

Perancangan Sistem Monitoring Kampus Terpusat Berbasis IoT

Martin Emmanuel Chang
IoT & Electrical Engineering
Calvin Institute of Technology
Jakarta, Indonesia
mchang99@students.calvin.ac.id

Wesley Hakim
IoT & Electrical Engineering
Calvin Institute of Technology
Jakarta, Indonesia
whakim11@students.calvin.ac.id

Erwin Anggadjaja, Ph.D.
IoT & Electrical Engineering
Calvin Institute of Technology
Jakarta, Indonesia
erwin.anggadjaja@calvin.ac.id

Aditya Heru Prathama, Ph.D
IoT & Electrical Engineering
Calvin Institute of Technology
Jakarta, Indonesia
aditya.prathama@calvin.ac.id

Abstract - Penelitian ini mengembangkan sistem pemantauan kampus terpadu berbasis *Internet of Things* (IoT) dengan fokus terhadap modularitas, skalabilitas dan pemrosesan data secara *real-time*. Memanfaatkan ESP8266 dan protokol MQTT, sistem mengintegrasikan berbagai sensor lingkungan—termasuk suhu, kelembaban, kualitas udara, cahaya, serta pengukuran energy dan aliran air—untuk memberikan visualisasi interaktif melalui dashboard berbasis Django. Prototipe diuji dalam skenario nyata di lingkungan *Calvin Institute of Technology* untuk mengevaluasi keandalan, kemudahan implementasi, dan efisiensi pengelolaan data. Studi ini menyimpulkan bahwa framework IoT ini efektif untuk implementasi smart campus dan memberikan rekomendasi integrasi analitik prediktif dan security hardening untuk pengembangan selanjutnya.

Kata kunci: sistem pemantauan kampus, *Internet of Things*, skalabilitas

I. Pendahuluan

Transformasi digital dalam pengelolaan fasilitas pendidikan mendorong kebutuhan akan sistem monitoring yang terpusat, efisien, dan *real-time*. Kampus modern seperti *Calvin Institute of Technology* menghadapi tantangan dalam mengintegrasikan data dari berbagai sumber fasilitas *indoor* dan *outdoor*, akibat sistem yang

masih bersifat konvensional dan terfragmentasi. Dengan luas area sebesar 1,4 hektar dan rencana ekspansi ke kawasan BSD, kebutuhan akan solusi berbasis *Internet of Things* (IoT) yang modular, *scalable*, dan terjangkau menjadi semakin mendesak untuk mendukung implementasi smart campus.

Permasalahan utama yang dihadapi adalah bagaimana merancang arsitektur sistem *monitoring* kampus yang mampu beradaptasi dengan dinamika lingkungan kampus, memilih protokol komunikasi serta platform visualisasi yang tepat, dan menjamin keamanan serta keandalan sistem. Untuk itu, penting merumuskan pendekatan yang memungkinkan integrasi sensor-sensor lingkungan, komunikasi data yang efisien, serta dashboard visualisasi yang mendukung pengambilan keputusan berbasis data secara *real-time*.

Penelitian ini bertujuan mengembangkan prototipe sistem monitoring kampus terpusat berbasis IoT dengan memanfaatkan perangkat ESP8266, protokol MQTT, dan dashboard berbasis Django. Selain itu, penelitian ini akan mengevaluasi performa sistem dari segi *latency*, modularitas, dan keandalannya dalam skenario *Proof of Concept* (PoC) di berbagai area representatif dalam lingkungan kampus. Dengan desain yang mengutamakan kemudahan penambahan *station* baru dan akses data secara fleksibel, sistem ini diharapkan dapat menjadi pondasi pengembangan *smart campus* ke depan.

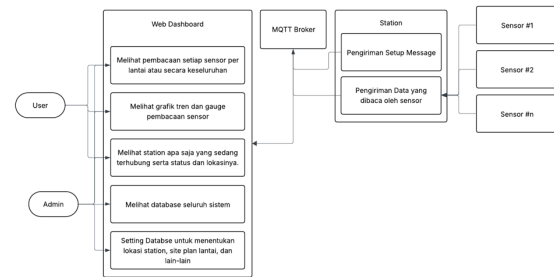
II. Spesifikasi

Sistem *monitoring* kampus ini menggunakan protokol komunikasi Wi-Fi melalui modul ESP8266 yang dipilih karena keunggulannya dari segi biaya, kemudahan produksi, skalabilitas, dan keandalan. Komunikasi data antar perangkat dilakukan menggunakan protokol MQTT dengan QoS 1, serta dilengkapi mekanisme *reconnect* otomatis saat koneksi terputus. Deteksi status koneksi dilakukan dengan mengandalkan ping interval lima detik, sementara topik-topik MQTT ditangani oleh broker untuk memastikan efisiensi komunikasi.

Dari sisi modularitas, sistem ini mendukung deteksi otomatis terhadap status sensor melalui monitoring heartbeat dan implementasi *plug-and-play*, serta antarmuka pengguna yang secara otomatis membentuk tampilan dan grafik untuk setiap *station* baru. Dashboard dibangun menggunakan Django Framework, menghadirkan visualisasi interaktif dalam bentuk grafik *real-time* dan rekap historis. Dashboard sendiri dapat diakses sesuai dengan peran pengguna, baik untuk administrator maupun mahasiswa, melalui jaringan Wi-Fi kampus.

III. Metode Penelitian

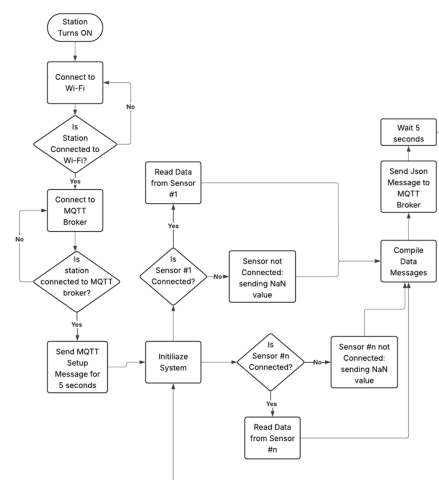
Arsitektur sistem monitoring kampus berbasis IoT yang dirancang dalam proyek ini terdiri dari beberapa komponen utama seperti sensor, modul *station*, MQTT Broker, dan Web Dashboard. Sensor-sensor terhubung ke *station* untuk membaca parameter lingkungan, lalu data dikirim melalui protokol MQTT ke dashboard. *Station* juga mengirimkan pesan setup berisi metadata yang digunakan untuk membentuk tampilan dinamis di *dashboard*. Web *dashboard* memungkinkan pengguna umum (*user*) melihat pembacaan sensor dan grafik tren, sedangkan admin memiliki akses tambahan untuk memantau status koneksi setiap *station* dan mengatur konfigurasi sistem melalui pengelolaan basis data. Diagram pada Gambar 1 menjadi dasar bagi seluruh tahapan metode penelitian yang dijelaskan selanjutnya.



Gambar 1 Diagram arsitektur

A. Perancangan *Station*

Sistem *station* dirancang dengan alur operasional yang terstruktur seperti terlihat pada gambar 2. Ketika *station* dinyalakan, langkah pertama yang dilakukan adalah mencoba melakukan koneksi ke jaringan Wi-Fi. Jika koneksi gagal, *station* akan terus melakukan percobaan hingga berhasil terhubung. Setelah konektivitas Wi-Fi berhasil, *station* kemudian terhubung ke MQTT broker untuk komunikasi data. Setelah konfirmasi koneksi MQTT berhasil, sistem melakukan inisialisasi dan memulai fungsi utamanya dengan memverifikasi koneksi ke berbagai sensor. Sistem secara metodis memeriksa status koneksi setiap sensor. Untuk sensor yang terhubung, operasi pembacaan data segera dimulai, sedangkan sensor yang tidak terhubung ditandai dengan nilai NaN (*Not a Number*) untuk menjaga integritas data. Semua data yang dikumpulkan, baik pembacaan valid maupun indikator error, dikompilasi menjadi pesan JSON terstruktur yang kemudian dikirimkan ke MQTT broker dalam interval 5 detik, memastikan pembaruan data secara berkala sekaligus mencegah kemacetan jaringan.



Gambar 2 Diagram sistem *station*

Semua *station* diimplementasikan menggunakan mikrokontroler ESP8266, dipilih karena kemampuan nirkabel yang unggul dan efisiensi biaya. *Casing* pelindung khusus telah dikembangkan untuk setiap jenis *station* untuk memastikan ketahanan di lingkungan penempatan masing-masing. Sistem pemantauan terdiri dari empat jenis *station* khusus: *Indoor Station* (untuk metrik lingkungan interior), *Outdoor Station* (untuk pemantauan iklim eksterior), *Energy Station* (untuk pelacakan konsumsi energi), dan *Water Station* (untuk pemantauan penggunaan dan kualitas air), masing-masing dirancang untuk memenuhi kebutuhan pemantauan spesifik. Tabel 1 menunjukkan biaya yang akan dikeluarkan untuk perancangan proyek ini.

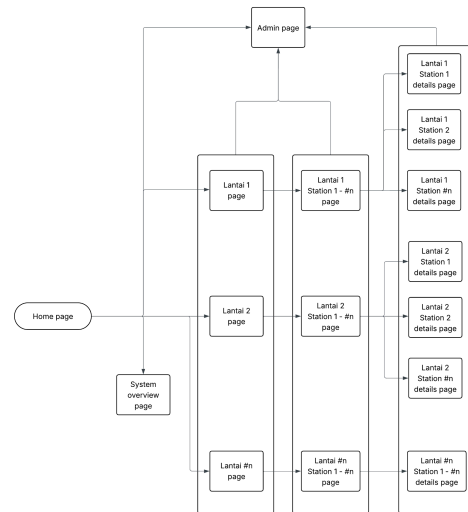
Tabel 1 Biaya perancangan sistem

No	Barang	Total Harga
1	Mikrokontroler dan Sensor	Rp1.656.300
2	<i>Casing</i> 3D Print	Rp154.000
3	Raspberry Pi	<i>Disediakan oleh pihak kampus</i>
4	Kabel, Konektor dan Alat pendukung lain	Rp244.858
Total		Rp2.055.158

B. Perancangan Dashboard

Arsitektur *dashboard* sistem pemantauan dirancang dengan struktur hierarkis yang komprehensif dan intuitif, seperti ditunjukkan pada gambar 3. Halaman utama (*Home page*) berfungsi sebagai titik akses sentral yang mengarahkan pengguna ke tiga jalur navigasi utama: halaman Admin untuk pengelolaan sistem, halaman *System Overview* untuk memantau kondisi keseluruhan, dan halaman khusus untuk setiap lantai (Lantai 1 hingga Lantai #n). Dari halaman lantai, pengguna dapat mengakses detail spesifik untuk setiap *station* yang terpasang pada lantai tersebut, dengan tampilan yang menggabungkan informasi *station* dari Lantai 1 *Station* 1 hingga Lantai #n *Station* #n. Struktur berlapis ini memungkinkan navigasi yang efisien mulai dari pandangan makro hingga detail mikro, memudahkan pemantauan data dari seluruh

gedung hingga ke tingkat sensor individual pada *station* tertentu, sehingga memaksimalkan kemampuan analisis dan pemantauan secara *real-time*.



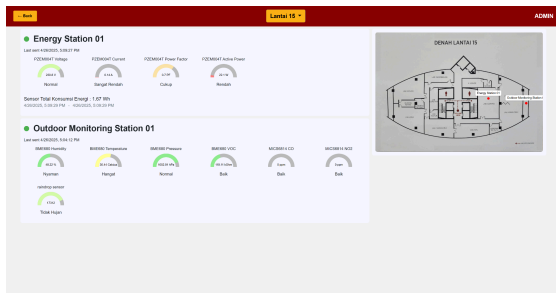
Gambar 3 Diagram alur dashboard

Dashboard juga dirancang untuk secara otomatis menghasilkan halaman dan blok grafik baru setiap kali *station* baru ditambahkan, berdasarkan metadata lokasi dan parameter yang dikirimkan saat setup awal, termasuk batas pengukuran, skala warna, konfigurasi visual yang dinamis, serta informasi lain yang terhubung dengan *station*. Mekanisme ping-pong antara server dan *station* memastikan kontinuitas operasi dengan memantau keaktifan setiap unit secara berkala. Dengan arsitektur ini, sistem siap dikembangkan lebih lanjut untuk mendukung lebih banyak *station* dan integrasi fitur analitik lanjutan tanpa memerlukan perubahan signifikan pada struktur kode dasar.

IV. IMPLEMENTASI, PENGUJIAN, DAN HASIL



Gambar 4 Foto semua *station*



Gambar 5 Halaman *dashboard* lantai

Gambar 4 dan Gambar 5 memperlihatkan rancangan lengkap seluruh stasiun dan tampilan dashboard yang telah dikembangkan. Sebelum implementasi di lingkungan kampus, sistem akan melalui serangkaian pengujian untuk memastikan semua komponen berfungsi dengan baik. Setelah itu, sistem akan diuji dalam dua tahap pengujian stres guna mengevaluasi performa dan ketahanan jangka panjang dalam berbagai skenario penggunaan.

Pengujian pertama adalah **Pengujian fungsional** dilaksanakan dalam tiga tahap utama:

- **Kestabilan Koneksi:** Koneksi antara MQTT broker dan server diuji selama tiga jam non-stop. Hasil menunjukkan tingkat uptime 100% tanpa terdeteksi downtime, sesuai ekspektasi, mengingat MQTT broker terintegrasi pada server kampus yang memiliki tingkat stabilitas tinggi.
- **Keandalan Pengiriman Data:** Pengujian dilakukan dengan mengaktifkan satu hingga empat stasiun secara simultan selama 30 menit. Rata-rata tingkat keberhasilan pengiriman (*delivery rate*) melebihi 95%, dengan interval pengiriman sekitar dua detik.
- **Respons Terhadap Gangguan Koneksi:** Kemampuan sistem untuk melakukan koneksi ulang otomatis diuji dengan memutus dan mengaktifkan kembali koneksi Wi-Fi. Semua stasiun berhasil tersambung kembali dalam waktu rata-rata 12 detik, yang tergolong memuaskan.

Pengujian Kedua adalah **Pengujian Modularitas Sensor** dimana sistem akan diuji dalam mendeteksi pelepasan dan pemasangan kembali sensor secara instan. Ketika sensor dilepas, dashboard otomatis menampilkan status "*none*". Setelah sensor

dipasang kembali, data pembacaan normal muncul kembali dalam waktu kurang dari satu detik. Selain itu, proses penambahan stasiun baru diuji dengan mengosongkan basis data dan mendaftarkan dua stasiun tambahan. Kedua stasiun langsung dikenali dan tampil pada dashboard dalam waktu kurang dari satu detik, menunjukkan responsivitas sistem terhadap perubahan konfigurasi.

Pengujian terakhir adalah **Evaluasi Antarmuka Dashboard**. Sebanyak 29 responden mengisi kuesioner untuk menilai pengalaman pengguna (UX) antarmuka dashboard. Hasil rata-rata skor adalah:

- Kemudahan navigasi: 4,86/5
- Konsistensi tampilan: 4,89/5
- Kenyamanan penggunaan: 4,79/5
- Kemudahan pemahaman grafik: 4,83/5
- Kejelasan informasi: 4,86/5

Fitur rekap dan histori data telah berfungsi dengan baik, mampu menampilkan data hingga tiga hari sebelumnya dan mendukung ekspor ke format CSV. Kecepatan pembaruan data real-time rata-rata 2,6 detik, masih di bawah ambang batas yang ditetapkan, meski tetap bergantung pada kestabilan jaringan internet.

Setelah semua pengujian biasa, sistem akan diuji dalam dua tahap pengujian stres.

Tahap Pertama: Simulasi skenario dilakukan pada setiap stasiun selama dua jam berturut-turut, mencakup tindakan seperti pemutusan dan penyambungan catu daya, pemasangan dan pelepasan sensor, simulasi pembacaan seperti asap, pemadaman lampu, pengaliran air, dan lainnya. Hasil pengujian menunjukkan bahwa setiap stasiun merespons dengan baik, dan dashboard mampu memperbarui data serta status stasiun dalam waktu kurang dari 10 detik, dengan catatan jaringan Wi-Fi dalam kondisi stabil.

Tahap Kedua: Sistem dijalankan secara terus-menerus selama 19 jam 44 menit. Dalam periode tersebut, tercatat sebanyak 749.962 baris data berhasil dikirim oleh seluruh stasiun. Namun, menjelang akhir pengujian, server berbasis Raspberry Pi mengalami keterbatasan dalam hal komputasi dan kapasitas penyimpanan. Hal ini menyebabkan proses penyimpanan data ke basis data serta pengambilan data untuk tampilan

dashboard menjadi sangat lambat, sehingga halaman dashboard sulit dimuat. Meskipun demikian, berdasarkan hasil ekspor dari basis data, seluruh stasiun dan sensor tetap beroperasi dengan baik sepanjang durasi pengujian stres.

IV. Kesimpulan

Proyek ini berhasil merancang dan mengimplementasikan sistem monitoring kampus berbasis IoT yang terintegrasi, mencakup pemantauan kondisi lingkungan dalam dan luar ruangan, penggunaan energi serta konsumsi air, dengan antarmuka dasbor yang mudah diakses melalui jaringan Wi-Fi kampus. Sistem ini menjadi solusi atas tantangan pemantauan kondisi kampus secara menyeluruh dan *real-time* tanpa memerlukan infrastruktur tambahan yang rumit. Dari hasil pengujian, sistem menunjukkan kinerja yang andal dan responsif dalam menyajikan data secara berkelanjutan.

Untuk pengembangan selanjutnya, sistem ini dapat diterapkan dalam skala yang lebih besar di seluruh area kampus dengan menambah jumlah *station*. Selain itu, Raspberry Pi dapat digantikan dengan server khusus untuk meningkatkan daya pemrosesan serta kemampuan menangani lebih banyak pengguna dan data secara bersamaan. Dengan arsitektur yang fleksibel dan mudah diadaptasi, sistem ini memiliki potensi besar sebagai fondasi bagi pengembangan smart campus yang cerdas, efisien, dan berkelanjutan.

DAFTAR PUSTAKA

- [1] B. I. AHMAD *et al.*, “AN IoT BASED SMART CAMPUS ARCHITECTURE FOR INSTITUTIONS IN DEVELOPING COUNTRIES,” *i-manager’s Journal on Embedded Systems*, vol. 7, no. 1, p. 18, 2018, doi: <https://doi.org/10.26634/jes.7.1.15635>.
- [2] Bhargava, “Leveraging Django: A Web Framework for Efficient Infrastructure Monitoring Tools,” *Ijraset.com*, 2023, <https://www.ijraset.com/research-paper/leveraging-django-a-web-framework-for-efficient-infrastructure-monitoring-tools> (accessed Apr. 11, 2025).
- [3] P. Nikolich, “Wi-Fi for IoT gives organizations low-cost connection option,” *Search IoT*, 2020, <https://www.techtarget.com/iotagenda/feature/Wi-F>
- i-for-IoT-gives-organizations-low-cost-connection-option* (accessed Apr. 11, 2025).
- [4] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, “MQTT-S — A publish/subscribe protocol for Wireless Sensor Networks,” *IEEE Xplore*, Jan. 01, 2008, <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4554519>
- [5] C. T. U. 11301 S. R. Laurel and Md 20708 888.522.7486, “IoT in Smart Campus Development | Capitol Technology University,” www.captechu.edu, Mar. 04, 2024, <https://www.captechu.edu/blog/smart-classrooms-connected-dorms-how-iot-transforming-college-campuses>
- [6] S. Sebastin Antony Joe, et al., “IoT-based Smart Campus Monitoring Based on an Improved Chimp Optimization Algorithm,” *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, vol. 11, no. 1, pp. 213-222, 2023, <http://section.iaesonline.com/index.php/IJEI/article/download/4410/815>
- [7] S. S. Manvi dan G. K. Shyam, “Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey,” *Journal of Network and Computer Applications*, vol. 41, pp. 424–440, 2014, https://www.researchgate.net/publication/259143215_Resource_management_for_Infrastructure_as_a_Service_IaaS_in_cloud_computing_A_survey
- [8] H. Suo, J. Wan, C. Zou, and J. Liu, “Security in the Internet of Things: A Review,” 2012 International Conference on Computer Science and Electronics Engineering, pp. 648–651, 2012, <https://ieeexplore.ieee.org/document/6188257>
- [9] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.