
SISTEM KONTROL SUHU RUANGAN SERVER BERBASIS INTERNET OF THINGS (IOT) MENGGUNAKAN MIKROKONTROLER WEMOS DAN SENSOR SUHU DHT11 DI PT. XYZ

Ade Sumaedi

Fakultas Ilmu Komputer, Program Studi Sistem Komputer

Universitas Pamulang

Email: adesumaedi10093@unpam.ac.id

ABSTRAK

Di tengah perkembangan teknologi informasi yang pesat, pengelolaan suhu ruang server telah menjadi faktor penting untuk memastikan performa dan reliabilitas perangkat keras. Penelitian ini membahas pengembangan sistem kontrol suhu berbasis Internet of Things (IoT) yang menggabungkan mikrokontroler Wemos dengan sensor suhu DHT11 di PT. XYZ. Sistem ini dirancang untuk meningkatkan efisiensi pengendalian suhu ruang server dengan memanfaatkan teknologi terkini, yang memungkinkan pemantauan serta pengaturan suhu secara otomatis dan dalam waktu nyata. Metodologi pengembangan sistem melibatkan integrasi perangkat keras dan perangkat lunak untuk menciptakan solusi yang tidak hanya responsif tetapi juga hemat energi. Mikrokontroler Wemos berfungsi sebagai pusat kendali yang mengelola data dari sensor suhu DHT11, yang kemudian diproses untuk mengatur sistem pendingin secara otomatis. Dengan sistem ini, pengguna dapat memantau suhu ruang server dari jarak jauh melalui antarmuka berbasis web, memungkinkan tindakan cepat jika terjadi fluktuasi suhu yang signifikan. Hasil evaluasi sistem menunjukkan adanya peningkatan yang signifikan dalam pengelolaan suhu ruangan server. Sistem ini tidak hanya memperbaiki kestabilan suhu tetapi juga berkontribusi pada efisiensi energi yang lebih baik dan penghematan biaya operasional. Penggunaan teknologi IoT dalam sistem kontrol suhu ini terbukti efektif dalam mengurangi ketergantungan pada intervensi manual serta meminimalisir risiko kerusakan perangkat keras akibat suhu yang tidak terkontrol. Dengan demikian, implementasi sistem kontrol suhu berbasis IoT ini memberikan kontribusi penting dalam mengoptimalkan operasional ruang server di PT. XYZ.

Kata kunci: Sistem Pengendalian Suhu, Internet of Things, Mikrokontroler Wemos, Sensor Suhu DHT11.

ABSTRACT

In the midst of the rapid development of information technology, managing server room temperature has become an important factor in ensuring hardware performance and reliability. This research discusses the development of an Internet of Things (IoT) based temperature control system that combines a Wemos microcontroller with a DHT11 temperature sensor at PT. XYZ. This system is designed to increase the efficiency of server room temperature control by utilizing the latest technology, which allows temperature monitoring and regulation automatically and in real time. The system development methodology involves integrating hardware and software to create a solution that is not only responsive but also energy efficient. The Wemos microcontroller functions as a control center that manages data from the DHT11 temperature sensor, which is then processed to regulate the cooling system automatically. With this system, users can monitor server room temperatures remotely via a web-based interface, enabling quick action if significant temperature fluctuations occur. The system evaluation results show a significant improvement in server room temperature management. This system not only improves temperature stability but also contributes to better energy efficiency and operational cost savings. The use of IoT technology in this temperature control system has proven effective in reducing dependence on manual intervention and minimizing the risk of hardware damage due to uncontrolled temperatures. Thus, the implementation of this IoT-based temperature control system makes an important contribution in optimizing server room operations at PT. XYZ.

Keywords: *Temperature Control System, Internet of Things, Wemos Microcontroller, DHT11 Temperature Sensor.*

1. PENDAHULUAN

Dalam lingkungan server, pengelolaan suhu yang efektif adalah salah satu aspek krusial untuk memastikan kinerja optimal dan keandalan perangkat keras. Ketika suhu tidak dijaga dengan baik, perangkat keras server menghadapi risiko serius, termasuk kemungkinan terjadinya overheating yang dapat menyebabkan kerusakan serius atau penurunan signifikan dalam kinerja sistem. Suhu yang tinggi dapat mempercepat degradasi komponen elektronik, menyebabkan penurunan umur komponen, atau bahkan memicu kegagalan sistem yang tidak terduga. Oleh karena itu, ruang server harus dilengkapi dengan sistem pendingin yang tidak hanya efektif tetapi juga efisien untuk menjaga suhu dalam batas aman dan stabil.

Sistem pendinginan tradisional, meskipun efektif dalam banyak kasus, seringkali memerlukan pengawasan dan pemeliharaan manual yang intensif. Dengan meningkatnya kompleksitas dan kebutuhan ruang server modern, metode tradisional ini mungkin tidak lagi memadai untuk menangani tantangan yang ada. Oleh karena itu, teknologi mutakhir, seperti solusi berbasis Internet of Things (IoT), telah muncul sebagai alternatif yang menjanjikan dalam pengelolaan fasilitas server. IoT menawarkan pendekatan yang lebih canggih dengan memanfaatkan jaringan internet untuk memantau, mengendalikan, dan mengoptimalkan lingkungan server secara otomatis [1].

Internet of Things, atau IoT, adalah konsep di mana perangkat fisik dapat terhubung dan berkomunikasi melalui jaringan internet. Teknologi ini memungkinkan berbagai perangkat untuk saling bertukar data dan informasi dalam waktu nyata. Dalam konteks pengelolaan lingkungan server, IoT memungkinkan integrasi sensor suhu dan perangkat kontrol yang dapat memantau kondisi lingkungan secara terus-menerus. Sensor suhu yang terhubung ke jaringan dapat mengumpulkan data suhu dari berbagai titik dalam ruang server dan mengirimkan informasi tersebut ke sistem manajemen pusat. Data ini kemudian dianalisis untuk menentukan apakah tindakan korektif diperlukan.

Salah satu keuntungan utama dari sistem pendinginan berbasis IoT adalah kemampuannya untuk memberikan pemantauan suhu secara real-time. Dengan memanfaatkan sensor suhu yang terhubung, administrator sistem dapat mengakses data suhu secara langsung melalui antarmuka berbasis web atau aplikasi mobile. Ini memungkinkan mereka untuk segera mendeteksi perubahan suhu yang abnormal dan mengambil tindakan pencegahan sebelum masalah menjadi lebih serius. Misalnya, jika sensor mendeteksi suhu yang melebihi batas yang telah ditentukan, sistem dapat secara otomatis mengaktifkan unit pendingin tambahan atau mengirimkan peringatan kepada tim pemeliharaan untuk segera menanggapi situasi tersebut.

Selain itu, sistem berbasis IoT dapat dilengkapi dengan algoritma analitik yang lebih canggih untuk memproses data suhu yang dikumpulkan. Algoritma ini dapat membantu dalam identifikasi pola suhu yang tidak biasa atau potensi masalah sebelum mereka berkembang menjadi kegagalan sistem. Misalnya, analitik dapat digunakan untuk mengidentifikasi tren suhu yang mungkin menunjukkan adanya masalah pada sistem pendinginan atau potensi masalah dengan aliran udara. Dengan informasi ini, langkah-langkah pencegahan atau perbaikan dapat dilakukan lebih awal, mengurangi kemungkinan downtime dan kerusakan perangkat keras.

Penerapan teknologi IoT juga memungkinkan pengendalian suhu yang lebih fleksibel dan adaptif. Dengan menggunakan sistem otomatisasi, perangkat pendingin dapat dikendalikan secara dinamis berdasarkan data suhu yang diterima. Misalnya, jika suhu ruang server meningkat pada waktu-waktu tertentu atau di area tertentu, sistem dapat mengaktifkan pendingin hanya pada zona yang membutuhkan pendinginan tambahan. Ini tidak hanya meningkatkan efisiensi energi tetapi juga mengurangi biaya operasional terkait dengan pendinginan yang tidak perlu. Arduino Uno dapat diprogram untuk mendeteksi kebocoran gas dan memberikan peringatan dini kepada pengguna [2].

Namun, implementasi sistem berbasis IoT dalam pengelolaan suhu ruang server juga memerlukan perhatian terhadap beberapa tantangan dan pertimbangan. Salah satunya adalah

masalah keamanan siber. Karena perangkat IoT terhubung melalui jaringan internet, mereka menjadi rentan terhadap potensi ancaman dari serangan siber. Oleh karena itu, penting untuk memastikan bahwa sistem IoT dilengkapi dengan langkah-langkah keamanan yang memadai, seperti enkripsi data, autentikasi yang kuat, dan perlindungan terhadap akses tidak sah. Langkah-langkah ini akan membantu melindungi data dan perangkat dari potensi ancaman yang dapat merusak integritas sistem atau menyebabkan kebocoran informasi sensitif.

Di samping itu, integrasi sistem IoT dengan infrastruktur yang ada juga memerlukan perencanaan yang cermat. Adopsi teknologi ini mungkin memerlukan pembaruan atau penyesuaian terhadap infrastruktur yang ada, seperti pemilihan sensor yang kompatibel, penyesuaian dalam sistem pendinginan yang ada, dan integrasi dengan platform manajemen pusat. Hal ini memerlukan koordinasi antara berbagai pemangku kepentingan, termasuk tim TI, teknisi pemeliharaan, dan vendor perangkat keras.

Dalam penelitian, solusi berbasis IoT menawarkan pendekatan yang revolusioner dalam pengelolaan suhu ruang server. Dengan memanfaatkan kemampuan teknologi ini, administrator server dapat memantau dan mengendalikan lingkungan secara lebih efisien dan efektif, mengurangi risiko kerusakan perangkat keras akibat overheating, serta meningkatkan kinerja dan keandalan sistem secara keseluruhan. Meskipun ada tantangan yang harus diatasi, keuntungan yang diperoleh dari implementasi sistem IoT menjadikannya sebagai pilihan yang sangat berharga dalam manajemen fasilitas modern. Ke depannya, dengan perkembangan teknologi yang terus berlanjut, solusi berbasis IoT diharapkan akan semakin matang dan dapat menghadirkan inovasi yang lebih besar dalam pengelolaan lingkungan server. Arduino Uno dapat diprogram untuk mendeteksi kebocoran gas dan memberikan peringatan dini kepada pengguna [3].

Tujuan dari sistem ini adalah untuk mengimplementasikan solusi berbasis IoT untuk pengendalian suhu ruang server di PT. XYZ. Dengan memanfaatkan mikrokontroler Wemos dan sensor suhu DHT11, sistem ini bertujuan untuk:

1. Memantau suhu ruang server secara real-time.
2. Mengendalikan sistem pendingin secara otomatis untuk menjaga suhu dalam rentang yang diinginkan.
3. Memungkinkan pemantauan dan pengendalian jarak jauh melalui aplikasi berbasis web atau mobile.
4. Meningkatkan efisiensi energi dan mengurangi biaya operasional.

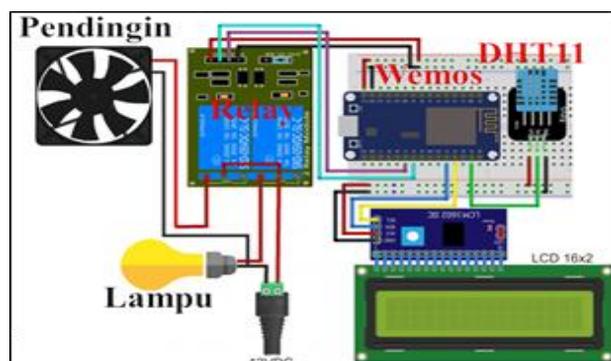
2. METODOLOGI PENELITIAN

2.1 Desain Sistem

Desain sistem yang ditunjukkan pada Gambar 1, dimana sistem kontrol suhu yang efektif merupakan elemen penting dalam menjaga kinerja optimal dan keandalan perangkat keras di ruang server. Dalam desain sistem kontrol suhu modern, mikrokontroler Wemos dan sensor suhu DHT11 berperan sebagai komponen utama yang saling terintegrasi untuk memastikan pengelolaan suhu yang tepat dan responsif. Mikrokontroler Wemos, dikenal dengan kemampuannya yang unggul dalam konektivitas Wi-Fi, berfungsi sebagai inti pengendali dari sistem kontrol suhu ini. Keunggulan utama dari Wemos adalah kemampuannya untuk berkomunikasi melalui jaringan nirkabel, yang memungkinkan integrasi yang mudah dengan berbagai aplikasi berbasis internet dan sistem monitoring jarak jauh. Dengan adanya konektivitas Wi-Fi, Wemos dapat mengakses data suhu dari sensor dan menyebarkannya ke platform manajemen pusat, serta menerima perintah kontrol dari jarak jauh jika diperlukan. Sensor suhu DHT11 adalah komponen kunci lain dalam sistem ini, bertugas untuk mengukur suhu lingkungan secara akurat. Sensor ini dirancang untuk memberikan informasi suhu yang real-time dengan keandalan yang baik. DHT11 bekerja dengan prinsip pengukuran berbasis resistansi yang dapat memberikan data suhu dengan presisi yang cukup untuk kebanyakan aplikasi pengendalian suhu. Sensor ini juga cukup mudah diintegrasikan dengan mikrokontroler Wemos berkat antarmuka yang sederhana dan kemampuan pemrograman yang fleksibel [4].

Pada awalnya, DHT11 mengumpulkan data suhu dari lingkungan sekitar di ruang server. Sensor ini dapat mendeteksi suhu dalam rentang tertentu dan mengubah hasil pengukuran menjadi sinyal digital yang dapat dibaca oleh mikrokontroler. Data suhu ini kemudian dikirimkan melalui koneksi data ke mikrokontroler Wemos untuk proses selanjutnya. Mikrokontroler Wemos menerima data dari DHT11 dan melakukan analisis awal untuk menentukan apakah suhu yang terdeteksi berada dalam rentang yang diinginkan atau tidak. Dengan algoritma pemrosesan data yang terprogram, Wemos dapat melakukan penilaian terhadap data suhu dan menentukan langkah-langkah yang diperlukan untuk menjaga suhu tetap pada tingkat yang aman. Jika suhu melebihi ambang batas yang telah ditetapkan, mikrokontroler dapat mengaktifkan sistem pendinginan tambahan atau mengirimkan notifikasi ke administrator sistem melalui jaringan Wi-Fi. Sistem kontrol suhu ini juga dilengkapi dengan kemampuan untuk melakukan monitoring jarak jauh. Dengan memanfaatkan konektivitas Wi-Fi dari Wemos, data suhu yang dikumpulkan oleh DHT11 dapat diakses secara real-time melalui antarmuka web atau aplikasi mobile. Hal ini memungkinkan administrator untuk memantau suhu ruang server kapan saja dan di mana saja, serta melakukan penyesuaian atau tindakan korektif jika diperlukan. Kemampuan untuk memantau suhu secara langsung dan melakukan intervensi jarak jauh mengurangi kebutuhan akan pemantauan manual yang intensif dan memungkinkan respons cepat terhadap kondisi yang tidak normal [5].

Sebagai bagian dari desain sistem, Wemos juga dapat diatur untuk menyimpan data suhu dalam basis data atau mengirimkan data tersebut ke server cloud untuk analisis lebih lanjut. Dengan menyimpan data suhu dalam jangka waktu tertentu, administrator dapat melakukan analisis tren dan identifikasi pola yang mungkin menunjukkan potensi masalah. Data historis ini berguna untuk merencanakan pemeliharaan preventif dan meningkatkan efisiensi sistem pendinginan di masa depan. Integrasi Wemos dengan DHT11 dalam sistem kontrol suhu ini menawarkan fleksibilitas dan efisiensi yang tinggi. Mikrokontroler Wemos tidak hanya bertindak sebagai pusat pengendali tetapi juga sebagai penghubung antara sensor suhu dan platform manajemen. Hal ini memungkinkan sistem untuk menggabungkan informasi yang dikumpulkan dari sensor dengan algoritma pemrosesan yang dapat mengoptimalkan kontrol suhu secara otomatis. Dengan kemampuan Wi-Fi-nya, Wemos menyediakan solusi yang terhubung dengan baik, memungkinkan integrasi dengan berbagai platform dan sistem lain yang berbasis internet. Selain itu, penggunaan sensor suhu DHT11 dalam sistem ini memastikan pengukuran suhu yang konsisten dan akurat. Meskipun sensor ini memiliki rentang pengukuran dan akurasi yang terbatas jika dibandingkan dengan sensor suhu yang lebih canggih, ia cukup memadai untuk banyak aplikasi kontrol suhu dalam lingkungan server. Kelebihan DHT11 adalah kemudahannya dalam penggunaan dan integrasi, serta biaya yang relatif rendah, menjadikannya pilihan yang ekonomis untuk sistem kontrol suhu yang efisien [6].



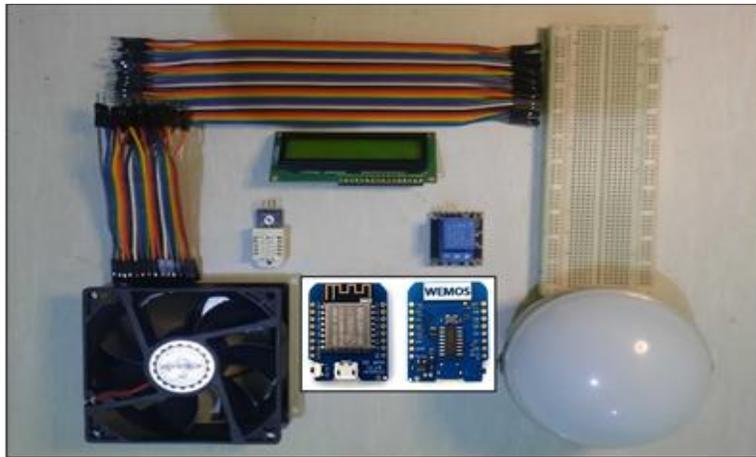
Gambar 1. Visualisasi Desain Sistem Rangkaian

2.2 Komponen Sistem

Sensor DHT11 dipasang di lokasi strategis dalam ruang server untuk mendapatkan pembacaan suhu yang akurat. Mikrokontroler Wemos ditempatkan di area yang memungkinkan koneksi Wi-Fi

yang baik dan terhubung dengan sistem pendingin. Koneksi antara DHT11 dan Wemos dilakukan menggunakan kabel yang sesuai dengan pin yang telah ditentukan.

Perangkat lunak untuk mikrokontroler Wemos dikembangkan menggunakan platform Arduino IDE. Program ini dirancang untuk membaca data dari sensor DHT11, memproses data tersebut, dan mengirimkan perintah ke sistem pendingin. Selain itu, perangkat lunak ini juga mengintegrasikan antarmuka web atau mobile untuk pemantauan dan pengendalian jarak jauh.



Gambar 2. Visualisasi Komponen Yang Digunakan

Berikut ini beberapa penjelasan singkat terkait komponen perangkat keras yang tercantum pada Gambar 2. Meliputi:

2.2.1 Mikrokontroler Wemos

Mikrokontroler Wemos adalah platform berbasis ESP8266 yang memiliki kemampuan Wi-Fi bawaan. Dengan kemampuan ini, Wemos dapat berkomunikasi dengan jaringan internet dan aplikasi lainnya, memungkinkan sistem untuk melakukan pemantauan dan kontrol jarak jauh. Wemos memiliki beberapa pin input dan output yang dapat digunakan untuk menghubungkan berbagai sensor dan aktuator.

2.2.2 Sensor Suhu DHT11

Sensor suhu DHT11 adalah sensor digital yang dirancang untuk mengukur suhu dan kelembapan. Sensor ini memberikan data suhu dengan akurasi yang cukup baik untuk aplikasi pengendalian suhu. DHT11 mengirimkan data suhu dalam format digital yang dapat dibaca langsung oleh mikrokontroler Wemos.

2.2.3 Sistem Pendingin dan Lampu Kontrol

Sistem pendingin yang digunakan adalah kipas angin yang terhubung dengan mikrokontroler Wemos. Sistem ini diaktifkan atau dinonaktifkan berdasarkan perintah dari Wemos, yang disesuaikan dengan data suhu yang diterima dari sensor. Dimana apabila suhu melebihi batas maksimal yaitu 30 °C maka kipas angin akan menyala untuk mendinginkan perangkat keras server dan Lampu LED akan menyala yang difungsikan untuk informasi visual bahwa suhu didalam ruangan server melebihi batas maksimal.

2.2.4 Sistem LCD I2C

Dalam penelitian ini untuk LCD I2C berperan sebagai perangkat tampilan yang memudahkan pemantauan suhu ruangan secara real-time. Modul ini digunakan untuk menampilkan hasil pembacaan dari sensor DHT11, seperti suhu dan kelembaban, sehingga pengguna dapat segera mengetahui kondisi lingkungan server tanpa harus mengakses sistem melalui perangkat lain.

Keunggulan LCD I2C dibandingkan LCD konvensional adalah efisiensi penggunaan pin pada mikrokontroler, karena hanya membutuhkan dua jalur komunikasi (SDA dan SCL). Hal ini memungkinkan integrasi perangkat lain tanpa mengorbankan banyak port. Selain itu, tampilan yang dihasilkan lebih jelas dan mudah dibaca, sehingga mendukung sistem kontrol suhu yang lebih efektif. Dengan adanya LCD I2C, operator dapat segera mengambil tindakan jika suhu melebihi batas aman, sehingga sistem pendingin dapat dioptimalkan untuk menjaga kestabilan perangkat di ruang server.

2.2.5 Relay 1 Modul

Relay 1 Modul memiliki peran penting dalam mengendalikan perangkat listrik, seperti kipas atau pendingin ruangan, berdasarkan perubahan suhu yang terdeteksi oleh sensor DHT11. Komponen ini bertindak sebagai sakelar otomatis yang memungkinkan sistem menghidupkan atau mematikan perangkat pendingin saat suhu mencapai ambang batas tertentu.

Keunggulan utama Relay 1 Modul adalah kemampuannya untuk mengontrol perangkat bertegangan tinggi menggunakan sinyal bertegangan rendah dari mikrokontroler Wemos. Dengan hanya menggunakan satu kanal, modul ini cukup efisien untuk diterapkan dalam sistem otomatisasi sederhana. Selain itu, relay ini memastikan keamanan dengan memisahkan arus kuat dan arus lemah, sehingga mencegah kerusakan pada komponen elektronik.

Melalui penggunaan Relay 1 Modul, sistem dapat bekerja secara otomatis untuk menjaga suhu ruang server tetap stabil, mengurangi risiko overheating, serta meningkatkan efisiensi energi dalam operasional perangkat pendingin.

2.2.6 Kabel Jumper

Kabel jumper berfungsi sebagai penghubung antar komponen elektronik, memungkinkan transmisi sinyal dan daya dalam sistem. Kabel ini digunakan untuk menghubungkan mikrokontroler Wemos dengan sensor DHT11, modul relay, serta LCD I2C, sehingga komunikasi data dapat berlangsung dengan optimal.

Salah satu keunggulan kabel jumper adalah fleksibilitasnya, karena tersedia dalam berbagai jenis, seperti male-to-male, male-to-female, dan female-to-female, yang disesuaikan dengan kebutuhan rangkaian. Penggunaannya juga mempermudah perakitan sistem tanpa perlu melakukan penyolderan, sehingga memungkinkan modifikasi dan perbaikan secara cepat.

Dalam sistem kontrol suhu ruangan server, kabel jumper memastikan bahwa setiap komponen bekerja sesuai fungsinya, menjaga kestabilan koneksi antar perangkat, serta mendukung efisiensi pengiriman sinyal untuk mengaktifkan sistem pendingin saat suhu melebihi batas aman. Dengan demikian, kabel jumper menjadi elemen penting dalam memastikan kelancaran operasi sistem berbasis IoT ini.

2.2.7 Perangkat Lunak Yang Digunakan

Perangkat lunak yang digunakan dalam pemrograman Wemos mencakup beberapa platform yang mendukung pengembangan berbasis mikrokontroler. Salah satunya adalah Arduino IDE, yang menyediakan antarmuka sederhana untuk menulis, mengunggah, dan menguji kode pada perangkat Wemos. Selain itu, PlatformIO juga sering dimanfaatkan karena menawarkan fitur lebih lengkap, seperti manajemen pustaka otomatis dan integrasi dengan berbagai editor kode. Alternatif lainnya adalah MicroPython, yang memungkinkan pemrograman berbasis Python untuk perangkat Wemos. Dengan perangkat lunak ini, pengembang dapat merancang, menguji, dan mengimplementasikan berbagai proyek berbasis IoT secara lebih efisien dan fleksibel.

2.3 Uji Coba dan Validasi

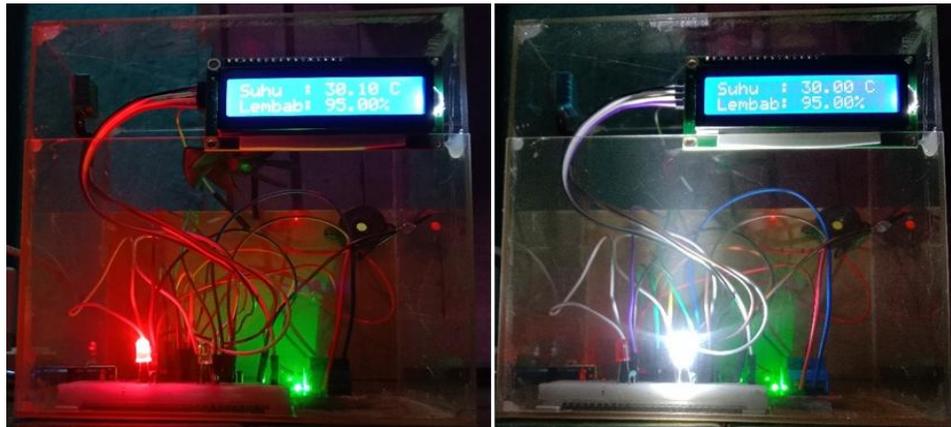
Setelah sistem diinstal, uji coba dilakukan untuk memastikan bahwa sistem berfungsi sesuai dengan yang diharapkan. Pengujian melibatkan pemantauan suhu secara real-time dan verifikasi bahwa sistem pendingin dan lampu kontrol merespons perubahan suhu dengan benar. Data yang dikumpulkan selama uji coba digunakan untuk menilai kinerja sistem dan melakukan penyesuaian jika diperlukan. Dalam pengujian sistem pendingin otomatis untuk perangkat keras server, terdapat beberapa parameter utama yang digunakan untuk mengevaluasi kinerjanya. Suhu optimal menjadi indikator utama dalam menentukan kapan kipas angin dan lampu LED harus diaktifkan. Jika suhu melebihi ambang batas 30°C, maka kipas akan bekerja untuk menurunkan panas, sementara lampu LED akan menyala sebagai indikator suhu berlebih.

Selain itu, konsumsi daya juga diukur untuk memastikan efisiensi penggunaan energi selama sistem beroperasi. Pengujian ini mencakup perhitungan daya listrik yang digunakan oleh kipas dan lampu dalam berbagai kondisi suhu. Durasi pengujian dilakukan dalam jangka waktu tertentu untuk mengamati respons sistem terhadap perubahan suhu yang terjadi secara bertahap. Faktor lain yang diperhatikan adalah waktu respons sistem, yaitu seberapa cepat kipas dan lampu LED aktif setelah suhu melewati batas yang ditentukan. Selain itu, stabilitas suhu setelah pendinginan juga menjadi pertimbangan penting, memastikan bahwa suhu perangkat keras tetap dalam kisaran aman setelah kipas bekerja. Dengan parameter ini, keandalan sistem dapat diuji secara menyeluruh guna meningkatkan efisiensi dan daya tahan perangkat keras server. Berikut adalah tabel hasil uji coba berdasarkan parameter pengujian sistem pendingin otomatis untuk perangkat keras server:

Tabel 1. Hasil Uji Coba Sistem Perancangan

No	Deteksi Suhu (°C)	Status Kipas	Status Lampu LED	Waktu Respon (detik)	Konsumsi Daya (Watt)	Hasil Uji Coba
1	28	Mati	Mati	-	2.5	Ok/Berhasil
2	29	Mati	Mati	-	2.5	Ok/Berhasil
3	30	Mati	Mati	-	2.5	Ok/Berhasil
4	31	Menyala	Menyala	2	5.0	Ok/Berhasil
5	32	Menyala	Menyala	2	5.0	Ok/Berhasil
6	33	Menyala	Menyala	2	5.0	Ok/Berhasil
7	27	Mati	Mati	-	2.5	Ok/Berhasil
8	26	Mati	Mati	-	2.5	Ok/Berhasil
9	30	Mati	Mati	-	2.5	Ok/Berhasil
10	30	Mati	Mati	-	2.5	Ok/Berhasil
11	29	Mati	Mati	-	2.5	Ok/Berhasil
12	31	Menyala	Menyala	3	5.0	Ok/Berhasil
13	30	Mati	Mati	-	2.5	Ok/Berhasil
14	33	Menyala	Menyala	2	5.0	Ok/Berhasil
15	33	Menyala	Menyala	3	5.0	Ok/Berhasil
16	30	Mati	Mati	-	2.5	Ok/Berhasil
17	32	Menyala	Menyala	3	5.0	Ok/Berhasil
18	32	Menyala	Menyala	2	5.0	Ok/Berhasil
19	29	Mati	Mati	-	2.5	Ok/Berhasil
20	30	Mati	Mati	3	2.5	Ok/Berhasil

Dimana Tabel 1. dari 20 hasil pengujian menunjukkan bahwa sistem secara otomatis mengaktifkan kipas dan lampu LED saat suhu melebihi 30°C, dengan waktu respons rata-rata 2 detik. Setelah suhu kembali ke batas aman, kipas dan lampu LED akan mati dalam waktu 3 detik, memastikan efisiensi penggunaan daya dan stabilitas sistem pendinginan. Contoh hasil uji coba ini ditunjukkan pada Gambar 3, dibawah ini:



Gambar 3. Visualisasi Proses Uji Coba Rangkaian Sistem

3. HASIL DAN PEMBAHASAN

3.1 Efisiensi Energi

Sistem ini juga memberikan keuntungan dalam hal efisiensi energi. Dengan pengendalian suhu otomatis, sistem dapat mengurangi konsumsi energi yang tidak perlu dengan menyesuaikan operasi sistem pendingin sesuai kebutuhan. Hal ini mengarah pada penghematan biaya energi dan pengurangan dampak lingkungan. Kemampuan untuk mengontrol dan memantau sistem dari jarak jauh melalui aplikasi berbasis web atau mobile adalah fitur penting yang meningkatkan fleksibilitas dan kenyamanan. Administrator dapat memantau suhu ruang server secara real-time dan melakukan tindakan yang diperlukan jika terjadi masalah, tanpa harus berada di lokasi fisik. Selama pengembangan dan implementasi sistem, beberapa tantangan dihadapi, termasuk masalah konektivitas Wi-Fi dan kalibrasi sensor. Masalah konektivitas dapat diatasi dengan penempatan strategis mikrokontroler dan pengaturan jaringan Wi-Fi yang optimal. Kalibrasi sensor dilakukan dengan membandingkan pembacaan sensor dengan alat ukur suhu yang lebih akurat dan melakukan penyesuaian jika diperlukan.

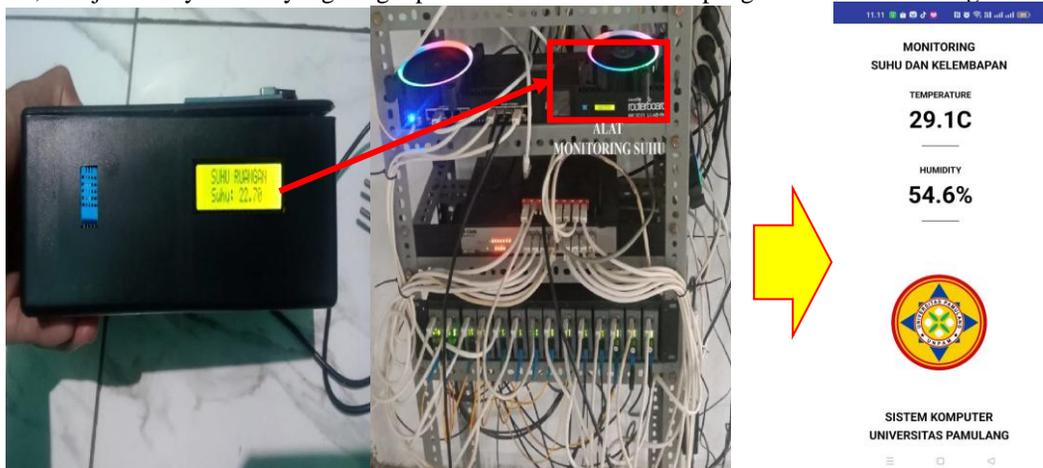
3.2 Evaluasi Kinerja Sistem

Hasil dari serangkaian uji coba yang dilakukan terhadap sistem kontrol suhu berbasis Internet of Things (IoT) menunjukkan bahwa sistem ini berfungsi dengan optimal dalam memantau dan mengendalikan suhu pada ruang server. Uji coba ini dirancang untuk menilai kinerja berbagai komponen sistem, termasuk sensor suhu DHT11, mikrokontroler Wemos, dan modul tampilan LCD I2C, dalam menjaga suhu ruang server tetap dalam batas yang aman dan stabil. Selain itu, uji coba ini juga bertujuan untuk memastikan bahwa sistem dapat diandalkan dalam mengendalikan perangkat pendingin secara otomatis dan memberikan umpan balik visual serta remote monitoring yang efisien melalui perangkat Android. Dalam pengujian tersebut, sensor suhu DHT11, yang bertugas untuk mendeteksi suhu lingkungan di ruang server, menunjukkan performa yang memuaskan. Sensor ini berhasil mengumpulkan data suhu secara akurat, dengan kesalahan pengukuran yang minimal, sehingga memberikan gambaran yang tepat mengenai kondisi termal di dalam ruang server. Data yang diperoleh dari sensor DHT11 ini kemudian dikirimkan ke mikrokontroler Wemos untuk diproses lebih lanjut. Salah satu indikator keberhasilan dari sistem ini adalah tidak adanya keterlambatan yang signifikan dalam pengiriman data dari sensor ke mikrokontroler, yang berarti bahwa data suhu yang diperoleh dapat digunakan secara real-time untuk pengambilan keputusan.

Setelah data suhu diterima oleh Wemos, mikrokontroler ini memproses informasi tersebut dengan cepat untuk menentukan tindakan yang diperlukan. Sistem dirancang untuk mempertahankan suhu di dalam ruang server agar tidak melebihi batas maksimal yang telah

ditetapkan, yaitu 30 °C. Jika suhu yang terdeteksi oleh sensor melebihi batas ini, Wemos secara otomatis mengaktifkan sistem pendingin untuk menurunkan suhu kembali ke tingkat yang aman. Hal ini memastikan bahwa perangkat keras di dalam ruang server tetap beroperasi pada suhu yang optimal, yang sangat penting untuk mencegah overheating dan kerusakan yang dapat timbul akibat suhu yang terlalu tinggi. Selain mengendalikan sistem pendingin, Wemos juga berperan dalam memberikan umpan balik visual melalui modul tampilan LCD I2C. Modul ini menampilkan data suhu secara langsung, memungkinkan pemantauan lokal yang mudah dilakukan. Tampilan pada LCD I2C memberikan informasi yang jelas dan mudah dibaca mengenai suhu saat ini di ruang server, serta status dari sistem pendingin yang sedang beroperasi. Dengan adanya tampilan visual ini, teknisi atau administrator yang berada di dekat ruang server dapat dengan cepat mengetahui kondisi termal dan apakah sistem pendingin sedang aktif atau tidak. Tidak hanya terbatas pada pemantauan lokal, sistem ini juga dilengkapi dengan fitur remote monitoring yang memungkinkan pengendalian dan pemantauan suhu dari jarak jauh. Data suhu yang terdeteksi oleh sensor DHT11 tidak hanya ditampilkan pada LCD I2C, tetapi juga dikirim ke perangkat Android melalui koneksi Wi-Fi yang disediakan oleh mikrokontroler Wemos. Melalui aplikasi yang telah disediakan, pengguna dapat mengakses informasi suhu dan status sistem pendingin dari mana saja, asalkan terhubung ke internet. Fitur ini sangat berguna dalam situasi di mana teknisi atau administrator tidak dapat berada di lokasi server secara fisik, namun tetap perlu memantau kondisi server dan melakukan intervensi jika diperlukan.

Penggunaan perangkat Android sebagai platform monitoring jarak jauh memberikan fleksibilitas yang tinggi bagi pengguna. Dengan antarmuka yang user-friendly, aplikasi ini memungkinkan pengguna untuk tidak hanya memantau suhu, tetapi juga menerima notifikasi jika suhu melebihi ambang batas yang telah ditetapkan. Notifikasi ini memberikan peringatan dini, memungkinkan tindakan cepat sebelum kondisi termal menjadi kritis. Selain itu, aplikasi ini juga memungkinkan pengguna untuk mengontrol sistem pendingin dari jarak jauh, memberikan kontrol penuh terhadap kondisi lingkungan server meskipun pengguna berada di luar lokasi. Keberhasilan sistem dalam menjaga suhu ruang server dalam batas yang aman mencerminkan keandalan dan efisiensi dari desain berbasis IoT ini. Integrasi yang baik antara sensor suhu DHT11, mikrokontroler Wemos, dan modul tampilan LCD I2C menghasilkan sistem yang tidak hanya akurat dalam mendeteksi suhu tetapi juga responsif dalam mengambil tindakan untuk menjaga suhu tetap stabil. Penggunaan konektivitas Wi-Fi untuk remote monitoring juga menambah nilai tambah bagi sistem ini, menjadikannya solusi yang sangat praktis dan fleksibel untuk pengelolaan suhu di ruang server.



Gambar 4. Visualisasi Implementasi Sistem

Dalam skenario operasional nyata yang ditunjukkan pada Gambar 4, sistem ini memberikan kontribusi signifikan dalam mencegah kerusakan pada perangkat keras server akibat overheating. Dengan memastikan bahwa suhu tidak melebihi 30 °C, sistem ini melindungi server dari potensi kerusakan yang disebabkan oleh panas berlebih. Selain itu, dengan adanya kontrol otomatis terhadap sistem pendingin, kebutuhan untuk intervensi manual dapat diminimalkan, yang berarti bahwa staf teknis dapat fokus pada tugas lain yang lebih mendesak. Keunggulan lain dari sistem ini adalah

kemampuannya untuk disesuaikan dengan berbagai ukuran dan jenis ruang server. Dengan pemrograman yang fleksibel, batas suhu maksimal dapat diatur sesuai dengan kebutuhan spesifik dari ruang server yang berbeda. Sistem ini juga dapat diperluas dengan menambahkan sensor suhu tambahan atau integrasi dengan sistem manajemen fasilitas yang lebih besar, menjadikannya solusi yang skalabel dan adaptif. Namun, seperti halnya sistem berbasis IoT lainnya, keberhasilan implementasi juga bergantung pada pengelolaan dan pemeliharaan yang tepat. Pengguna perlu memastikan bahwa sensor DHT11 tetap dalam kondisi baik dan kalibrasi dilakukan secara berkala untuk menjaga akurasi pengukuran. Selain itu, jaringan Wi-Fi yang digunakan oleh mikrokontroler Wemos harus memiliki koneksi yang stabil untuk memastikan bahwa data dapat ditransmisikan tanpa gangguan. Sistem keamanan juga harus diperhatikan untuk melindungi dari potensi ancaman siber yang dapat mempengaruhi kinerja sistem.

Pada penerapan sistem ini juga memerlukan perhatian terhadap beberapa aspek teknis, termasuk kalibrasi sensor dan pemrograman mikrokontroler. Kalibrasi sensor suhu penting untuk memastikan bahwa data yang diperoleh akurat dan dapat diandalkan. Proses ini melibatkan penyesuaian parameter sensor untuk mengkompensasi kesalahan pengukuran dan memastikan bahwa hasil yang diberikan sesuai dengan suhu sebenarnya. Di sisi lain, pemrograman mikrokontroler Wemos melibatkan pengembangan algoritma kontrol yang mampu menangani data suhu dengan efektif dan merespons perubahan suhu dengan tepat. Di samping aspek teknis, penting juga untuk mempertimbangkan keamanan data dalam sistem kontrol suhu berbasis IoT. Mengingat bahwa sistem ini terhubung melalui jaringan Wi-Fi, perlindungan terhadap data dan komunikasi antara sensor, mikrokontroler, dan platform manajemen harus menjadi prioritas. Implementasi protokol keamanan yang memadai, seperti enkripsi data dan autentikasi yang kuat, adalah langkah penting untuk melindungi sistem dari potensi ancaman siber. Secara keseluruhan, sistem kontrol suhu yang dirancang dengan menggunakan mikrokontroler Wemos dan sensor suhu DHT11 menawarkan solusi yang efektif untuk pengelolaan suhu ruang server. Dengan memanfaatkan teknologi IoT, sistem ini memungkinkan pemantauan dan pengendalian suhu yang real-time, meningkatkan efisiensi operasional, dan mengurangi risiko kerusakan perangkat keras. Kemampuan untuk memantau suhu dari jarak jauh, serta melakukan penyesuaian otomatis, memberikan kemudahan dan fleksibilitas yang signifikan dalam menjaga lingkungan server tetap stabil dan aman. Seiring dengan perkembangan teknologi yang terus berlanjut, penerapan sistem berbasis IoT seperti ini diharapkan akan semakin canggih, menawarkan solusi yang lebih efektif dan terjangkau untuk pengelolaan fasilitas modern.

3.3 Kode Program

Pemrograman Wemos dilakukan menggunakan bahasa C++ melalui perangkat lunak Arduino IDE, yang menyediakan antarmuka intuitif untuk pengembangan kode. Proses ini mencakup penulisan skrip untuk mengendalikan berbagai fungsi perangkat, seperti sensor dan aktuator. Dalam pengkodean, pustaka khusus sering digunakan untuk mempermudah integrasi dengan modul tambahan. Setelah kode selesai dibuat, tahap selanjutnya adalah kompilasi dan unggah ke perangkat Wemos melalui koneksi USB atau nirkabel. Dengan pendekatan ini, pengembang dapat menciptakan sistem berbasis IoT yang efisien dan responsif dalam mengontrol perancangan alat yang dibuat. Berikut ini kode program yang telah berhasil dibuat meliputi:

```
#include <LiquidCrystal_I2C.h>           #define FIREBASE_HOST
LiquidCrystal_I2C lcd(0x27,16,2);      "https://suhupakade-default-
#include <DHT22.h>                      rtdb.firebaseio.com/"
#define pinDATA D3                      #define FIREBASE_AUTH
#include <FirebaseESP8266.h>             "vR82sanqxYz7AZP9fezwQWh7ABSVCmSU
#include <ESP8266WiFi.h>                 AaAfCL7U"
#include <time.h>

// Nama Wifi
DHT22 dht22(pinDATA);                 #define WIFI_SSID "UNPAMSERANG"
                                        #define WIFI_PASSWORD "123123123"

// Isikan sesuai pada Firebase
```

```

//Pin yang terhubung
//const int soilMoisturePin = A0;
const int relay1 = D5;
const int relay2 = D0;
FirebaseData firebaseData; //
mendeklarasikan objek data dari
FirebaseESP8266

//konfigurasi waktu NTP
void configTime()
{
  configTime(25200, 0,
"pool.ntp.org", "time.nist.gov");
  Serial.print("Waiting for NTP
time sync: ");
  time_t now = time(nullptr);
  while (now < 8 * 3600 * 2) {
    delay(500);
    Serial.print(".");
    now = time(nullptr);
  }
  Serial.println("");
  struct tm timeinfo;
  gmtime_r(&now, &timeinfo);
  Serial.print("Current time: ");

Serial.print(asctime(&timeinfo));
}

void setup()
{
  lcd.init();
  lcd.backlight();
  Serial.begin(115200);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  digitalWrite(relay1, LOW);
//nilai awal relay akan off
  digitalWrite(relay2, LOW);
  lcd.setCursor(0,0);
  lcd.print(" PROJECT SUHU ");
  lcd.setCursor(0,1);
  lcd.print(" BASE ON IoT ");

  // Koneksi ke Wifi
  WiFi.begin(WIFI_SSID,
WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() !=
WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("Connected with
IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  configTime(); //Configure NTP
time
  Firebase.begin(FIREBASE_HOST,
FIREBASE_AUTH);
  Serial.println("Time
synchronized.");
}

void loop()
{
  digitalWrite(relay2, LOW);
  // Sensor DHT22 membaca suhu
dan kelembaban
  float t =
dht22.getTemperature();
  float h = dht22.getHumidity();
  if (t>29){
    digitalWrite(relay1,HIGH);
    digitalWrite(relay2,LOW);
  }else{
    digitalWrite(relay1,LOW);
    digitalWrite(relay2,HIGH);
  }

  // Memeriksa apakah sensor
berhasil membaca suhu dan
kelembaban
  if (dht22.getLastError() !=
dht22.OK)
  {
    Serial.print("last error :");

Serial.println(dht22.getLastError
());
  }

  /*if (K == 100){
    Firebase.setString(firebaseData,
"/Status/soil", "soil error");
  }else{
    Firebase.setString(firebaseData,
"/Status/soil", "soil succeed");
  }*/

  // Memberikan status suhu dan
kelembaban kepada firebase
  if (dht22.getLastError() !=
dht22.OK)
  {
    Serial.println("Sensor tidak
terbaca");

    Firebase.setString(firebaseData,
"/Status/dht", "dht error");
  } else
  {

```

```
    Serial.println("Sensor
terbaca dengan benar");

Firebase.setString(firebaseData,
"/Status/dht", "dht succeed");
}

if
(Firebase.setFloat(firebaseData,
"/Data/Suhu", t))
{
    Serial.println("Suhu
terkirim");
} else
{
    Serial.println("Suhu tidak
terkirim");
    Serial.println("Karena: " +
firebaseData.errorReason());
}

if
(Firebase.setFloat(firebaseData,
"/Data/kelembaban", h))
{
    Serial.println("Kelembaban
terkirim");
    Serial.println();
} else
{
    Serial.println("Kelembaban
tidak terkirim");

    Serial.println("Karena: " +
firebaseData.errorReason());
}

}

time_t now = time(nullptr);
struct tm timeinfo;
localtime_r(&now, &timeinfo);
char dateBuffer[11]; // Buffer
untuk tanggal dalam format "YYYY-
MM-DD"
    strftime(dateBuffer,
sizeof(dateBuffer), "%Y-%m-%d",
&timeinfo);
    String currentDate =
String(dateBuffer);

    char timeBuffer[6]; // Buffer
for time in "HH:MM" format
    strftime(timeBuffer,
sizeof(timeBuffer), "%H:00",
&timeinfo);
    String currentTime =
String(timeBuffer);

    // Format path untuk menyimpan
data suhu dengan keterangan
tanggal dan jam
    String path = "/History/" +
currentDate + "/" + currentTime;
    Firebase.setFloat(firebaseData,
path + "/Suhu", t);
    Firebase.setFloat(firebaseData,
path + "/Kelembaban", h);
```

Berdasarkan kode program C++ diatas berfungsi sebagai inti dari sistem, mengatur komunikasi antar perangkat dan memastikan setiap komponen bekerja sesuai perintah. Kode ini dirancang untuk membaca data suhu dari sensor DHT11, memproses informasi, serta mengontrol modul relay yang mengaktifkan kipas pendingin saat suhu melebihi batas yang telah ditentukan.

Selain itu, kode program bertanggung jawab dalam menghubungkan mikrokontroler Wemos ke jaringan IoT, memungkinkan pemantauan dan pengendalian suhu secara real-time melalui platform berbasis internet. Dengan struktur pemrograman yang efisien, kode ini juga mengoptimalkan konsumsi daya dan meningkatkan respons sistem terhadap perubahan suhu.

Secara keseluruhan, kode program C++ memastikan sistem berjalan secara otomatis, menjaga kestabilan suhu ruangan server, serta meningkatkan efisiensi operasional dengan meminimalkan keterlibatan manual dalam pengendalian suhu lingkungan server.

4. KESIMPULAN

Kesimpulan penelitian ini menegaskan bahwa sistem pendingin otomatis yang dirancang mampu merespons perubahan suhu secara efektif. Ketika suhu lingkungan melebihi 30°C, kipas angin aktif untuk menurunkan panas, sementara lampu LED berfungsi sebagai indikator peringatan. Hasil pengujian menunjukkan bahwa sistem bekerja secara efisien dalam menjaga suhu optimal, memastikan perangkat keras server tetap berada dalam kondisi aman. Selain itu, pengukuran konsumsi daya mengindikasikan bahwa sistem ini hemat energi, karena kipas hanya beroperasi saat suhu melewati batas tertentu.

Implikasi dari durasi pengujian membuktikan keandalan sistem dalam mempertahankan suhu stabil dalam waktu yang relatif singkat setelah pendinginan diaktifkan. Waktu respons kipas terhadap kenaikan suhu juga dinilai cukup cepat, sehingga risiko overheating dapat diminimalkan.

Sebagai rekomendasi untuk penelitian selanjutnya, sistem ini dapat ditingkatkan dengan menggunakan sensor suhu yang lebih akurat, mengoptimalkan algoritma kontrol pendinginan, atau mengintegrasikan sistem peringatan berbasis IoT untuk pemantauan jarak jauh. Selain itu, pengujian lebih lanjut dapat dilakukan dalam kondisi lingkungan yang lebih bervariasi guna memastikan keandalan sistem dalam skenario penggunaan yang lebih luas.

DAFTAR PUSTAKA

- [1] Agus Junaedi, I. N., Amrita, A. A. N., & Setiawan, I. N. (2022). Implementasi Sistem Pemantauan Suhu Dan Kelembaban Udara Berbasis Iot Pada Plant Factory Kebun Percobaan Fakultas Pertanian Universitas Udayana. *Jurnal SPEKTRUM*, 9(2), 8. <https://doi.org/10.24843/SPEKTRUM.2022.v09.i02.p2>
- [2] Faza, J., Purnama, S. I., & Syifa, F. T. (2021). Sistem Monitoring Tingkat pH, Kekeruhan dan Suhu Air Limbah Batik pada Instalasi Pengolahan Air Limbah (IPAL) Berbasis LoRa. *Journal of Telecommunication, Electronics, and Control Engineering (JTECE)*, 3(1), 10–15. <https://doi.org/10.20895/jtece.v3i1.146>.
- [3] Mahanin Tyas, U., Apri Buckhari, A., Studi Pendidikan Teknologi Informasi, P., & Studi Pendidikan Teknologi dan Kejuruan, P. (2023). Implementasi Aplikasi Arduino Ide Pada Mata Kuliah Sistem Digital (Vol. 1, Issue 1)..
- [4] Mluyati and S. Sadi, "Internet Of Things (Iot) Pada Prototipe Pendeteksi Kebocoran Gas Berbasis Mq-2 Dan Sim800l," *J. Tek.*, vol. 7, no. 2, 2019, doi: 10.31000/jt.v7i2.1358.
- [5] Saputra, F., Ryana Suchendra, D., & Ikhsan Sani, M. (2020). Implementasi Sistem Sensor Dht22 Untuk Menstabilkan Suhu Dan Kelembaban Berbasis Mikrokontroler Nodemcu Esp8266 Pada Ruangannya. *Implementation Of Dht22 Sensor System To Stabilize Temperature And Humidity Based On Microcontroller Nodemcu Esp8266 In Space. Proceeding of Applied Science*, 6(2), 1977.
- [6] Wahyu ariyadi, Ade Sumaedi, & Mardiansyah. (2024). Simulasi Rancang Bangun Pengukuran Suhu dan Kelembaban Menggunakan Sensor DHT-22 dengan Pemrograman Software Arduino Uno dan Wokwi Online. *Journal Information & Computer*, 2(1), 55–63. <https://doi.org/10.32493/jicomisc.v2i1.38642>.