Abdoul Sow

NWIT 101

**WRITTEN REPORT**:

**Raspberry Pi–Based Temperature and Humidity Monitoring System**

Project Member:

Abdoul Sow

1. Introduction

Environmental monitoring plays an important role in many real-world applications such as smart homes, agriculture, and server room management. This project focuses on building a simple and low-cost temperature and humidity monitoring system using a Raspberry Pi and a DHT22 sensor. The system measures environmental data in real time and displays the results using a Python program.

2. Project Purpose

The purpose of this project is to design and implement a Raspberry Pi–based system that can accurately measure temperature and humidity. This project demonstrates how to use GPIO pins, sensors, and Python programming to interact with hardware components. It also provides hands-on experience with embedded systems and Internet of Things (IoT) concepts.

3. Project Product Features

The main features of the project include:

• Real-time temperature measurement in Celsius and Fahrenheit

• Real-time humidity measurement in percentage (%)

• Use of Raspberry Pi GPIO pins for digital sensor communication

• Python-based program for data processing and display

• Low-cost and easy-to-recreate design

4. Project Materials and Tools

Hardware Components

• Raspberry Pi 4

• DHT22 temperature and humidity sensor

• Male-to-female jumper wires

• Power supply for Raspberry Pi

• Monitor, keyboard, and mouse

Software and Tools

• Raspberry Pi OS

• Python 3

• Thonny IDE

• Adafruit DHT library

5. Project Development History

The project development was completed in several steps to ensure proper functionality and accuracy.

Step 1: Project Planning

The project began by researching sensors capable of measuring both temperature and humidity. The DHT22 sensor was selected due to its reliability, accuracy, and compatibility with the Raspberry Pi.

Step 2: Hardware Setup

The DHT22 sensor was connected to the Raspberry Pi using jumper wires. The sensor’s power pin was connected to the 3.3V pin, the ground pin was connected to a GND pin, and the data pin was connected to GPIO4. This wiring configuration ensured safe voltage levels and reliable data transmission.

A close up of a wire

Description automatically generatedA transparent box with wires and a fan

Description automatically generatedA computer device with wires

Description automatically generated with medium confidence

Step 3: Software Installation

Raspberry Pi OS was installed on the device, and Python 3 was used as the programming language. The Adafruit DHT library was installed to enable communication between the Raspberry Pi and the DHT22 sensor.

Step 4: Programming and Testing

A Python program was written using the Thonny IDE to read temperature and humidity values from the sensor. The program continuously reads data and displays the results on the screen. Multiple tests were conducted to verify stable and accurate sensor readings.

A computer screen with text on it

Description automatically generatedA computer screen with text on it

Description automatically generated

6. Project Prototype / Final Product

The final prototype successfully displays real-time temperature and humidity values on the Raspberry Pi terminal. The sensor readings were consistent and accurate, confirming that the system functions as intended. Screenshots of the wiring setup, code, and output were captured as evidence of successful operation.

A screen with text on it

Description automatically generated

7. Results and Analysis

During testing, the temperature readings averaged around 24.1°C to 25.5°C, while humidity readings were approximately 28% to 29%. Minor fluctuations in readings were observed, which is normal due to changes in the surrounding environment. Overall, the results demonstrate that the DHT22 sensor provides reliable environmental data.

8. Lessons Learned

Through this project, I learned how to properly use Raspberry Pi GPIO pins, safely connect sensors using correct voltage levels, and write Python programs to communicate with hardware devices. I also gained experience troubleshooting wiring and software issues and learned how to test and validate sensor data in a real-world application.

9. Future Improvements

Future improvements to this project may include adding data logging to store sensor readings, creating a web-based dashboard to display data remotely, or setting up alerts when temperature or humidity values exceed certain thresholds. Additional sensors such as air quality or motion sensors could also be integrated.

10. Conclusion

This project successfully demonstrated the use of a Raspberry Pi and a DHT22 sensor to build a temperature and humidity monitoring system. The project achieved its goals by accurately collecting and displaying environmental data using Python. This system serves as a strong foundation for more advanced IoT and embedded system applications.

References:

Adafruit Industries. (2023). DHT22 (AM2302) temperature-humidity sensor guide. <https://learn.adafruit.com/dht>

Raspberry Pi Foundation. (2024). Raspberry Pi documentation. <https://www.raspberrypi.com/documentation/>

Python Software Foundation. (2023). Python 3 documentation. <https://docs.python.org/3/>