

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Perancangan *Smart Lock*

Smart lock mencontohkan kombinasi kenyamanan manusia dan potensi risiko keamanan ini. Ini adalah sistem masuk tanpa kunci yang terhubung dengan *Bluetooth* atau *Wi-Fi*, atau *Z-Wave* atau pendekatan jaringan nirkabel energi rendah lainnya yang dirancang khusus untuk otomatisasi rumah pintar. Kunci yang terhubung dengan jaringan memberikan kenyamanan dan kontrol yang besar kepada pemilik rumah. Pemilik menggunakan perintah suara atau perintah aplikasi smartphone untuk mengaktifkan kunci dan mengatur kontrol akses dan jadwal dari jarak jauh. Beberapa menyediakan fitur *geofencing* yang secara otomatis mengunci pintu jika posisi ponsel pemilik bergerak di luar perimeter yang ditentukan. Mirip dengan sistem alarm rumah, perangkat tersebut mengirimkan peringatan kepada pemilik rumah dalam kasus masuk paksa atau perusakan sistem. Kunci ini sangat nyaman untuk pemilik rumah sewa liburan yang menggunakan sistem untuk mengelola akses dari jarak jauh bagi pelanggan (DeNardis, 2020: 104).

Menurut Jeyalakshmi (2023) kunci biasa kita menggunakan kunci untuk membuka dan menutup pintu serta dapat menggunakan sistem kunci nomor untuk membuka pintu. Dalam sistem kunci pintar, sistem kunci

sempurna disediakan dengan keamanan maksimal. Jika terjadi masalah pencurian, informasi tersebut dapat dikirim secara otomatis ke kantor polisi terdekat atau tetangga. Dalam sistem ini dengan menggunakan arduino hanya dengan sekali login dapat membuka dan menutup pintu dengan keamanan maksimal.

2. *Internet of Things*

Menurut Everard (2021:3) *internet of things* memberikan peluang besar bagi kita untuk mengotomatisasi dan menyederhanakan kehidupan kita. Untuk memanfaatkan peluang ini, kita memerlukan perangkat terbuka yang dapat saling terhubung. Sayangnya, perusahaan teknologi sering lebih tertarik menjual perangkat tertutup yang membatasi kita hanya menggunakan layanan mereka dan berfungsi hanya dengan perangkat lain dari pabrikan yang sama. Namun, hal ini tidak menjadi masalah bagi para pembuat. Dengan banyaknya pilihan papan berbasis *internet* yang dapat diprogram dan berbagai sensor serta aktuator yang menakjubkan, kita dapat membangun sistem *Internet of Things* sendiri.

Menurut Suparjiman et al. (2023:11) *internet of things* (IoT) adalah jaringan yang menghubungkan perangkat komputasi, mesin mekanik dan digital, benda, hewan, dan bahkan manusia.. Setiap entitas dalam IoT dilengkapi dengan ID unik dan mampu mentransfer data melalui jaringan. Kita bisa menganggap IoT sebagai evolusi teknologi yang memperluas penggunaan *internet* dalam kehidupan sehari-hari. Dengan semakin banyaknya pengguna *internet* dan adopsi teknologi *broadband* serta *Wi-Fi*,

IoT akan semakin banyak digunakan di seluruh dunia untuk berbagai keperluan.

3. *Solenoid door lock*

Menurut Wahyuzi & Yendri (2023:9) *solenoid door lock* adalah komponen yang berfungsi sebagai kunci pintu otomatis. *Solenoid* ini dirancang khusus untuk mengunci pintu dan beroperasi pada tegangan 12V. Prinsip kerjanya menggunakan elektromagnetik dengan mode NC (*Normally Close*). *Solenoid* ini juga memerlukan antarmuka relay 5 Volt.

Menurut Widharma & Wiranata (2022:145) kunci pintu *solenoid* adalah jenis *solenoid* yang secara khusus digunakan sebagai pengunci pintu elektronik. *Solenoid* ini memiliki dua mode kerja: *Normally Close* (NC) dan *Normally Open* (NO).



Gambar 2. 1 *Solenoid Door Lock*

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

4. MC 38 Sensor

Menurut Main (2003:139) sakelar pintu bekerja berdasarkan prinsip sakelar *magnet* dua bagian. Sakelar yang sensitif terhadap medan *magnet*

dipasang pada rangka tetap, dan kabel dari sakelar diarahkan melalui dinding ke panel kontrol. Pada pintu, sebuah *magnet* dipasang dekat dengan sakelar saat pintu ditutup, sehingga sakelar tetap tertutup. Ketika pintu dibuka, magnet menjauh dari sakelar, menyebabkan sakelar menjadi “terbuka”, yang kemudian diindikasikan oleh panel kontrol pusat dan mengaktifkan alarm. Sakelar *magnetik* dapat berbentuk *normal* terbuka (NO) atau biasanya tertutup (NC) untuk memenuhi berbagai desain kabel dan opsi pengontrol.

5. Relay

Menurut Nugroho et al. (2020:126) relay adalah sakelar sederhana yang dioperasikan secara elektrik dan mekanis. Relay terdiri dari elektromagnet dan sekelompok kontak. Mekanisme pengalihan (*switching*) dilakukan dengan bantuan elektromagnet. Selain itu, ada juga prinsip operasi lain yang terlibat. Relay yang dipasang di tempat-tempat otomasi menggunakan sinyal daya kecil untuk mengontrol rangkaian pengendali. Penggunaan relai dimulai sejak penemuan telepon. Relay memainkan peran penting dalam mengalihkan panggilan (*switching*) dalam sistem telepon, dan juga digunakan dalam telegrafi jarak jauh. Setelah komputer ditemukan, relay juga digunakan untuk fungsi Boolean dan operasi logika lainnya. Di industri, relay yang membutuhkan daya tinggi untuk menggerakkan motor listrik dan sejenisnya disebut kontaktor.

Menurut Mamta (2023:66-67) relay adalah saklar yang dioperasikan secara elektrik yang menggunakan elektromagnet untuk membuka atau menutup kontakannya secara mekanis. Relay digunakan untuk mengontrol perangkat berdaya tinggi atau bertegangan tinggi, seperti motor, lampu, pemanas, atau peralatan, menggunakan sinyal kontrol berdaya rendah. Modul relai biasanya memiliki input kontrol itu memungkinkan koneksi ke mikrokontroler, Arduino, atau sirkuit digital lainnya. Input ini biasanya mencakup pin sinyal dan pin ground, yang menerima sinyal kontrol untuk mengaktifkan atau menonaktifkan relay.



Gambar 2. 2 Relay

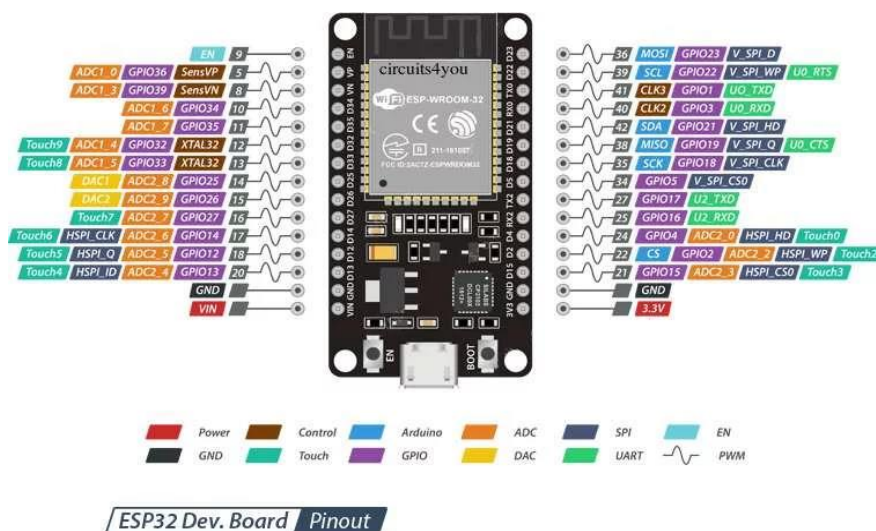
Sumber : Zilwu et al. (2022), Buku Praktikum Otomatisasi dan Digitalisasi, hal. 34

6. Nodemcu ESP32

Menurut Rudra et al. (2022:74) ESP32 adalah serangkaian mikrokontroler *system-on-chip* (SOC) berdaya rendah dan berbiaya rendah dari *Espressif Systems*, sebuah perusahaan yang berbasis di Shanghai. ESP32 diproduksi menggunakan teknologi 40 nm oleh TSMC. Sebagai SOC, ini terintegrasi dengan *Bluetooth dual-mode on-chip* dan *Wi-Fi full stack*. ESP32 diaktifkan dengan *mikroprosesor Tensilica Xtensa LX6* dalam variasi inti

ganda atau inti tunggal atau mikroprosesor inti ganda *Xtensa LX7* atau *mikroprosesor* inti tunggal *RISC-V*. Ini juga mencakup penguat daya, penguat penerima kebisingan rendah, sakelar antenna *internal*, balun RF, filter, dan modul manajemen daya

Menurut Setyawan et al. (2022:8) ESP32 adalah salah satu keluarga mikrokontroler yang diperkenalkan dan dikembangkan oleh *Espressif Systems*. ESP32 ini merupakan penerus dari mikrokontroler ESP8266. Mikrokontroler ini kompatibel dengan Arduino IDE. Pada mikrokontroler ini sudah tersedia modul *Wi-Fi* dan dipadukan dengan BLE (*Bluetooth Low Energy*) di dalam chipnya, sehingga sangat berguna dan dapat menjadi pilihan yang baik untuk membuat sistem aplikasi IoT. Seiring berkembangnya IoT, Anda perlu mengetahui komponen dasar yang memengaruhi evolusi IoT: hal-hal seperti *gateway*, *gateway cloud*, pemroses data *streaming*, data lake, gudang data besar, analitik data, pembelajaran mesin, dan model.



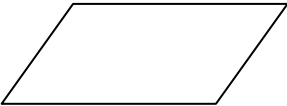

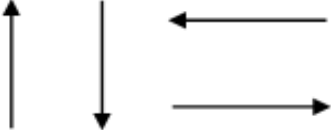

Gambar 2. 3 *Nodemcu* ESP 32

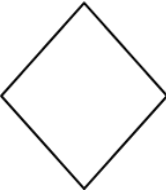

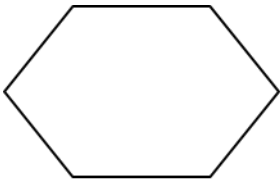

Sumber : Budijanto et al. (2022), *Interfacing dengan ESP32*, hal. 3

7. Flowchart

Menurut Widharma & Wiranata (2022:34) *flowchart* adalah diagram alir secara visual menggambarkan langkah-langkah dan urutan prosedur dalam suatu program. *Flowchart* membantu analis dan *programmer* memecah masalah menjadi bagian-bagian yang lebih kecil dan menganalisis berbagai alternatif dalam operasinya. Saat seorang analis dan *programmer* membuat *flowchart*, mereka dapat lebih mudah memahami dan mengelola proses yang kompleks. Berikut ini tabel simbol *flowchart* antara lain sebagai berikut :

Tabel 2. 1 Tabel simbol *flowchart*

No	Simbol	Keterangan
1		<i>Input/output</i> ; digunakan untuk merepresentasikan data <i>input/output</i>
2		Proses; digunakan untuk merepresentasikan suatu proses.
3		Garis alir; digunakan untuk merepresentasikan suatu proses dan menunjukkan arus dari proses.
4		Penghubung; menampilkan tautan ke halaman yang sama atau halaman lain.

5		Keputusan; digunakan untuk suatu seleksi kondisi didalam program.
6		Terdefinisi; menunjukkan suatu operasi yang rinciannya dijelaskan di tempat lain..
7		Persiapan; digunakan untuk menetapkan nilai awal suatu besaran
8		Terminal; menandai awal dan akhir dari suatu proses

Sumber : Widharma & Wiranata (2022), Mikrontroller dan Aplikasi, hal. 34

8. UML

Unified Modeling Language (UML) merupakan sebuah alat yang dipakai untuk menggambarkan dan mencatat hasil analisis dan desain dalam bentuk visual atau juga merupakan pemodelan sistem secara grafis dan juga sebagai konversi pemodelan merujuk pada metode yang digunakan untuk menggambarkan hubungan antara perangkat lunak sistem dan objek yang terkait

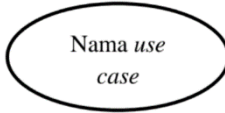

Menurut Habibi et al. (2020:79) UML (*Unified Modeling Language*) adalah bahasa pemodelan yang menggunakan konsep berorientasi


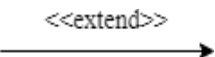

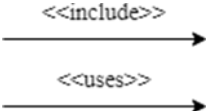
objek (*object-oriented*). UML digunakan untuk menspesifikasi, memvisualisasikan, membangun, dan mendokumentasikan informasi yang dihasilkan selama proses pengembangan perangkat lunak, termasuk pemodelan bisnis dan sistem non-perangkat lunak lainnya. Diagram UML terdiri dari berbagai jenis pemodelan. UML diciptakan oleh *Grady Booch*, *James Rumbaugh*, dan *Ivar Jacobson* di bawah naungan *Rational Software Corps*. UML menyediakan dapat membantu memodelkan sistem dari berbagai prespektif. *Diagram* UML terdiri dari beberapa pemodelan di antaranya:

1. *Use Case Diagram*

Ketika memulai pemodelan perangkat lunak berorientasi objek, *diagram use case* harus dibuat terlebih dahulu. *Diagram* ini akan menunjukkan tindakan yang dilakukan oleh aktor (Wijayanto et al., 2024:76).

Tabel 2. 2 *Use case diagram*

No.	Notasi	Keterangan	Simbol
1	<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang saling bertukar pesan antara unit dan aktor.	
2	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi tersebut.	


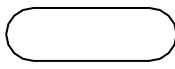
3	<i>Association</i>	Interaksi antara aktor dan use case.	
4	<i>Extend</i>	Hubungan antara use case tambahan dengan sebuah use case di mana use case tambahan tersebut dapat berdiri sendiri tanpa bergantung pada use case utama.	
5	<i>Generalizatiob</i>	Hubungan generalisasi dan spesialisasi antara dua use case di mana salah satu fungsi lebih umum daripada yang lainnya.	
6	<i>Uses atau Include</i>	Berfungsi atau sebagai prasyarat untuk menjalankan use case.	

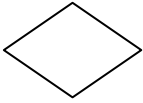


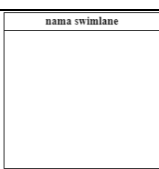
Sumber : Wijayanto et al. (2024), Buku Ajar Analisa perancangan sistem Informasi, hal.76-77

2. Activity Diagram

Menurut Wijayanto et al. (2024:78) *activity diagram* adalah *diagram* yang menggambarkan alur kerja (*workflow*) atau aktivitas dari suatu sistem atau proses bisnis. *Diagram* ini membantu memvisualisasikan langkah-langkah yang terlibat dalam proses tersebut.

Tabel 2. 3 *Activity diagram*

No.	Notasi	Keterangan	Simbol
1	Status awal	Status awal aktivitas pada sebuah diagram aktivitas memiliki titik awal.	
2	Aktivitas	Aktivitas yang dilakukan oleh sistem. Aktivitas ini biasanya dimulai dengan kata kerja..	




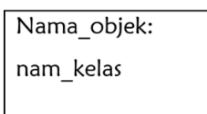
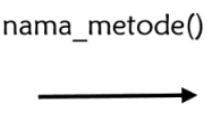
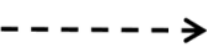
3	<i>Decision</i>	Asosiasi percabangan yang terjadi ketika ada lebih dari satu pilihan aktivitas.	
4	<i>Join</i>	Asosiasi penggabungan yang terjadi ketika lebih dari satu aktivitas digabungkan menjadi satu.	
5	Status akhir	Status akhir yang dicapai oleh sistem, sebuah diagram aktivitas memiliki titik akhir.	<p>Status akhir</p> 
6	<i>Swimlane</i>	Memisahkan unit bisnis yang bertanggung jawab atas aktivitas yang terjadi	

Sumber : Wijayanto et al. (2024), Buku Ajar Analisa perancangan sistem Informasi, hal.79-80

3. *Sequence Diagram*

Sequence diagram atau diagram urutan adalah *diagram* yang digunakan untuk menunjukkan alur interaksi antar objek dalam suatu sistem. Diagram ini membantu memvisualisasikan bagaimana objek-objek tersebut berinteraksi satu sama lain dalam urutan waktu tertentu tersebut (Wijayanto et al., 2024:80).

Tabel 2. 4 *Sequence diagram*

No.	Notasi	Keterangan	Simbol
1	Aktor	Pengguna yang berinteraksi dengan sistem.	
2	Garis hidup	Menggambarkan siklus hidup suatu objek.	
3	Waktu aktif	Mendeklarasikan objek atau benda yang aktif dan berinteraksi.	
4	Objek	Deklarasikan objek atau benda yang berinteraksi	
5	Pesan <i>type call</i>	Mengindikasikan bahwa sebuah objek melakukan operasi pada objek lain atau pada dirinya sendiri.	
6	Pesan <i>type return</i>	Menunjukkan bahwa objek yang telah melakukan operasi mengembalikan pengembalian ke objek yang ditentukan	

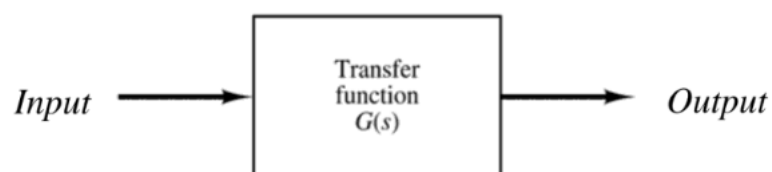
Sumber : Wijayanto et al. (2024), Buku Ajar Analisa perancangan sistem Informasi, hal.80-81

9. *Diagram Blok*

Menurut Ma'arif (2021:18-19) *diagram blok* adalah representasi sistem yang menunjukkan kinerja setiap komponen melalui aliran sinyal. *Diagram* ini menggambarkan hubungan antar komponen dengan lebih

realistis dibandingkan representasi matematis, karena menunjukkan aliran sinyal yang sebenarnya. Dalam diagram blok, semua variabel sistem dihubungkan melalui *blok fungsional*. *Blok fungsional* ini berfungsi sebagai simbol operasi matematika pada sinyal masukan yang menghasilkan keluaran. Fungsi alih komponen dimasukkan ke dalam blok yang sesuai dan dihubungkan dengan panah untuk menunjukkan arah aliran sinyal. Perlu dicatat bahwa sinyal hanya dapat mengalir sesuai arah panah, sehingga diagram blok sistem kendali secara eksplisit menunjukkan sifat satu arah..

Menurut Dwita et al. (2020:206) *diagram blok* adalah representasi visual dari fungsi yang dilakukan oleh setiap komponen dan aliran sinyalnya. Dalam *diagram blok*, semua variabel sistem dihubungkan menggunakan *blok fungsional*. *Diagram blok* menyajikan informasi tentang perilaku dinamis tetapi tidak mencakup informasi mengenai konstruksi fisik sistem. Sebuah sistem dapat digambarkan dengan *diagram blok* yang berbeda tergantung pada sudut pandang analisis.



Gambar 2. 4 *Diagram Blok*

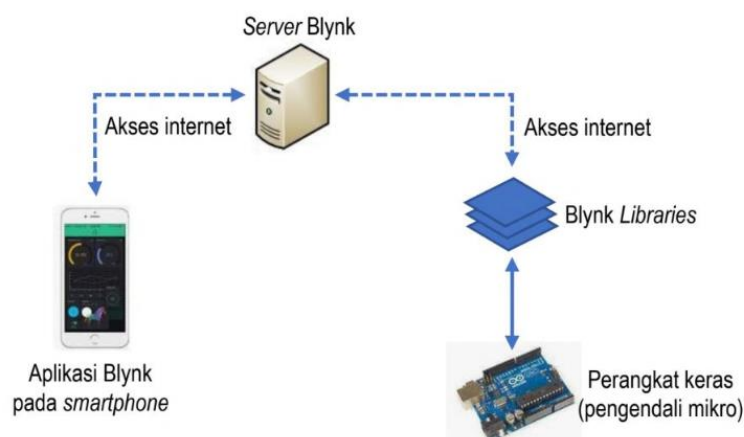
Sumber : Ma'arif (2021), Dasar sistem kendali pemodelan, pengendalian, analisis, simulasi, dan implementasi, hal. 19

10. Aplikasi Blynk

Menurut Swathi (2022:28) blynk adalah *platform Internet of Things* yang bertujuan untuk menyederhanakan pembuatan aplikasi *mobile* dan *web*

untuk *Internet of Things. Platform* ini dengan mudah menghubungkan lebih dari 400 model perangkat keras seperti Arduino, ESP8266, ESP32, Raspberry Pi, dan MCU serupa, serta memungkinkan pembuatan aplikasi IoT *mobile* untuk iOS dan Android dalam waktu 5 menit dengan metode *drag-and-drop*. Dengan menggunakan Blynk, kita dapat membuat aplikasi *smartphone* yang memungkinkan interaksi mudah dengan mikrokontroler atau bahkan komputer penuh seperti Raspberry Pi.

Menurut Susanthi & Andrianto (2024:16) blynk adalah *platform* perangkat lunak yang memudahkan pengendalian perangkat keras seperti Arduino, ESP8266 *Nodemcu*, ESP32, dan mikrokontroler lainnya menggunakan *smartphone* melalui *internet*. blynk terdiri dari tiga komponen utama: aplikasi blynk yang berjalan di *smartphone* dengan sistem operasi Android atau iOS, server Blynk yang menghubungkan komunikasi antara *smartphone* dan perangkat keras, serta blynk *libraries* yang memungkinkan perangkat keras berkomunikasi dengan *server* blynk.



Gambar 2. 5 Blynk

Sumber : Susanthi & Andrianto (2024), Mesin Pembuat Minuman Kopi Otomatis Menggunakan Pengendali Mikro ESP32 dan Smartphone Android, hal. 16

11. Arduino IDE

Menurut Ozsahin & Ozsahin (2021:81) *arduino integrated development environment* (IDE) adalah perangkat lunak sumber terbuka yang terutama digunakan untuk menulis dan mengunggah program dan kode ke modul yang kompatibel dengan arduino. Arduino terdiri dari papan sirkuit perangkat keras dan perangkat lunak yang dapat diprogram. Papan sirkuit ini dapat membaca input dan mengubahnya menjadi output menggunakan serangkaian instruksi atau kode yang telah ditentukan oleh pengguna. Perangkat input dapat berupa lampu, sensor, pesan, atau tombol yang dapat diaktifkan sebagai output. Ini memerlukan serangkaian kode atau bahasa pemrograman dan perangkat lunak arduino ide. Arduino telah digunakan dalam beberapa tahun terakhir untuk sejumlah penemuan ilmiah seperti pemrograman robotika, alat musik, pencetakan 3D, detektor gempa bumi, perubahan iklim, asap, dll. Bahasa pemrograman untuk arduino adalah bahasa pemrograman umum yang disebut C++ dengan penambahan fungsi dan metode khusus, dan ini akan dijelaskan lebih lanjut di bagian selanjutnya dari makalah ini. Ketika sebuah “*sketch*” dibuat, yang merupakan label yang diberikan untuk file kode arduino, pemrosesan dan kompilasi ke dalam bahasa mesin kemudian dijalankan.

Menurut Vijayalakshmi & Raghavendra (2023:18) *arduino integrated development environment* (IDE) adalah *platform* perangkat lunak yang digunakan untuk memprogram papan Arduino. IDE ini menyediakan antarmuka yang ramah pengguna untuk menulis, mengompilasi, dan

mengunggah kodes ke mikrokontroler Arduino. Arduino IDE banyak digunakan oleh pemula maupun pengembang berpengalaman karena kesederhanaannya, fleksibilitasnya, dan dukungan perpustakaan yang luas. Artikel ini akan membahas fitur-fitur kunci dan fungsi dari Arduino IDE. Arduino IDE menawarkan lingkungan pemrograman yang sederhana dan intuitif, sehingga mudah diakses oleh *individu* yang belum memiliki pengalaman pemrograman sebelumnya. IDE ini menyediakan editor teks di mana pengguna dapat menulis kode menggunakan bahasa pemrograman Arduino, yang berbasis pada C dan C++. Fitur penyorotan sintaks dan otomatisasi penyelesaian kode membantu memudahkan proses pemrograman dan mengurangi kesalahan. IDE juga menyediakan serial monitor, yang memungkinkan pengguna melihat dan memperbaiki keluaran dari papan Arduino. Salah satu keuntungan utama dari Arduino IDE adalah kompatibilitasnya dengan berbagai jenis papan Arduino. IDE ini mendukung Arduino Uno yang populer, serta varian lain seperti Arduino Mega, Nano, dan Due. Kompatibilitas ini memastikan bahwa pengguna dapat menulis kode sekali dan menjalankannya dengan mudah di berbagai perangkat keras Arduino tanpa perlu melakukan modifikasi yang signifikan. Fitur penting lainnya adalah dukungan perpustakaan yang luas. Perpustakaan adalah modul kode yang telah ditulis sebelumnya dan memberikan *fungsi* tambahan pada proyek Arduino. IDE dilengkapi dengan *manajer* perpustakaan bawaan, yang memungkinkan pengguna mencari dan menginstal perpustakaan langsung dari dalam IDE. Proses penambahan *fungsi*, seperti

mengendalikan motor, berkomunikasi dengan sensor, atau menghubungkan ke perangkat *eksternal*, menjadi lebih sederhana berkat fitur ini. Arduino IDE menyediakan alur kerja yang lancar untuk mengompilasi dan mengunggah kode ke papan Arduino. Setelah kode ditulis, pengguna dapat memverifikasi dan mengompilasinya hanya dengan satu kali klik.

B. Kajian Empiris

Memberikan informasi tentang penelitian sebelumnya, hasil tinjauan literatur yang berkaitan dengan atau dengan penelitian saat ini digunakan untuk melengkapi atau menyempurnakan penelitian sebelumnya. Materi yang relevan ditemukan dalam jurnal testis dan produk lainnya, dan kemudian dibandingkan.

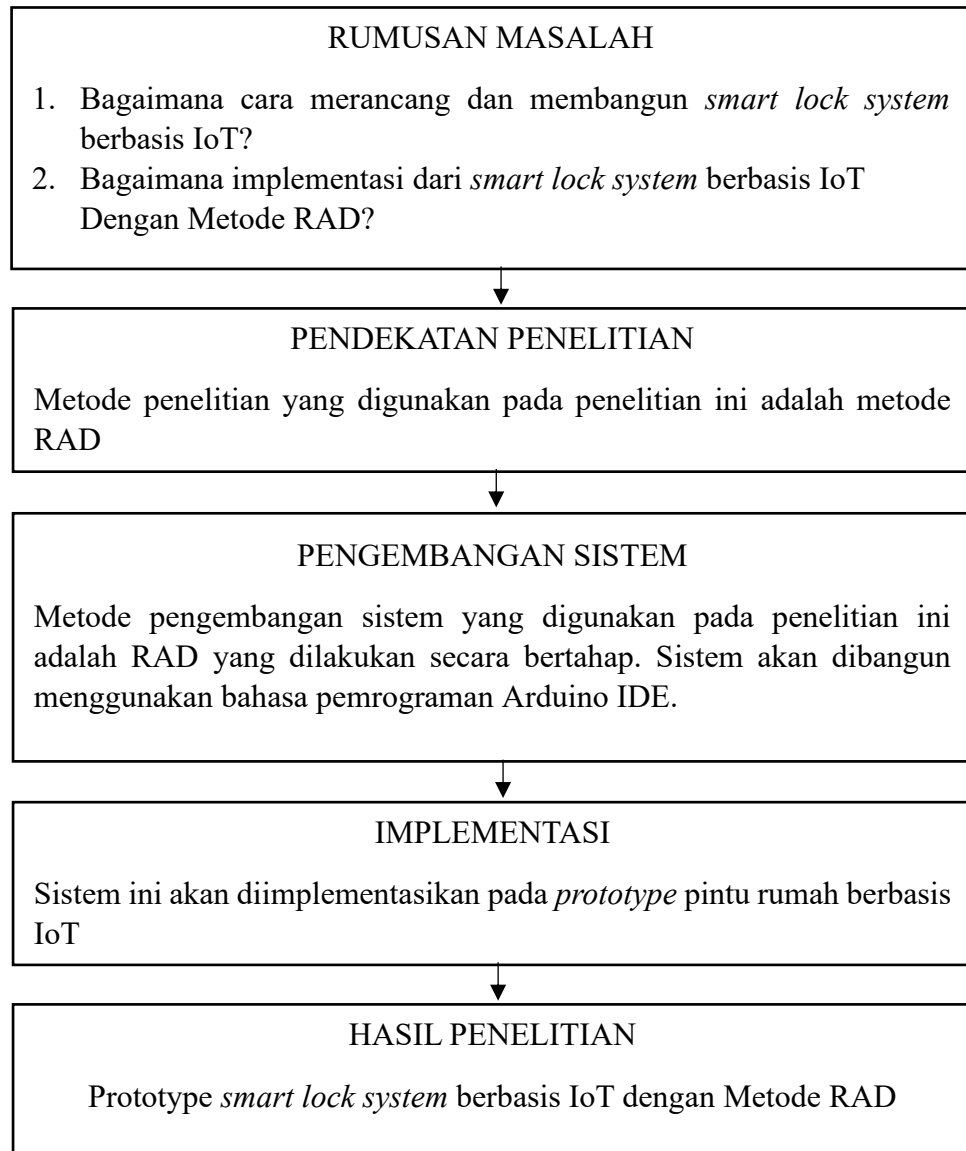
Pada penelitian yang dilakukan oleh Tawakal dengan tema *smart door lock* Menggunakan Akses e-KTP Berbasis *Internet of Things*. Dalam penelitian ini, penulis menggunakan metode penelitian yang terdiri dari beberapa tahap, termasuk analisis latar belakang, perancangan sistem, pengujian, dan analisis hasil. Penulis juga menggunakan beberapa alat dan bahan, seperti RFID *reader*, buzzer, lampu LED, dan *Solenoid door lock*, untuk mengembangkan sistem keamanan pintu rumah yang efektif. Dalam kesimpulan, penulis menunjukkan bahwa sistem keamanan pintu rumah yang menggunakan e-KTP dan IoT dapat meningkatkan keamanan rumah dan memberikan kontrol yang efektif. Mereka juga menunjukkan bahwa teknologi *Internet of Things* dan RFID dapat digunakan untuk meningkatkan keamanan rumah dan memberikan solusi yang lebih efektif untuk masalah keamanan rumah (Tawakal & Ramdhani, 2021).

Penelitian yang dilakukan oleh Pakar dengan tema perancangan sistem *smart door* dengan *notifikasi telegram*. Dalam jurnal "perancangan sistem *smart door* dengan *notifikasi telegram*," metode yang digunakan meliputi penggunaan teknologi RFID, integrasi telegram sebagai sistem notifikasi, pengujian koneksi dan program, perancangan sistem. Dengan menggunakan metode ini, penelitian berhasil mengimplementasikan sistem keamanan pintu pintar yang efektif dengan memanfaatkan teknologi RFID dan *notifikasi telegram* untuk meningkatkan keamanan dan kenyamanan pengguna melalui *Internet of Things* berbasis Telegram (Pakar et al., 2023).

Dengan demikian hasil kedua penelitian ini, menunjukkan bagaimana teknologi *Internet of Things* dapat diterapkan dalam meningkatkan keamanan rumah melalui sistem pintu pintar.

C. Kerangka Berpikir

Kerangka kerja pada penelitian ini meliputi masalah, pendekatan penelitian pengembangan sistem, mengimplementasikan sistem, hasil dan pembahasan. Kerangka berfikir penelitian ini dapat dilihat pada gambar berikut:



Gambar 2. 6 Kerangka Berfikir