

BAB II

LANDASAN TEORI

Bab ini menyajikan tinjauan pustaka yang bertujuan untuk memberikan dasar teoritis dan landasan ilmiah bagi penelitian yang dilakukan. Tinjauan ini mencakup kajian terhadap penelitian-penelitian sebelumnya serta teori-teori yang relevan. Dalam merancang penelitian ini, peneliti merujuk pada berbagai studi terdahulu yang berkaitan, guna memperkuat perumusan masalah serta meninjau berbagai aspek yang berhubungan dengan topik yang diangkat dalam penelitian ini.

2.1. Penelitian Terdahulu

Pada bagian ini, akan diuraikan sejumlah penelitian terdahulu yang memiliki relevansi dengan topik yang dibahas, berdasarkan hasil studi literatur, dokumen, dan arsip yang mendukung. Peneliti memanfaatkan referensi dari berbagai studi sebelumnya sebagai pijakan dalam proses perancangan dan pengembangan sistem. Mengingat bahwa topik mengenai monitoring daya listrik berbasis *Internet of Things* telah diteliti oleh beberapa peneliti sebelumnya, berikut ini disajikan ringkasan dari penelitian-penelitian yang berkaitan.

Pada penelitian yang dilakukan oleh Wonohadidjojo dan Wibawa (2023) mengembangkan sistem pemantauan dan pengendalian alat elektronik berbasis *Internet of Things* (IoT) dengan memanfaatkan stop kontak pintar yang dapat dikendalikan melalui aplikasi *smartphone*. Sistem ini menggunakan kombinasi perangkat keras berupa Arduino Uno R3, NodeMCU ESP8266, sensor arus ACS712, sensor tegangan ZMPT101b, serta *modul relay*, dan terhubung ke Firebase sebagai *database real-time*. Aplikasi dikembangkan menggunakan Flutter agar dapat berjalan di Android dan iOS. Hasil pengujian menunjukkan bahwa sistem mampu mengontrol dan memantau tegangan serta arus listrik dengan tingkat akurasi yang cukup baik, yaitu error sebesar 0,1% untuk tegangan dan 0,2% untuk arus. Penelitian ini berhasil membuktikan bahwa kontrol jarak jauh terhadap peralatan listrik rumah tangga dapat dilakukan secara efektif melalui koneksi internet, meskipun masih terdapat kendala pada kestabilan koneksi antar mikrokontroler.

Pada penelitian yang dilakukan oleh James William Jokanan dan tim (2022) mengusung rancang bangun sistem *monitoring* daya listrik berbasis *Internet of Things* (IoT) dengan memanfaatkan Firebase sebagai basis data dan aplikasi Android sebagai antarmuka pengguna. Sistem yang dikembangkan menggunakan sensor PZEM-004T untuk mendeteksi nilai tegangan, arus, daya, dan energi listrik. Data dari sensor ini dibaca oleh mikrokontroler NodeMCU yang dilengkapi modul WiFi, kemudian dikirimkan secara real-time ke Google

Firestore. Aplikasi Android yang dibangun dengan Android Studio kemudian mengambil data tersebut untuk ditampilkan dalam bentuk grafik dan angka melalui antarmuka yang sederhana. Pengujian dilakukan dengan membandingkan hasil pembacaan sensor dengan alat ukur standar seperti clamp multimeter dan power meter. Hasil pengukuran menunjukkan tingkat error yang masih dapat diterima, yaitu rata-rata sebesar 0,35% untuk tegangan, 6,7% untuk arus, 6,6% untuk daya, dan 7,8% untuk energi. Dengan tingkat akurasi keseluruhan sekitar 95%, alat *monitoring* ini dinilai cukup efektif dan akurat dalam memantau konsumsi daya listrik. Latar belakang penelitian ini adalah membuat aplikasi untuk memantau penggunaan daya listrik di rumah secara real-time. Pengendalian ini dilakukan menggunakan mikrokontroler ESP8266 sebagai chip yang mendukung konektivitas internet melalui Wifi. Sensor yang digunakan yaitu sensor arus (ACS712) dan sensor tegangan (ZMPT101B) serta teknologi *Internet of Things* (IoT) melalui platform Blynk. Dengan aplikasi ini, pengguna dapat mengatur pemakaian energi listrik sesuai kebutuhan.

Pada penelitian yang dilakukan oleh Megi Lestari, Irwan, dan Indah Riezky Pratiwi (2024) bertujuan untuk merancang dan mengembangkan sistem pemantauan daya listrik berbasis website dengan memanfaatkan teknologi *Internet of Things* (IoT). Sistem ini dirancang untuk memberikan kemudahan kepada pengguna dalam memantau penggunaan daya listrik dari berbagai perangkat rumah tangga maupun industri secara jarak jauh melalui koneksi internet. Dalam implementasinya, sistem menggunakan sensor PZEM-004T yang berfungsi untuk mengukur arus, tegangan, daya, dan energi listrik dari perangkat elektronik. Data hasil pengukuran kemudian diolah oleh mikrokontroler ESP32 dan dikirimkan ke database melalui jaringan internet. *Website* yang menampilkan data tersebut dibangun menggunakan *framework* CodeIgniter 4 dengan antarmuka berbasis Bootstrap, dan data ditampilkan secara real-time dalam bentuk grafik. Hasil pengujian menunjukkan bahwa sistem mampu bekerja dengan akurat dan responsif. Berdasarkan perbandingan data antara sensor PZEM-004T dan alat ukur *power quality analyzer*, diperoleh rata-rata error sebesar 3,85 watt dengan tingkat akurasi sebesar 96,15%. Hal ini membuktikan bahwa sistem yang dibangun cukup andal untuk digunakan dalam pemantauan daya listrik sehari-hari. Dengan adanya sistem ini, pengguna dapat dengan mudah mengakses informasi konsumsi daya melalui website kapan saja dan di mana saja, sehingga dapat membantu mereka dalam mengontrol penggunaan energi listrik secara lebih bijak dan efisien.

Pada penelitian yang dilakukan Alya Rizky Natasya (2025) mengungkap tentang rancang bangun sistem *monitoring* daya listrik berbasis *Internet of Things* menggunakan protokol *message queuing telemetry transport* (mqtt). Penelitian ini menggunakan metode pendekatan

kuantitatif dengan model pengembangan sistem *prototype*, yang bertujuan untuk merancang dan membangun alat monitoring daya listrik berbasis *Internet of Things* (IoT). Model pengembangan *prototype* dipilih karena memungkinkan proses pembuatan sistem dilakukan secara bertahap, dimulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga perbaikan sistem secara berkelanjutan. Penelitian ini juga mengadopsi prinsip pengembangan sistem berbasis kebutuhan pengguna dan kemudahan pemantauan energi secara real-time. Perangkat keras utama yang digunakan meliputi sensor PZEM-004T untuk mengukur tegangan, arus, daya, dan energi listrik, serta mikrokontroler ESP32 yang dilengkapi modul WiFi untuk memproses dan mengirimkan data ke Firebase. Sistem ini dirancang agar data yang ditangkap oleh sensor dapat ditampilkan dalam bentuk grafik maupun angka melalui aplikasi Android, sehingga pengguna dapat memantau penggunaan daya listrik kapan saja dan di mana saja.

Dan pada penelitian Ignacia Putri Dinayah, Rizka Novembrianto (2023) mengenai emisi karbon, Penyumbang gas rumah kaca terbesar berasal dari bidang energi yakni penggunaan listrik dari aktivitas dalam gedung yang memiliki fungsi komersial seperti mal, perkantoran, hotel, dan apartemen. Penggunaan listrik tersebut menghasilkan 70% emisi CO₂. Untuk mengurangi emisi CO₂, beberapa langkah dapat diterapkan, termasuk mematikan perangkat elektronik yang tidak digunakan. Pada tabel 2.1 dijelaskan mengenai analisis GAP dari beberapa penelitian terdahulu.

Tabel 2.1 Ringkasan Penelitian Terdahulu

No	Penulis & Tahun	Judul Penelitian	Metode	Hasil Penelitian	Analisis GAP	Kebaruan
1	Wonoha didjojo dan Wibawa (2023)	Sistem pemantauan dan pengendalian alat elektronik berbasis <i>Internet of Things</i> (IoT) dengan memanfaatkan stop kontak	Metode rekayasa perangkat keras dan perangkat lunak dengan pendekatan eksperimental. Sistem menggunakan NodeMCU,	Sistem stop kontak pintar berbasis IoT berhasil berfungsi dengan baik dan mampu memantau serta mengendalikan aliran listrik	Penelitian sebelumnya umumnya menggunakan jaringan jarak pendek dan hanya mendukung satu platform serta belum terintegrasi	Mengintegrasikan pemantauan arus dan tegangan serta pengendalian perangkat listrik secara <i>real-time</i> berbasis

No	Penulis & Tahun	Judul Penelitian	Metode	Hasil Penelitian	Analisis GAP	Kebaruan
		pintar yang dapat dikendalikan melalui aplikasi <i>smartphone</i>	Arduino Uno R3, sensor ACS712 dan ZMPT101b, modul relay, serta aplikasi Flutter dan Firebase.	secara <i>real-time</i> melalui aplikasi <i>smartphone</i> .	dengan penyimpanan <i>cloud</i> .	<i>cloud</i> menggunakan Firebase dengan aplikasi lintas platform (Android dan iOS).
2	James William Jokanan dkk. (2022)	Rancang Bangun Alat Monitoring Daya Listrik Berbasis IoT Menggunakan Firebase dan Aplikasi Android	Metode kuantitatif dengan pendekatan prototyping menggunakan sensor PZEM-004T dan NodeMCU, data dikirim ke Firebase dan ditampilkan pada aplikasi Android.	Sistem monitoring daya listrik berfungsi dengan baik dengan tingkat error rendah dan mampu menampilkan data serta notifikasi pemakaian listrik secara <i>real-time</i> .	Penelitian sebelumnya masih bergantung pada aplikasi pihak ketiga dan belum optimal dalam pemantauan <i>real-time</i> .	Pemanfaatan aplikasi Android khusus yang terintegrasi dengan Firebase dan sensor PZEM-004T untuk monitoring daya listrik <i>real-time</i> dengan fitur notifikasi batas pemakaian.
3	Megi Lestari dkk. (2024)	Sistem Pemantauan Daya Listrik Berbasis Website	Metode waterfall dengan tahapan analisis,	Sistem monitoring daya listrik berbasis website	Penelitian sebelumnya belum mendukung pemantauan	Penyediaan sistem monitoring daya listrik berbasis

No	Penulis & Tahun	Judul Penelitian	Metode	Hasil Penelitian	Analisis GAP	Kebaruan
			perancangan, implementasi, dan pengujian. Menggunakan ESP32 dan sensor PZEM-004T.	berjalan baik dengan akurasi rata-rata 96,15% dan menampilkan data secara <i>real-time</i> .	jarak jauh yang mudah diakses lintas perangkat.	website yang dapat diakses lintas perangkat tanpa aplikasi khusus, dengan integrasi ESP32 dan PZEM-004T.
4	Alya Rizky Natasya (2025)	Rancang Bangun Sistem Monitoring Daya Listrik Berbasis <i>Internet of Things</i> Menggunakan Protokol MQTT	Metode <i>prototyping</i> dengan penerapan protokol MQTT, NodeMCU ESP8266, <i>website monitoring</i> , dan notifikasi Telegram Bot.	Sistem monitoring daya listrik bekerja stabil dengan akurasi tinggi serta mampu memberikan notifikasi otomatis saat konsumsi melebihi batas.	Studi sebelumnya belum mengoptimalkan protokol MQTT dan belum menyediakan notifikasi <i>real-time</i> yang responsif.	Optimalisasi protokol MQTT untuk efisiensi transmisi data serta integrasi notifikasi Telegram Bot dan estimasi biaya listrik secara <i>real-time</i> .
5	Ignacia Putri Dinayah dan Rizka Novembrianto (2023)	Analisis Emisi CO ₂ dari Penggunaan Listrik di Lingkungan Kampus UPN “Veteran” Jawa Timur	Analisis data sekunder konsumsi listrik dengan perhitungan emisi karbon menggunakan faktor emisi	Total emisi karbon dari penggunaan listrik di UPN “Veteran” Jawa Timur sebesar 156,362 ton	Tidak menggunakan sistem pemantauan berbasis sensor atau IoT serta belum	Fokus pada analisis dampak lingkungan dari konsumsi listrik sebagai dasar

No	Penulis & Tahun	Judul Penelitian	Metode	Hasil Penelitian	Analisis GAP	Kebaruan
			standar nasional.	CO ₂ /kWh dengan gedung kuliah sebagai penyumbang terbesar.	mempertimbangkan variasi faktor emisi regional.	evaluasi emisi karbon, meskipun tanpa integrasi sistem monitoring IoT.

2.2. Landasan Teori

Pada bab ini membahas landasan teori yang memiliki kaitan erat dengan topik penelitian. Tujuan dari penyajian landasan teori ini adalah untuk memberikan pijakan konseptual yang kuat, mendukung argumen yang dikemukakan dalam penelitian, serta memperjelas definisi dan peran dari variabel-variabel yang digunakan.

2.2.1. Metode *Prototype*

Metode *Prototype* merupakan salah satu model pengembangan sistem yang berorientasi pada pengguna, dimana proses pengembangan dilakukan melalui pembuatan model awal atau sketsa sistem yang kemudian dievaluasi oleh pengguna untuk memperoleh umpan balik sebelum sistem akhir dibangun. Pendekatan ini memungkinkan interaksi intensif antara pengguna dan pengembang, sehingga kebutuhan sistem dapat dipahami secara lebih akurat sejak tahap awal pengembangan.

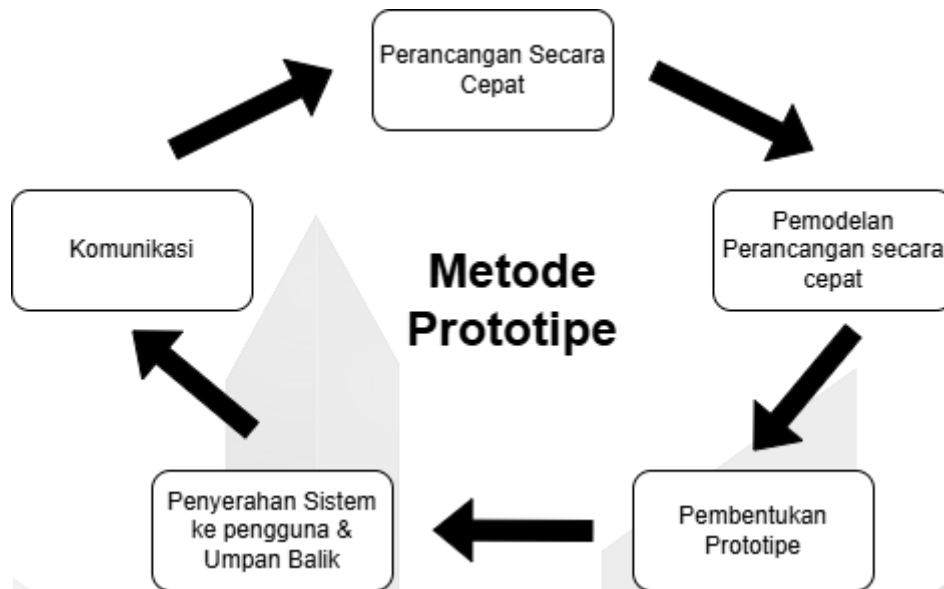
Metode ini berorientasi pada pembangunan sistem secara bertahap dan iteratif, di mana pengembang membuat model awal (*prototype*) untuk kemudian dievaluasi langsung oleh pengguna. Hasil evaluasi dan umpan balik tersebut menjadi dasar perbaikan hingga sistem akhir mencapai bentuk yang sesuai dengan kebutuhan pengguna. *Prototype* merupakan salah satu pendekatan software engineering yang efektif untuk menjembatani kesenjangan antara kebutuhan pengguna dan rancangan sistem. Dalam konteks penelitian lokal, Wonohadidjojo & Wibawa (2023) menyatakan bahwa metode *prototype* sangat sesuai untuk pengembangan sistem berbasis IoT karena melibatkan interaksi aktif antara pengguna dan sistem selama proses pengembangan. Model ini juga digunakan dalam penelitian Jokanan dkk. (2022) yang mengembangkan sistem monitoring daya listrik berbasis Firebase dan Android, di mana

prototype awal diuji untuk memastikan akurasi dan kestabilan data sebelum implementasi akhir.

Selain itu, Natasya (2025) juga menerapkan metode *prototype* dalam penelitian monitoring daya listrik berbasis protokol MQTT, dengan hasil bahwa pendekatan iteratif mempercepat proses validasi dan mengurangi kesalahan rancangan pada tahap akhir. Oleh karena itu, metode *prototype* dipilih dalam penelitian ini karena dapat menghasilkan sistem yang sesuai kebutuhan pengguna, fleksibel terhadap perubahan, dan mampu diuji sejak tahap awal pengembangan. Tahapan pelaksanaan metode *prototype* dalam penelitian ini meliputi:

1. Identifikasi dan Analisis Kebutuhan, dilakukan melalui observasi dan wawancara untuk menentukan kebutuhan fungsional sistem, seperti pemantauan tegangan, arus, daya, energi, dan pembatas arus (*control limit current*).
2. Perancangan Cepat (Quick Design), berupa penyusunan diagram blok, alur sistem, dan desain awal antarmuka aplikasi berbasis Flutter untuk menggambarkan rancangan sistem secara umum.
3. Pembuatan *Prototype* Awal, yaitu implementasi rancangan menggunakan ESP32, sensor PZEM-004T, relay, dan Firebase Realtime Database untuk menghasilkan sistem yang dapat menampilkan data listrik secara real-time.
4. Evaluasi dan Umpan Balik Pengguna, di mana *prototype* diuji oleh pengguna untuk memastikan kesesuaian fungsi dan tampilan, kemudian diberikan masukan untuk perbaikan sistem.
5. Revisi dan Penyempurnaan *Prototype*, dilakukan berdasarkan hasil evaluasi untuk meningkatkan akurasi sensor, optimasi tampilan aplikasi, serta penambahan fitur estimasi biaya dan emisi karbon.
6. Implementasi dan Pengujian Sistem Akhir, mencakup uji fungsionalitas (*black box testing*), uji stabilitas sensor, dan uji *Quality of Service (QoS)* untuk memastikan sistem berjalan stabil, akurat, dan efisien.

Metode ini dinilai paling sesuai dengan karakteristik penelitian karena sistem yang dikembangkan bersifat real-time, interaktif, dan berorientasi pada pengguna, sebagaimana juga dibuktikan oleh penelitian Natasya (2025) dan Muhtadi (2025) yang berhasil mengimplementasikan sistem monitoring listrik berbasis IoT dengan hasil akurasi dan respon sistem yang baik. Secara alur dapat digambarkan pada gambar 2.1



Gambar 2.1 Metode *Prototype*

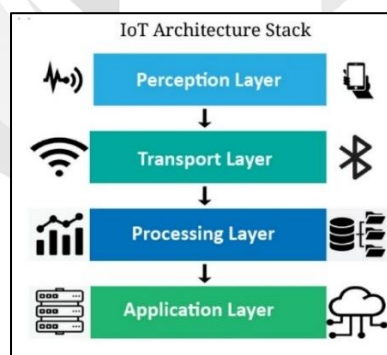
2.2.2. Sistem *Monitoring*

Sistem dapat dipahami sebagai sekumpulan elemen yang saling berinteraksi dan bekerja sama untuk mencapai tujuan tertentu. Dalam perspektif lain, sistem juga diartikan sebagai himpunan komponen yang saling terkait satu sama lain, baik secara struktural maupun fungsional. Sementara itu, *monitoring* merujuk pada suatu proses pengawasan yang bertujuan untuk memastikan tercapainya tujuan organisasi dan manajemen secara efektif. *Monitoring* juga dapat didefinisikan sebagai suatu mekanisme evaluatif yang menilai kesesuaian pelaksanaan kegiatan terhadap rencana yang telah ditetapkan, mengidentifikasi permasalahan yang muncul agar dapat segera ditangani, serta mengevaluasi efektivitas metode kerja dan manajerial yang diterapkan. Selain itu, monitoring juga mencermati keterkaitan antara aktivitas yang dilakukan dengan tujuan yang ingin dicapai, guna mengukur tingkat kemajuan yang diperoleh. Dengan demikian, monitoring merupakan proses penelaahan dan evaluasi terhadap informasi mengenai pelaksanaan suatu kegiatan atau proyek, untuk menilai peningkatan kinerja serta memastikan kepatuhan terhadap kebijakan dan regulasi yang berlaku (Laurentinus et al., 2024). Sistem monitoring merupakan suatu mekanisme yang berfungsi untuk melakukan pengumpulan, pengolahan, dan evaluasi data secara berkelanjutan guna memastikan proses suatu kegiatan berjalan sesuai dengan rencana dan sasaran yang telah ditentukan. Menurut (Manurung et al., 2025) sistem monitoring memiliki peran penting dalam memantau pelaksanaan kegiatan serta memberikan umpan balik yang berguna bagi pengambilan keputusan manajerial secara tepat waktu. Dalam penelitian lain (Al Fath et al., 2025)

menjelaskan bahwa sistem monitoring modern biasanya terintegrasi dengan teknologi informasi berbasis IoT dan cloud computing, yang memungkinkan pengawasan dilakukan secara real-time dari berbagai lokasi. Selain itu, (Purba et al., 2025) menegaskan bahwa sistem monitoring tidak hanya berfungsi sebagai alat pengawasan pasif, tetapi juga menjadi sistem proaktif untuk meningkatkan efisiensi, transparansi, dan akuntabilitas dalam berbagai bidang seperti pendidikan, industri, dan pemerintahan. Dengan demikian, sistem monitoring dapat dipahami sebagai komponen penting dalam mendukung pengambilan keputusan dan peningkatan kinerja organisasi secara berkelanjutan.

2.2.3. *Internet of Things*

Internet of Things (IoT) adalah konsep yang menghubungkan berbagai perangkat fisik ke jaringan internet sehingga perangkat tersebut dapat saling berkomunikasi, bertukar data, dan dikendalikan secara otomatis tanpa campur tangan langsung manusia. *Perception Layer* berfungsi sebagai lapisan paling bawah yang mendeteksi dan mengumpulkan data dari lingkungan fisik melalui sensor dan aktuator, seperti sensor suhu, kelembapan, arus, atau tegangan. Data yang terkumpul kemudian dikirim melalui *Transport Layer*, yaitu lapisan komunikasi yang menggunakan berbagai protokol seperti Wi-Fi, Bluetooth, ZigBee, MQTT, atau HTTP untuk mentransfer data secara real-time ke sistem penyimpanan atau *cloud*. Setelah itu, *Processing Layer* bertugas menyimpan, memproses, dan menganalisis data menggunakan server, database, atau layanan cloud computing seperti Firebase untuk menghasilkan informasi yang bermanfaat dan mendukung pengambilan keputusan. Terakhir, *Application Layer* menyajikan hasil pemrosesan data dalam bentuk antarmuka pengguna, baik melalui aplikasi *mobile*, *dashboard web*, maupun sistem otomasi. Keempat lapisan ini bekerja secara terintegrasi membentuk ekosistem IoT yang memungkinkan proses *monitoring*, analisis, dan pengendalian berbagai sistem berjalan secara efisien, *real-time*, dan cerdas. Dari penjelasan diatas untuk visual layer dapat dilihat pada gambar 2.2



Gambar 2.2 *Internet of Things*
(Sumber : www.researchgate.net)

2.2.4. Tegangan Listrik

Tegangan listrik merupakan selisih potensial listrik antara dua titik dalam suatu rangkaian, yang diukur dalam satuan volt. Besaran ini menunjukkan energi potensial dalam medan listrik yang mendorong terjadinya aliran arus listrik melalui konduktor. Berdasarkan besar kecilnya perbedaan potensial tersebut, tegangan listrik dapat digolongkan sebagai sangat rendah, tinggi, atau sangat tinggi. Secara prinsip, tegangan listrik menyebabkan muatan negatif berpindah dari daerah dengan tegangan lebih rendah ke daerah dengan tegangan lebih tinggi. Namun, secara konvensional, arah arus listrik dalam konduktor dianggap mengalir dari tegangan tinggi ke tegangan rendah (Alan et al., 2025)

2.2.5. Arus Listrik

Arus listrik adalah aliran muatan listrik yang terjadi akibat adanya perbedaan potensial antara dua titik dalam suatu rangkaian listrik. Arus ini biasanya dibawa oleh elektron dalam konduktor logam, dan besarnya diukur dalam satuan ampere (A). Secara konvensional, arus dianggap mengalir dari kutub positif ke kutub negatif, meskipun arah sebenarnya dari aliran elektron adalah sebaliknya (Sidiq, 2023).

2.2.6. Daya Listrik

Daya listrik adalah laju penggunaan energi listrik dalam suatu rangkaian, yang menunjukkan seberapa cepat energi listrik dikonversi menjadi bentuk energi lain seperti panas, cahaya, atau gerak (Azizah, 2025). Daya listrik dinyatakan dalam satuan watt (W), dan secara matematis dirumuskan sebagai hasil kali antara tegangan (volt) dan arus listrik (ampere), yaitu:

$$P = V \times I \quad (2.1)$$

Rumus 2.1 Daya Listrik

Dimana :

P = Daya Listrik Dengan Satuan Watt (W)

V = Tegangan Listrik dengan satuan Volt (V)

I = Arus listrik dengan satuan ampere (A)

2.2.7. Energi Listrik

Energi listrik didefinisikan sebagai energi yang dihasilkan oleh pergerakan muatan listrik dalam medan listrik atau dalam suatu rangkaian listrik tertutup (Nasrudin et al., 2024). Secara fisika, energi listrik dapat dihitung melalui hubungan antara tegangan listrik (V), arus listrik (I), dan waktu (t), dengan rumus umum:

$$W = V \cdot I \cdot t \text{ atau } W = I^2 \cdot R \cdot t \quad (2.2)$$

Rumus 2.2 Energi Listrik

Dimana:

- W = Energi dalam joule (J)
- V = Tegangan dalam volt (V)
- I = Arus dalam ampere (A)
- R = Hambatan dalam ohm (Ω)
- t = Waktu

Energi listrik dapat pula dipahami sebagai bentuk energi yang berasal dari aliran elektron melalui konduktor akibat perbedaan potensial. Energi ini dapat dikonversi ke berbagai bentuk energi lainnya, seperti energi panas, cahaya, dan gerak, tergantung pada medium dan perangkat yang digunakan.

2.2.8. Perhitungan kWh

Kilowatt-hour adalah satuan energi yang digunakan untuk mengukur jumlah energi listrik yang digunakan oleh suatu perangkat listrik selama periode waktu tertentu. Satu *kilowatt-hour* didefinisikan sebagai energi yang digunakan oleh perangkat dengan daya 1 kilowatt (1000 watt) yang beroperasi selama 1 jam. Dengan demikian, secara matematis, energi dalam kWh dapat dirumuskan sebagai:

$$\text{Energi (kWh)} = P \times t / 1000 \quad (2.3)$$

Rumus 2.3 Kilowatt-hour (kWh)

Dimana :

- P = Daya (dalam watt)
- t = Waktu (dalam jam)
- 1000 = Faktor konversi dari watt ke kilowatt.

2.2.9. Perhitungan Estimasi Biaya Konsumsi Listrik

Perhitungan estimasi biaya konsumsi listrik didasarkan pada hubungan antara daya listrik yang digunakan, lama waktu pemakaian, serta tarif listrik per satuan energi (kWh) yang berlaku. Secara konseptual, besarnya energi listrik yang dikonsumsi oleh suatu peralatan diperoleh dengan mengalikan daya alat tersebut (dalam satuan watt) dengan durasi penggunaannya (dalam jam), kemudian hasilnya dibagi seribu untuk mengonversi satuan watt-

hour menjadi *kilowatt-hour* (kWh). Nilai energi dalam kWh inilah yang menjadi dasar perhitungan biaya listrik, yaitu dengan mengalikan total energi yang dikonsumsi dengan tarif listrik per kWh yang ditetapkan oleh PLN. Secara matematis, hubungan tersebut dapat dinyatakan melalui persamaan:

$$\text{Biaya Listrik (Rp)} = W \times t \times \text{Tarif per kWh (Rp)} / 1000 \quad (2.4)$$

Rumus 2.4 Perhitungan Estimasi Biaya

Dimana:

W = Daya

t = Waktu

Tarif per kWh = harga listrik per kilowatt-jam (1.444,70 untuk pelanggan nonsubsidi 1.300 VA ke atas — data PLN 2025)

/1000 = karena 1 kWh = 1000 watt-jam

Rumus ini menunjukkan bahwa biaya listrik berbanding lurus dengan besar daya peralatan dan lama waktu pemakaian, sehingga semakin tinggi konsumsi daya dan semakin lama penggunaan suatu perangkat, maka semakin besar pula energi listrik yang digunakan serta biaya yang harus dikeluarkan. Pendekatan ini sejalan dengan prinsip dasar analisis konsumsi energi yang digunakan dalam berbagai penelitian empiris mengenai efisiensi dan perilaku konsumsi listrik rumah tangga (Rasyid & Kristina, 2021)

Berdasarkan dokumen resmi tarif listrik PLN, untuk periode Triwulan IV 2025 dan Triwulan I 2026 (Januari–Maret 2026), tidak terjadi perubahan tarif listrik per kWh dibanding periode sebelumnya. Hal ini sejalan dengan keputusan pemerintah dalam rangka menjaga stabilitas ekonomi dan daya beli masyarakat, sehingga tarif listrik tetap berlaku sama sampai dengan awal tahun 2026. Untuk rumah tangga pelanggan non-subsidi, tarif listrik per kWh ditetapkan sebagai berikut: pelanggan dengan daya 900 VA non-subsidi sebesar sekitar Rp 1.352/kWh, daya 1.300 VA dan 2.200 VA sebesar Rp 1.444,70/kWh, serta pelanggan dengan daya lebih besar (≥ 3.500 VA) sebesar Rp 1.699,53/kWh. Sementara itu, untuk pelanggan listrik bersubsidi, tarifnya lebih rendah, yakni sekitar Rp 415/kWh untuk daya 450 VA dan Rp 605/kWh untuk daya 900 VA subsidi. Tarif untuk golongan bisnis, industri, dan layanan pemerintah juga telah ditetapkan sesuai ketentuan tarif PLN berdasarkan golongan daya dan jenis layanan pelanggan.

Penetapan tarif ini dijadikan sebagai dasar perhitungan biaya konsumsi energi listrik dalam penelitian, khususnya pada bagian analisis biaya penggunaan listrik serta estimasi efisiensi energi. Dengan mengacu pada tarif resmi per kWh tersebut, penelitian ini memiliki landasan data operasional yang valid, konsisten, dan sesuai dengan regulasi nasional yang berlaku pada periode waktu penelitian.

2.2.10. Perhitungan Emisi Karbon

Perhitungan emisi karbon pada dasarnya dilakukan untuk mengetahui besarnya gas rumah kaca, khususnya CO₂, yang dilepaskan akibat konsumsi energi listrik. Rumus yang umum digunakan adalah

$$ECO_2 = KE \times FE \times GWP \quad (2.5)$$

Rumus 2.5 Perhitungan Emisi Karbon

Dimana:

ECO₂ = Emisi karbon (dinyatakan dalam kilogram CO₂)

KE = Jumlah konsumsi energi listrik dalam satuan kilowatt-hour (kWh)

FE = Faktor emisi listrik (kg CO₂/kWh) yang nilainya ditetapkan oleh instansi berwenang, GWP atau Global Warming Potential dari CO₂ bernilai 1 sebagai faktor pengali

Untuk perhitungan emisi CO₂ dari konsumsi listrik, penelitian ini menggunakan faktor emisi listrik yang ditetapkan dalam dokumen Faktor Emisi Gas Rumah Kaca (GRK) Sistem Interkoneksi Ketenagalistrikan oleh Direktorat Jenderal Ketenagalistrikan, Kementerian Energi dan Sumber Daya Mineral Republik Indonesia (nilai dalam ton CO₂/MWh kemudian dikonversi ke kg CO₂/kWh). Selain itu, laporan emisi PLN mencatat bahwa intensitas emisi listrik nasional berada sekitar 0,73 kg CO₂e/kWh, yang mencerminkan rata-rata emisi CO₂ per unit listrik yang dihasilkan dan disuplai.

2.2.11. Arduino IDE

Arduino Integrated Development Environment (Arduino IDE) merupakan perangkat lunak sumber terbuka (open-source software) yang dirancang untuk memudahkan proses pemrograman papan mikrokontroler Arduino dan papan lain yang kompatibel seperti ESP8266 dan ESP32. Arduino IDE pertama kali dikembangkan oleh tim Arduino dengan tujuan utama untuk mendemokratisasi pemrograman mikrokontroler, khususnya dalam lingkungan pendidikan, *prototype* cepat, dan penelitian (D. Setiawan et al., 2022). Arduino IDE memiliki

antarmuka yang sederhana, ringan, dan ramah pengguna, sehingga cocok digunakan oleh pemula maupun pengembang profesional.

Arduino IDE menggunakan bahasa pemrograman berbasis C/C++, dengan tambahan pustaka (library) bawaan yang memungkinkan pengguna untuk mengakses fitur perangkat keras secara mudah, seperti GPIO, ADC, komunikasi serial, I2C, PWM, dan lainnya. Salah satu keunggulan Arduino IDE adalah adanya ribuan pustaka eksternal yang dapat diinstal langsung dari *Library Manager*, yang mendukung berbagai sensor, aktuator, dan modul komunikasi, seperti sensor suhu DHT11, sensor ultrasonik HC-SR04, serta modul WiFi ESP8266/ESP32 (D. Setiawan et al., 2022)

Selain itu, Arduino IDE juga menyediakan fitur *Serial Monitor* dan *Serial Plotter* yang memungkinkan pengguna untuk melakukan *debugging* dan pemantauan data secara *real-time*. Proses kompilasi kode dilakukan oleh *compiler avr-gcc*, dan program hasil kompilasi dapat diunggah ke papan mikrokontroler melalui port USB menggunakan *bootloader* bawaan. Arduino IDE juga mendukung integrasi dengan berbagai papan melalui *Board Manager*, dimana pengguna dapat menginstal dukungan untuk papan-papan lain seperti ESP32, STM32, dan ATtiny melalui URL eksternal (Hartanto et al., 2024). Dapat dilihat pada gambar 2.3 fitur, *tools* dan menu pada Arduino IDE dan penjelasan mengenai tiap menunya dijelaskan pada tabel 2.2.



Gambar 2.3 Arduino IDE
(Sumber : www.opta.findernet.com)

Tabel 2.2 Penjelasan Menu pada Arduino IDE

No.	Nama Menu	Fungsi
1.	<i>Verify / Compile</i>	Memeriksa kesalahan sintaks dalam kode sebelum diunggah ke papan Arduino.
2.	<i>Upload</i>	Mengunggah program yang telah dikompilasi ke papan mikrokontroler.
3.	<i>Select Board & Port</i>	Memilih jenis papan Arduino dan port komunikasi (misalnya COM3, ttyUSB0).
4.	<i>Open Serial Monitor</i>	Menampilkan data serial dari Arduino ke komputer dalam bentuk teks.
5.	<i>Open Serial Plotter</i>	Menampilkan data serial dalam bentuk grafik (plot) secara real-time.
6.	<i>Sketchbook</i>	Menampilkan daftar proyek/kode Arduino yang telah disimpan.
7.	<i>Board Manager</i>	Menambahkan atau memperbarui jenis papan yang didukung oleh IDE.
8.	<i>Library Manager</i>	Mengelola pustaka/library Arduino yang digunakan dalam pemrograman.
9.	<i>Debugger</i>	Melakukan debug terhadap kode secara bertahap untuk mendeteksi bug.
10.	<i>Search</i>	Mencari kata kunci, variabel, atau fungsi dalam proyek/kode.

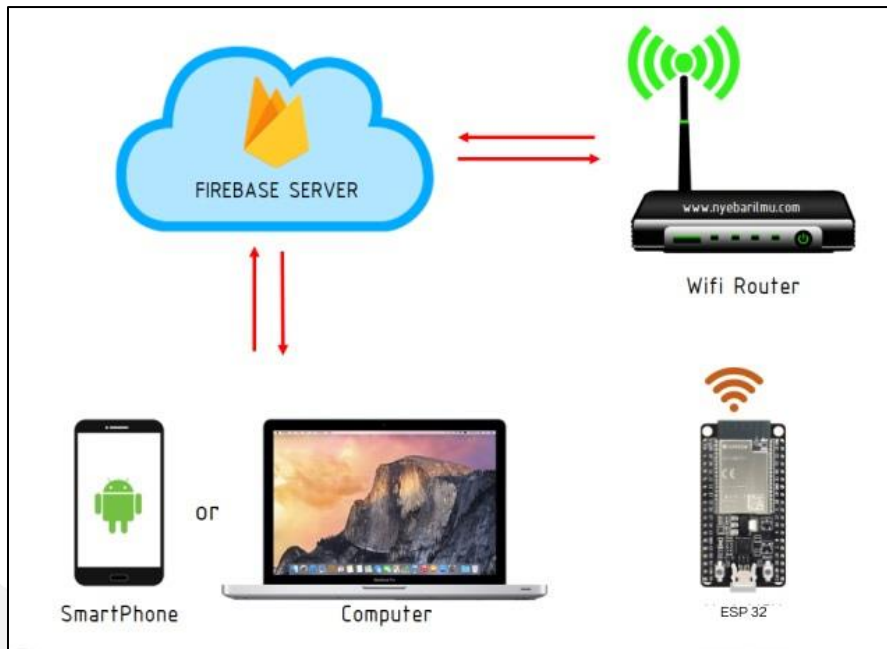
2.2.12. *Firebase*

Firebase adalah *platform* pengembangan aplikasi berbasis cloud yang dikembangkan oleh Google dan dirancang untuk menyederhanakan proses pengembangan *backend* aplikasi web maupun *mobile*. *Firebase* menyediakan berbagai layanan seperti *Realtime Database*, *Cloud Firestore*, *Authentication*, *Cloud Messaging*, dan *Cloud Functions* yang dapat digunakan secara terpadu tanpa perlu membangun infrastruktur server secara manual. *Firebase* menjadi populer di kalangan pengembang karena kemudahan integrasinya, terutama dengan *framework frontend* dan mikrokontroler seperti ESP32 untuk aplikasi *Internet of Things (IoT)* (Putra & Suryana, 2023)

Salah satu layanan paling banyak digunakan dari *Firestore* adalah *Realtime Database*, yaitu sistem basis data NoSQL berbasis JSON yang memungkinkan sinkronisasi data secara waktu nyata (*real-time*) antara klien dan server. Setiap kali ada perubahan data di server, *Firestore* secara otomatis akan mengirim pembaruan ke semua klien yang terhubung, sehingga sangat ideal untuk aplikasi yang memerlukan data dinamis seperti sistem monitoring IoT, smart home, dan dashboard kontrol (Putra & Suryana, 2023). *Firestore* juga menyediakan *Cloud Firestore*, sistem *database* yang lebih modern, fleksibel, dan skalabel, yang mendukung kueri kompleks serta struktur dokumen dan koleksi.

Firestore bekerja dengan prinsip komunikasi *client-server* berbasis cloud, di mana setiap interaksi antara aplikasi pengguna, server, dan perangkat IoT dilakukan melalui protokol komunikasi yang aman dan efisien. Secara umum, *Firestore* menggunakan kombinasi beberapa protokol utama, yaitu HTTPS (*Hypertext Transfer Protocol Secure*), *WebSocket*, dan gRPC (*Google Remote Procedure Call*) untuk mengelola pertukaran data secara *real-time* maupun *request-response*. Protokol HTTPS digunakan sebagai jalur komunikasi utama antara klien dan server *Firestore* untuk memastikan keamanan data selama proses autentikasi, pembacaan, dan penulisan data. HTTPS bekerja di atas protokol TLS/SSL yang mengenkripsi data, sehingga mencegah penyadapan dan manipulasi informasi selama transmisi (Prasetyo et al., 2023). Untuk mendukung kebutuhan sinkronisasi data *real-time*, *Firestore Realtime Database* memanfaatkan protokol *WebSocket*, yaitu protokol komunikasi dua arah (*full-duplex*) yang menjaga koneksi tetap terbuka antara klien dan server. Dengan *WebSocket*, setiap perubahan data pada server akan langsung dikirim ke seluruh klien yang terhubung tanpa perlu melakukan permintaan berulang (*polling*), sehingga efisien dan hemat *bandwidth* (Putra & Suryana, 2023)

Dapat dilihat pada gambar 2.4 sistem kerja *firebase* yang bekerja dengan prinsip komunikasi *client-server* berbasis cloud, di mana *Firestore* bertindak sebagai *backend* yang menyediakan berbagai layanan terintegrasi bagi aplikasi *frontend* maupun perangkat *Internet of Things* (IoT).



Gambar 2.4 Sistem Kerja *Firebase*
(Sumber : www.nyebarilmu.com)

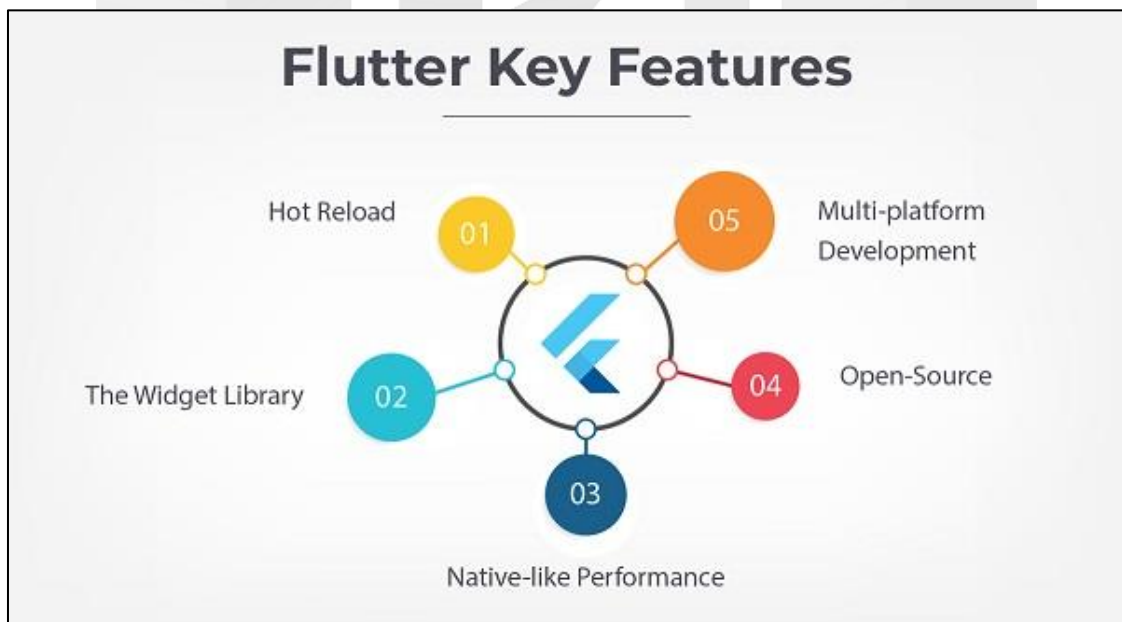
2.2.13. Flutter

Flutter merupakan *Software Development Kit (SDK) open-source* yang dikembangkan oleh Google, yang digunakan untuk membangun antarmuka aplikasi (*user interface*) secara cepat dan dinamis untuk berbagai platform, seperti Android, iOS, web, hingga desktop, dengan menggunakan satu basis kode. Flutter pertama kali dirilis pada tahun 2017 dan kini menjadi salah satu teknologi pengembangan lintas platform (*cross-platform*) yang paling populer karena efisiensinya dalam mempercepat proses pengembangan aplikasi (Pradipta et al., 2023) Bahasa pemrograman utama yang digunakan dalam Flutter adalah Dart, yang dikembangkan juga oleh Google dan dirancang untuk menghasilkan performa tinggi pada antarmuka pengguna yang kompleks.

Salah satu fitur unggulan dari Flutter adalah kemampuan *Hot Reload*, yang memungkinkan pengembang untuk melihat hasil perubahan kode secara langsung tanpa perlu melakukan kompilasi ulang secara penuh. Fitur ini sangat mempercepat proses *debugging* dan iterasi antarmuka. Selain itu, Flutter dilengkapi dengan *widget-widjet bawaan (built-in widgets)* yang sangat kaya dan dapat dikustomisasi, sehingga pengembang dapat menciptakan antarmuka yang responsif dan menarik dengan lebih sedikit baris kode (P. R. Setiawan et al., 2022).

Flutter sangat cocok untuk dikombinasikan dengan berbagai layanan *backend* seperti Firebase, karena mendukung integrasi pustaka-pustaka pihak ketiga seperti *firebase_auth*, *cloud_firestore*, dan *firebase_messaging*. Integrasi ini memungkinkan aplikasi yang dibangun dengan Flutter dapat melakukan proses autentikasi, menyimpan dan membaca data dari *database cloud*, serta menerima notifikasi push secara langsung. Dengan menggunakan arsitektur *client-server*, Flutter bertindak sebagai *frontend client* yang berinteraksi langsung dengan server melalui API atau layanan *cloud* seperti Firebase (Hidayat et al., 2024).

Pada gambar 2.5 dapat dilihat mengenai fitur utama dari Flutter yang membuatnya unggul sebagai *framework* pengembangan aplikasi lintas platform. *Hot Reload* memungkinkan pengembang melihat perubahan kode secara instan tanpa kompilasi ulang, mempercepat proses pengembangan. *Widget Library* menyediakan komponen UI yang lengkap dan dapat disesuaikan. *Native-like Performance* memastikan performa tinggi karena Flutter merender antarmuka langsung ke layar menggunakan mesin Skia. Sebagai *open-source*, Flutter bebas digunakan dan didukung komunitas aktif. Terakhir, fitur *multi-platform development* memungkinkan satu basis kode digunakan untuk berbagai platform seperti Android, iOS, web, dan desktop, sehingga lebih efisien dan hemat biaya.



Gambar 2.5 Flutter Key Features
(Sumber : www.genspark.ai)

2.2.14. Mikrokontroler

NodeMCU ESP32 merupakan salah satu modul pengembangan berbasis mikrokontroler ESP32 yang diproduksi oleh *Espressif Systems*. Modul ini dilengkapi dengan konektivitas nirkabel *Wi-Fi* dan *Bluetooth*, serta memiliki fitur-fitur yang memudahkan pengembangan sistem berbasis *Internet of Things* (IoT) secara cepat dan efisien. NodeMCU sendiri merupakan singkatan dari “*Node Microcontroller Unit*” dan mengintegrasikan board ESP32 dengan antarmuka USB, regulator daya, serta konektor GPIO yang mudah diakses. Keunggulan dari NodeMCU ESP32 dibandingkan dengan mikrokontroler konvensional adalah integrasi antara pemrosesan data dan komunikasi nirkabel dalam satu chip, yang menjadikannya solusi ideal untuk sistem otomatisasi, pemantauan jarak jauh, serta pengembangan *prototype* perangkat cerdas (Nasrudin et al., 2022).

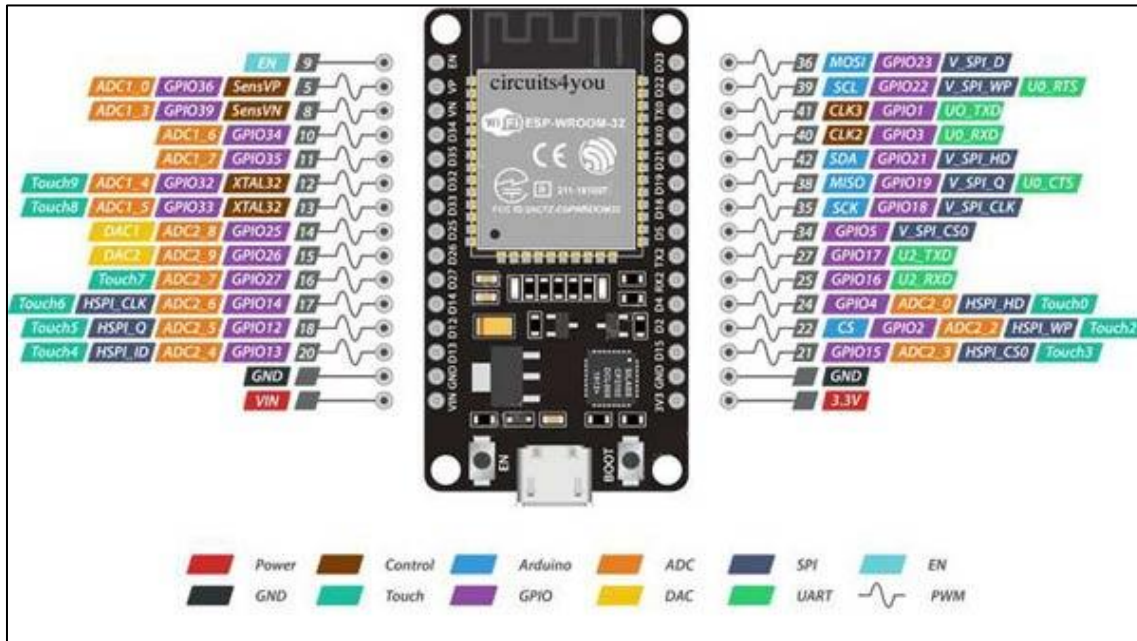
Modul ini menggunakan chip ESP32 dengan arsitektur *dual core Xtensa LX6* berkecepatan hingga 240 MHz, dilengkapi dengan memori SRAM sebesar 520 KiB dan dukungan memori *flash* eksternal hingga 16 MB. NodeMCU ESP32 juga mendukung berbagai protokol komunikasi digital seperti UART, SPI, I2C, serta dilengkapi dengan pin ADC 12-bit, PWM, touch sensor, dan DAC yang sangat bermanfaat dalam akuisisi dan kendali sinyal analog maupun digital (Kumar & Yadav, 2023). Modul ini dapat dijalankan dalam berbagai mode hemat daya (*deep-sleep*, *light-sleep*), dengan konsumsi daya yang sangat rendah sehingga sangat sesuai untuk aplikasi bertenaga baterai.

NodeMCU ESP32 mendukung beberapa platform pemrograman, di antaranya Arduino IDE, MicroPython, PlatformIO, dan ESP-IDF (*Espressif IoT Development Framework*). Hal ini menjadikan NodeMCU ESP32 sebagai alat yang fleksibel, baik untuk pengguna pemula maupun pengembang profesional. Selain itu, NodeMCU ESP32 juga dapat digunakan dalam berbagai sistem seperti pemantauan suhu dan kelembaban, pengontrol pintu otomatis, sistem penyiraman tanaman otomatis, serta sistem monitoring energi berbasis IoT

Menurut Espressif (2024), NodeMCU ESP32 menjadi populer dalam proyek-proyek IoT karena memiliki kemampuan konektivitas yang stabil, harga terjangkau, dan komunitas pengembang yang besar. Dengan didukung oleh fitur keamanan seperti enkripsi flash, secure boot, dan kriptografi perangkat keras (AES, RSA, SHA-2), modul ini juga cocok untuk diterapkan dalam sistem yang memerlukan proteksi data, seperti smart lock, pembayaran digital, atau pengendalian perangkat dari jarak jauh.

Dapat dilihat pada gambar 2.6 NodeMCU ESP32 merupakan modul mikrokontroler yang didesain untuk konektivitas jaringan *Wi-Fi* dan *Bluetooth*. Modul ini juga mendukung

konfigurasi pin yang memyudahkan pengguna dalam menghubungkan berbagai sensor dan perangkat lain.



Gambar 2.6 NodeMCU ESP32
(Sumber : www.ardutech.com)

2.2.15. PZEM 004T & Transformator Arus

Sensor PZEM-004T adalah modul pengukuran daya listrik berbasis mikrokontroler yang banyak digunakan dalam sistem monitoring energi berbasis *Internet of Things* (IoT). Sensor ini mampu mengukur beberapa parameter kelistrikan penting secara *real-time*, seperti tegangan (*volt*), arus listrik (*ampere*), daya aktif (*watt*), energi (kWh), frekuensi (Hz), dan faktor daya ($\cos \phi$). Modul PZEM-004T bekerja dengan prinsip pembacaan sinyal dari transformator arus (CT) dan tegangan AC dari jaringan listrik, yang kemudian diproses oleh mikrokontroler internal untuk menghasilkan data digital yang dikirim melalui komunikasi serial (UART) ke mikrokontroler eksternal seperti ESP32 atau Arduino (Rohman & Prasetya, 2022).

Salah satu keunggulan PZEM-004T adalah kemampuannya untuk melakukan pengukuran non-invasif menggunakan CT sensor (transformator arus tipe clamp) tanpa perlu memutus aliran listrik. CT ini bekerja dengan prinsip elektromagnetik, di mana arus listrik yang mengalir melalui konduktor primer akan menimbulkan fluks magnetik yang kemudian diinduksi ke lilitan sekunder pada CT. Arus yang dihasilkan pada sisi sekunder sebanding dengan arus pada sisi primer, sehingga dapat digunakan untuk mengukur nilai arus secara aman dan akurat. CT yang digunakan pada modul PZEM-004T umumnya memiliki rasio

transformasi 1000:1 dan dilengkapi dengan konektor jack agar mudah dipasang dan dilepas (Wibowo & Kartika, 2023).

Modul PZEM-004T dirancang untuk memudahkan integrasi dengan sistem monitoring berbasis mikrokontroler. Komunikasi antara PZEM-004T dan mikrokontroler dapat dilakukan melalui port UART (Tx/Rx) dengan protokol komunikasi khusus dari *Peacefair*, produsen modul ini. Modul ini juga tersedia dalam dua versi: versi TTL dan versi isolasi optik, di mana versi isolasi lebih aman untuk digunakan pada sistem bertegangan tinggi karena memisahkan sirkuit logika dari sirkuit daya.

Dalam penerapan praktisnya, PZEM-004T sangat cocok digunakan untuk sistem monitoring konsumsi daya listrik pada rumah tangga, industri kecil, dan sistem tenaga surya. Modul ini dapat dikombinasikan dengan platform cloud seperti *Firebase* atau *Blynk* untuk menampilkan data secara *real-time* melalui perangkat mobile. Akurasi pengukuran yang tinggi dan kemudahan integrasi menjadikan sensor ini sebagai pilihan populer dalam berbagai proyek monitoring energi berbasis IoT. Pada gambar 2.7 adalah bentuk dari sensor PZEM 004T dan *Current Transformer*



Gambar 2.7 PZEM 004T & Transformator Arus
(Sumber : www.mikroavr.com)

2.2.16. 4 Channel Relay Module 30A

4 Channel Relay Module 30A adalah sebuah perangkat elektronik yang berfungsi sebagai saklar elektromekanis, memungkinkan sistem mikrokontroler untuk mengendalikan perangkat listrik bertegangan tinggi dengan arus besar secara otomatis. Modul ini terdiri atas empat buah relay yang masing-masing dapat dioperasikan secara independen untuk mengontrol

empat beban listrik terpisah. Kapasitas arus yang dapat ditangani oleh masing-masing channel mencapai hingga 30 ampere, sehingga modul ini sangat ideal digunakan dalam sistem otomasi rumah, proyek industri, serta aplikasi lain yang membutuhkan kontrol terhadap beban listrik berdaya besar.

Secara prinsip kerja, relay berfungsi sebagai saklar yang digerakkan oleh sinyal listrik bertegangan rendah (biasanya 3,3V atau 5V) yang berasal dari mikrokontroler seperti Arduino, ESP32, Raspberry Pi, atau PLC. Ketika sinyal input diterima, koil pada relay akan menjadi magnet dan menarik kontak saklar sehingga menghubungkan atau memutuskan sirkuit beban. Proses ini memungkinkan mikrokontroler yang tidak mampu mengalirkan arus besar secara langsung untuk tetap dapat mengendalikan peralatan listrik seperti pompa air, motor AC, lampu jalan, pendingin ruangan, dan peralatan elektronik lainnya.

Modul ini umumnya sudah dilengkapi dengan optocoupler atau isolator optik pada setiap channel-nya, yang berfungsi untuk memisahkan secara elektrik antara bagian kendali (mikrokontroler) dengan bagian beban (perangkat listrik). Fitur ini memberikan perlindungan tambahan terhadap lonjakan tegangan atau gangguan listrik yang dapat merusak mikrokontroler. Selain itu, biasanya juga terdapat indikator LED pada setiap channel yang menunjukkan status *ON/OFF* dari masing-masing relay, memudahkan dalam proses pemantauan dan debugging saat pengujian.

Dalam konteks sistem otomasi berbasis *Internet of Things (IoT)*, *4 Channel Relay Module 30A* memiliki peran vital sebagai penghubung antara logika digital dari mikrokontroler dan aktuator atau perangkat fisik yang memerlukan arus tinggi. Kemampuannya dalam menangani arus besar dan memberikan isolasi keamanan menjadikannya solusi andal dalam merancang sistem kontrol otomatis untuk lingkungan industri maupun rumah tangga. Pada gambar 2.8 adalah bentuk dari *4 Channel Relay Module 30A*



Gambar 2.8 4 Channel Relay Module 30 A
(Sumber : www.circuitdesigner.com)

2.2.17. Power Supply Modul HLK-PM01

HLK-PM01 adalah modul konversi daya miniatur AC ke DC yang dirancang oleh Hi-Link. Modul ini mampu mengubah tegangan AC (100–240V) menjadi tegangan DC 5V dengan arus maksimum sebesar 600mA. Modul ini sepenuhnya tertutup, memiliki desain yang kompak, dan dapat dipasang langsung ke papan rangkaian (PCB). Fungsi utama HLK-PM01 adalah memberikan catu daya stabil 5V DC dari sumber listrik PLN, yang sangat dibutuhkan dalam berbagai sistem digital dan sensor. Karena kemudahan penggunaannya dan keandalannya, HLK-PM01 sering dipakai dalam aplikasi *Internet of Things* (IoT), otomasi rumah, serta pengendalian perangkat mikrokontroler.

Dalam aplikasi sistem kendali berbasis mikrokontroler, terutama proyek-proyek yang menggunakan ESP32, Arduino, atau modul sensor digital, HLK-PM01 digunakan sebagai sumber daya utama. Modul ini biasanya dipasang secara permanen pada PCB dan langsung dihubungkan ke sumber AC PLN. Keluaran 5V-nya kemudian digunakan untuk menyuplai daya ke mikrokontroler dan perangkat pendukung lainnya.

Penggunaan HLK-PM01 perlu memperhatikan aspek keamanan, karena langsung berhubungan dengan tegangan tinggi. Perlu dilakukan isolasi fisik, pelindung casing, serta tata letak PCB yang sesuai standar keselamatan. Pada gambar 2.9 adalah bentuk dari *Power Supply Modul HLK-PM01*



Gambar 2.9 Power Supply Modul HLK-PM01
(Sumber : www.tronic.lk)

2.2.18. OLED 1.3 Inch

OLED (Organic Light Emitting Diode) 1.3 inch adalah modul tampilan mini yang digunakan untuk menampilkan informasi dalam berbagai proyek sistem tertanam, khususnya yang menggunakan mikrokontroler seperti Arduino, ESP8266, atau ESP32. Modul ini umumnya memiliki resolusi 128 x 64 piksel, dan mendukung antarmuka komunikasi I2C atau SPI, yang memungkinkannya dihubungkan dengan mudah ke berbagai sistem mikrokontroler hanya dengan dua atau empat pin. Dibandingkan dengan layar LCD konvensional, OLED memiliki keunggulan utama berupa kontras tinggi, sudut pandang lebar, hemat daya, dan tidak memerlukan *backlight*, karena tiap piksel dapat menyala sendiri (Sutrisno & Haryanto, 2022).

Dalam pengembangan sistem monitoring berbasis IoT, OLED 1.3 inch digunakan untuk menampilkan data sensor secara lokal seperti suhu, kelembaban, arus listrik, tegangan, daya, atau status koneksi Wi-Fi. Karena ukurannya yang kecil dan konsumsi daya yang rendah, OLED ini sangat cocok digunakan pada perangkat portabel atau sistem berbasis baterai. Modul OLED 1.3 inch biasanya menggunakan chip SSD1306 atau SH1106 sebagai driver tampilan, dan dapat diprogram menggunakan pustaka seperti Adafruit_SSD1306 dan U8g2 pada Arduino IDE atau PlatformIO.

Modul ini mampu menampilkan teks, simbol, bahkan grafik sederhana, dan mendukung penggunaan beberapa ukuran font serta animasi ringan. Dalam sistem berbasis ESP32, OLED

1.3 inch sering dijadikan antarmuka utama pengguna (user interface) untuk menampilkan status sensor, mode sistem, atau hasil komunikasi dengan platform cloud seperti *Firebase*. Selain itu, karena mendukung pengoperasian 3.3V atau 5V, modul ini fleksibel untuk berbagai jenis mikrokontroler tanpa perlu konversi level tegangan (Pramono & Yuliana, 2023). Secara keseluruhan, OLED 1.3 inch merupakan solusi tampilan yang efisien, ringan, dan cukup informatif untuk digunakan dalam berbagai proyek mikrokontroler, baik untuk keperluan riset, pembelajaran, maupun sistem IoT yang membutuhkan visualisasi data langsung di perangkat. Pada gambar 2.10 adalah bentuk OLED 1.3 Inch



Gambar 2.10 OLED 1.3 Inch
(Sumber : www.robotistan.com)

2.2.19. Push Button Switch

Menurut *ScienceDirect Topics*, tombol tekan (*pushbutton*) adalah mekanisme kontrol yang ketika ditekan, mengaktifkan sirkuit internal untuk mengirim sinyal ke mikrokontroler atau sistem kontrol lain. Push button switch merupakan komponen yang berfungsi untuk menghubungkan dan memutus aliran listrik melalui proses perpindahan konduktor saat terjadi interaksi pengguna. Pemutusan atau penghubungan aliran listrik terjadi ketika tombol ditekan, sedangkan pada kondisi tidak ditekan, rangkaian berada pada keadaan normal. Mekanisme kerja ini dikenal sebagai sistem *normally open* atau *normally closed*, yang memungkinkan rangkaian berada dalam kondisi tidak terkunci (*unlock*).

Secara fungsional, tombol tekan digunakan sebagai pengendali aliran listrik pada berbagai perangkat elektronik, seperti buzzer, LED, relay, dan komponen keluaran lainnya. Selain itu, push button switch berperan dalam menginisiasi kerja suatu sistem agar rangkaian

dapat aktif sesuai kebutuhan. Dalam pengembangan sistem berbasis digital, tombol tekan dapat diintegrasikan dengan perangkat lunak maupun basis data berbasis web untuk meningkatkan efektivitas pengendalian dan interaksi sistem. Pada gambar 2.11 adalah bentuk dari *Push Button Switch*



Gambar 2.11 Push Button Switch

(Sumber : www.edukasiElektronika.com)

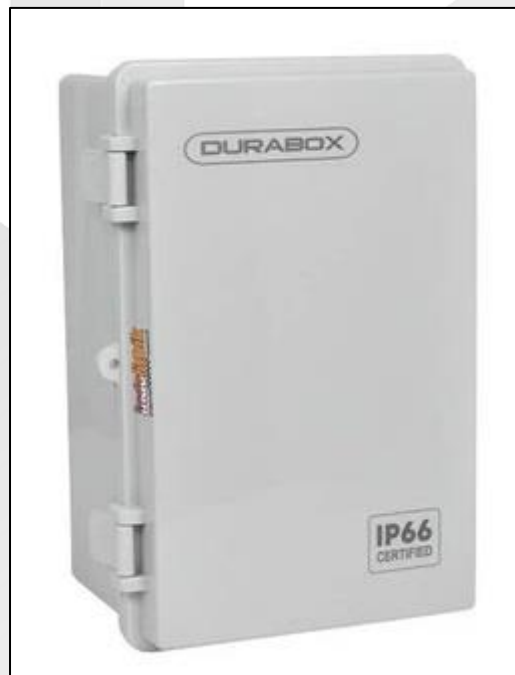
2.2.20. Box Panel (Enclosure)

Box panel merupakan komponen pendukung yang berfungsi sebagai pelindung utama perangkat keras dalam sistem monitoring listrik berbasis *Internet of Things* (IoT). Box panel digunakan untuk menempatkan dan melindungi komponen elektronik seperti mikrokontroler, sensor listrik, modul komunikasi, serta rangkaian pendukung lainnya dari pengaruh lingkungan luar yang dapat mengganggu kinerja sistem.

Pada penelitian ini digunakan box panel berbahan plastik industri (engineering plastic) berukuran 190mm x 290mm x 140mm dengan standar perlindungan IP66. Material plastik industri dipilih karena memiliki sifat isolator listrik, sehingga aman digunakan untuk aplikasi kelistrikan dan mampu mengurangi risiko korsleting akibat kontak langsung dengan bagian konduktif. Selain itu, bahan ini memiliki ketahanan yang baik terhadap korosi, berbeda dengan material logam yang berpotensi mengalami oksidasi pada lingkungan lembap. Standar Ingress Protection (IP) 66 menunjukkan bahwa box panel memiliki tingkat perlindungan yang sangat baik terhadap debu (dust-tight) dan semburan air bertekanan tinggi dari segala arah. Dengan

spesifikasi tersebut, box panel mampu melindungi perangkat elektronik dari gangguan lingkungan seperti debu, kelembapan, dan percikan air yang umum dijumpai di lingkungan perkantoran maupun ruang utilitas listrik.

Selain aspek perlindungan, box panel juga dirancang secara ergonomis dengan dilengkapi tombol kontrol dan layar indikator pada bagian depan. Keberadaan layar indikator berfungsi untuk menampilkan informasi kondisi sistem secara langsung, sedangkan tombol kontrol digunakan untuk keperluan pengoperasian atau reset sistem apabila diperlukan. Desain ini mendukung kemudahan pemantauan dan pemeliharaan sistem tanpa harus membuka panel secara langsung. Untuk bentuknya dapat dilihat pada gambar 2.12



Gambar 2.12 Box Panel
(Sumber : www.indolistrik.com)

2.2.21. Black Box Testing

Black Box Testing, atau dikenal juga sebagai *Behavioral Testing*, merupakan metode pengujian perangkat lunak yang berfokus pada pemeriksaan hasil dari input dan output tanpa perlu mengetahui struktur internal atau kode programnya. Pengujian ini dilakukan pada tahap akhir pengembangan perangkat lunak untuk memastikan bahwa sistem berfungsi dengan baik sesuai dengan yang diharapkan. Dalam pelaksanaannya, penguji tidak diwajibkan memiliki kemampuan pemrograman karena pengujian ini dapat dilakukan oleh siapa pun (Nurfauziah & Jamaliyah, 2022)

Dalam metode ini, fokus utama terletak pada input yang dimasukkan ke dalam sistem, output yang dihasilkan, serta respons sistem terhadap setiap input tertentu. Tujuan dari

pengujian ini adalah untuk memastikan bahwa sistem berjalan sesuai dengan spesifikasi yang telah ditentukan, tanpa perlu meninjau atau memeriksa detail implementasi internalnya. Hasil dari penerapan metode *black box testing* menunjukkan bahwa perangkat lunak secara keseluruhan belum sepenuhnya memenuhi persyaratan fungsional yang telah ditetapkan. Oleh karena itu, dibutuhkan pengembangan lebih lanjut terhadap sistem yang ada guna memperbaiki kekurangan tersebut. Pengembangan ini bertujuan agar sistem baru yang dihasilkan mampu meningkatkan kualitas layanan serta menyajikan informasi dengan lebih baik dan efisien dibandingkan sistem sebelumnya.(MARDIATI & SAPUTRA, 2025)

Proses penelitian diawali dengan identifikasi permasalahan, kemudian dilanjutkan dengan analisis dan perancangan sistem sebagai solusi. Pengumpulan data dilakukan melalui observasi dan wawancara. Setelah itu, sistem dirancang menggunakan Bagan Alir (*Flowchart*) atau *Data Flow Diagram* (DFD), dilengkapi dengan rancangan antarmuka. Tahapan berikutnya mencakup pembuatan basis data, tabel, dan pengolahan data sesuai kebutuhan, kemudian sistem dikembangkan secara bertahap berdasarkan rancangan yang telah dibuat. Setelah sistem selesai, dilakukan pengujian menggunakan metode *Black Box Testing* untuk memastikan fungsionalitasnya berjalan dengan baik.



Gambar 2.13 Metode Black Box Testing
(Sumber : www.qatestlab.com)

Pada gambar 2.12 tersebut menampilkan berbagai metode dalam pengujian *Black Box* (*Black Box Testing Methods*) yang digunakan untuk mengevaluasi fungsionalitas perangkat lunak tanpa mengetahui struktur internal atau kode programnya. Tujuan utama dari pendekatan ini adalah untuk memastikan bahwa sistem berfungsi sesuai dengan spesifikasi yang telah ditetapkan melalui pengujian input dan output.

Berikut adalah penjelasan dari masing-masing metode yang terdapat dalam gambar:

1. *Boundary Value Analysis* (Analisis Nilai Batas)

Metode ini berfokus pada pengujian nilai batas dari input, yaitu nilai minimum, maksimum, serta nilai di sekitar batas tersebut. Tujuannya adalah untuk memastikan bahwa sistem dapat memproses data dengan benar pada titik-titik ekstrem, karena kesalahan sering kali terjadi di sekitar nilai batas.

2. *Equivalence Class Testing* (Pengujian Kelas Ekuivalensi)

Pada metode ini, data uji dibagi ke dalam beberapa kelompok atau kelas yang dianggap ekuivalen. Jika satu nilai dalam suatu kelas menghasilkan hasil yang benar, maka nilai lain dalam kelas tersebut diasumsikan juga akan menghasilkan hasil yang sama. Teknik ini membantu mengurangi jumlah kasus uji tanpa mengurangi efektivitas pengujian.

3. *Error Guessing* (Menebak Kesalahan)

Metode ini mengandalkan pengalaman dan intuisi penguji dalam menebak area atau bagian program yang kemungkinan besar mengandung kesalahan. Penguji biasanya membuat skenario uji berdasarkan kesalahan umum yang sering terjadi pada perangkat lunak serupa.

4. *Decision Table Testing* (Pengujian Tabel Keputusan)

Teknik ini digunakan untuk menguji sistem yang memiliki banyak kombinasi kondisi dan tindakan. Tabel keputusan menggambarkan berbagai kondisi input beserta hasil output yang diharapkan, sehingga penguji dapat memastikan semua kemungkinan kombinasi telah diuji.

5. *Graph-Based Testing* (Pengujian Berbasis Grafik)

Dalam metode ini, hubungan antar komponen atau alur data digambarkan dalam bentuk grafik. Setiap simpul (node) merepresentasikan status atau kejadian, sedangkan setiap sisi (edge) menunjukkan transisi antar kejadian. Tujuannya adalah untuk memastikan bahwa semua jalur dan hubungan dalam sistem berfungsi sebagaimana mestinya.

2.2.22. *Quality of Service*

Quality of Service (QoS) merupakan suatu metode yang digunakan untuk mengukur, mengelola, serta meningkatkan kualitas layanan yang diberikan oleh suatu jaringan atau sistem komunikasi. Melalui QoS, administrator jaringan dapat mengatur prioritas terhadap jenis lalu lintas tertentu, sehingga performa jaringan dapat dioptimalkan. Selain itu, QoS juga berfungsi

untuk menentukan atribut layanan yang mencakup aspek kualitas maupun kuantitas dari layanan yang disediakan (Irfan et al., 2025).

Quality of Service (QoS) adalah teknologi jaringan yang digunakan untuk mengatur dan mengendalikan lalu lintas data agar layanan yang bersifat penting atau kritis mendapatkan prioritas lebih tinggi dibandingkan jenis lalu lintas lainnya. Dalam sebuah jaringan, berbagai jenis data seperti panggilan VoIP, streaming video, permainan daring, hingga transfer file dapat berlangsung secara bersamaan. Tanpa adanya penerapan QoS, semua jenis lalu lintas akan diperlakukan sama, yang dapat menyebabkan penurunan kualitas layanan, misalnya *buffering* pada video atau *delay* pada komunikasi suara. Oleh karena itu, QoS menerapkan berbagai mekanisme seperti pengelolaan *bandwidth*, pengurangan *latensi*, pengendalian *jitter*, dan pencegahan kehilangan paket (*packet loss*) guna meningkatkan kinerja dan stabilitas jaringan secara keseluruhan. (Ardiansyah et al., 2025)