OTOMATISASI DETEKSI ANOMALI PERANGKAT KERAS DATA CENTER BERBASIS N8N DAN LLM LOKAL

AUTOMATED DATA CENTER HARDWARE ANOMALY DETECTION BASED ON N8N AND LOCAL LLM

Ilham Bintang Herlambang[1], Ragil Hidayatulloh[1], Aditya Herumurti Wijaksono[1], Raka Hanasta Syahrafi[1], Haura Taqiya Azza Nabila[1*], Iqbal Ramadhani Mukhlis[1]
*E-mail: 23082010165@student.upnjatim.ac.id,

[1]Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur

Abstrak

Efisiensi operasional pusat data modern memerlukan sistem monitoring yang mampu menghubungkan kondisi lingkungan internal, informasi ancaman eksternal, dan proses mitigasi secara terpadu. Namun, pemantauan suhu, kelembapan, data seismik, dan rekomendasi tindakan masih sering berjalan secara terpisah sehingga proses identifikasi risiko dan penelusuran keputusan mitigasi belum optimal. Penelitian ini bertujuan merancang prototipe arsitektur Big Data-IoT simulatif sebagai model pendukung monitoring dan mitigasi risiko pada pusat data studi kasus PT XYZ. Metode yang digunakan adalah pendekatan prototyping melalui integrasi data lingkungan dari perangkat IoT berbasis ESP32 dan sensor DHT11, data seismik publik dari BMKG API, backend sebagai lapisan validasi dan integrasi, basis data sebagai penyimpanan historis, workflow automation sebagai mekanisme orkestrasi, AI sebagai pendukung rekomendasi, serta lingkungan deployment berbasis Proxmox untuk pengelolaan layanan. Hasil implementasi menunjukkan bahwa sistem mampu membentuk alur monitoring yang terdokumentasi, mulai dari penerimaan data suhu dan kelembapan, pengambilan informasi seismik, pengolahan konteks mitigasi, pencatatan riwayat rekomendasi, hingga penyediaan data monitoring melalui endpoint sistem. Dengan demikian, integrasi IoT fisik dalam skenario simulatif, data seismik publik, backend, basis data, workflow automation, AI, dan deployment berbasis Proxmox dapat digunakan sebagai pendekatan awal untuk membangun sistem monitoring pusat data yang ringan, modular, adaptif, dan mudah dikembangkan. Ruang lingkup penelitian difokuskan pada validasi prototipe dan alur integrasi data, sehingga pengembangan lebih lanjut dapat diarahkan pada implementasi skala produksi dan perluasan sumber data sensor.

Kata kunci*: *Big Data-IoT, monitoring pusat data, data pipeline, workflow automation, mitigasi risiko.

Abstract

Modern data center operational efficiency requires a monitoring system capable of integrating internal environmental conditions, external threat information, and mitigation processes in a unified manner. However, temperature monitoring, humidity monitoring, seismic data processing, and mitigation recommendations often operate separately, making risk identification and mitigation decision tracking less optimal. This study aims to design a simulated Big Data-IoT architectural prototype as a supporting model for monitoring and risk mitigation in the data center case study of PT XYZ. The method used is a prototyping approach through the integration of environmental data from ESP32-based IoT devices with DHT11 sensors, public seismic data from the BMKG API, a backend layer for validation and integration, a database for historical storage, workflow automation as an orchestration mechanism, AI as a recommendation support system, and a Proxmox-based deployment environment for service management. The implementation results show that the system is able to establish a documented monitoring flow, starting from receiving temperature and humidity data, retrieving seismic information, processing mitigation context, recording recommendation history, and providing monitoring data through system endpoints. Therefore, the integration of physical IoT in a simulated scenario, public seismic data, backend services, databases, workflow automation, AI, and Proxmox-based deployment can serve as an initial approach to building a lightweight, modular, adaptive, and extensible data center monitoring system. The scope of this study focuses on prototype validation and data integration flow, so further development can be directed toward production-scale implementation and the expansion of sensor data sources.

Keywords: *Big Data-IoT, data center monitoring, data pipeline, workflow automation, risk mitigation.*

1. PENDAHULUAN

Transformasi digital mendorong pusat data untuk beroperasi secara semakin andal, responsif, dan berkelanjutan. Keandalan tersebut dipengaruhi oleh stabilitas kondisi lingkungan fisik yang menopang perangkat keras di dalamnya. Suhu dan kelembapan menjadi parameter penting karena perubahan yang tidak terkendali dapat meningkatkan risiko gangguan operasional serta mempercepat penurunan keandalan infrastruktur [1]. Selain kondisi lingkungan internal, informasi ancaman eksternal seperti kejadian seismik juga perlu diperhitungkan sebagai bagian dari konteks mitigasi risiko pada pusat data [7].

Pemanfaatan *Artificial Intelligence for IT Operations* (AIOps) telah berkembang untuk mendukung pemantauan dan analisis infrastruktur teknologi informasi. Namun, penerapannya masih banyak berfokus pada metrik *server*, log sistem, performa aplikasi, dan telemetri jaringan [19]. Pada praktiknya, pemantauan suhu, kelembapan, informasi gempa, dan pencatatan keputusan mitigasi sering kali berjalan secara terpisah sehingga proses identifikasi risiko belum sepenuhnya terdokumentasi dalam satu alur terpadu [4].

Perkembangan *Internet of Things* (IoT) memungkinkan pengambilan data lingkungan dilakukan secara periodik melalui sensor dan mikrokontroler. Dalam penelitian ini, perangkat IoT menggunakan ESP32 dan sensor DHT11 untuk memperoleh data suhu serta kelembapan [15]. Data dikirim dalam bentuk agregasi per jendela waktu (seperti nilai maksimum, minimum, dan

rata-rata) sehingga proses pengiriman serta penyimpanan tetap ringan [6]. Pendekatan ini mendukung kebutuhan *monitoring* termal tanpa harus mengirimkan aliran data mentah secara terus-menerus.

Selain data lingkungan internal, sistem juga memanfaatkan data seismik publik dari BMKG API sebagai representasi ancaman eksternal terhadap pusat data. Informasi seperti waktu kejadian, magnitudo, kedalaman, wilayah, dan koordinat episentrum digunakan untuk memperkaya konteks mitigasi risiko [11]. Dengan pendekatan ini, potensi risiko seismik tidak diperoleh dari sensor getaran lokal, melainkan dari sumber data publik yang relevan dan diakses secara berkala oleh *server*.

Backend dikembangkan menggunakan Fastify sebagai API *server* [13]. Sistem ini juga menggunakan MySQL sebagai basis data relasional [18]. *Backend* berperan sebagai lapisan integrasi yang menerima data IoT, melakukan validasi *payload*, menyimpan riwayat data, mengambil informasi gempa periodik, serta meneruskan *event* ke proses orkestrasi. Lingkungan *deployment* berbasis Proxmox digunakan untuk mendukung pengelolaan layanan secara lebih terpisah, modular, dan mudah dikembangkan [3].

Workflow automation digunakan untuk menghubungkan *backend* dengan proses analisis berbasis AI. Platform n8n memungkinkan integrasi berbasis *webhook*, koneksi API, dan otomasi lintas layanan secara *low-code* [12]. Dalam rancangan ini, n8n tidak ditempatkan sebagai penyimpanan utama data mentah, melainkan komponen orkestrasi yang memproses *event* termal maupun seismik setelah dipicu oleh *backend*. Selanjutnya, layanan AI berbasis *cloud* digunakan untuk membantu menghasilkan status kondisi dan rekomendasi mitigasi [17].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan merancang prototipe arsitektur Big Data-IoT simulatif untuk *monitoring* termal dan mitigasi risiko seismik pada pusat data studi kasus PT XYZ [8]. Fokus penelitian ini adalah memvalidasi alur integrasi data mulai dari pengiriman data lingkungan, pengambilan data seismik, pemrosesan rekomendasi, hingga penyediaan informasi melalui *endpoint* sistem dan *dashboard* pendukung [14].

Penelitian ini diharapkan dapat memberikan model awal sistem *monitoring* pusat data yang ringan, modular, adaptif, dan mudah dikembangkan. Ruang lingkup difokuskan pada prototipe dan skenario simulatif, sehingga pengembangan lebih lanjut dapat diarahkan pada implementasi skala produksi nyata.

2. METODOLOGI

Penelitian ini menggunakan pendekatan *prototyping* untuk merancang dan memvalidasi arsitektur Big Data-IoT pada sistem *monitoring* lingkungan pusat data [8]. Sistem mengintegrasikan perangkat IoT, *backend* API, basis data, BMKG API, *workflow* n8n, AI, dan *dashboard* dalam satu alur data terpusat [9]. Pengujian dilakukan untuk memastikan proses akuisisi dan penyajian informasi dapat berjalan secara terhubung dan aman [16].

Secara umum, arsitektur sistem terdiri atas dua jalur utama. Jalur pertama adalah alur telemetri termal, yaitu data suhu dan kelembapan dari ESP32-DHT11 yang dikirim setiap satu menit ke *backend* [1]. Jalur kedua adalah alur mitigasi gempa, yaitu data seismik dari BMKG API yang diperoleh melalui proses *polling* oleh server [11]. Kedua jalur tersebut diintegrasikan menggunakan platform n8n [5]. Proses ini memanfaatkan AI untuk menghasilkan rekomendasi mitigasi termal [4]. Hasil akhir dari proses mitigasi tersebut ditampilkan secara visual melalui *dashboard* [14].

Gambar 2.1 Arsitektur Alur Monitoring dan Mitigasi Data Center

Gambar 2.1 menunjukkan dua alur utama sistem, yaitu alur telemetri termal dan alur mitigasi gempa. Pada alur telemetri termal, ESP32-DHT11 membaca suhu dan mengirimkan data ke *backend* untuk divalidasi dan disimpan ke basis data MySQL [18]. Setelah data tersimpan, *backend* memicu *workflow* n8n untuk memproses konteks termal secara otomasi *low-code* [12]. Pada alur mitigasi gempa, data yang dikumpulkan diteruskan ke modul AI untuk dianalisis lebih lanjut [17]. *Dashboard* berperan sebagai media penyajian data yang membaca hasil *monitoring* dan mitigasi dari *endpoint backend*.

2.1 Tahapan Penelitian

2.1.1 Analisis Kebutuhan Monitoring dan Mitigasi Risiko

Tahap awal dilakukan dengan mengidentifikasi kebutuhan *monitoring* suhu dan kelembapan sebagai parameter lingkungan internal [15]. Informasi seismik juga ditambahkan sebagai bagian dari konteks ancaman eksternal [7]. Data lingkungan diperoleh melalui ESP32-DHT11, sedangkan data seismik diperoleh dari BMKG API. Skenario pengujian disusun untuk memastikan proses akuisisi data, pengiriman ke *backend*, penyimpanan basis data, integrasi data gempa, pemrosesan *workflow*, dan pencatatan rekomendasi mitigasi berjalan sesuai fungsi. Rincian skenario pengujian ditunjukkan pada Tabel 2.1.

Tabel 2.1 Skenario Pengujian Sistem

Kode	Kondisi Pengujian	Deskripsi	Tujuan Pengujian
TS-1	Akuisisi data lingkungan	ESP32-DHT11 membaca suhu dan kelembapan dalam interval satu menit	Memastikan perangkat mampu menghasilkan data lingkungan secara periodik
TS-2	Ingestion data IoT	ESP32 mengirim payload data suhu dan kelembapan ke backend melalui HTTP POST	Memastikan backend mampu menerima, memvalidasi, dan menyimpan data termal
TS-3	Trigger workflow termal	Backend meneruskan event termal ke n8n setelah data berhasil tersimpan	Memastikan data lingkungan dapat diproses melalui workflow eksternal
TS-4	Integrasi data seismik	Server mengambil informasi gempa terbaru dari BMKG API	Memastikan data seismik dapat digunakan sebagai konteks mitigasi risiko
TS-5	Trigger workflow seismik	Data gempa terbaru diteruskan ke n8n untuk dianalisis	Memastikan workflow mampu menilai potensi dampak gempa terhadap pusat data
TS-6	Pencatatan rekomendasi	Hasil analisis n8n dan AI dikirim kembali ke backend	Memastikan rekomendasi mitigasi tersimpan sebagai audit keputusan
TS-7	Penyajian data monitoring	Web atau dashboard membaca data dari endpoint backend	Memastikan data monitoring dan riwayat mitigasi dapat diakses oleh pengguna

2.1.2 Akuisisi Data Lingkungan Menggunakan ESP32-DHT11

Akuisisi data dilakukan menggunakan perangkat IoT fisik berupa ESP32 yang menunjang komputasi *edge* [20]. Sensor DHT11 digunakan untuk membaca suhu dan kelembapan pada lingkungan yang merepresentasikan ruang pusat data [2]. Data dibaca dalam jendela waktu satu menit, kemudian dikirim ke *backend* dalam bentuk agregasi yang mencakup nilai maksimum, minimum, rata-rata, serta *timestamp* pengukuran. Pendekatan ini membuat proses pengiriman data *payload* lebih ringan karena perangkat tidak perlu mengirimkan seluruh data mentah secara terus-menerus [6].

2.1.3 Ingestion, Validasi, dan Penyimpanan Data

Backend dikembangkan menggunakan arsitektur berbasis Node.js dan Fastify sebagai API server yang cepat [13]. *Backend* menerima *payload* dari ESP32-DHT11, memvalidasi struktur data, menyimpan data suhu dan kelembapan sebagai log termal, serta meneruskan *event* yang relevan ke *workflow* n8n. Dengan alur ini, n8n tidak berperan sebagai penyimpanan utama data mentah, melainkan sebagai komponen orkestrasi setelah data berhasil divalidasi dan disimpan.

2.1.4 Integrasi Data Seismik dari BMKG API

Sistem memanfaatkan layanan publik dari BMKG API sebagai sumber data seismik eksternal. Server melakukan *polling* secara berkala untuk memperoleh informasi gempa terbaru, seperti waktu kejadian, magnitudo, kedalaman, wilayah, koordinat episentrum, dan potensi dampak. Data seismik ini digunakan sebagai konteks tambahan dalam proses mitigasi risiko infrastruktur pusat data dan diteruskan ke *workflow* n8n apabila terdapat informasi gempa terbaru.

2.1.5 Orkestrasi Workflow n8n dan Pemrosesan AI

n8n digunakan sebagai platform workflow automation yang menghubungkan backend dengan proses analisis berbasis AI. Workflow termal dipicu dari data suhu dan kelembapan yang telah tersimpan, sedangkan workflow seismik dipicu dari data gempa terbaru BMKG API. Hasil pemrosesan AI dikirim kembali ke backend untuk disimpan sebagai log mitigasi, sehingga seluruh rekomendasi dapat terdokumentasi dan ditelusuri kembali.

2.1.6 Penyediaan Data Monitoring melalui Web/API

Setelah data termal, data seismik, dan rekomendasi mitigasi tersimpan, *backend* menyediakan *endpoint* API yang digunakan oleh web atau *dashboard* untuk membaca data *monitoring*. *Dashboard* berfungsi sebagai media penyajian informasi, sedangkan proses utama tetap berada pada *backend*, basis data, dan *workflow* n8n. Pola ini membuat sistem lebih terstruktur karena *frontend* hanya menampilkan data yang telah diproses dan tersimpan.

2.1.7 Deployment Berbasis Proxmox

Sistem di-*deploy* pada lingkungan virtualisasi berbasis Proxmox untuk mendukung pengelolaan layanan komputasi secara terisolasi [3]. Lingkungan ini digunakan untuk menjalankan *backend* Fastify, basis data MySQL, n8n *workflow engine*, *reverse proxy*, serta komponen pendukung lainnya secara stabil [10]. Pemisahan layanan ini membantu proses pengelolaan, pengujian, dan pengembangan sistem agar lebih fleksibel.

2.1.8 Pengujian dan Validasi Sistem

Pengujian dilakukan untuk memastikan seluruh alur integrasi berjalan sesuai fungsi. Pengujian mencakup pembacaan dan pengiriman data ESP32-DHT11, validasi *payload* oleh *backend*, penyimpanan data ke MySQL, pengambilan data gempa dari BMKG API, pengiriman *event* ke *workflow* n8n, serta pencatatan rekomendasi mitigasi dari AI. Keberhasilan pengujian ditunjukkan oleh tersimpannya data termal, diperolehnya data seismik, berjalannya *workflow* n8n, dan tersimpannya hasil rekomendasi sebagai audit keputusan.

3. HASIL DAN PEMBAHASAN

Hasil penelitian menunjukkan bahwa arsitektur Big Data-IoT yang dikembangkan mampu mengintegrasikan data lingkungan dari perangkat ESP32-DHT11, data seismik dari BMKG

API, backend Fastify, basis data MySQL, workflow n8n, AI, serta dashboard web dalam satu alur sistem. Pengujian dilakukan dengan meninjau keberhasilan pengiriman data suhu dan kelembapan, penyimpanan data ke basis data, pengambilan informasi gempa, pengiriman event ke workflow n8n, pencatatan rekomendasi mitigasi, serta penyajian hasil melalui dashboard.

3.1 Hasil Pengujian Akuisisi Data IoT

Pengujian pada tahap akuisisi data bertujuan untuk memastikan fungsionalitas perangkat ESP32-DHT11 dalam membaca parameter suhu dan kelembapan lingkungan, sekaligus mengirimkan data tersebut secara periodik menuju *backend*. Hasil pengujian membuktikan bahwa perangkat mampu menghasilkan dan mengirimkan *payload* data yang teragregasi secara konsisten dalam interval satu menit. *Payload* yang ditransmisikan tersebut telah memuat parameter esensial sistem, meliputi identitas perangkat, rentang waktu pengukuran (*timestamp*), serta nilai maksimum, minimum, dan rata-rata untuk variabel suhu maupun kelembapan.

Tabel 3.1 Hasil Pengujian Akuisisi Data ESP32-DHT11

Komponen	Hasil Pengujian	Keterangan
ESP32	Berhasil menjalankan proses pengiriman data	Data dikirim ke backend melalui HTTP POST
DHT11	Berhasil membaca suhu dan kelembapan	Sensor digunakan sebagai sumber data lingkungan internal
Payload agregasi	Berhasil terbentuk	Memuat nilai maksimum, minimum, rata-rata, dan timestamp
Interval pengiriman	Berhasil berjalan setiap satu menit	Sesuai dengan rancangan sistem
Format data	Sesuai dengan struktur endpoint backend	Data dapat diterima dan diproses oleh API

Hasil ini menunjukkan bahwa perangkat IoT fisik dapat digunakan sebagai sumber data lingkungan internal pusat data. Penggunaan payload agregasi membuat data yang dikirim lebih ringan dibandingkan pengiriman data mentah secara kontinu, sehingga sesuai untuk kebutuhan monitoring berkala.

3.2 Hasil Pengujian Backend dan Penyimpanan Data

Pengujian backend dilakukan untuk memastikan bahwa API dapat menerima, memvalidasi, dan menyimpan data dari perangkat IoT. Hasil pengujian menunjukkan bahwa data suhu dan kelembapan yang dikirim dari ESP32-DHT11 dapat diterima oleh backend Fastify dan disimpan ke basis data MySQL sebagai log termal.

Tabel 3.2 Hasil Pengujian API dan Penyimpanan Data

No	Komponen/Endpoint	Fungsi	Hasil Pengujian
1	POST /api/sensor/ingest	Menerima data suhu dan kelembapan dari ESP32-DHT11	Berhasil menerima payload dan menyimpan data

2	Validasi payload	Memastikan struktur data sesuai format yang ditentukan	Payload valid diterima, data tidak sesuai ditolak
3	MySQL thermal_logs	Menyimpan riwayat data suhu dan kelembapan	Data termal berhasil tersimpan
4	Trigger webhook thermal	Meneruskan event termal ke n8n setelah data tersimpan	Event berhasil dikirim ke workflow
5	Endpoint monitoring	Menyediakan data untuk dashboard	Data dapat dibaca oleh dashboard

Hasil tersebut menunjukkan bahwa backend berfungsi sebagai pusat integrasi data. Data dari IoT tidak langsung diproses oleh n8n, melainkan terlebih dahulu diterima, divalidasi, dan disimpan oleh backend. Dengan pola ini, alur data menjadi lebih terstruktur dan hasil pengukuran tetap terdokumentasi pada basis data.

3.3 Hasil Integrasi Data Seismik BMKG API

Pengujian integrasi BMKG API dilakukan untuk memastikan server dapat memperoleh informasi gempa terbaru sebagai data eksternal. Data yang diperoleh meliputi waktu kejadian, magnitudo, kedalaman, wilayah, koordinat episentrum, dan potensi dampak. Informasi ini digunakan sebagai konteks tambahan dalam proses mitigasi risiko pusat data.

Tabel 3.3 Hasil Integrasi Data Seismik BMKG API

Komponen Data	Keterangan	Hasil Pengujian
Tanggal dan jam	Waktu kejadian gempa	Berhasil diperoleh dari BMKG API
Magnitudo	Besaran kekuatan gempa	Berhasil dibaca sebagai konteks risiko
Kedalaman	Kedalaman pusat gempa	Berhasil diperoleh dari data BMKG
Koordinat	Lokasi episentrum gempa	Berhasil digunakan untuk konteks lokasi
Wilayah	Deskripsi lokasi kejadian	Berhasil ditampilkan/diolah oleh sistem
Potensi	Informasi potensi dampak	Berhasil digunakan sebagai konteks mitigasi
Polling data	Pengambilan data gempa secara berkala	Berhasil dilakukan oleh server

Hasil ini menunjukkan bahwa data seismik dapat diintegrasikan tanpa menggunakan sensor getaran lokal. Dengan memanfaatkan BMKG API, sistem tetap memperoleh konteks ancaman eksternal yang relevan untuk mendukung proses mitigasi.

3.4 Hasil Orkestrasi Workflow n8n dan AI

Pengujian workflow n8n dilakukan untuk memastikan bahwa event dari backend dapat diteruskan ke proses analisis berbasis AI. Terdapat dua jalur workflow yang digunakan, yaitu

workflow termal dan workflow seismik. Workflow termal memproses data suhu dan kelembapan yang diterima dari ESP32-DHT11, sedangkan workflow seismik memproses data gempa yang diperoleh dari BMKG API. Hasil pemrosesan AI dikirim kembali ke backend dan disimpan sebagai log rekomendasi mitigasi.

Tabel 3.4 Hasil Pengujian Workflow n8n dan AI

No	Workflow	Input	Proses	Output	Hasil
1	Workflow termal	Data suhu dan kelembapan dari backend	n8n menerima event melalui webhook dan meneruskan konteks ke AI	Status kondisi atau rekomendasi mitigasi termal	Berhasil diproses
2	Workflow seismik	Data gempa dari BMKG API	n8n menerima event gempa dan memproses konteks lokasi serta magnitudo	Rekomendasi mitigasi risiko seismik	Berhasil diproses
3	Callback AI	Hasil rekomendasi dari AI	n8n mengirim hasil kembali ke backend	Data tersimpan sebagai log mitigasi	Berhasil tersimpan
4	Audit keputusan	Data hasil rekomendasi	Backend menyimpan rekomendasi ke basis data	Riwayat mitigasi dapat ditelusuri	Berhasil tersedia

Hasil ini menunjukkan bahwa n8n berperan sebagai komponen orkestrasi, bukan sebagai penyimpanan utama data mentah. AI digunakan untuk membantu menghasilkan rekomendasi berdasarkan konteks data yang diterima. Dengan demikian, sistem tidak hanya menyimpan data, tetapi juga menghasilkan informasi pendukung keputusan.

3.5 Hasil Penyajian Dashboard

Dashboard digunakan sebagai media penyajian hasil monitoring dan rekomendasi mitigasi. Data yang ditampilkan pada dashboard tidak berasal langsung dari n8n, melainkan diambil dari endpoint backend. Dengan demikian, dashboard berfungsi sebagai komponen pembaca data yang telah tersimpan dan diproses oleh sistem.

Dashboard berhasil menampilkan informasi utama, seperti data suhu dan kelembapan terbaru, informasi gempa, status sistem, serta riwayat rekomendasi mitigasi. Hal ini menunjukkan bahwa alur data dari perangkat IoT, BMKG API, n8n, AI, dan backend dapat disajikan kembali dalam bentuk antarmuka yang mudah dipahami pengguna.

Gambar 3.1 Tampilan Dashboard Monitoring Sistem

Gambar 3.1 menunjukkan tampilan dashboard monitoring sistem yang menyajikan data suhu, kelembapan, informasi seismik, dan riwayat rekomendasi mitigasi. Dashboard membaca data melalui endpoint backend sehingga seluruh informasi yang ditampilkan berasal dari data yang telah tersimpan dan diproses oleh sistem.

3.6 Pembahasan

Berdasarkan hasil pengujian, sistem mampu membentuk alur monitoring terintegrasi dari perangkat IoT fisik hingga pencatatan rekomendasi mitigasi. Data suhu dan kelembapan dari

ESP32-DHT11 berperan sebagai sumber data lingkungan internal, sedangkan BMKG API menjadi sumber data eksternal untuk konteks seismik. Integrasi keduanya memungkinkan proses mitigasi mempertimbangkan kondisi lokal sekaligus potensi ancaman dari luar pusat data.

Backend berperan sebagai pusat integrasi dengan memvalidasi, menyimpan, dan meneruskan data ke workflow n8n. MySQL mendukung penyimpanan historis yang terstruktur, sedangkan n8n memudahkan otomasi antar layanan. Integrasi AI melalui n8n menambah kemampuan sistem dalam menghasilkan rekomendasi mitigasi berdasarkan konteks data termal dan seismik, kemudian menyimpannya kembali sebagai audit keputusan.

Secara keseluruhan, kombinasi ESP32-DHT11, Fastify, MySQL, BMKG API, n8n, AI, dashboard, dan deployment berbasis Proxmox dapat digunakan sebagai model monitoring dan mitigasi risiko pusat data yang ringan, modular, adaptif, dan mudah dikembangkan.

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Berdasarkan hasil perancangan dan pengujian, arsitektur Big Data-IoT yang dikembangkan mampu mengintegrasikan data lingkungan internal dan data ancaman eksternal dalam satu alur monitoring pusat data. Sistem menerima data suhu dan kelembapan dari ESP32-DHT11, menyimpannya ke MySQL, mengambil informasi gempa dari BMKG API, meneruskan event ke n8n, serta mencatat rekomendasi mitigasi berbasis AI sebagai audit keputusan. Alur ini merepresentasikan proses Big Data melalui akuisisi data, ingestion, penyimpanan historis, pemrosesan event, analisis, dan penyajian informasi.

Integrasi Fastify, MySQL, BMKG API, n8n, AI, dashboard, dan deployment berbasis Proxmox membentuk pipeline yang modular dan mudah dikembangkan. Backend berperan sebagai pusat ingestion dan integrasi data, MySQL sebagai penyimpanan, n8n sebagai orkestrator workflow, AI sebagai komponen analisis, dan dashboard sebagai media penyajian informasi.

Kontribusi utama penelitian ini adalah model awal monitoring dan mitigasi risiko pusat data yang menggabungkan data IoT fisik dengan data seismik publik. Hasil penelitian menunjukkan bahwa data lingkungan dapat diperkaya dengan konteks ancaman eksternal sehingga menghasilkan rekomendasi mitigasi yang lebih terdokumentasi, adaptif, dan bernilai bagi proses pengambilan keputusan.

4.2 Saran

Berdasarkan hasil penelitian, pengembangan selanjutnya dapat difokuskan pada tiga aspek utama. Pertama, peningkatan perangkat dengan menggunakan sensor yang lebih presisi, seperti DHT22 atau sensor industri, serta penambahan sensor lain untuk memperluas cakupan monitoring. Kedua, penguatan analisis dan keamanan melalui pengembangan workflow n8n-AI yang mempertimbangkan magnitudo, jarak episentrum, kondisi termal, dan riwayat kejadian, disertai autentikasi serta validasi pada endpoint API dan webhook. Ketiga, pengujian pada lingkungan pusat data nyata untuk mengevaluasi stabilitas, latensi, dan ketahanan sistem, termasuk optimasi deployment Proxmox melalui monitoring resource, backup otomatis, dan konfigurasi reverse proxy yang lebih aman.

5. DAFTAR RUJUKAN

[1] Al-Qarafi, A., et al., "IoT-Based Environmental Monitoring System for Data Centers," *IEEE Access*, vol. 10, pp. 45120-45132, 2022.

[2] Arifin, Z., et al., "Design of Smart Environmental Monitoring for Data Center using NodeMCU and DHT11," *Journal of Physics: Conference Series*, vol. 1803, no. 1, 2021.

[3] Carvalho, V. R., et al., "Evaluating Proxmox VE for Edge IoT Data Processing Deployments," *Procedia Computer Science*, vol. 215, pp. 112-120, 2023.

[4] Chen, Y., et al., "AIOps for Predictive Cooling and Thermal Management in Data Centers," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1589-1598, 2023.

[5] Kim, T. H., et al., "Workflow Automation in Smart IT Environments using n8n and Node-RED," *Building and Environment*, vol. 204, 108154, 2021.
[6] Lee, J. K., et al., "Optimizing HTTP Payload for Lightweight IoT Devices," *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11520-11530, 2021.
[7] Nguyen, H. M., et al., "Seismic Risk Assessment and Mitigation Strategies for Critical Data Center Infrastructure," *Reliability Engineering & System Safety*, vol. 218, 108160, 2022.
[8] Prabowo, W. A., et al., "Metodologi Prototyping pada Pengembangan Sistem Pemantauan Infrastruktur IoT," *Jurnal Informatika Universitas Pamulang*, vol. 8, no. 1, pp. 45-52, 2023.
[9] Pratama, M. R., et al., "Arsitektur Big Data untuk Analisis Log Termal pada Smart Data Center," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 10, no. 3, pp. 589-598, 2023.
[10] Purwanto, G. H., et al., "Analisis Kinerja Proxmox Virtual Environment pada Infrastruktur Server Berbasis Cloud," *Jurnal Komputer Terapan*, vol. 7, no. 2, pp. 112-121, 2021.
[11] Putra, D. S., et al., "Implementasi API BMKG untuk Sistem Peringatan Dini Gempa Bumi Berbasis IoT," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, vol. 10, no. 4, pp. 312-320, 2021.
[12] Rahmadani, E. F., et al., "Pemanfaatan n8n sebagai Orkestrator Alur Kerja Berbasis Low-Code pada Sistem IoT," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 1, pp. 88-95, 2023.
[13] Rao, B. P., et al., "Performance Evaluation of Node.js and Fastify for High-Throughput API Gateways," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 5, pp. 215-222, 2022.
[14] Santoso, U. B., et al., "Dashboard Monitoring Kinerja Perangkat Data Center menggunakan Integrasi Restful API," *Jurnal Sistem Informasi Bisnis*, vol. 12, no. 1, pp. 76-84, 2022.
[15] Setiawan, R., et al., "Integrasi Sensor DHT11 dan ESP32 untuk Pemantauan Suhu Ruang Server," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 2, pp. 410-419, 2022.
[16] Sharma, P. K., et al., "A Secure and Efficient Architecture for IoT-Based Data Center Monitoring," *Future Generation Computer Systems*, vol. 115, pp. 240-250, 2021.
[17] Smith, N. A., et al., "Applying Large Language Models and AI for IT Operations (AIOps)," *ACM Computing Surveys*, vol. 55, no. 10, pp. 1-35, 2022.
[18] Wibowo, S. A., et al., "Penggunaan MySQL dan Node.js dalam Pengelolaan Data IoT Skala Menengah," *Jurnal Infotel*, vol. 15, no. 2, pp. 134-141, 2023.
[19] Wei, D., Luo, J., Liu, G., Liu, X., and Liu, J., "AIOps Architecture in Data Center Site Infrastructure Monitoring," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–10, 2022.
[20] Zahrani, F. A., "An Edge Computing Architecture for Data Center Environmental Monitoring using ESP32," *Sensors*, vol. 22, no. 8, 2955, 2022.