

SISTEM PEMANTAUAN KUALITAS UDARA BERBASIS INTERNET OF THINGS (IoT) DENGAN VISUALISASI DATA PADA APLIKASI ANDROID (STUDI KASUS: PT. SOUTH PACIFIC VISCOSE)

Akbar Nugraha Dimiyati^{1*}, Purwantoro.², Aji Primajaya³

^{1,2,3}Universitas Singaperbangsa Karawang; Jl. HS. Ronggowaluyo Telukjambe Timur; (0267) 641177

Keywords:

Internet of Things, Kualitas Udara, *PMS5003*, Android, *MQ135*.

Correspondent Email:

akbarnugrahad02@gmail.com



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstrak. Pencemaran udara merupakan permasalahan lingkungan yang berdampak signifikan terhadap kesehatan manusia, khususnya dalam meningkatkan risiko penyakit pernapasan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem pemantauan kualitas udara berbasis Internet of Things (IoT) yang mampu memberikan data secara real-time dan divisualisasikan dalam bentuk aplikasi Android. Sistem ini menggunakan sensor *MQ135* dan *PMS5003* yang terhubung dengan modul *ESP32* untuk mendeteksi parameter kualitas udara seperti *PM2.5*, *PM10*, *NH₃*, *CO₂*, dan *NO_x*. Data dikirimkan melalui koneksi WiFi ke server backend yang dibangun menggunakan *Express.js* dan *REST API*, kemudian ditampilkan di aplikasi Android yang dikembangkan dengan *Retrofit*, dan *Room Database*. Studi kasus dilakukan di PT. South Pacific Viscose untuk menguji fungsionalitas sistem. Hasil pengujian menunjukkan bahwa sistem dapat bekerja secara efektif dalam memantau dan menampilkan data kualitas udara secara akurat dan real-time.

Abstract. Air pollution is an environmental issue that significantly impacts human health, particularly by increasing the risk of respiratory diseases. This study aims to design and implement an Internet of Things (IoT)-based air quality monitoring system capable of providing real-time data visualized through an Android application. The system utilizes *MQ135* and *PMS5003* sensors connected to an *ESP32* module to detect air quality parameters such as *PM2.5*, *PM10*, *NH₃*, *CO₂*, and *NO_x*. Data is transmitted via WiFi to a backend server built with *Express.js* and *REST API*, then displayed in an Android application developed using *Retrofit*, and *Room Database*. A case study at PT. South Pacific Viscose was conducted to evaluate the system's functionality. The testing results indicate that the system effectively monitors and displays air quality data accurately and in real time.

1. PENDAHULUAN

Kualitas udara merupakan salah satu aspek lingkungan yang sangat penting bagi kesehatan manusia dan keberlanjutan ekosistem. Menurut Damayanti & Handriyono kualitas udara adalah

keadaan udara yang mengandung gas dan partikular yang berkaitan dengan kesehatan manusia[1]. Udara yang bersih dan sehat adalah prasyarat bagi kehidupan yang baik, namun banyak negara, termasuk Indonesia,

menghadapi tantangan serius terkait polusi udara. Sumber polusi udara meliputi emisi kendaraan bermotor, industri, dan pembakaran bahan bakar fosil yang menghasilkan zat-zat berbahaya seperti partikel debu (PM2.5 dan PM10), gas karbon dioksida (CO₂), karbon monoksida (CO), nitrogen oksida (NO_x), serta amonia (NH₃)[2]. Kualitas udara yang buruk telah terbukti berdampak negatif pada kesehatan manusia, terutama bagi mereka yang memiliki masalah pernapasan, serta memperburuk fenomena pemanasan global dan perubahan iklim.

Masalah utama yang perlu diselesaikan adalah bagaimana memantau kualitas udara secara efektif di lokasi-lokasi tertentu, seperti area industri, agar dapat mendeteksi polutan berbahaya secara real-time. Tanpa makanan dan air, manusia mampu bertahan hidup sekitar 5 hari. Namun tanpa udara, manusia hanya dapat bertahan hidup selama sekitar 5 menit[3]. Saat ini, pemantauan kualitas udara di banyak lokasi masih belum terintegrasi dengan baik, dan kurangnya informasi yang cepat serta akurat membuat masyarakat, khususnya karyawan di lingkungan industri, tidak selalu sadar akan bahaya yang ada. Hal ini sangat penting di area industri seperti PT. South Pacific Viscose, di mana PT. South Pacific Viscose (SPV) merupakan perusahaan yang memproduksi serat viscose dengan mengolah bahan baku kayu melalui proses kimia yang kompleks, yang berpotensi menghasilkan polusi udara. Sayangnya, hingga kini belum ada penelitian yang memadai terkait kualitas udara di sekitar pabrik, khususnya terhadap PM2.5, PM10, NO_x, CO₂, dan NH₃ yang berisiko bagi kesehatan. Sistem pemantauan di PT. South Pacific Viscose pun masih terbatas dan belum mendukung akses real-time, sehingga kurang optimal untuk pemantauan berkelanjutan. Selain itu, potensi kebocoran gas akibat kurangnya perawatan atau insiden operasional dapat membahayakan pekerja dan mencemari lingkungan.

Beberapa penelitian sebelumnya telah mengembangkan sistem pemantauan kualitas udara berbasis IoT. Sadi menggunakan Thingsboard untuk monitoring CO, PM2.5, dan NO₂ secara real-time, namun tanpa penyimpanan data dan aplikasi mobile[4]. Hasanuddin & Herdianto menggunakan Thingspeak, tetapi sistemnya terbatas dan

belum ada notifikasi[5]. Yushananta pada 2023 mengembangkan sistem berbiaya rendah, namun minim fitur lanjutan seperti visualisasi dan penyimpanan data[6]. Budianto & Sumanto merancang sistem real-time dengan DBMS, namun belum menyediakan visualisasi informatif dan aplikasi mobile[7]. Muttaqin menggunakan Blynk dan sensor DHT11 serta MQ135, tetapi terbatas dalam parameter dan belum terintegrasi dalam aplikasi Android[8]. Dari penelitian-penelitian ini, terlihat perlunya sistem yang lebih lengkap, mencakup lebih banyak parameter, penyimpanan historis, visualisasi, dan integrasi mobile.

Penelitian ini bertujuan untuk merancang sistem pemantauan kualitas udara berbasis Internet of Things (IoT) yang menggunakan sensor MQ-135 dan PMS5003 untuk mendeteksi keberadaan gas berbahaya seperti NO_x, CO₂, dan NH₃, serta partikel PM2.5 dan PM10 di lingkungan PT. South Pacific Viscose. Sistem ini akan dilengkapi dengan fitur pemantauan data secara real-time melalui integrasi dengan aplikasi Android, sehingga informasi kualitas udara dapat diakses secara langsung dan mudah oleh pengguna. Dengan adanya sistem ini, diharapkan pemantauan kondisi udara di area industri dapat dilakukan secara lebih menyeluruh dan berkelanjutan, serta membantu karyawan, khususnya yang memiliki gangguan pernapasan, dalam menghindari area dengan tingkat polusi udara yang tinggi.

2. TINJAUAN PUSTAKA

Internet of Things (IoT) adalah konsep di mana perangkat fisik dan objek terhubung melalui internet untuk berkomunikasi dan bertukar data secara otomatis. Terdiri dari tiga elemen utama: objek dengan modul IoT, perangkat penghubung (modem dan router), dan pusat data berbasis cloud. IoT memungkinkan perangkat seperti peralatan rumah tangga dan kendaraan beroperasi lebih cerdas, meningkatkan efisiensi dan otomatisasi di berbagai sektor, termasuk kesehatan, transportasi, pertanian, dan manufaktur [9].

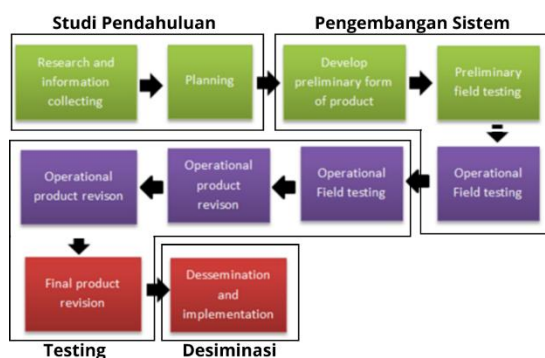
Menurut Haryoga, konsep dasar dari IoT adalah tentang menghubungkan sensor-sensor cerdas yang terintegrasi dengan internet yang dirancang agar dapat mengendalikan sistem yang dibangun[10]. Cara kerja Internet of Things (IoT) sangat mudah hanya dengan

menyambungkan sensor-sensor yang dibutuhkan perangkat sehari-hari kedalam cloud, lalu perangkat tersebut akan mengontrol sensor secara real-time dari jarak jauh.

Dashboard adalah alat visual yang menyajikan data penting, memungkinkan pengguna dengan cepat memantau metrik dan kinerja sistem, memberikan gambaran yang jelas untuk analisis tren dan pengambilan keputusan berbasis data, di mana keberhasilannya bergantung pada kemampuan visual dalam menyampaikan informasi secara akurat, serta desain antarmuka yang memperhatikan estetika, ergonomi, dan efektivitas demi kemudahan pengguna[11]. Dalam penelitian ini, dashboard berfungsi untuk menyajikan data hasil pemantauan dari setiap sensor (PM2.5, PM10, serta gas berbahaya seperti CO₂, CO, NH₃, dan NO_x) secara real-time. Tujuan dari penggunaan dashboard ini adalah agar para pegawai dapat mengakses informasi pemantauan dan membuat keputusan yang tepat untuk menghindari paparan terhadap partikel dan gas berbahaya.

3. METODE PENELITIAN

Penelitian dan pengembangan, atau research and development (R&D) dalam Bahasa Inggris, adalah metode penelitian yang kini banyak digunakan di dunia akademik untuk merancang dan menguji efektivitas suatu produk. Metode ini bertujuan menghasilkan produk dengan melalui proses identifikasi potensi masalah, perancangan, dan pengembangan sebagai solusi yang paling optimal [12].



Gambar 1. Tahapan R&D Borg & Gall

Menurut Brog dan Gall (1993: 772), Research and Development (R&D) adalah proses pengembangan dan validasi produk berbasis penelitian. Penelitian ini mengadaptasi

model R&D Borg & Gall yang disederhanakan menjadi 4 tahap: studi pendahuluan, pengembangan sistem (identifikasi kebutuhan, desain sistem, pembuatan alat), testing, dan desiminasi. Tahap testing bersifat iteratif untuk memastikan keakuratan sistem sebelum implementasi.

3.1. Studi Pendahuluan

Tahap awal sebelum memulai penelitian adalah melakukan analisis terhadap data atau informasi yang berasal dari penelitian sebelumnya yang relevan dengan topik yang akan dikaji. Proses ini dikenal sebagai studi pendahuluan. Tahapan ini bertujuan untuk memahami konteks penelitian, mengidentifikasi kesenjangan yang ada, serta menentukan arah dan fokus penelitian agar lebih terarah dan berbasis pada landasan yang kuat.

3.2. Pengembangan Sistem

Tahapan pengembangan sistem meliputi identifikasi kebutuhan, perancangan perangkat, dan pembuatan perangkat.

3.2.1. Identifikasi Kebutuhan

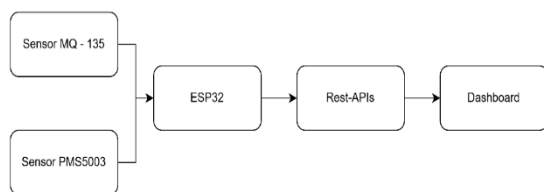
Tahap awal dalam pengembangan sistem adalah identifikasi kebutuhan, yang bertujuan menentukan perangkat keras dan perangkat lunak yang diperlukan untuk membangun sistem pemantauan kualitas udara di area parkir. Perangkat keras terdiri dari tiga bagian utama: input, proses, dan output. Sensor PMS5003 dan MQ-135 digunakan sebagai input untuk mendeteksi parameter kualitas udara, sementara mikrokontroler ESP-32 memproses data yang dihasilkan sensor, dan dashboard digunakan untuk menampilkan hasil pemantauan sebagai output. Perangkat lunak yang digunakan mencakup Android Studio, Arduino IDE, Fritzing, dan Draw.io untuk mendukung pengembangan program, desain rangkaian, dan visualisasi sistem. Identifikasi ini memungkinkan peneliti memilih perangkat yang tepat dan memastikan komponen yang digunakan dapat mendukung penelitian secara optimal.

3.2.2. Desain Sistem

Tahapan berikutnya adalah perancangan sistem, yang terdiri dari dua bagian utama, yaitu desain perangkat lunak dan desain perangkat keras.

Pada desain perangkat lunak, diagram blok akan dibuat menggunakan perangkat lunak draw.io, sedangkan desain perangkat keras untuk alat pendeteksi kualitas udara akan dirancang menggunakan perangkat lunak Fritzing.

Tahap ini bertujuan untuk menjadi pedoman dalam proses perancangan alat yang akan dikembangkan. Rancangan sistem yang akan digunakan dalam penelitian ini dapat dilihat pada Gambar 2 di bawah ini.

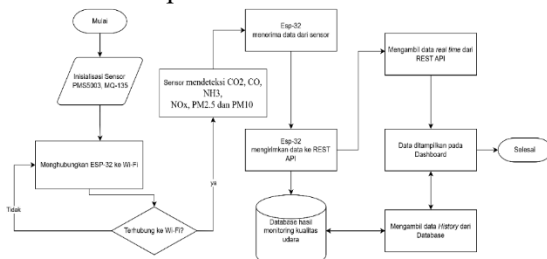


Gambar 2. Diagram Blok Sistem.

Pada Gambar 2. sensor yang digunakan akan mengirimkan data dalam bentuk analog, yang kemudian dikonversi menjadi data digital oleh mikrokontroler ESP32. Setelah itu, data tersebut diproses di dalam mikrokontroler untuk menghasilkan output yang akan ditampilkan melalui dashboard.

3.2.3. Pembuatan Alat

Berikut ini merupakan penjelasan mengenai alur pembuatan perangkat dan sistem yang diilustrasikan pada Gambar 3. di bawah ini.



Gambar 3. Flowchart Sistem.

Gambar 3. menunjukkan ketika sistem mulai dijalankan, sensor PMS5003 dan MQ-135 akan diinisialisasi untuk mendeteksi parameter kualitas udara seperti CO₂, CO, NH₃, NO_x, PM2.5, dan PM10. Selanjutnya, ESP32 akan menerima data dari sensor tersebut dan menghubungkannya ke jaringan Wi-Fi. Jika koneksi Wi-Fi berhasil, ESP32 akan mengirimkan data ke REST API menggunakan protokol HTTP. Data tersebut akan disimpan dalam database hasil monitoring kualitas udara.

Kemudian, REST API memungkinkan pengambilan data secara real-time dan juga data historis dari database. Data yang telah diambil akan ditampilkan pada dashboard yang dapat diakses oleh pengguna melalui perangkat android. Jika koneksi ke database gagal, sistem akan mencoba mengirim ulang data hingga proses berhasil.

4. HASIL DAN PEMBAHASAN

Penelitian sistem pemantauan kualitas udara berbasis internet of things (iot) dengan visualisasi data pada aplikasi android dengan objek penelitian kualitas udara pada area parkir PT. South Pacific Viscose, berdasarkan tahapan penelitian, proses yang meliputi analisis kebutuhan, perancangan sistem, pemrograman, implementasi, dan uji coba merupakan bagian dari tahapan pengembangan sistem. Tahapan ini bertujuan untuk memastikan sistem yang dibangun dapat berjalan sesuai dengan tujuan dan kebutuhan pengguna, mulai dari identifikasi masalah hingga pengujian sistem agar siap digunakan secara optimal.

4.1. Identifikasi Kebutuhan

Pada lingkungan PT. South Pacific Viscose, pemantauan kualitas udara masih dilakukan secara manual dan kurang terintegrasi dengan baik. Hal ini menyebabkan kesulitan dalam mendeteksi polusi udara secara real-time, serta kurangnya informasi yang cepat dan akurat tentang kualitas udara yang dapat membahayakan kesehatan pekerja dan lingkungan sekitar. Polusi udara yang terjadi di area industri, seperti debu dan gas berbahaya, dapat berdampak negatif bagi kesehatan pekerja, terutama mereka yang memiliki masalah pernapasan.

Untuk meningkatkan efektivitas pemantauan kualitas udara di area industri PT. South Pacific Viscose, diperlukan sistem pemantauan berbasis Internet of Things (IoT). Sistem ini menggunakan berbagai sensor, seperti MQ-135 untuk mendeteksi gas berbahaya dan PMS5003 untuk mendeteksi partikel debu, yang kemudian terhubung dengan aplikasi Android untuk memvisualisasikan data kualitas udara secara real-time. Sistem ini memungkinkan pemantauan kualitas udara yang lebih akurat dan cepat, sehingga pekerja dapat menghindari area dengan kualitas udara yang buruk demi

kesehatan mereka. Adapun perangkat yang digunakan dalam penelitian ini dijelaskan lebih lanjut pada Tabel 1.

Tabel 1. Alat yang dibutuhkan.

No	Komponen	Fungsi
1.	Modul ESP-32	Modul ESP-32 berfungsi sebagai otak dari sistem yang menghubungkan perangkat IoT dengan internet[13]. Modul ini memiliki kemampuan Wi-Fi dan Bluetooth, yang memungkinkan pengiriman data secara nirkabel ke server atau aplikasi Android. Selain itu, ESP-32 juga mengontrol proses pengambilan data dari sensor dan mengirimkan data tersebut secara real-time untuk dianalisis dan ditampilkan pada dashboard aplikasi.
2.	MQ-135	MQ-135 berfungsi untuk mendeteksi polutan gas berbahaya di udara, seperti karbon dioksida (CO ₂), karbon monoksida (CO), amonia (NH ₃), dan nitrogen oksida (NO _x)[14]. Sensor ini bekerja dengan mendeteksi perubahan resistansi pada sensor ketika gas tertentu ada di sekitar, lalu mengubahnya menjadi sinyal analog yang dapat diolah oleh mikrokontroler untuk memantau kualitas udara di lingkungan sekitar.
3.	PMS5003	PMS5003 berfungsi untuk mendeteksi keberadaan partikel debu seperti PM _{2.5} dan PM ₁₀ di udara. Sensor ini menggunakan metode optik untuk menghitung jumlah partikel debu dalam udara, dan hasil pengukurannya memberikan informasi yang sangat penting terkait kualitas udara[15]. Data yang diperoleh dari PMS5003 memungkinkan sistem untuk memberikan informasi mengenai tingkat polusi debu yang dapat

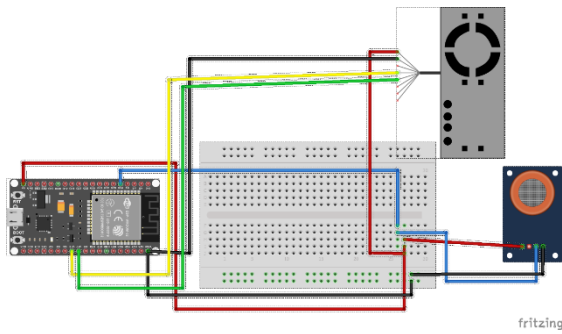
		mempengaruhi kesehatan manusia.
4.	Kabel Jumper	Kabel jumper berfungsi untuk menghubungkan berbagai komponen dalam rangkaian elektronik, baik pada breadboard maupun pada papan sirkuit tercetak (PCB)[16]. Kabel ini digunakan untuk mentransmisikan sinyal dan daya antar komponen tanpa memerlukan penyolderan, yang memudahkan dalam pengujian dan pengembangan sistem.
5.	Breadboard	Breadboard berfungsi sebagai tempat untuk merakit dan menguji rangkaian elektronik sementara, tanpa memerlukan penyolderan komponen[16]. Ini memungkinkan pengujian berbagai konfigurasi sirkuit secara mudah dan cepat. Komponen-komponen seperti sensor, ESP-32, dan kabel jumper dapat ditempatkan di breadboard untuk memastikan semuanya terhubung dengan benar sebelum dipindahkan ke sirkuit tetap.

4.2. Desain Sistem

Proses perancangan sistem dalam penelitian ini dilakukan melalui beberapa tahap, yaitu perancangan sistem IoT dan aplikasi Android.

4.2.1. Desain Sistem Perangkat IoT

Dalam Pembuatan sistem pemantauan kualitas udara membutuhkan komponen untuk mendukung pembuatan sistem pemantauan kualitas udara ini. Alat yang di butuhkan adalah Esp-32, sensor MQ-135, sensor PMS5003, Kabel Jumper, dan Breadboard. Pada tahap perancangan perangkat keras, penting untuk membuat skema yang menggambarkan bagaimana perangkat keras saling terhubung. Di tahap ini, dibuat Wiring diagram alat dan flowchart untuk menggambarkan alur kerja dan koneksi antara komponen. Berikut adalah proses pembuatan Wiring Schematics perangkat IoT:



Gambar 4. Wiring Schematics.

Setelah melakukan tahap wiring pada Gambar 4.1, dan memastikan tidak ada yang salah dalam melakukan wiring agar mencegah terjadinya kegagalan dalam tahap pemrograman. Jalur pin perangkat keras, untuk pin MQ-135 terdapat pada tabel 2, dan untuk pin PMS5003 terdapat pada tabel 3.

Tabel 2. Pin MQ-135

MQ-135	ESP32
VCC	3.3V
GND	GND
AO	34

Tabel 3. Pin PMS5003

PMS5003	ESP32
VCC	5V
GND	GND
RX	16
TX	17

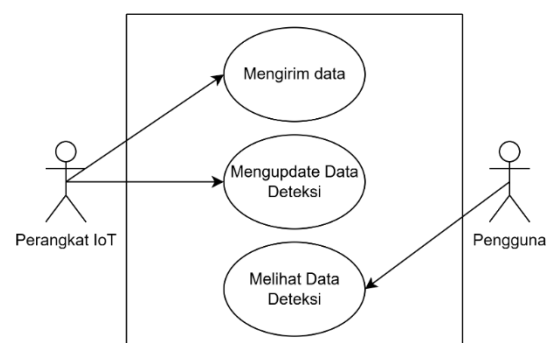
Setelah memastikan semua pin telah terhubung dengan benar antara satu modul dengan modul lainnya, langkah selanjutnya adalah menginstal library yang diperlukan untuk memprogram perangkat keras menggunakan Arduino IDE.

4.2.2. Desain Sistem Aplikasi Android

Langkah awal dalam pengembangan aplikasi ini adalah membangun REST API sebagai perantara untuk menerima dan mengirim data dari ESP32. Data yang diterima akan disimpan dalam Room Database di Android dengan satu tabel untuk efisiensi. REST API memastikan komunikasi stabil antara IoT dan aplikasi mobile, memungkinkan sinkronisasi real-time. Sistem ini dirancang untuk memantau kualitas udara secara akurat, meningkatkan efisiensi pencatatan, serta mengurangi risiko

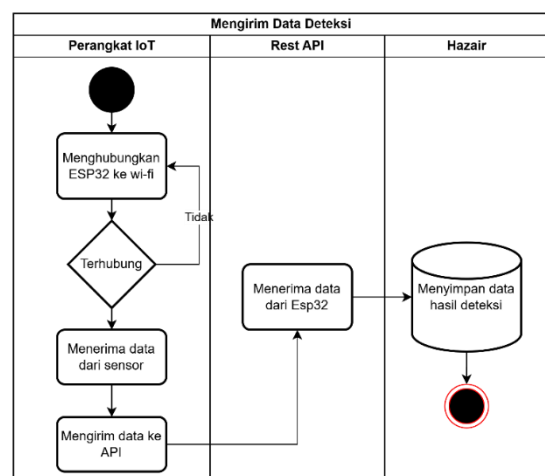
kehilangan data akibat keterbatasan konektivitas atau kesalahan perangkat.

Berikut desain sistem dan pembuatan database yang dilengkapi dengan UML (Unified Modeling Language) untuk mempermudah visualisasi struktur dan alur data dalam sistem[17]. UML digunakan agar pengembangan aplikasi lebih terorganisir, memudahkan komunikasi antar tim, dan memberikan gambaran jelas tentang komponen serta hubungan antar bagian dalam sistem. Berikut design sistem dan pembuatan database :



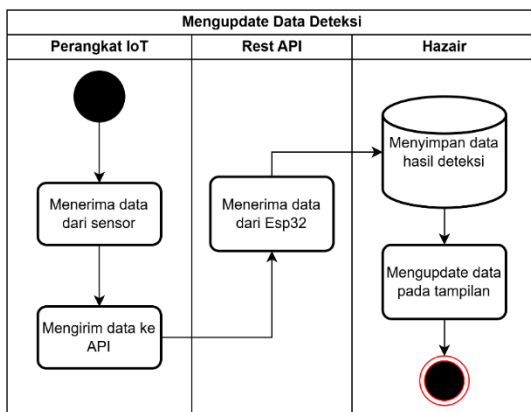
Gambar 5. Usecase diagram sistem.

Pada Gambar 5. merupakan usecase diagram, tugas perangkat iot adalah mendeteksi kualitas udara dan mengrrimkan data. Sedangkan actor pengguna hanya bisa melihat data yang telah divisualisasikan. Selanjutnya membuat activity diagram, yang berfungsi untuk menggambarkan aliran kerja atau aktivitas dalam sistem.



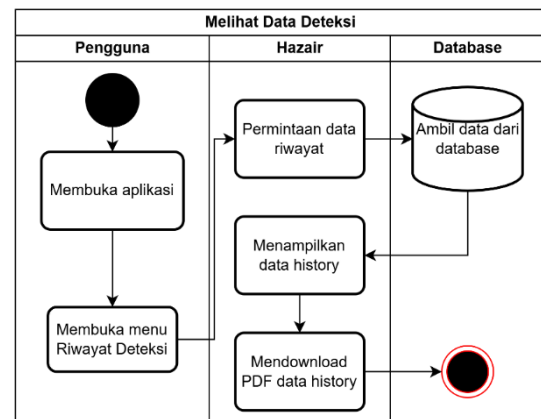
Gambar 6. Activity diagram mengirim data deteksi.

Gambar 6. menunjukkan activity diagram proses pengiriman data deteksi kualitas udara. Proses dimulai dari perangkat ESP32 yang terhubung ke jaringan Wi-Fi. Jika koneksi berhasil, perangkat akan menerima data dari sensor MQ135 dan PMS5003, lalu mengirimkannya ke REST API. Selanjutnya, REST API menerima data tersebut dan menyimpannya ke dalam database HazAir. Jika koneksi Wi-Fi gagal, sistem akan mencoba kembali hingga berhasil.



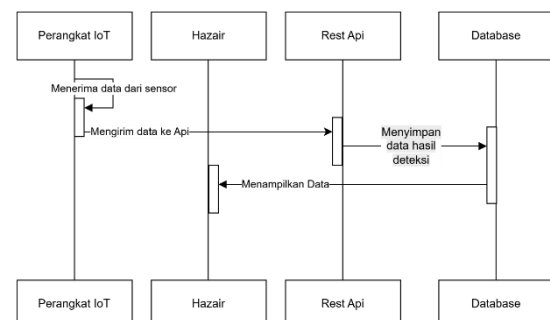
Gambar 7. Activity diagram mengupdate data deteksi.

Activity diagram pada Gambar 7. menggambarkan proses pembaruan data deteksi kualitas udara dalam sistem HazAir. Proses dimulai dari perangkat ESP32 yang kembali menerima data dari sensor. Data kemudian dikirimkan ke REST API, diteruskan ke database untuk disimpan, dan hasil pembaruan ditampilkan pada antarmuka pengguna di aplikasi HazAir. Alur ini memastikan bahwa informasi yang ditampilkan di aplikasi selalu diperbarui secara real-time berdasarkan data terbaru dari perangkat.



Gambar 8. Activity diagram melihat data deteksi.

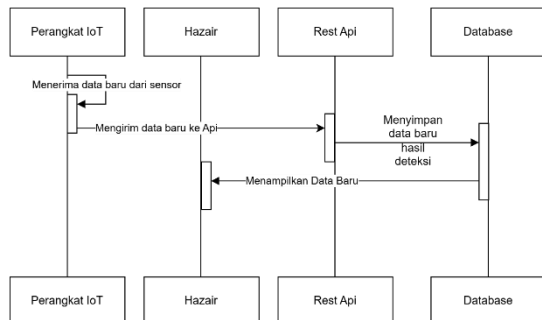
Gambar 8. menunjukkan alur aktivitas ketika pengguna melihat data riwayat deteksi kualitas udara melalui aplikasi HazAir. Proses dimulai saat pengguna membuka aplikasi dan memilih menu "Riwayat Deteksi". Sistem kemudian mengirim permintaan ke database untuk mengambil data historis. Setelah data diperoleh, HazAir menampilkannya dalam bentuk riwayat deteksi. Pengguna juga dapat mengunduh data riwayat tersebut dalam format PDF sebagai dokumentasi. Setelah menggambarkan alur aktivitas melalui activity diagram, bagian selanjutnya akan memperlihatkan detail interaksi antar komponen melalui sequence diagram. Sequence diagram digunakan untuk menunjukkan urutan komunikasi dan pesan yang terjadi antara objek dalam skenario tertentu secara kronologis.



Gambar 9. Sequence diagram mengirim data deteksi.

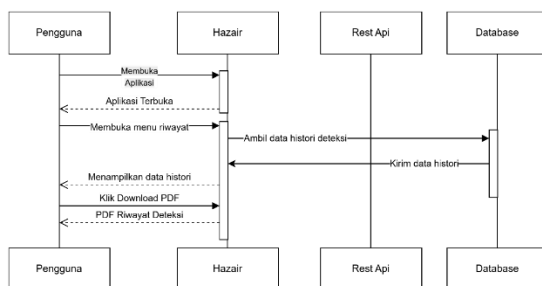
Diagram ini menggambarkan interaksi detail saat perangkat ESP32 mulai menginisialisasi sensor dan mengecek koneksi jaringan. Setelah koneksi berhasil, data sensor dikirim ke REST API yang kemudian menyimpannya ke dalam database.

Sequence diagram ini memperjelas urutan teknis komunikasi dan pengiriman data antar komponen.



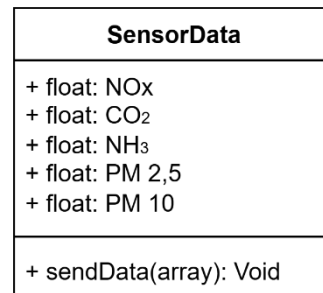
Gambar 4.7 Sequence diagram mengupdate data deteksi.

Pada gambar 4.7 diagram ini ditunjukkan bagaimana sistem secara berkala menerima data baru dari sensor, mengirimkannya ke API, lalu melakukan pembaruan ke database. Setelah data tersimpan, aplikasi HazAir akan mengambil data terbaru dan memperbarui tampilan kepada pengguna. Ini memperkuat pemahaman alur pembaruan secara real-time dari sisi interaksi sistem.



Gambar 4.8 Sequence diagram melihat data deteksi.

Diagram ini menjelaskan urutan langkah saat pengguna mengakses riwayat data deteksi. Mulai dari pengguna membuka aplikasi, sistem mengirim permintaan data, database merespons dengan mengirim data yang relevan, hingga data tersebut ditampilkan ke pengguna. Termasuk juga proses pengunduhan data dalam bentuk PDF sebagai output tambahan dari sistem. Selanjutnya adalah merancang apa saja database yang akan digunakan, berikut rancangan database pada Gambar 4.9.



Gambar 4.9 Class diagram.

Rancangan struktur database pada sistem HazAir ditunjukkan pada Gambar 4.9 dalam bentuk class diagram. Class diagram tersebut merepresentasikan entitas SensorData yang berfungsi menyimpan data hasil pembacaan sensor kualitas udara dari perangkat IoT. Perancangan class ini menjadi dasar dalam pembuatan struktur tabel database serta digunakan untuk memproses dan menampilkan data pada sistem HazAir, baik di sisi backend maupun aplikasi Android.

Gambar 4.10 Room database.

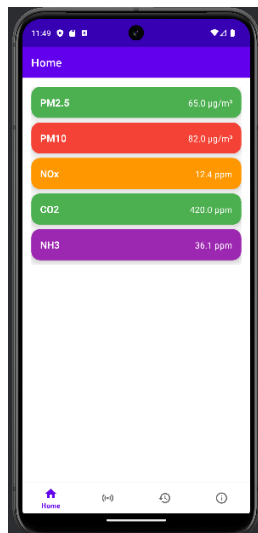
Pada Gambar 4.10, database yang dibuat bernama database sensor_data_database, yang didalamnya terdapat 1 tabel, yaitu : tabel data_sensor. Didalam tabel tersebut terdapat beberapa kolom, berikut kolom kolom pada masing masing tabel :

Tabel 4. Tabel data sensor.

N o	Field	Type	Siz e	Ket
1	id	Varchar	100	Primar y key
2	pm25	Double	5	-
3	pm10	Double	5	-
4	nox	Double	5	-
5	nh3	Double	5	-
6	Timesta mp	Timesta mp	-	-

Tabel 4. data_sensor berisikan kolom id (Varchar), pm25 (Double), pm10 (Double), nox (Double), nh3 (Double), co2 (Double), dan Timestamp (Date). Masing-masing kolom

memiliki ukuran (size) tertentu. Untuk kolom id mempunyai keterangan Primary Key, yaitu kolom atau kombinasi kolom dalam tabel database yang mempunyai kemampuan untuk mengidentifikasi setiap baris dalam tabel tersebut secara unik.



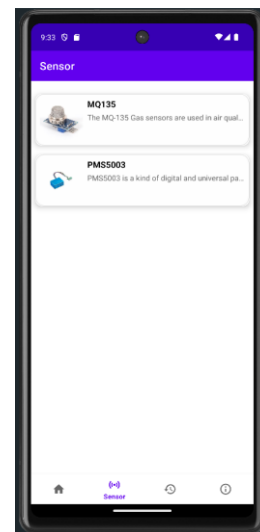
Gambar 4.11 Tampilan dashboard android.

Gambar 4.11 menampilkan dashboard Android yang memonitor kualitas udara dengan parameter PM2.5 ($65.0 \mu\text{g}/\text{m}^3$), PM10 ($82.0 \mu\text{g}/\text{m}^3$), NOx (12.4 ppm), CO2 (420.0 ppm), dan NH3 (36.1 ppm). Warna pada antarmuka—seperti hijau (aman), kuning (sedang), oranye/merah (berbahaya)—menggambarkan tingkat polusi berdasarkan standar AQI, membantu pengguna menilai risiko dan mengambil tindakan sesuai kebutuhan.



Gambar 4.12 Tampilan riwayat deteksi.

Gambar 4.12 menampilkan halaman riwayat deteksi yang menyimpan data pengukuran kualitas udara dari sensor, termasuk parameter seperti PM2.5, PM10, NOx, CO2, dan NH3 beserta waktu pencatatannya. Fitur ini memungkinkan pengguna memantau tren polusi dan menganalisis perubahan kualitas udara secara periodik untuk evaluasi lebih lanjut. Data disajikan dalam format tabel atau grafik yang mudah dipahami.



Gambar 4.13 Tampilan sensor yang digunakan.

Gambar 4.13 menampilkan antarmuka informasi sensor yang digunakan dalam proyek ini, dengan tampilan minimalis namun informatif. Ketika pengguna mengeklik salah satu sensor, sistem akan menampilkan tiga komponen utama: gambar ilustrasi sensor untuk identifikasi visual, nama sensor serta deskripsi singkat tentang fungsi dan aplikasinya dalam pemantauan kualitas udara.

4.3. Pembuatan Alat

Implementasi perangkat dilaksanakan dengan mengintegrasikan komponen perangkat keras dan perangkat lunak sesuai dengan desain sistem yang telah dirancang sebelumnya. Pada tahap ini, perangkat diaktifkan dan konfigurasi sistem dilakukan berdasarkan spesifikasi teknis yang telah ditetapkan. Selanjutnya, dilakukan pengujian awal guna memverifikasi stabilitas dan keandalan operasional sistem.



Gambar 4.15 Perakitan Alat.

Gambar 4.14 menunjukkan semua perangkat yang telah dirakit sesuai dengan desain yang telah ditentukan, dan telah dipastikan bahwa tidak ada kesalahan dalam pemasangan pin. Setelah memastikan bahwa semua komponen terpasang dengan benar, langkah selanjutnya adalah menghubungkan perangkat ke internet dan mengatur alamat IP. Untuk memeriksa koneksi internet yang tersedia, pengguna dapat melihat jaringan internet pada laptop atau komputer, dan untuk mendapatkan alamat IP, dapat dilakukan melalui menu > command prompt > ipconfig.

```
Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . : 
IPv6 Address. . . . . : 2404:c8:20c1:8567:9ad9:4398:6de5:6566
Temporary IPv6 Address. . . . . : 2404:c8:20c1:8567:ed48:a917:3f74:702c
Link-local IPv6 Address . . . . . : fe80::828b:b512:3b3f:d673%12
IPv4 Address. . . . . : 192.168.239.128
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::24f8:9fff:fe3e:20cf%12
                          192.168.239.94
```

Gambar 4.15 IP.

Setelah memperoleh alamat IP yang ditunjukkan pada Gambar 4.15, langkah berikutnya adalah memasukkan jaringan internet dan alamat IP tersebut ke dalam Arduino IDE untuk melakukan proses sinkronisasi antara ESP32 dan API.

```
5 const char *ssid = "KEBAR"; //ssid wifi
6 const char *password = "2110631170045"; //password wifi
7 const char *serverUrl = "http://192.168.239.128:3000/sensor-data";
```

Gambar 4.16 Sinkronisasi IP dan Internet.

Setelah proses sinkronisasi IP dan internet pada Gambar 4.16, selanjutnya adalah memastikan internet dan IP terhubung dengan sempurna, setelah mendapatkan Alamat IP, selanjutnya adalah mengaktifkan database melalui software Visual Studio Code.

```
18:26:27.084 -> MQ135 berhasil dibaca.
18:26:27.084 -> PMS5003 berhasil dibaca.
18:26:27.084 -> MQ135 - NOx: 4.51 ppb, CO2: 90.26 ppm, NH3: 22.56 ppm
18:26:27.084 -> PMS5003 - PM2.5: 53 µg/m³, PM10: 54
```

Gambar 4.17 Informasi pada Serial Monitor.

Pada Gambar 4.17 ditampilkan informasi dari serial monitor yang menunjukkan status koneksi perangkat. Informasi ini berfungsi sebagai indikator bahwa perangkat telah berhasil terhubung dengan jaringan dan siap untuk mengirim data.



Gambar 4.18 Informasi Riwayat data sensor pada dashboard aplikasi.

Pada Gambar 4.18 ditampilkan informasi riwayat data sensor yang tersedia pada dashboard aplikasi. Informasi ini menyajikan data historis hasil pemantauan kualitas udara yang telah direkam oleh sensor ke server dan dikirim ke perangkat android melalui server. Dengan adanya fitur ini, pengguna dapat melihat tren perubahan kualitas udara dari waktu ke waktu.

4.4. Pengujian

Setelah tahap implementasi selesai, langkah berikutnya adalah menjelaskan hasil dari proses tersebut, termasuk hasil pengujian yang telah dilakukan. Pengujian dilakukan menggunakan metode black-box testing untuk mengevaluasi kinerja sistem. Berikut adalah tahapan hasil pengujian perangkat IoT dan aplikasi Android.

4.4.1. Hasil Pengujian Perangkat IoT

Hasil evaluasi menunjukkan bahwa seluruh skenario berjalan dengan lancar. Ringkasan hasil pengujian sistem IoT disusun berdasarkan sejumlah aspek kunci yang telah diuji :

Tabel 4.5 Tabel pengujian perangkat IoT dengan balckbox.

No	Skenario	Hasil	Ket.
1.	Supply Power Test	Perangkat menyala saat diberikan supply power	Berhasil
2.	Wi-Fi Connection Test	ESP32 terhubung ke wifi dalam 2 m/s.	Berhasil
3.	MQ-135 Gas Sensor Test	Sensor MQ-135 dapat mendeteksi gas berbahaya.	Berhasil
4.	PMS5003 Particle Sensor Test	Sensor PMS5003 dapat mendeteksi partikel debu (PM2.5, PM10).	Berhasil
5.	Data Transmission Test	Data sensor berhasil dikirim ke REST API dan disimpan dalam database.	Berhasil
6.	Android App Display Test	Aplikasi Android dapat menampilkan data dengan benar sesuai parameter yang dipantau.	Berhasil
7.	System Load Handling Test	Sistem dapat menangani beban data tinggi atau gangguan jaringan dengan baik tanpa mengganggu masalah krusial.	Berhasil

Pada Tabel 4.5 ditampilkan hasil dari pengujian blackbox testing yang dilakukan pada perangkat keras. Pengujian ini dilakukan berdasarkan skenario yang telah dirancang sebelumnya untuk memastikan bahwa setiap

komponen perangkat keras dapat berfungsi sesuai dengan spesifikasi yang diharapkan.

4.4.2. Hasil Pengujian Aplikasi Adroid

Pengujian mengonfirmasi bahwa semua skenario terlaksana dengan baik. Berikut merupakan ringkasan hasil evaluasi pengujian aplikasi Android yang didasarkan pada berbagai aspek utama yang diuji :

Tabel 4.6 Tabel pengujian aplikasi Android dengan balckbox.

No	Skenario	Hasil	Ket.
1	Launch App Test	Aplikasi terbuka dengan lancar tanpa error	Berhasi 1
2	User Login Test	Aplikasi menerima input login dan masuk dengan validasi yang tepat	Berhasi 1
3	Data Retrieval Test	Data berhasil diambil dari REST API dan ditampilkan secara akurat	Berhasi 1
4	UI Responsi veness Test	Tampilan antarmuka responsif dan sesuai dengan desain	Berhasi 1
5	Real-time Data Update Test	Data diperbarui secara real-time tanpa delay yang signifikan	Berhasi 1
6	PDF Save Test	Pdf tersimpan ke penyimpanan perangkat pada android	Berhasi 1
7	Error Handling Test	Aplikasi menampilkan pesan kesalahan yang sesuai pada kondisi gangguan koneksi atau server	Berhasi 1
8	System Load Handling Test	Aplikasi mampu menangani beban data tinggi tanpa mengalami crash atau lag	Berhasi 1

Pada Tabel 4.6 ditampilkan hasil dari pengujian blackbox testing yang dilakukan

pada aplikasi Android. Pengujian ini dilakukan berdasarkan skenario yang telah dirancang sebelumnya untuk memastikan bahwa setiap fitur dalam aplikasi berfungsi sesuai dengan spesifikasi yang diharapkan.

4.5. Desiminasi

Berdasarkan hasil penelitian yang telah dilakukan dalam merancang sistem pemantauan kualitas udara berbasis Internet of Things (IoT), terdapat beberapa aspek yang perlu dibahas secara mendalam. Sistem ini dirancang dengan mempertimbangkan kebutuhan pengukuran kualitas udara yang efektif, serta proses perancangan perangkat keras dan perangkat lunak yang terintegrasi. Proses perancangan sistem meliputi pembuatan desain sistem menggunakan diagram UML, desain interface aplikasi Android untuk visualisasi data, serta perancangan perangkat keras dengan penggunaan komponen IoT seperti mikrokontroler ESP-32, sensor MQ-135, dan PMS5003.

Pada tahap implementasi, desain yang telah dirancang diterapkan dengan mengintegrasikan perangkat keras dengan server melalui koneksi HTTP. Proses implementasi ini melibatkan pengambilan data kualitas udara dari sensor yang kemudian dikirim ke server untuk diproses dan divisualisasikan pada aplikasi Android. Setelah perangkat keras dan perangkat lunak berhasil diintegrasikan, pengujian dilakukan menggunakan metode blackbox testing untuk memastikan bahwa sistem berjalan sesuai dengan yang diharapkan, baik dari segi fungsi perangkat keras maupun perangkat lunak.

Sensor Data History

No	PM2.5 ($\mu\text{g}/\text{m}^3$)	PM10 ($\mu\text{g}/\text{m}^3$)	NOx (ppb)	NH ₃ (ppm)	CO ₂ (ppm)	Timestamp
1.	52,5	69,7	32,5	0,234	408,8	03/02/2025 10:00:00
2.	49,3	71,9	28,1	0,247	442,5	03/02/2025 10:00:10
3.	53,2	84,5	33,2	0,17	442,1	03/02/2025 10:00:20
4.	57,6	81	30	0,176	470,8	03/02/2025 10:00:30
5.	48,8	78,9	30,4	0,162	444,2	03/02/2025 10:00:40
6.	48,8	80,1	33,8	0,193	455,7	03/02/2025 10:00:50
7.	57,9	82,1	35,7	0,214	452,1	03/02/2025 10:01:00
8.	53,8	76,8	32,1	0,23	418	03/02/2025 10:01:10
9.	47,7	75,3	32,9	0,198	461,5	03/02/2025 10:01:20

Gambar 4.20 PDF Riwayat hasil deteksi

Selain itu, pada Gambar 4.20 ditampilkan tampilan PDF riwayat hasil deteksi kualitas udara pada tanggal 3 Februari 2025. Fitur ini

memungkinkan pengguna untuk melihat rekaman data kualitas udara dalam format yang lebih terstruktur, sehingga memudahkan analisis lebih lanjut. Dengan adanya fitur ini, pengguna dapat memantau perubahan kualitas udara dari waktu ke waktu dan mengambil tindakan yang diperlukan berdasarkan data historis yang tersedia.

Hasil pengujian menunjukkan bahwa sistem pemantauan kualitas udara berbasis IoT ini dapat bekerja dengan baik, di mana data kualitas udara dapat terkirim secara real-time ke aplikasi Android untuk divisualisasikan dengan jelas. Namun, dalam hal keamanan, sistem ini masih memiliki beberapa kelemahan, terutama terkait dengan kemungkinan kecurangan pada pengumpulan data yang dapat terjadi jika perangkat IoT tidak terpasang di lokasi yang tepat atau tidak terpantau dengan baik. Untuk itu, perlu adanya langkah-langkah lebih lanjut, seperti penerapan pengawasan lokasi dan kebijakan penggunaan sensor yang lebih ketat, guna meminimalisir potensi penyalahgunaan.

Secara keseluruhan, sistem yang telah dirancang dapat memberikan solusi yang efisien dalam memantau kualitas udara secara terus-menerus. Namun, beberapa perbaikan perlu dilakukan untuk meningkatkan akurasi dan keamanannya. Dengan pengembangan lebih lanjut, sistem ini berpotensi untuk diimplementasikan di berbagai tempat, termasuk di industri seperti PT. South Pacific Viscose, untuk mendukung upaya menjaga kualitas udara yang sehat bagi karyawan dan lingkungan sekitar.

5. KESIMPULAN

Berdasarkan hasil perancangan dan penerapan sistem pemantauan kualitas udara menggunakan sensor MQ-135, dan sensor PMS5003 dengan visualisasi data pada aplikasi android di area parkir PT. South Pacific Viscose, dapat di simpulkan beberapa hal berikut:

- Sistem pemantauan kualitas udara berbasis IoT berhasil dirancang dengan menggunakan sensor MQ-135 (gas berbahaya) dan PMS5003 (partikel debu), diproses oleh ESP-32, dan mengintegrasikan data ke aplikasi Android melalui REST API dan Room Database.

Sistem ini mampu menampilkan parameter kualitas udara secara real-time dengan akurasi yang baik, memberikan solusi lebih efisien dibandingkan metode pemantauan manual.

- b. Aplikasi Android yang dikembangkan berhasil menampilkan data kualitas udara dalam bentuk dashboard interaktif, memanfaatkan teknologi Retrofit, LiveData, dan ViewModel untuk memastikan kecepatan dan stabilitas akses data. Sistem ini membantu pekerja, khususnya yang memiliki gangguan pernapasan, untuk memonitor kondisi udara secara real-time dan mengambil tindakan preventif guna meminimalkan dampak kesehatan akibat polusi di lingkungan industri.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] T. V. Damayanti And R. E. Handriyono, "Monitoring Kualitas Udara Ambien Melalui Stasiun Pemantau Kualitas Udara Wonorejo, Kebonsari Dan Tandes Kota Surabaya," *Environ. Eng. J. Itats*, Vol. 2, No. 1, Pp. 11–18, Mar. 2022, Doi: 10.31284/J.Envitats.2022.V2i1.2897.
- [2] A. P. Marpaung, "Evaluasi Kualitas Udara Dan Dampaknya Terhadap Kesehatan Pernafasan Penduduk Kota Medan," *J. Ilm. Maksitek*, Pp. 105–111, Jun. 2023.
- [3] H. Subagiyo, R. Tri Wahyuni, M. Akbar, And F. Ulfa, "Rancang Bangun Sensor Node Untuk Pemantauan Kualitas Udara," *J. Sains Teknol. Dan Ind.*, Vol. 18, No. 1, Pp. 72–72, Jan. 2021, Doi: 10.24014/Sitekin.V18i1.11461.
- [4] S. Sadi, S. Mulyati, And P. B. Setiawan, "Internet Of Things Pada Sistem Monitoring Kualitas Udara Menggunakan Web Server," *Formosa J. Multidiscip. Res.*, Vol. 1, No. 4, Pp. 1085–1094, Aug. 2022, Doi: 10.55927/Fjmr.V1i4.679.
- [5] M. Hasanuddin And H. Herdianto, "Sistem Monitoring Dan Deteksi Dini Pencemaran Udara Berbasis Internet Of Things (Iot)," *J. Comput. Syst. Inform. Josyc*, Vol. 4, No. 4, Pp. 976–984, Aug. 2023, Doi: 10.47065/Josyc.V4i4.4034.
- [6] P. Yushananta, "Very Low-Cost, Internet Of Things (Iot) Air Quality Monitoring Platform," *J. Aisyah J. Ilmu Kesehat.*, Vol. 8, No. 2, Apr. 2023, Doi: 10.30604/Jika.V8i2.1919.
- [7] H. Budianto And B. Sumanto, "Perancangan Sistem Monitoring Kualitas Udara Dalam Ruang Berbasis Internet Of Things," *J. List. Instrumentasi Dan Elektron. Terap.*, Vol. 5, No. 1, Pp. 9–9, Apr. 2024, Doi: 10.22146/Juliet.V5i1.87423.
- [8] R. Muttaqin, W. S. W. Prayitno, N. E. Setyaningsih, And U. Nurbaiti, "Rancang Bangun Sistem Pemantauan Kualitas Udara Berbasis Iot (Internet Of Things) Dengan Sensor Dht11 Dan Sensor Mq135," *J. Pengelolaan Lab. Pendidik.*, Vol. 6, No. 2, Pp. 102–115, May 2024, Doi: 10.14710/Jplp.6.2.102-115.
- [9] I. P. Sari, A. Novita, A.-K. Al-Khowarizmi, F. Ramadhani, And A. Satria, "Pemanfaatan Internet Of Things (Iot) Pada Bidang Pertanian Menggunakan Arduino Uno3," *Blend Sains J. Tek.*, Vol. 2, No. 4, Pp. 337–343, Jun. 2024, Doi: 10.56211/Blendsains.V2i4.505.
- [10] A. Ramadhani Pri Haryoga, P. Purwanto, And E. H. Nurkifli, "Perancangan Sistem Absensi Pengurus Menggunakan Rfid Berbasis Internet Of Things (Iot) Pada Sekretariat Bem Fasilkom Unsika," *Jati J. Mhs. Tek. Inform.*, Vol. 8, No. 3, Pp. 3845–3851, Jun. 2024, Doi: 10.36040/Jati.V8i3.9812.
- [11] I. Wahyudi And A. Syazili, "Dashboard Monitoring Website Dosen Studi Kasus Universitas Bina Darma," *J. Pengemb. Sist. Inf. Dan Inform.*, Vol. 2, No. 3, Pp. 188–197, Nov. 2021, Doi: 10.47747/Jpsii.V2i3.555.
- [12] M. Waruwu, "Metode Penelitian Dan Pengembangan (R&D): Konsep, Jenis, Tahapan Dan Kelebihan," *J. Ilm. Profesi Pendidik.*, Vol. 9, No. 2, Pp. 1220–1230, May 2024, Doi: 10.29303/Jipp.V9i2.2141.
- [13] A. Munandar, N. D. M. Veronika, D. Abdulllah, And E. Sahputra, "Miniature Design Of Liquid Filling Machine Automatically Using Esp32 Based Iot (Internet Of Things)," *J. Komput. Inf. Dan Teknol.*, Vol. 3, No. 1, Jun. 2023, Doi: 10.53697/Jkomitek.V3i1.1185.
- [14] I. A. Rombang, L. B. Setyawan, And G. Dewantoro, "Perancangan Prototipe Alat Deteksi Asap Rokok Dengan Sistem Purifier Menggunakan Sensor Mq-135 Dan Mq-2," *Techne J. Ilm. Elektrotek.*, Vol. 21, No. 1, Pp. 131–144, Apr. 2022, Doi: 10.31358/Techne.V21i1.312.
- [15] H. Suryantoro And M. Kusriyanto, "Sistem Monitoring Partikel (Pm2.5) Air Purifier Untuk Mengetahui Kualitas Udara Berbasis Sensor Pms5003 Dan Arduino," *Indones. J. Lab.*, No. 3, Pp. 88–88, Nov. 2023, Doi: 10.22146/Ijl.V0i3.88043.

- [16] S. A. Setiawan, M. Hidayat, And Sutarti, "Prototype Lampu Penerangan Jalan Otomatis Menggunakan Sensor Ldr Berbasis Arduino Uno," *Prosisko J. Pengemb. Ris. Dan Obs. Sist. Komput.*, Vol. 11, No. 1, Pp. 119–127, Mar. 2024, Doi: 10.30656/Prosisko.V11i1.8257.
- [17] M. Z. Abdillah And I. G. S. Pranata, "Analisis Dan Perancangan Sistem Informasi Manajemen Gereja Menggunakan Uml (Unifed Modelling Language)," *J. Inform. Dan Tek. Elektro Terap.*, Vol. 12, No. 3, Art. No. 3, Aug. 2024, Doi: 10.23960/Jitet.V12i3.4831.