Wireless Laboratory Monitoring System

Real-Time Temperature/Humidity Heatmaps

Project Objectives

- Real-time monitoring of temperature and humidity in the laboratory
- Visualization as heatmaps
- Designed to be extensible (future sensor support)
- Flexibility between wireless and wired sensors
- Easy maintenance and future scalability

Sensor Layout and Measurement Layers

- Laboratory size: approximately 7x7 m
- Three measurement layers:
  - **Bottom (0.5 m)** – measures floor temperature
  - **Middle (1.5 m)** – human activity level
  - **Top (2.5 m)** – ventilation/airflow monitoring
- Additional placements:
  - Sensors in corners + center
  - Sensors near ventilation and door
  - One outdoor sensor for comparison

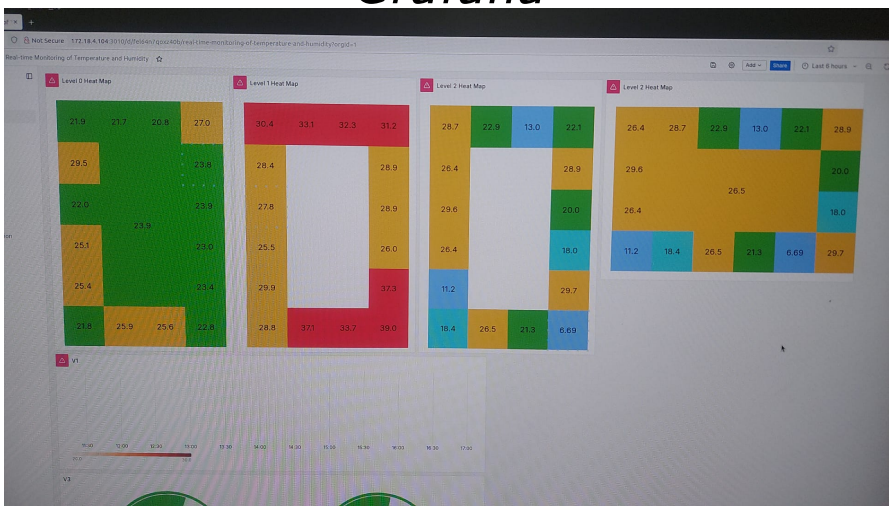
Data Storage: InfluxDB

- Time-series optimized database
- Open-source, fast, supports complex queries
- Compatible with Grafana, Python, and more

Data Visualization – Grafana Python

- **Grafana**
  - Heatmap by zone/layer
  - Remote access, scalable, alerting support

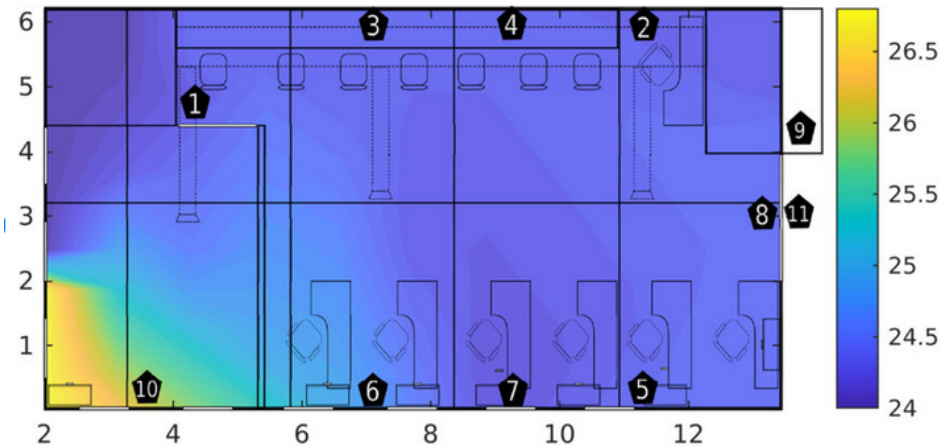
Grafana



Full Grafana

Data Visualization – Web / Mobile App

- **Custom Python**
  - Python script for interpolated heatmaps
  - Can be embedded in Grafana or a website



Heat map of the laboratory using real data and interpolation (Scenario 1).

- **Web / Mobile App**
  - Interactive UI, powerful visuals, requires more development and hosting

Sensor Type Comparison

Wired Sensors

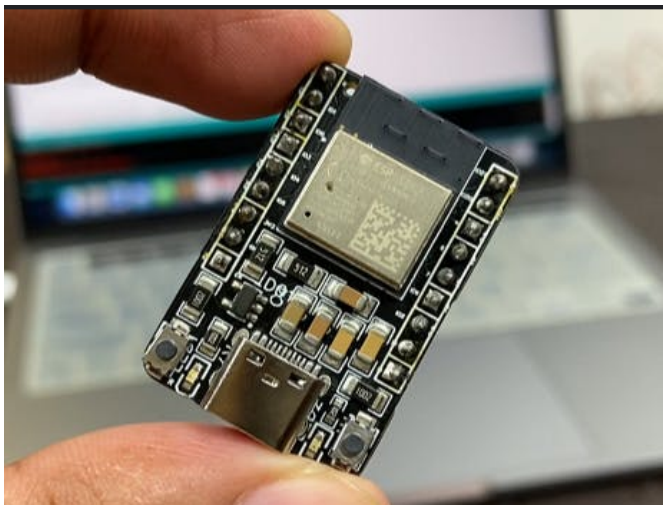
- Reliable, no battery concerns
- Easier software integration
- Drawback: requires cables, less placement flexibility

Wireless Sensors

- Flexible, easy to reposition
- Need power optimization
- Can communicate via Wi-Fi or LoRa

ESP32 Sensor Node

- Open-source, programmable via Arduino IDE / MicroPython
- Equipped with:
  - Wi-Fi, deep sleep mode for power saving
  - Additional sensor support (CO, motion)
- Sends data to InfluxDB
- 3D printed case for protection



ESP32

Cutie 3D

ESP32 v2: Constant Power Supply

- No battery – powered directly from wall outlet
- Multiple nodes in series along wall
- Improved stability and reduced maintenance
- Compact size (2-3 cm)

Sensor Alternatives

- Pre-built wireless sensors
- Pros:
  - Tested, high accuracy, often Python APIs
- Cons:
  - Closed-source, more expensive, harder to modify

Compatible Sensors with Python

- **DHT22 (with ESP32)** – temperature and humidity
- **BME280** – temperature, humidity, atmospheric pressure
- **AM2320** – simple I2C sensor
- **Tibbo Modbus** – wired version, Python via pymodbus
- All can be integrated using `micropython`, `paho-mqtt`, `requests` or `pymodbus`

System Architecture

- Sensor nodes: ESP32 with temperature/humidity sensors
- Data transfer: Wi-Fi or LoRa
- Storage: InfluxDB
- Visualization: Grafana / custom app
- Extra possibilities: alerts, motion detection, CO integration

Conclusion

- **Ideal option:** ESP32 with constant power
  - Easy to maintain, scalable, cheap, open-source
- **Alternative 2:** Pre-built wireless sensors
  - Simple but less flexible and more expensive
- **Alternative 3:** Tibbo + Raspberry Pi gateway
  - Very reliable but harder to scale