

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. *Internet Of Things (IoT)*

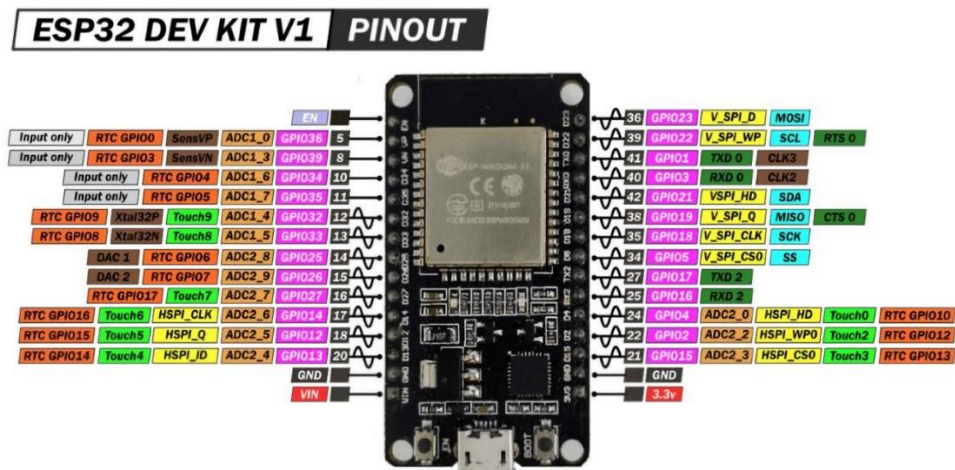
Menurut Erwin *et al.*, (2023:2) *Internet Of Things (IoT)* merupakan bentuk produk inovasi yang memiliki peluang untuk meningkatkan ekonomi. Data yang dikumpulkan berasal dari jutaan perangkat yang terhubung untuk membantu meningkatkan efisiensi, mengurangi biaya operasional dan meningkatkan layanan. Selain itu, IoT telah membuka peluang baru untuk pemodelan bisnis dan bisnis berbasis layanan yang berfokus pada pengalaman pengguna. Karakteristik IoT memungkinkan akses teknologi ini digunakan kapan saja, dimana saja, melalui berbagai perangkat, oleh siapapun serta untuk beragam tujuan.

Menurut Angellia *et al.*, (2024:1) *Internet Of Things (IoT)* merupakan suatu konsep tentang perangkat yang terhubung ke internet untuk berkomunikasi satu sama lain melalui internet. Melalui penerapan IoT, pengguna dapat melakukan berbagai aktivitas, seperti pencarian, pengolahan, serta pengiriman data secara otomatis. Contoh paling sederhana adalah pada pengoperasian mesin di lingkungan industri. Dalam suatu pabrik, mesin-mesin dapat beroperasi secara otomatis dan saling berkomunikasi satu sama lain secara eksklusif. Oleh karena itu, mereka memiliki kemampuan untuk mengatur proses produksi secara mandiri tanpa keterlibatan langsung dengan manusia.

2. *Nodemcu* ESP32

Menurut Budi *et al.*, (2026:146) ESP32 adalah hasil dari pengembangan Espressif System yang berbasis chip (SoC) yang serbaguna dan memiliki prosesor, modul *WiFi*, dan *Bluetooth* di dalamnya. *Prosesor dual-core Xtensa LX6*, yang beroperasi hingga 240 MHz, memiliki arsitektur *RISC-V* dan mendukung berbagai periferal seperti *UART*, *SPI*, *I2C*, *GPIO*, *ADC*, dan *DAC*. ESP32 telah menjadi salah satu platform yang paling populer untuk pengembangan proyek IoT karena kemampuannya yang serbaguna, hemat daya listrik dan biayanya yang terjangkau,

Menurut Setyawan *et al.*, (2022:8) ESP32 merupakan salah satu keluarga mikrokontroler yang dikembangkan oleh *Espressif Systems* sebagai penerus dari mikrokontroler ESP8266. Mikrokontroler ini memiliki kompatibilitas yang tinggi terhadap Arduino IDE dan tersedia modul *Wi-Fi* serta dipadukan dengan *Bluetooth Low Energy* (BLE) didalam chipnya sehingga sangat praktis dan dapat menjadi pilihan yang baik untuk membuat sistem aplikasi IoT. Dalam perkembangan IoT perlu mengetahui komponen dasar yang mempengaruhi perkembangan IoT, komponen dasar ini termasuk *gateway*, *gateway cloud*, pemrosesan data *streaming*, data lake, gudang data besar dan analitik data.



Gambar 2. 1 Nodemcu ESP32

Sumber : Ramdhan, (2026:1), Panduan Membuat *Smart Home* 4 Relay Berbasis ESP32 + Kodular

3. Sensor Warna TCS34725

Menurut Romadhon & Umam, (2022:73) Modul sensor warna GY-33 TCS 34725 merupakan sensor yang digunakan untuk mendeteksi warna dengan memanfaatkan elemen penginderaan cahaya RGB serta *Clear*. Sensor ini dilengkapi dengan *infrared (IR) blocking filter* yang terintegrasi pada chip yang terlokalisasi pada foto sensor warna, sehingga mampu meminimalkan bagian spektrum inframerah yang diterima dan menghasilkan pengukuran warna yang akurat. Selain itu, sensor ini memiliki rentang dinamis yang tinggi, yaitu mencapai 3.800.000:1, serta didukung oleh pengaturan waktu integrasi dan penguatan (*gain*) yang dapat disesuaikan dan ideal untuk digunakan di belakang kaca gelap. Sensor ini beroperasi pada tegangan kerja sekitar 3.3 Volt hingga batas tertentu sesuai dengan spesifikasi teknisnya. Selain itu, modul ini dilengkapi dengan 7 kaki/pin yaitu *LED*, *INT*, *SDA*, *SCL*, *3V3*, *GND* dan *VIN*.

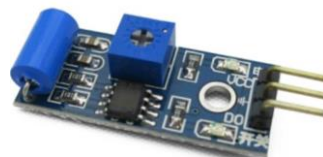


Gambar 2.2 Sensor Warna TCS34725

Sumber : Romadhon & Umam, (2022:74), *Project Sistem Kontrol Berbasis Arduino*

4. *SW-420 Vibration Sensor*

Menurut Bindhu *et al.*, (2023:129) Modul Sensor Getaran SW420 merupakan perangkat pendeteksi getaran yang bekerja berdasarkan prinsip pemutusan sementara rangkaian dan *output* rendah saat terjadi getaran atau gerakan. Pada kondisi normal, sensor berada dalam keadaan rangkaian tertutup, sedangkan ketika terjadi getaran sensor akan menghasilkan sinyal keluaran dengan logika rendah (*low output*). Secara prinsip kerja, ketika sensor mendeteksi adanya getaran, nilai resistansi pada potensiometer akan mengalami peningkatan secara signifikan hingga mendekati tak terhingga, sehingga aliran arus listrik menjadi terhambat.



Gambar 2.3 *SW-420 Vibration Sensor*

Sumber : Khakim, (2024:66), *Buku Ajar Elektronika Dasar*

5. Solenoid Door Lock

Menurut Wahyuzi & Yendri, (2023:9) *Solenoid Door Lock* merupakan jenis *solenoid* yang digunakan sebagai pengunci pintu otomatis. *Solenoid* ini bekerja dengan tegangan *supply* sebesar 12V dan menggunakan mekanisme kerja NC (*Normally Close*) yaitu kondisi awal dalam keadaan terkunci. Prinsip kerjanya menggunakan prinsip elektromagnetik untuk dapat menggerakkan pengunci. Selain itu, *solenoid* ini memerlukan antarmuka relay 5V DC.

Menurut Widharma & Wiranata, (2022:145) pengunci pintu *solenoid door lock* yaitu jenis solenoid yang digunakan sebagai solenoid untuk pengunci pintu elektronik. *Solenoid door lock* ini terdapat dua mode cara kerja: *Normally Close* (NC) dan *Normally Open* (NO).



Gambar 2.4 *Solenoid Door Lock*

Sumber : Wahyuzi & Yendri, (2023:9), Kotak Pintar Penerima Paket untuk Mencegah Penularan Covid-19 Berbasis *Internet of Things*

6. Relay 5V DC

Menurut Pratiwi *et al.*, (2025:27) Relay 5V *Single Channel* membutuhkan tegangan *input* 5V dan berfungsi sebagai saklar elektronik untuk memungkinkan perangkat yang membutuhkan tegangan atau arus tinggi untuk mengontrolnya melalui sinyal tegangan rendah, seperti mikrokontroler atau *mikroprosesor*. Selain itu, terdapat tiga terminal *output*

utama pada relay ini: *Normally Open (NO)*, *Normally Closed (NC)* dan *Common (COM)*. Ketika relay tidak aktif, kontak *NO* dan *COM* dalam keadaan terbuka, sedangkan *NC* dan *COM* terhubung. Saat relay aktif, kontak *NO* dan *COM* terhubung, sedangkan *NC* dan *COM* terputus.



Gambar 2. 5 Relay 5V DC

Sumber : Pratiwi *et al.*, (2025:26), Kotak Pintar Penerima Paket untuk Mencegah Penularan Covid-19 Berbasis *Internet of Things*

7. Sensor Sidik Jari (*Fingerprint Sensor*)

Menurut Nehru *et al.*, (2023:56) Sensor sidik jari yang biasanya digunakan dalam sistem keamanan dan pengenalan identitas seseorang menggunakan pola sidik jari individu untuk mengidentifikasi orang. Dalam perkembangan teknologi modern, sensor memiliki peranan yang sangat penting karena mereka memungkinkan berbagai sistem dan otomatisasi agar dapat bekerja dengan baik. Pemanfaatan sensor ini terdapat pada perangkat rumah tangga hingga sistem yang lebih kompleks dalam sektor industri maupun penelitian.



Gambar 2. 6 Sensor Sidik Jari (*Fingerprint Sensor*)

Sumber : Setiawan, (2025:40), *Internet Of Things : Mengenal Nodemcu ESP8266 dan Membuat Proyek IoT Sederhana*

8. Buzzer

Menurut Budiprasojo, (2022:88) Buzzer merupakan perangkat yang digunakan untuk menghasilkan sinyal suara melalui proses *mekanis*, *elektromekanis*, atau *piezoelektrik*. Sementara itu, Liem *et al.*, (2024:21) menjelaskan bahwa Buzzer memiliki rentang tegangan kerja antara 3 hingga 24 volt. Kebanyakan alarm ini digunakan sebagai peringatan bahwa orang harus mengingat atau menangani peristiwa atau kondisi tertentu, seperti kebakaran, perubahan suhu atau keberadaan asap sehingga memungkinkan tindakan penanganan dapat segera dilakukan.



Gambar 2.7 Buzzer

Sumber : Setiawan, (2025:42), *Internet Of Things : Mengenal Nodemcu ESP8266 dan Membuat Proyek IoT Sederhana*

9. LCD Display

Menurut Lukman *et al.*, (2024:75) LCD I2C merupakan modul display yang memanfaatkan protokol komunikasi I2C (*Inter-Integrated Circuit*) untuk menampilkan data yang diterima. Protokol I2C dapat mendukung berbagai perangkat mikrokontroler seperti sensor ataupun ESP32 sehingga dapat saling berkomunikasi melalui dua jalur utama, yaitu jalur SDA (*Serial Data Line*) yang digunakan untuk mengirim dan menerima data sedangkan jalur SCL (*Serial Clock Line*) yang digunakan untuk mengatur waktu transfer data. Selain itu, LCD I2C ini digunakan untuk menampilkan

data sensor atau status sistem seperti suhu, kelembapan ataupun informasi sesuai dengan kebutuhan pengguna. Dengan menggunakan protokol komunikasi I2C membuat penggunaan pin pada mikrokontroler menjadi lebih efisien yang mana dapat membuka peluang bagi pengembang untuk mengembangkan sistem yang lebih canggih dan terintegrasi.



Gambar 2. 8 LCD Display

Sumber : Fathoni *et al.*, (2026:68), *Smart Home* Berbasis Arduino

10. Arduino IDE

Menurut Siregar, (2024:55) aplikasi yang disebut Arduino *Integrated Development Environment* (IDE) adalah program aplikasi yang programnya ditulis pada lintas platform kemudian diunggah ke papan mikrokontroler Arduino yang kompatibel. Arduino IDE memiliki antarmuka yang ramah pengguna dan banyak fitur pendukung untuk membuat pengembangan kode untuk mikrokontroler Arduino lebih mudah. Software Arduino IDE dibuat dari bahasa pemrograman Java dan dilengkapi dengan *library* C/C++ (*wiring*) yang memudahkan operasi *input* dan *output*. Arduino IDE dapat digunakan untuk membuat *sketch* pemrograman atau dengan kata lain Arduino IDE sebagai media untuk memprogram *board* tertentu.

Menurut Setyawan *et al.*, (2022:9) *Arduino Integrated Development Environment* (IDE) adalah perangkat lunak yang mendukung pemrograman Arduino dan dapat diunduh secara gratis dari situs *web* resmi Arduino IDE. Arduino IDE dapat digunakan sebagai *editor* teks untuk membuat, mengedit dan memvalidasi kode program. Aplikasi menunjukkan saat menyimpan dan mengeksport dan dimana kesalahan ditampilkan. Sudut kanan bawah menunjukkan *port serial* yang digunakan. Verifikasi dan kompilasi mengubah *sketch* menjadi kode biner yang dapat diunggah ke mikrokontroler. *Monitor serial* digunakan untuk membuka antarmuka komunikasi serial dan pesan aplikasi yang dieksekusi akan muncul. Bagian ini akan menampilkan notifikasi aplikasi aktif serta pesan *sketch*. Misalnya, ketika aplikasi mengkompilasi atau ketika *sketch* mengalami kesalahan dibagian ini.

11. Kodular

Menurut Putra *et al.*, (2023:82) kodular merupakan *platform online* yang menyediakan *tools* untuk membantu pengembang dalam membuat aplikasi Android menggunakan *block programming*. Kodular memiliki bagian antarmuka yaitu *Designer* dan *Blocks*. Tampilan *designer* memungkinkan membuat desain atau tampilan aplikasi Android yang sesuai dengan kebutuhan baik dari segi antarmuka (*UI*) dan pengalaman pengguna (*UX*), sedangkan tampilan *Blocks* memungkinkan untuk membuat program aplikasi Android yang dirancang dengan baik, sehingga memungkinkan aplikasi Android yang telah dibuat dapat berjalan dengan maksimal.

12. *Firebase*

Menurut (Enterprise, 2022:153) *Firebase* adalah *database server* buatan google. Pengembang aplikasi memilih *Firebase* karena didukung oleh Google yang dianggap sebagai salah satu perusahaan IT terbesar di dunia dan *firebase* memiliki layanan tambahan untuk pengembangan aplikasi seperti *hosting* dan penyimpanan. Dengan demikian, bentuk *database firebase* tidak memiliki struktur baris dan kolom seperti *SQL*, tetapi struktur *database NoSQL* berbentuk *collection*, *document* dan *field*.

Menurut Sari & Indra, (2021:17) *Firebase* adalah *platform* yang mendukung aplikasi *real time*. Ketika data melakukan pembaruan, maka aplikasi yang terhubung dengan *platform* ini akan *diupdate* secara *real time* pada setiap perangkat melalui *website* maupun *smartphone*. *Firebase* memiliki pustaka (*library*) yang lengkap untuk sebagian besar *platform web* dan *mobile* dan tentunya dapat diintegrasikan dengan berbagai *framework* seperti *node*, *java*, *javascript* dan lainnya.

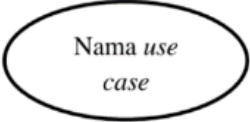
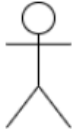
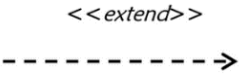
13. UML

Menurut Stianingsih & Hadinata, (2025:96) *Unified Modeling Language* (UML) adalah bahasa pemodelan standar berbasis grafis yang digunakan untuk memodelkan struktur, perilaku dan interaksi sistem secara terstruktur dan mudah dipahami. Selain itu, UML adalah alat yang fleksibel dan penting untuk berbagai proyek perangkat lunak, sehingga kesalahan desain dapat ditemukan lebih awal yang mengurangi waktu pengembangan dan biaya. Diagram UML terdiri dari beberapa pemodelan di antaranya :

a. *Use Case Diagram*

Menurut Wijayanto *et al.*, (2024:76) *use case diagram* harus dibuat sebelum memulai pemodelan perangkat lunak berorientasi objek. Diagram ini akan menggambarkan apa yang dilakukan oleh aktor.

Tabel 2. 1 *Use Case Diagram*

No.	Notasi	Simbol	Keterangan
1	<i>Use Case</i>		Fungsionalitas yang diberikan oleh sistem melalui unit yang berkomunikasi satu sama lain dan aktor.
2	Aktor atau <i>Actor</i>		Orang atau proses yang berinteraksi dengan sistem yang dibuat di luar sistem.
3	<i>Association</i>		Interaksi antara aktor dan <i>use case</i>
4	<i>Extend</i>		Ditambahkan ke sebuah <i>use case</i> yang dapat berdiri sendiri tanpa <i>use case</i> tambahan
6	<i>Include</i> atau <i>Uses</i>		Berfungsi untuk dijalankannya suatu <i>use case</i>

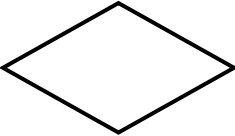
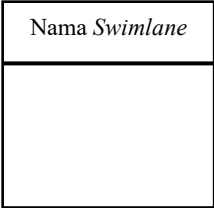
Sumber : Wijayanto *et al.*, (2024:76), Buku Ajar Analisa Perancangan Sistem Informasi

b. *Activity Diagram*

Menurut Wijayanto *et al.*, (2024:78) *activity diagram* merupakan diagram yang menunjukkan aktivitas atau alur kerja dari suatu proses.

Berikut ini adalah simbol yang terdapat pada *Activity Diagram* :

Tabel 2. 2 *Activity Diagram*

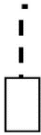
No.	Notasi	Simbol	Keterangan
1	Status Awal		Pada status awal <i>activity diagram</i> terdapat titik awal
2	Aktifitas		Aktivitas sistem yang biasanya dimulai dengan kata kerja.
3	Percabangan atau <i>Decision</i>		Asosiasi percabangan yang muncul jika terdapat lebih dari satu pilihan aktivitas.
4	Penggabungan atau <i>Join</i>		Asosiasi penggabungan jika terdapat lebih dari satu aktivitas digabungkan menjadi satu.
5	Status Akhir		Status akhir yang dilakukan sistem ditunjukkan dalam diagram aktivitas
6	<i>Swimlane</i>		Memisahkan unit bisnis bertanggung jawab atas aktivitas yang dilakukan.

Sumber : Wijayanto *et al.*, (2024:79), Buku Ajar Analisa Perancangan Sistem Informasi

c. *Sequence Diagram*

Menurut Wijayanto *et al.*, (2024:80) *Sequence diagram* adalah diagram yang menunjukkan alur interaksi antar objek. Isi diagram urutan harus sama dengan isi diagram kelas

Tabel 2. 3 *Sequence Diagram*

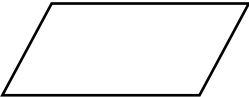
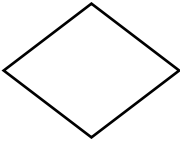
No.	Notasi	Simbol	Keterangan
1	Aktor		Pengguna yang menggunakan sistem
2	Garis hidup		Menjelaskan kehidupan pada suatu objek
3	Waktu aktif		Menunjukkan objek yang berinteraksi dan aktif.
4	Objek	Nama_objek: nam_kelas	Merepresentasikan objek yang saling berinteraksi dalam suatu proses.
5	Pesan type <i>call</i>	nama_metode() 	Menunjukkan bahwa operasi dapat dilakukan pada objek lain atau pada dirinya sendiri
6	Pesan type <i>return</i>		Menunjukkan bahwa objek mengembalikan hasil eksekusi operasi kepada objek yang sebelumnya mengirimkan permintaan.

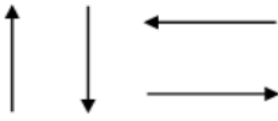
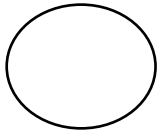
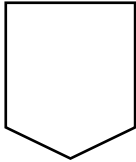
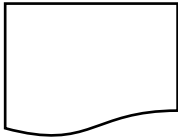
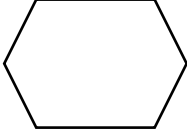
Sumber : Wijayanto *et al.*, (2024:80), Buku Ajar Analisa Perancangan Sistem Informasi

14. Flowchart

Menurut Murad & Wahyuddin, (2022:39) *Flowchart* adalah gambaran grafik dari tahapan penyelesaian suatu masalah yang terdiri dari sekumpulan simbol yang masing-masing mewakili suatu kegiatan. Suatu algoritma program disajikan dalam diagram aliran. Secara khusus, *flowchart* membantu menjelaskan situasi yang sedang dan akan terjadi dengan menggunakan simbol dan tanda penghubungnya. Selain itu, diagram aliran ini dapat membantu menjelaskan alur sistem, menunjukkan kekurangan atau kelebihan dari berbagai proses yang terlibat dalam tahapan sistem. Salah satu fungsi utama *flowchart* adalah untuk mempermudah penyelesaian masalah, terutama masalah yang perlu dipelajari dan dievaluasi lebih lanjut. Berikut ini tabel simbol *flowchart* antara lain sebagai berikut :

Tabel 2. 4 Tabel Simbol *Flowchart*

No.	Simbol	Keterangan
1		Terminal; menunjukkan awal dan akhir dari suatu proses.
2		<i>Input/Output</i> ; digunakan untuk menunjukkan proses input/output data.
3		Proses; digunakan untuk menyatakan suatu proses.
4		Keputusan; simbol yang menunjukkan situasi tertentu yang menghasilkan dua pilihan, yaitu Ya atau Tidak.

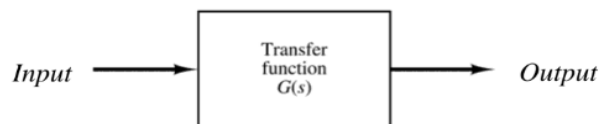
5		Arus atau <i>Flow</i> ; simbol yang menunjukkan alur proses dan bagaimana suatu proses berjalan
6		<i>Connector</i> ; simbol yang menunjukkan bagaimana satu proses berhubungan ke proses lainnya di halaman yang sama.
7		<i>Offline Connector</i> ; simbol yang menunjukkan bagaimana satu proses berhubungan ke proses lainnya di halaman yang berbeda.
8		Proses terdefinisi; menunjukkan suatu proses yang sudah didefinisikan ditempat lain.
9		Dokumen; digunakan untuk mencetak hasil keluaran menjadi bentuk dokumen (melalui <i>printer</i>)
10		Persiapan; digunakan untuk pemberian nilai awal suatu variable.

Sumber : Murad & Wahyuddin, (2022:41), Ekonomi Makro Suatu Analisis dan Aplikasi “Komputer”

15. Diagram Blok

Menurut Evalina *et al.*, (2024:15) *diagram blok* adalah gambaran sederhana tentang hubungan masukan dan keluaran antara masukan dan keluaran sistem. Ini memberikan cara yang paling mudah untuk menunjukkan hubungan fungsional di antara berbagai komponen sistem kontrol yang disebut sebagai elemen sistem.

Menurut Ma'arif, (2021:18) *diagram blok* merupakan sistem yang menunjukkan kinerja setiap komponen sebagai aliran sinyal. Diagram blok ini menunjukkan hubungan yang ada di antara komponen. Diagram blok memiliki keuntungan yang lebih realistis untuk menunjukkan aliran sinyal sebenarnya, yang membedakannya dari representasi matematis. Blok fungsional yang juga disebut sebagai blok dan menunjukkan operasi matematika yang terjadi pada sinyal masukan ke blok yang menghasilkan keluaran. Fungsi pengalihan komponen terletak dalam blok yang sesuai yang dihubungkan dengan panah untuk menunjukkan arah aliran sinyal.



Gambar 2. 9 Diagram Blok

Sumber : Ma'arif, (2021:19), Dasar Sistem kendali pemodelan, pengendalian, analisis, simulasi dan implementasi

B. Kajian Empiris

Dalam penelitian ini terdapat beberapa kebaharuan yang menjadi dasar pengembangan sistem yang diperoleh melalui perbandingan dengan penelitian sebelumnya yang relevan, sebagai berikut

1. Andriyanti *et al.*, (2025) dalam penelitian yang berjudul “Pengembangan Sistem Kotak Amal Berbasis ESP32 dan Sensor TCS3200 untuk Monitoring dan Analisis Donasi *Real-Time*”, penelitian ini bertujuan untuk meningkatkan efisien dan transparan dalam pengelolaan donasi. Penelitian ini menggunakan metode penelitian perancangan dan implementasi sistem yang meliputi perancangan terhadap perangkat hardware dan perangkat software. Hasil penelitian ini

menunjukkan bahwa sistem dapat mengidentifikasi karakteristik warna dari uang kertas pada kotak amal masjid dan melakukan monitoring donasi secara *real time*. Namun, Tingkat akurasi pendeteksian uang kertas dipengaruhi oleh kondisi pencahayaan dan kualitas fisik uang kertas.

2. Soendara *et al.*, (2025) dalam penelitian yang berjudul “Kotak Amal Pintar Berbasis *Internet of Things* (IoT) Menggunakan *Platform* Kodular di SMAN 1 Serangpanjang”, penelitian ini bertujuan untuk mengembangkan kotak amal yang dapat dimonitoring untuk meningkatkan keamanan dan transparansi dalam manajemen donasi. Metode penelitian ini yaitu Analisis kebutuhan, penelitian literatur, perancangan sistem, implementasi, dan pengujian dan evaluasi adalah proses yang digunakan. Hasil penelitian ini menunjukkan bahwa prototipe kotak amal otomatis dapat memonitoring kotak amal secara *real time*.
3. Yasharsujud *et al.*, (2023) dalam penelitian yang berjudul “Sistem Keamanan Kotak Amal Berbasis *Internet of Things* (IoT)”, Penelitian ini menggunakan metode penelitian yaitu studi literatur, observasi, analisis kebutuhan, perancangan perangkat keras dan perangkat lunak, implementasi serta pengujian sistem. Hasil penelitian ini menunjukkan bahwa sistem dapat memberikan notifikasi dan pemantauan lokasi kotak amal secara *real time* dengan akurasi yang cukup tinggi. Namun, permasalahan kondisi jaringan internet dapat menyebabkan ketidaksesuaian data lokasi.

Berdasarkan penelitian sebelumnya, kebaharuan penelitian ini terdapat pada pengembangan sistem kotak amal yang mengintegrasikan beberapa fitur dalam satu perangkat. Sistem yang dikembangkan tidak hanya mampu melakukan

perhitungan dana amal secara otomatis, tetapi juga dilengkapi dengan sistem keamanan yang lebih komprehensif. Penggunaan sensor *autentikasi* sidik jari (*fingerprint*) yang memungkinkan pembatasan akses pada kotak amal hanya kepada pengurus masjid yang telah terdaftar pada sistem, sehingga meningkatkan keamanan kotak amal dibandingkan dengan penelitian sebelumnya yang umumnya hanya menggunakan RFID atau belum memiliki sistem autentikasi yang kuat. Selain itu, sistem ini memiliki nilai tambah keamanan dengan menggunakan *solenoid door lock* sebagai pengunci pintu kotak amal secara otomatis serta buzzer sebagai indikator peringatan dini terhadap indikasi potensi tindakan pencurian atau pembobolan kotak amal masjid. Dengan demikian, penelitian ini menawarkan sistem yang terintegrasi dengan menggabungkan aspek efisiensi, keamanan dan transparansi dalam satu kesatuan perangkat kotak amal.

C. Kerangka Berpikir

Kerangka berpikir adalah alur pikir peneliti sebagai dasar-dasar pemikiran untuk memperkuat sub fokus yang menjadi latar belakang dari penelitian ini. Dalam penelitian dibutuhkan sebuah landasan yang mendasari penelitian agar lebih terarah. Kerangka berpikir pada penelitian ini meliputi masalah, pendekatan penelitian pengembangan sistem, mengimplementasikan sistem, hasil dan pembahasan dari alat *smart charity box* berbasis *Internet of Things (IoT)*. Kerangka berpikir penelitian ini dapat dilihat pada gambar berikut :



Gambar 2. 10 Kerangka Berpikir