Inventory Management

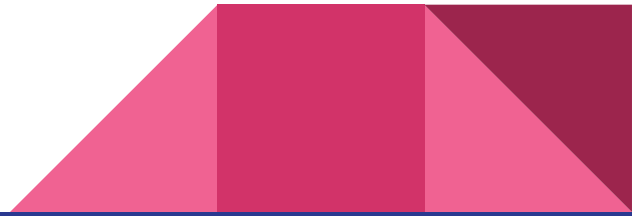
CYBERPHYSICAL SYSTEMS AND INTERNET OF THINGS - M.EIC - 2022-2023

João Carlos Machado Rocha Pires (UP201806079)

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Agenda

- Application
- Planned Proof-of-Concept
- Realised Proof-of-Concept
    - Reasons for change
- Biggest Challenges
- What I’ve learned
- Live Demo
- References



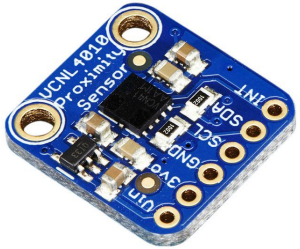
Application

- Fully-functional prototype of an inventory management system, using software and hardware, the connection between both, and a physical mockup of a warehouse for demonstration purposes.
- Each interconnected device is associated to a section of the warehouse:
  - Responsible to track the physical products associated to;
  - Highlight itself whenever needed;
  - Gather environmental information using the sensors associated to each device.
- Usage of AWS IoT Core services for bi-directional communication between the interconnected devices and the Outsystems application.
- All the communication is made by WiFi using the MQTT protocol (publisher/subscriber).

Planned Proof-of-Concept

Materials:

- 3 x ESP32 (M5 Atom Lite)
- 1 / 2 x Load Cell HX711 Amplifier (Sensor de Peso)
- 1 / 2 x VCNL4010 (Sensor de Proximidade)



Software:

- Azure IoT services
- Microsoft Power Apps

Realised Proof-of-Concept

Materials:

- 3 x ESP32 (M5 Atom Lite)
- 1 x TVOC/eCO2 Gas Sensor Unit (SGP30)
- 2 x ENV III Unit with Temperature, Humidity, and Air Pressure Sensor (SHT30+QMP6988)

Software:

- AWS IoT Core services
- Outsystems

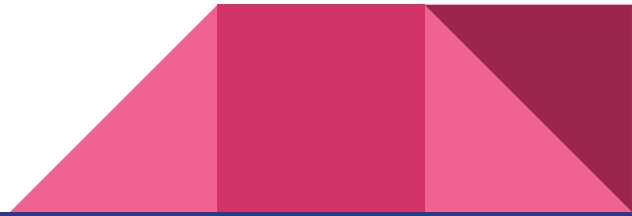
Realised Proof-of-Concept - Reasons for change

Materials

- Not available during the project development

Software

- Problems with the setup led to the usage of an alternative technology
- With the change on the IoT cloud platform, the technology to build the frontend was also changed



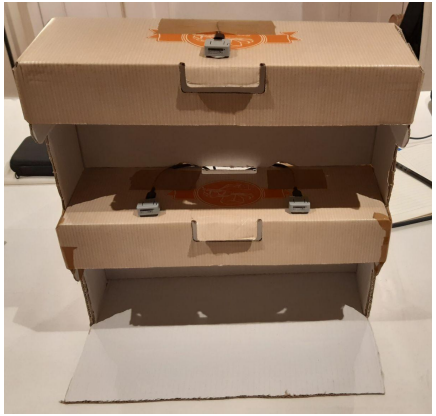
Biggest Challenges

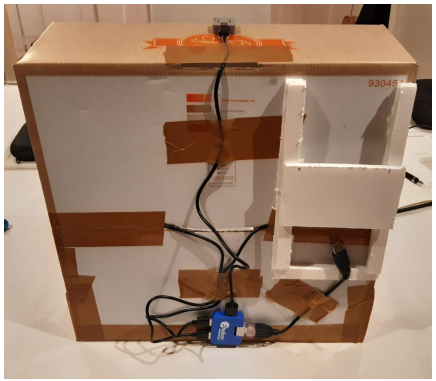
- Setup of the devices
  - WiFi communication
- Setup of the cloud platform
  - MQTT protocol implementation
  - Communication with the devices
- Testing
- Time constraints

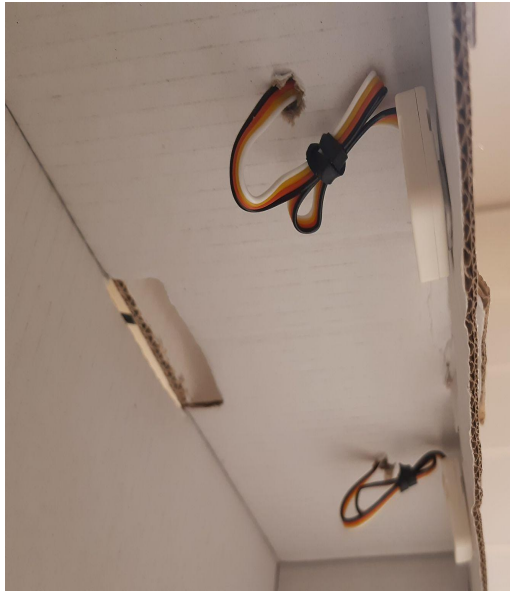
What I’ve learned

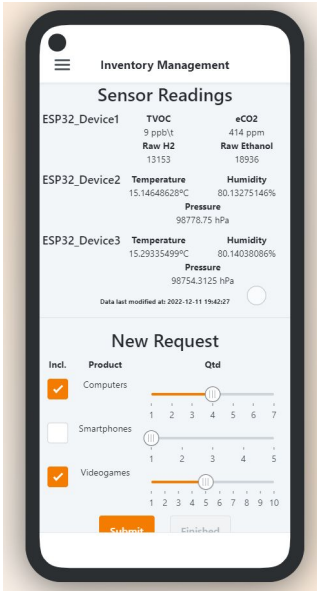
- Program an ESP32 device with WiFi capabilities
- Setup of an IoT cloud project (work with AWS IoT Core services)
- Implementation and setup of the MQTT protocol in different technologies
- Not everything goes as planned, so alternatives have to be considered prior to the start of a project (have a plan B / backup)

Live Demo








Inventory Management
Sensor Readings
ESP32_Device1
TVOC
9 ppb\t
eCO2
414 ppm
Raw H2
13153
Raw Ethanol
18936
ESP32_Device2
Temperature
15.14648628°C
Humidity
80.13275146%
Pressure
98778.75 hPa
ESP32_Device3
Temperature
15.29335499°C
Humidity
80.14038086%
Pressure
98754.3125 hPa
Data last modified at: 2022-12-11 19:42:27
New Request
Incl.
Product
Qtd
Computers
Smartphones
Videogames

References

ATOM Lite Official Documentation

ATOM Lite Official Github

ATOM Lite - Get Started with Arduino

AWS IoT Core services setup

Arduino programming with MQTT protocol

AWS Credentials Generator guide

AWS IoT Core services connection with ESP32 devices

Outsystems - build IoT mobile application

Outsystems - connection with AWS IoT Core services

Outsystems Forge - MQTT Mobile Client