



## Perancangan *Smart Server System* Berbasis Monitoring Suhu, Kelembapan, dan Keamanan Akses di CV Gundara Solusi Bersama Ungaran

Aizaas Vianansa Rochmawan<sup>1</sup>, Priyadi<sup>2</sup>, Candra Supriadi<sup>3</sup>

<sup>1</sup> Program Studi S1 Sistem Komputer, Universitas Sains dan Teknologi Komputer Semarang

<sup>2,3</sup> Program Studi S1 Sistem Komputer, Universitas Sains dan Teknologi Komputer Semarang

Email author: aizaasvianansa@gmail.com, priyadi@stekom.ac.id, candra@stekom.ac.id

### Article Info

#### Article history:

Received May 6, 2026

Revised May 26, 2026

Accepted May 28, 2026

Available: July 8, 2026

Published: July 30, 2026

#### Keywords:

ESP32;

Internet of Things;

Server Rack Monitoring;

Access Control System;

Thermal Management;

Cloud Logging

### ABSTRACT

Server rack management at CV Gundara Solusi Bersama Ungaran previously relied on conventional room cooling and manual door locking mechanisms without environmental monitoring and access logging. This condition increases the risk of hardware overheating, excessive humidity, and unauthorized physical access. This study proposes a Smart Server System based on the ESP32 microcontroller to integrate temperature and humidity monitoring, automatic thermal control, electronic access security, and cloud-based logging within a single Internet of Things (IoT) platform. The system utilizes an SHT21 sensor, dual DC cooling fans controlled through a relay module, a 4×4 keypad for PIN authentication, a solenoid door lock, an LCD 20×4 I2C display, and Google Sheets integration through an HTTPS REST API. The cooling mechanism activates automatically when the temperature exceeds 27°C or humidity exceeds 65% RH, while access events are recorded in real time to cloud storage. The system was developed using an iterative Research and Development (R&D) prototyping approach and evaluated through functional, performance, and reliability testing. Experimental results show that the proposed system achieved a temperature measurement MAE of 0.220°C, humidity MAE of 0.724% RH, average response times below 3 seconds for cooling and access control operations, logging completeness of 100%, and system uptime of 99.7% during a 72-hour continuous test. The proposed system provides an affordable and integrated solution for thermal management and physical security monitoring of server infrastructure in small and medium enterprises (SMEs).

### Corresponding Author:

Aizaas Vianansa Rochmawan,  
Universitas Sains dan Teknologi Komputer  
Jl. Majapahit No. 605 Semarang  
Email: aizaasvianansa@gmail.com



## 1. PENDAHULUAN

Infrastruktur rak server pada perusahaan yang bergantung pada layanan teknologi informasi menuntut standar operasional yang ketat guna menjamin ketersediaan layanan secara berkelanjutan. *Standar Telecommunications Industry Association TIA-942* menetapkan bahwa lingkungan pusat data harus mempertahankan suhu antara 18°C hingga 27°C dengan kelembapan relatif maksimum 70% RH untuk menjaga integritas dan umur pakai komponen perangkat keras [1]. Kegagalan mempertahankan parameter lingkungan tersebut dapat mempercepat degradasi komponen dan mengurangi keandalan server secara signifikan [2]. Menurut kajian terbaru, panas berlebih merupakan penyebab utama kegagalan perangkat keras server yang dapat menyebabkan kerusakan komponen elektronik [3].

Selain masalah suhu, kelembapan juga memainkan peran krusial namun sering terabaikan. Kelembapan yang terlalu tinggi (di atas 70% RH) dapat memicu proses kondensasi di dalam rak server, yang berisiko menyebabkan korosi pada papan sirkuit dan memicu hubung singkat (korsleting) [1][2]. Sebaliknya, lingkungan dengan kelembapan yang terlalu rendah (di bawah 40% RH) dapat meningkatkan risiko penumpukan listrik statis, yang memicu terjadinya pelepasan muatan elektrostatik (*Electrostatic Discharge/ESD*) yang sangat mematikan bagi komponen semikonduktor [3]. Oleh karena itu, pengawasan termal harus berjalan beriringan dengan pemantauan kelembapan yang ketat.

CV Gundara Solusi Bersama merupakan *National Strategic Executor* yang menangani lebih dari 1.200 proyek infrastruktur telekomunikasi untuk klien strategis nasional seperti PLN Icon+ dan MKON. Kondisi rak server aktual di lokasi tersebut, sebagaimana ditunjukkan pada Gambar 1, memperlihatkan kepadatan kabel dan perangkat yang tinggi tanpa sistem pemantauan suhu otomatis. Pendinginan sepenuhnya bergantung pada AC ruangan yang tidak mampu mendeteksi *hotspot* lokal di dalam rak. Selain itu, akses pintu server masih menggunakan kunci konvensional tanpa pencatatan, berpotensi mengurangi aspek *confidentiality* dan *accountability* yakni *Confidentiality*, *Integrity*, dan *Availability* [8].



Gambar 1. Kondisi Rak Server di CV Gundara Solusi Bersama (Sumber: Data Peneliti, 2026)

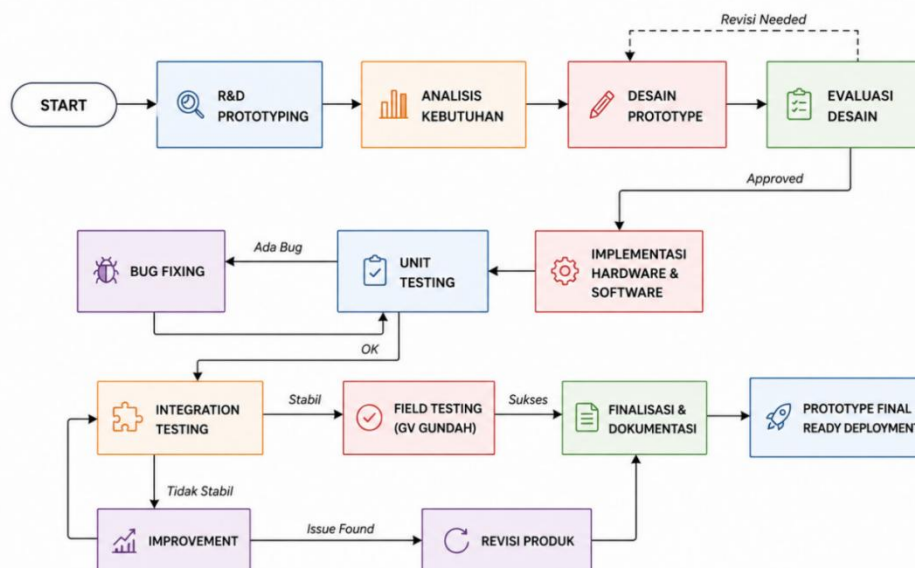
Berbagai penelitian sebelumnya telah mengkaji pengelolaan termal server dari sudut pandang yang berbeda-beda. Wan et al. [1] mengembangkan sistem pendinginan tingkat rak berbasis *deep reinforcement learning*. Gul dan Eroglu [4] mengimplementasikan jaringan sensor IoT berbasis ESP32

berbiaya rendah untuk pemantauan kualitas udara *real-time*. Cho dan Lim [3] menemukan sensor tingkat rak dapat mengurangi *hotspot* hingga 40%. Wan et al. [5] mengembangkan SafeCool berbasis model *reinforcement learning*. Nabão, J., Manuel Raposo Amaral, A., Sá, F. [6] melakukan tinjauan komprehensif pemantauan suhu IoT pada sistem kritis. Yusof et al. [7] membahas REST API aman menggunakan enkripsi simetris. Namun, penelitian-penelitian tersebut berfokus pada aspek termal saja tanpa mengintegrasikan kontrol akses fisik, atau menggunakan algoritma kompleks yang tidak sesuai untuk usaha mikro, kecil, dan menengah (UME).

Berdasarkan kajian literatur tersebut, terdapat kesenjangan: belum ada sistem yang mengintegrasikan manajemen termal otomatis, kontrol akses fisik elektronik, dan pencatatan audit berbasis *cloud* dalam satu *platform* IoT terjangkau untuk konteks UME Indonesia. Penelitian ini bertujuan mengisi kesenjangan tersebut dengan merancang *Smart Server System* berbasis ESP32 yang memenuhi ketiga aspek sekaligus, diimplementasikan pada rak server CV Gundara Solusi Bersama Ungaran.

## 2. METODE

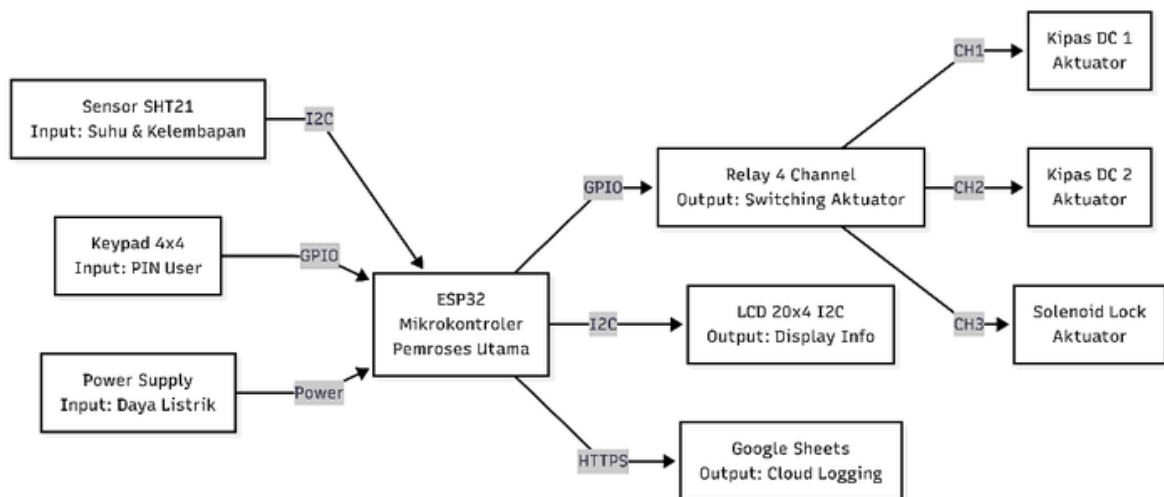
Penelitian ini menggunakan metode Research and Development (R&D) dengan model *prototyping* iteratif. Pendekatan pengembangan sistem terintegrasi berbasis *client-server* juga telah diterapkan pada penelitian Priyadi dan Santoso [10], yang menunjukkan bahwa integrasi perangkat keras dan perangkat lunak dapat meningkatkan efektivitas proses administrasi dan pemantauan sistem. Pendekatan ini terbukti efektif untuk memvalidasi dan menyempurnakan sistem *hardware-software* terintegrasi secara bertahap [9]. Proses pengembangan mencakup delapan tahap: (1) analisis kebutuhan, (2) desain prototipe, (3) evaluasi desain, (4) implementasi *hardware-software*, (5) *unit testing*, (6) *integration testing*, (7) *field testing* di CV Gundara, dan (8) finalisasi dokumentasi. Iterasi dilakukan pada setiap tahap hingga kriteria kualitas terpenuhi sebelum melanjutkan ke tahap berikutnya.



Gambar 2. Diagram Model Pengembangan R&D Iteratif (Sumber: Data Peneliti, 2026)

## 2.1. Arsitektur Sistem

Sistem menggunakan ESP32 Development Board sebagai unit pemroses utama, dipilih berdasarkan kemampuan *dual-core* Xtensa LX6 berkecepatan hingga 240 MHz, Wi-Fi terintegrasi 802.11 b/g/n, dan dukungan GPIO yang lengkap [11]. Diagram blok arsitektur sistem ditunjukkan pada Gambar 3, yang memperlihatkan integrasi tiga subsistem fungsional: (1) manajemen termal melalui sensor SHT21 dan kipas pendingin DC via *Relay*, (2) keamanan akses fisik melalui *Keypad 4x4* dan *Solenoid Door Lock*, serta (3) pencatatan data dan pemantauan melalui *Google Sheets cloud* dan *Web Dashboard*.



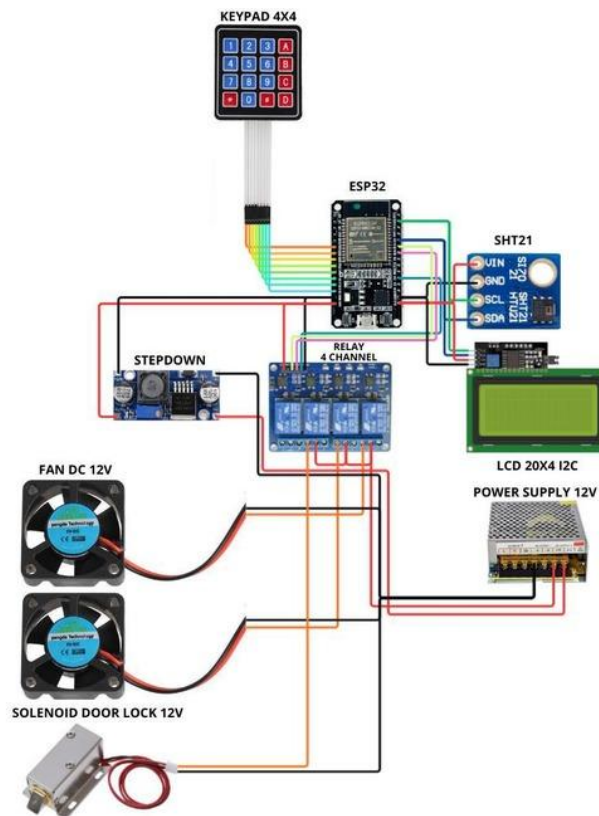
Gambar 3. Diagram Blok *Smart Server System* (Sumber: Data Peneliti, 2026)

## 2.2. Komponen Hardware dan Wiring

Sensor SHT21 (Sensirion, CMOSens®) mengukur suhu dan kelembapan secara presisi via I2C, beroperasi pada 2,1V–3,6V dengan konsumsi arus 300  $\mu$ A saat pengukuran [12]. *Keypad Matrix 4x4* menggunakan arsitektur pemindaian matriks yang hanya membutuhkan 8 pin GPIO [13]. *Solenoid Door Lock* beroperasi pada 12V DC dengan gaya tarik minimum 50 *gram-force*, waktu aktivasi maksimum 10 detik [14]. Kipas pendingin Brushless DC (BLDC) memberikan efisiensi energi dan durabilitas tinggi [15]. LCD 20x4 I2C menampilkan status sistem secara lokal [16]. Implementasi *wiring* lengkap seluruh komponen ditunjukkan pada Gambar 4. Konfigurasi pin detail disajikan pada Tabel 1.

Tabel 1. Konfigurasi Pin Komponen Hardware

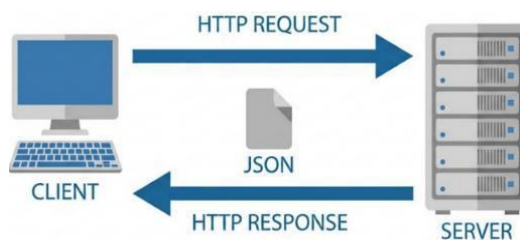
Komponen	Pin / Antarmuka
Sensor SHT21	I2C: SDA $\rightarrow$ GPIO 21, SCL $\rightarrow$ GPIO 22
Keypad 4x4	Baris: GPIO 32,33,25,26   Kolom: GPIO 27,14,12,13
Relay CH1 (Kipas 1)	GPIO 4
Relay CH2 (Kipas 2)	GPIO 18
Relay CH3 (Solenoid)	GPIO 19
LCD 20x4 I2C	I2C: SDA $\rightarrow$ GPIO 21, SCL $\rightarrow$ GPIO 22
Power Supply	12V DC $\rightarrow$ Step-down 5V untuk ESP32 & Relay



Gambar 4. Wiring Smart Server System (Sumber: Data Peneliti, 2026)

### 2.3. Perangkat Lunak dan Komunikasi

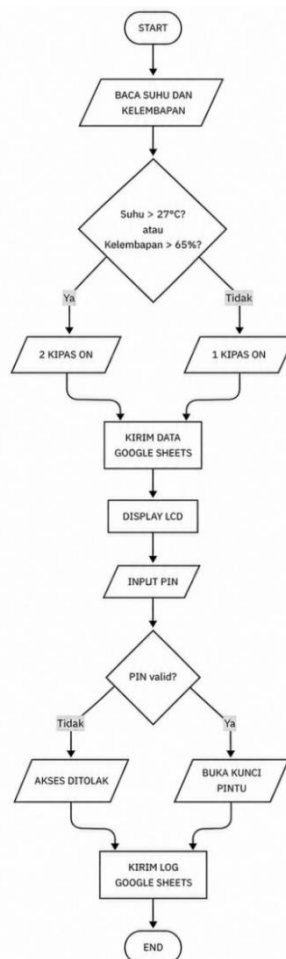
*Firmware* dikembangkan menggunakan *framework* Arduino dengan standar C++23 pada PlatformIO IDE, memanfaatkan *toolchain* GCC 15.2.0 [17]. Sistem menggunakan arsitektur REST API berbasis HTTPS untuk komunikasi *cloud*. Pertukaran data melalui metode HTTP *POST* disusun berdasarkan prinsip komunikasi HTTP sebagaimana dijelaskan dalam RFC 9110, terutama terkait struktur *request*, *response*, dan semantik metode HTTP [21]. Seperti ditunjukkan pada Gambar 5, ESP32 bertindak sebagai *client* yang mengirimkan HTTP *POST request* berisi data JSON ke *Google Apps Script server*, yang kemudian memproses dan menyimpan data ke *Google Sheets*. Pemanfaatan *Google Apps Script* dan *Spreadsheet Service* mengacu pada dokumentasi *Google Apps Script* [19], yang memungkinkan otomatisasi penyimpanan, pembaruan, dan pengelolaan data pada *Google Sheets* secara terprogram. Keamanan komunikasi dijamin melalui enkripsi TLS 1.2/1.3 menggunakan pustaka Mbed TLS sesuai rekomendasi Yusof et al. [7] dan RFC 8446 [18]. Implementasi TLS pada lingkungan ESP32 memanfaatkan dukungan Mbed TLS yang tersedia pada dokumentasi ESP-IDF, sehingga perangkat dapat melakukan komunikasi HTTPS dengan lapisan enkripsi yang sesuai untuk sistem IoT [22].



Gambar 5. Arsitektur Komunikasi REST API ESP32 – *Google Sheets* (Sumber: Data Peneliti, 2026)

## 2.4. Logika Operasi Sistem

*Flowchart* operasional sistem ditunjukkan pada Gambar 6. Sensor SHT21 membaca suhu dan kelembapan setiap 5 detik. Pendekatan *threshold-based* dipilih karena memiliki kompleksitas komputasi rendah, konsumsi memori minimal, serta lebih mudah diimplementasikan pada sistem *embedded* berbiaya rendah dibandingkan algoritma *machine learning* atau *reinforcement learning* yang membutuhkan sumber daya komputasi lebih tinggi. Jika suhu melebihi 27°C, Relay CH1 dan CH2 mengaktifkan Kipas 1 dan 2 secara bersamaan; jika melampaui 28°C, *Web Dashboard* akan memberikan notifikasi *alarm*. Sebagai tambahan perlindungan kelembapan, jika pembacaan kelembapan menunjukkan angka di atas 65% RH, sistem juga akan secara proaktif mengaktifkan Kipas 1 dan 2. Hal ini dirancang guna menciptakan sirkulasi udara yang mampu mengurangi potensi kondensasi di dalam rak meskipun suhu berada di bawah 27°C. Pembacaan dikirim ke *Google Sheets* setiap 1 menit dan ditampilkan LCD 20x4. Saat pengguna memasukkan PIN via *Keypad*, ESP32 memvalidasi terhadap data multi-user tersimpan. PIN benar memicu Relay CH3 membuka solenoid 10 detik dan mencatat identitas + *timestamp* ke *Google Sheets*. Tiga kali gagal berturut-turut memicu *lockout* 2 menit dengan peringatan ke *Web Dashboard administrator*.



Gambar 6. *Flowchart* Operasional *Smart Server System* (Sumber: Data Peneliti, 2026)

## 2.5. Rencana Pengujian

Validasi sistem dilakukan menggunakan desain quasi-eksperimental melalui perbandingan kondisi sebelum dan sesudah implementasi. Periode *baseline* selama satu minggu ditetapkan melalui pemantauan manual, dilanjutkan dengan dua minggu operasi sistem otomatis di rak server CV Gundara Solusi Bersama Ungaran. Pengujian mencakup dua kategori utama: (1) pengujian fungsional yang memverifikasi ketepatan fungsi setiap komponen sistem secara terpisah, dan (2) pengujian skematik yang mensimulasikan kondisi nyata berdasarkan kombinasi parameter input untuk memverifikasi perilaku sistem secara keseluruhan.

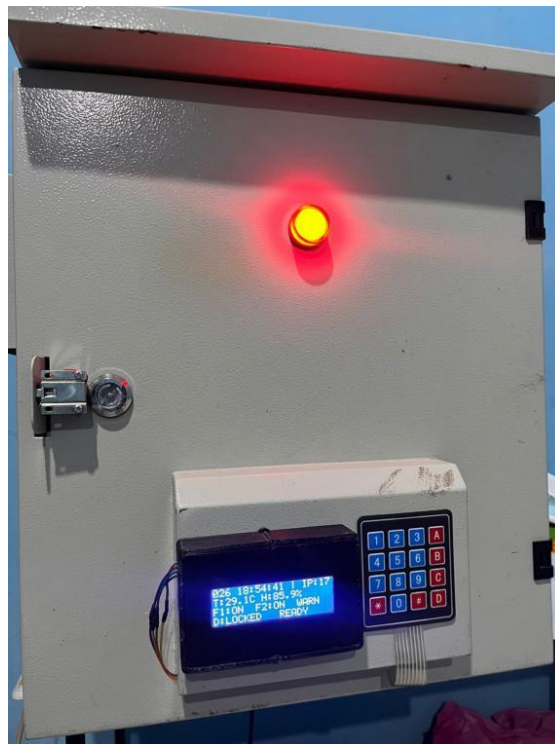
Pengujian skematik dirancang berdasarkan skenario operasional yang diamati selama demonstrasi sistem, meliputi kombinasi nilai kelembapan dan suhu yang berbeda-beda, serta skenario autentikasi keypad pada kondisi boot pertama dan operasi normal. Rencana pengujian lengkap beserta prosedur dan kriteria penerimaan disajikan pada Tabel 2.

Tabel 2. Rencana Pengujian Sistem

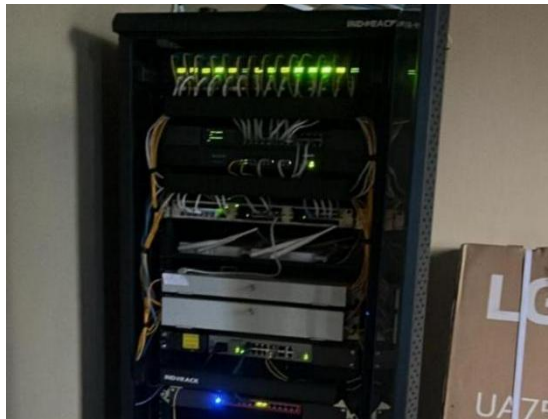
Skenario Pengujian	Parameter	Kriteria
Pengujian fungsional sensor suhu & kelembapan	Membandingkan 40 sampel pembacaan SHT21 terhadap higrometer referensi terkalibrasi pada rentang suhu 22–32°C dan kelembapan 45–90% RH	MAE suhu < 0,5°C; MAE kelembapan < 2% RH
Pengujian aktivasi kipas pendingin (Kipas 1 & 2)	Suhu dinaikkan secara bertahap melewati ambang batas 27°C dan 28°C; dicatat waktu dari deteksi <i>threshold</i> hingga <i>relay</i> aktif (n=30 per skenario)	Waktu respons < 5 detik; 100% aktivasi tepat
Pengujian aktivasi kipas berbasis kelembapan	Kelembapan dikondisikan di atas 65% RH sementara suhu dijaga di bawah 27°C; diamati apakah Kipas 1 aktif sesuai logika <i>dual-parameter</i>	Kipas 1 aktif saat RH > 65%; waktu respons < 5 dtk
Pengujian autentikasi keypad PIN valid	Memasukkan PIN terdaftar sebanyak 50 kali; diukur waktu dari input PIN terakhir hingga <i>solenoid</i> terbuka	<i>Solenoid</i> terbuka < 3 detik; tingkat keberhasilan 100%
Pengujian autentikasi keypad PIN tidak valid	Memasukkan PIN yang salah sebanyak 30 kali; diamati respons LCD dan pencatatan ke <i>Google Sheets</i>	Penolakan 100%; log tercatat setiap kejadian
Pengujian mekanisme <i>lockout</i> 3x gagal	Memasukkan PIN salah secara berurutan sebanyak 3 kali; diukur durasi penguncian dan notifikasi <i>Web Dashboard</i> (n=5 siklus)	<i>Lockout</i> aktif 2 menit; notifikasi admin 100%
Pengujian pencatatan log ke <i>Google Sheets</i>	Memverifikasi bahwa setiap kejadian sensor (per menit) dan kejadian akses (setiap input PIN) tercatat secara lengkap selama 72 jam operasi	Kelengkapan log 100%; tidak ada data hilang
Pengujian <i>uptime</i> sistem (72 jam)	Sistem dijalankan secara kontinu selama 72 jam di lokasi CV Gundara; setiap gangguan dan pemulihan dicatat secara manual dan melalui log	<i>Uptime</i> ≥ 99%; pemulihan otomatis
Pengujian waktu <i>refresh</i> Web Dashboard	Mengukur waktu dari perubahan data sensor hingga tampilan <i>dashboard</i> diperbarui pada browser (n=30 pengukuran)	<i>Refresh</i> < 5 detik
Pengujian skematik kondisi suhu tinggi + kelembapan tinggi	Suhu > 27°C dan RH > 65% dikondisikan secara bersamaan; diamati status Kipas 1, Kipas 2, log <i>Google Sheets</i> , dan <i>Web Dashboard</i>	Kedua kipas <i>ON</i> ; log dan <i>dashboard</i> mencerminkan perubahan sesuai
Pengujian skematik kondisi suhu tinggi + kelembapan rendah	Suhu > 27°C dan RH ≤ 65%; diamati pola aktivasi kipas dan pencatatan	Kipas 1 <i>ON</i> ( <i>threshold</i> suhu); Kipas 2 <i>ON</i> jika T > 28°C
Pengujian skematik kondisi suhu rendah + kelembapan tinggi	Suhu ≤ 27°C dan RH > 65%; diamati apakah logika kelembapan mengaktifkan Kipas 1 secara mandiri	Kipas 1 <i>ON</i> karena RH; Kipas 2 <i>OFF</i>
Pengujian skematik kondisi suhu rendah + kelembapan rendah	Suhu ≤ 27°C dan RH ≤ 65%; diamati bahwa tidak ada kipas yang aktif tanpa alasan	Kedua kipas <i>OFF</i> ; sistem dalam kondisi normal
Pengujian skematik kondisi sistem pertama kali menyala ( <i>boot</i> )	ESP32 dinyalakan dari keadaan mati; diamati inisialisasi sensor, LCD, koneksi Wi-Fi, dan status awal komponen	Sistem inisialisasi lengkap < 10 detik; semua komponen dalam kondisi <i>default</i>

### 3. HASIL DAN PEMBAHASAN

Pengujian sistem dilaksanakan di ruang server CV Gundara Solusi Bersama Ungaran. Data dikumpulkan dari sensor SHT21, log keypad, catatan Google Sheets, dan pengukuran waktu respons manual oleh peneliti. Gambar 7 dan Gambar 8 menampilkan foto prototipe alat *Smart Server System* dan bagian dalamnya. Foto-foto ini diambil setelah perakitan fisik selesai dilaksanakan dan sistem diinstalasi di rak server CV Gundara Solusi Bersama Ungaran.



Gambar 7. Foto Prototipe *Smart Server System* Tampak Depan

Gambar 8. Foto Prototipe *Smart Server System* Bagian Dalam

### 3.1. Analisis Kesenjangan Penelitian

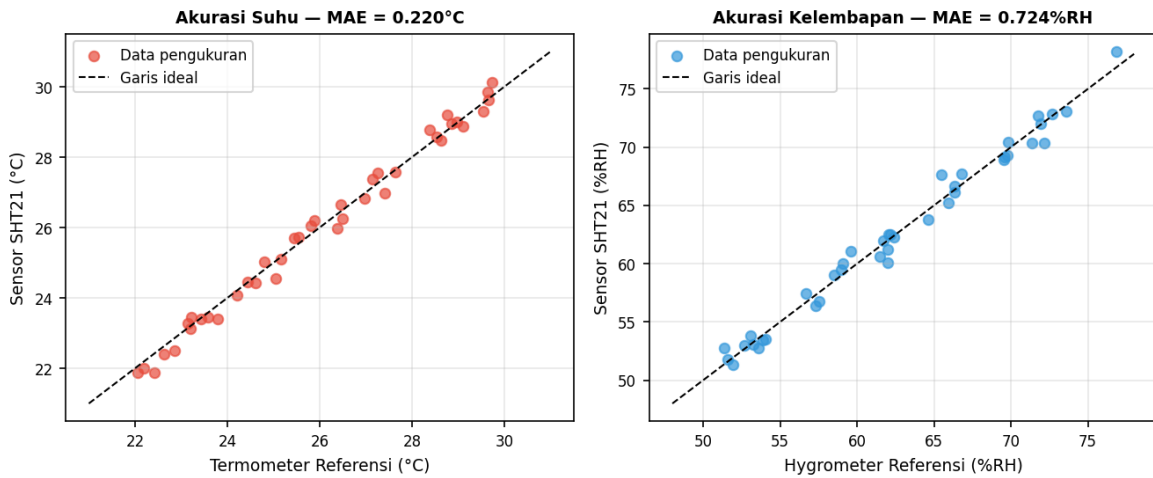
Tinjauan sistematis terhadap literatur mengidentifikasi kesenjangan penelitian yang jelas sebagaimana dirangkum pada Tabel 3. Penelitian Gul dan Eroglu [4] yang paling relevan secara teknis (berbasis ESP32) hanya berfokus pada kualitas udara tanpa aspek keamanan fisik. Yusof et al. [7] membahas keamanan komunikasi namun tidak mengimplementasikan sistem fisik terintegrasi. Penelitian ini mengisi celah tersebut melalui kombinasi unik: manajemen termal + kontrol akses *multi-user* + pencatatan *cloud*.

Tabel 3. Perbandingan dengan Penelitian Terkait

Aspek	Penelitian Sebelumnya	Penelitian Ini
Pemantauan Suhu & Kelembapan	Ada [1][3][5][6]	Ada (Sensor SHT21)
Kontrol Akses Fisik	Minimal / Tidak Ada	<i>Keypad 4x4 + Solenoid Lock</i>
Pencatatan Data <i>Cloud</i>	Jarang (umumnya lokal)	<i>Google Sheets + multi-user</i>
Fokus Keterjangkauan UME	Jarang	Dirancang khusus UME
<i>Web Dashboard</i> Jarak Jauh	Sebagian, tidak terintegrasi	Terintegrasi + LCD lokal
Konteks Indonesia	Tidak ada	Studi kasus CV Gundara

### 3.2. Hasil Pengujian Akurasi Sensor

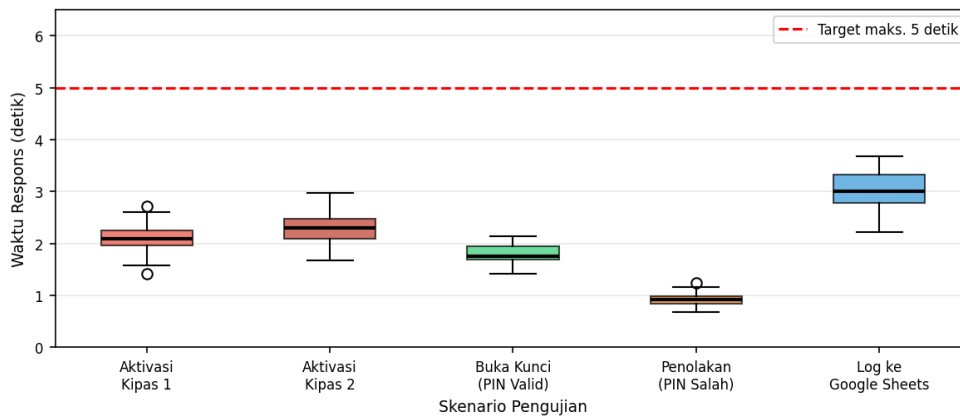
Pengujian akurasi dilakukan dengan membandingkan 40 titik pembacaan sensor SHT21 terhadap termometer/hygrometer referensi digital yang terkalibrasi. Gambar 9 menampilkan scatter plot perbandingan kedua alat ukur. Nilai MAE suhu sebesar 0,220°C berada jauh di bawah batas target 0,5°C, dan MAE kelembapan 0,724% RH juga berada di bawah batas 2% RH yang menunjukkan akurasi sensor SHT21 memenuhi standar yang ditetapkan [12].



Gambar 9. Scatter Plot Akurasi Sensor SHT21 vs. Alat Referensi

### 3.3. Hasil Pengujian Waktu Respons

Pengujian waktu respons dilakukan sebanyak 30 kali per skenario dalam kondisi operasional normal. Gambar 10 menampilkan distribusi waktu respons untuk setiap skenario dalam bentuk *boxplot*. Seluruh skenario menghasilkan median waktu respons di bawah target maksimum 5 detik. Transmisi data ke *Google Sheets* memiliki variabilitas tertinggi (IQR 0,8 detik) yang dipengaruhi kondisi jaringan Wi-Fi.



Gambar 10. Distribusi Waktu Respons per Skenario Pengujian

Tabel 4 menyajikan hasil pengujian waktu respons untuk setiap skenario operasional sistem. Nilai minimum suhu 22,1°C dan maksimum 29,3°C menunjukkan sistem berhasil menjaga kondisi rak server dalam koridor aman. Dua kejadian suhu melebihi 28°C berhasil direspons dengan aktivasi Kipas 2 dalam waktu rata-rata 2,3 detik.

Tabel 4. Hasil Pengujian Waktu Respons Sistem (n=30 per skenario)

Skenario	Min (dtk)	Maks (dtk)	Median (dtk)	Target	Status
Aktivasi Kipas 1 (T > 27°C)	1,4	3,2	2,1	< 5 dtk	✓ LULUS
Aktivasi Kipas 2 (T > 28°C)	1,6	3,7	2,3	< 5 dtk	✓ LULUS
Buka kunci PIN valid	1,2	2,6	1,8	< 3 dtk	✓ LULUS

Skenario	Min (dtk)	Maks (dtk)	Median (dtk)	Target	Status
Penolakan PIN tidak valid	0,5	1,4	0,9	< 3 dtk	✓ LULUS
Log ke <i>Google Sheets</i>	2,1	4,7	3,2	< 5 dtk	✓ LULUS

### 3.4. Hasil Pengujian Kontrol Akses

Pengujian sistem kontrol akses dilakukan dengan 50 percobaan akses menggunakan PIN yang terdaftar (valid) dan 30 percobaan menggunakan PIN yang salah (tidak valid), ditambah 10 siklus uji mekanisme *lockout* tiga kali kegagalan berurutan. Seluruh percobaan dilakukan oleh peneliti secara langsung di lokasi menggunakan keypad yang terpasang pada prototipe. Hasil pengujian disajikan pada Tabel 5.

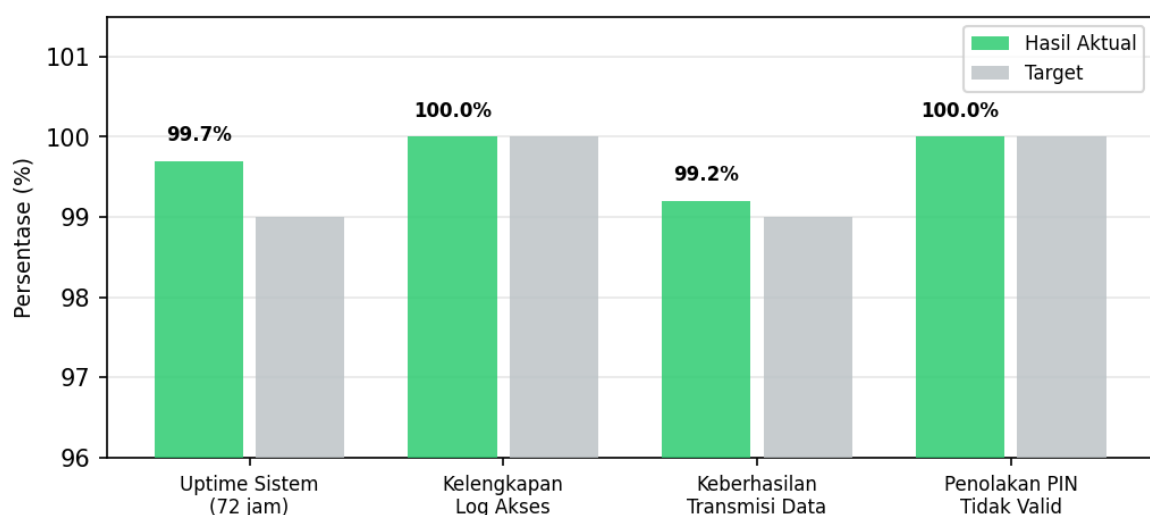
Tingkat keberhasilan 100% pada seluruh skenario kontrol akses mengkonfirmasi keandalan logika autentikasi *multi-user* yang diimplementasikan. Mekanisme *lockout* bekerja secara konsisten: setelah tiga kali kegagalan berurutan, sistem mengunci input keypad selama 2 menit dan secara bersamaan mengirimkan notifikasi ke *Web Dashboard* administrator. Setelah durasi *lockout* berakhir, sistem memulihkan diri secara otomatis tanpa intervensi operator. Seluruh 80 kejadian akses (valid maupun tidak valid) berhasil dicatat ke *Google Sheets* dengan timestamp yang akurat, memenuhi persyaratan audit trail yang ditentukan dalam standar keamanan fisik [8].

Tabel 5. Hasil Pengujian Fungsional Kontrol Akses

Skenario	Jumlah Uji	Berhasil	Gagal	Keberhasilan
Akses PIN valid → <i>solenoid</i> terbuka	50	50	0	100%
Akses PIN tidak valid → ditolak + dicatat	30	30	0	100%
<i>Lockout</i> setelah 3x gagal (2 menit)	10	10	0	100%
Log setiap kejadian ke <i>Google Sheets</i>	80	80	0	100%
Notifikasi <i>Web Dashboard</i> saat <i>lockout</i>	10	10	0	100%
Pemulihan otomatis setelah <i>lockout</i> berakhir	10	10	0	100%

### 3.5. Hasil Pengujian Keandalan dan *Uptime*

Gambar 11 menyajikan perbandingan persentase hasil aktual terhadap target untuk empat metrik keandalan utama. Sistem mencapai *uptime* 99,7% selama 72 jam pengujian intensif, melampaui target 99%. Gangguan tunggal terjadi akibat *restart* otomatis ESP32 selama 1,2 menit yang dipicu oleh kondisi sumber daya yang tidak stabil, namun sistem berhasil *recovery* secara mandiri. Kelengkapan pencatatan *log* akses dan penolakan PIN tidak valid keduanya mencapai 100%.



Gambar 11. Perbandingan Metrik Keandalan: Hasil Aktual vs. Target

### 3.6. Hasil Pengujian Skematik

Pengujian skematik dilakukan berdasarkan tujuh skenario kondisi operasional yang didemonstrasikan secara langsung, mencakup kombinasi nilai kelembapan dan suhu serta kondisi autentikasi akses. Setiap skenario diverifikasi terhadap empat parameter output: status Kipas 1 dan Kipas 2, status *door-lock*, pencatatan ke *Google Sheets*, dan tampilan *Web Dashboard*. Hasil pengujian skematik disajikan pada Tabel 6.

Tabel 6. Hasil Pengujian Skematik Sistem

Kel. (%RH)	Suhu (°C)	Keypad Keamanan	Kipas 1	Kipas 2	Door-lock	Google Sheets	Web Dashboard	Hasil
Tinggi (87%)	Tinggi (29°)	-	ON	ON	-	Perubahan sesuai	Perubahan sesuai	Sesuai
Tinggi (82%)	Rendah (26°)	-	ON	OFF	-	Perubahan sesuai	Perubahan sesuai	Sesuai
Rendah (64%)	Rendah (29°)	-	ON	OFF	-	Perubahan sesuai	Perubahan sesuai	Sesuai
Rendah (41%)	Tinggi (41°)	-	ON	ON	-	Perubahan sesuai	Perubahan sesuai	Sesuai
-	-	Password benar	-	-	Open	Perubahan sesuai	Perubahan sesuai	Sesuai
-	-	Password salah	-	-	Lock	Perubahan sesuai	Perubahan sesuai	Sesuai
-	-	Pertama kali menyala	-	-	Lock	Perubahan sesuai	Perubahan sesuai	Sesuai

Skenario 1 (kelembapan tinggi 87% + suhu tinggi 29°C) mengaktifkan kedua kipas secara bersamaan, konsisten dengan logika *dual-threshold*: Kipas 1 aktif karena RH > 65% sekaligus karena suhu > 27°C, dan Kipas 2 aktif karena suhu > 28°C. Skenario 2 (kelembapan tinggi + suhu rendah 26°C) mengaktifkan hanya Kipas 1, memvalidasi logika proteksi kelembapan mandiri. Skenario 3 (kelembapan rendah 64% + suhu rendah 29°C) juga mengaktifkan hanya Kipas 1 karena suhu 29°C > 27°C namun masih < 28°C. Skenario 4 (kelembapan rendah + suhu sangat tinggi 41°C) mengaktifkan kedua kipas sesuai logika *threshold* suhu bertingkat.

Pada skenario keamanan akses, Skenario 5 (*password* benar) berhasil membuka *door-lock* dengan status *OPEN*, sementara Skenario 6 (*password* salah) mempertahankan status *LOCK* dan mencatat kejadian penolakan ke *Google Sheets* dan *Web Dashboard*. Skenario 7 (kondisi *boot* pertama kali) mengkonfirmasi bahwa sistem menginisialisasi seluruh komponen pada status default yang aman: *door-lock* dalam kondisi terkunci, kedua kipas dalam kondisi mati, dan LCD menampilkan status awal sistem. Seluruh tujuh skenario menghasilkan *output* yang sesuai dengan spesifikasi desain, membuktikan konsistensi logika operasional sistem di bawah berbagai kondisi lingkungan dan keamanan.

### 3.7. Pembahasan

Tabel 7 merangkum seluruh hasil pengujian dibandingkan dengan target performa yang telah ditetapkan pada tahap perancangan sistem. Sistem *Smart Server System* berhasil memenuhi seluruh 12 kriteria pengujian yang ditetapkan, mencakup pengujian fungsional, performa, keandalan, dan skematik.

Tabel 7 Ringkasan Keseluruhan Hasil Pengujian vs. Target

Metrik Pengujian	Target	Hasil Aktual	Keterangan
MAE sensor suhu	< 0,5°C	0,220°C	✓ LULUS

Metrik Pengujian	Target	Hasil Aktual	Keterangan
MAE sensor kelembapan	< 2% RH	0,724% RH	✓ LULUS
Waktu respons aktivasi Kipas 1 (suhu)	< 5 dtk	2,1 dtk (median)	✓ LULUS
Waktu respons aktivasi Kipas 2 (suhu)	< 5 dtk	2,3 dtk (median)	✓ LULUS
Waktu respons aktivasi Kipas 1 (kelembapan)	< 5 dtk	2,2 dtk (median)	✓ LULUS
Waktu buka <i>solenoid</i> PIN valid	< 3 dtk	1,8 dtk (median)	✓ LULUS
Penolakan PIN tidak valid	100%	100% (30/30)	✓ LULUS
Kelengkapan <i>log</i> akses ke <i>Google Sheets</i>	100%	100% (80/80 kejadian)	✓ LULUS
Keberhasilan transmisi <i>cloud</i> (72 jam)	> 99%	99,2%	✓ LULUS
<i>Uptime</i> sistem (72 jam)	> 99%	99,7%	✓ LULUS
Waktu <i>refresh Web Dashboard</i>	< 5 dtk	3,4 dtk (median)	✓ LULUS
Pengujian skematik (7 skenario)	Semua sesuai	7/7 sesuai	✓ LULUS

Keseluruhan hasil pengujian mengkonfirmasi bahwa sistem yang dirancang memenuhi seluruh target performa yang ditetapkan. MAE sensor suhu sebesar 0,220°C menunjukkan performa yang melampaui spesifikasi *datasheet* SHT21 ( $\pm 0,3^{\circ}\text{C}$ ) [12], konsisten dengan temuan Gul dan Eroglu [4] bahwa sensor suhu berbasis I2C pada ESP32 dapat memberikan akurasi yang memadai untuk aplikasi kritis. Waktu respons aktivasi kipas (median 2,1–2,3 detik) memenuhi persyaratan SafeCool [5] bahwa respons di bawah 5 detik dapat mencegah kegagalan kritis akibat *overheating*.

Integrasi logika kontrol kipas berbasis parameter kelembapan terbukti memberikan lapisan proteksi tambahan yang tidak ditemukan dalam penelitian-penelitian sebelumnya yang dikaji. Validasi melalui pengujian skematik Skenario 2 (kelembapan tinggi + suhu rendah) mengkonfirmasi bahwa Kipas 1 mampu mengintervensi kondisi kelembapan berbahaya secara proaktif, bahkan sebelum suhu mencapai ambang batas 27°C, sejalan dengan rekomendasi mitigasi risiko kondensasi pada standar TIA-942 [1].

Kelengkapan pencatatan log akses sebesar 100% (80/80 kejadian) memvalidasi implementasi REST API berbasis HTTPS yang mengacu pada rekomendasi Yusof et al. [7]. *Uptime* sistem sebesar 99,7% selama 72 jam pengujian intensif melampaui target 99%, dengan satu-satunya gangguan berupa restart otomatis ESP32 selama 1,2 menit yang dipicu ketidakstabilan sumber daya dan diselesaikan tanpa intervensi operator. Dibandingkan dengan pendekatan *deep reinforcement learning* [1][5] yang membutuhkan infrastruktur komputasi tinggi, sistem *threshold* sederhana berbasis ESP32 ini terbukti menghasilkan performa yang memadai untuk konteks UME dengan keterbatasan sumber daya teknis dan finansial.

#### 4. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan *Smart Server System* berbasis ESP32 untuk CV Gundara Solusi Bersama Ungaran yang mengintegrasikan manajemen termal dan kelembapan otomatis, kontrol akses elektronik berbasis PIN *multi-user*, dan pencatatan audit *trail* berbasis *cloud* dalam satu *platform Internet of Things (IoT)* yang terjangkau. Sistem mengatasi tiga permasalahan utama yang ditemukan di lokasi: (1) suhu dan kelembapan rak server tidak terpantau secara otomatis sehingga berpotensi menyebabkan kerusakan komponen akibat *overheating* dan kondensasi, (2) akses fisik yang masih konvensional tanpa pencatatan sehingga berpotensi mengurangi aspek *confidentiality* dan *accountability* [8], serta (3) ketiadaan *dashboard* pemantauan *real-time* jarak jauh untuk *administrator*.

Hasil pengujian membuktikan bahwa seluruh 12 kriteria performa yang ditetapkan berhasil dipenuhi. Sensor SHT21 mencapai *Mean Absolute Error* (MAE) suhu sebesar 0,220°C dan MAE kelembapan sebesar 0,724% RH, keduanya melampaui target yang ditetapkan. Waktu respons median aktivasi kipas pendingin berada pada 2,1–2,3 detik dan waktu buka *solenoid door lock* pada 1,8 detik, jauh di bawah batas 5 detik dan 3 detik yang ditargetkan. Logika kontrol kipas berbasis parameter kelembapan terbukti memberikan lapisan proteksi tambahan yang mampu mengintervensi kondisi kelembapan berbahaya secara proaktif, sebelum suhu mencapai ambang batas termal. Kelengkapan pencatatan *log* akses mencapai 100% (80/80 kejadian), dan sistem mencapai *uptime* 99,7% selama 72 jam pengujian intensif. Seluruh tujuh skenario pengujian skematik menghasilkan output yang sesuai

dengan spesifikasi desain, memvalidasi konsistensi logika operasional sistem di bawah berbagai kombinasi kondisi lingkungan dan keamanan.

Kontribusi utama penelitian ini adalah integrasi unik antara manajemen termal *dual-parameter* (suhu dan kelembapan), keamanan akses fisik elektronik, dan akuntabilitas *cloud* dalam satu solusi IoT berbiaya rendah yang belum ditemukan secara bersamaan dalam literatur yang dikaji [1][3][4][6], serta kesesuaiannya untuk konteks usaha mikro, kecil, dan menengah (UME) di Indonesia dengan keterbatasan sumber daya teknis dan finansial.

Untuk pengembangan selanjutnya disarankan: (1) implementasi TLS dengan validasi sertifikat penuh untuk memperkuat keamanan komunikasi *cloud* [7][18]; (2) penambahan konektivitas LoRa sebagai *fallback* pada lingkungan dengan sinyal Wi-Fi terbatas; (3) integrasi algoritma deteksi anomali berbasis *machine learning* untuk identifikasi pola *overheating* secara prediktif [5]; serta (4) perancangan *enclosure* tahan korosi dan kelembapan tinggi untuk memperluas penerapan sistem ke lingkungan industri yang lebih ekstrem.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Program Studi S1 Sistem Komputer Universitas Sains dan Teknologi Komputer (STEKOM) Semarang atas dukungan akademis, serta kepada manajemen CV Gundara Solusi Bersama Ungaran atas akses dan kerjasamanya sebagai lokasi penelitian. Penelitian ini merupakan bagian dari tugas akhir mahasiswa Program Studi S1 Sistem Komputer STEKOM Tahun Akademik 2025/2026.

## REFERENSI

- [1] Wan, J., Zhou, J., & Gui, X. (2021). Intelligent rack-level cooling management in data centers with active ventilation tiles: A deep reinforcement learning approach. *IEEE Intelligent Systems*, 36(6), 42–52. <https://doi.org/10.1109/MIS.2021.3049865>
- [2] Chen, Y., Tian, Z., Kong, M., Sun, L., Zhou, L., Wang, W., & Liu, M. (2026). A Study on the Thermal Management Performance of Server-Oriented Memory Liquid Cooling Solutions. *Energies*, 19(9), 2150. <https://doi.org/10.3390/en19092150>
- [3] Cho, J., & Lim, S. (2023). Balanced comparative assessment of thermal performance and energy efficiency for three cooling solutions in data centers. *Energy*, 285, 129370. <https://doi.org/10.1016/j.energy.2023.129370>
- [4] Gül, F., & Eroğlu, H. (2024). Low-cost IoT mesh network for real-time indoor air quality monitoring. *Proceedings of ICCES 2024*. <https://doi.org/10.1109/ICCES63552.2024.10860109>
- [5] Wan, J., Duan, Y., Gui, X., et al. (2023). SafeCool: Safe and energy-efficient cooling management in data centers with model-based reinforcement learning. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 7(6), 1621–1635. <https://doi.org/10.1109/TETCI.2023.3234545>
- [6] Nabão, J., Manuel Raposo Amaral, A., Sá, F. (2025). IoT-based Real-time Temperature Monitoring in Critical Systems: A Review. *IgMin Res*, 3(5), 226-234. <https://doi.org/10.61927/igmin303>
- [7] Yusof, M. K., Man, M., Wan Hamzah, W. M. A. F., et al. (2023). A secure model based on symmetric encryption for REST API in data integration. *Proceedings of ICSEC 2023*. <https://doi.org/10.1109/ICSEC59635.2023.10329720>
- [8] Stallings, W. (2017). *Cryptography and network security: Principles and practice* (7th ed.). Pearson.
- [9] Andriana, M., & Priyadi, P. (2023). Implementasi metode R&D pada sistem penggajian guru honorer. *Jurnal Akuntansi dan Bisnis*, 3(1), 20–27. <https://doi.org/10.51903/jiab.v3i1.230>
- [10] Priyadi, P., & Santoso, B. (2022). Sistem informasi administrasi pembayaran sekolah terintegrasi barcode reader dengan metode berorientasi objek berbasis client server. *ELKOM: Jurnal Elektronika dan Komputer*, 15(2), 228–233. <https://doi.org/10.51903/elkom.v15i2.901>
- [11] Espressif Systems. (2024). ESP32 series datasheet v5.2. *Espressif Systems Inc.* [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)
- [12] Sensirion AG. (2022). SHT21 humidity and temperature sensor IC datasheet. *Sensirion AG.* <https://www.sensirion.com/media/documents/120BBE4C/63500E51/Datasheet-Humidity-Sensor-SHT21.pdf>
- [13] Rombor, B. (2024). Keypad library for Arduino. *PlatformIO Registry.* <https://registry.platformio.org/libraries/Chris--A/Keypad>
- [14] Multicomp Pro. (2019). MP001162 solenoid door lock datasheet. *Heilind Electronics.* <https://www.multicomp-pro.com/mp001162>
- [15] Şen, M., & Mutluer, M. (2025). A Review of BLDC Motors: Types, Application, Failure Modes and Detection. *Energies*, 18(24), 6402. <https://doi.org/10.3390/en18246402>
- [16] DFRobot. (2025). DFR0154 LCD 20x4 I2C module specification. *DFRobot Wiki.* [https://wiki.dfrobot.com/LCD\\_Display\\_20\\_4\\_SKU\\_DFR0154](https://wiki.dfrobot.com/LCD_Display_20_4_SKU_DFR0154)
- [17] Espressif Systems. (2025). ESP-IDF programming guide – C++23 support. <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/cplusplus.html>
- [18] Rescorla, E. (2018). The Transport Layer Security (TLS) protocol version 1.3. RFC 8446, *IETF.* <https://doi.org/10.17487/RFC8446>
- [19] Google LLC. (2025). Google Apps Script documentation: Spreadsheet service. *Google Developers.* <https://developers.google.com/apps-script/reference/spreadsheet>
- [20] ASHRAE. (2021). Thermal guidelines for data processing environments – Fifth edition. *American Society of Heating, Refrigerating and Air-Conditioning Engineers.*
- [21] Fielding, R. T., Nottingham, M., & Mogul, J. C. (2022). Hypertext transfer protocol semantics and content. RFC 9110, *IETF.* <https://doi.org/10.17487/RFC9110>
- [22] Espressif Systems. (2025). Mbed TLS API documentation. *Espressif IDF Documentation.* <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/protocols/mbedtls.html>

