Université de Lorraine

Comfort project

Aida Nauanova
Andrei Danila
Irina Samus
Sidorela Uku

Work plan

- Learn and experiment with the Arduino board.
- Research and document established standards for comfort levels in administrative buildings.
- Set up the Flask application and organize the project structure.
- Implement user roles and develop the admin panel for managing users.
- UI framework research and selection of tools.
- Mockup creation and design. Dependencies installation and setup.

1

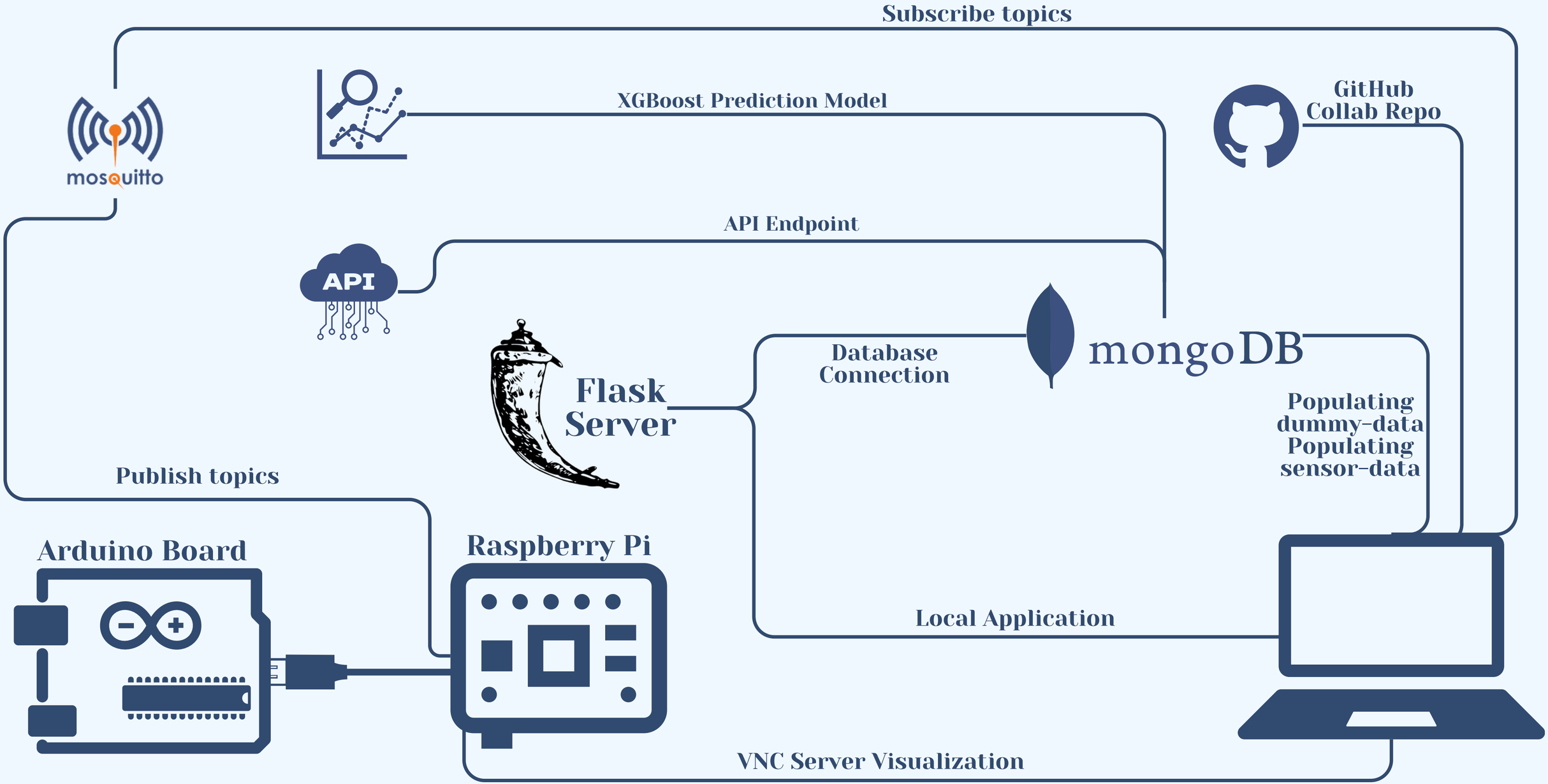
- Mapping functions to translate the analog code to desired metrics.
- Find and implement an open-cloud broker. Use the Mosquitto broker to publish the topics.
- Introduce continuous dummy-data generation for the rooms that don't have a sensor board.
- Implement the initial version of data visualization using Plotly Express.
- Developing guest and admin flow interfaces.
- Integration with backend, optimization and performance tuning.

2

- Develop code to subscribe to the topics and send this data to the cloud database.
- Testing the system to ensure seamless data flow from sensor to database and predict future comfort parameters.
- Implement the Analytic Hierarchy Process (AHP) for user preferences analysis.
- Develop API endpoints for fetching the latest data.
- UI/UX refinement and testing, implement and test solutions for a Responsive UI.
- Documentation of the work and code review.

3

Visual Overview



Sensor Layer

Light Sensor:

- **Metric:** Light Intensity
- **Unit:** Lux (lx)
- **Standard:** IESNA RP-1
- **Comfort:** 300-500 lux

Sound Sensor:

- **Metric:** Sound Intensity
- **Unit:** Decibels (dB)
- **Standard**: ANSI/ASA S12.60-2010/Part 1
- **Comfort:** 30-40 decibels (dBA)

Temperature Sensor:

- **Metric:** Temperature
- **Unit:** Celsius (°C)
- **Standard:** ASHRAE Standard 55-2020
- **Comfort:** 20-24°C

Humidity Sensor:

- **Metric:** Relative Humidity
- **Unit:** Percentage (%)
- **Standard:** ASHRAE Standard 55-2020
- **Comfort:** 40-60%

Communication Layer

Protocols

- **MQTT:** Real-time, lightweight IoT communication protocol.
- **HTTPS:** Secure data transmission over the **internet.**

Security Measures:

- **Encryption:** Protects data from unauthorized access.
- **Authentication:** Verifies device and user identity.
- **Access Control:** Limits access to authorized entities.

Data Layer

Database Management (MongoDB):

- Store data for flexibility.
- Leverage MongoDB's model for integration.

MongoDB API

- Querying and Filtering
- Data Manipulation

Data Enrichment:

- Enhance data with timestamps metadata.

Integration with Analytics Tools:

- Visualize processed IoT data using an open-source Plotly library

Application Layer

User Interface:

- Design intuitive and user-friendly interfaces for sensor data visualization, room rankings, and historical trends.

User Controls:

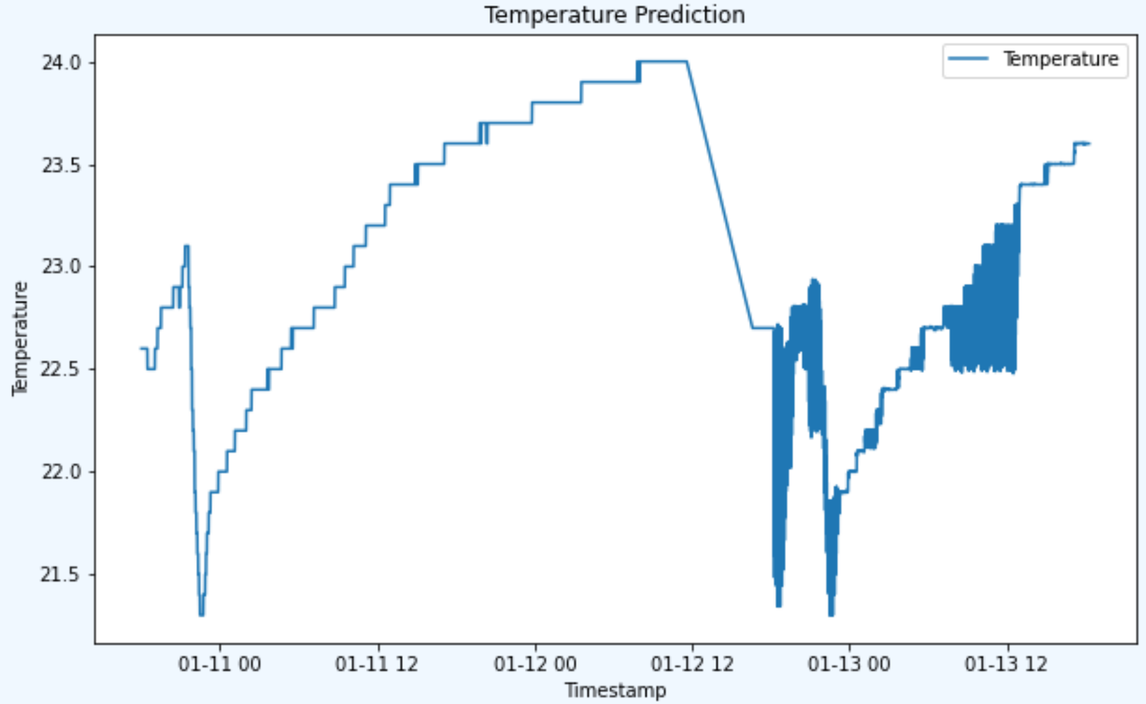
- Enable users to customize settings, view room rankings, and access historical data for a tailored experience.

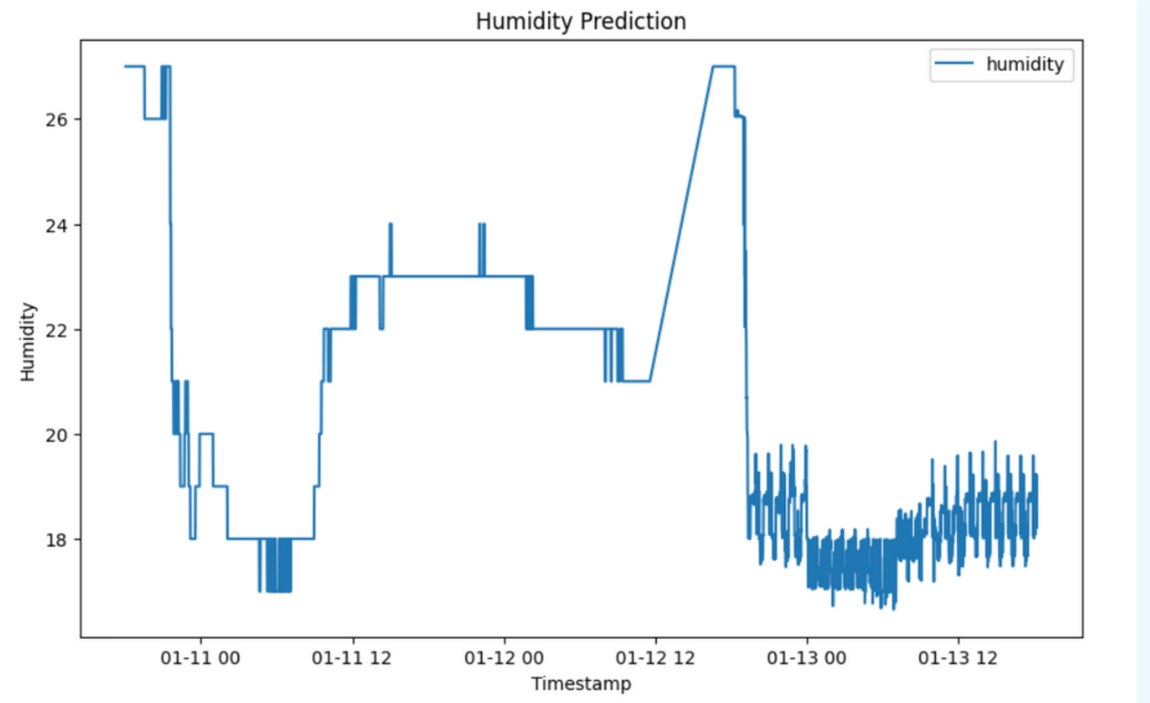
Reports and Visualizations:

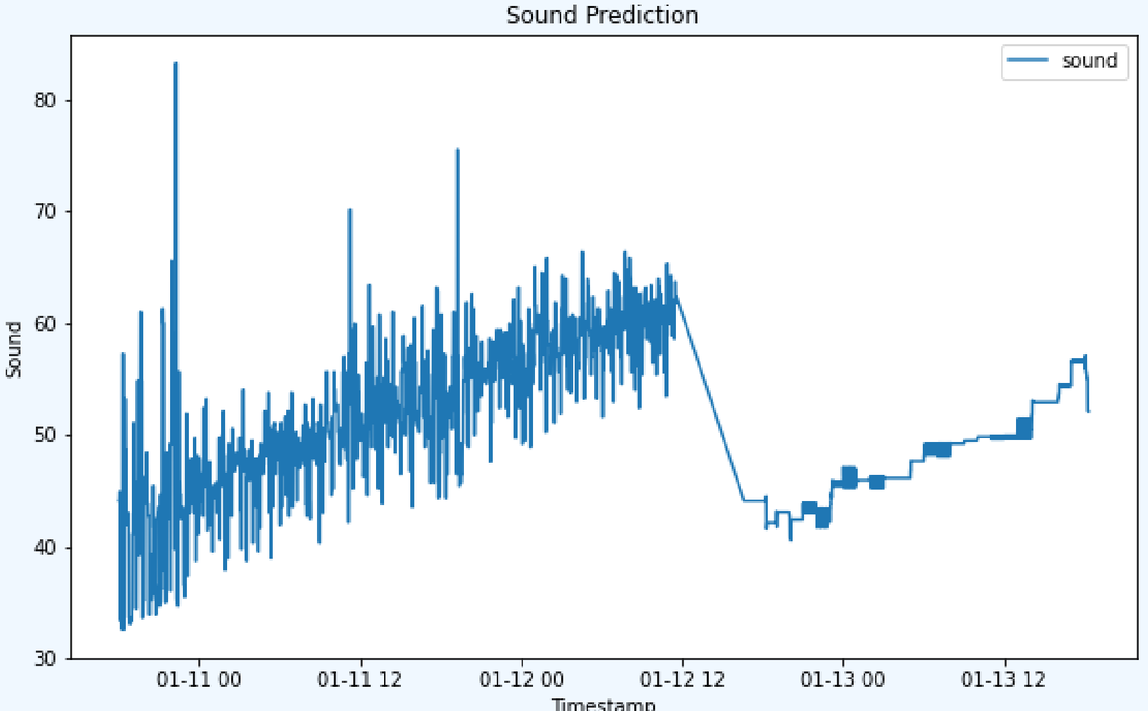
- Generate dynamic reports and visualizations using historical and real-time data for in-depth analysis.

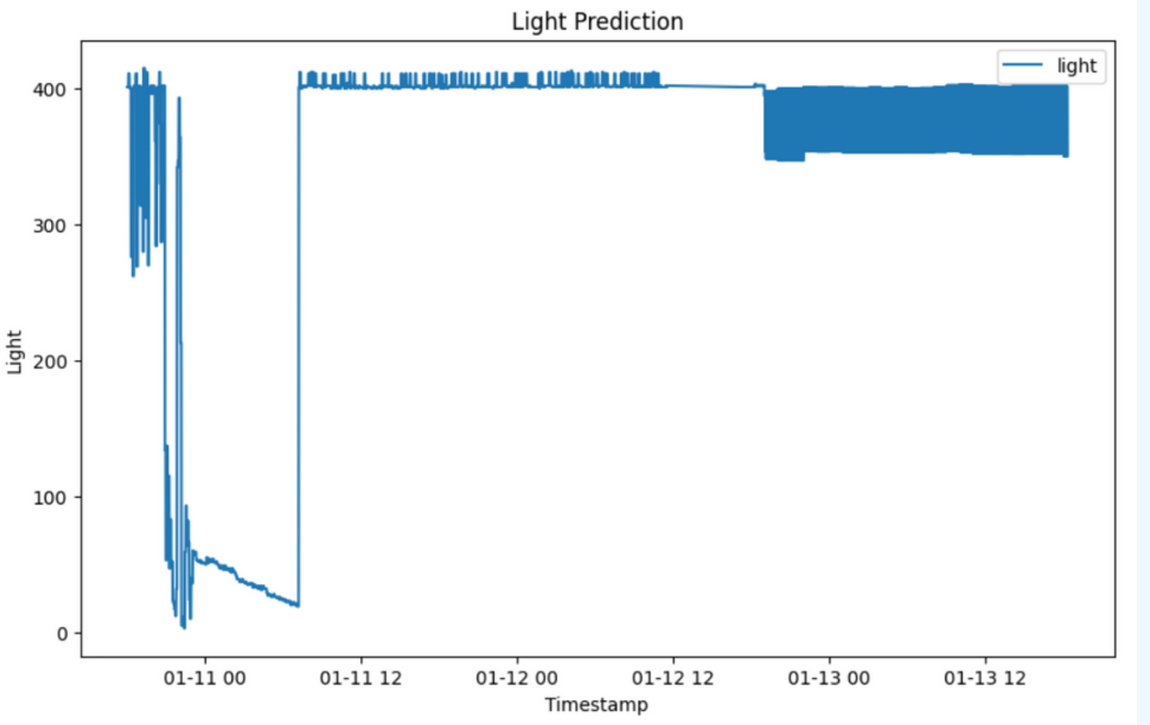
Forecast

- Fetching data from MongoDB
- Features Engineering using LagFeatures
- XGBoost model
- Evaluation of the model


Temperature Prediction
Temperature
24.0
23.5
23.0
22.5
22.0
21.5
Temperature
01-11 00
01-11 12
01-12 00
01-12 12
01-13 00
01-13 12
Timestamp


Humidity Prediction
humidity
26
24
22
20
18
Humidity
01-11 00
01-11 12
01-12 00
01-12 12
01-13 00
01-13 12
Timestamp


Sound Prediction
sound
80
70
60
50
40
30
Sound
01-11 00
01-11 12
01-12 00
01-12 12
01-13 00
01-13 12
Timestamp


Light Prediction
light
400
300
200
100
0
Light
01-11 00
01-11 12
01-12 00
01-12 12
01-13 00
01-13 12
Timestamp

Video Demo

Thank You