ROOM TEMPERATURE MONITORING AND EQUALIZATION SYSTEM USING SENSORS AND FAN

Revito Pahlevi[1], Seliwati[2]
Politeknik Piksi Ganesha[1,2]
email : revitopahle12@gmail.com , seliwati93@gmail.com

Abstract: The temperature in a room can fluctuate, becoming either hot or cold depending on the heat generated by objects within the room. In a workspace, various hardware devices often emit heat suddenly, surpassing the standard room temperature. Due to the heavy workload of these devices, their performance can slow down as the temperature rises, making it uncomfortable for workers and difficult for them to concentrate. To address this, an automatic temperature equalization system needs to be implemented, monitored using a DHT11 sensor. The system operates using an Arduino Uno, DHT11 Digital Temperature and Humidity Sensor, ESP-01, and a fan. This system, integrated with ESP-01, reads the DHT11 sensor to measure room humidity and temperature and sends the data to a website for mobile monitoring. The fan is used to distribute the air temperature for equalization. Testing results show that the system successfully balances the room temperature automatically.

Keywords: Temperature equalization, DHT11 Sensor, Fan, ESP-01, Internet of Things.

Introduction

Maintaining a stable and evenly distributed room temperature is crucial, especially in workspaces filled with electronic devices such as computers, servers, or other hardware. Excessive heat in such environments can cause devices to operate below optimal performance, increase the risk of hardware damage, reduce operational efficiency, and create an uncomfortable working environment. Therefore, a system capable of automatically monitoring and regulating room temperature is needed to ensure that the temperature remains within a safe and comfortable range. This temperature monitoring and equalization system is designed to address these challenges by utilizing sensor and microcontroller technology. The system uses two DHT11 temperature sensors placed at different locations, with the Arduino Uno serving as the central controller that processes the sensor data, displays it on a 16x2 LCD, and controls cooling systems like fans to maintain a stable room temperature. Additionally, with the integration of ESP-01, the temperature data can be monitored in real-time via a web platform, allowing users to remotely control the room's temperature conditions and receive notifications. The system also automatically equalizes the temperature based on a 3-degree Celsius difference between sensors or when one of the sensors exceeds 30 degrees Celsius. The ideal temperature range for a workspace is between 22-27 degrees Celsius.

E-Mail: pahlevirevito@gmail.com

Literature Review

In daily life, individuals require a comfortable space to concentrate on the tasks they are working on. The surrounding environment is one of the factors that influences comfort in a room during activities. Temperature and humidity in the room can affect work effectiveness. Working in an overly hot room can reduce physical performance and cause fatigue, while excessively cold temperatures can decrease motor flexibility, leading to physical stiffness(Aulia et al., 2021).

In unstable temperature conditions, the performance of hardware can be disrupted. One of the causes is overheating, which can lead to a decline in hardware performance. Additionally, high humidity levels in the room can have negative effects on the lifespan of the components within that workspace(Kris et al., 2022). Kevin Ashton, the originator of the term "Internet of Things," defines 'Internet of Things' in his e-book titled "Making Sense of IoT" as a collection of sensors connected to the internet that operate similarly to the internet itself, forming continuous open connections and sharing data freely. This opens up opportunities for unexpected applications and allows computers to understand their surrounding environments, thus becoming an integrated part of human life(Erwan Eko Prasetiyo, 2017).

Laravel is a PHP-based framework used to assist in website development. It offers numerous modern features that greatly support developers in application creation. Some of these features include Bundles, Eloquent ORM (Object-Relational Mapping), Query Builder, Resource Controller, Blade, Migration, Middleware, and Automatic Pagination. Additionally, Laravel has several advantages, such as the use of the Artisan Command Line Interface (CLI), integration with the PHP Composer package manager, and a more concise, easily understandable, and expressive code writing style(Indah Melyani et al., 2023).

Arduino is a microcontroller board equipped with an Integrated Circuit (IC) chip from Atmel, used for programming control systems. Arduino has several pins used for input and output: 14 digital pins (D0-D13), which can be used for digital input or output, including PWM pins; 6 analog pins (A0-A5), which are used for reading analog signals; power pins such as 5V, 3.3V, and GND for powering external components; and special pins like TX and RX for serial communication, as well as a RESET pin to reset the system(Nugroho & Djaksana, 2022).

The ESP8266 wireless module is a low-cost Wi-Fi module that fully supports TCP/IP usage. This module is produced by Espressif, a company based in China. In 2014, AI-Thinker, a third-party manufacturer, released the ESP-01 module, which uses AT Commands for its configuration. With its affordable price, low power consumption, and small size, many developers became interested in further developing this module. In October 2014, Espressif launched a software development kit (SDK) that allowed more developers to work on this module(Yuliansyah, 2016).

An LCD is a flat-panel display that utilizes the light-modulating properties of liquid crystals, which do not emit light directly. LCDs can display a variety of images, as seen on computers, or fixed images with low information content, such as preset words and numbers on a digital clock. Both types use the same basic technology, but varied images are made up of many small pixels, while fixed displays feature larger elements(Patil1, 2024).

The DHT-11 is a temperature and humidity sensor that offers high output stability and long-term reliability. This sensor measures environmental temperature by generating a digital signal on the data pin, eliminating the need for other analog input signals in its operation. Below are the specifications of the DHT-11 sensor(Anif & Hari Prasetyo, n.d.).

The relay module operates at a low voltage of 5 volts direct current (VDC). This relay can be used to switch electricity with a maximum voltage of 250 volts alternating current (VAC) and a current of up to 10 amperes (A). If the current exceeds 10 A, for example, up to 15 A, then the maximum voltage allowed to flow through the relay is 125 VAC. To control this relay, an operating voltage of 5 VDC with a current ranging from 15 to 20 milliamperes (mA) is required(Yulianto, 2023).

Methodology

To simplify the design process, an overview of the entire system is provided, as shown in Figure 1 below.

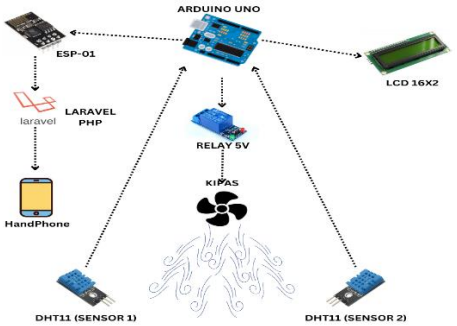

Figure 1. System Overview

The initial step in system design begins with creating a block diagram to monitor room temperature using the DHT11 sensor. The goal is to simplify the device design process. The complete system block diagram is shown in Figure 2.

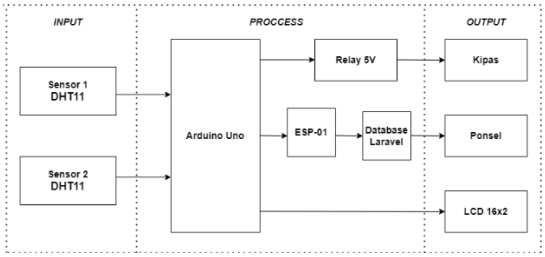

Figure 2. Overall System Block Diagram

Two DHT11 sensors are used to measure the room temperature, while the Arduino Uno processes the data from these sensors to display it on an LCD and send it to a mobile phone via the ESP-01. The Arduino Uno also controls the relay module to turn the fan on or off. The LCD and mobile phone serve as the monitoring interfaces for this system. This section provides a detailed explanation, starting from the hardware requirements to the software used in the development of this project, as summarized in Table 1.

Table 1. System Requirements Description

Hardware Requirements		
No.	**Item**	**Description**
1.	Arduino Uno	328P
2.	Sensor DHT11	5V
3.	LCD	16x2 *green*
4.	Modul relay	5V 1 *Channel*
6.	Fan	12V
7.	ESP-01	8226
8.	Mobile Phone	Android
Software Requirements		
1.	Operating System	Windows 10
2.	*Compiler*	Arduino IDE
3	Database	Laravel

In this system, the hardware is housed in an acrylic casing measuring 26x22x9 cm. The 3 mm thick acrylic is sufficient to accommodate all hardware components. Figure 3 below shows the model of the system casing.

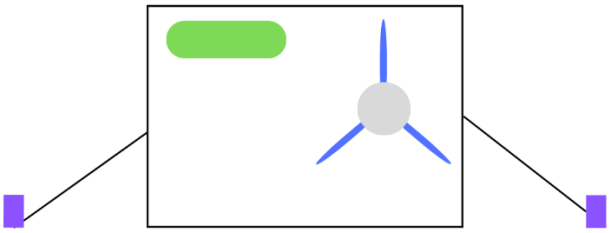

Figure 3. System Casing Design

This casing is designed to allow for easy relocation of the hardware without the need to remove components one by one. The LCD is positioned in the upper left corner of the casing, while the fan blades are located on the right side. Two sensors are placed on the front left and right sides of the casing at predetermined distances. More detailed information regarding the placement of the hardware can be found in Figure 4, while the explanations and functions of each component are available in Table 2.

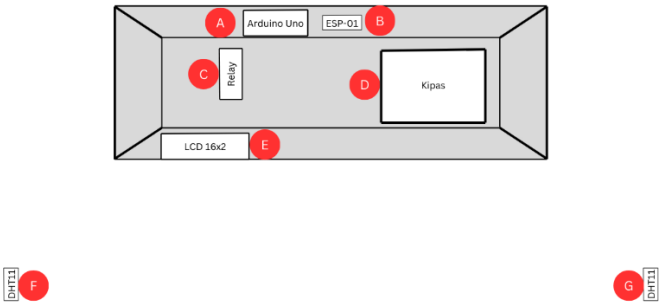

Figure 4. Casing Design Description

Table 2. Casing Design Description

Bagian	Keterangan
A	Arduino Uno
B	ESP-01
C	Relay 5V
D	Kipas 12V
E	LCD 16x2
F	Sensor satu DHT11
G	Sensor dua DHT11

Below is Figure 5, which presents the overall system flow diagram.

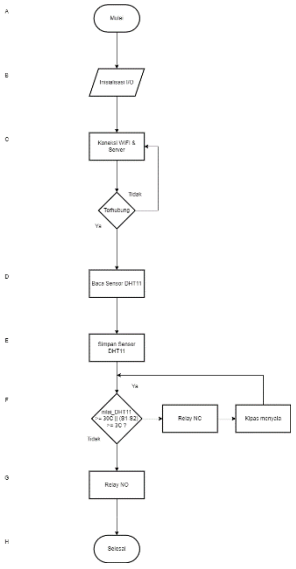

Figure 5. Overall System Flow Diagram

Figure 5 above shows the overall system flow diagram. Further explanations can be found in Table 3 below.

Table 3. Casing Design Description.

Part	Information
A	The initiation of the system process.
B	Initialization of input and output.
C	Process of connecting WiFi and the server. On the ESP-01, if the WiFi and server connections are successful, it will proceed to the next process; if not, it will return to the WiFi and server connection process..
D	Process of reading the DHT11 sensor.
E	Process of storing DHT11 sensor data.
F	If the temperature exceeds 30 degrees Celsius or the difference is 3 degrees Celsius, the relay will be normally closed (NC). Otherwise, the relay will be normally open (NO), and the process will continue to the next step.
G	Relay NO
H	Process completed.

Results & Discussion

In this system, the DHT11 sensor plays a pivotal role in equalizing room temperature. Tests were conducted to assess the duration required by the system to achieve uniform temperature distribution. This calculation is critical for ensuring the system functions optimally. The tests were performed in a room where the temperature had been pre-set.

Table 4. DHT11 Sensor Testing

Testing	Expected Relation	Temperature before equalization	Temperature after equalization	Equalization Time
Trial – 1	Data uploaded to ESP-01	T1-22.30° T2-25.30°	T1-24.10° T2-24.10°	00:13
Trial – 2	Data uploaded to ESP-01	T1-27.10° T2-24.10°	T1-25.30° T2-25.30°	00:26
Trial – 3	Data uploaded to ESP-01	T1-24.80° T2-27.80°	T1-25.90° T2-25.90°	00:45
Trial – 4	Data uploaded to ESP-01	T1-26.20° T2-29.20°	T1-27.60° T2-27.60°	00:23
Trial – 5	Data uploaded to ESP-01	T1-25.10° T2-22.10°	T1-24.70° T2-24.70°	00:19
Trial – 6	Data uploaded to ESP-01	T1-27.00° T2-30.00°	T1-29.00° T2-29.00°	01:24
Trial – 7	Data uploaded to ESP-01	T1-30.00° T2-27.00°	T1-29.20° T2-29.20°	01.06
Trial – 8	Data uploaded to ESP-01	T1-26.40° T2-29.40°	T1-28.10° T2-28.10°	00:47
Trial – 9	Data uploaded to ESP-01	T1-28.90° T2-25.90°	T1-27.90° T2-27.90°	00:58
Trial – 10	Data uploaded to ESP-01	T1-25.10° T2-28.10°	T1-29.00° T2-29.00°	00:55

In this test, the fan will turn on when the temperature difference between the two sensors reaches 3 degrees Celsius. The method used involves placing two DHT11 sensors at different locations. One sensor will be exposed to heat from nearby operating hardware, and the equalization time is measured when the fan starts functioning. The data generated by the sensors is in the form of digital signals, which are then processed by the Arduino Uno to produce temperature values in degrees Celsius (°C).

The Arduino Uno, integrated with the ESP-01 module, enables the device to connect to the internet via a Wi-Fi network. In this system, the internet connection is used to send data from the hardware to a server. Testing was conducted to connect the Arduino Uno to the available Wi-Fi network. The ESP-01 module can connect to a Wi-Fi network by configuring the SSID and network password, as well as connecting to the server by configuring the server's IP address and port. In this test, the ESP-01 module will connect to the Wi-Fi hotspot used by the user. Figure 6 shows the serial monitor display of the ESP-01 module while it is connecting to the Wi-Fi network and server.

```
Attempting to connect to SSID: MYRepublic

Connected to WiFi network.
SSID: MYRepublic
BSSID: 6A:13:8D:38:19:E0
IP Address: 192.168.124.129
Signal strength (RSSI): -41 dBm
Connected to server
HTTP/1.1 200 OK
Date: Tue, 08 Oct 2024 15:06:15 GMT
Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.0.30
X-Powered-By: PHP/8.0.30
Content-Length: 2
Content-Type: text/html; charset=UTF-8
```

Figure 6. ESP-01 Testing

The system operation will begin once the ESP-01 successfully connects to the Wi-Fi network. After the ESP-01 establishes the internet connection, the system will then connect to the Laravel database. Once the connection is successful, the next step is to start reading the sensors and activate the equalization system. The sensor readings will be displayed on the 16x2 LCD screen and sent to the server for viewing on the mobile phone, as shown in Figure 7.

Figure 7. LCD and Mobile Testing for Displaying Temperature Data

Temperature testing was carried out to ensure that the equalization system functions according to the specified temperature difference and that the temperatures become balanced, with only a small difference between the two sensors. The equalization process activates the

fan to distribute air, controlled by the relay module. This test was repeated several times to ensure optimal sensor performance.

In a room measuring 2x2x2.5 meters, the close placement of the sensors to the equipment allows for accurate air temperature readings. The hardware is positioned in one corner of the room, while the control system is located in the center between the equipment. When the equalization system is active, notifications will continuously appear, indicating that the fan is running. Once the temperature is balanced, the notifications will stop. Here is Figure 8, which shows the notification when the system is activated.

Figure 8. Notification display on mobile and matching data with the LCD

Based on the test results, it can be concluded that the temperature data obtained from the DHT11 sensor can be displayed on the 16x2 LCD screen as well as on the mobile phone. The automatic equalization system functions properly, operating based on the air temperature detected by the DHT11 sensor. The system is also capable of sending automatic notifications successfully.

My field notes during data collection, combined with my personal experience throughout the process, had a significant impact during the early stages of the design process. One of the research participants, who is also the author of this paper, stated that;

> *I would like to express my heartfelt gratitude to everyone who has played a role in supporting and inspiring us. Thank you to my family for their unwavering support, to my research colleagues for sharing valuable insights, to our institution for providing the necessary facilities, to the participants who took part, and to those who offered financial support. All of these contributions have been crucial to the success of this research.*

References

Anif, M., & Hari Prasetyo, B. (n.d.). *Kendali dan Monitoring Ruang Server dengan Sensor Suhu DHT-11, Gas MQ-2 serta Notifikasi SMS*. https://seminar.iaii.or.id/index.php/SISFOTEK/article/view/113

Aulia, R., Fauzan, R. A., & Lubis, I. (2021). Pengendalian Suhu Ruangan Menggunakan Menggunakan FAN dan DHT11 Berbasis Arduino. *CESS (Journal of Computer Engineering, System and Science)*, *6*(1), 30. https://doi.org/10.24114/cess.v6i1.21113

Erwan Eko Prasetiyo. (2017). Aplikasi Internet Of Things (Iot) Untuk Pemantauan Dan Pengendalian Beban Listrik Di Ruangan. *Jurnal Teknika STTKD*, *4*(2), 28–39. https://jurnal.sttkd.ac.id/index.php/ts/article/view/51

Indah Melyani, R., Rosita, R., & Aji, S. (2023). Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel dengan Metode Agile Software Development. *Jurnal Sistem Informasi Akuntansi (JASIKA)*, *3*(1), 31–36. https://doi.org/10.31294/jasika.v3i01.2195

Kris, N. F., Mandar, G., & Hamid, M. (2022). Rancang Bangun Sistem Pengukuran Suhu dan Kelembaban Ruang Server Berbasis Microcontroller. *Jurnal Teknik Informatika (J-Tifa)*, *5*(2), 35–40. https://doi.org/10.52046/j-tifa.v5i2.1228

Nugroho, B. A., & Djaksana, Y. M. (2022). Implementasi Mikrokontroler Arduino Uno dan Multi Sensor Pada Tempat Sampah. *Jurnal Scientia Sacra: Jurnal Sains*, *2*(4), 70–77. http://pijarpemikiran.com/index.php/Scientia

Patil1, V. M. (2024). Liquid Temperature Measurement Using Arduino. *International Scientific Journal of Engineering and Management*, *03*(03), 1–9. https://doi.org/10.55041/isjem01448

Yuliansyah, H. (2016). Uji Kinerja Pengiriman Data Secara Wireless Menggunakan Modul ESP8266 Berbasis Rest Architecture. *Jurnal Rekayasa Dan Teknologi Elektro*, *10*(2 (Mei 2016)), 68–77. https://doi.org/10.23960/ELC.V10N2.217

Yulianto, Y. (2023). Relay Driver Based on Arduino UNO to Bridge the Gap of The Digital Output Voltage of The Node MCU ESP32. *Engineering, MAthematics and Computer Science Journal (EMACS)*, *5*(3), 129–135. https://doi.org/10.21512/emacsjournal.v5i3.9697