Larbi Ben M'Hidi University
Faculty of exact sciences and natural and life sciences Dr. Kouah S.
MI Department 3rd Year IS
Software Engineering

Mini project: Modeling an Internet of Things (IoT) system

1. **Objective**: Analysis and design of an information system using UML, with its implementation in an object-oriented programming language (Java).
2. **System description:** An IoT system for basic management of a Smart home is composed of (See Figure below):
   - A data **server , a machine having the following characteristics: (the characteristics of your PC).**
   - An **Arduino- Uno** as a type **microcontroller**.
   - A **test card , Breadboard** type **- 830 contacts** , to test the different components.
   - An **HC06** type **Bluetooth module** .
   - Two **sensors**, namely: **temperature sensor** (**LM35**) and the **CO** $_2$ sensor (**MQ-7**).
   - A **battery** for powering the components.
   - A set of wires (ie **pointing cables**) connecting these different components.
   - Two **LEDs** (small lamps), red and orange, which are used to alert the administrator.

The sensors and Bluetooth are placed directly on the breadboard and connected with the microcontroller using the wires. The battery is plugged directly into the microcontroller.
The sensors each measure the corresponding physical measure and transmit data to the Microcontroller. In turn, the Microcontroller transmits this data, via Bluetooth, to the server for possible processing. The server is responsible for displaying and analyzing the received data. If the temperature exceeds 45°, the red LED lights up. Thus, if the CO $_2$ level exceeds 100 ppm, the orange LED lights up.
There are two types of actors in this system:

1. **User** (multiple): consults the data. Each user has an account.
2. **Administrator:** manages users (creation, authentication, updating and removal), consults data and makes decisions.

3. **Required work:**
   1. **Model** this IoT system by developing the following diagrams: **Use cases, classes**, **sequences**, **activity**, **components** and **deployment diagrams**
   2. **Implement** the system thus modeled in Java using a simple communication mechanism between entities, such as communication via files or shared memory (use of variables).

   Note : The data from the sensors must be generated **randomly and periodically** (every 30 seconds).
4. **Optional work:**
   - Creation of a user-friendly interface for the application.
   - Representation of data received from sensors in the form of histograms.

Report to be submitted:

- A **report** containing the above diagrams with the listing of the **commented code** .
- Explain how you created your application with a conclusion on the work carried out.