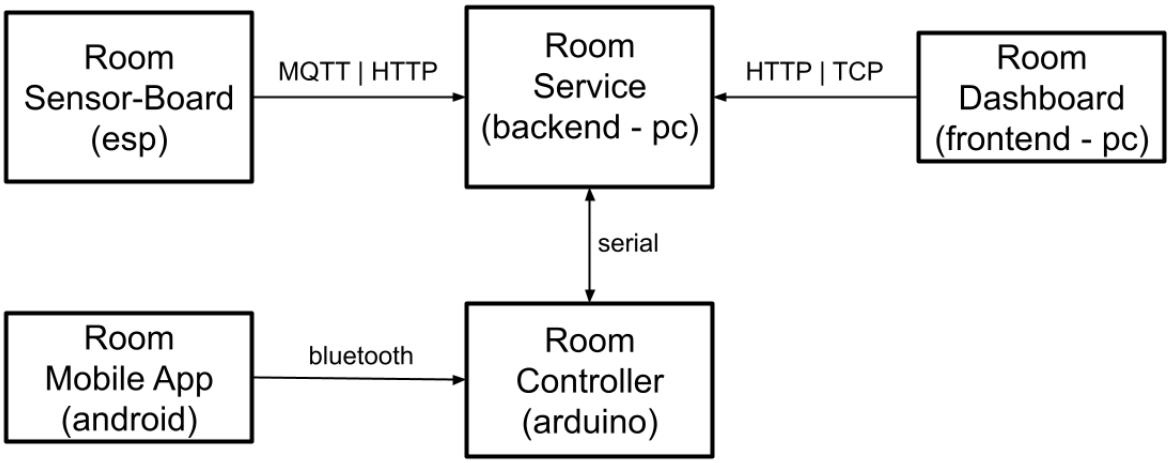
**Embedded Systems and IoT - a.y. 2022-2023**

# Assignment #03 - *Smart Room*

**v0.9-20221212**

We want to realise an IoT system implementing a simplified version of a *smart room*, as a smart system monitoring and controlling the state of a room (e.g. in a Campus).

The system is composed of 5 subsystems:




- **Room Sensor-Board (esp)**
  - embedded system to monitor the state of the room by using a set of sensors
  - It interacts with the Room Service (via MQTT[1])
- **Room Service (backend - pc)**
  - service functioning as the main unit governing the management of the room
  - it interacts through the serial line with the Controller (arduino)
  - it interacts via MQTT[1] with the Room SensorBoard (esp)
  - it interacts via HTTP[2] with the Dashboard (frontend/PC)
- **Room Controller (Arduino)**
  - embedded system controlling lighting and roller blinds
  - it interacts via serial line with the Room Service and via BT with the Room App
- **Room App (Android - smartphone)**
  - mobile app that makes it possible to manually control lights and roller blinds
  - it interacts with the Room Controller via Bluetooth
- **Room Dashboard (Frontend/web app on the PC)**
  - front-end to visualise and track the state of the room

[1] HTTP can be used instead of MQTT, if needed
[2] TCP can be used instead of HTTP, if needed

        - it interacts with the Room Service

*Hardware components*

- Room Sensor-board
    - SoC ESP32 board (or ESP8266) including
        - a green led
        - 1 PIR
        - 1 photoresistor analog sensor
- Room Controller
    - Microcontroller Arduino UNO board including:
        - 1 green led simulating a light subsystem
        - 1 servo motor simulating the roller blind subsystem
        - 1 Bluetooth module HC-06 o HC-05

*General Behaviour of the system*

The Smart Room system is meant to control the lighting system and roller blinds according to the following policy:

- If no one is in the room, the light (of the lighting subsystem) should be off
- If someone enters in the the room and the room is dark, then the light should be turned on (if it was off)
- The roller blinds are fully rolled up automatically the first time someone enters in the room, from 8:00 (if someone enters)
- The roller blinds are fully unrolled at 19:00 (if they are up and no one is in the room), or as soon as someone who is still in the room at 19:00 leaves the room.
- Through the mobile app, a user can:
    - turn on or off the light
    - roll up / unroll – also partially (from 0 to 100%)
- Through the dashboard a room manager can:
    - track the state of the room
        - in particular in which hours and how long the lights where on
    - fully control the light and roller blinds

**Remark**: the light controlled by this policy representing the lighting system is represented by the green  led in the Room Controller.

It can be assumed that the room is accessed from 8:00 to 19:00.

Further details:

- About the Room Sensor-board

    - The led should be on when someone is in the room and off when no one is the room
- About the Room Controller
    - The servo motor controls/simulates the roller blinds
        - 0° means roller blinds completely rolled-up
        - 180° means roller blinds completely unrolled
    - the green light simulates the lighting system: on/off

- No specific constraints/requirements are given for the Room Mobile App and the Room Dashboard

---

*The assignment*

Design and develop a prototype of the Smart Room system, considering the following requirements

- **Room Sensor-Board - based on ESP32**
    - must use either the MQTT or HTTP to communicate with the Room Service
- **Room Controller - based on Arduino**
    - the control logic must be designed and implemented using finite state machines (synchronous or asynchronous)
    - must communicate with the Room Service via serial line
- **Room Service - in execution on a PC**
    - no specific constraints about the programming/sw technology to be used
    - must use either MQTT or HTTP to communicate with the Room Sensor-Board
- **Room App - based on Android (either real device or emulated) or any other mobile platform**
    - for real device, the communication with the Room Control must be base on the BT wireless technology
    - for emulated devices, the communication can be done using the serial line communicating with the Android Emulator through a software bridge, as presented in lab
- **Room Dashboard - to be run on a PC**
    - no specific constraints on the technologies to be used
    - it can be implemented as a web app running in a browser or a PC app based on sockets

**The Deliverable**

The deliverable consists in a zipped folder **assignment-03.zip** including:

- 5 subfolders (one for each subsystem)
    - room-service
    - room-sensor-board
    - room-controller
    - room-dashboard
    - room-app

- **doc** folder
    - including a brief report **(report.pdf)** describing the system, including also a description of FSMs, a representation of the schema/breadboard and the link to a short video demonstrating the system.