IoT Temperature & Humidity Monitoring System with AI-Based Comfort Classification

---

Team:
Youssef Mohamed Abdelshahid 2022/05890
Adham Mohamed 2022/04058
Ammar Alaa 2022/01210

1. Introduction

Indoor environmental conditions such as temperature and humidity play a critical role in human comfort, productivity, health, and equipment safety. Traditional monitoring systems often present raw numerical values that are difficult for non-technical users to interpret.

This project presents a **smart IoT-based environmental monitoring system** that not only measures temperature and humidity using a DHT22 sensor but also applies **embedded machine learning on an ESP32 microcontroller** to automatically classify the environmental comfort level. The system transmits both raw sensor data and AI predictions to a cloud dashboard for real-time visualization and long-term analysis.

The proposed solution is **low-cost, scalable, internet-accessible, and capable of local decision-making**, making it suitable for smart homes, offices, classrooms, greenhouses, and research environments.

---

2. Project Objectives

The main objectives of this project are:

1. To design a low-cost IoT system for monitoring temperature and humidity.
2. To perform **real-time environmental sensing** using an ESP32 and DHT22 sensor.
3. To implement **edge AI inference** using a lightweight embedded ML model.
4. To classify environmental comfort into predefined categories automatically.
5. To upload sensor data and AI predictions to a cloud platform.
6. To visualize real-time and historical data using an online dashboard.
7. To ensure the system works independently of continuous internet access for inference.

---

3. System Overview

The system consists of three main layers:

3.1 Sensing Layer

- DHT22 temperature and humidity sensor
- ESP32 microcontroller

3.2 Edge Intelligence Layer

- Embedded ML model stored as a C/C++ header file
- Real-time inference executed locally on the ESP32

3.3 Cloud & Visualization Layer

- Wi-Fi communication
- ThingSpeak cloud platform
- Real-time dashboards and data logging

---

4. Hardware Components

4.1 ESP32 Microcontroller

The ESP32 is a low-power, high-performance microcontroller with built-in Wi-Fi and Bluetooth. It is suitable for IoT applications due to:

- Integrated Wi-Fi module
- Sufficient RAM and flash for ML inference
- Low power consumption
- Wide GPIO support

4.2 DHT22 Sensor

The DHT22 sensor measures:

- Temperature (°C)
- Relative humidity (%)

Key features:

- Digital output
- Good accuracy for indoor monitoring
- Simple interface with ESP32

---

5. Software Components

5.1 Development Environment

- Arduino IDE
- ESP32 Board Support Package
- Adafruit DHT sensor libraries

5.2 Embedded ML Model

- Provided as a `.h` file
- Accepts temperature and humidity as input
- Outputs class probabilities
- Runs entirely on the ESP32 (Edge AI)

5.3 Cloud Platform

- ThingSpeak
- HTTP-based data upload
- Automatic chart generation
- Long-term data storage

---

6. System Architecture & Data Flow

1. The DHT22 sensor measures temperature and humidity.
2. The ESP32 reads the sensor values.
3. Sensor values are passed to the embedded ML model.
4. The model outputs probabilities for each comfort class.
5. The ESP32 selects the class with the highest probability.
6. Raw data and AI results are transmitted to ThingSpeak via Wi-Fi.
7. The dashboard updates graphs and widgets in real time.

---

7. AI Comfort Classification

7.1 Model Inputs

- Temperature (°C)
- Humidity (%)

7.2 Model Outputs

The ML model produces three probability values corresponding to:

Class Index	Label
0	Comfortable
1	Uncomfortable
2	Warm

The final predicted class is selected using **maximum probability**.

7.3 Why Edge AI?

- Eliminates dependency on cloud inference
- Reduces latency
- Improves privacy
- Ensures system functionality even without internet

---

8. ThingSpeak Channel Design

8.1 Field Mapping

Field	Data
Field 1	Temperature (°C)
Field 2	Humidity (%)
Field 3	Comfort Class Index
Field 4	Probability – Comfortable
Field 5	Probability – Uncomfortable
Field 6	Probability – Warm

8.2 Dashboard Features

- Live numeric indicators
- Line graphs for temperature and humidity
- Step plot for comfort classification
- Probability visualization for AI confidence
- Public or private access

---

9. Implementation Details

9.1 Sensor Reading

- Data is read periodically from DHT22
- Invalid readings are discarded

9.2 ML Inference

- Model is executed locally using `score(input, output)`
- Inference time is minimal
- No external dependencies

9.3 Cloud Communication

- HTTP GET requests
- Minimum 15-second update interval
- Automatic reconnection handling

---

10. Applications

This system can be applied in:

- Smart home climate monitoring
- Greenhouse and agriculture monitoring
- Server room environment control
- Classroom and office comfort tracking
- Sleep environment optimization
- Environmental research and data logging

---

11. Advantages of the System

- Low hardware cost
- Real-time monitoring
- Embedded AI decision-making
- Cloud-based visualization
- Scalable and extensible
- Internet-independent inference

---

12. Limitations

- DHT22 is not suitable for extreme industrial environments
- ThingSpeak dashboard customization is limited
- Comfort classification depends on training data quality

---

13. Future Enhancements

- Add OLED/LCD display for local feedback
- Integrate mobile app dashboard
- Add alert notifications (email/SMS)
- Support additional sensors (CO_2, light, air quality)
- Retrain ML model with user-specific comfort preferences
- Implement deep sleep for battery-powered operation

---

14. Conclusion

This project successfully demonstrates an **IoT-based environmental monitoring system enhanced with embedded artificial intelligence**. By combining real-time sensing, edge AI inference, and cloud visualization, the system provides both raw environmental data and meaningful comfort interpretation. The design is practical, scalable, and suitable for real-world deployment, making it an effective solution for smart environments and IoT-based research applications.