IoT Application: Project Proposal Guidelines

Introduction

This document outlines the project proposal guidelines for the IoT Applications course for M1 students. Each group, equipped with an ESP32, Arduino Nano 33 IoT, a Raspberry Pi 4, and a sensor box, is tasked to develop an innovative IoT application.

If you don't have an SD Card for the Raspberry PI, consider using a PC as a hub/gateway hosting the IoT Stack.

Project Scope

Core Components:
- Microcontrollers: ESP32, Arduino Nano 33 IoT
- Server: Raspberry Pi 4 with IoTStack or a PC with IoT Stack (refer to this: https://github.com/josephazar/NODE-RED-IOT-STACK)
- Sensors: Various (temperature, humidity, accelerometer, etc.)
  - Extra Credit: Considering additional sensors than the DHT11 (temperature, humidity) and accelerometer

Communication Protocols:
- At least two communication protocols should be used in the application.
- Mandatory: MQTT (Message Queuing Telemetry Transport)
- Optional: HTTP, TCP (At least one to be included), WebSocket, BLE (Bluetooth Low Energy)

Server and Microcontroller Interaction:
- Microcontrollers should communicate with the server
- Microcontrollers can communicate with each other (BLE) ⇒ Optional
- The server can communicate with another server (Node-RED with Express or Cloud / Google Sheet / Firebase / etc) ⇒ Optional
- The server must send commands to the microcontrollers.
  - Ex: turning on LED, triggering some action, etc.

Software Stack:
- Arduino IDE for ESP32 and NANO 33 IOT.
- Node-RED: For at least two automation tasks.

- Database: InfluxDB (mandatory), SQL database (optional, if needed).
- Visualization: Grafana (mandatory), web visualization (optional, if needed).
- Web Server: NodeJS/Express (if needed).
- Android mobile app: Optional, if needed, for a BLE use case.

Extra Credit: Establishing BLE communication between ESP32 and Arduino.

Project Proposal Requirements

Project Idea: A brief description of the chosen application or use case.
Data Collection: Strategies for gathering data from sensors.
Communication: Explanation of how different communication protocols are implemented and integrated.
Data Processing and Storage: Methods for processing and storing data.
Visualization: Approach for data visualization using Grafana or similar tools.
Automation Tasks: Description of at least four automation tasks implemented in Node-RED.
Challenges and Solutions: Any anticipated challenges and proposed solutions.

Evaluation Criteria

- **Innovation:** Originality of the project idea and application.
  - If you have an interesting idea and lack the sensors/materials, you can create a simulator in NodeJS or Python to simulate the device's presence.
- **Complexity:** Effective use of the provided hardware and software stack.
  - Some ideas:
    - Sending data to Google Sheets / Firebase from Node-RED
    - Sending email if something abnormal happened (ex: high temperature)
    - A simple mobile app to show the data in real-time and send commands like turning on/off a led
      - Mobile app sends command to Node-RED, which in turn sends to IoT
    - Connecting your webcam to Node-RED for monitoring
    - Etc
- **Integration:** Seamless integration of different components and protocols.
- **Teamwork**: Coordination and task distribution within each group.
- **Presentation:** Clarity and thoroughness in explaining the project architecture, data flow, and functionalities.

- ○ Do not keep the presentation and demo for last minute
- **Github repo**: A Github repo showcasing your skills and the type of projects you can handle is always a good thing to have.

<u>Presentation date:</u> Wednesday, 4 December 2024