

BAB 2

TINJAUAN PUSTAKA

2.1 Landasan Teori

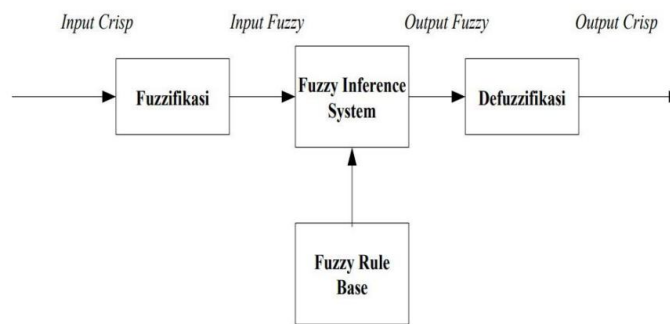
2.1.1 *Internet of Things(IoT)*

Internet untuk Segala(-nya) (bahasa Inggris: *Internet of Things*, atau dikenal juga dengan singkatan IoT) merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. *Internet of Things* berkaitan erat dengan *Machine-to-Machine* (M2M) atau lebih dikenal dengan perangkat cerdas.

Pada dasarnya, *Internet of Things* mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis Internet. Istilah *Internet of Things* awalnya disarankan oleh Kevin Ashton pada tahun 1999 dan mulai terkenal melalui *Auto-ID Center* di MIT.

2.1.2 *Logika fuzzy Mamdani*

Internet Logika *fuzzy* Mamdani adalah cabang logika yang memperkenalkan konsep derajat keanggotaan dalam himpunan, sehingga memungkinkan adanya keanggotaan yang bersifat salah atau benar (Sistem *Fuzzy*, 2019). Struktur utama sistem logika *fuzzy* Mamdani dijelaskan dalam gambar 2.1. Dalam *inferensi fuzzy*, penalaran dapat dilakukan dengan memanfaatkan input atau masukan serta aturan *fuzzy* yang telah ditetapkan untuk hasil keluaran dari sistem *fuzzy* tersebut (Suprpto & Simanjuntak, 2020). Serta, menawarkan alternatif yang lebih fleksibel dibandingkan dengan logika *boolean* memberikan nilai benar atau salah secara mutlak, tanpa derajat keanggotaan, sedangkan logika *fuzzy* memungkinkan adanya derajat keanggotaan dengan nilai yang berada di antara 0 dan 1(Sistem *Fuzzy*, 2019).



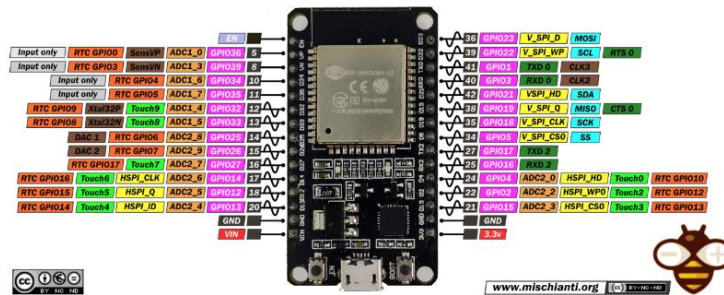
Gambar 2.1 Struktur Utama Logika Fuzzy Mamdani

Pada Gambar 2.1 proses dimulai dari tahap fuzzifikasi, yaitu konversi input bersifat pasti (*crisp*) seperti jarak dalam cm atau kecepatan dalam m/s ke dalam nilai *fuzzy* berdasarkan fungsi keanggotaan segitiga. Nilai-nilai ini dinormalisasi ke dalam skala $[0-1]$ sesuai dengan derajat keanggotaannya dalam himpunan *fuzzy*, seperti Dekat, Sedang, atau Jauh. Selanjutnya, nilai *fuzzy* tersebut diproses oleh *Fuzzy Inference System*, yang bekerja berdasarkan seperangkat aturan yang telah ditentukan dalam *fuzzy rule base*. Pada tahap ini, sistem mengevaluasi setiap aturan berdasarkan input fuzzy yang diterima, lalu menghasilkan output *fuzzy* sesuai dengan hasil evaluasi.

Hasil dari inferensi ini kemudian masuk ke tahap *defuzzifikasi*, yaitu proses konversi *output fuzzy* menjadi output *crisp* yang dapat digunakan secara langsung oleh sistem fisik, seperti mengatur tingkat kecerahan lampu. Teknik *defuzzifikasi* yang umum digunakan adalah metode centroid (titik tengah), yang menghitung rata-rata tertimbang dari semua output *fuzzy* yang aktif. Dengan tahapan-tahapan ini, sistem *fuzzy* Mamdani mampu mengambil keputusan secara cerdas dan adaptif terhadap kondisi yang berubah-ubah di lingkungan sekitar.

2.1.3 Mikrocontroller ESP32

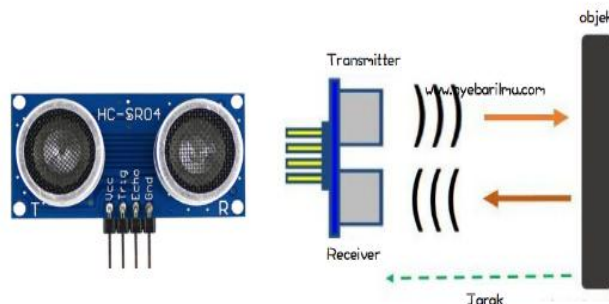
ESP32 adalah mikrokontroler yang dikenalkan oleh *Espressif System* merupakan penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul WiFi dalam *chip* sehingga sangat mendukung untuk membuat sistem aplikasi Internet of Things. ESP32 sendiri tidak jauh berbeda dengan ESP8266 yang familiar di pasaran, hanya saja ESP32 lebih kompleks dibandingkan ESP8266, cocok untuk sobat dengan proyek yang besar.



Gambar 2.2 Pinout Dari ESP 32 Dev Kit V1

2.1.4 Sensor Ultrasonik

Sensor ultrasonik adalah alat yang beroperasi dengan memanfaatkan pantulan gelombang suara untuk mengidentifikasi keberadaan suatu *object* tertentu yang berada di depannya. Sensor ini menggunakan gelombang ultrasonik dengan frekuensi lebih dari 20.000 Hz untuk mengukur jarak dan waktu tertentu. Selain mengukur jarak, sensor ultrasonik juga memiliki kemampuan untuk mendeteksi keretakan dan jenis benda yang mampu memantulkan sinyal.

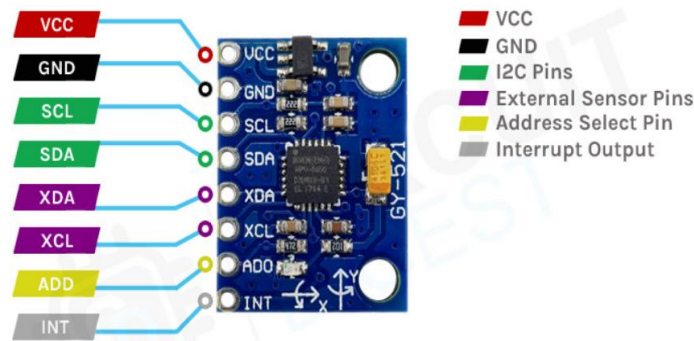


Gambar 2.3 Prinsip Kerja Sensor Ultrasonik

2.1.5 Sensor Accelerometer

Accelerometer adalah perangkat yang mengukur percepatan yang tepat dari sebuah objek. *Accelerometer* yang tepat adalah percepatan (laju perubahan kecepatan) dari objek relatif terhadap pengamat yang jatuh bebas (yaitu, relatif terhadap kerangka acuan inersia). *Accelerometer* yang tepat berbeda dari percepatan koordinat, yang merupakan percepatan terhadap sistem koordinat tertentu, yang mungkin atau mungkin tidak mengalami percepatan. Misalnya, *accelerometer* yang diam di permukaan Bumi akan mengukur percepatan karena gravitasi Bumi lurus ke atas sekitar $g \approx 9,81 \text{ m/s}^2$. Dalam penelitian ini,

accelerometer pada modul *accelerometer* (MPU5060) dimanfaatkan untuk mendeteksi perubahan orientasi atau kemiringan tiang lampu berdasarkan perubahan nilai percepatan pada sumbu X, Y, dan Z. Data tersebut kemudian diolah menjadi nilai sudut kemiringan yang digunakan sebagai dasar penentuan status di Bot Telegram.

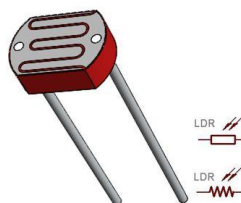


Gambar 2.4 Pinout Modul MPU6050

2.1.6 Sensor LDR

Protokol Sensor *light dependent resistor* (LDR) adalah jenis resistor yang nilai resistansinya dapat berubah ketika ada pengaruh cahaya dari luar. Karena *light dependent resistor* termasuk kedalam jenis resistor, maka *light dependent resistor* memiliki nilai tahanan. Besar kecilnya nilai hambatan pada sensor *light dependent resistor* ini tergantung pada intensitas cahaya yang diterima oleh sensor LDR tersebut.

Ketika cahaya sekitar dalam keadaan gelap maka LDR akan menangkap respon dan memberikan nilai resistansi yang besar. Sebaliknya, saat cahaya sekitar dalam kondisi terang maka sensor LDR akan memberikan nilai resistansi yang rendah. Nilai hambatan dari *light dependent resistor* dapat berubah tergantung dari besar kecilnya intensitas cahaya sekitar

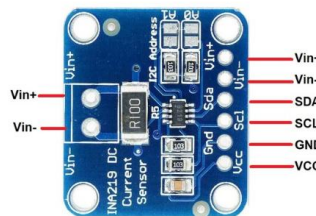


Gambar 2.5 Pinout Sensor LDR

2.1.7 Sensor INA219

Sensor INA219 modul sensor yang digunakan untuk memonitor tegangan dan arus pada suatu rangkaian listrik. Modul ini dilengkapi dengan antarmuka I2C atau SMBUS-*COMPATIBLE*, memungkinkannya untuk memonitor tegangan shunt dan pasokan tegangan bus. Kemampuan konversi waktu dan penyaringan juga dimiliki oleh INA219. Dengan amplifier input yang memiliki rentang maksimum $\pm 320\text{mV}$, modul ini dapat mengukur arus hingga $\pm 3,2\text{A}$. Resolusi ADC internal sebesar 12 bit, sehingga resolusi untuk rentang arus 3.2A adalah 0,8 mA. Dengan gain internal yang diatur pada minimum div8, arus maksimumnya mencapai $\pm 400\text{mA}$ dengan resolusi 0,1 mA. INA219 dapat mengidentifikasi tegangan shunt pada rentang 0 hingga 26 V.

Cara kerja sensor ini adalah dengan mendeteksi tegangan bus dan tegangan shunt secara bergantian, lalu mengonversinya secara langsung menjadi sinyal digital menggunakan ADC. Sinyal digital tersebut kemudian dikirimkan ke register arus dan register tegangan. Untuk memperoleh nilai daya, sensor ini melakukan perkalian antara nilai tegangan dan arus yang diperoleh dari masing-masing register. Berikut adalah gambar Sensor INA 219 dan gambar *Pin Out* Sensor INA 219 dapat dilihat pada Gambar 2.6 dibawah ini.



Gambar 2.6 Pinout Modul INA219

2.1.8 Node-RED

Node-RED adalah alat bantu pemrograman yang dapat menghubungkan perangkat keras, API, dan layanan online dengan cara yang sangat mudah dan menarik (Fajrur, 2020). Node-RED menggunakan bahasa pemrograman visual, di mana program dibuat sebagai *flow* dari *node-node* yang saling berhubungan. Node-RED terdiri dari runtime berbasis Node.js yang dapat diakses melalui editor berbasis *browser*. Node-RED menyediakan berbagai macam *node* yang dapat digunakan untuk membuat *flow*, dan dapat dengan mudah diperluas dengan

menginstal node baru yang dibuat oleh komunitas. *Flow* yang dibuat dalam Node-RED disimpan dalam bentuk JSON, yang dapat dengan mudah diimpor dan diekspor untuk dibagikan dengan orang lain (OpenJS Foundation, n.d.).

Node-RED dapat digunakan untuk berbagai macam jenis aplikasi, termasuk *Internet of Things* (IoT) (TeknoJurnal & Intel Developer Zone, 2016). Node-RED dapat menghimpun data dari berbagai sensor *node*, seperti sensor yang terpasang bersama mikrokontroler serta modul komunikasi jaringan (WiFi atau Ethernet). Node-RED juga dapat mempresentasikan data-data yang dikumpulkan dalam berbagai data visual, seperti bagan garis (*line chart*) dan meteran (*gauge*) (Fajrur, 2020).

2.1.9 *Api Bot Telegram*

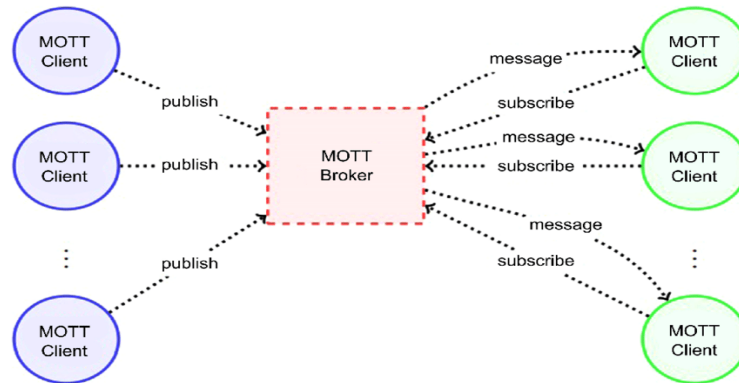
Bot Telegram berkomunikasi dengan server Telegram melalui *request* HTTP. API Bot Telegram adalah salah satu bentuk spesifikasi dari *interface* tersebut, contohnya sebuah daftar panjang berbagai *method* dan tipe data (*data type*), yang biasa disebut dengan referensi atau reference. API ini mendefinisikan aksi apa saja yang bisa dilakukan oleh bot Telegram. Kamu dapat menemukannya di tab Sumber Daya (navigasi atas halaman ini) bagian Telegram.

Singkatnya, ketika bot mengirim pesan, pesan tersebut akan dikirim sebagai request HTTP ke server API Bot. Server tersebut di-*hosting* di api.telegram.org. Server akan menerjemahkan request tadi menjadi protokol utama Telegram yang disebut MTProto, lalu meneruskannya ke backend Telegram yang bertugas mengirim pesan ke pengguna yang dituju.

2.1.10 *Protokol MQTT*

Protokol MQTT adalah protokol komunikasi antar-mesin yang digunakan untuk mengirimkan data dan pesan dalam waktu yang cepat. Protokol ini menggunakan sistem '*publish-subscribe*'. Menurut Aloufi dan Alhazmi (2020), pada Gambar 2.7, MQTT terdiri dari *client* dan *broker*. Bagian tengahnya adalah broker, yang mengelola pesan dan transaksi antar *client*. Mesin yang mengirim informasi disebut *publisher*, dan mesin yang menerima informasi disebut *subscriber*. *Publisher* akan mengirim data melalui broker sebagai perantara penyedia pesan, lalu *subscriber* akan dapat menerima data yang sesuai dari *broker*. Untuk menentukan data yang dikirim dan diterima, data diberi nama yang disebut

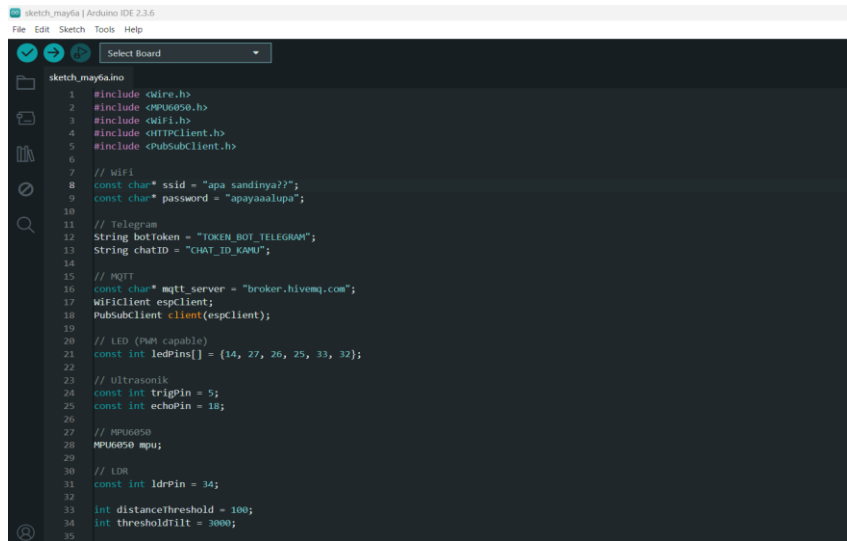
‘*topic*’. Semua perangkat dapat mengirim dan menerima data dengan topik yang sama, dan satu perangkat juga dapat mengirim dan menerima data dengan topik yang berbeda-beda.



Gambar 2.7 Diagram Model Protokol MQTT

2.1.11 Arduino IDE

Arduino IDE merupakan salah satu perangkat lunak yang digunakan sebagai pengedit teks serta mempermudah proses pemrograman beberapa jenis mikrokontroler yang ada, di antaranya adalah Arduino IDE dan NodeMCU ESP8266. Perangkat lunak tersebut bisa dipasangkan pada *board mikrokontroller* untuk mempermudah proses mengupload dan komunikasi dengan mikrokontroller. Proses pemrograman mikrokontroller menggunakan Arduino IDE, dapat menggunakan bahasa pemrograman C atau C++ (Rosyady, Ikhsan, & Feter, 2022). Gambar 2.8 adalah jendela dari Arduino IDE beserta fitur yang ada di dalamnya seperti *Sketchbook*, *Board Manager*, *Library Manager*, *Debugger*, dan *Search*. Selain itu ada fitur *Serial Monitor* untuk melakukan komunikasi antara Arduino IDE dan mikrokontrol.



Gambar 2.8 Tampilan Jendela Arduino IDE

2.1.12 MongoDB

MongoDB menjadi salah satu jenis basis data *NoSQL*, dimana ini merupakan konsep dari penyimpanan non-relational data. Terdapat empat cara penyimpanan yang digunakan oleh jenis basis data ini, yaitu *key-value*, *big table*, *document-oriented*, dan *graph*. Cara penyimpanan yang digunakan oleh MongoDB adalah *document-oriented* dalam bentuk dokumen yang mirip dengan *JavaScript Object Notation* (JSON) (Tiwari, 2011).

Mongoddb adalah sebuah basis data *open source* yang dibuat dengan bahasa pemrograman C++. MongoDB dikembangkan oleh 10gen sejak bulan Oktober 2007, dan digunakan oleh khalayak umum sejak bulan Februari 2009. Sejak penelitian ini dilakukan, MongoDB sudah berada pada versi 5.0.7 dan ada pada berbagai macam sistem operasi seperti windows, ubuntu, Redhat, Debian dan berbagai sistem operasi lain.

Basis dari mongoddb adalah dokumen (*Document Oriented Database*) dan termasuk sistem basis data yang menganut paham *NoSQL*. *NoSQL* adalah singkatan dari *Not Only SQL*, ini berarti mongo adalah sebuah basis data yang tidak harus menggunakan perintah *SQL* untuk melakukan proses manipulasi data.

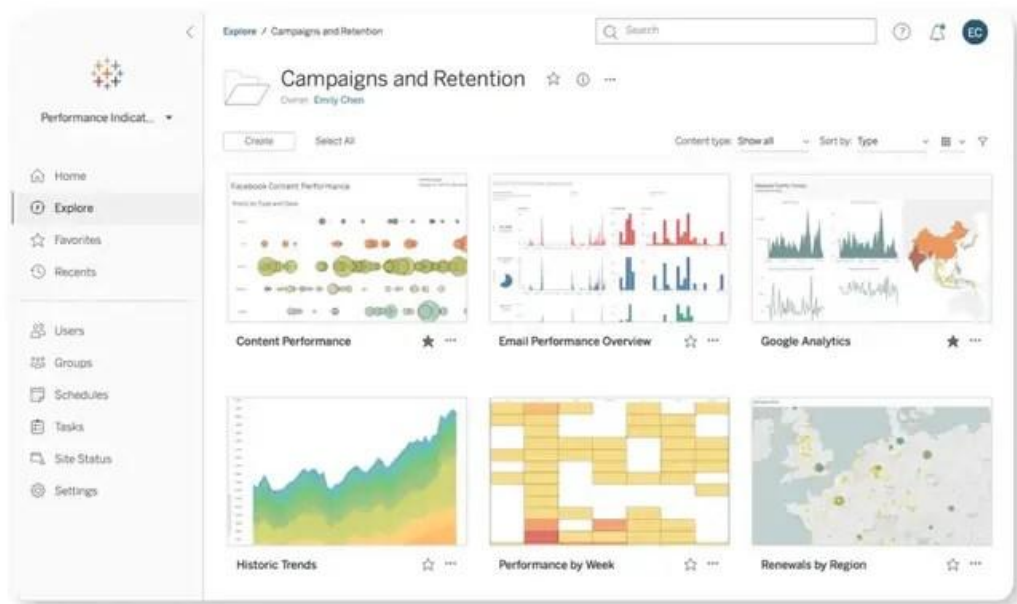
Mongoddb tidak menggunakan istilah tabel, baris dan kolom. Format penulisan pada mongoddb menggunakan BSON(*Binary JSON*) format ini persis dengan format JSON namun perbedaanya BSON mendukung tipe data yang sangat

luas misalnya tipe data untuk *date*, menyimpan *script javascript*, *regex (regular expression)* dan lain sebagainya.

2.1.13 Software Tableau

Tableau adalah perangkat lunak intelijen bisnis yang mudah digunakan, terutama untuk visualisasi data, analisis data, dan pelaporan. Sangat mudah digunakan karena menggunakan sistem *drag and drop*. *Tableau* dapat menggabungkan data dari berbagai sumber data seperti *spreadsheet*, *database*, data *cloud*, dan data besar ke dalam satu program untuk digunakan dalam analisis dinamis (Purwani et al., 2021).

Tableau tersedia dalam versi publik maupun *desktop*. Versi publik berarti setiap orang dapat melihat dan mengakses hasil visualisasi data. Untuk *Tableau* versi *desktop* hasil visualisasi data yang digunakan dapat diatur kata sandi sehingga cocok untuk menjalankan bisnis dan perusahaan.



Gambar 2.9 Tampilan Software Tableau

2.1.14 Usability

Usability (kebergunaan) adalah ukuran seberapa mudah suatu sistem digunakan untuk mencapai tujuan penggunaannya secara efektif, efisien, dan memuaskan (ISO 9241-11). Tiga komponen utamanya adalah *effectiveness* (keberhasilan pengguna mencapai tujuan), *efficiency* (usaha yang dibutuhkan untuk

mencapai tujuan tersebut), dan *satisfaction* (kenyamanan pengguna saat menggunakan sistem).

Pengujian usability pada penelitian ini menggunakan *USE Questionnaire (Usefulness, Satisfaction, and Ease of Use)* yang dikembangkan oleh Lund (2001). Instrumen ini mengukur empat dimensi: *usefulness* (manfaat sistem bagi pengguna), *ease of use* (kemudahan pengoperasian), *ease of learning* (kemudahan dipelajari), dan *satisfaction* (kepuasan pengguna). *USE Questionnaire* dipilih karena bersifat umum dan dapat diterapkan pada berbagai jenis sistem, termasuk perangkat IoT seperti alat monitoring dan kontrol LPJU pada penelitian ini.

Pengukuran menggunakan Skala *Likert* 5 tingkat: Sangat Setuju (5), Setuju (4), Netral (3), Tidak Setuju (2), Sangat Tidak Setuju (1). Skor dihitung dengan rumus:

$$Skor = T \times Pn$$

Dengan T = jumlah responden yang memilih suatu jawaban, Pn = bobot skor jawaban tersebut. Hasilnya dikonversi ke persentase dengan rumus:

$$Index(\%) = \frac{Total\ Skor}{Skor\ Maksimum} \times 100\%$$

Dengan Skor Maksimum = skor tertinggi (5) × jumlah responden. Hasil persentase diinterpretasikan ke kategori: 0–20% Sangat Kurang Baik, 21–40% Kurang Baik, 41–60% Cukup, 61–80% Baik, dan 81–100% Sangat Baik.

2.2 Penelitian Terdahulu

Sistem PJU Cerdas berbasis Arduino dan NodeMCU yang terhubung dengan IoT dikembangkan oleh Hidayat dkk. (2024). Dengan desain fokus pada kontrol pencahayaan otomatis mengikuti waktu dan kondisi lingkungan sekitar, sekaligus memungkinkan pemantauan dari jarak jauh. Dengan pendekatan ini, penelitian ini bertujuan untuk meningkatkan efisiensi energi penggunaan penerangan dan meningkatkan kenyamanan dan keselamatan pengguna jalan pada malam hari.

Berbeda dengan hal tersebut, Setiawan dkk. (2023) mengembangkan sistem kendali otomatis lampu jalan berbasis ESP32 dan sensor LDR yang menggunakan tingkat cahaya di sekitarnya sebagai acuan untuk menyalakan atau mematikan

lampu secara otomatis. Karena terhubung dengan IoT, pengguna dapat memantau dan mengoperasikan sistem secara *real-time* hal inilah yang meningkatkan efisiensi dan efektivitas operasionalnya.

Pendekatan lain dikemukakan oleh Lubis (2021) melalui miniatur lampu jalan otomatis berbasis sensor ultrasonik dan Arduino Mega lampu akan menyala jika sensor mendeteksi *object* atau kendaraan yang melintas. Prinsip kerja ini memungkinkan konsumsi energi hanya terjadi saat dibutuhkan, atau berdasarkan jadwal maupun tingkat pencahayaan.

Penelitian yang dilakukan oleh Rahmawati & Iqbal, (2023) menghadirkan pendekatan berbeda dengan mengembangkan *Smart Lamp Alarm* berbasis sensor *accelerometer* dan ESP32, yang dapat mendeteksi getaran akibat gempa bumi. Sistem ini memberikan peringatan dini melalui nyala lampu dan notifikasi, meningkatkan aspek keselamatan publik dalam menghadapi bencana alam. Pendekatan ini memperluas fungsi sistem pencahayaan menjadi alat mitigasi risiko.

Hasil penelitian ini sejalan dengan studi oleh Luzentut dkk. (2020), yang merancang dan mengembangkan pengendali sistem pencahayaan berbasis mikrokontroler dengan logika *fuzzy*. Sistem ini menggunakan sensor LDR untuk mendeteksi intensitas cahaya di lingkungan sekitar, kemudian menerapkan logika *fuzzy* untuk menentukan tingkat intensitas cahaya lampu. Tujuan utama penelitian ini adalah mencapai penghematan energi melalui pengendalian pencahayaan otomatis yang menyesuaikan dengan kondisi cahaya. Hasil studi menunjukkan bahwa sistem tersebut mampu mengatur pencahayaan lampu secara efektif dan efisien untuk mengurangi konsumsi daya.

Tabel 2.1 Ringkasan penelitian terdahulu

Kategori	Peneliti 1	Peneliti 2	Peneliti 3	Peneliti 4	Peneliti 5
Penulis	Hidayat dkk. (2024)	Setiawan dkk. (2023)	Lubis (2021)	Rahmawati & Iqbal (2023)	Lumentut dkk. (2020)
Tahun	2024	2023	2021	2023	2020

Kategori	Peneliti 1	Peneliti 2	Peneliti 3	Peneliti 4	Peneliti 5
Judul	Pengembangan Sistem Penerangan Jalan Pintar Berbasis IoT dengan Arduino dan NodeMCU	PERANCANGAN SISTEM KENDALI OTOMATIS LAMPU JALAN BERBASIS INTERNET OF THINGS. INFOTECH	Miniatur Rancang Bangun Penerangan Lampu Jalan Otomatis pada Malam Hari Menggunakan Sensor Ultrasonik Berbasis Arduino Mega	Smart Lamp Alarm Pendeteksi Gempa Bumi dengan Accelerometer Berbasis IoT	Rancang Bangun Alat Pengontrol Tingkat Pencahayaannya Lampu Berbasis Mikrokontroler dengan Menggunakan Logika <i>Fuzzy</i>
Teknik/ Metode	Arduino, NodeMCU, IoT	ESP32, Sensor LDR, IoT	Sensor Ultrasonik, Arduino Mega	Sensor Accelerometer, ESP32, IoT	Mikrokontroler, Sensor LDR, Logika <i>Fuzzy</i>
Hasil Utama	Sistem berhasil dikendalikan melalui internet dan menyesuaikan pencahayaan secara otomatis	Sistem mampu menyalakan dan mematikan lampu otomatis berbasis intensitas cahaya	Sistem menyala hanya saat objek melintas, meningkatkan efisiensi daya	Sistem mampu mendeteksi getaran dan memberi alarm secara real-time	Sistem berhasil menyesuaikan tingkat pencahayaan secara otomatis sesuai kondisi cahaya sekitar

Kategori	Peneliti 1	Peneliti 2	Peneliti 3	Peneliti 4	Peneliti 5
Relevansi dengan Penelitian	Meningkatkan efisiensi energi dan otomatisasi penerangan jalan	Mengotomatiskan lampu jalan sesuai kondisi cahaya sekitar	Menyalakan lampu saat terdeteksi kendaraan	Deteksi dini gempa dan peringatan visual melalui lampu pintar	Mengontrol pencahayaan lampu secara otomatis berbasis intensitas cahaya untuk efisiensi energi

Berdasarkan kajian terhadap beberapa penelitian terdahulu pada Tabel 2.1, diketahui bahwa banyak studi telah membahas sistem penerangan jalan pintar berbasis *Internet of Things* (IoT) menggunakan berbagai pendekatan, seperti menggunakan sensor LDR yang pada penelitian sebelumnya untuk mendeteksi intensitas cahaya, sensor ultrasonik untuk mendeteksi keberadaan *object*, dan menggunakan prosesor ESP32 atau NodeMCU digunakan untuk pemantauan dasarnya. Selain itu, terdapat penelitian yang juga memanfaatkan sensor *accelerometer* untuk mendeteksi getaran apabila terjadi gempa dan menerapkan logika *fuzzy* untuk pengaturan pencahayaan secara otomatis. Namun, sebagian besar penelitian ini masih berfokus pada satu atau dua aspek spesifik dan belum mengintegrasikan berbagai fitur pemantauan dan keamanan menyeluruh ke dalam satu sistem.

Penelitian ini memiliki perbedaan dengan penelitian sebelumnya karena mengintegrasikan beberapa sensor sekaligus diantaranya sensor LDR, sensor ultrasonik, *accelerometer*, dan sensor konsumsi daya INA21 ke dalam satu sistem penerangan jalan pintar berbasis ESP32. Sistem ini dirancang tidak hanya untuk secara otomatis menyesuaikan pencahayaan berdasarkan kondisi lingkungan dan keberadaan *object*, tetapi juga untuk mendeteksi gangguan atau kemiringan tiang menggunakan *accelerometer* dan memantau data konsumsi daya secara *real-time*. Selain itu, penelitian ini juga menggunakan Node-RED sebagai *dashboard* visualisasi pemantauan untuk menampilkan kondisi lampu, data sensor, serta konsumsi daya secara *real-time* dan mengirimkan notifikasi melalui Telegram.

Oleh karena itu, penelitian ini diharapkan dapat berkontribusi pada peningkatan efisiensi energi, mendukung pemantauan penerangan jalan secara *real-time*, meningkatkan keamanan, dan mencerminkan *Internet of Things* (IoT) dalam sistem penerangan jalan.