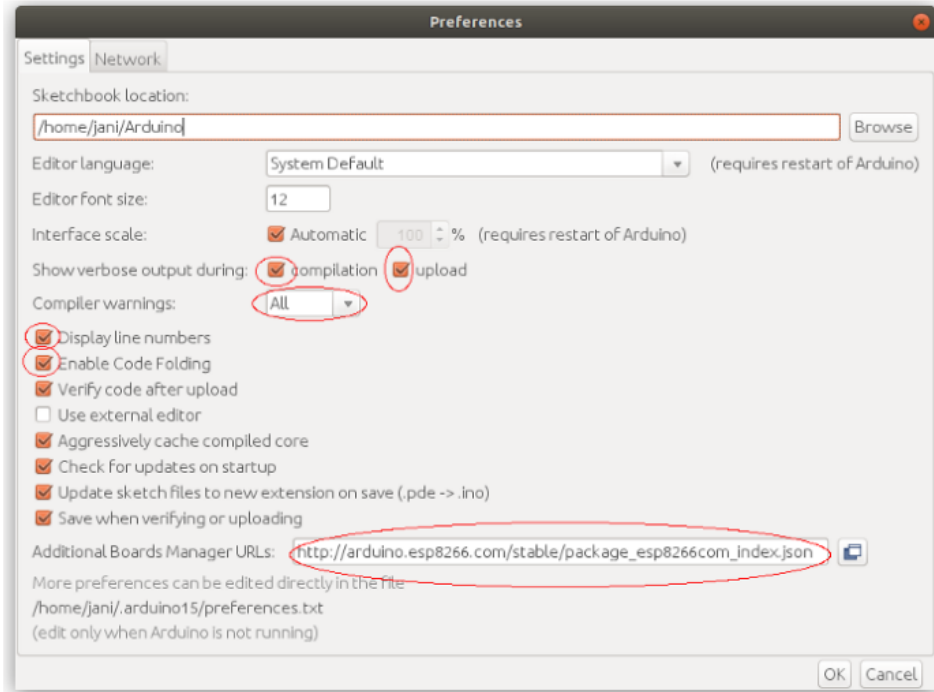
GPI015	GPI00	GPI02	Mode
0V	0V	3.3V	Uart Bootloader
0V	3.3V	3.3V	Boot sketch (SPI flash)
3.3V	x	x	SDIO mode (not used for Arduino)

GPIO	Function	State	Restrictions
0	Boot mode select	3.3V	No Hi-Z
1	TX0	-	Not usable during Serial transmission
2	Boot mode select TX1	3.3V (boot only)	Don't connect to ground at boot time Sends debug data at boot time
3	RX0	-	Not usable during Serial transmission
4	SDA (I ² C)	-	-
5	SCL (I ² C)	-	-
6 - 11	Flash connection	x	Not usable, and not broken out
12	MISO (SPI)	-	-
13	MOSI (SPI)	-	-
14	SCK (SPI)	-	-
15	SS (SPI)	0V	Pull-up resistor not usable
16	Wake up from sleep	-	No pull-up resistor, but pull-down instead Should be connected to RST to wake up

Let's get started.

| *Enough talking, let's play!*

Installing Board - Boards Manager



Installation guide link shared.

Let's Blink a LED first

Use the same code used in UNO. The Built-In or External LED can be used.



Creating an account in Ubidots

(Please create a STEM account) <https://ubidots.com/stem/>

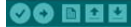
Understanding limits of STEM account

- Devices: Up to **10 devices**. First **3** devices free, then **\$4.5** / device / month
- Variables: Up to **10 variables** per device.
- Plugins: Up to **1 plugin**, with **10,000** plugin executions per month.
- Data Ingestion: **4,000 dots** per day across all of your devices.
- Data Extraction: **500,000 dots** per day across all of your account.
- Data Rate: **1 request per second**, across all of your devices.
- Data Retention: **1 month**.
- Dashboards: Up to **3** dashboards, with up to **10** widgets each.
- SMS & Voice calls: **10** SMS and **1** Voice call for free per month (US & Canada only), then [standard rates apply](#).
- Emails: 100 Emails for free per month, then [standard rates apply](#).
- Uptime: No SLA - shared server resources across all STEM users. The speed and reliability is based on total platform requests at any given second.
- Support: [Community-based](#), or self-service using our [Help Center](#) and [API Docs](#).

Sending data to Ubidots

1_Sending_Data_to_Ubidots | Arduino 1.8.16

File Edit Sketch Tools Help



1_Sending_Data_to_Ubidots \$

```
1 #include "Ubidots.h"
2
3
4 const char* UBIDOTS_TOKEN = ""; // Put here your Ubidots TOKEN
5 const char* WIFI_SSID = ""; // Put here your Wi-Fi SSID
6 const char* WIFI_PASS = ""; // Put here your Wi-Fi password
7 Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);
8
9 void setup()
10 {
11   Serial.begin(115200);
12   ubidots.wifiConnect(WIFI_SSID, WIFI_PASS);
13   //ubidots.setDebug(true);
14 }
15
16 void loop()
17 {
18   int value1 = random(0, 100);
19   int value2 = random(100, 300);
20
21   ubidots.add("Random_num_1", value1);
22   ubidots.add("Random_num_2", value2);
23
24   bool bufferSent = false;
25   bufferSent = ubidots.send();
26
27   if (bufferSent)
28   {
29     Serial.println("Values sent by the device");
30   }
```

Changing Credentials

```
const char* UBIDOTS_TOKEN = ""; // Put here your Ubidots TOKEN
const char* WIFI_SSID = ""; // Put here your Wi-Fi SSID
const char* WIFI_PASS = ""; // Put here your Wi-Fi password
Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);
```

Upload code once done!

Your serial monitor

COM15

Send

```
17:43:34.635 -> .....WiFi connected
17:43:40.377 -> IP address:
17:43:40.377 -> 192.168.233.127
17:43:40.377 -> -----
17:43:40.377 -> payload:
17:43:40.377 -> {"Random_num_1":{"value":32},"Random_num_2":{"value":197}}
17:43:40.377 -> -----
17:43:40.377 ->
17:43:40.377 -> Sending data...
```

Autoscroll Show timestamp

No line ending Clear output

Take a look at your Ubidots account now.

The screenshot shows the Ubidots interface with the 'Devices' page selected. At the top, there is a dark blue header with the Ubidots logo, navigation menus for 'Devices' and 'Data', and user profile icons. Below the header, the 'Devices' section contains a search bar and a table of devices. The table has columns for 'NAME', 'LAST ACTIVITY', 'CREATED AT', and 'ACTIONS'. The first row is highlighted in light blue, and an orange arrow points from the text 'New device created' to the 'LAST ACTIVITY' cell of this row, which says 'a few seconds ago'. The second row shows a device with a red icon, last active '14 minutes ago', and a trash icon in the actions column. At the bottom of the table, there is a 'ROWS PER PAGE' dropdown set to '30' and navigation arrows.

	NAME	LAST ACTIVITY	CREATED AT ↓	ACTIONS
	ecfab632a50	a few seconds ago	2022-09-14 17:29:28 +05:30	
	3b3f2020740e802538...	14 minutes ago	2022-09-14 17:29:59 +05:30	

These are called variables/bin.

The screenshot displays the Ubidots web interface. At the top, the Ubidots logo is on the left, and navigation menus for 'Devices' and 'Data' are in the center. On the right, there are icons for user profile, notifications, and help. Below the navigation bar, a '← Devices' breadcrumb is on the left, and a 'SET LOCATION' button is on the right. The main content area features a world map background. On the left, a device card for 'ecfab632a50' is shown with a teal header. Below the header, the device details are listed: Description (Change description), API Label (ecfab632a50), ID (6321c47820c23701b1472bbe), Token (masked), and Tags (Add new tag, Last activity: a few seconds ago). To the right of the device card, there are two variable cards. The first variable, 'random_num_1', has a value of 83.00 and a last activity of 'a few seconds ago'. The second variable, 'random_num_2', has a value of 247.00 and a last activity of 'a few seconds ago'. Below these cards is a 'VARIABLES PER PAGE' selector set to 30. On the far right, there is a dashed box labeled 'Add Variable' with a plus sign icon.

Creating a dashboard

The screenshot shows the Ubidots web interface. At the top left is the Ubidots logo. The top navigation bar contains 'Devices -' and 'Data -' menus. The 'Data' menu is open, showing 'Dashboards' and 'Events' options. An orange arrow points from the 'Dashboards' option down to the main content area. The main content area displays the message 'No Dashboards created yet' and 'Create Dashboards to visualize your data in realtime', with a blue 'Add new Dashboard' button below it. The top right corner features a notification bell, a help icon, and a user profile icon.

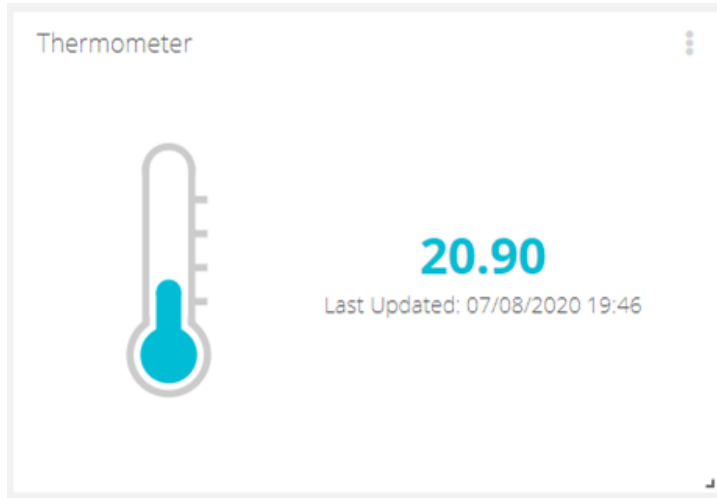
Name your dashboard

The screenshot shows the Ubidots interface with a modal dialog titled "Add new Dashboard". The dialog has two tabs: "SETTINGS" and "APPEARANCE". The "SETTINGS" tab is selected and contains the following fields:

- Name:** A text input field containing "Random Data".
- Default time range:** A dropdown menu showing "Last 24 hours" with a calendar icon.
- Dynamic Dashboard:** A dropdown menu showing "Static" with a subtext "Update widgets based on selected Device".
- Width:** A dropdown menu showing "Auto".
- Date format:** A dropdown menu showing "09/14/2022 18:05" with a list of other date formats below it.

The background shows a greyed-out dashboard area with the text "No Dashboards created yet" and a button labeled "Add new Dashboard". An orange arrow points from this button to the "Name" field in the dialog.

You can explore rest of the options.



Lets measure Temperature

(Please create a STEM account) <https://ubidots.com/stem/>

