

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian mengenai sistem monitoring lingkungan telah dilakukan dengan memanfaatkan teknologi sensor dan sistem komunikasi untuk memperoleh informasi mengenai kondisi lingkungan. Kumar et al. membahas perkembangan sistem monitoring lingkungan serta teknologi yang digunakan dalam proses pengukuran dan pengumpulan data. Kajian tersebut menunjukkan bahwa sistem monitoring lingkungan melibatkan proses akuisisi data dari sensor sebagai bagian penting dalam memperoleh informasi mengenai parameter yang diamati. Penelitian ini relevan sebagai dasar pengembangan sistem karena data hasil pengukuran dari perangkat sensor menjadi sumber data yang selanjutnya dikelola dan ditampilkan pada sistem monitoring yang dikembangkan (A. Kumar et al., 2013).

Pengembangan sistem monitoring kualitas udara berbasis IoT juga dibahas pada penelitian yang berjudul “*Smart Air Quality Monitoring Using IoT*” melalui kajian terhadap berbagai sistem pemantauan kualitas udara yang memanfaatkan perangkat terhubung. Kajian tersebut membahas penggunaan sensor dan teknologi IoT dalam mendukung proses monitoring kualitas udara serta berbagai aspek yang berkaitan dengan pengembangan sistem pemantauan. Penelitian tersebut relevan dengan penelitian ini karena sama-sama berkaitan dengan pengelolaan data hasil pengukuran parameter lingkungan untuk mendukung proses pemantauan (L. R. Kumar et al., 2024).

Pada aspek kualitas data, Castell et al. (2017), melakukan penelitian mengenai penggunaan platform sensor berbiaya rendah untuk monitoring kualitas udara dan estimasi paparan. Penelitian tersebut mengevaluasi potensi penggunaan sensor low-cost sekaligus menunjukkan adanya keterbatasan yang perlu diperhatikan dalam pemanfaatan hasil pengukuran sensor. Hasil tersebut relevan dengan penelitian ini karena data yang disimpan dan divisualisasikan dalam sistem monitoring bergantung pada data hasil pengukuran dari perangkat sensor, sehingga kualitas data menjadi salah satu aspek yang perlu diperhatikan dalam

proses pemantauan (Castell et al., 2017)

Dalam pengelolaan data, Bader et al. (2017), melakukan survei dan perbandingan terhadap beberapa basis data *open-source time-series*. Penelitian tersebut membahas karakteristik berbagai basis data yang digunakan untuk menangani data deret waktu. Kajian ini relevan karena data monitoring yang dihasilkan oleh sensor memiliki informasi waktu pengukuran sehingga pengelolaan data berdasarkan karakteristik waktu menjadi salah satu pertimbangan dalam pemilihan sistem basis data (Bader et al., 2017).

Penelitian mengenai analisis performa basis data time-series untuk aplikasi IoT telah dilakukan oleh Daqouri dengan judul “*Performance Analysis of Time Series Databases for IoT Applications*”. Penelitian tersebut membandingkan performa InfluxDB dan TimescaleDB sebagai basis data time-series dengan MariaDB sebagai basis data relasional. Evaluasi dilakukan menggunakan empat jenis kueri, yaitu Key Lookup, Range, Aggregate, dan Time Sensitive Aggregate, dengan ukuran dataset mulai dari 1 juta hingga 50 juta data. Hasil penelitian menunjukkan bahwa performa masing-masing basis data berbeda berdasarkan jenis kueri dan ukuran dataset. Oleh karena itu, pemilihan basis data untuk aplikasi IoT perlu disesuaikan dengan kebutuhan dan karakteristik penggunaan sistem (Daqouri, 2024).

Sebagai salah satu teknologi basis data yang digunakan dalam pengelolaan data deret waktu, InfluxDB menyediakan sistem penyimpanan dan pengolahan data yang berorientasi pada time series. Dokumentasi InfluxDB menjelaskan fitur dan penggunaan basis data tersebut dalam pengelolaan data berbasis waktu. Dalam penelitian ini, InfluxDB digunakan untuk menyimpan data hasil monitoring yang memiliki informasi timestamp, sehingga dokumentasi tersebut menjadi referensi dalam penerapan dan penggunaan basis data pada sistem yang dikembangkan. (InfluxData, 2023).

Pada tahap penyajian data, Grafana menyediakan platform yang digunakan untuk membangun dashboard dan melakukan visualisasi data dari sumber data yang terhubung. Dokumentasi Grafana menjelaskan berbagai konsep dan fitur yang digunakan dalam proses pembuatan serta pengelolaan visualisasi data. Teknologi tersebut relevan dengan penelitian ini karena Grafana digunakan

sebagai platform untuk menampilkan data hasil monitoring dalam bentuk dashboard sehingga pengguna dapat melakukan pemantauan terhadap informasi yang tersedia pada sistem (Grafana Labs, 2023).

Berdasarkan penelitian dan referensi yang telah dikaji, pengembangan sistem monitoring mencakup beberapa aspek, mulai dari pengambilan data lingkungan, kualitas hasil pengukuran, hingga pengelolaan dan penyajian data. Kajian tersebut menunjukkan bahwa data hasil pengukuran tidak hanya perlu dikumpulkan, tetapi juga dikelola dan disajikan agar dapat mendukung proses pemantauan. Berdasarkan hal tersebut, penelitian ini berfokus pada pengembangan aspek pengelolaan dan visualisasi data dari sistem monitoring yang telah tersedia sebelumnya. Data hasil pengukuran dari perangkat sensor digunakan sebagai masukan sistem untuk selanjutnya disimpan dan ditampilkan melalui dashboard.

Sistem yang menjadi dasar pengembangan penelitian ini merupakan hasil implementasi sebelumnya pada Project PBL Sistem Instrumentasi Lingkungan. Sistem tersebut menggunakan PC server berbasis Debian Linux sebagai pusat pengelolaan yang terintegrasi dengan MQTT Broker untuk komunikasi data antarperangkat, Node-RED untuk pengolahan alur data, serta MongoDB sebagai media penyimpanan. Infrastruktur tersebut telah mendukung proses pengiriman, pengolahan, dan penyimpanan data secara terpusat dalam jaringan lokal. Pada penelitian ini, infrastruktur yang telah tersedia dimanfaatkan kembali dengan penambahan InfluxDB untuk pengelolaan data berbasis *time-series* dan Grafana sebagai platform visualisasi. Dengan pengembangan tersebut, penelitian difokuskan pada penyimpanan dan penyajian data hasil monitoring agar informasi dapat ditampilkan melalui dashboard sesuai dengan kebutuhan sistem.

2.2 Landasan Teori

2.2.1 Internet Of Things (IOT)

Internet of Things (IoT) didefinisikan sebagai sebuah paradigma di mana objek-objek fisik yang dilengkapi dengan sensor, aktuator, dan kemampuan pemrosesan dihubungkan melalui jaringan internet untuk saling berkomunikasi dan bertukar data. IoT bukan sekadar menghubungkan perangkat ke internet, melainkan sebuah ekosistem cerdas yang mampu melakukan identifikasi,

penginderaan, dan pemrosesan informasi secara otomatis tanpa memerlukan intervensi manusia yang intensif (Gubbi et al., 2012).

Arsitektur IoT umumnya terbagi menjadi tiga lapisan utama, yaitu *perception layer* sebagai lapisan pengumpul data dari lingkungan fisik menggunakan sensor, *network layer* sebagai media transmisi data melalui jaringan komunikasi, serta *application layer* sebagai lapisan yang menyajikan informasi kepada pengguna dalam bentuk yang mudah dipahami.

Implementasi IoT saat ini telah berkembang pesat di berbagai bidang, seperti sistem pemantauan lingkungan, kesehatan, pertanian cerdas, hingga otomatisasi industri. Keberhasilan implementasi IoT sangat dipengaruhi oleh efisiensi komunikasi data, keandalan perangkat sensor, serta skalabilitas sistem penyimpanan data yang digunakan (Atzori et al., 2025).

2.2.2 Akademi Sistem Monitoring Kualitas Udara

Sistem monitoring kualitas udara merupakan sebuah kesatuan infrastruktur yang dirancang untuk mengumpulkan, mengolah, dan menyajikan data parameter polutan di udara ambien secara *real-time*. Sistem ini memanfaatkan sensor untuk mendeteksi berbagai jenis polutan, kemudian mengirimkan data tersebut ke server untuk dianalisis dan divisualisasikan.

Menurut Kumar et al. (2015), sistem *monitoring* berbasis IoT umumnya terdiri dari tiga komponen utama, yaitu lapisan sensor sebagai pengumpul data, lapisan komunikasi sebagai media transmisi data, serta lapisan aplikasi sebagai pusat pengolahan dan visualisasi data.

Keandalan sistem monitoring kualitas udara sangat bergantung pada *integritas* data yang dikirimkan oleh sensor serta kemampuan sistem dalam mengolah dan menyajikan informasi secara akurat. Selain itu, sistem juga harus mampu menangani data dalam jumlah besar secara kontinu, mengingat karakteristik data kualitas udara yang bersifat *Time-series* dan dihasilkan secara terus-menerus.

2.2.3 Indeks Standar Pencemar Udara (ISPU)

Indeks Standar Pencemar Udara (ISPU) merupakan parameter yang digunakan untuk menyampaikan informasi kualitas udara kepada masyarakat

berdasarkan tingkat konsentrasi polutan. ISPU diatur dalam Peraturan Menteri Lingkungan Hidup dan Kehutanan Republik Indonesia Nomor 14 Tahun 2020.

Dalam sistem monitoring kualitas udara, ISPU tidak hanya digunakan sebagai indikator nilai, tetapi juga diintegrasikan ke dalam sistem visualisasi untuk mempermudah pemahaman pengguna. Penerapan thresholding warna pada dashboard, seperti hijau untuk kategori baik hingga hitam untuk kategori berbahaya, memungkinkan pengguna memahami kondisi kualitas udara secara instan tanpa harus memahami satuan teknis seperti PPM atau $\mu\text{g}/\text{m}^3$.

Parameter utama yang digunakan dalam pengukuran kualitas udara meliputi PM_{2.5}, PM₁₀, karbon monoksida (CO), sulfur dioksida (SO₂), nitrogen dioksida (NO₂), dan ozon (O₃), yang masing-masing memiliki dampak berbeda terhadap kesehatan manusia (World Health Organization, 2021). Berikut table data ISPU :

Tabel 2.1 Tabel Konversi Batas Konsentrasi Parameter ke ISPU

ISPU	PM ₁₀ ($\mu\text{g}/\text{m}^3$)	PM _{2.5} ($\mu\text{g}/\text{m}^3$)	SO ₂ ($\mu\text{g}/\text{m}^3$)	CO ($\mu\text{g}/\text{m}^3$)	O ₃ ($\mu\text{g}/\text{m}^3$)	NO ₂ ($\mu\text{g}/\text{m}^3$)
0–50	50	15,5	52	4.000	120	80
51–100	150	55,4	180	8.000	235	200
101–200	350	150,4	400	15.000	400	1.130
201–300	420	250,4	800	30.000	800	2.260
>300	500	500	1.200	45.000	1.000	3.000

Perhitungan ISPU dilakukan berdasarkan nilai ISPU batas atas, ISPU batas bawah, ambien batas atas, ambien batas bawah, dan konsentrasi ambiens hasil pengukuran. Persamaan matematika perhitungan ISPU sebagai berikut:

$$I = \frac{Ia - Ib}{Xa - Xb}(Xx - Xb) + Ib$$

Keterangan :

I = ISPU terhitung

Ia = ISPU batas atas

Ib = ISPU batas bawah

Xa = Konsentrasi ambien batas atas

Xb = Konsentrasi ambien batas bawah

Xx = Konsentrasi ambien nyata hasil pengukuran

Tabel 2.2 Data ISPU

Nilai Ispu	Kategori	Penjelasan Dampak	Warna Indikator
0-50	Baik	Tidak memberikan dampak bagi kesehatan manusia/hewan.	Hijau
51-100	Sedang	Tidak berpengaruh pada kesehatan manusia, tetapi berpengaruh pada tumbuhan sensitif.	Biru
101-200	Tidak Sehat	Bersifat merugikan pada manusia, hewan, dan tumbuhan.	Kuning
201-300	Sangat Tidak Sehat	Kualitas udara yang dapat merugikan kesehatan pada sejumlah segmen populasi.	Merah
>300	Berbahaya	Kualitas udara yang secara umum dapat merugikan kesehatan yang serius.	Hitam

Berikut merupakan parameter utama yang umum digunakan dalam pengukuran kualitas udara:

1. PM2.5 (*Particulate Matter* $\leq 2.5 \mu\text{m}$)

Partikel halus dengan ukuran ≤ 2.5 mikrometer yang dapat masuk hingga ke paru-paru bahkan ke aliran darah. PM2.5 sangat berbahaya karena dapat menyebabkan gangguan pernapasan, penyakit *kardiovaskular*, serta menurunkan fungsi paru-paru.

2. PM10 (*Particulate Matter* $\leq 10 \mu\text{m}$)

Partikel debu dengan ukuran ≤ 10 mikrometer yang dapat terhirup dan masuk ke saluran pernapasan bagian atas. Paparan jangka panjang dapat menyebabkan iritasi saluran pernapasan dan gangguan kesehatan lainnya.

3. Karbon Monoksida (CO)

Gas beracun yang tidak berwarna dan tidak berbau, dihasilkan dari pembakaran tidak sempurna. CO dapat mengikat hemoglobin dalam darah sehingga mengurangi kemampuan darah dalam mengangkut oksigen.

4. Sulfur Dioksida (SO₂)

Gas yang dihasilkan dari pembakaran bahan bakar fosil yang mengandung sulfur. SO₂ dapat menyebabkan iritasi pada sistem pernapasan dan berkontribusi terhadap hujan asam.

5. Nitrogen Dioksida (NO₂)

Gas hasil emisi kendaraan bermotor dan proses industri. NO₂ dapat

menyebabkan peradangan pada saluran pernapasan serta menurunkan fungsi paru-paru

6. Ozon (O₃)

Ozon di permukaan tanah (*ground-level ozone*) merupakan polutan sekunder yang terbentuk dari reaksi kimia antara gas pencemar di bawah sinar matahari. O₃ dapat menyebabkan iritasi mata, gangguan pernapasan, serta memperburuk penyakit asma.

2.2.4 MQTT Broker

Dalam sistem monitoring kualitas udara berbasis IoT, pengiriman data dari perangkat sensor ke server merupakan komponen yang sangat krusial. Sistem ini menggunakan protokol MQTT (*Message Queuing Telemetry Transport*), yaitu protokol komunikasi ringan yang bekerja berdasarkan arsitektur *publish/subscribe*.

MQTT broker berfungsi sebagai pusat distribusi data yang menerima pesan dari publisher dan meneruskannya ke *subscriber* sesuai dengan topik yang telah ditentukan. Pendekatan ini memungkinkan komunikasi yang efisien dengan penggunaan bandwidth yang rendah, sehingga sangat cocok untuk perangkat IoT dengan keterbatasan sumber daya (Banks & Gupta, 2014).

Penggunaan MQTT juga memungkinkan pengiriman data secara fleksibel, baik berdasarkan *interval* waktu tertentu maupun berdasarkan perubahan nilai sensor, sehingga dapat mengoptimalkan penggunaan daya dan jaringan.

Tabel 2.3 Ringkasan Alur Data MQTT

Tahapan	Aksi Teknis	Perangkat/Software
<i>Data Acquisition</i>	Pembacaan nilai analog/digital dari 5 sensor.	ESP32 + MQ-7, SO ₂ , O ₃ , NO ₂ , PM _{2.5}
<i>Packet Formatting</i>	Konversi nilai sensor menjadi format Line Protocol.	Arduino IDE
<i>Publishing</i>	Pengiriman paket ke Broker melalui port 1883.	ESP32 (WiFi)
<i>Routing</i>	Broker mengarahkan data berdasarkan Topic.	Mosquitto Broker
<i>Storage</i>	Data ditarik oleh Subscriber dan ditulis ke DB.	InfluxDB

2.2.5 InfluxDB (Time-series Database)

InfluxDB merupakan basis data yang dirancang khusus untuk menangani data berbasis waktu (*Time-series data*), yaitu data yang dicatat secara berurutan berdasarkan waktu tertentu. Data jenis ini banyak digunakan dalam sistem monitoring dan IoT karena karakteristiknya yang terus bertambah secara kontinu.

Berbeda dengan database relasional tradisional, InfluxDB dioptimalkan untuk beban tulis tinggi (*high write throughput*) dan kueri berbasis rentang waktu. Hal ini memungkinkan sistem untuk menyimpan dan mengakses data dalam jumlah besar dengan performa yang tetap stabil (InfluxData, 2023).

InfluxDB menyediakan fitur retention policy untuk mengatur masa simpan data serta struktur data berupa *measurement*, *tag*, *field*, dan *timestamp* yang mendukung pengelolaan data *Time-series* secara efisien. Manajemen data pada InfluxDB dalam sistem ini meliputi beberapa tahapan sebagai berikut:

1) Penerimaan data

Data kualitas udara dikirim dari perangkat IoT secara berkala dan diterima oleh InfluxDB sebagai data *time-series*.

2) Validasi data

Data yang masuk diperiksa untuk memastikan tidak terjadi kesalahan pembacaan atau data kosong sebelum disimpan.

3) Penyimpanan data

Data disimpan dalam bentuk *measurement* dengan struktur *tag*, *field*, dan *timestamp* sehingga tersusun berdasarkan waktu.

4) Pengelompokan dan indexing data

Tag digunakan sebagai metadata untuk mengelompokkan data berdasarkan lokasi atau identitas perangkat sehingga proses pencarian menjadi lebih cepat.

5) Pengaturan masa simpan dan penghapusan data

InfluxDB menggunakan retention policy untuk mengatur lama penyimpanan data sehingga data lama dapat dihapus secara otomatis. Selain itu, penghapusan data juga dapat dilakukan secara manual menggunakan query jika diperlukan.

6) Pengolahan dan agregasi data

Data dapat diolah menjadi nilai rata-rata, maksimum, dan minimum berdasarkan interval waktu tertentu untuk kebutuhan analisis.

7) Akses dan visualisasi data

Data yang tersimpan dapat diakses kembali dan divisualisasikan menggunakan Grafana untuk memudahkan pemantauan.

Kemampuan InfluxDB dalam melakukan agregasi data seperti rata-rata, maksimum, dan minimum menjadikannya sangat sesuai untuk sistem monitoring kualitas udara yang membutuhkan analisis tren dalam jangka waktu tertentu. Dengan karakteristik tersebut, InfluxDB menjadi solusi yang tepat untuk sistem monitoring kualitas udara berbasis IoT yang membutuhkan penyimpanan data kontinu, analisis berbasis waktu, serta visualisasi data secara cepat dan akurat. Berikut Gambar 2.1 struktur data influxdb line protocol.

```
myMeasurement,tag1=value1,tag2=value2 fieldKey="fieldValue" 1556813561098000000
```

Gambar 2.1 Struktur Data InfluxDB Line Protocol

Pada Gambar 2.1 struktur data influxdb line protocol ini terdapat 4 komponen utama yang dapat dilihat pada Table 2.4 komponen *line protocol* pada system.

Tabel 2.4 Komponen Line Protocol pada Sistem

Komponen	Contoh Implementasi	Keterangan	Type Data
<i>Measurement</i>	Monitoring_Kualitas_Udara	Nama tabel/wadah data	<i>String</i>
<i>Tag Set</i>	lokasi=PCR, id_alat=ESP32_01	Metadata untuk kueri cepat	<i>String (Indexed)</i>
<i>Field Set</i>	co_val=10.5, so2_val=0.2	Nilai mentah dari 5 sensor	<i>Float/Integer</i>
<i>Timestamp</i>	1740890000000000000	Penanda waktu nanodetik	<i>Unix Epoch</i>

1. *Measurement*

Measurement merupakan nama wadah atau kategori data. Konsep ini mirip dengan tabel pada database relasional.

Contoh : *Monitoring _Kualitas_Udara*

Pada sistem ini, *measurement* digunakan untuk mengelompokkan seluruh parameter yang berkaitan dengan kualitas udara.

2. *Tag Set*

Tag adalah metadata yang digunakan untuk proses *filtering* dan pengelompokan data secara cepat. Tag bersifat *indexed*, sehingga pencarian data berdasarkan tag menjadi lebih efisien.

Contoh: lokasi=PCR, id_alat=ESP32_01

Tag umumnya berisi lokasi sensor, ID perangkat. Karena *terindeks*, tag sangat cocok digunakan sebagai parameter filter pada *dashboard* Grafana (*Variables*).

3. *Field Set*

Field merupakan nilai utama hasil pembacaan sensor. *Field* tidak *terindeks* dan biasanya digunakan untuk perhitungan statistik seperti rata-rata, *maksimum*, atau *minimum*.

Contoh: co_val=10.5, so2_val=0.2, no2_val=0.15, o3_val=0.08, pm25_val=35.2, pm10_val=50.4

4. *Timestamp*

Timestamp menunjukkan waktu pencatatan data dalam format *Unix Epoch*. Secara default, InfluxDB menggunakan presisi nanodetik.

Contoh: 1740890000000000000

Timestamp memungkinkan sistem untuk menyusun data berdasarkan waktu, melakukan analisis tren, membuat grafik *time-series*. *Field* dapat berupa *float*, *integer*, *Boolean*, *string*. Dalam sistem *monitoring* kualitas udara, field menyimpan nilai konsentrasi gas seperti CO, SO₂, NO₂, PM2.5, dan PM10.

2.2.6 Grafana

Grafana merupakan platform visualisasi data yang digunakan untuk membuat dashboard interaktif dari berbagai sumber data. Dalam sistem monitoring kualitas udara, Grafana berperan sebagai lapisan visualisasi yang mengubah data mentah menjadi informasi yang mudah dipahami oleh pengguna.

Grafana mampu melakukan kueri langsung ke database seperti InfluxDB dan menyediakan berbagai fitur analisis, seperti agregasi data, pembuatan grafik *time-series*, serta sistem peringatan (*alerting*). Selain itu, Grafana mendukung penggunaan variabel dinamis yang memungkinkan pengguna memilih data berdasarkan parameter tertentu, seperti lokasi sensor.

Dengan fitur seperti gauge panel, *Time-series* graph, dan stat panel, Grafana mampu menyajikan informasi kualitas udara secara real-time maupun historis, sehingga membantu pengguna dalam memahami pola perubahan kualitas udara.

Tabel 2.5 Fitur Utama Grafana pada Sistem Monitoring Kualitas Udara

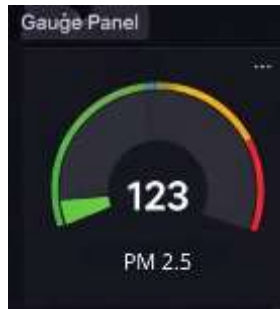
Komponen / Fitur	Fungsi Teknis	Implementasi dalam Sistem
<i>Gauge Panel</i>	Visualisasi radial dengan jarum penunjuk.	Menampilkan konsentrasi gas polutan secara real-time.
<i>Time Series Graph</i>	Grafik garis berbasis deret waktu.	Memetakan tren fluktuasi kualitas udara dalam rentang 24 jam.
<i>Stat Panel</i>	Menampilkan angka nilai tunggal secara dominan.	Menunjukkan angka eksak konsentrasi dengan latar warna ISPU.
<i>Variables</i>	Fitur filter dinamis (drop-down).	Memilih tampilan data berdasarkan lokasi (misal: PCR_Outdoor).
<i>Value Mapping</i>	Transformasi nilai numerik ke teks/warna.	Mengubah angka menjadi label status (misal: "Baik" atau "Bahaya").

Berikut penjelasan pada table 2.5 fitur utama Grafana pada system monitoring kualitas udara:

1. *Gauge Panel*

Fitur ini berfungsi sebagai indikator visual radial yang memetakan nilai numerik terakhir dari InfluxDB ke dalam skala melingkar. Secara teknis, panel ini dikonfigurasi dengan *Thresholds* (ambang batas) untuk memberikan warna otomatis pada area tertentu sesuai standar kualitas udara. Penentuan nilai minimum dan maksimum dilakukan pada bagian *Standard Options* agar jarum penunjuk berada pada skala yang akurat.

Tampilan gauge panel menyerupai *spidometer* kendaraan, di mana terdapat jarum penunjuk atau bar melingkar yang bergerak mengikuti angka konsentrasi gas secara *real-time*. Area skala terbagi menjadi beberapa zona warna (hijau, kuning, merah) yang memberikan informasi tingkat keamanan udara secara instan tanpa perlu membaca tabel data. Berikut gambar 2.2 tampilan gauge panel.



Gambar 2.2 Gauge Panel

2. Time Series Graph

Merupakan grafik berbasis waktu yang menarik data historis dari InfluxDB menggunakan kueri Flux. Secara teknis, panel ini menggunakan fungsi agregasi untuk merata-ratakan ribuan data poin dalam rentang waktu tertentu (misalnya per 10 menit) melalui fungsi *mean()* agar grafik tetap ringan, mudah dibaca, dan tidak membebani kinerja browser.

Tampilan time series graph berupa grafik garis (*line chart*) yang memanjang dari kiri ke kanan. Sumbu X menunjukkan waktu (misalnya rentang 24 jam), sedangkan sumbu Y menunjukkan kadar polutan. Tampilan ini sangat krusial untuk menganalisis tren fluktuasi polusi dan melihat pola kenaikan pada jam-jam tertentu. Berikut gambar 2.3 tampilan time series graph



Gambar 2.3 Time Series Graph

3. Stat Panel

Digunakan untuk menampilkan nilai tunggal (*single stat*) yang diambil dari data terbaru (*last value*) di *database*. Secara teknis, panel ini mendukung fitur *Color Mode* yang memungkinkan seluruh kotak panel berubah warna jika nilai yang diterima melewati ambang batas tertentu, memberikan peringatan visual

yang cepat.

Tampilan stat panel berupa angka eksak yang ditampilkan sangat besar dan mencolok di tengah panel. Biasanya didampingi oleh grafik kecil samar di latar belakang (*sparkline*) yang menunjukkan pergerakan singkat data tersebut dalam beberapa menit terakhir untuk memberikan konteks perubahan nilai. Berikut gambar 2.4 tampilan stat panel.

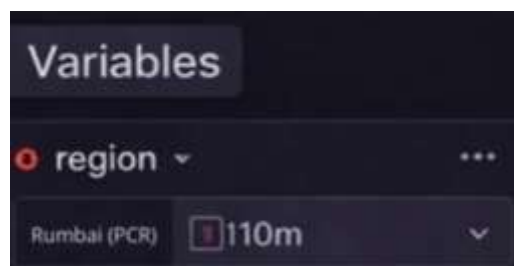


Gambar 2.4 Stat Panel

4. Variables

Fitur ini bertindak sebagai parameter kueri dinamis untuk meningkatkan interaktivitas. Secara teknis, variabel dikonfigurasi menggunakan kueri *tagValues* untuk mengambil daftar nama sensor dari InfluxDB secara otomatis. Variabel ini kemudian disisipkan ke dalam kueri Flux utama menggunakan sintaks *v.variable_name*, sehingga satu *dashboard* bisa digunakan untuk banyak sensor.

Tampilan variables muncul sebagai menu *drop-down* di bagian kiri atas *dashboard*. Pengguna dapat mengklik menu ini untuk memilih lokasi sensor yang berbeda ("PCR_Outdoor"), dan secara otomatis seluruh panel (*Gauge*, *Graph*, *Stat*) akan memperbarui datanya sesuai lokasi terpilih. Berikut gambar 2.5 tampilan *variables*.



Gambar 2.5 Variables

5. Value Mapping

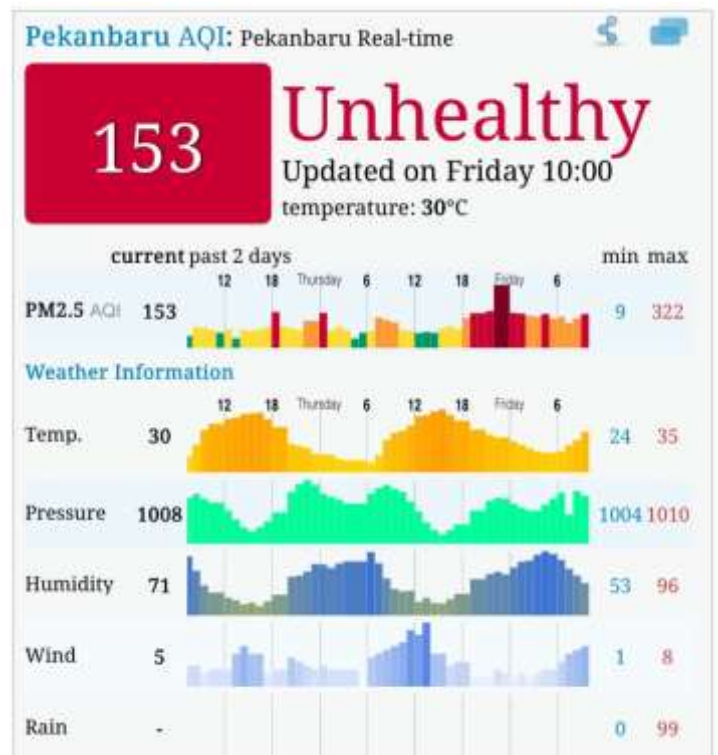
Fitur pengolah logika tampilan yang menerjemahkan angka mentah menjadi bahasa manusia. Secara teknis, sistem melakukan pemetaan rentang nilai (*range mapping*), di mana Grafana mencocokkan data numerik dari InfluxDB dengan label teks dan warna yang telah ditentukan dalam logika program.

Tampilan value mapping di dalam panel (terutama *Stat dan Gauge*), angka-angka teknis yang rumit akan berubah menjadi label teks *informatif*, seperti tulisan "Baik" dengan warna hijau atau "Berbahaya" dengan warna hitam, sehingga memudahkan interpretasi bagi masyarakat awam. Berikut gambar 2.6 tampilan value mapping.



Gambar 2.6 Value Mapping

Dalam proses perancangan dasbor monitoring, visualisasi data yang ditampilkan pada Grafana mengadopsi acuan tata letak (*layout*) dan skema penyajian informasi dari sistem pemantauan kualitas udara global, yakni standar tampilan Indeks Kualitas Udara (*Air Quality Index*) yang digunakan, sebagaimana disajikan pada Gambar 2.7.



Gambar 2. 7 Tampilan Referensi *Air Quality Index* Pekanbaru sebagai Acuan Tata Letak Dasbor (Sumber: aqicn.org)

Penggunaan acuan visual ini bertujuan agar dasbor yang dikembangkan mampu menyajikan data secara lebih informatif, intuitif, dan memiliki standar keterbacaan yang setara dengan sistem monitoring lingkungan profesional. Desain ini mengutamakan kemudahan interpretasi bagi pengguna, di mana posisi indikator status utama dan grafik tren historis disusun dengan mempertimbangkan kenyamanan visual pengguna saat melakukan pemantauan kualitas udara secara kontinu. Contoh rujukan tampilan lengkap dari platform tersebut dapat dilihat pada Lampiran F.

2.2.7 Open Weather

OpenWeather merupakan sebuah layanan antarmuka pemrograman aplikasi atau API (*Application Programming Interface*) daring yang menyediakan data meteorologi dan cuaca global secara *real-time*, prakiraan cuaca, serta data historis cuaca. Layanan ini mengumpulkan dan memproses data yang bersumber dari puluhan ribu stasiun cuaca profesional, radar meteorologi, dan satelit di seluruh dunia yang diproses menggunakan model *machine learning* untuk menghasilkan

informasi yang akurat. Dalam pengembangan sistem *Internet of Things* (IoT), layanan seperti OpenWeather sering dimanfaatkan sebagai sumber data eksternal (sekunder) untuk melengkapi dan memvalidasi pembacaan dari sensor fisik yang ada pada perangkat.

Secara teknis, pertukaran data dengan layanan OpenWeather beroperasi menggunakan arsitektur *RESTful API* melalui protokol HTTP (*Hypertext Transfer Protocol*). Perangkat klien atau mikrokontroler akan melakukan permintaan data (*request*) ke server OpenWeather dengan menyertakan kredensial berupa API Key beserta parameter lokasi geografis, seperti nama kota atau titik koordinat (lintang dan bujur). Sebagai respons, server akan mengembalikan paket data yang umumnya berformat JSON (*JavaScript Object Notation*). Format JSON sering digunakan dalam arsitektur IoT karena strukturnya yang ringan, mudah dibaca, serta membutuhkan sumber daya komputasi yang minimal untuk proses penguraian (*parsing*) data.

Pada konteks pemantauan kualitas udara, integrasi data dari API cuaca memiliki urgensi teoretis yang tinggi. Parameter meteorologi seperti suhu lingkungan, tingkat kelembapan relatif, tekanan udara, serta arah dan kecepatan angin memiliki korelasi langsung terhadap dinamika polusi udara. Faktor-faktor cuaca tersebut sangat memengaruhi proses dispersi (penyebaran), dilusi (pengenceran), dan akumulasi polutan maupun partikulat di atmosfer suatu wilayah. Oleh karena itu, penggabungan data cuaca berbasis API dengan metrik kualitas udara memungkinkan sistem untuk menyajikan dataset yang lebih komprehensif guna keperluan pemantauan dan analisis tingkat lanjut.

2.2.8 *Telegraf (Penghubung MQTT broker dan InfluxDB)*

Telegraf merupakan agen pengumpul data (*data collection agent*) berbasis *plugin* yang dikembangkan oleh InfluxData dan berfungsi untuk mengumpulkan, memproses, serta mengirimkan data ke berbagai sistem penyimpanan, khususnya InfluxDB. Dalam arsitektur sistem IOT, Telegraf berperan sebagai penghubung (*bridge*) antara MQTT *broker* dan InfluxDB, sehingga memungkinkan integrasi data dilakukan secara otomatis tanpa memerlukan pengembangan aplikasi tambahan.

Dalam sistem monitoring kualitas udara ini, Telegraf bertindak sebagai

MQTT *subscriber* yang berlangganan pada topik tertentu di MQTT *broker*. Data sensor yang dikirimkan oleh perangkat IoT melalui mekanisme publish akan diterima oleh Telegraf, kemudian diproses dan diteruskan ke InfluxDB dalam bentuk *data Time-Series*. Pendekatan ini lebih efisien dibandingkan metode *manual* seperti pengiriman data menggunakan *script* atau API karena bersifat otomatis dan berkelanjutan.

Secara umum, alur kerja integrasi data menggunakan Telegraf dapat dijelaskan sebagai berikut:

- 1 Open Weather mengirimkan data sensor ke MQTT *broker*.
- 2 Telegraf melakukan *subscribe* pada topik MQTT yang telah ditentukan.
- 3 Data yang diterima diproses menggunakan *plugin input* MQTT.
- 4 Data kemudian dikirim ke InfluxDB menggunakan *plugin output*.
- 5 Data yang tersimpan divisualisasikan menggunakan Grafana.

Telegraf menggunakan sistem plugin yang fleksibel, sehingga mendukung berbagai sumber data dan format data, seperti JSON, CSV, maupun line *protocol*. *Plugin inputs.mqtt consumer* memungkinkan Telegraf untuk membaca data dari MQTT *broker* secara langsung, sedangkan *plugin outputs.influxdb_v2* digunakan untuk mengirimkan data ke InfluxDB dengan konfigurasi tertentu, seperti URL server, token autentikasi, organisasi, dan bucket penyimpanan.

Selain itu, Telegraf juga menyediakan fitur pemrosesan data (*data processing*) sebelum data dikirim ke database, seperti filtering, parsing, serta agregasi sederhana. Fitur ini memungkinkan sistem untuk mengurangi beban penyimpanan serta meningkatkan efisiensi pengolahan data (InfluxData, 2023).

Konfigurasi Telegraf dilakukan melalui file konfigurasi *telegraf.conf*, yang berisi parameter koneksi ke MQTT *broker* dan InfluxDB. Pengaturan ini mencakup alamat server MQTT, topik yang digunakan, serta informasi autentikasi dan *endpoint* dari InfluxDB. Dengan konfigurasi yang tepat, Telegraf mampu bekerja secara otomatis sebagai *pipeline* data yang stabil dan berkelanjutan.

Dibandingkan dengan metode integrasi manual menggunakan API atau *script* khusus, penggunaan Telegraf memberikan keuntungan dalam hal kemudahan implementasi, skalabilitas, serta efisiensi sistem. Hal ini sangat penting dalam sistem *monitoring* kualitas udara yang menghasilkan data secara

kontinu dan dalam jumlah besar.

Dengan demikian, Telegraf menjadi komponen penting dalam arsitektur sistem IoT yang berfungsi untuk memastikan aliran data dari MQTT *broker* ke InfluxDB berjalan secara *real-time*, terstruktur, dan andal. Keberadaan Telegraf juga memungkinkan sistem untuk dikembangkan lebih lanjut dengan menambahkan berbagai sumber data lain tanpa perubahan signifikan pada arsitektur utama.