

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

Kajian teoritis dalam penelitian ini membuat berbagai teori yang relevan dan mendukung, yang diambil dari sumber-sumber terpercaya seperti buku. Teori-teori tersebut dijadikan acuan utama dalam pelaksanaan penelitian, antara lain:

1. *Charging Station*

Charging station menjadi sebuah fasilitas atau perangkat yang dirancang untuk menyediakan daya listrik guna mengisi ulang baterai perangkat elektronik, seperti *smartphone*, tablet, dan gadget lainnya. Stasiun pengisian daya biasanya terdiri dari beberapa port pengisian daya yang memungkinkan pengguna untuk mengisi daya lebih dari satu perangkat secara bersamaan. Dalam konteks penggunaan publik, *charging station* banyak dijumpai di tempat-tempat umum seperti bandara, pusat perbelanjaan, kampus, kafe, dan ruang tunggu, untuk memberikan kenyamanan kepada pengguna dalam menjaga daya perangkatnya tetap optimal.

Menurut Shidqi et al., (2021) *Charging station* adalah sarana umum yang disediakan untuk digunakan secara bersama dan menjadi hak setiap anggota masyarakat, fasilitas ini memiliki banyak *port* yang memungkinkan

beberapa perangkat mengisi daya secara bersamaan tanpa menurunkan kecepatan pengisian. Seiring perkembangan teknologi, *charging station* kini tidak hanya berfungsi sebagai tempat pengisian daya, namun juga dapat dilengkapi dengan sistem monitoring berbasis *Internet of Things* (IoT) untuk menghubungkan konsumsi daya, status pengisian, serta sistem pembayaran otomatis secara akurat. Dengan adanya inovasi berbasis *Internet of Things* (IoT), stasiun pengisian daya mampu memberikan informasi yang lebih akurat dan bermanfaat bagi pengelola dan pengguna, seperti penggunaan energi (kWh), durasi pengisian, serta kontrol perangkat dari jarak jauh. Hal ini menjadikan stasiun pengisian daya sebagai bagian penting dalam mendukung mobilitas masyarakat modern yang sangat bergantung pada perangkat elektronik portabel.

2. *Smartphone*

Smartphone merupakan evolusi dari telepon genggam yang sebelumnya hanya berfungsi sebagai alat komunikasi. Kini, perangkat ini telah berkembang menjadi alat serbaguna yang mampu mendukung berbagai aktivitas manusia dalam kehidupan sehari-hari (Saputra, 2023). *Smartphone* juga disebut sebagai perangkat komunikasi modern yang sangat canggih karena mendukung penggunaan media sosial, pengiriman pesan teks, serta panggilan telepon. Selain itu, *smartphone* juga dilengkapi berbagai fitur unggulan dan menarik, seperti kamera beresolusi tinggi dan akses internet.

Smartphone menjadi perangkat genggam yang memiliki fungsi dan kapabilitas layaknya sebuah komputer. Meskipun belum terdapat standar resmi dari pabrikan untuk mendefinisikan *smartphone*, sebagian besar *smartphone* menggunakan sistem operasi penuh yang memungkinkan pengembang menjalankan berbagai aplikasi dasar. Bagi sebagian orang, *smartphone* diartikan sebagai ponsel dengan fitur-fitur canggih seperti email, akses internet, kemampuan membaca *e-book*, serta dilengkapi papan ketik fisik atau virtual. Secara sederhana, *smartphone* dapat dianggap sebagai komputer mini yang juga berfungsi sebagai telepon (Sipaayung & Munawaroh, 2024). Selain itu, *smartphone* juga memiliki kemampuan integrasi dengan berbagai perangkat digital lainnya melalui koneksi seperti *Bluetooth*, *Wi-Fi*, dan *NFC*, serta mendukung berbagai aplikasi berbasis *cloud* yang memungkinkan sinkronisasi data. Hal ini menjadikan *smartphone* tidak hanya sebagai alat komunikasi, tetapi juga sebagai pusat kendali aktivitas digital sehari-hari, mulai dari pekerjaan, hiburan, hingga manajemen informasi pribadi.

3. Internet of Things (IoT)

Internet of Things (IoT) adalah sistem yang menghubungkan berbagai objek dan perangkat, yang memungkinkan mereka untuk berkomunikasi dan terintegrasi satu sama lain melalui jaringan internet, dan memungkinkan pertukaran data otomatis dan koordinasi antar perangkat (Mambang, 2021). Sederhananya, perangkat *Internet of Things* (IoT) dapat dikontrol dari jarak jauh tanpa memerlukan manusia untuk

berinteraksi langsung. Penyebaran Internet telah memungkinkan teknologi kendali jarak jauh, yang memudahkan kita melakukan berbagai aktivitas tanpa terikat oleh batasan ruang dan waktu (Budiyanti, 2021).

Internet of Things (IoT) memiliki berbagai data yang dikumpulkan dari perangkat yang terhubung internet diproses menjadi aliran data besar dalam ekosistem *Internet of Things* (IoT), sebagai pelaku utama *Internet of Things* (IoT) dan *cloud* saling berintegrasi dalam mengelola, menyimpan, dan menganalisis data yang dihasilkan oleh perangkat *Internet of Things* (IoT) (Wibowo, 2023). Infrastruktur jaringan merupakan elemen kunci *Internet of Things* (IoT) dan mencakup jaringan komunikasi kabel dan nirkabel, serta infrastruktur *cloud* dan sistem *back-end* yang memiliki kemampuan untuk menghubungkan, menyimpan, mengelola, dan menganalisis data dari perangkat *Internet of Things* (IoT) (Rizal et al., 2023). Sehingga keputusan dan tindakan yang tepat dan tepat waktu dapat diambil berdasarkan analisis data *Internet of Things* (IoT) tersebut.

Dalam konsep rumah pintar, *Internet of Things* (IoT) menawarkan manfaat besar bagi penghuninya. Mereka dapat mengontrol alat rumah tangga menggunakan *smartphone* untuk memantau dan mengontrol parameter penting seperti kegunaan listrik. Adanya *Internet of Things* (IoT) konsumsi listrik ini akan membantu masyarakat memantau penggunaan energi secara lebih rinci dan *real-time*. Masyarakat dapat mengidentifikasi perangkat mana yang mengonsumsi daya paling banyak

dan mengambil tindakan yang diperlukan jika terjadi lonjakan penggunaan. Hal ini tidak hanya meningkatkan kenyamanan, tetapi juga menghemat energi dan menurunkan pengeluaran tagihan listrik. Dengan begitu, pengguna rumah tangga dapat melakukan aktivitas sehari-hari dengan lebih mudah.

4. ESP32

ESP32 merupakan mikrokontroler generasi lanjutan yang dikembangkan oleh *Expressif Systems* dan dirilis pada 6 September 2016, sebagai penerus ESP8266 dengan fitur yang lebih mumpuni dan tangguh (Mutiara et al., 2020). ESP32 memiliki modul *Wi-Fi* dan *Bluetooth* terintegrasi, yang memungkinkannya berkomunikasi secara nirkabel dan terhubung dengan mudah ke jaringan *Wi-Fi* dan perangkat *Bluetooth* lainnya (Fauzi et al., 2023).



Gambar 2.1 ESP32

Sumber : (Hoddie et al., 2020)

Dilengkapi dengan konektivitas yang handal, ESP32 dilengkapi dengan prosesor *dual-core* dan berbagai mode tidur. Fitur-fitur ini memungkinkan ESP32 beroperasi dengan konsumsi daya yang efisien,

menjadikannya solusi ideal untuk aplikasi *Internet of Things* (IoT) yang membutuhkan daya tahan baterai yang lama. ESP32 memiliki rangkaian antarmuka peripheral yang lengkap, termasuk SPI, I2C, UART, dan GPIO, yang memungkinkan koneksi fleksibel ke berbagai perangkat dan sensor. Selain itu, ESP32 dapat diprogram melalui berbagai *platform*, termasuk Arduino IDE, PlatformIO, dan Espressif IDF, sehingga menjadikannya alat yang mudah digunakan bagi para pengembang (Nur, 2023).

Dengan fitur-fitur ESP32 yang luar biasa telah menjadikannya pilihan populer untuk berbagai aplikasi, termasuk *Internet of Things* (IoT), kontrol perangkat, otomatisasi, proyek elektronik hobi, dan lainnya. Dengan kemampuan komunikasi nirkabel yang andal dan kinerja tinggi, ini adalah *platform* ideal untuk desain yang fleksibel dan tangguh untuk memenuhi beragam kebutuhan proyek elektronik masa kini.

5. Sensor PZEM-004T

Menurut Kaci et al., (2024) PZEM-004T merupakan modul pemantauan dan pengukuran energi yang dirancang untuk penggunaan *plug-and-play* dalam instalasi listrik rumah. Modul ini dapat digunakan untuk mengukur tegangan, arus, dan daya aktif, yang dapat dihubungkan ke Arduino atau mikrokontroler lainnya. Sensor ini sangat berguna untuk menggabungkan penggunaan listrik dan membantu mengoptimalkan efisiensi energi.



Gambar 2.2 Sensor PZEM-004T

Modul PZEM-004T memerlukan catu daya 5 VDC dan menggunakan komunikasi serial RX dan TX untuk bertukar data dengan mikrokontroler. Modul ini juga memiliki kemampuan menyimpan nilai konsumsi energi listrik dalam satuan kWh yang telah diukur, sehingga ketika modul kehilangan daya atau dimatikan, data kWh terakhir tetap tersimpan dan dapat dibaca ulang saat modul kembali mendapatkan tegangan. Sensor ini kompatibel dengan berbagai mikrokontroler dan *platform* open source seperti Arduino, ESP8266, ESP32, dan Raspberry Pi, memudahkan integrasi dalam berbagai proyek pengukuran listrik (Alfian et al., 2021). Kemampuan PZEM-004T memungkinkan pengguna untuk dengan mudah mengintegrasikan dan mengelola penggunaan energi mereka secara efisien dan terjangkau.

6. Relay

Relay adalah saklar yang dikontrol secara elektrik dan meliputi komponen elektromekanik, yang terdiri dari dua bagian utama yaitu elektromagnet (kumparan) dan mekanik (kontak sakelar). Relay bekerja berdasarkan prinsip elektromagnetik, di mana arus listrik kecil dalam kumparan dapat menggerakkan kontak sakelar untuk menghubungkan

atau memutus arus listrik bertegangan lebih tinggi (Akbar et al., 2022). Relay sebagai komponen elektronika yang memiliki peranan penting dalam rangkaian elektronika dan listrik untuk mengendalikan perangkat yang membutuhkan arus besar tanpa harus terhubung langsung dengan kontroler yang menggunakan arus kecil. Relay juga disebut berfungsi sebagai alat pengaman.



Gambar 2.3 Modul Relay

Sumber : (Fadhillah et al., 2020)

Modul relay merupakan salah satu perangkat yang bekerja dengan prinsip elektromagnetik, di mana energi listrik digunakan untuk menggerakkan kontaktor agar dapat dipindahkan dari posisi ON ke OFF atau sebaliknya. Perubahan kontak posisi ini disebabkan oleh adanya medan magnet yang dihasilkan oleh kumparan listrik. Perbedaan utama antara relay dan saklar terletak pada mekanisme perpindahannya; relay bekerja secara otomatis berdasarkan sinyal logika, sedangkan saklar umumnya dioperasikan secara manual. Secara umum, fungsi modul relay adalah sebagai saklar elektrik yang dikendalikan secara otomatis (Jepri et al., 2022).

7. Arduino IDE

Menurut Saramuddin, (2023) Arduino IDE (*Integrated Development Environment*) merupakan perangkat lunak yang khusus digunakan untuk memprogram papan Arduino. Dengan kata lain, Arduino IDE berfungsi sebagai *platform* untuk menulis, mengedit, dan mengkompilasi kode program yang berjalan di Arduino. Selain itu, Arduino IDE juga berfungsi sebagai editor teks, yang memungkinkan pengguna untuk menulis dan memodifikasi program dengan mudah sebelum mengunggahnya ke papan Arduino


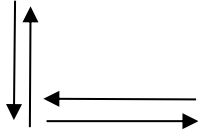
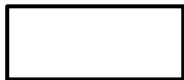

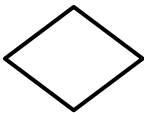
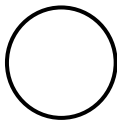
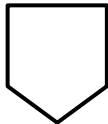
Arduino menjadi *platform* pembuatan *prototype* sumber terbuka yang mudah digunakan, yang terdiri dari papan mikrokontroler dan perangkat lunak Arduino IDE untuk menulis dan mengunggah kode. Papan ini cocok untuk berbagai proyek elektronik, karena dapat membaca sinyal masukan dari sensor dan menghasilkan keluaran seperti mengendalikan motor, menyalakan LED, dan menghubungkan ke cloud (Burhanudin et al., 2023). Arduino IDE sangat populer di kalangan pembuat dan pengembang karena kemudahan penggunaannya dan dukungannya terhadap berbagai papan Arduino. Arduino IDE diprogram dalam bahasa pemrograman C dan memiliki ekosistem yang memudahkan penggunaannya.




8. Flowchart

Menurut Dianta (2021) Flowchart adalah diagram alir yang secara grafis dan sistematis menampilkan proses logika program dan mengaturnya dalam urutan untuk memudahkan pemahaman, analisis, dan menemukan solusi saat mengembangkan aplikasi atau program. Flowchart menjadi suatu diagram alir yang disajikan untuk menggambarkan proses-proses logis dalam suatu program secara berurutan, grafis dan sistematis sehingga lebih mudah dipahami dan dianalisis dalam menemukan solusi pengembangan aplikasi atau program (Putri et al., 2022).

Dalam menyusun suatu algoritma, memerlukan alat seperti flowchart yang menggambarkan langkah-langkah untuk memecahkan masalah secara sistematis dan terstruktur, dan pada saat yang sama sekaligus melatih keterampilan pemecahan masalah yang baik. Keuntungan menggunakan flowchart adalah menciptakan representasi grafis dari langkah-langkah dalam suatu proses, membuatnya lebih mudah untuk dilihat dan dipahami. Penggunaan flowchart mempermudah kita dalam mengidentifikasi bagian-bagian yang mungkin terlewat saat menganalisis suatu permasalahan. Selain itu, flowchart juga berperan sebagai alat bantu komunikasi antara programmer dengan tim perancang sistem, sehingga proses pengembangan dapat dilakukan berdasarkan rancangan yang telah disusun terlebih dahulu (Haryati, 2022).

Tabel 2.1 Simbol Flowchart

Simbol	Nama	Keterangan
	<i>Terminator Symbol</i>	Simbol yang digunakan untuk awal permulaan (start) atau akhir (stop) dari suatu program.
	<i>Flow Direction Symbol</i>	Simbol yang digunakan untuk menghubungkan dari simbol satu dengan simbol yang lain.
	<i>Processing Symbol</i>	Simbol yang digunakan untuk menunjukkan pengolahan yang dilakukan oleh komputer.
	<i>Input Output Data</i>	Simbol yang digunakan untuk menyatakan proses input dan output data.
	<i>Keputusan (Decision)</i>	Simbol yang digunakan untuk memberikan pilihan antara ya atau tidak
	<i>On Page Connector</i>	Simbol untuk keluar-masuk yang menghubungkan proses dalam halaman yang sama.
	<i>Off Page Connector</i>	Simbol untuk keluar-masuk yang menghubungkan proses dalam halaman yang berbeda-beda.

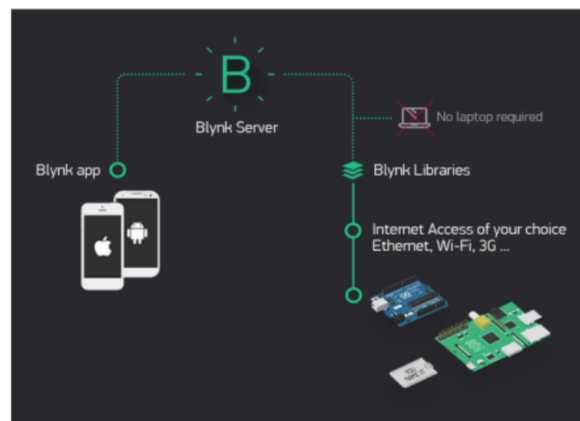
	<i>Predefined Process</i>	Simbol untuk pelaksanaan suatu bagian sub program/prosedur.
	Simbol Dokumen	Simbol yang menyatakan input/output dalam bentuk cetak kertas.
	<i>Preparation</i>	Simbol untuk mempersiapkan penyimpanan yang digunakan sebagai tempat pengolahan di dalam storage.

Sumber : (Indahyanti et al., 2020)

Simbol-simbol pada gambar di atas memiliki berbagai jenis dan fungsi yang beragam. Beberapa simbol berperan sebagai penghubung antar simbol, seperti simbol aliran, referensi *on-page*, dan referensi *off-page*. Selain itu, terdapat simbol yang digunakan untuk menunjukkan proses yang sedang berlangsung, serta simbol lainnya yang berfungsi untuk memasukkan data *input* dan menampilkan hasil *output*. (Huda et al., 2021). Simbol-simbol ini tidak hanya membantu menggambarkan urutan langkah-langkah dalam suatu sistem, tetapi juga mempermudah pemahaman terhadap hubungan antar proses, lokasi input dan output, serta titik awal dan akhir dari suatu alur kerja. Dengan menggunakan simbol-simbol yang sesuai, sebuah proses dapat dijelaskan secara sistematis, sehingga memudahkan analisis, perancangan, maupun komunikasi antar tim dalam pengembangan sistem atau pemecahan masalah.

9. Blynk

Menurut Syukhron et al., (2021) blynk merupakan platform sistem operasi dengan konektivitas *Wi-Fi* untuk berkomunikasi dengan perangkat keras yang digunakan. Blynk terdiri dari tiga komponen utama: aplikasi, server, dan pustaka. Server Blynk bertanggung jawab untuk mengelola semua komunikasi antara *smartphone* dan perangkat keras. Diagram blok yang menunjukkan aliran komunikasi dalam aplikasi Blynk ditunjukkan pada gambar dibawah.



Gambar 2.4 Blynk

Sumber : (Syukhron et al., 2021)

Aplikasi Blynk menyediakan dasbor tempat pengguna dapat membuat antarmuka grafis mereka sendiri menggunakan widget yang disediakan. Melalui aplikasi ini, pengguna dapat menyimpan dan menampilkan data sensor secara *real-time* dan memantau status perangkat *Internet of Things* (IoT) dari jarak jauh tanpa pemrograman yang rumit. Blynk dapat diunduh secara gratis untuk perangkat iOS dan Android. Aplikasi ini memungkinkan pengguna untuk mengembangkan dan

mengendalikan proyek *Internet of Things* (IoT) mereka langsung dari perangkat seluler. Versi gratisnya memiliki fungsionalitas dasar yang cukup untuk proyek kecil dan pengujian, sedangkan versi berbayarnya menawarkan fitur tambahan dan kapasitas yang lebih besar untuk mendukung proyek yang lebih kompleks.




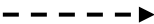
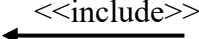

10. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa grafis yang digunakan untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan sistem dalam pengembangan perangkat lunak yang berorientasi pada objek. Perancangan ini bertujuan agar perangkat lunak memenuhi kebutuhan pengguna dan memiliki kinerja tinggi, sehingga menjadi tahap penting dalam proses pengembangannya (Sumirat et al., 2023). Tujuan UML adalah untuk menyediakan kosakata terpadu dari terminologi berorientasi objek dan teknik diagram yang digunakan untuk memodelkan proyek pengembangan sistem dari tahap analisis hingga tahap implementasi (Santoso & Migunani, 2021).

a. *Use Case* Diagram

Use Case Diagram merupakan cara untuk mencatat persyaratan fungsional suatu sistem dengan menggambarkan fungsionalitas sistem yang diharapkan. Diagram ini digunakan untuk memfasilitasi komunikasi antara perancang sistem dan klien serta membantu dalam merancang kasus uji untuk setiap fungsionalitas sistem (Hasanah et al., 2020).

Tabel 2.2 Simbol *Use Case* Diagram



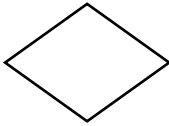

Gambar	Simbol	Keterangan
	<i>Actor</i>	Simbol yang menggambarkan peran orang, sistem lain, atau perangkat saat berinteraksi dengan <i>use case</i> .
	<i>Use Case</i>	Simbol yang menggambarkan abstraksi dan hubungan antara aktor dan <i>use case</i> .
	<i>Association</i>	Simbol yang menggambarkan hubungan abstrak antara aktor dan <i>use case</i> .
	<i>Generalist</i>	Simbol yang menunjukkan spesialisasi aktor untuk berpartisipasi dalam <i>use case</i> tertentu.
	<i>Include</i>	Simbol yang menunjukkan <i>use case</i> merupakan bagian integral dari fungsionalitas <i>use case</i> lain.
	<i>Extend</i>	Simbol yang menunjukkan <i>use case</i> merupakan perluasan dari <i>use case</i> lain dan hanya berlaku jika kondisi tertentu terpenuhi.

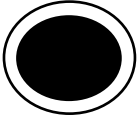
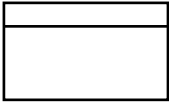
Sumber : (Andriyanto, 2022)

b. *Activity Diagram*

Activity Diagram merupakan teknik pemodelan yang sangat berguna dalam desain perangkat lunak karena membantu memahami secara visual keseluruhan proses bisnis dalam suatu sistem. Diagram tersebut menggambarkan aktivitas yang saling bergantung dalam suatu proses bisnis, sehingga memudahkan pengaturan dan pemahaman urutan serta hubungan antara aktivitas tersebut (Andriyanto, 2022). Diagram aktivitas menyediakan tampilan grafis alur kerja, sehingga memudahkan analisis dan komunikasi tentang cara kerja suatu proses bisnis dari awal hingga akhir.

Tabel 2.3 Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	Status Awal	Simbol yang menandai dimulainya suatu aktivitas dalam diagram aktivitas.
	Aktivitas	Simbol yang mewakili suatu aktivitas atau tindakan yang akan dilakukan oleh suatu sistem.
	Percabangan / <i>Decision</i>	Simbol yang mewakili cabang dalam aliran aktivitas. Digunakan ketika ada beberapa pilihan jalur aktivitas berdasarkan kondisi tertentu.
	Penggabungan / <i>Join</i>	Simbol yang menunjukkan beberapa jalur aktivitas yang bergabung menjadi satu aliran.

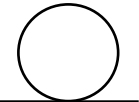
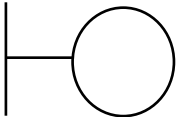
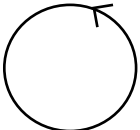
	Status Akhir	Simbol yang menandai akhir suatu aktivitas dalam diagram.
	Swimlane	Simbol yang memisahkan tanggung jawab antara berbagai organisasi atau departemen bisnis dalam pelaksanaan suatu aktivitas.

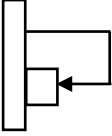


Sumber : (Andriyanto, 2022)

c. *Sequence Diagram*

Sequence Diagram merupakan jenis diagram interaksi yang memodelkan komunikasi antara objek dalam suatu *use case*. Diagram ini merinci perilaku suatu sistem dengan menunjukkan bagaimana berbagai bagian sistem berinteraksi untuk menjalankan fungsinya dan urutan kejadian saat *use case* tertentu dijalankan (Rusli & Triandini, 2022).

Tabel 2.4 Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	<i>Entity Class</i>	Simbol mewakili sistem yang menjadi dasar desain basis data.
	<i>Boundary Class</i>	Simbol yang fungsinya untuk mengelola komunikasi antara lingkungan eksternal dan sistem.
	<i>Control Class</i>	Simbol yang mewakili kelas objek yang berisi logika atau perilaku.

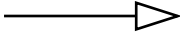
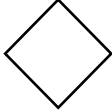
	<i>Recursive</i>	Simbol yang mewakili pengiriman pesan yang ditujukan kepada dirinya sendiri.
	<i>Activation</i>	Simbol yang menunjukkan durasi proses aktif atau aktivasi dalam sistem.
	<i>Life Line</i>	Simbol yang mewakili aktivitas atau tindakan yang dilakukan oleh suatu objek.

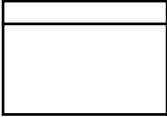
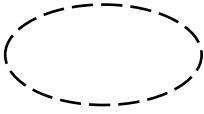
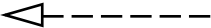
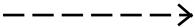

Sumber : (Sumirat et al., 2023)

d. *Class Diagram*

Class diagram adalah model statis dari kelas-kelas dan hubungan di antara kelas-kelas tersebut yang tidak berubah seiring waktu dalam suatu sistem. Diagram tersebut memvisualisasikan atribut (status) dan metode (perilaku) yang dimiliki kelas-kelas, dan menunjukkan bagaimana kelas-kelas tersebut berinteraksi melalui hubungan yang ada (Indriyani et al., 2019).

Tabel 2.5 Simbol *Class Diagram*

Gambar	Nama	Keterangan
	<i>Generalization</i>	Garis lurus yang menggambarkan pewarisan properti dan struktur data dari kelas induk ke kelas turunan.
	<i>Nary Association</i>	Simbol yang membatasi asosiasi hanya pada dua objek.

	<i>Class</i>	Simbol yang menggambarkan kumpulan objek yang memiliki atribut dan operasi yang serupa.
	<i>Collaboration</i>	Simbol yang menggambarkan urutan langkah dalam sistem yang memberikan hasil kepada aktor.
	<i>Realization</i>	Simbol yang menunjukkan bahwa operasi hanya dilakukan oleh satu objek.
	<i>Dependency</i>	Simbol yang menunjukkan bahwa perubahan pada elemen independen akan memengaruhi elemen yang bergantung padanya.
	<i>Association</i>	Simbol yang menggambarkan hubungan antara satu objek dan objek lainnya.

Sumber : (Ramdany, 2024)

B. Kajian Empiris

Dalam penelitian ini, penulis mengacu pada beberapa penelitian empiris yang relevan sebagai dasar pengembangan Alat Ukur Konsumsi Listrik *Charging Station Smartphone* Berbasis *Internet of Things* (IoT). Ketiga penelitian yang dipilih memiliki fokus dan objek penelitian yang serupa sehingga mendukung pengembangan sistem ini.

Pada penelitian yang dilakukan oleh Malik & Kamarudin, (2020), dengan judul “*Energy Meter Using a Smartphone*”, pengembangan sistem pemantauan konsumsi energi berbasis *Internet of Things* (IoT) menggunakan

kombinasi utama perangkat keras sensor arus SCT-013-000, dan sensor tegangan ZMPT101B, berhasil menyediakan pemantauan konsumsi listrik secara *real-time* untuk berbagai beban listrik rumah tangga. Seluruh data dikirimkan melalui koneksi *Wi-Fi* ke aplikasi Blynk di ponsel pintar sehingga pengguna dapat mengontrol arus, tegangan, daya, kWh, dan estimasi tagihan listrik secara langsung. Sistem ini juga mampu mengirimkan *notifikasi* kepada pengguna jika terjadi penggunaan daya di luar batas yang ditetapkan, serta memungkinkan pengguna untuk mengambil keputusan lebih cermat dalam penggunaan energi. Hasil pengujian membuktikan sistem ini akurat, efektif dalam meningkatkan kesadaran pengguna terhadap konsumsi energi, dan adaptif untuk dikembangkan lebih lanjut pada berbagai objek pemantauan, termasuk lebih lanjut terminal *charger smartphone* di tempat umum.

Penelitian serupa yang dilakukan oleh Chaiyong & Sonasang, (2022) dengan judul “*Application of Energy Monitoring Using the IoT*” dapat menyimpulkan bahwa sistem monitoring konsumsi energi berbasis *Internet of Things* (IoT) yang dikembangkan menggunakan komponen utama NodeMCU ESP8266 sebagai mikrokontroler dengan konektivitas *Wi-Fi*, sensor arus serta tegangan PZEM-004T sebagai alat ukur parameter digital, serta aplikasi pendukung seperti Blynk dan *ThingSpeak* untuk *visualisasi* data secara *real-time* dan penyimpanan *cloud*, mampu menggabungkan dan merekam data listrik dari berbagai beban secara akurat, efektif, dan mudah diakses melalui *smartphone*. Sistem ini juga terintegrasi dengan layanan *notifikasi Line Notify* untuk memberikan peringatan instan jika terdapat anomali pemakaian energi,

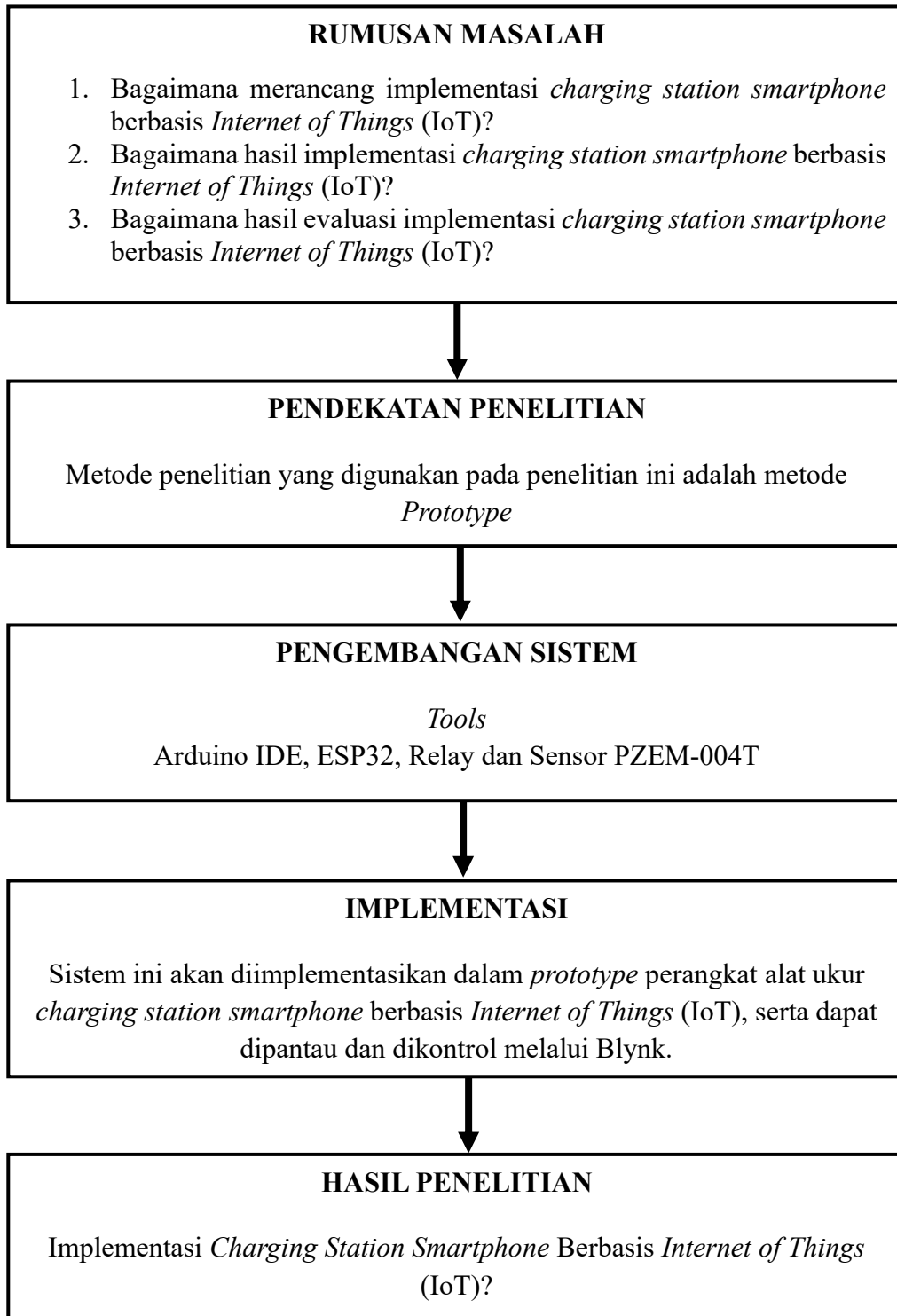
sehingga dapat meningkatkan tingkat kepedulian dan respon pengguna terhadap pemborosan listrik. Pengujian menunjukkan tingkat kesalahan yang sangat rendah pada pengukuran daya dan kinerja sistem yang handal, menjadikan konsep dan implementasi ini sangat sesuai sebagai acuan pengembangan pemantauan konsumsi energi pada perangkat seperti terminal *charger smartphone* di fasilitas umum.

Berdasarkan hasil penelitian ini, peneliti merancang Alat Ukur konsumsi Listrik *charger station smartphone* berbasis *Internet of Things* (IoT) menggunakan PZEM-004T dan ESP32 yang mampu mengukur tegangan, arus, daya, dan energi dengan akurasi tinggi dan keluaran digital yang mudah diintegrasikan ke *Internet of Things* (IoT). Sistem ini lebih unggul dibandingkan penggunaan sensor arus analog seperti SCT-013 dengan Arduino yang hanya mengukur arus secara terbatas dan tanpa kemampuan pemantauan multi-parameter secara mendetail. Dengan demikian, kombinasi PZEM-004T dan NodeMCU ESP32 dinilai lebih praktis, akurat, dan efisien untuk memantau konsumsi listrik berbasis *Internet of Things* (IoT), baik pada lingkungan rumah tangga maupun fasilitas umum seperti terminal *charger smartphone*.

C. Kerangka Berpikir

Kerangka berpikir dalam penelitian ini berfungsi sebagai panduan dalam menyusun langkah-langkah sistematis dari awal hingga akhir untuk mencapai tujuan penelitian. Kerangka ini membantu proses berjalan secara terstruktur dengan menunjukkan hubungan antara konsep, metode, dan

teknologi yang digunakan dalam pengembangan sistem pemantauan konsumsi listrik berbasis *Internet of Things* (IoT). Berikut ini adalah penjelasan tentang kerangka berpikir yang ditunjukkan pada Gambar 2.5.



Gambar 2.5 Kerangka Berpikir