

Smart Temperature and Humidity Monitoring pada Data center Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta Berbasis Raspberry Pi 5 dan Notifikasi Discord

Ari Widiyatmoko^{1*}, Imam Suharjo¹

¹Universitas Mercu Buana Yogyakarta, Sleman, Indonesia

ariwidiyatmoko24@gmail.com^{1*}

| Received: 28/06/2025 |

Revised: 14/07/2025 |

Accepted: 28/07/2025 |

Copyright©2025 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Abstrak

Data center merupakan bagian yang vital dalam sebuah infrastruktur Teknologi Informasi dan Komunikasi (TIK) yang membutuhkan pengawasan lingkungan fisik secara kontinu, khususnya suhu dan Kelembapan. Parameter-parameter tersebut memiliki pengaruh signifikan terhadap kestabilan dan umur perangkat server. Dengan demikian, diperlukan suatu sistem pemantauan yang memiliki tingkat akurasi tinggi, dapat diandalkan, dan mampu merespons secara cepat. Penelitian ini bertujuan untuk merancang serta mengembangkan sistem pemantauan suhu dan Kelembapan yang menggunakan Raspberry Pi 5 sebagai pusat kendali, memanfaatkan sensor DHT22 untuk mengukur suhu dan Kelembapan, serta menyimpan data hasil pengukuran tersebut dalam basis data MariaDB, Dashboard monitoring menggunakan Grafana, serta sistem notifikasi otomatis ke platform Discord. Lokasi penelitian difokuskan pada *Data center* yang dikelola oleh Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta, yang menjadi objek utama pengujian dan implementasi sistem. Sistem ini memungkinkan pemantauan secara *real-time*, pemberitahuan ketika parameter lingkungan melebihi batas aman yang telah ditentukan, dan laporan harian otomatis yang dikirim ke channel Discord. Metode yang digunakan adalah pendekatan rekayasa sistem, dengan tahapan meliputi studi literatur, analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi. Hasil pengujian menunjukkan bahwa sistem mampu melakukan pembacaan sensor secara akurat (rentang toleransi maksimal $\pm 0.4^{\circ}\text{C}$ untuk suhu dan $\pm 1.4\%$ untuk kelembapan) dan menyimpan data secara berkala ke dalam basis data. Sistem juga dapat mengirimkan notifikasi secara otomatis ke Discord apabila suhu atau Kelembapan melebihi ambang batas ($18\text{--}27^{\circ}\text{C}$ dan $30\text{--}60\%$ RH), serta mengirimkan laporan visual dashboard setiap hari secara terjadwal. Dengan adanya sistem monitoring ini diharapkan dapat membantu tugas tim pengelola *Data center* Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta dalam melakukan pemantauan dan manajemen keamanan lingkungan *Data center* sehingga lebih efektif dan responsif.

Kata kunci: suhu dan Kelembapan, Raspberry Pi 5, DHT22, Grafana, Discord

Abstract

A data center is a vital component of Information and Communication Technology (ICT) infrastructure that requires continuous physical environmental monitoring, particularly for temperature and humidity. These parameters significantly affect the stability and longevity of server equipment. Therefore, a monitoring system with high accuracy, reliability, and rapid response capability is essential. This study aims to design and develop an integrated temperature and humidity monitoring system using a Raspberry Pi 5 as the control unit, employing a DHT22 sensor to measure environmental conditions. The collected data is stored in a MariaDB database, visualized through a Grafana dashboard, and complemented with automated notification delivery via the Discord platform. The implementation and testing were conducted at the Data center managed by the Department of Communication and Informatics of the Special Region of Yogyakarta. The system supports real-time monitoring, alerts when environmental parameters exceed defined safety thresholds, and automated daily reports sent to a designated Discord channel. The research follows an engineering-based system development method, comprising stages of literature study, needs analysis, system design, implementation, testing, and evaluation. The testing results indicate that the system performs accurate sensor readings (within a tolerance range of $\pm 0.4^{\circ}\text{C}$ for temperature and $\pm 1.4\%$ for humidity) and successfully stores data periodically in the database. It also reliably sends automatic alerts to Discord when temperature or humidity levels exceed the specified thresholds ($18\text{--}27^{\circ}\text{C}$ and $30\text{--}60\%$ RH) and delivers visual dashboard reports twice daily on a schedule. This monitoring system is expected to support the operational team of the Data center in managing environmental security more effectively and responsively.

Keywords: temperature and humidity, Raspberry Pi 5, DHT22, Grafana, Discord

Pendahuluan

Teknologi informasi dan komunikasi (TIK) memegang peran sentral dalam mendukung aktivitas pemerintahan modern. Pemerintah Daerah Daerah Istimewa Yogyakarta, melalui Dinas Komunikasi dan Informatika, mengelola *Data center* sebagai infrastruktur vital untuk memastikan keberlangsungan pelayanan publik serta operasional pemerintahan berbasis digital. Sebagai tulang punggung operasional, kondisi lingkungan internal *data center*, khususnya suhu dan Kelembapan, harus dijaga dalam batas aman untuk menghindari kerusakan perangkat keras dan gangguan sistem.

Menurut panduan dari American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), suhu udara kering (*dry-bulb temperature*) yang direkomendasikan untuk peralatan kelas A1 hingga A4 adalah antara 18°C hingga 27°C (ASHRAE, 2021). Untuk Kelembapan relatif, ASHRAE merekomendasikan rentang antara 20% hingga 80%, dengan titik embun maksimum 15°C . Menjaga suhu dan Kelembapan tetap stabil sangat penting agar perangkat tidak rusak akibat listrik statis atau uap air yang bisa muncul, karena hal tersebut dapat mengganggu kinerja sistem di dalam *data center*. Tingkat Kelembapan yang berlebihan

berpotensi menimbulkan hubungan arus pendek, sementara tingkat Kelembapan yang terlalu rendah dapat memicu timbulnya muatan listrik statis (Rizkiawan, 2024).

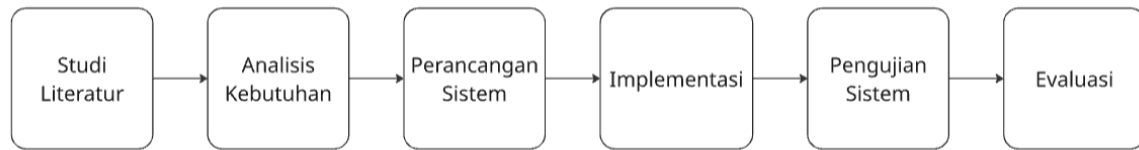
Pengelolaan suhu dan Kelembapan yang tidak optimal dapat meningkatkan risiko kegagalan perangkat TIK, yang pada gilirannya berpotensi mengganggu kelancaran pelayanan publik. Oleh karena itu, pemantauan lingkungan secara *real-time* merupakan aspek yang esensial dalam menjaga kestabilan operasional. Namun demikian, infrastruktur Data center Pemerintah Daerah DIY saat ini belum terintegrasi dengan sistem monitoring otomatis yang mampu mendeteksi dan memberikan notifikasi secara langsung apabila terjadi deviasi suhu maupun Kelembapan dari batas normal yang telah ditetapkan. Sistem monitoring yang ada hanya dapat diakses melalui jaringan privat tanpa mekanisme peringatan dini, sehingga pengelolaan risiko menjadi kurang responsif.

Menanggapi tantangan tersebut, diperlukan sebuah sistem monitoring berbasis teknologi *Internet of Things* (IoT) yang mampu melakukan pemantauan suhu dan Kelembapan secara terus-menerus dan memberikan peringatan secara otomatis. Dalam penelitian ini, Raspberry Pi 5 dipilih sebagai platform utama untuk membangun sistem monitoring tersebut, dengan integrasi pengiriman notifikasi melalui aplikasi Discord sebagai sarana pemberitahuan instan kepada tim pengelola. Beberapa penelitian sebelumnya turut menjadi acuan dalam pengembangan sistem ini. Penelitian oleh Angga (2020) mengembangkan sistem monitoring suhu ruang server menggunakan Raspberry Pi 4 dan sensor DHT22, dengan pengiriman data melalui protokol MQTT serta notifikasi melalui Telegram, yang memungkinkan pemantauan suhu secara *real-time* dari jarak jauh (Angga, 2020). Penelitian lain oleh Fadillah & Permata (2024) memanfaatkan sensor DHT11 dan NodeMCU ESP8266 untuk monitoring suhu dan kelembapan di ruang *data center* secara *real-time*. Meskipun sistem tersebut mampu menampilkan dan menyimpan data, belum terdapat fitur notifikasi otomatis untuk mempercepat respons terhadap perubahan kondisi lingkungan (Fadillah, 2024).

Penelitian ini difokuskan pada perancangan dan implementasi sistem pemantauan suhu dan Kelembapan berbasis Raspberry Pi 5 secara efisien, disertai pengembangan fitur notifikasi otomatis melalui platform Discord. Selain itu, penelitian ini juga bertujuan untuk mengevaluasi sejauh mana sistem tersebut efektif dalam menunjang proses pengawasan lingkungan *data center* secara *real-time* dan responsif. Manfaat praktis dari penelitian ini adalah menyediakan solusi monitoring lingkungan yang murah, efektif, dan mudah diakses secara *real-time*. Sementara itu, secara akademis, penelitian ini dapat menjadi referensi dalam pengembangan sistem monitoring berbasis IoT di sektor pemerintahan, serta mendorong inovasi pemanfaatan mikrokontroler dan platform komunikasi modern.

Metodologi Penelitian

Penelitian ini menggunakan pendekatan rekayasa (*engineering*) untuk merancang dan mengimplementasikan sistem monitoring suhu dan Kelembapan yang terintegrasi. Alur Metode penelitian dapat dilihat pada gambar dibawah ini.



Gambar 1. Alur Metode Penelitian

Studi Literatur

Studi Literatur dilakukan dengan pengumpulan informasi yang bersumber dari buku, jurnal, artikel ilmiah, dan sumber lain terkait implementasi teknologi *Internet of Things* (IoT), perangkat mini komputer Raspberry Pi, sensor suhu dan Kelembapan DHT22, basis data MariaDB, *dashboard* visualisasi data dengan Grafana, serta sistem notifikasi dengan Discord. Studi literatur ini bertujuan untuk memahami konsep dasar serta teknologi yang akan diterapkan dalam penelitian.

Analisis Kebutuhan

Tahap ini bertujuan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional dari sistem monitoring yang akan dikembangkan. Analisis dilakukan berdasarkan observasi langsung terhadap kondisi lingkungan Data center Dinas Kominfo DIY serta wawancara dengan pihak terkait. Hasil analisis ini digunakan sebagai dasar dalam merancang sistem sesuai kebutuhan di lapangan. Analisis kebutuhan difokuskan pada dua aspek utama, yaitu kebutuhan perangkat keras dan perangkat lunak.

Berikut ini adalah kebutuhan dalam membangun Smart Temperature and Humidity Monitoring pada Data center Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta:

Perangkat Keras

Perangkat keras merupakan perangkat fisik yang berfungsi sebagai media untuk menampung perangkat lunak dan sebagai pembaca kondisi suhu dan Kelembapan di dalam *Data center*. Perangkat keras yang dibutuhkan meliputi:

1. Raspberry PI 5

Raspberry Pi adalah komputer mini berukuran kartu kredit yang dikembangkan oleh Raspberry Pi Foundation. Perangkat ini memiliki kemampuan komputasi yang cukup untuk menjalankan sistem operasi lengkap berbasis Linux dan mendukung berbagai aplikasi *Internet of Things* (IoT), multimedia, maupun komputasi ringan.

Raspberry Pi memiliki kemampuan yang cukup untuk menangani tugas-tugas pengumpulan dan pemrosesan data, serta dapat terhubung dengan berbagai jenis sensor untuk keperluan monitoring.

Raspberry Pi 5 merupakan versi terbaru yang memiliki peningkatan performa signifikan dibandingkan generasi sebelumnya. Dengan prosesor ARM Cortex-A76 quad-core 2.4 GHz dan RAM LPDDR4X hingga 8 GB, Raspberry Pi 5 mampu menangani beban komputasi lebih besar, transfer data cepat dengan USB 3.0, dan visualisasi data melalui jaringan secara *real-time*.

Raspberry Pi 5 dipilih sebagai platform utama karena menawarkan keseimbangan antara kapabilitas komputasi yang tinggi dan efisiensi biaya, menjadikannya solusi tepat untuk implementasi sistem monitoring berskala kecil hingga menengah. Sebagai perangkat komputasi yang mumpuni, Raspberry Pi 5 mampu menjalankan algoritma yang kompleks dan melakukan pemrosesan data sensor dengan tingkat akurasi yang tinggi (Prakoso, 2025).

2. Sensor DHT22

Sensor DHT22 atau juga dikenal sensor AM2302, merupakan salah satu jenis sensor suhu dan Kelembapan dengan keluaran data digital yang memiliki tingkat ketelitian yang tinggi. Sensor ini bekerja dengan sistem komunikasi satu kabel (*single-wire*) dan memiliki cakupan pengukuran suhu dari $-40\text{ }^{\circ}\text{C}$ hingga $80\text{ }^{\circ}\text{C}$ dengan akurasi sekitar $\pm 0,5\text{ }^{\circ}\text{C}$. Selain itu, sensor ini mampu mengukur Kelembapan dalam rentang 0 hingga 100 persen *Relative Humidity* (Suwandi, 2023).

Spesifikasi ini, sensor DHT memenuhi kebutuhan pembacaan suhu dan Kelembapan di ruang *Data center* dan komunikasi data dengan Raspberry Pi 5 juga lebih sederhana karena hanya menggunakan satu jalur komunikasi dengan data yang diolah sudah berbentuk digital. Sehingga tidak memerlukan proses kalibrasi tambahan.

Perangkat Lunak

Perangkat lunak dalam penelitian ini berfungsi sebagai pengendali utama dari seluruh sistem monitoring suhu dan Kelembapan. Perangkat lunak bertugas untuk membaca data dari sensor, menyimpan ke dalam basis data, menampilkan data secara visual, serta mengirimkan notifikasi ke platform komunikasi. Perangkat lunak yang dibutuhkan meliputi:

1. Python

Python merupakan bahasa pemrograman yang memiliki karakteristik interpretatif dan menyediakan fitur fungsional yang lengkap. Desainnya berfokus pada keterbacaan kode, sehingga memudahkan pengembang dalam memahami dan memelihara program yang ditulis (Semendawai, 2021). Python digunakan sebagai bahasa pemrograman utama untuk mengembangkan *code* yang mengatur pembacaan data dari sensor DHT22, pengolahan data, pengiriman data ke MariaDB, serta integrasi dengan Discord untuk notifikasi. Python dipilih karena struktur sintaks yang sederhana, banyak Pustaka yang bisa dimanfaatkan dalam menyusun kode dan dapat dijalankan dalam lingkungan Raspberry Pi.

2. Basis Data MariaDB

MariaDB adalah sistem manajemen database bersifat *open-source* yang banyak dimanfaatkan, yang muncul sebagai hasil pengembangan lanjutan dari MySQL. MariaDB memiliki kemiripan dengan MySQL dari sisi fitur, struktur dan perintah-perintah yang digunakan atau dengan kata lain MariaDB kompatibel dengan MySQL (Jeffrey, 2020). Basis data MariaDB ini digunakan untuk menampung dan menyimpan hasil pembacaan suhu dan Kelembapan dari sensor DHT22. Database pembacaan sensor suhu dan Kelembapan yang menjadi sumber data dalam Dashboard monitoring Grafana. Basis data MariaDB ini kompatibel digunakan dalam lingkungan Raspberry Pi karena secara kinerja tidak memakan banyak *resource*.

3. Grafana

Grafana adalah perangkat lunak *open-source* yang digunakan untuk menyajikan data melalui tampilan dashboard interaktif, yang dapat mengambil informasi dari berbagai jenis sumber seperti basis data dan sistem pemantauan. Aplikasi ini juga menyediakan fitur pemberitahuan otomatis yang berfungsi untuk melakukan pemantauan kondisi sistem secara langsung dan berkelanjutan. Melalui template yang dapat dikonfigurasi dan disesuaikan, grafana mempermudah pengguna dalam menyusun tampilan data visual sesuai kebutuhan berdasarkan variabel yang tersedia. Platform ini berfokus pada dukungan visualisasi data dalam format deret waktu yang mudah untuk dipahami (Rahman, 2020).

4. Discord

Discord merupakan platform komunikasi berbasis *cloud* yang menawarkan berbagai layanan seperti pesan teks, panggilan suara, dan video, yang diatur melalui server dan saluran (*channels*) yang dapat disesuaikan sesuai kebutuhan penggunanya. Dirancang untuk memfasilitasi kolaborasi dan komunikasi *real-time*, Discord menawarkan sistem notifikasi pintar, integrasi API melalui *webhook* dan *bot*, serta dukungan berbagai platform yaitu *desktop*, *web*, dan *mobile*. Berbeda dengan aplikasi pesan instan pada umumnya, Discord memberikan keleluasaan bagi pengguna untuk memanfaatkan *bot* yang fungsinya dapat disesuaikan dan dikembangkan secara mandiri sesuai kebutuhan (Setyawan, 2024).

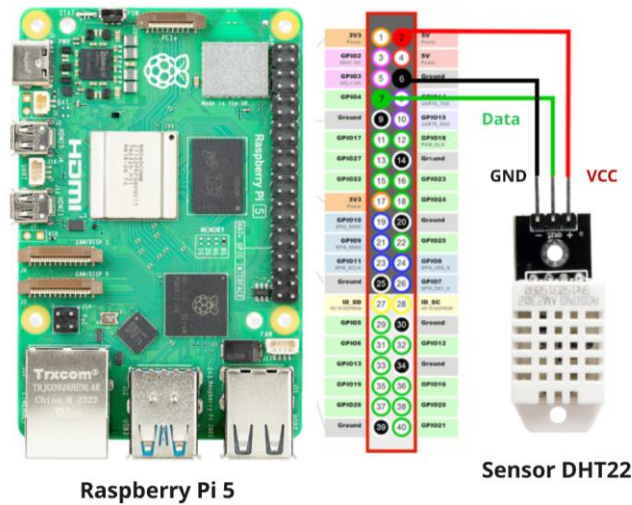
Salah satu fitur unggulan dari platform ini adalah fitur *Webhook*. Melalui *webhook*, sistem atau aplikasi dapat mengirimkan notifikasi, laporan, atau pembaruan secara langsung tanpa perlu interaksi manual pengguna. Fitur ini sangat berguna untuk mengintegrasikan Discord dengan sistem *monitoring*, *bot*, atau layanan pihak ketiga lainnya, sehingga memungkinkan pengiriman informasi secara *real-time* dan efisien ke dalam platform Discord. Selain itu alasan utama penggunaan platform ini karena jalur komunikasi pengelola *Data Centar* pada Dinas Komunikasi dan Informatika DIY menggunakan kanal Discord. Sehingga sangat relevan jika sistem *monitoring* nantinya menggunakan platform ini.

Perancangan Sistem

Tahap perancangan sistem dilakukan setelah diketahui kebutuhan yang diperlukan di lapangan. Tahap perancangan sistem ini terdiri dari perancangan perangkat keras dan perancangan perangkat lunak.

Perancangan Perangkat Keras

Perancangan ini meliputi desain dari perangkat keras yang digunakan yaitu koneksi sensor DHT22 dan Raspberry Pi 5 sebagai alat *monitoring*. Selain itu juga dibutuhkan rancangan *case* atau wadah dari alat *monitoring* ini.



Gambar 2. Rancangan Koneksi Perangkat Keras

Gambar 2 menunjukkan rancangan sambungan antara Raspberry Pi 5 sebagai *controller* dengan sensor DHT22. Terdapat 3 sambungan utama yaitu jalur power positif (VCC), jalur power negatif (GND) dan jalur Data. Jalur VCC dan GND digunakan untuk memberikan sumber power atau catu daya untuk sensor DHT22. Sumber power diambil dari Raspberry Pi 5. Jalur Data merupakan jalur pembacaan output atau keluaran dari sensor DHT22 untuk dibaca oleh pin masukan Raspberry Pi 5 melalui pin nomor 7 atau GPIO4. GPIO4 memiliki fungsi sebagai jalur masukan digital tunggal untuk membaca keluaran sensor DHT22 berupa data suhu dan kelembapan.

Rangkaian perangkat keras ini membutuhkan wadah untuk melindungi rangkaian Raspberry Pi 5 dan sensor DHT22 dari gangguan luar. Desain case atau wadah menggunakan aplikasi Onshape. Aplikasi Onshape ini merupakan aplikasi desain gambar 3D berbasis cloud sehingga tidak memerlukan spesifikasi khusus untuk komputer yang digunakan untuk menggunakannya.



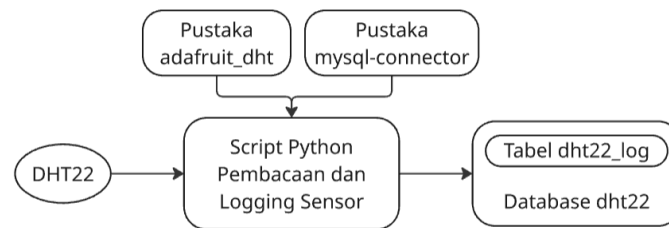
Gambar 3. Desain Case Alat Monitoring

Output hasil desain dari aplikasi Onshape nantinya akan dicetak menggunakan mesin 3D printer. Mesin 3D printer akan mencetak setiap detail dari desain yang diinputkan. Material yang digunakan untuk mencetak case adalah Polylactic Acid (PLA). Material ini dipilih karena berasal dari sumber daya terbarukan dan mempunyai akurasi dan ketelitian yang baik.

Perancangan Perangkat Lunak

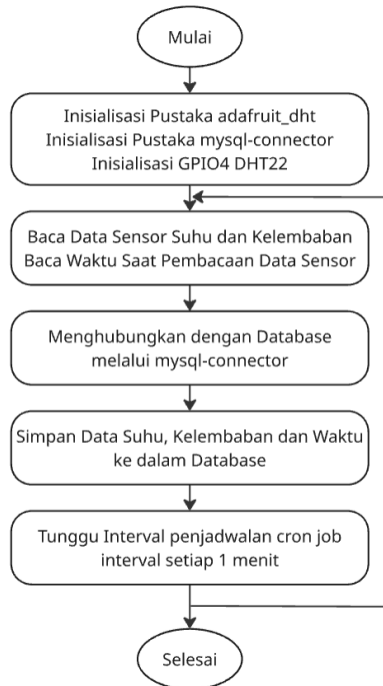
Perancangan perangkat lunak merupakan tahap penting dalam pengembangan sistem monitoring suhu dan kelembapan yang bertujuan untuk memastikan setiap proses dapat berjalan sesuai dengan kebutuhan sistem. Perancangan ini mencakup penyusunan alur kerja, pembuatan skrip program untuk masing-masing fungsi, serta desain integrasi antarkomponen seperti sensor, basis data, sistem notifikasi, dashboard visualisasi, dan fitur pelaporan. Secara umum, perancangan perangkat lunak dibagi ke dalam empat komponen utama berikut:

1. Rancangan Pemrosesan Pembacaan Sensor Suhu dan Kelembapan



Gambar 4. Diagram Komponen Program Pembacaan Sensor DHT22

Pemrosesan pembacaan sensor suhu dan kelembapan dimulai dengan penyusunan *Script* python yang berfungsi untuk mengolah data pembacaan dari sensor DHT22 yang terkoneksi ke Raspberry Pi 5. *Script* python ini memanfaatkan pustaka `adafruit_dht` yang secara khusus digunakan untuk berkomunikasi dengan sensor DHT22. GPIO4 digunakan sebagai antarmuka sensor dengan Raspberry Pi 5 dan diinisialisasikan dalam program. Program akan melakukan pembacaan data suhu dan kelembapan dan kemudian mencatatkan data tersebut ke dalam basis data MariaDB secara berkala. Untuk proses ini, digunakan pustaka `mysql-connector-python` yang memungkinkan *Script* python melakukan koneksi dan eksekusi perintah SQL ke basis data MariaDB. Nilai suhu, kelembapan, serta waktu pembacaan kemudian dicatat ke dalam tabel `dht22_log`.



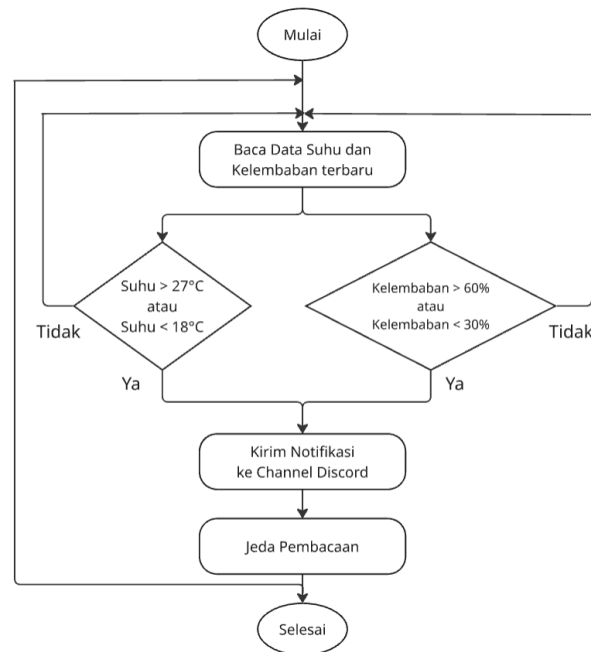
Gambar 5. Alur Pemrosesan Data Sensor Suhu dan Kelembapan

Script ini akan dijalankan dengan sistem penjadwalan menggunakan cron job agar melakukan pembacaan dan penyimpanan data dalam interval waktu tertentu yaitu setiap 1 menit. Dengan menggunakan sistem penjadwalan, data suhu dan kelembapan akan selalu diperbarui secara *real-time* dan tersimpan di dalam basis data.

2. Rancangan Fungsi pengiriman Notifikasi Kondisi Suhu dan Kelembapan

Fungsi pengiriman notifikasi pada sistem ini dirancang untuk memberikan peringatan secara otomatis apabila terjadi kondisi suhu atau kelembapan di luar batas yang telah ditentukan. Proses ini berjalan di atas perangkat Raspberry Pi 5 menggunakan *Script* Python, yang secara periodik membaca data terbaru dari basis data MariaDB, tempat semua hasil pembacaan sensor dicatat.

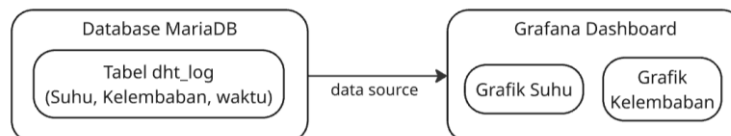
Nilai suhu dan kelembapan yang diambil akan dibandingkan dengan ambang batas tertentu yang telah ditentukan sebelumnya, baik untuk batas minimum maupun maksimum. Jika sistem mendeteksi bahwa suhu atau kelembapan berada di luar rentang tersebut, maka akan memicu pengiriman notifikasi ke platform komunikasi Discord. Pengiriman peringatan kondisi suhu dan kelembapan diluar rentang yang aman ini melalui alamat *webhook* dari *channel* Discord yang sebelumnya telah dikonfigurasi. *Script* Python yang menjalankan fungsi ini akan terus berjalan secara *real-time*,



Gambar 6. Alur Pengiriman Notifikasi Discord

3. Rancangan *Dashboard* Monitoring Suhu dan Kelembapan

Perancangan *dashboard* monitoring berfokus pada penyajian data suhu dan kelembapan secara visual dalam bentuk grafik, sehingga pengguna dapat memantau kondisi lingkungan di *data center* secara *real-time* maupun historis. Visualisasi ini dibangun menggunakan Grafana, yang diinstal pada perangkat Raspberry Pi 5, dengan MariaDB sebagai sumber data utama.



Gambar 7. Bagan Dashboard Monitoring

Data yang divisualisasikan berasal dari tabel `dht22_log`, tempat data hasil pembacaan sensor DHT22 disimpan. Data ini dimuat ke Grafana melalui data source MySQL/MariaDB, yang dikonfigurasi selama proses setup awal.

4. Rancangan Fungsi Pelaporan Monitoring Suhu dan Kelembapan

Fungsi pelaporan monitoring suhu dan kelembapan dirancang untuk memberikan laporan visual berkala kepada pengelola *Data center*. Fungsi ini melengkapi sistem monitoring dengan menyajikan dokumentasi otomatis dalam bentuk tangkapan layar (*screenshot*) dari *dashboard* Grafana, yang kemudian dikirimkan ke saluran pelaporan Discord.

Proses ini dijalankan melalui Script Python dengan alur sebagai berikut:



Gambar 8. Alur Fungsi Pelaporan

- a. Mengakses *dashboard* Grafana menggunakan *browser* otomatisasi dengan menggunakan Selenium.
- b. Menangkap tampilan *dashboard monitoring* suhu dan kelembapan dalam bentuk gambar.
- c. Mengirim gambar tersebut sebagai lampiran dalam pesan ke *channel* Discord melalui Webhook.

Fungsi pelaporan ini juga menggunakan fitur penjadwalan cron job pada Raspberry Pi 5 agar pelaporan dilakukan secara terjadwal dengan periode setiap 12 jam sekali pada jam 6 pagi dan jam 6 sore. Dengan adanya fungsi ini diharapkan akan membantu pengelola *Data center* untuk mendapatkan *update* harian tanpa harus mengakses *dashboard* secara manual.

Implementasi

Tahap Implementasi ini kelanjutan dari tahap perancangan sistem yaitu dengan menerapkan atau mengimplementasikan rancangan perangkat keras dan perangkat lunak yang telah dibuat dan disusun sebelumnya.

Implementasi Rancangan Perangkat Keras

Implementasi ini dimulai dengan penyambungan sensor DHT22 ke perangkat Raspberry Pi 5. Sambungan ini menggunakan kabel untuk menghubungkan setiap pin dari sensor DHT22 menuju pin power dan masukan dari Raspberry Pi 5 sesuai wiring diagram rancangan. Selanjutnya Rangkaian Raspberry Pi 5 dan sensor DHT22 dirakit pada case atau wadah yang telah dicetak menggunakan mesin 3D printer.

Implementasi Rancangan Perangkat Lunak

Proses implementasi rancangan perangkat lunak mencakup beberapa tahapan yaitu:

1. Implementasi Program Pemrosesan Pembacaan Sensor Suhu dan Kelembapan

Tahap ini diawali dengan instalasi pustaka (*library*) pendukung sensor DHT22 pada perangkat Raspberry Pi 5, serta instalasi MariaDB lengkap dengan pembuatan basis data dan tabel

yang digunakan untuk menyimpan hasil pembacaan sensor. Selanjutnya, program pembacaan suhu dan kelembapan diunggah dan dijalankan menggunakan bahasa pemrograman Python. Hasil akhir dari proses ini adalah tersimpannya data suhu dan kelembapan secara otomatis ke dalam basis data MariaDB.

2. Implementasi Program Pengiriman Notifikasi Kondisi Suhu dan Kelembapan

Program pemantauan berbasis Python diinstal pada Raspberry Pi 5 untuk menjalankan pemantauan suhu dan kelembapan secara *real-time*. Ketika nilai terdeteksi berada di luar batas toleransi yang ditentukan, sistem secara otomatis mengirimkan notifikasi ke Discord melalui integrasi webhook yang telah dikonfigurasi.

3. Implementasi Dashboard Monitoring Suhu dan Kelembapan

Tahapan ini mencakup instalasi aplikasi Grafana pada Raspberry Pi 5 dan konfigurasi dashboard pemantauan. Dashboard dirancang untuk menampilkan data suhu dan kelembapan secara *real-time*, yang diambil langsung dari basis data MariaDB. Visualisasi data ini membantu dalam memantau kondisi lingkungan *data center* secara informatif dan interaktif.

4. Implementasi Program Pelaporan Monitoring Suhu dan Kelembapan

Tahap implementasi ini berfokus pada integrasi pelaporan *data monitoring* yang ditampilkan pada Dashboard monitoring di Grafana dengan Channel Pelaporan Discord. Integrasi ini memerlukan komponen selenium untuk dapat mencapture Dashboard monitoring. Hasil capture Dashboard monitoring akan diolah dan dikirimkan ke Channel Pelaporan Discord oleh program python yang telah disusun sebelumnya secara periodik.

5. Implementasi Penjadwalan untuk Fungsi Logging dan Pelaporan harian

Untuk mendukung otomatisasi, dilakukan penjadwalan tugas menggunakan cron pada sistem operasi Raspberry Pi. Penjadwalan ini mencakup proses pencatatan data sensor secara berkala (logging) dan pengiriman laporan harian *dashboard monitoring* ke Discord. Dengan adanya penjadwalan ini, sistem dapat berjalan tanpa intervensi manual dan memastikan keberlangsungan proses monitoring.

Pengujian Sistem

Pengujian sistem dilakukan secara langsung di lingkungan operasional sesungguhnya, yaitu di *Data center* Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta. Tujuan utama dari tahap ini adalah untuk memastikan bahwa sistem *Smart Temperature and Humidity Monitoring* berfungsi secara optimal sesuai dengan rancangan dan kebutuhan pengguna. Adapun skenario pengujian yang dilakukan meliputi:

1. Validasi Pembacaan Sensor

Pengujian dilakukan untuk memastikan bahwa data suhu dan kelembapan yang dibaca oleh sensor DHT22 sesuai dengan kondisi lingkungan aktual. Hasil pembacaan dibandingkan dengan alat ukur pembanding (termometer dan hygrometer digital) untuk memverifikasi akurasi data yang dikirimkan ke sistem.

2. Pengujian Notifikasi Ambang Batas

Sistem diuji dengan mensimulasikan kondisi suhu dan kelembapan yang melebihi atau berada di bawah ambang batas yang telah ditentukan. Tujuannya adalah untuk memastikan bahwa sistem dapat mengirimkan notifikasi secara otomatis ke saluran Discord yang telah dikonfigurasi, sebagai peringatan dini terhadap kondisi lingkungan yang tidak normal.

a. Pengujian Pelaporan Harian ke Discord

Sistem diuji dalam menghasilkan *snapshot* dari tampilan *dashboard* (Grafana) dan mengirimkannya secara otomatis ke channel Discord setiap hari. Pengujian ini bertujuan untuk memverifikasi fungsi otomatisasi pelaporan berkala dan memastikan isi laporan sesuai dengan data yang ditampilkan.

Seluruh skenario pengujian ini dirancang untuk mengukur keakuratan, keandalan, dan konsistensi sistem dalam menjalankan fungsinya sebagai solusi pemantauan suhu dan kelembapan secara cerdas dan terintegrasi.

Evaluasi

Dilakukan evaluasi terhadap hasil sistem berdasarkan:

1. Akurasi data sensor,
2. Kecepatan notifikasi,
3. Keandalan laporan harian,
4. Umpan balik dari Pengelola *Data center* Dinas Komunikasi dan Informatika DIY terhadap kemudahan pemantauan.

Evaluasi juga memperhatikan potensi pengembangan sistem di masa depan, seperti penambahan sensor lain, kontrol suhu otomatis, atau integrasi ke *dashboard* SKPD secara lebih luas.

Hasil dan Pembahasan

Hasil Implementasi Sistem

Implementasi Perangkat Keras

Perangkat keras Monitoring suhu dan kelembapan berhasil dirakit sesuai dengan desain atau rancangan awal. Raspberry Pi 5 dan Sensor DHT22 dirangkai dalam sebuah case dari 3D printing seperti terlihat pada Gambar 9. Perangkat Raspberry Pi 5 dan Sensor DHT22 dapat berfungsi sesuai spesifikasi rancangan.



Gambar 9. Perangkat Keras Monitoring

Implementasi Perangkat Lunak

Implementasi perangkat lunak dilakukan untuk merealisasikan seluruh fungsi monitoring suhu dan kelembapan secara terintegrasi. Proses implementasi dilakukan secara bertahap pada perangkat Raspberry Pi 5, mencakup instalasi *library*, pembuatan basis data, penyusunan dan pengujian *Script* Python, konfigurasi *dashboard monitoring*, serta integrasi sistem notifikasi dan pelaporan otomatis.

Gambar 10. menunjukkan tiga *Script* utama berbasis Python yang diimplementasikan untuk menjalankan keseluruhan sistem.

```
(lab_app) root@raspberrypi:/var/www/lab_app/datacenter_monitoring# ls -al
total 116656
drwxr-xr-x  3 root root    4096 Jun  3 20:21 .
drwxr-xr-x 17 root root    4096 Jun 27 04:07 ..
-rwxr-xr-x  1 root root 84021168 Jan  1 1980 chromedriver.debug
-rw-r--r--  1 root root 26227610 Nov 28 2024 chromedriver-v33.2.1-linux-arm64.zip
ip
-rw-r--r--  1 root root    1202 Feb  9 20:00 dht22_log.py
-rw-r--r--  1 root root    2560 Feb  9 20:29 dht22_monitor.py
-rw-r--r--  1 root root    3107 Jun  3 20:21 dht22_report.py
drwxr-xr-x  3 root root    4096 Feb 11 21:30 gen
-rw-r--r--  1 root root    1096 Jan  1 1980 LICENSE
-rw-r--r--  1 root root 9169546 Jan  1 1980 LICENSES.chromium.html
(lab_app) root@raspberrypi:/var/www/lab_app/datacenter_monitoring#
```

Gambar 10. *Script* Python Sistem Monitoring

Script Python *dht22_log.py* berfungsi melakukan pembacaan sensor DHT22 dan menyimpan hasil pembacaan suhu dan kelembapan ke dalam database MariaDB. Script Python *dht22_monitor.py* berfungsi untuk membaca data terbaru dari database dan memeriksa apakah suhu dan kelembapan berada di luar ambang batas yang telah ditentukan (18–27°C untuk suhu dan 30–60% RH untuk kelembapan). *Script* Python ini juga bertugas memberikan notifikasi melalui channel Discord saat suhu dan kelembapan berada di luar kondisi aman. Script Python *dht22_report.py* bertugas untuk mengambil tangkapan layar (*screenshot*) dari dashboard Grafana menggunakan pustaka selenium, lalu mengirimkannya ke channel Discord pelaporan. Penjadwalan dilakukan dua kali sehari, yaitu pukul 06.00 dan 18.00 WIB.

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| dht22    |
| information_schema |
| mysql    |
| performance_schema |
| sys      |
| testDB   |
+-----+
6 rows in set (0.004 sec)

MariaDB [(none)]>
```

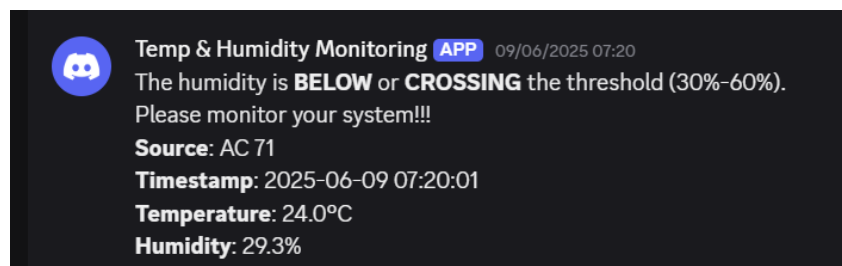
Gambar 11. Database dht22

Database MariaDB berhasil diinstal dan dikonfigurasi pada perangkat Raspberry Pi 5. Database dht22 seperti ditunjukkan pada Gambar 11, merupakan database yang digunakan dalam sistem monitoring ini. Selanjutnya dibuat table dht22_log dengan struktur seperti terlihat pada Gambar 12 untuk menampung data pembacaan suhu dan kelembapan. Hasil pengujian awal menunjukkan data pembacaan sensor suhu dan kelembapan dapat disimpan pada tabel dht22_log.

```
MariaDB [dht22]> describe dht22_log;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id     | int(11) | NO | PRI | NULL | auto_increment |
| timestamp | timestamp | YES | | current_timestamp() |
| temperature | float | YES | | NULL |
| humidity | float | YES | | NULL |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)
```

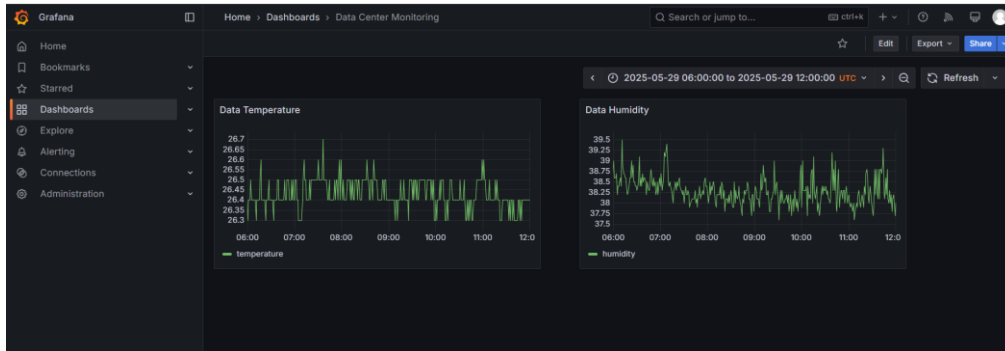
Gambar 12. Struktur Tabel dht22_log

Fungsi pengiriman notifikasi ke channel Discord saat kondisi suhu dan kelembapan diluar ambang batas aman berhasil dijalankan melalui Script Python dht22_monitor.py. Gambar 13 menunjukkan saat kondisi kelembapan terbaca pada nilai 29.3% atau dibawah nilai ambang batas yaitu lebih kecil dari 30% sehingga notifikasi dikirimkan.



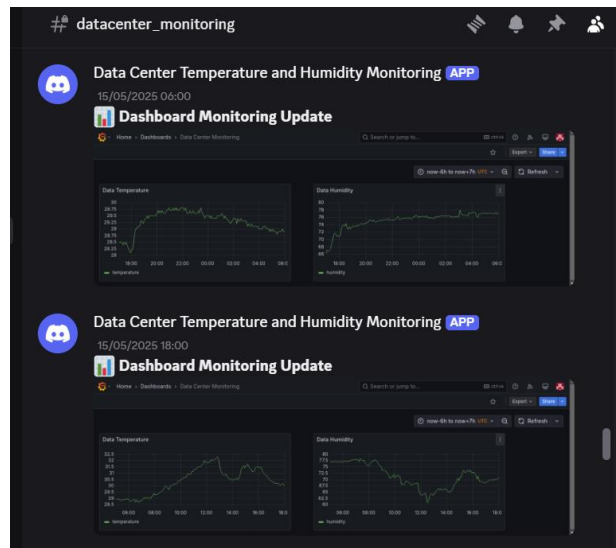
Gambar 13. Notifikasi Ambang Batas ke Discord

Untuk penyajian data, *dashboard monitoring* menggunakan Grafana berhasil diinstal dan dikonfigurasi pada Raspberry Pi 5. Dashboard ini menampilkan data suhu dan kelembapan dari MariaDB dalam bentuk grafik deret waktu (*time series*) seperti ditunjukkan pada Gambar 14.



Gambar 14. Dashboard Monitoring Grafana

Sistem pelaporan berhasil diintegrasikan secara menyeluruh dan bekerja sesuai dengan rancangan awal. Laporan dalam bentuk gambar dashboard seperti terlihat pada Gambar 15 dikirim ke Discord secara otomatis dalam interval 12 jam sekali pada jam 06.00 dan 18.00.



Gambar 15. Pelaporan Harian ke Discord

Secara keseluruhan, proses implementasi perangkat lunak dapat berjalan lancar dan sesuai dengan desain sistem. Setiap komponen terpasang dengan baik dan dapat beroperasi secara otomatis, mendukung keberhasilan sistem *monitoring* suhu dan kelembapan yang akurat, *real-time*, dan mudah diakses.

Hasil Pengujian Sistem

Pengujian dilakukan secara langsung di lingkungan operasional Data center Dinas Komunikasi dan Informatika DIY dengan scenario dan hasil sebagai berikut:

Validasi Pembacaan Sensor

Pengujian dilakukan dengan membandingkan hasil pembacaan dari sensor DHT22 dengan alat ukur termometer dan hygrometer yang telah dikalibrasi sebelumnya. Pembacaan dilakukan secara simultan pada kondisi lingkungan yang sama, dan hasilnya dicatat selama 10 kali percobaan dalam interval waktu 1 jam.

Tabel 1. Pengujian Validasi Pembacaan Sensor DHT22

No.	Waktu Pengujian	Suhu Sensor DHT22 (°C)	Suhu Termometer (°C)	Selisih Suhu (°C)	Kelembaban DHT22 (%)	Kelembaban Hygrometer (%)	Selisih Kelembaban (%)
1	29 Mei 2025 / 08.00	22.5	22.3	0.2	48.3	49.4	1.1
2	29 Mei 2025 / 09.00	22.4	22.3	0.1	48.4	49.3	0.9
3	29 Mei 2025 / 10.00	22.4	22.2	0.2	47.9	48.6	0.7
4	29 Mei 2025 / 11.00	22.5	22.1	0.4	48.2	49.2	1.0
5	29 Mei 2025 / 12.00	22.4	22.2	0.2	48.0	48.9	0.9
6	29 Mei 2025 / 13.00	24.3	24.2	0.1	46.5	47.7	1.2
7	29 Mei 2025 / 14.00	25.4	25.4	0	48.1	48.9	0.8
8	29 Mei 2025 / 15.00	22.9	22.8	0.1	48.5	49.3	0.8
9	29 Mei 2025 / 16.00	23.6	23.2	0.4	48.3	49.5	1.2
10	29 Mei 2025 / 17.00	24.1	23.9	0.2	46.4	47.8	1.4

Berdasarkan data pada Tabel 1, pembacaan sensor DHT22 menunjukkan hasil yang konsisten dan akurat jika dibandingkan dengan alat ukur manual seperti termometer digital dan hygrometer. Perbedaan nilai rata-rata berada dalam rentang toleransi maksimal $\pm 0.4^{\circ}\text{C}$ untuk suhu dan $\pm 1.4\%$ untuk kelembapan.

Pengujian Notifikasi Ambang Batas

Pengujian ini dilakukan untuk memastikan bahwa sistem dapat mendeteksi dan merespon kondisi suhu atau kelembapan yang berada di luar rentang aman yang telah ditetapkan, yaitu suhu $18\text{--}27^{\circ}\text{C}$ dan kelembapan $30\text{--}60\%$ RH. Pengujian dilakukan dengan mensimulasikan berbagai kondisi lingkungan.

Tabel 2. Pengujian Notifikasi Ambang Batas

No.	Waktu Pengujian	Simulasi Kondisi	Pembacaan Sensor	Notifikasi dikirim	Keterangan
1	13 Juni 2025 / 14.40	Suhu melebihi batas	Suhu: 30.4°C Kelembapan: 48% RH	Ya	Suhu tinggi
2	13 Juni 2025 / 14.55	Suhu di bawah batas	Suhu: 16.8°C Kelembapan: 55% RH	Ya	Suhu rendah
3	13 Juni 2025 / 15.05	Suhu dalam rentang aman	Suhu: 25.1°C Kelembapan: 51% RH	Tidak	Kondisi aman
4	13 Juni 2025 / 15.15	Kelembapan melebihi batas	Suhu: 24.7°C Kelembapan: 72% RH	Ya	Kelembapan tinggi
5	13 Juni 2025 / 15.30	Kelembapan di bawah batas	Suhu: 23.4°C Kelembapan: 28% RH	Ya	Kelembapan rendah
6	13 Juni 2025 / 15.50	Kelembapan dalam rentang aman	Suhu: 26.2°C Kelembapan: 45% RH	Tidak	Kondisi aman

Dari enam skenario pengujian yang ditunjukkan pada Tabel 2, sistem berhasil mengirimkan notifikasi secara otomatis pada empat kondisi yang berada di luar ambang batas, yaitu:

- a. Suhu melebihi batas atas (30.4°C)
- b. Suhu di bawah batas bawah (16.8°C)
- c. Kelembapan melebihi batas atas (72% RH)
- d. Kelembapan di bawah batas bawah (28% RH)

Waktu respon notifikasi tercatat sangat cepat, rata-rata di bawah 5 detik setelah data disimpan ke basis data. Hal ini menunjukkan bahwa sistem monitoring memiliki respon waktu yang *real-time*, sesuai kebutuhan pengawasan *Data center*. Pada kondisi lingkungan yang berada dalam batas aman, sistem tidak mengirimkan notifikasi, yang menandakan bahwa fungsi logika pembandingan ambang batas bekerja dengan baik dan tidak menghasilkan notifikasi palsu (*false alert*). Secara keseluruhan, fungsi notifikasi ambang batas telah berfungsi sesuai rancangan, berjalan otomatis, dan dapat diandalkan untuk memberikan peringatan dini kepada pengelola *Data center* melalui platform Discord.

Pengujian Pelaporan Harian ke Discord

Pengujian pelaporan harian bertujuan untuk memastikan bahwa sistem dapat secara otomatis mengambil tangkapan layar (*screenshot*) dari *dashboard monitoring* suhu dan kelembapan di Grafana, kemudian mengirimkannya ke channel Discord pada waktu yang telah dijadwalkan, yaitu setiap 12 jam sekali, yakni pukul 06:00 WIB dan 18:00 WIB.

Pengujian dilakukan selama 5 hari berturut-turut untuk memverifikasi:

- a. Keberhasilan pengambilan *screenshot dashboard*,
- b. Keberhasilan pengiriman laporan visual ke Discord,
- c. Konsistensi waktu pengiriman sesuai jadwal,
- d. Kesesuaian isi gambar laporan dengan tampilan *dashboard* aktual.

Tabel 3. Pengujian Pelaporan Harian

No.	Waktu Pengujian/ Penjadwalan	Waktu Pengiriman Laporan	Screenshot berhasil diambil	Gambar dikirim ke Discord	Kesesuaian Data
1	14 Juni 2025 06.00	14 Juni 2025 06:00:12	Ya	Terkirim	Sesuai
2	14 Juni 2025 18.00	14 Juni 2025 18:00:11	Ya	Terkirim	Sesuai
3	15 Juni 2025 06.00	15 Juni 2025 06:00:16	Ya	Terkirim	Sesuai
4	15 Juni 2025 18.00	15 Juni 2025 18:00:11	Ya	Terkirim	Sesuai
5	16 Juni 2025 06.00	16 Juni 2025 06:00:12	Ya	Terkirim	Sesuai
6	16 Juni 2025 18.00	16 Juni 2025 18:00:12	Ya	Terkirim	Sesuai
7	17 Juni 2025 06.00	17 Juni 2025 06:00:29	Ya	Terkirim	Sesuai
8	17 Juni 2025 18.00	17 Juni 2025 18:00:12	Ya	Terkirim	Sesuai
9	18 Juni 2025 06.00	18 Juni 2025 06:00:13	Ya	Terkirim	Sesuai
10	18 Juni 2025 18.00	18 Juni 2025 18:00:11	Ya	Terkirim	Sesuai

Hasil pengujian seperti terlihat pada Tabel 3, menunjukkan bahwa sistem berhasil menjalankan fungsi pelaporan sesuai jadwal dengan tingkat keberhasilan 100% selama periode pengujian. Gambar yang dikirimkan juga sesuai dengan kondisi terkini dari *data monitoring* pada *dashboard* Grafana.

Evaluasi Sistem

Setelah dilakukan implementasi dan pengujian terhadap sistem yang telah dibangun diperoleh beberapa hal yang menjadi evaluasi, yaitu:

1. Dalam aspek Akurasi dan Keandalan, Sistem menunjukkan tingkat akurasi yang tinggi dalam pembacaan sensor dan tidak terdapat kehilangan data selama pengujian berjalan dalam beberapa hari.
2. Dalam aspek Kecepatan dan Respons Notifikasi, sistem dapat mengirimkan notifikasi dalam rentang waktu kurang dari 5 detik setelah kondisi tidak aman terdeteksi. Hal ini menunjukkan sistem dapat diandalkan sebagai alat peringatan dini.
3. Dalam aspek Kualitas Pelaporan Harian, Laporan visual yang dikirim ke Discord memberikan gambaran lengkap dan *real-time* kepada pengelola tanpa perlu mengakses dashboard secara langsung.
4. Dalam aspek Umpan Balik Pengguna, pihak pengelola *Data center* menyampaikan bahwa sistem ini sangat membantu dan mudah digunakan. Rekomendasi dari pengguna mendorong pengembangan lanjutan seperti penambahan sensor lainnya dan pengiriman laporan mingguan dalam format PDF.

Pembahasan

Berdasarkan hasil pengujian dan evaluasi, sistem terbukti mampu memenuhi kebutuhan monitoring suhu dan Kelembapan di *Data center* Dinas Komunikasi dan Informatika DIY secara efektif dan efisien. Pendekatan berbasis Raspberry Pi 5 dan integrasi dengan platform Discord menjadikan sistem ini sebagai solusi terjangkau dan fleksibel untuk diterapkan pada lingkungan serupa. Penggunaan komponen *open-source* juga memberikan nilai tambah dalam hal kustomisasi dan pengembangan lebih lanjut.

Sistem ini masih dapat dikembangkan lebih lanjut dengan menambahkan fitur seperti penyimpanan cloud, kendali otomatis pendingin ruangan, serta integrasi dengan sistem manajemen *data center* yang lebih kompleks.

Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai perancangan dan implementasi sistem Smart Temperature and Humidity Monitoring pada Data center Dinas Komunikasi dan Informatika Daerah Istimewa Yogyakarta, maka dapat disimpulkan beberapa hal sebagai berikut:

2. Sistem berhasil dirancang dan diimplementasikan dengan baik berdasarkan pendekatan rekayasa sistem, mulai dari analisis kebutuhan, perancangan perangkat keras dan lunak, hingga tahap implementasi dan pengujian langsung di lingkungan Data center.
3. Perangkat keras yang digunakan, yaitu Raspberry Pi 5 dan sensor DHT22, mampu melakukan pembacaan suhu dan kelembapan secara akurat dan konsisten. Rangkaian perangkat telah dirancang secara modular dan dilengkapi *casing* pelindung yang dicetak dengan 3D printer.

4. Hasil pengujian sistem menunjukkan kinerja yang andal. Sensor memberikan data yang akurat, sistem notifikasi bekerja cepat (dalam waktu kurang dari 5 detik), dan laporan rutin berhasil dikirim sesuai jadwal. Seluruh komponen sistem berjalan secara otomatis dan minim intervensi manual.
5. Umpan balik dari pengelola Data center menunjukkan bahwa sistem ini sangat membantu dalam pemantauan lingkungan server. Hal ini memperkuat nilai kebermanfaatan sistem dalam mendukung pengelolaan infrastruktur TI yang lebih cerdas dan efisien.

Penelitian ini juga masih memiliki keterbatasan, seperti cakupan pemantauan yang hanya mencakup satu titik pengukuran, belum adanya mekanisme redundansi perangkat, serta keterbatasan integrasi sistem yang hanya mengandalkan satu platform komunikasi (Discord). Oleh karena itu, untuk kedepannya bisa dilakukan pengembangan agar sistem dilengkapi dengan kemampuan multi-sensor, penerapan sistem cadangan untuk menjaga keandalan, serta integrasi ke berbagai platform monitoring dan komunikasi guna memperluas jangkauan, meningkatkan ketahanan sistem, dan mendukung pengambilan keputusan yang lebih efektif di lingkungan data center.

Daftar Pustaka

- Angga, S. I. (2020). Monitoring Suhu Ruang Server Universitas Semarang Dengan Protokol Komunikasi MQTT dan Notifikasi Telegram Menggunakan Raspberry Pi 4.
- ASHRAE. (2021). *Thermal Guidelines for Data Processing Environments – Expanded Data center Classes and Usage Guidance*. American Society of Heating, Refrigerating and Air-Conditioning Engineers.
- Fadillah, M. &. (2024). Monitoring Suhu dan Kelembapan Di Ruang *Data center* Dinas Komunikasi Informatika Statistik dan Persandian Provinsi Banten Berbasis *Internet of Things* (IoT). *Journal of Electrical Engineering and Computer (JEECOM)*, 98-109.
- Jeffry. (2020). Analisis kinerja web server pada SIM Manajemen Diklat Poltekpel Sorong menggunakan RDBMS MySQL dan MariaDB. *Journal of System and Computer Engineering (JSCE)*, 1(1), 12-20.
- Prakoso, K. S. (2025). Implementasi speech recognition berbasis Raspberry Pi 5 pada ekosistem smart-home menggunakan algoritma Gated Recurrent Unit (GRU). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 9(3).
- Rahman, D. A. (2020). Monitoring Server dengan Prometheus dan Grafana serta Notifikasi Telegram. *Jurnal Ilmiah Teknologi Sistem Informasi*, 1(4), 133 - 138.
- Rizkiawan, M. A. (2024). Pemantauan ruang *data center* berdasarkan suhu dan kelembapan dengan *Internet of Things*. *Jambura Journal of Electrical and Electronics Engineering*, 6(2), 115–122.
- Semendawai, J. N. (2021). Perancangan aplikasi otomatisasi menggunakan bahasa pemrograman Python pada aktivitas monitoring pemakaian data harian kartu *Internet of Things*. *Jurnal Rekayasa Elektro Sriwijaya*, 3(1), 193-198.

- Setyawan, A. P. (2024). Utilization of Discord as an alternative learning media online. Setyawan, A., Purnamasari, Y., & Mucholifah, U. H. (2024). Utilization of Discord as an alternative learning media online. *Proceedings of International Conference on Applied Social Science in Education*, 1(1), 395., (p. 395).
- Suwandi, I. (2023). Perancangan sistem pengukuran suhu dan kelembapan otomatis dengan MD_Parola dan sensor DHT22. *Jurnal of Power Electric and Renewable Energy (JPER)*, 1(2), 30-37.