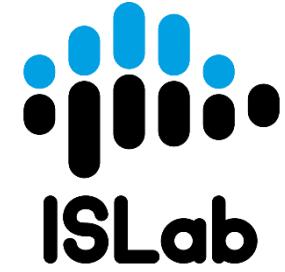


University of Minho
School of Engineering



Sensorization

Autonomous Systems

Perfil Sistemas Inteligentes @ MEI/MiEI 1º/4º - 2º semestre
Bruno Fernandes, Cesar Analide

19/04/2021

Ambient Intelligence (Aml)

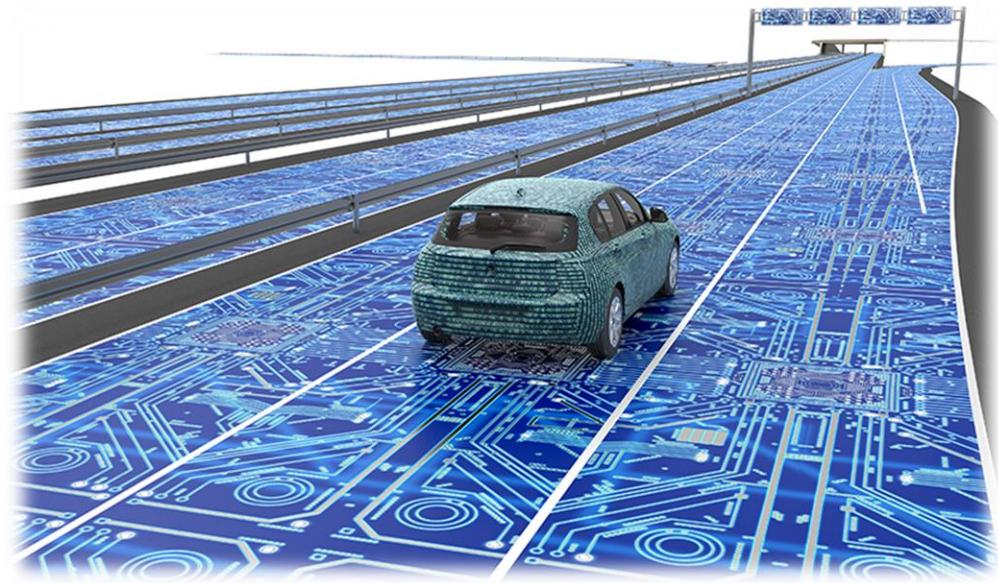
2

CONCEPTS

Sensorization

Hands On

Ambient Intelligence refers to environments that are **sensitive** and **responsive** to the **presence of people** in a non-intrusive manner! As devices grow smaller, connected and more integrated with the environment, the technology disappears into our surroundings!



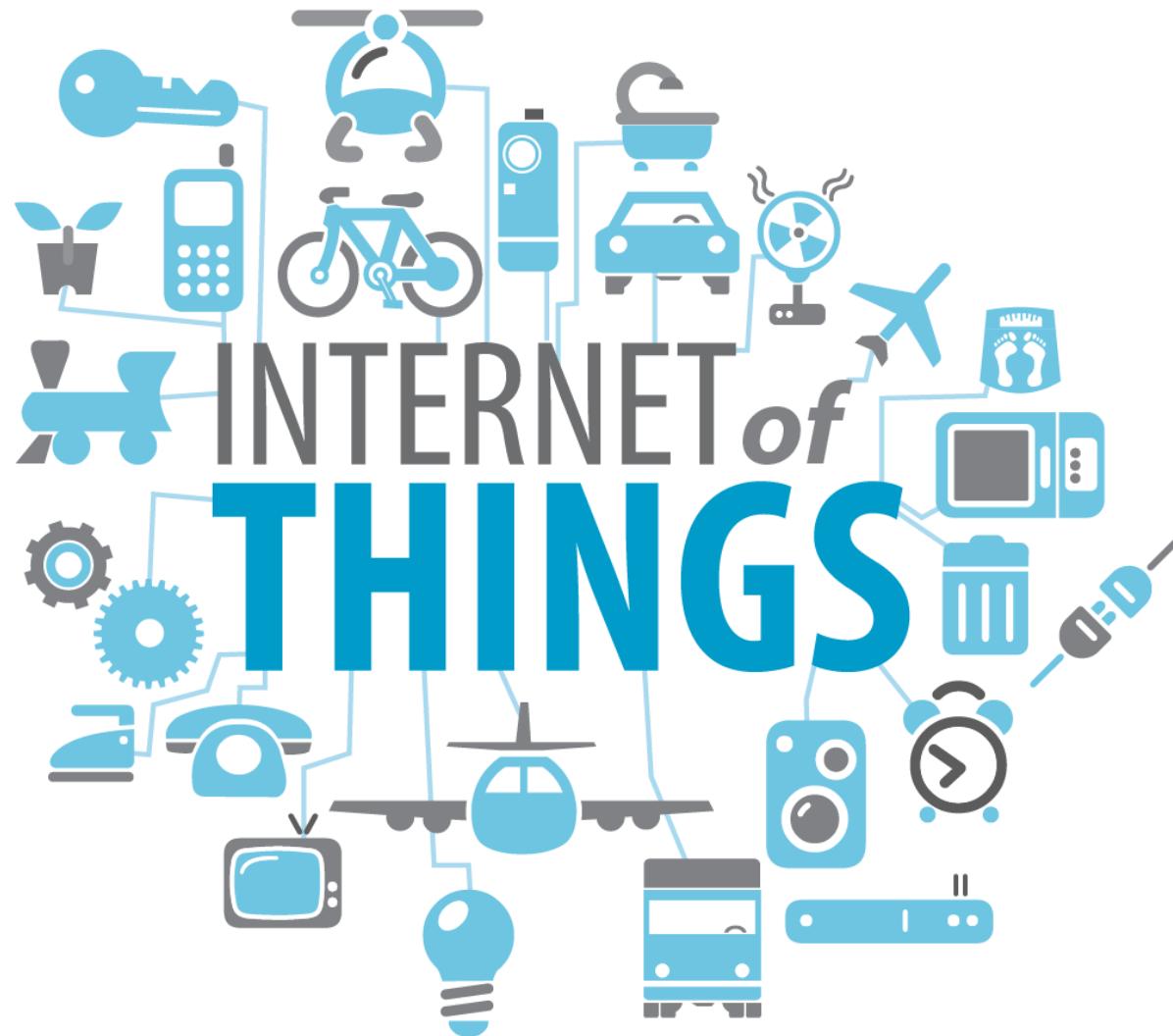
Internet of Things (IoT)

3

CONCEPTS

Sensorization

Hands On



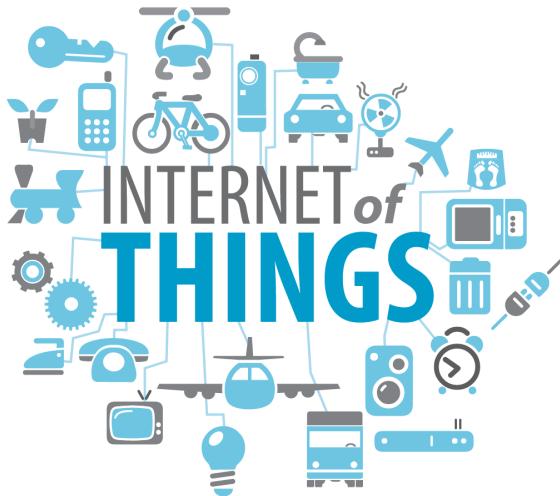
Internet of Things (IoT)

4

CONCEPTS

Sensorization

Hands On



An open and comprehensive **network of intelligent objects** that have the capacity to auto-organize, share information, data and resources, **reacting** and **acting** in face of situations and changes in the environment.

Internet of People (IoP)

5

CONCEPTS

Sensorization

Hands On



Internet of People (IoP)

6

CONCEPTS

Sensorization

Hands On



A dynamic global **network** where **things** and **people** communicate and understand each other; where everyone and everything can sense the other and the world, and act on such knowledge and information, aiming to enhance **people's quality of life**.

Smart Cities

7

CONCEPTS

Sensorization

Hands On



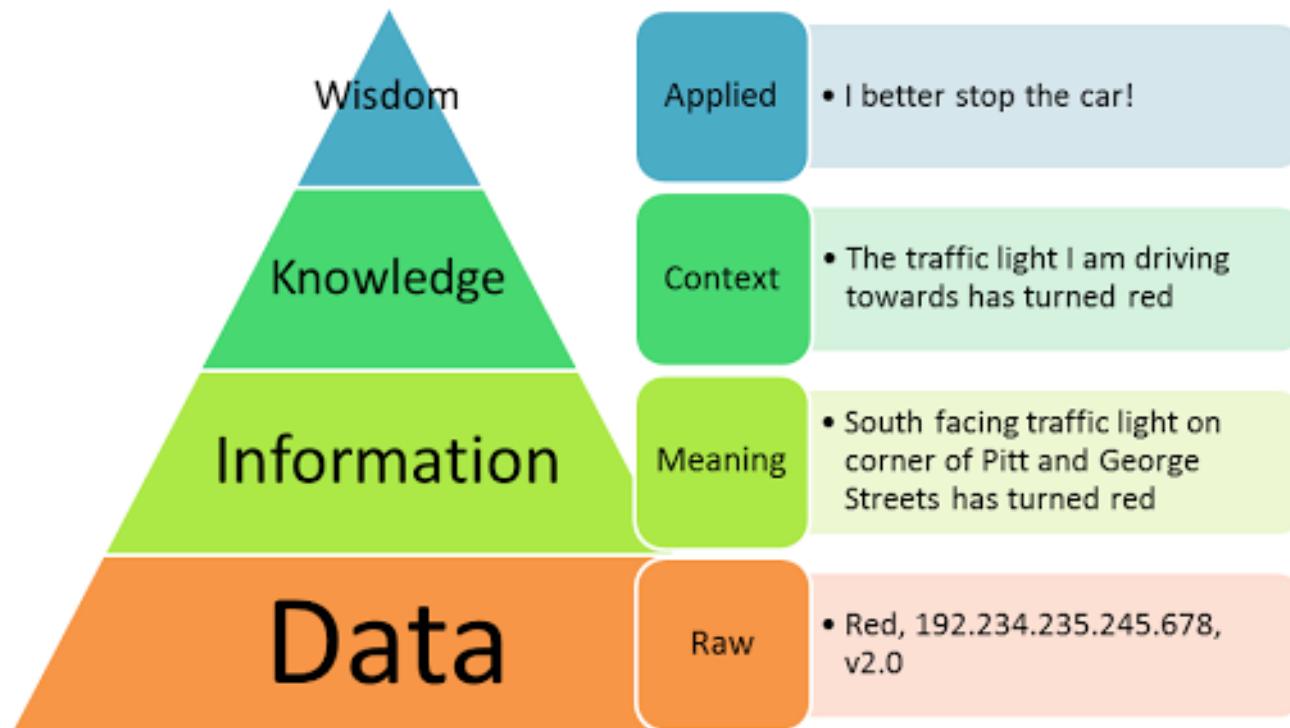
Data Pyramid (aka DIKW pyramid)

8

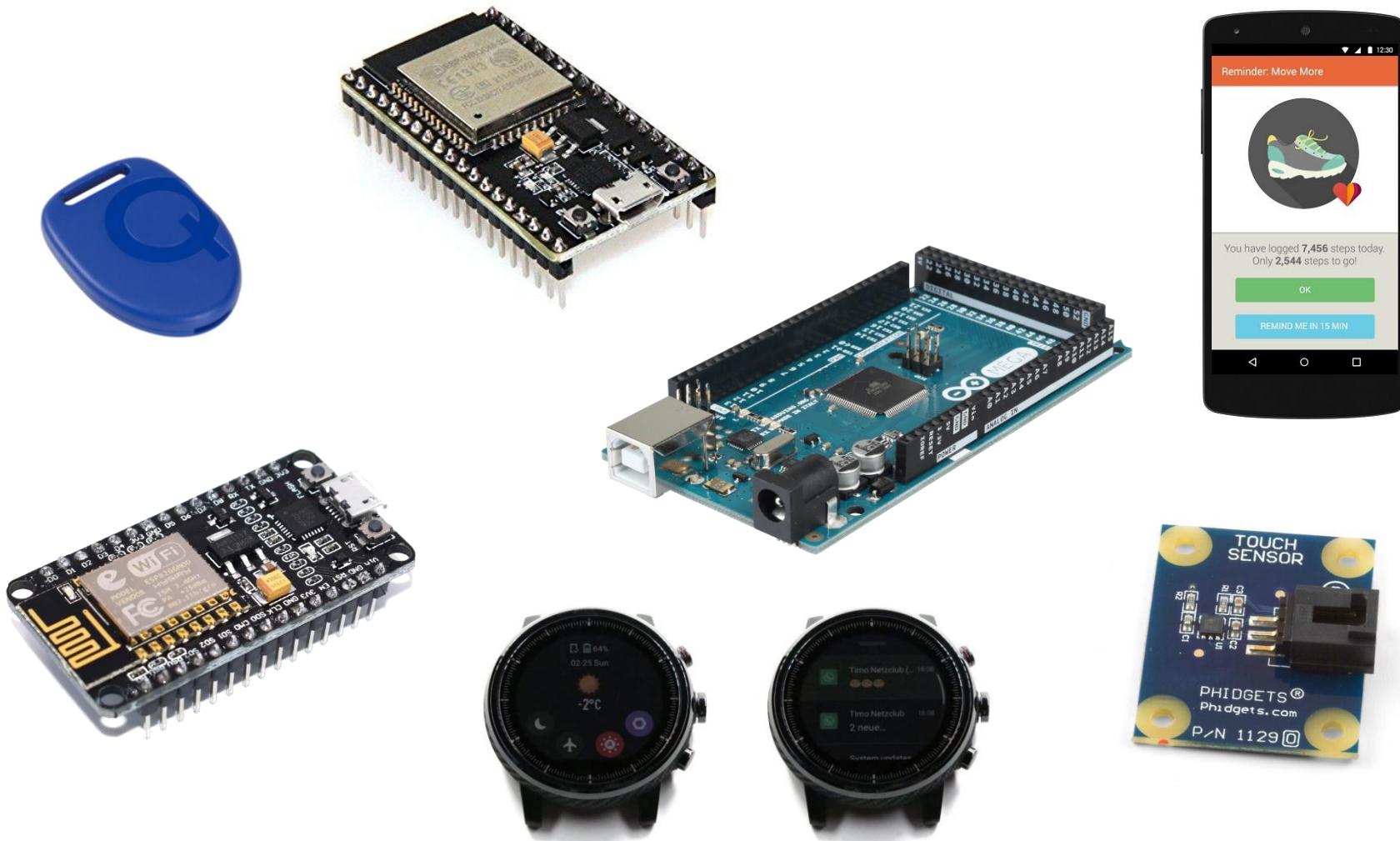
CONCEPTS

Sensorization

Hands On



Sensorization



Sensorization

10

Concepts

SENSORIZATION

Hands On

A **sensor** is something that is able to **percept phenomena that is being observed** and **translate its state**

Traditionally sensors were **physical** and observed physical phenomena but sensors may also be **virtual**:

- Access to web API
- Mathematical formulae

Currently, data fusion can also infer virtual assets such as:

- Emotions
- Well Being
- Sustainability
- Happiness

Ambient Intelligence

11

Concepts

SENSORIZATION

Hands On

1. Data Acquisition

- Sensors
- Services
- Data processing

2. Reasoning

- Data Modeling
- Machine Learning
- Decision Models

3. Actuation

- Notifications
- Interactions
- Actions

Beacons

12

Concepts

SENSORIZATION

Hands On

A **bluetooth-based sensor** with low-cost, low-power transmitters (a Bluetooth Smart/LE signal), which **notify bluetooth devices** of one's presence.

This signal makes it **possible to identify the beacon** as well as other telemetry information about the receiving device. It has **no user interface or GPS** capabilities.

The beacon works as such:

- periodically wakes up
- transmits a BLE signal
- returns to a low-power state



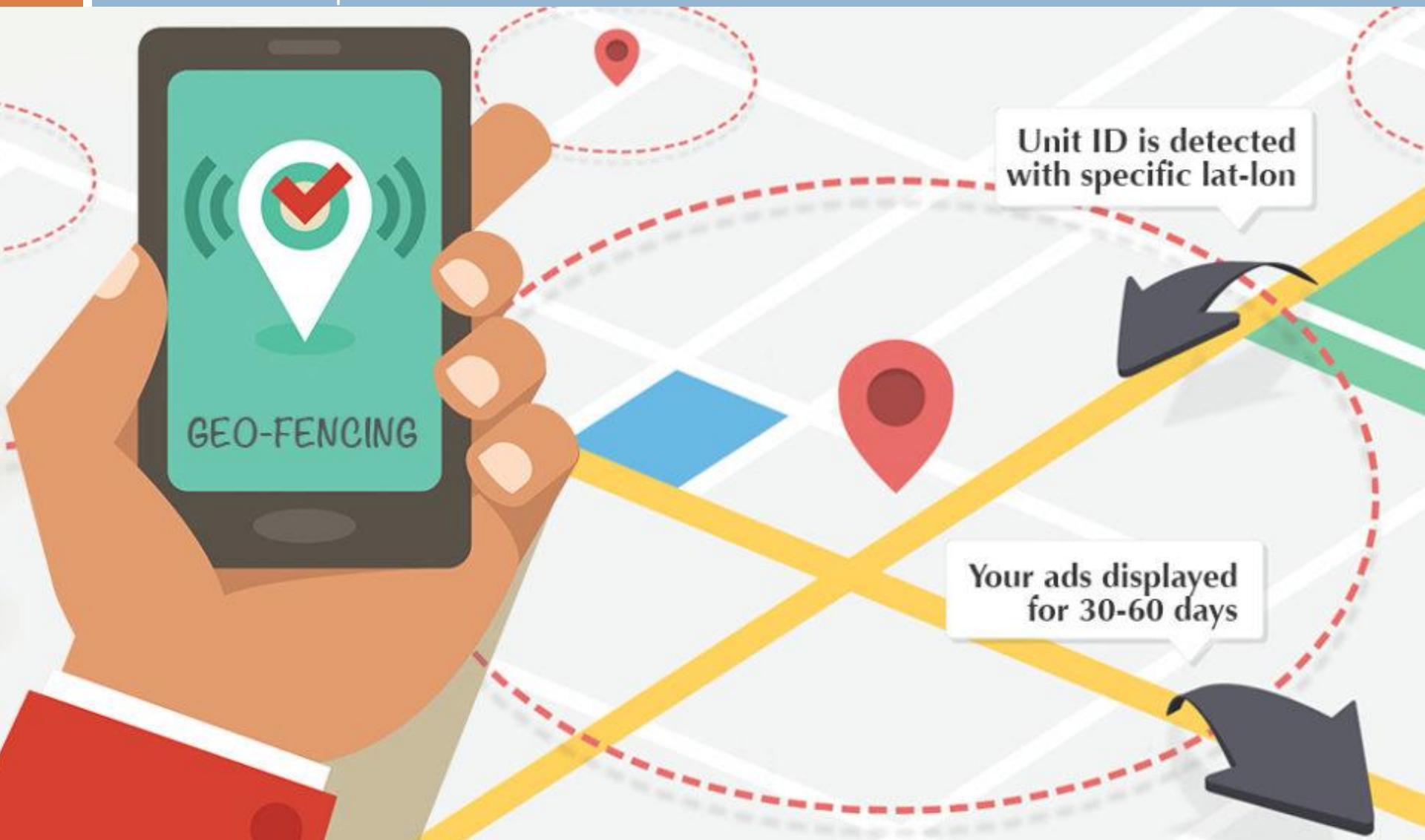
Beacons - Case Studies

13

Concepts

SENSORIZATION

Hands On



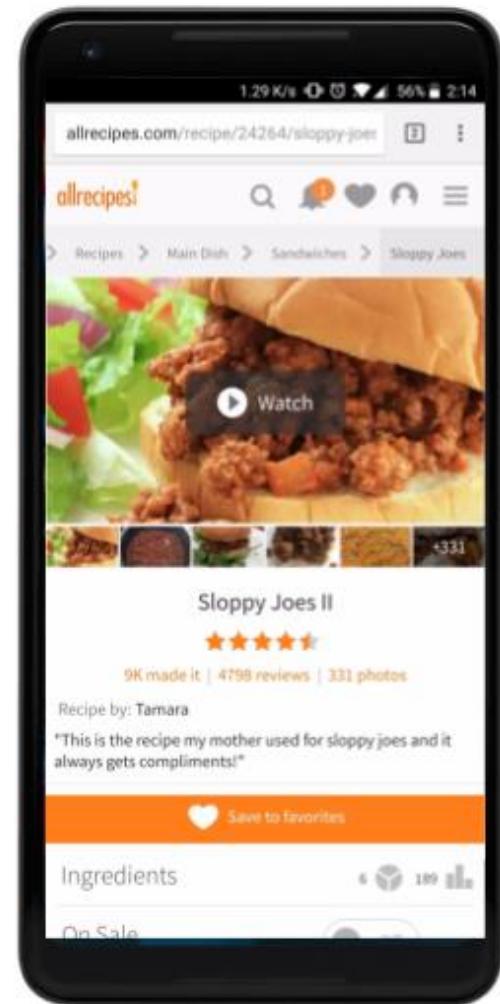
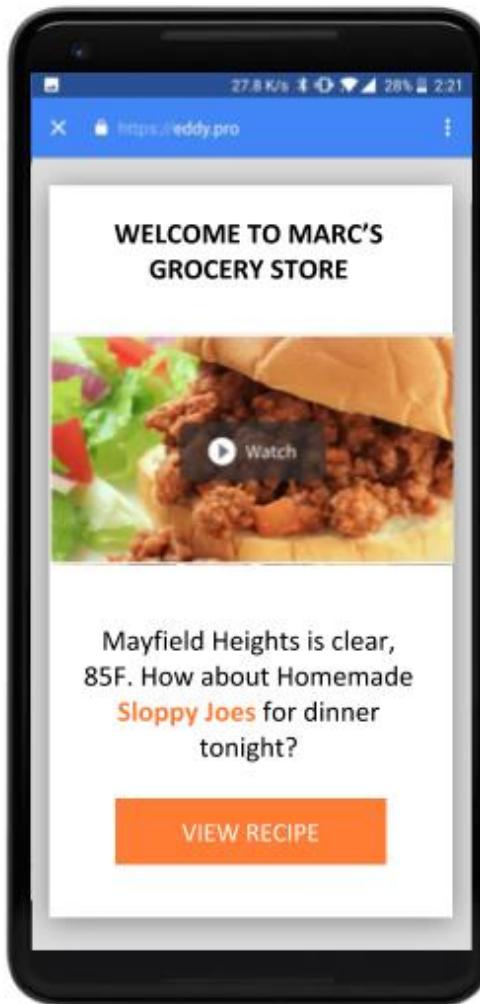
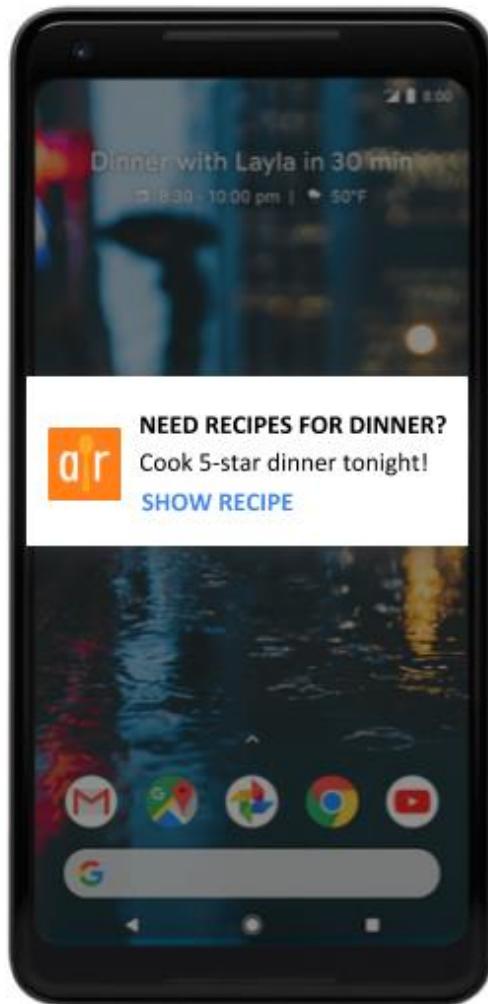
Beacons - Case Studies

14

Concepts

SENSORIZATION

Hands On



ESP8266

15

Concepts

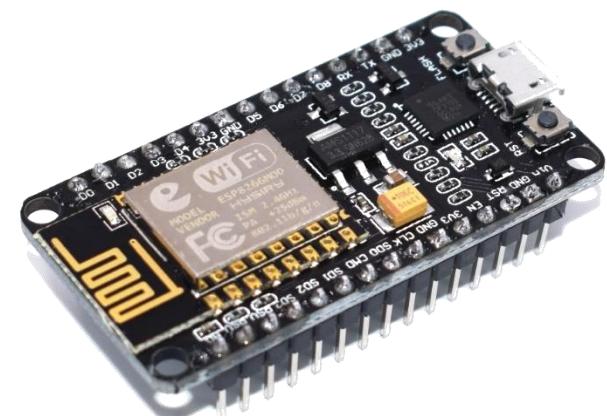
SENSORIZATION

Hands On

A low-power **Arduino type board**, suitable for the IoT, that can facilitate the **bridge towards Smart Cities**, removing the need for wired communication and processing.

A very interesting set of features...

- Wi-Fi capability (2.4 GHz band)
- 4 MB of flash memory
- a micro-USB interface
- a built-in antenna
- open-source
- small dimensions (4.8x2.4x0.5cm)
- low weight (109g)
- Ultra-Low Power Consumption



ESP32

16

Concepts

SENSORIZATION

Hands On

A low-power **Arduino type board**, suitable for the IoT, that can facilitate the **bridge towards Smart Cities**, removing the need for wired communication and processing.

A very interesting set of features...

- The same as the previous slide!
- **BLE connectivity** (Hybrid Wi-Fi & Bluetooth Chip)!
- **Dual-core!**

However, being this a new player the availability (and **documentation**) of libraries for the ESP32 is **significantly lower** when compared to the ESP8266.



Arduino(-type) Boards - Case Studies

17

Concepts

SENSORIZATION

Hands On



Arduino(-type) Boards - Case Studies

18

Concepts

SENSORIZATION

Hands On



Arduino(-type) Boards - Case Studies

19

Concepts

SENSORIZATION

Hands On



Soft Sensors

20

Concepts

SENSORIZATION

Hands On



Soft Sensors

21

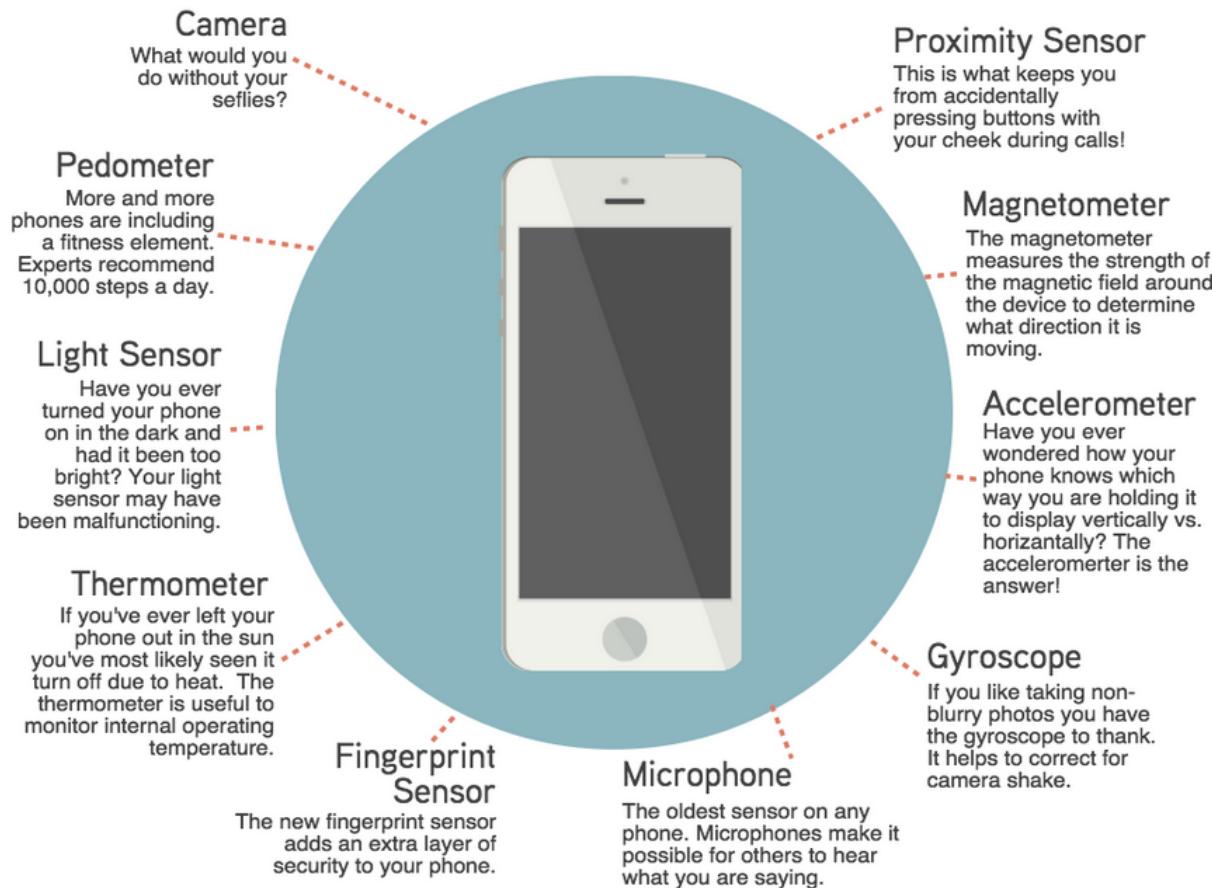
Concepts

SENSORIZATION

Hands On

Sensors Everywhere

The average smartphone has at least 10 sensors.
Here are the most common.



Virtual Sensors

22

Concepts

SENSORIZATION

Hands On



<https://openweathermap.org/api>



<https://docs.openaq.org/>



FOR DEVELOPERS

<https://developer.tomtom.com/>



<https://www.openuv.io/uvindex>



<https://pro.whitepages.com/apis/>



<https://developers.google.com/>



<https://developers.coinbase.com/>

Virtual Sensors

23

Concepts

SENSORIZATION

Hands On

- Essentially, you need to make **API calls** through **HTTP Get Requests**
- Parse the received JSON (you could go for XML if you really want to...)
- You may, or may not, need an **API key**
- You will definitely have a limited amount of requests to make

TomTom API

24

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal homepage. At the top, there's a navigation bar with tabs for "Maps APIs" (which is highlighted in red), "Navigation", and "Data services". Below the tabs are links for "Products", "Documentation", "Community", "Support", and "Pricing". A red button labeled "MY DASHBOARD" is on the right, along with a search icon and a user profile for "Bruno Fernandes". The main visual is a large image of a tablet displaying a map with a rocket launching from it, surrounded by colorful geometric shapes. A red "GET STARTED" button is in the top left of this image. Below the image, a red button with a dropdown arrow has the text "Scroll to explore". The central text on the page reads "Build different with TomTom Maps APIs." and "Powered by TomTom's real time maps and traffic data, with global coverage & advanced functionalities."

TomTom API

25

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal's Key Management page. On the left, a sidebar lists 'DASHBOARD', 'Keys' (selected), 'Credits', 'Payments', and 'Activity'. The main area shows a table of keys, with one row highlighted. A red 'ADD NEW KEY' button is at the top right of this table. A blue arrow points from the text 'ADD NEW KEY' in the 'Sensorization' section below to this button. The table columns include 'Key' (redacted), 'Products' (with checkboxes for various APIs like Map Display API, Search API, etc.), and 'Security' (Domain whitelist set to Off). To the right of the table, a callout bubble says 'Not sure what's next? We are here to help.' and contains a link 'TRY OUR API KEY GUIDE >'. A cartoon character icon is also present.

Maps APIs Navigation Data services

TOMTOM
DEVELOPER PORTAL

Products Documentation Community Support Pricing MY DASHBOARD Q Bruno Fernandes ▾

DASHBOARD

Keys >

Credits >

Payments >

Activity >

ADD NEW KEY

Key [REDACTED]

Products

Map Display API	Search API
Extended Search API	Routing API
Extended Routing API	Traffic Flow API ✓
Traffic Incidents API ✓	Geofencing API
Location History API	Notifications API

Security

Domain whitelist Off

Not sure what's next? We are here to help.

TRY OUR API KEY GUIDE >

2500 free transactions per day on all APIs

(<https://developer.tomtom.com/user/me/apps/>)

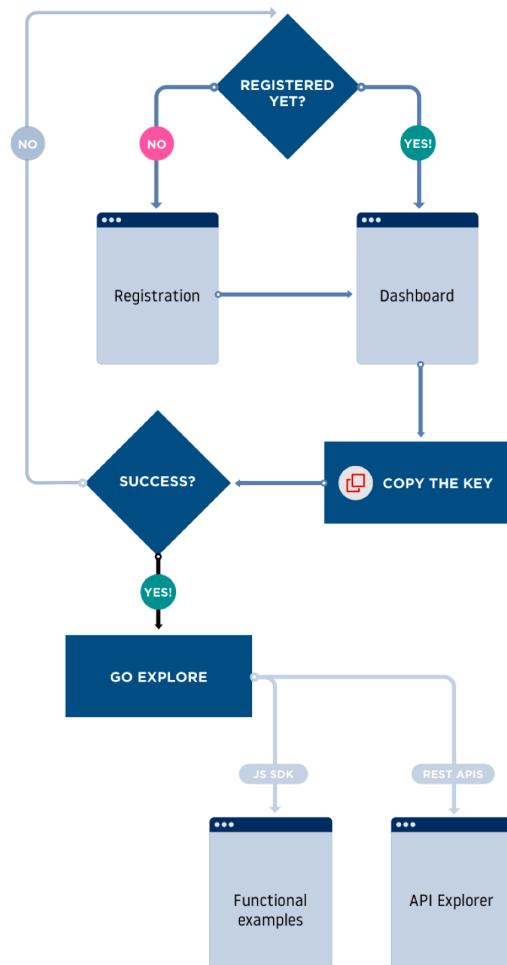
TomTom API

26

Concepts

SENSORIZATION

Hands On



TomTom API

27

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal homepage. At the top, there's a navigation bar with links for 'Maps APIs', 'Navigation', and 'Data services'. Below the navigation is a main menu with 'Products', 'Documentation', 'Community', 'Support', and 'Pricing'. To the right of the menu is a red button labeled 'MY DASHBOARD' and a user profile icon for 'Bruno Fernandes'. The main content area features a section titled 'How to use TomTom API Keys?'. It includes a tip about replacing placeholder keys in URLs, a code snippet for a specific URL, and instructions for general API requests. There's also a link to try out the Maps SDK for Web in CodePen.

Maps APIs Navigation Data services

Products Documentation Community Support Pricing **MY DASHBOARD** Bruno Fernandes ▾

How to use TomTom API Keys?

👉 Eager to see the result right away? Here is an example that we've prepared for you to get started. Remember to replace `Your_API_Key` with your own key (that you can grab from the [dashboard](#)). Then paste the whole query in your browser address bar and hit enter.

```
1 | https://api.tomtom.com/map/1/tile/basic/main/o/o/o.png?view=Unified&key=YOUR_API_KEY
```

You should now see a map tile containing the whole world. Success!

👉 Let's now look at a more general example. Through the documentation on Developer Portal, you will find API request URL formats looking somewhat like the one below:

```
http|https://baseURL/map/versionNumber/tile/layer/style/zoom/X/Y.format?key=Your_API_Key&view=vi(...)
```

Replace `Your_API_Key` with your API key and `baseURL` with `api.tomtom.com`. Then replace **required parameters** according to the service documentation (you'll find all the details in the *Required parameters* tables inside each endpoint documentation) and you are ready to make your second request with TomTom APIs. Remember that you can also evaluate TomTom APIs with our API Explorer ([Maps](#), [Search](#), [Routing](#), [Traffic](#)) without writing a line of code.

How to try out Maps SDK for Web in CodePen using my API Key?

TomTom API

28

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal for the Traffic API. At the top, there's a navigation bar with links for 'Maps APIs', 'Navigation', and 'Data services'. Below the navigation is the TomTom logo and a menu with 'Products', 'Documentation', 'Community', 'Support', and 'Pricing'. A red button labeled 'MY DASHBOARD' is on the right, along with a search icon and a user profile for 'Bruno Fernandes'. The main content area has a title 'Traffic API' with a car icon. A prominent red button says 'GET YOUR KEY'. Below it, a horizontal menu has 'HOME' underlined. Other menu items include 'DOCUMENTATION', 'SUPPORT', 'API EXPLORER', and 'RELEASE NOTES'. A sidebar on the right is labeled 'FEEDBACK'. The main content section starts with 'Service version: 4' and 'Last edit: 2019.05.16'. It has a heading 'On this page' with links to 'Purpose >', 'Common use cases you can implement >', 'Getting Started >', and 'Features >'. Under 'Purpose', there's a section titled 'What is the Traffic API?'. It describes the API as a suite of web services for real-time traffic data. It mentions 'Traffic Incidents' and 'Traffic Flow' as key components. A note states that the services are based on TomTom Traffic™. Below this is a 'Getting started' section with a numbered list: 1. Register/Sign in to the TomTom Developer Portal. 2. Request an evaluation API key to access this service. 3. Read the API documentation and start coding.

TomTom API

29

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal's Traffic API Market Coverage page. At the top, there are navigation links for Maps APIs, Navigation, and Data services. Below that is a header with the TomTom logo, a search bar, and a user profile for Bruno Fernandes. The main content area features a large "Traffic API" heading with a car icon, a prominent red "GET YOUR KEY" button, and a "Market Coverage" section. The "Market Coverage" section contains a table showing the availability of Traffic Incidents and Traffic Flow APIs across various countries by continent. A "FEEDBACK" button is located on the right side of the table.

Continent	Country	Country Code	Traffic Incidents API	Traffic Flow API
Africa	Egypt	EG/EGY	•	•
	Kenya	KE/KEN	•	•
	Lesotho	LS/LSO	•	•
	Morocco	MA/MAR	•	•
	Mozambique	MZ/MOZ	•	•
	Nigeria	NG/NGA	•	•
	South Africa	ZA/ZAF	•	•
Americas	Argentina	AR/ARG	•	•
	Brazil	BR/BRA	•	•
	Canada	CA/CAN	•	•
	Chile	CL/CHL	•	•

(<https://developer.tomtom.com/traffic-api/market-coverage>)

TomTom API

30

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal interface. At the top, there's a navigation bar with links for 'Maps APIs', 'Navigation', 'Data services', 'Products', 'Documentation' (which is highlighted in red), 'Community', 'Support', 'Pricing', 'MY DASHBOARD', and a user profile for 'Bruno Fernandes'. Below the navigation, the URL path is // TRAFFIC API / TRAFFIC API DOCUMENTATION / TRAFFIC FLOW / FLOW SEGMENT DATA. On the left, a sidebar menu has 'Traffic Flow' expanded, with 'Flow Segment Data' selected (also highlighted in red). Other options in the sidebar include 'Traffic Incidents', 'Raster Flow Tiles', 'Vector Flow Tiles', 'Release Notes', 'Map Styles', and 'Vector Tile Structure'. The main content area is titled 'Request data' and contains sections for 'HTTPS method' (set to 'GET'), 'URL format' (with instructions and examples), 'Example' (a curl command), 'curl command' (another curl command), and 'Request parameters' (a table with one row for 'baseURL'). A 'FEEDBACK' button is located on the right side of the page.

Request data

HTTPS method: GET

URL format

For ease of viewing and identification:

- Required constants and parameters are shown in **bold** text.
- Optional parameters are shown in plain text.

```
http(s)://baseURL/traffic/services/versionNumber/flowSegmentData/style/zoom/format?key=Your_API_Key&point=point&unit=unit&thickne
```

Example

```
https://api.tomtom.com/traffic/services/4/flowSegmentData/absolute/10/xml?key=Your_API_Key&point=52.41072,4.84239
```

curl command

```
curl 'https://api.tomtom.com/traffic/services/4/flowSegmentData/absolute/10/xml?key=Your_API_Key&point=52.41072,4.84239'
```

Request parameters

The following table describes the parameters that can be used in a request.

- Required parameters **must be used** or the call will fail.
- Parameters and values are case-sensitive.
- Optional parameters may be used.
- If there is a default value that will be assumed when an optional parameter is not used, it is shown in the table.

Parameter	Description
baseURL	The base URL for calling TomTom services. Value: api.tomtom.com

(<https://developer.tomtom.com/traffic-api/traffic-api-documentation/traffic-flow>)

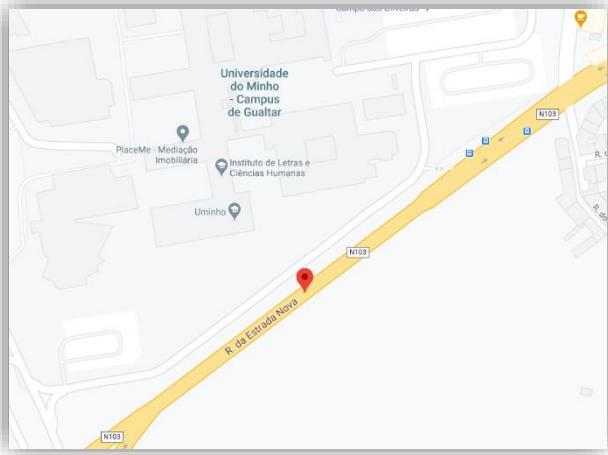
TomTom API

31

Concepts

SENSORIZATION

Hands On



```
{  
  "flowSegmentData": {  
    "frc": "FRC2",  
    "currentSpeed": 27,  
    "freeFlowSpeed": 35,  
    "currentTravelTime": 51,  
    "freeFlowTravelTime": 39,  
    "confidence": 0.9800000190734863,  
    "roadClosure": false,  
    "coordinates": {  
      "coordinate": [  
        {  
          "latitude": 41.557950157701924,  
          "longitude": -8.396579638426061  
        },  
        {  
          "latitude": 41.55806672431858,  
          "longitude": -8.396506078025567  
        },  
        {  
          "latitude": 41.55808329853029,  
          "longitude": -8.396491691006048  
        },  
        {  
          "latitude": 41.558274049923824,  
          "longitude": -8.396246091669411  
        },  
        {  
          "latitude": 41.55830085581175,  
          "longitude": -8.396234368638687  
        },  
        {  
          "latitude": 41.55832935270641,  
          "longitude": -8.396184658943241  
        },  
        ...  
      ]  
    }  
  }  
}
```

https://api.tomtom.com/traffic/services/4/flowSegmentData/absolute/10/json?point=41.559485%2C-8.395822&unit=KMPH&openLr=true&key=*****

TomTom API

32

Concepts

SENSORIZATION

Hands On

The screenshot shows the TomTom Developer Portal interface. At the top, there's a navigation bar with links for 'Maps APIs', 'Navigation', and 'Data services'. Below the navigation bar, the 'MY DASHBOARD' button is highlighted in red, along with a user profile for 'Bruno Fernandes'. The main content area is titled 'TRAFFIC API DOCUMENTATION / TRAFFIC INCIDENTS / INCIDENT DETAILS'. On the left, there's a sidebar with sections for 'Incident Details' (which is currently selected), 'Incident Viewport', 'Raster Incident Tiles', 'Vector Incident Tiles', 'Release Notes', 'Traffic Flow', 'Map Styles', and 'Vector Tile Structure'. The main content area is titled 'Request data' and contains sections for 'HTTPS method' (set to 'GET'), 'URL format', 'Example' (showing a URL template), 'curl command' (showing a curl command template), and 'Request parameters'. A table titled 'Required parameters' is partially visible at the bottom. A 'FEEDBACK' button is located on the right side of the page.

(<https://developer.tomtom.com/traffic-api/traffic-api-documentation/traffic-incidents>)

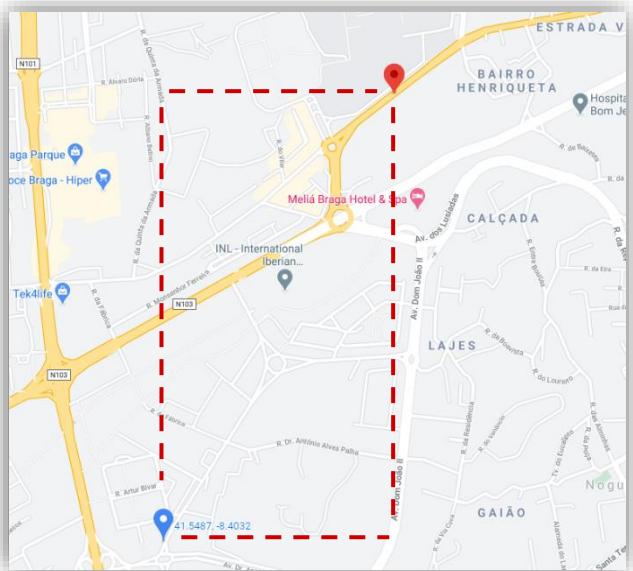
TomTom API

33

Concepts

SENSORIZATION

Hands On



Response body came empty!

No incidents within the designated area!

Hands On

34

Concepts

Sensorization

HANDS ON

The screenshot shows a software application window titled "Commsensing | Arduino 1.8.2". The interface includes a toolbar with icons for file operations, a menu bar with "Fichero", "Editar", "Recursos", "Ferramentas", and "Ajuda", and a top status bar showing "Connected to Violation-MBED4" at address "192.168.1.81".

The main area contains two tabs: "Commsensing" (selected) and "Sensorización". The "Commsensing" tab displays a serial monitor window with the title "COM1". The monitor shows a log of probe requests captured by the device. The log entries include:

```
*** 0395246 Probe Request Capture by Bruno Fernandes ***
----- Connected to Violation-MBED4 IP address: 192.168.1.81
*** Connecting to MQTT... Connected! ***
*** All setup has been made! Timer 0 is enabled and publish of data will happen every 4000 milliseconds ***
*** Timer activated ***
Probe request From: da:ca:19:24:32:00 Rssi: -60 Millis Since Detected: 33396
Probe request From: da:ca:19:24:32:00 Rssi: -60 Millis Since Detected: 47444
Probe request From: da:ca:19:24:32:00 Rssi: -60 Millis Since Detected: 47449
Probe request From: da:ca:19:24:32:00 Rssi: -43 Millis Since Detected: 47550
Probe request From: da:ca:19:24:32:00 Rssi: -42 Millis Since Detected: 47632
Probe request From: da:ca:19:24:32:00 Rssi: -42 Millis Since Detected: 47725
Probe request From: da:ca:19:24:32:00 Rssi: -39 Millis Since Detected: 47797
Probe request From: da:ca:19:24:32:00 Rssi: -39 Millis Since Detected: 47879
Probe request From: da:ca:19:24:32:00 Rssi: -34 Millis Since Detected: 47962
Probe request From: da:ca:19:24:32:00 Rssi: -39 Millis Since Detected: 48044
Probe request From: da:ca:19:24:32:00 Rssi: -71 Millis Since Detected: 48126
```

Below the monitor are checkboxes for "Avanza automatico de linea" and "Show timestamp", along with buttons for "Nova linha", "9600 baud", and "Clear output".

The "Sensorización" tab shows a table with columns "ID", "Name", and "Value". The table contains the following data:

ID	Name	Value
1	Temperature	22.5
2	Humidity	55.2
3	Light	100.0
4	CO2	1000.0
5	UV	100.0

At the bottom of the screen, a terminal window shows the command "Serial.println("Times is disabled");" being run.

Bottom status bar: "Running 0x4000 bytes starting at 0x00000000" and "Optimizing 374336 bytes from C:\Users\brunof\Documents\Arduino\Commsensing\src\app.h to flash at 0x00000000".