

Rancang Bangun Dan Analisis Kinerja Sistem Kendali AC Berbasis IoT Monitoring Berbasis Web

Rangga Ali Ardyansyah^{1*}, Hamdani Arif^{2**}

* Teknologi Rekayasa Perangkat Lunak, Politeknik Negeri Batam
ranggaali541@gmail.com¹, hamdaniarif@polibatam.ac.id²

Article Info

Article history:

Received ...

Revised ...

Accepted ...

Keyword:

Internet of Things, ESP32, air conditioner control system, real-time temperature monitoring, BME280 sensor.

ABSTRACT

The use of the Internet of Things (IoT) enables more efficient control of electronic devices over a network. This study developed an IoT-based air conditioning control system for the server room at PT Patlite Indonesia using an ESP32, a BME280 sensor, and an infrared transmitter. The system provides a web interface for both manual and automatic control, real-time temperature and humidity monitoring, and centralized data logging. Testing was conducted through quantitative experiments covering response time, infrared communication reliability, sensor accuracy, and user acceptance testing. The results show that the system is capable of controlling the air conditioner via a local network with stable performance, while also improving the ease of managing server room conditions.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Kemajuan teknologi Internet of Things (IoT) memungkinkan integrasi perangkat elektronik melalui jaringan untuk mendukung otomatisasi dan pemantauan terpusat. Penerapan IoT pada sistem monitoring lingkungan memberikan akses data suhu dan kelembapan secara berkala sehingga proses pengendalian dapat dilakukan lebih terukur [1]. Salah satu aplikasi penting adalah pengendalian sistem pendingin ruangan, khususnya pada ruang server yang membutuhkan kestabilan suhu dan kelembapan agar perangkat teknologi informasi beroperasi optimal.

Beberapa penelitian sebelumnya telah menerapkan IoT untuk monitoring dan pengendalian AC maupun kondisi lingkungan. [2] mengembangkan sistem kontrol dan monitoring AC berbasis IoT, sedangkan [3] mengembangkan sistem otomatisasi AC berbasis IoT untuk pengendalian suhu ruangan. Penelitian lain juga memanfaatkan ESP32 untuk melakukan *environmental sensing* terhadap suhu dan kelembapan [4]. Meskipun demikian, penelitian ini berfokus pada integrasi pengendalian AC melalui komunikasi inframerah, monitoring kondisi lingkungan, serta mekanisme kendali otomatis berbasis aturan yang terintegrasi dalam satu sistem berbasis web.

Di PT Patlite Indonesia, pengaturan AC masih dilakukan dengan cara yang biasa, yaitu menggunakan timer dan

pengendalian langsung pada perangkat. Namun, cara ini memiliki kelemahan, terutama ketika kita membutuhkan penyesuaian yang cepat. Selain itu, cara ini juga belum mendukung pemantauan suhu secara real-time maupun pencatatan data historis. Oleh karena itu, diperlukan sistem kendali AC yang berbasis IoT, yang lebih fleksibel, terpusat, dan dapat menyediakan data lingkungan secara akurat.

Penelitian ini bertujuan untuk merancang sistem kendali AC yang menggunakan mikrokontroler ESP32, sensor BME280, dan komunikasi inframerah yang terintegrasi dengan dashboard berbasis web. Sistem ini memungkinkan kita untuk mengendalikan AC secara manual atau otomatis, memantau suhu dan kelembapan secara real-time, serta mencatat riwayat aktivitas. Evaluasi dilakukan melalui pengujian waktu respons, keandalan komunikasi inframerah, akurasi sensor, dan penerimaan pengguna.

Penelitian ini memberikan kontribusi pada penerapan IoT untuk pengelolaan AC di ruang server yang lebih efisien. Hasilnya menunjukkan bahwa sistem ini mampu memberikan kontrol terpusat dan monitoring lingkungan yang lebih baik dibandingkan dengan metode konvensional. Dengan demikian, kita dapat meningkatkan kinerja perangkat teknologi informasi dan mengurangi biaya operasional.

II. METODE

A. Metode Pengembangan Sistem

Penelitian ini menerapkan metode *prototyping*, sebuah pendekatan iteratif dalam pengembangan perangkat lunak yang diawali dengan pembuatan purwarupa awal untuk kemudian dievaluasi dan direvisi berdasarkan umpan balik pengguna hingga mencapai spesifikasi yang diinginkan [5]. Metode ini dinilai ideal karena sistem IoT yang dirancang membutuhkan integrasi yang selaras antara perangkat keras dan lunak, sehingga memerlukan pengujian berkala, terutama pada komponen ESP32, sensor BME280, modul IR LED, dan *dashboard* web.

Prosedur *prototyping* dalam riset ini mencakup lima langkah utama: analisis kebutuhan, desain purwarupa, implementasi, pengujian dan evaluasi, serta penyempurnaan akhir. Tahap awal berfokus pada identifikasi kendala lapangan dan kebutuhan pengguna. Langkah berikutnya adalah merancang arsitektur sistem, komunikasi data, perangkat keras, perangkat lunak, serta antarmuka *dashboard*. Pada fase implementasi, sistem kendali AC dikembangkan dengan memanfaatkan mikrokontroler ESP32 yang terintegrasi dengan sensor BME280 dan IR LED, sementara *dashboard* pemantau dikembangkan menggunakan bahasa PHP dan basis data MySQL. Akhirnya, sistem diuji berdasarkan parameter kinerja spesifik untuk mendeteksi kekurangan yang perlu diperbaiki pada tahap penyempurnaan.

B. Metode Eksperimen Kuantitatif

Penelitian ini menerapkan metode eksperimen dengan pendekatan kuantitatif sebagai strategi analisis utama. Melalui metode eksperimen, kinerja sistem kendali AC berbasis IoT yang telah dirancang dapat diuji dan dievaluasi secara langsung dalam kondisi lingkungan yang terkontrol. Sementara itu, pendekatan kuantitatif dipilih karena seluruh data yang dikumpulkan berbentuk nilai numerik, sehingga memungkinkan proses analisis dilakukan secara objektif.

Prosedur pengujian difokuskan untuk mendapatkan parameter data yang terukur dari mekanisme komunikasi inframerah, transmisi perintah, serta akurasi pembacaan sensor. Dalam skenario ini, intervensi yang diterapkan adalah penggunaan sistem kendali AC berbasis IoT, sedangkan aspek yang diamati adalah efektivitas performa sistem pada tiap skenario pengujian. Secara teoritis, metode eksperimen berfungsi untuk mengukur dampak dari suatu perlakuan terhadap variabel lain dalam ruang lingkup yang dikendalikan [6], sedangkan pendekatan kuantitatif berperan dalam menyediakan data numerik yang dapat dianalisis secara akurat [7].

C. Analisis dan Perhitungan

Evaluasi terhadap performa sistem dilakukan dengan mengukur beberapa parameter yang diperoleh dari data hasil pengujian. Indikator utama yang dianalisis meliputi durasi waktu respons sistem, persentase keberhasilan pengiriman sinyal inframerah (IR), serta tingkat presisi sensor BME280.

Ketiga metrik tersebut menjadi acuan penting untuk mengukur kecepatan operasional, keandalan fungsionalitas kendali AC, dan validitas data kondisi lingkungan yang disajikan [8]. Di samping pengukuran performa teknis tersebut, pengujian User Acceptance Testing (UAT) juga diterapkan untuk menilai sejauh mana tingkat penerimaan dan kepuasan pengguna terhadap sistem yang telah dibangun.

1) *Waktu Respons Sistem (Response Time)*: Waktu respons dievaluasi berdasarkan selang waktu sejak instruksi dikirimkan pengguna melalui *dashboard* web hingga perintah tersebut berhasil dijalankan oleh unit AC. Parameter ini berfungsi untuk menilai kecepatan serta efisiensi proses transmisi data antara *dashboard* web, server, dan mikrokontroler ESP32. Nilai rata-rata waktu respons kemudian dihitung menggunakan persamaan (1):

$$\bar{t} = \frac{\sum t_i}{n}$$

di mana $\bar{t}$ melambangkan nilai rata-rata waktu respons, t_i melambangkan durasi respons pada uji coba ke- i , dan n menyatakan akumulasi jumlah percobaan.

2) *Tingkat Keberhasilan Pengiriman Sinyal Inframerah*: Parameter ini diaplikasikan untuk menilai tingkat keberhasilan sistem dalam mentransmisikan sinyal inframerah dari ESP32 ke unit AC sesuai instruksi yang diberikan. Pengujian dilakukan melalui berbagai perintah kendali untuk mengevaluasi keandalan komunikasi inframerah dalam menjalankan fungsi pengendalian AC. Persentase keberhasilan dihitung menggunakan rumus (2):

$$\text{Success Rate}(\%) = \frac{\text{Jumlah Berhasil}}{\text{Total Percobaan}} \times 100\%$$

di mana *Success Rate* diukur berdasarkan rasio antara jumlah pengiriman sinyal yang berhasil terhadap total percobaan pengiriman.

3) *Akurasi Sensor BME280*: Pengujian akurasi BME280 dilakukan dengan membandingkan nilai keluaran sensor terhadap alat ukur standar yang bertindak sebagai referensi. Komparasi ini bertujuan untuk mengukur besarnya deviasi nilai hasil pembacaan guna merepresentasikan kondisi lingkungan yang sebenarnya [9]. Persentase galat (*error*) atau deviasi dihitung menggunakan Persamaan (3):

$$\text{Deviasi}(\%) = \left| \frac{\text{Sensor} - \text{Referensi}}{\text{Referensi}} \right| \times 100\%$$

di mana nilai sensor merupakan hasil pengukuran dari modul BME280 dan nilai referensi diperoleh dari instrumen pembandingan standar. Selanjutnya, tingkat akurasi dari sensor tersebut dikalkulasi menggunakan rumus (4):

$$\text{Akurasi}(\%) = 100\% - \text{Deviasi}(\%)$$

Berdasarkan korelasi kedua parameter tersebut, semakin kecil nilai deviasi atau margin kesalahan yang dihasilkan,

maka tingkat akurasi sensor akan semakin tinggi mendekati kondisi riil di lapangan.

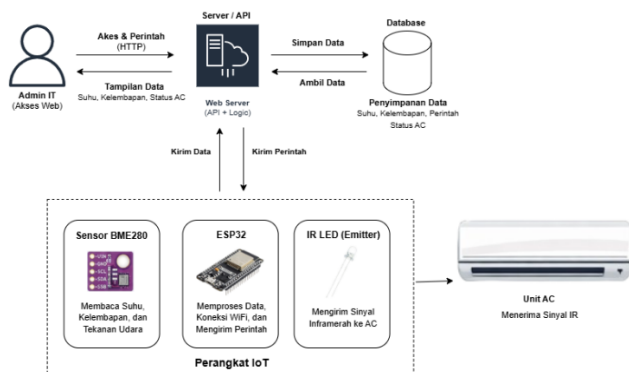
4) *User Acceptance Testing (UAT)*: Metode UAT diterapkan untuk mengukur tingkat akseptabilitas pengguna terhadap operasionalisasi sistem kendali AC berbasis IoT yang telah diimplementasikan. Prosedur pengujian dilakukan dengan melibatkan pengguna secara langsung untuk mengeksplorasi seluruh fungsionalitas sistem melalui antarmuka *dashboard* web, yang kemudian diikuti dengan pemberian skor penilaian menggunakan metrik skala 0–10. Persentase penerimaan pengguna dihitung berdasarkan perbandingan antara total skor yang diperoleh dengan skor maksimum menggunakan rumus (5):

$$UAT(\%) = \frac{\text{Total Skor}}{\text{Skor Maksimum}} \times 100\%$$

di mana *User Acceptance Testing (UAT)* dihitung berdasarkan rasio antara total skor yang diperoleh terhadap skor maksimum yang dapat diperoleh. Nilai persentase tersebut digunakan untuk mengetahui tingkat penerimaan pengguna serta kesesuaian sistem dengan kebutuhan pengguna.

D. Infrastruktur dan Mekanisme Komunikasi Sistem

Sistem kendali AC berbasis IoT dirancang dengan mengintegrasikan ESP32, sensor BME280, IR LED, server aplikasi, basis data MySQL, dan dashboard web dalam satu jaringan lokal. Infrastruktur ini mendukung fungsi pemantauan kondisi lingkungan sekaligus pengendalian AC secara manual maupun otomatis. ESP32 berfungsi sebagai pengendali utama yang mengeksekusi perintah yang dikirim melalui dashboard web, membaca data suhu dan kelembapan melalui sensor BME280, serta mengirimkan sinyal inframerah ke unit AC. Sensor BME280 digunakan untuk memperoleh data lingkungan secara akurat [10].



Gambar 1. Diagram Blok Infrastruktur dan Aliran Data Sistem

Pada proses monitoring, sensor BME280 mengukur suhu dan kelembapan, kemudian ESP32 mengirimkan data ke server melalui REST API berbasis HTTP/JSON. Data disimpan dalam basis data MySQL dan ditampilkan melalui dashboard web. REST API berperan sebagai penghubung

antara perangkat IoT dan server sehingga memungkinkan data sensor dikirim dan diproses oleh sistem [11].

Pada proses pengendalian, instruksi dari dashboard diteruskan ke server dan dikirimkan ke ESP32 untuk dieksekusi melalui IR LED menuju unit AC. Mekanisme komunikasi inframerah digunakan untuk meniru perintah dari remote control sehingga pengendalian dapat dilakukan tanpa modifikasi pada perangkat AC [12]. Proses ini bersifat satu arah karena unit AC tidak memberikan umpan balik kepada pengirim.

Dengan alur ini, sistem mampu menjalankan dua fungsi utama secara terintegrasi: pemantauan kondisi lingkungan dan pengendalian AC, di mana data sensor bergerak dari perangkat ke server/dashboard, sedangkan perintah kendali mengalir dari dashboard ke ESP32 dan dikonversi menjadi sinyal inframerah.

E. Logika dan Alur Kendali Otomatis

Sistem kendali otomatis dirancang dengan mengombinasikan *hysteresis control*, analisis tren suhu, dan *rule-based control*. Pendekatan ini digunakan untuk menentukan tindakan pengendalian AC berdasarkan kondisi suhu ruangan sekaligus meminimalkan pengiriman perintah yang tidak diperlukan akibat fluktuasi suhu. ESP32 secara berkala membaca data dari sensor BME280, kemudian mengolahnya sebagai masukan dalam proses pengambilan keputusan.

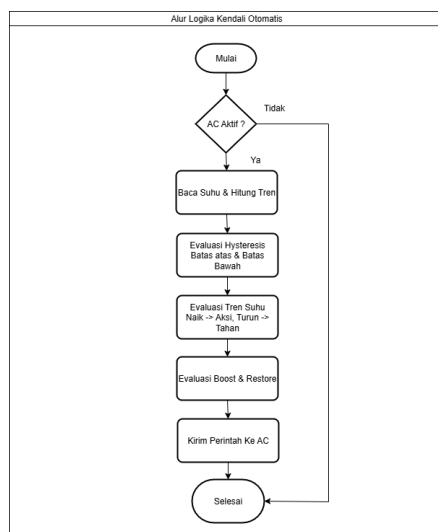
Hysteresis control memberikan zona toleransi terhadap perubahan suhu di sekitar ambang batas dengan menetapkan *upper threshold* dan *lower threshold*. Metode ini mencegah perubahan kecil langsung memicu perintah kendali, sehingga sistem lebih stabil [13]. Dalam implementasi, *upper threshold* ditentukan oleh parameter *temp_threshold*, sedangkan *lower threshold* ditetapkan 2°C di bawahnya.

Analisis tren suhu digunakan untuk mendeteksi arah perubahan suhu dengan membandingkan nilai saat ini dengan pembacaan sebelumnya. Informasi tren ini (*uptrend* atau *downtrend*) menjadi salah satu masukan dalam pengambilan keputusan [14].

Hasil evaluasi dari *hysteresis control* dan tren suhu kemudian diproses oleh *rule-based control* menggunakan aturan IF–THEN untuk menghasilkan tindakan sesuai kondisi masukan [15]. Misalnya, ketika suhu melebihi *upper threshold* dan menunjukkan tren kenaikan, sistem menurunkan *target_temp* sebesar 1°C dan mengirimkan perintah melalui sinyal inframerah. Sebaliknya, jika tren tidak meningkat, sistem menahan pengiriman perintah. Aturan tambahan juga diterapkan, seperti penurunan langsung *target_temp* saat suhu mencapai *upper threshold* + 2°C, aktivasi *boost mode* ketika suhu tetap tinggi meski *target_temp* sudah minimum, serta mekanisme *restore timer* ketika suhu stabil di bawah *lower threshold*.

Secara keseluruhan, alur kendali dimulai dari pembacaan sensor, evaluasi kondisi berdasarkan batas *hysteresis* dan tren suhu, lalu penerapan aturan kendali untuk menentukan

apakah perintah baru perlu dikirimkan. Jika diperlukan, ESP32 mengirimkan sinyal inframerah ke unit AC. Alur ini digambarkan pada Gambar 2.

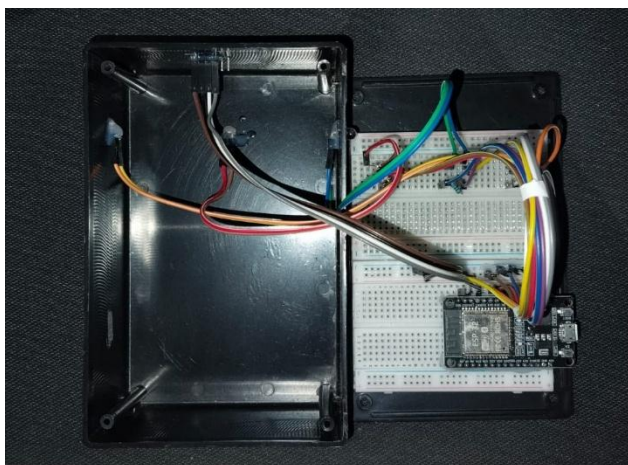


Gambar 2. Flowchart Logika Kendali Otomatis pada ESP32

III. HASIL DAN PEMBAHASAN

A. Implementasi Sistem

Implementasi perangkat keras direalisasikan menggunakan mikrokontroler ESP32 sebagai unit pemroses utama, sensor BME280 untuk mengukur suhu dan kelembapan, serta IR LED sebagai pemancar sinyal inframerah ke unit AC. Rangkaian utama tersebut merupakan infrastruktur final sistem yang hasil perakitan serta interkoneksinya (*wiring*) ditunjukkan pada Gambar 3.

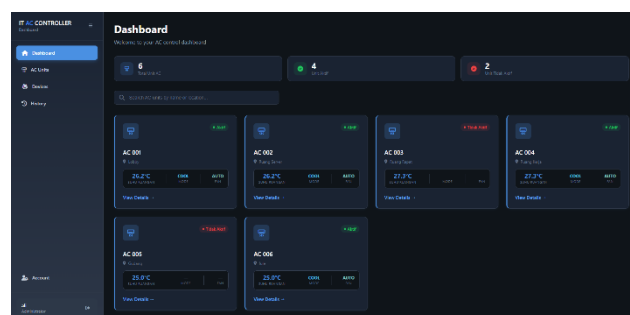


Gambar 3. Hasil Implementasi dan Wiring Perangkat Utama Sistem

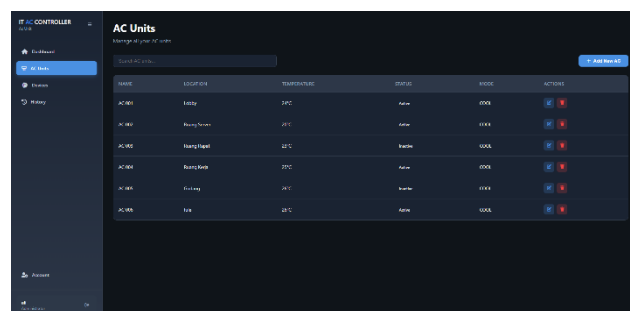
Pada fase awal, perangkat pendukung berupa *IR Receiver* diaplikasikan khusus untuk menangkap dan memetakan arsitektur *raw code* sinyal inframerah dari *remote* asli AC. Hasil pemetaan tersebut kemudian diekstraksi menjadi basis

data perintah inframerah yang ditanamkan pada perangkat utama. Komponen pendukung ini hanya berfungsi pada tahap persiapan data kendali dan dieliminasi pada implementasi akhir sistem.

Sementara itu, implementasi perangkat diwujudkan melalui dashboard berbasis web yang berfungsi sebagai antarmuka untuk pemantauan sekaligus pengendalian sistem. Pada halaman Dashboard, ditampilkan informasi umum mengenai status unit AC serta visualisasi kondisi lingkungan secara real-time. Di sisi lain, halaman *AC Unit* menyediakan daftar inventarisasi unit AC beserta panel fungsi kendali interaktifnya. Representasi antarmuka web tersebut diperlihatkan pada Gambar 4 dan Gambar 5.



Gambar 4. Implementasi Halaman Dashboard



Gambar 5. Implementasi Halaman AC Unit

B. Hasil Pengujian Waktu Respons

Pengujian waktu respons dilakukan untuk mengetahui durasi yang dibutuhkan sistem dalam memproses perintah kendali dari dashboard hingga perintah diterima oleh unit AC. Pengujian dilakukan terhadap 17 jenis perintah, meliputi perintah *ON*, *OFF*, dan pengaturan suhu 16–30°C. Hasil pengujian ditunjukkan pada Tabel 1.

TABEL 1
HASIL PENGUJIAN WAKTU RESPON SISTEM

No	Jenis Perintah	Waktu Respons (detik)
1	ON AC	0.30
2	OFF AC	0.30
3	Suhu 16°	0.32
4	Suhu 17°	0.32
5	Suhu 18°	0.32
6	Suhu 19°	0.32
7	Suhu 20°	0.32

8	Suhu 21°	0.32
9	Suhu 22°	0.32
10	Suhu 23°	0.32
11	Suhu 24°	0.32
12	Suhu 25°	0.32
13	Suhu 26°	0.32
14	Suhu 27°	0.32
15	Suhu 28°	0.32
16	Suhu 29°	0.32
17	Suhu 30°	0.32
Rata-rata		0.3176

Berdasarkan hasil pengujian, total waktu respons dari 17 percobaan adalah 5,40 detik dengan rata-rata sebesar 0,3176 detik atau 0,32 detik. Waktu respons untuk perintah *ON* dan *OFF* tercatat sebesar 0,30 detik, sedangkan seluruh perintah pengaturan suhu memiliki waktu respons sebesar 0,32 detik. Hasil tersebut menunjukkan bahwa sistem mampu memproses perintah kendali dengan waktu respons yang relatif cepat dan mendukung pengendalian AC secara *real-time*.

C. Hasil Pengujian Keberhasilan Sinyal Inframerah

Pengujian keberhasilan sinyal inframerah dilakukan untuk mengetahui kemampuan sistem dalam mengirimkan dan mengeksekusi perintah kendali dari ESP32 ke unit AC. Pengujian dilakukan terhadap 17 perintah yang terdiri atas perintah *ON*, *OFF*, dan pengaturan suhu 16–30°C. Hasil pengujian ditunjukkan pada Tabel 2.

TABEL 2
HASIL PENGUJIAN SINYAL INFRAMERAH

No	Jenis Perintah	Status
1	ON AC	Berhasil
2	OFF AC	Berhasil
3	Suhu 16°	Berhasil
4	Suhu 17°	Berhasil
5	Suhu 18°	Berhasil
6	Suhu 19°	Berhasil
7	Suhu 20°	Berhasil
8	Suhu 21°	Berhasil
9	Suhu 22°	Berhasil
10	Suhu 23°	Berhasil
11	Suhu 24°	Berhasil
12	Suhu 25°	Berhasil
13	Suhu 26°	Berhasil
14	Suhu 27°	Berhasil
15	Suhu 28°	Berhasil
16	Suhu 29°	Berhasil
17	Suhu 30°	Berhasil
Total Percobaan		17

Berdasarkan hasil pengujian, seluruh 17 dari 17 percobaan berhasil dieksekusi, sehingga diperoleh tingkat keberhasilan pengiriman sinyal inframerah sebesar 100%. Hasil tersebut menunjukkan bahwa sistem mampu mentransmisikan

perintah inframerah dari ESP32 ke unit AC secara konsisten pada seluruh perintah yang diuji.

D. Hasil Pengujian Akurasi Sensor BME280

Pengujian akurasi sensor BME280 dilakukan dengan membandingkan hasil pengukuran suhu dan kelembapan yang diperoleh sensor terhadap alat ukur referensi HTC-2. Pengujian masing-masing parameter dilakukan sebanyak 10 kali pengukuran. Hasil pengujian suhu dan kelembapan disajikan pada Tabel 3 dan Tabel 4.

TABEL 3
HASIL PENGUJIAN SUHU BME280

No	BME280	Referensi (HTC-2)	Selisih	Deviasi (%)	Akurasi (%)
1	31,0	31,5	0,5	1,59	98,41
2	30,9	31,2	0,3	0,96	99,04
3	30,9	31,1	0,2	0,64	99,36
4	30,9	31,0	0,1	0,32	99,68
5	30,9	30,9	0,0	0,00	100,00
6	30,9	30,8	0,1	0,32	99,68
7	30,9	30,2	0,7	2,32	97,68
8	30,8	30,1	0,7	2,33	97,67
9	30,8	30,1	0,7	2,33	97,67
10	30,8	30,0	0,8	2,67	97,33
Rata-rata	30,88	30,69	0,41	1,35	98,65

Data tersebut menunjukkan bahwa hasil pembacaan suhu BME280 memiliki rata-rata deviasi sebesar 1,35% dengan tingkat akurasi 98,65%.

TABEL 4
HASIL PENGUJIAN KELEMBAPAN BME280

No	BME280	Referensi HTC-2	Selisih	Deviasi (%)	Akurasi (%)
1	60,3	59,0	1,3	2,20	97,80
2	60,5	59,0	1,5	2,54	97,46
3	60,5	59,0	1,5	2,54	97,46
4	60,5	59,0	1,5	2,54	97,46
5	60,5	61,0	0,5	0,82	99,18
6	60,5	61,0	0,5	0,82	99,18
7	60,7	61,0	0,3	0,49	99,51
8	60,4	61,0	0,6	0,98	99,02
9	60,3	61,0	0,7	1,15	98,85
10	60,2	61,0	0,8	1,31	98,69
Rata-rata	60,44	60,00	0,92	1,54	98,46

Pada parameter kelembapan, diperoleh rata-rata deviasi sebesar 1,54% dengan tingkat akurasi 98,46%.

Berdasarkan hasil pengujian kedua parameter, tingkat akurasi sensor BME280 terbukti berada di atas 98%. Selisih pengukuran yang diperoleh terhadap alat ukur standar masih dalam batas wajar akibat adanya faktor toleransi komponen serta perbedaan karakteristik waktu respons. Dengan performa yang tinggi tersebut, modul BME280 dinilai sangat

layak diimplementasikan sebagai unit pemantau lingkungan pada arsitektur sistem kendali AC berbasis IoT.

E. Hasil User Acceptance Testing (UAT)

Pengujian User Acceptance Testing (UAT) dilakukan untuk menilai penerimaan pengguna terhadap sistem kendali AC berbasis IoT. Satu staf IT bertindak sebagai pengguna utama, mencoba fungsi sistem melalui dashboard web, lalu memberikan penilaian dengan skala 0–10.

TABEL 5
HASIL USER ACCEPTANCE TETSING

No	Pertanyaan	Nilai
1	Sistem mudah dipahami dan digunakan.	9
2	Tampilan antarmuka sistem mudah dipelajari.	8
3	Fitur pengelolaan unit AC berjalan sesuai kebutuhan.	9
4	Fitur pengendalian AC (menyalakan, mematikan, dan mengubah suhu) berfungsi dengan baik.	10
5	Fitur penjadwalan otomatis AC mudah digunakan.	9
6	Informasi suhu dan kelembapan yang ditampilkan sistem mudah dipahami.	8
7	Riwayat aktivitas AC membantu proses monitoring perangkat.	9
8	Sistem membantu pekerjaan pengelolaan AC menjadi lebih efektif.	10
9	Kinerja sistem secara keseluruhan telah memenuhi kebutuhan operasional pengguna.	9
10	Fitur pengaturan batas suhu ruangan (<i>temp_threshold</i>) berfungsi sesuai kebutuhan	10
11	Status perangkat IoT (online/offline) mudah dipantau melalui sistem.	9
12	Proses login dan logout sistem berjalan dengan baik dan aman.	10
13	Notifikasi dan pesan yang ditampilkan sistem mudah dipahami.	9
14	Sistem memberikan respons yang cepat terhadap perintah pengendalian AC.	10
15	Kinerja sistem secara keseluruhan telah memenuhi kebutuhan operasional pengguna.	9

Berdasarkan hasil penilaian, diperoleh total skor 138 dari skor maksimal 150, sehingga tingkat penerimaan pengguna mencapai 92%.

Hasil UAT membuktikan bahwa sistem kendali AC berbasis IoT ini memperoleh respons sangat positif dari pengguna. Seluruh fungsionalitas sistem terkonfirmasi berjalan optimal dalam mempermudah manajemen pendingin ruangan. Dibandingkan metode manual konvensional, pengguna menilai sistem ini mampu meningkatkan efektivitas pemantauan dan pengendalian suhu secara signifikan. Dengan demikian, platform IoT yang dikembangkan telah memenuhi seluruh spesifikasi kebutuhan pengguna serta dinyatakan layak untuk diimplementasikan pada lingkungan operasional perusahaan.

F. Hasil Pengujian Kendali Otomatis

Pengujian ini dilakukan untuk memverifikasi akurasi pengambilan keputusan aktual pada mikrokontroler ESP32 berdasarkan aturan (*rules*) yang telah dirancang. Validasi dilakukan dengan mengamati *output serial monitor* pada tiga skenario kondisi lingkungan yang representatif, yaitu tren suhu naik, tren suhu turun, dan kondisi suhu ekstrem (*boost mode* aktif) sebagaimana divisualisasikan pada Gambar 6.

(a) Kondisi tren suhu naik. Log menunjukkan target suhu diperbarui menjadi 26.00, sensor suhu 31.1°C, kelembapan 67.7%, tekanan 1009.8 hPa. Log juga mencatat perintah turunan ke 26.00 dan status AC1 ON, AC2 OFF.

(b) Kondisi tren suhu turun dan fungsi restore. Log menunjukkan target suhu diperbarui menjadi 25.00, sensor suhu 31.1°C, kelembapan 68.0%, tekanan 1009.8 hPa. Log mencatat perintah turunan ke 25.00 dan status AC1 ON, AC2 OFF.

(c) Kondisi suhu sangat tinggi dengan mode boost aktif. Log menunjukkan target suhu diperbarui menjadi 24.00, sensor suhu 30.4°C, kelembapan 92.4%, tekanan 1008.5 hPa. Log mencatat perintah turunan ke 24.00 dan status AC1 ON, AC2 OFF.

Gambar 6. Hasil pengujian serial monitor ESP32 pada: (a) kondisi tren suhu naik, (b) kondisi tren suhu turun dan fungsi restore, serta (c) kondisi suhu sangat tinggi dengan mode boost aktif.

Pada pengujian kondisi tren naik dengan parameter ambang batas (*temp_threshold*) disetel pada 25.0°C, sistem secara responsif mendeteksi kenaikan suhu ruangan melampaui batas atas. Berdasarkan log aktivitas pada Gambar 6(a), ESP32 berhasil menurunkan variabel suhu target (*target_temp*) secara bertahap (dari 27.0°C hingga 24.0°C) disertai pengiriman sinyal inframerah (IR) pada setiap siklus evaluasi untuk mempercepat pendinginan.

Sebaliknya, pada kondisi tren turun dan stabil dengan batas atas 30.0°C, data pada Gambar 6(b) menunjukkan bahwa sistem secara adaptif menahan transmisi sinyal IR meskipun suhu aktual masih berada di atas ambang batas. Fitur kontrol otomatis ini juga menunjukkan mekanisme pengaman (*restore function*), di mana sistem menerapkan jeda

konfirmasi satu siklus dan penghitung waktu (*timer*) selama 5 menit sebelum mengembalikan fungsi AC ke pengaturan awal setelah suhu ruang berhasil diturunkan di bawah batas minimum.

Terakhir, pada pengujian kondisi suhu sangat tinggi saat suhu melampaui batas kritis ($\geq$ batas atas + 2°C), *output* pada Gambar 6(c) membuktikan bahwa sistem secara konsisten mempertahankan status *boost* aktif pada AC pasangan. Sistem berhasil mencegah redundansi data dengan tidak memicu pengiriman ulang sinyal inframerah yang tidak diperlukan selama status kondisi lingkungan tidak mengalami fluktuasi perubahan. Seluruh hasil pengujian ini menegaskan bahwa algoritma kontrol berbasis aturan yang ditanamkan pada ESP32 memiliki tingkat keandalan yang tinggi dalam mengotomatisasi pendingin ruangan.

IV. KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem kendali AC berbasis Internet of Things (IoT) dengan memanfaatkan mikrokontroler ESP32, sensor BME280, dan komunikasi inframerah. Sistem yang dikembangkan menyediakan dashboard berbasis web yang terintegrasi dengan fungsi pengendalian manual maupun otomatis, pemantauan suhu dan kelembapan secara real-time, serta pencatatan riwayat aktivitas.

Hasil pengujian menunjukkan kinerja sistem yang baik, dengan rata-rata waktu respons 0,32 detik, tingkat keberhasilan pengiriman sinyal inframerah mencapai 100%, serta akurasi sensor suhu dan kelembapan masing-masing sebesar 98,65% dan 98,46%. Evaluasi penerimaan pengguna melalui User Acceptance Testing (UAT) memperoleh nilai 92%, yang menandakan sistem diterima dengan sangat baik.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada Politeknik Negeri Batam, khususnya Program Studi Teknologi Rekayasa Perangkat Lunak, atas dukungan akademik dalam pelaksanaan penelitian ini. Penulis juga berterima kasih kepada Bapak Hamdani Arif, S.Pd., M.Sc. selaku dosen pembimbing atas arahan dan bimbingan yang diberikan, serta pihak PT Patlite Indonesia yang telah menyediakan fasilitas dan data penelitian. Dukungan dari rekan-rekan selama proses penelitian turut memberikan kontribusi berharga.

DAFTAR PUSTAKA

- [1] A. Ariefudin, B. Darmawan, and Supriono, "Rancang bangun purwarupa sistem kontrol suhu dan monitoring kelembaban udara berbasis IoT," *Jurnal Teknik Elektro*, vol. 12, no. 2, pp. 45–56, 2024.
- [2] N. A. R. Muhammad and M. R. Amirul, "Perancangan sistem kontrol dan monitoring Air Conditioner ruangan berbasis Internet of Things," *Jurnal INFORMA Politeknik Indonusa Surakarta*, 2024.
- [3] A. A. Prasetyo, R. Arifin, and R. I. Vidyastri, "Sistem otomatisasi AC berbasis IoT untuk pengendalian suhu ruangan secara optimal," *Jurnal Teknik Informatika*, vol. 15, no. 3, pp. 78–89, 2025.
- [4] R. D. Romadhona, M. F. Rahmat, and V. Nugroho, "Environmental sensing sebagai informasi suhu, kelembapan, dan gas berbasis IoT menggunakan ESP32," *Jurnal Sensor Indonesia*, vol. 10, no. 4, pp. 112–125, 2024.
- [5] R. Sari, D. Nugroho, and A. Prasetyo, "Iterative and incremental methods for IoT product development," *Journal of Applied Engineering and Technology*, vol. 10, no. 2, pp. 45–54, 2025.
- [6] Sugiyono, *Metode penelitian kuantitatif, kualitatif, dan R&D*. Bandung: Alfabeta, 2022.
- [7] J. W. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*, 5th ed. Thousand Oaks, CA: SAGE Publications, 2021.
- [8] D. Sinaga, D. B. Sirait, K. K. Pakpahan, M. Sihombing, and H. Hutauruk, "Design and implementation of an ESP32-based IoT smart room system with automated temperature, ventilation, and lighting control," *International Journal of Smart Home Technology*, vol. 14, no. 1, pp. 34–48, 2026.
- [9] A. V. Rachmawati, Dzulkifli, and M. Yantidewi, "Analisis kalibrasi sensor BME280 dengan pendekatan regresi linear pada pengukuran temperatur, kelembaban relatif, dan titik embun," *Jurnal Kolaboratif Sains*, vol. 7, no. 5, 2023.
- [10] Bosch Sensortec, BME280 Combined Humidity, Pressure, and Temperature Sensor Datasheet, Document revision 1.24, 2024.
- [11] G. A. Mutiara, P. Periyadi, M. R. Alfarisi, M. A. R. Zain, M. G. Rijali, and F. N. Rochim, "Design and implementation of a REST API-based client-server architecture for IoT monitoring," *International Journal of Advanced Technology and Engineering Exploration*, vol. 12, no. 124, 2025.
- [12] A. Aris, J. L. Mutiawati, D. A. Fitria, E. Nurkhotimah, and A. Baihaqi, "Monitoring air conditioner (AC) ruangan server menggunakan sensor infrared berbasis ESP32," *Jurnal IoT Indonesia*, vol. 8, no. 1, pp. 23–34, 2025.
- [13] J. Smith, K. Lee, and R. Patel, "Design and application of hysteresis control in HVAC thermostat systems," *Energy and Buildings*, vol. 310, p. 112450, 2025.
- [14] Y. Huang and B. Hughes, "Review of HVAC forecasting and control strategies," *Building and Environment*, 2026.
- [15] "Are rule-based techniques enough for occupant-centric control? Trends, key challenges, and future opportunities in HVAC systems," *Energy and Buildings*, vol. 360, p. 117415, Elsevier, 2026.