



SISTEM KENDALI OTOMATIS RUMAH PINTAR BERBASIS ANDROID

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan Studi
Jenjang Program Diploma Tiga

Oleh:

Nama	NIM
Irma Luthfianalela	18040151

**PROGRAM STUDI DIII TEKNIK KOMPUTER
POLITEKNIK HARAPAN BERSAMA TEGAL
2021**

HALAMAN PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : Irma Luthfianalela
NIM : 18040151
Jurusan / Program Studi : DIII Teknik Komputer
Jenis Karya : Tugas Akhir

Adalah mahasiswa Program Studi DIII Teknik Komputer Politeknik Harapan Bersama, dengan ini saya menyatakan bahwa laporan Tugas Akhir yang berjudul **“SISTEM KENDALI OTOMATIS RUMAH PINTAR BERBASIS ANDROID”**.

Merupakan hasil pemikiran dan kerjasama sendiri secara orisinal dan saya susun secara mandiri dan tidak melanggar kode etika hak karya cipta. Pada pelaporan Tugas Akhir ini juga bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik tertentu di suatu perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau disebutkan dalam daftar pustaka.

Apabila di kemudian hari ternyata Laporan Tugas Akhir ini terbukti melanggar kode etik karya cipta atau merupakan karya yang dikategorikan mengandung unsur plagiarisme, maka saya bersedia untuk melakukan penelitian baru dan menyusun laporannya sebagai Laporan Tugas Akhir, sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya dan sesungguhnya.

Tegal, Mei 2021



Irma Luthfianalela
18040151

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPERLUAN AKADEMIS**

Sebagai civitas akademika Politeknik Harapan Bersama Tegal, Saya yang bertanda tangan dibawah ini:

Nama : Irma Luthfianalela
NIM : 18040151
Jurusan / Program Studi : DIII Teknik Komputer
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Harapan Bersama Tegal **Hak Bebas Royalti *Noneksklusif*** (*None-exclusive Royalty Free Right*) atas Tugas Akhir saya yang berjudul:

**“SISTEM KENDALI OTOMATIS RUMAH PINTAR BERBASIS
ANDROID”**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti *Noneksklusif* ini Politeknik Harapan Bersama Tegal berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan Tugas Akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Tegal

Pada Tanggal : 19 Mei 2021

Yang menyatakan



Irma Luthfianalela

18040151

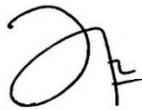
HALAMAN PERSETUJUAN

Tugas Akhir (TA) yang berjudul “**SISTEM KENDALI OTOMATIS RUMAH PINTAR BERBASIS ANDROID**” yang disusun oleh Irma Luthfianalela, NIM 18040151 telah mendapat persetujuan pembimbing dan siap dipertahankan di depan tim penguji Tugas Akhir (TA) Program Studi Diploma III Teknik Komputer PoliTeknik Harapan Bersama Tegal.

Tegal, Mei 2021

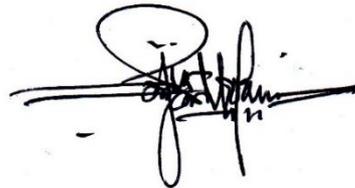
Menyetujui,

Pembimbing I,



Ida Afriliana, ST, M.Kom
NIPY. 12.013.168

Pembimbing II,



Nurohim, S.ST, M.Kom
NIPY. 09.017.342

HALAMAN PENGESAHAN

Judul : SISTEM KENDALI OTOMATIS RUMAH PINTAR
BERBASIS ANDROID
Nama : Irma Luthfianalela
NIM : 18040151
Program Studi : Teknik Komputer
Jenjang : Diploma III

**Dinyatakan LULUS setelah dipertahankan di depan Tim Penguji Tugas Akhir
Program Studi DIII Teknik Komputer Politeknik Harapan Bersama Tegal**

Tegal, Mei 2021

Tim Penguji:

Nama		Tanda Tangan
1. Ketua	: Arfan Haqiqi Sulasmoro, M.Kom	1.
2. Anggota I	: M. Teguh Prihandoyo, M.Kom	2.
3. Anggota II	: Nurohim, S.ST, M.Kom	3.

Mengetahui,

Ketua Program Studi DIII Teknik Komputer,
Politeknik Harapan Bersama Tegal


Rais, S.Pd., M.Kom
NIPY. 07.011.083

HALAMAN MOTTO

”Boleh jadi kamu membenci sesuatu, padahal ia amat baik bagimu, dan boleh jadi (pula) kamu menyukai sesuatu, padahal ia amat buruk bagimu. Allah mengetahui, sedang kamu tidak mengetahui”.

(QS. Al Baqarah : 216)

“Maka sesungguhnya bersama kesulitan itu ada kemudahan”.

(QS. Insyirah : 5)

“Dan bahwasanya seorang manusia tiada memperoleh selain apa yang telah diusahakannya”.

(Qs. An Najm : 39)

HALAMAN PERSEMBAHAN

1. Allah SWT, karena hanya atas izin dan karuniaNya lah maka laporan ini dapat dibuat dan selesai pada waktunya.
2. Bapak dan Ibu yang telah memberikan motivasi dan dukungan moral maupun materi serta do'a yang tiada hentinya.
3. Bapak Nizar Suhendra, S.E, MPP selaku Direktur Politeknik Harapan Bersama Tegal.
4. Bapak Rais, S.Pd, M.Kom selaku Ketua Program Studi DIII Teknik Komputer Politeknik harapan Bersama Tegal.
5. Ibu Ida Afriliana, ST, M.Kom selaku Dosen Pembimbing I.
6. Bapak Nurohim, S.ST, M.Kom selaku Dosen Pembimbing II.
7. Bapak Suheri selaku narasumber pemilik rumah.
8. Semua pihak yang telah mendukung, membantu serta mendoakan penyelesaian Tugas Akhir ini.

ABSTRAK

Padatnya kegiatan di luar rumah membuat seseorang terkadang lupa akan hal keamanan rumah. Hal seperti itu dapat menimbulkan kerugian besar yang berdampak pada keamanan rumah dan perangkat elektronik lainnya. Sulitnya mengontrol keadaan rumah tentunya menjadi beban bagi mereka yang memiliki banyak kesibukan di luar rumah. Untuk mengatasi permasalahan tersebut maka dibuatlah suatu aplikasi pada sistem kendali otomatis rumah pintar berbasis android untuk membantu pemilik rumah dalam mengontrol keadaan rumah saat sedang berada di luar rumah. Pada penelitian ini dibuatlah aplikasi berbasis android dengan menggunakan *app inventor*. Rancangan sistem meliputi *hardware* dan *software*. *Hardware* meliputi mikrokontroler sebagai pusat kendali, relay sebagai pemutus dan penyambung arus listrik, rancangan *software* meliputi aplikasi berbasis android yang digunakan untuk membuka dan menutup pengunci pintu serta mengendalikan perangkat elektronik. Hasil perancangan adalah sebuah aplikasi berbasis android yang digunakan untuk membuka dan menutup pengunci pintu serta mengendalikan perangkat elektronik melalui *smartphone*. Sistem ini menggunakan koneksi *wifi* untuk terhubung ke mikrokontroler.

Kata Kunci : Aplikasi, Android, Mikrokontroler, *Wifi*.

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah SWT, Tuhan Yang Maha Pengasih dan Maha Penyayang yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya hingga terselesaikannya laporan Tugas Akhir dengan judul **“SISTEM KENDALI OTOMATIS RUMAH PINTAR BERBASIS ANDROID”**.

Tugas Akhir merupakan suatu kewajiban yang harus dilaksanakan untuk memenuhi salah satu syarat kelulusan dalam mencapai derajat Ahli Madya Komputer pada program Studi Diploma III Teknik Komputer Politeknik Harapan Bersama Tegal. Selama melaksanakan penelitian dan kemudian tersusun dalam laporan Tugas Akhir ini, banyak pihak yang telah memberikan bantuan, dukungan dan bimbingan.

Pada kesempatan ini, tidak lupa diucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Nizar Suhendra, S.E, MPP selaku Direktur Politeknik Harapan Bersama Tegal.
2. Bapak Rais, S.Pd, M.kom selaku Ketua Program Studi DIII Teknik Komputer Politeknik Harapan Bersama Tegal.
3. Ibu Ida Afriliana, ST, M.Kom selaku Dosen Pembimbing I.
4. Bapak Nurohim, S.ST, M.Kom selaku Dosen Pembimbing II.
5. Bapak Suheri selaku narasumber pemilik rumah.
6. Semua pihak yang telah mendukung, membantu serta mendoakan penyelesaian laporan Tugas Akhir ini.

Semoga laporan Tugas Akhir ini dapat memberikan sumbangan untuk pengembangan ilmu pengetahuan dan teknologi.

Tegal, Mei 2021

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN KEASLIAN	ii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PENGESAHAN.....	v
HALAMAN MOTTO	vi
HALAMAN PERSEMBAHAN	vii
ABSTRAK.....	viii
KATA PENGANTAR	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan dan Manfaat	3
1.5 Sistematika Laporan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Teori Terkait	6
2.2 Landasan Teori	8
BAB III METODOLOGI PENELITIAN	30
3.1 Prosedur Penelitian	30
3.2 Metode Pengumpulan Data.....	31
3.3 Waktu dan Tempat Penelitian.....	32
BAB IV ANALISIS DAN PERANCANGAN SISTEM.....	33
4.1 Analisis Permasalahan	33
4.2 Analisis Kebutuhan Sistem.....	33
4.3 Perancangan Sistem	35

BAB V	HASIL DAN PEMBAHASAN.....	43
5.1	Implementasi Sistem.....	43
5.2	Hasil Pengujian.....	50
BAB VI	SIMPULAN DAN SARAN.....	59
6.1	Simpulan.....	59
6.2	Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN	63

DAFTAR TABEL

	Halaman
Tabel 2.1 Simbol <i>Use Case Diagram</i>	11
Tabel 2.2 Simbol <i>Activity Diagram</i>	13
Tabel 2.3 Simbol <i>Sequence Diagram</i>	14
Tabel 2.4 Simbol <i>Class Diagram</i>	16
Tabel 2.5 Simbol <i>Component Diagram</i>	17
Tabel 2.6 Simbol <i>Deployment Diagram</i>	18
Tabel 4.1 Komponen Aplikasi	35
Tabel 5.1 Rangkaian Arduino Uno	45
Tabel 5.2 Rangkaian Sensor <i>LDR</i>	45
Tabel 5.3 Rangkaian Sensor <i>DHT11</i>	45
Tabel 5.4 Rangkaian Sensor <i>PIR</i>	45
Tabel 5.5 Rangkaian <i>Module WiFi ESP8266</i>	45
Tabel 5.6 Rangkaian Kipas	46
Tabel 5.7 Rangkaian Lampu 1	46
Tabel 5.8 Rangkaian Lampu 2	46
Tabel 5.9 Rangkaian Pintu	46
Tabel 5.10 Pengujian Lampu	51
Tabel 5.11 Pengujian Kipas	53
Tabel 5.12 Pengujian Pintu	55
Tabel 5.13 Pengujian <i>Buzzer</i>	57

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Arduino Uno.....	19
Gambar 2.2 <i>Module</i> Relay	20
Gambar 2.3 Sensor <i>PIR</i>	22
Gambar 2.4 <i>Module</i> Sensor <i>LDR</i>	23
Gambar 2.5 Sensor <i>DHT11</i>	24
Gambar 2.6 Kabel <i>Jumper</i>	25
Gambar 2.7 <i>Buzzer</i>	26
Gambar 2.8 <i>Project Board</i>	26
Gambar 2.9 <i>AC Light Dimmer</i>	27
Gambar 2.10 <i>Push Button</i>	28
Gambar 2.11 <i>Solenoid Door Lock</i>	29
Gambar 2.12 <i>Module Wifi ESP8266</i>	29
Gambar 3.1 Alur Prosedur Penelitian	30
Gambar 4.1 Diagram Blok Aplikasi.....	36
Gambar 4.2 <i>Use Case</i> Rumah Pintar.....	38
Gambar 4.3 <i>Activity Diagram</i> Keluar	39
Gambar 4.4 <i>Activity Diagram</i> Mode Manual.....	40
Gambar 4.5 <i>Activity Diagram</i> Mode Otomatis	41
Gambar 4.6 <i>Sequence Diagram</i> Keluar	41
Gambar 4.7 <i>Sequence Diagram</i> Mode Manual.....	42
Gambar 4.8 <i>Sequence Diagram</i> Mode Otomatis	42
Gambar 5.1 Implementasi Alat	44
Gambar 5.2 Tampilan Menu Utama Aplikasi Rumah Pintar	48
Gambar 5.3 Tampilan Mode Manual Aplikasi Rumah Pintar	49
Gambar 5.4 Tampilan Mode Otomatis Aplikasi Rumah Pintar.....	50
Gambar 5.5 Tampilan Pengujian Lampu Menyala	53
Gambar 5.6 Tampilan Pengujian Lampu Mati.....	53
Gambar 5.7 Tampilan Pengujian Kipas Memutar.....	54
Gambar 5.8 Tampilan Pengujian Kipas Berhenti	55
Gambar 5.9 Tampilan Pengujian Pintu Terbuka.....	56
Gambar 5.10 Tampilan Pengujian Pintu Tertutup	56
Gambar 5.11 Tampilan Pengujian <i>Buzzer</i> Aktif	57
Gambar 5.12 Tampilan Pengujian <i>Buzzer</i> Nonaktif	58

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Surat Kesediaan Pembimbing TA	A-1
Lampiran 2 <i>Block Code</i> Aplikasi	B-1
Lampiran 3 <i>Sources Code</i>	C-1
Lampiran 4 Dokumentasi Observasi	D-1

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Aplikasi merupakan sebuah perangkat lunak atau program yang diciptakan dan dikembangkan untuk melakukan tugas-tugas tertentu pada perangkat komputer, laptop ataupun *smartphone*. Aplikasi android adalah suatu *software* untuk perangkat *mobile* yang mencakup sistem operasi, *middleware* dan aplikasi kunci. Pengembangan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. Serangkaian aplikasi inti android antara lain *email*, program *SMS*, kalender, peta, *browser*, kontak, dan lain-lain.

Saat ini *smartphone* sudah menjadi kebutuhan sekunder bagi semua kalangan manusia. Mulai dari anak-anak, remaja, bahkan orang tua pun sudah memiliki *smartphone*. Jadi *smartphone* sudah tidaklah asing bagi semua orang. Biasanya *smartphone* digunakan sebagai media komunikasi, hiburan, bahkan media pembelajaran. Maka dari itu di zaman seperti sekarang ini *smartphone* sangatlah penting. Kemanapun orang pergi *smartphone* tidaklah lupa.

Fungsi *smartphone* bukan hanya digunakan sebatas pada media komunikasi antar sesama pengguna. Salah satu jenis sistem operasi yang digunakan pada *smartphone* adalah android. Di dalam sistem operasi android terdapat aplikasi android yang juga dapat dimanfaatkan sebagai aplikasi yang

dapat mengendalikan piranti rumah, seperti kontrol pintu dan perangkat elektronik. Kelengahan pemilik rumah menjadi faktor utama banyaknya tindak kriminal, seperti pencurian dikarenakan lupa mengunci pintu atau bisa juga meninggalkan rumah dalam keadaan lampu yang tidak menyala.

Padatnya kegiatan di luar rumah membuat seseorang terkadang lupa akan hal keamanan rumah. Hal seperti itu dapat menimbulkan kerugian besar yang berdampak pada keamanan rumah dan perangkat elektronik lainnya. Sulitnya mengontrol keadaan rumah tentunya menjadi beban bagi mereka yang memiliki banyak kesibukan di luar rumah.

Oleh karena itu, perlu dirancang suatu aplikasi pada sistem kendali otomatis rumah pintar berbasis android untuk membantu pemilik rumah dalam mengontrol keadaan rumah dan juga mempermudah pemilik rumah dalam mengontrol perangkat elektronik. Dengan adanya aplikasi ini diharapkan dapat lebih mengoptimalkan kinerja manusia, karena dapat mengontrol keadaan rumah melalui aplikasi android.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah di atas, diperoleh rumusan masalah yaitu bagaimana menghasilkan sebuah aplikasi pada Sistem Kendali Otomatis Rumah Pintar Berbasis Android?

1.3 Batasan Masalah

Batasan masalah dibuat agar maksud dan tujuan dari penelitian ini terfokus sesuai dengan tujuan dan fungsinya adalah sebagai berikut:

1. menggunakan *MIT App Inventor* untuk membuat aplikasi android.
2. penghubung antara arduino dan aplikasi menggunakan modul *Wifi* dan *Arduino Uno*.
3. aplikasi ini hanya dapat digunakan untuk mengendalikan alat yang terkoneksi.
4. versi Sistem Operasi Android yang digunakan minimal versi *Lollipop*.
5. aplikasi ini hanya dapat digunakan untuk membuka dan menutup pintu, menyalakan dan mematikan lampu, kipas, serta *buzzer*.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan dari dibuatnya penelitian ini adalah menghasilkan sebuah aplikasi pada sistem kendali otomatis rumah pintar untuk membantu pemilik rumah dalam mengontrol keadaan rumah dan perlengkapan elektronik hanya dengan menggunakan *smartphone* sekaligus menghindari tindak kejahatan.

1.4.2 Manfaat

1. Bagi Mahasiswa

- a. Menambah wawasan dan pengetahuan sehingga dapat meningkatkan kreativitas mahasiswa.

- b. Menerapkan pengetahuan mahasiswa tentang bagaimana cara membuat aplikasi berbasis android.
- c. Memberi bekal untuk menyiapkan diri dalam dunia kerja.

2. Bagi Akademik

- a. Sebagai wujud dari perkembangan ilmu pengetahuan dan teknologi (IPTEK).
- b. Sebagai tolak ukur mahasiswa dalam menyusun laporan.
- c. Menambah referensi dan informasi mengenai aplikasi android khususnya di Perpustakaan Politeknik Harapan Bersama Tegal.

3. Bagi Masyarakat

- a. Memudahkan pemilik rumah dalam mengontrol keadaan rumah dan perangkat elektronik.
- b. Meningkatkan keamanan rumah sehingga dapat meminimalisir pencurian.

1.5 Sistematika Laporan

Pembahasan Tugas Akhir ini akan dibagi menjadi enam Bab dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Bab ini meliputi Latar Belakang Masalah, Rumusan Masalah, Batasan Masalah, Tujuan dan Manfaat, serta Sistematika Laporan.

BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan tentang Penelitian Terkait dan Landasan Teori yang mendukung perencanaan serta pembuatan aplikasi.

BAB III METODOLOGI PENELITIAN

Bab ini membahas Prosedur Penelitian, Metode Pengumpulan Data, Waktu dan Tempat Penelitian.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tentang Analisis Permasalahan, Analisis Kebutuhan Sistem, Perancangan Sistem, dan Desain *Input/Output*.

BAB V HASIL DAN PEMBAHASAN

Bab ini membahas tentang Implementasi pada sistem dan Hasil Pengujian pada aplikasi Tugas Akhir.

BAB VI SIMPULAN DAN SARAN

Bab ini berisi Simpulan dan Saran dari hasil pembahasan yang telah diperoleh.

BAB II

TINJAUAN PUSTAKA

2.1. Teori Terkait

Pada penelitian yang dilakukan oleh Muhamad Muslihudin, Willy Renvillia, Taufiq, Andreas Andoyo, dan Fery Susanto (2018) dalam jurnal penelitiannya yang berjudul Implementasi Aplikasi Rumah Pintar Berbasis Android dengan *Arduino Microcontroller*. Penelitian ini berfokus pada pembuatan sistem rumah pintar agar dapat menghubungkan jaringan komunikasi dengan peralatan listrik yang dimungkinkan untuk dikontrol, dimonitoring, atau diakses dari jarak jauh dengan tujuan meningkatkan efisiensi, kenyamanan, dan keamanan. [1]

Penelitian yang hampir sama oleh Dodon Yedri dan Rahmi Eka Putri (2018) yang berjudul Sistem Pengontrolan dan Keamanan Rumah Pintar (*Smarthome*) Berbasis Android. Dari penelitian ini sistem dapat menghidupkan dan mematikan lampu melalui *smartphone* dengan tingkat keberhasilan 100%, sensor *PIR* dapat mendeteksi gerakan, *webcam* dapat mengambil objek dan sistem mengirimkan notifikasi ke *smartphone* maksimal pada jarak 5,5 meter dan *buzzer* dapat aktif seketika pada saat gerakan terdeteksi. Selain itu, sistem dapat menampilkan informasi suhu, kelembaban dan pemakaian arus pada *smartphone*. [2]

Penelitian selanjutnya oleh Ahmad Sahru Romoadhon dan Devie Rosa Anamisa (2017) dengan jurnal yang berjudul Sistem Kontrol Peralatan Listrik

pada *Smarthome* menggunakan Android. Pada penelitian ini terjadi kegagalan apabila posisi mobil terlalu mendekati samping kanan atau kiri dari pintu pagar sehingga salah satu sensor ultrasonik tidak bekerja dengan baik. Namun rancangan yang dibuat berhasil, dapat meningkatkan kenyamanan bagi pemilik rumah karena dapat mengontrol rumah dengan menggunakan android. [3]

Pada penelitian lain yang dilakukan oleh Fauzan Masykur dan Fiqiana Prasetiyowati (2016) yang berjudul Perancangan Aplikasi Rumah Pintar. Pada penelitian ini hampir sama dengan penelitian sebelumnya yaitu untuk memudahkan pemilik rumah dalam mengontrol keadaan rumah, selain itu juga agar menghemat biaya listrik. [4]

Penelitian selanjutnya yaitu oleh Marti Widya Sari dan Hafid Hardyanto (2016) yang berjudul Implementasi Aplikasi *Monitoring* Pengendalian Pintu Gerbang Rumah menggunakan *App Inventor* Berbasis Android. Pada penelitian ini menghasilkan jarak/*range* maksimal modul *bluetooth* pada aplikasi ini adalah 9,5 m, sehingga dalam aplikasinya *user* harus berada dalam radius kurang atau sama dengan 9,5 m. Selanjutnya untuk posisi derajat *user* dalam menggunakan aplikasi ini dapat dilakukan dari 0-360 derajat. Untuk respon kecepatan dalam penggunaan modul *bluetooth* di

bawah 30 detik, sehingga memungkinkan pengguna untuk menghemat waktu dengan membuka pintu gerbang dari dalam mobil. [5]

Penelitian yang terakhir yaitu oleh Achmad Rafiqi Azka, Elang Derdian Marindani, dan Rudy Dwi Nyoto (2018) yang berjudul Rancang Bangun Sistem Pengendali *Smarthome* menggunakan Mikrokontroler dengan *Speech Command* pada *Smarthome* Android. Pada penelitian ini mikrokontroler dapat menghidupkan kipas ketika sensor gas mendeteksi kebocoran gas dan dapat menghidupkan lampu ketika sensor cahaya tidak mendeteksi cahaya. Rata-rata *booting startup* yang dibutuhkan *Raspberry Pi 3* untuk kembali pada pengaturan sebelum dimatikan adalah 22,8 detik. Pengendalian perangkat rumah dengan *speech command* memiliki tingkat akurasi 100%. [6]

2.2 Landasan Teori

2.2.1 Android

Android adalah sistem operasi yang dirancang oleh *Google* dengan basis kernel *Linux* untuk mendukung kinerja perangkat elektronik layar sentuh, seperti tablet atau *smartphone*. Android bersifat *open source* atau bebas digunakan, dimodifikasi, diperbaiki dan didistribusikan oleh para pembuat ataupun pengembang perangkat lunak. Dengan sifat *open source* perusahaan teknologi bebas menggunakan *OS* ini diperangkatnya tanpa *lisensi* alias gratis.

Para pembuat aplikasi bebas membuat aplikasi dengan kode-kode sumber yang dikeluarkan *google*. Dengan seperti itu android memiliki jutaan *support* aplikasi gratis/berbayar yang dapat diunduh melalui *google play*. [5]

2.2.2 Sistem Kendali

Sistem kendali atau sistem kontrol (*control system*) adalah suatu alat (kumpulan alat) untuk mengendalikan, memerintah, dan mengatur keadaan dari suatu sistem. Istilah sistem kendali ini dapat dipraktikkan secara manual untuk mengendalikan setir mobil pada saat mengendarai/menyetir mobil. Misalnya dengan menggunakan prinsip loloh balik. Dalam sistem yang otomatis, alat semacam ini sering dipakai untuk peluru kendali sehingga peluru akan mencapai sasaran yang diinginkan.

Banyak contoh lain dalam bidang industri/instrumentasi dan dalam kehidupan sehari-hari dimana sistem ini dipakai. Alat pendingin (*AC*) merupakan contoh yang banyak dijumpai yang menggunakan prinsip sistem kendali, karena suhu ruangan dapat dikendalikan sehingga ruangan berada pada suhu yang diinginkan. [7]

2.2.3 MIT App Inventor

MIT App Inventor merupakan *platform* untuk memudahkan proses pembuatan aplikasi sederhana tanpa harus mempelajari atau menggunakan bahasa pemrograman yang terlalu banyak. *App Inventor* dapat mendesain aplikasi android sesuai keinginan dengan

menggunakan berbagai macam *layout* dan komponen yang tersedia. *App Inventor* memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi perangkat lunak bagi sistem operasi Android.

App Inventor menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada Scratch dan StarLogo TNG, yang memungkinkan pengguna untuk *drag and drop* objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat android. Dalam menciptakan *App Inventor*, Google telah melakukan riset yang berhubungan dengan komputasi *educational* dan menyelesaikan lingkungan pengembangan *online Google*. [5]

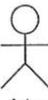
2.2.4 UML (*Unified Modeling Language*)

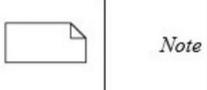
Menurut (Pressman, 2010:841) *Unified Modeling Language* (UML) adalah bahasa standar untuk menulis denah perangkat lunak. *UML* dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak. Dengan kata lain, seperti arsitek bangunan membuat denah yang akan digunakan oleh sebuah perusahaan konstruksi, arsitek *software* membuat diagram *UML* untuk membantu pengembang perangkat lunak membangun perangkat lunak. Jika dapat memahami kosakata *UML*, maka dapat lebih mudah memahami dan menentukan sistem dan menjelaskan desain sistem kepada orang lain. [8]

Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. *UML* merupakan sebuah standar penulisan atau semacam *blue print* dimana di dalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik. Terdapat beberapa diagram *UML* yang sering digunakan dalam pengembangan sebuah sistem, yaitu:

1. *use case*: merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara aktor dan sistem. Di dalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem.

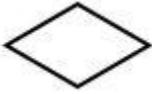
Tabel 2.1 Simbol *Use Case Diagram*

No	Simbol	Nama	Keterangan
1.	 Actor	Aktor	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i> .
2.	 Dependency	Dependency	Suatu proses ketika hubungan pada suatu elemen mandiri atau biasa disebut sebagai <i>independent</i> yang kemudian mempengaruhi elemen lain yang tidak bergantung pada elemen mandiri tersebut (<i>independent</i>).

No	Simbol	Nama	Keterangan
3.		<i>Generalization</i>	Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .
4.		<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
5.		<i>Extend</i>	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi.
6.		<i>Association</i>	Abstraksi dari penghubung antara aktor dan <i>use case</i> .
7.		<i>System</i>	Sistem melakukan spesifikasi paket dengan menunjukkan sistem secara terbatas.
8.		<i>Use case</i>	Abstraksi dan interaksi antara sistem dan aktor.
9.		<i>Collaboration</i>	Berbagai aturan dan elemen yang bekerja guna menyediakan <i>action</i> yang lebih besar dari jumlah elemennya.
10.		<i>Note</i>	Elemen fisik yang terdapat ketika aplikasi mulai dijalankan dan menggambarkan suatu daya komputasi.

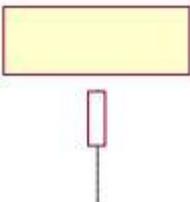
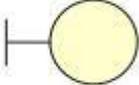
2. *activity diagram*: merupakan gambaran alir dari aktivitas-aktivitas di dalam sistem yang berjalan.

Tabel 2.2 Simbol *Activity Diagram*

No	Simbol	Nama	Keterangan
1.		Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		<i>Swimlane</i>	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
4.		Penggabungan / <i>Join</i>	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
5.		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.		Percabangan / <i>Decision</i>	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.

3. *sequence diagram*: menggambarkan interaksi antar objek di dalam dan di sekitar sistem yang berupa *message* yang digambarkan terhadap waktu.

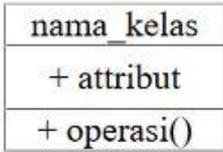
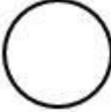
Tabel 2.3 Simbol *Sequence Diagram*

No	Simbol	Nama	Keterangan
1.		Aktor	Merepresentasikan entitas yang berada di luar sistem dan berinteraksi dengan sistem.
2.		<i>Lifeline</i>	Menghubungkan objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya).
3.		<i>General</i>	Merepresentasikan entitas tunggal dalam <i>sequence diagram</i> .
4.		<i>Boundary</i>	Berupa tepi dari sistem, seperti <i>user interface</i> atau suatu alat yang berinteraksi dengan sistem yang lain.
5.		<i>Control</i>	Element mengatur aliran dari informasi untuk sebuah skenario. Objek ini umumnya mengatur perilaku dan perilaku bisnis.
6.		<i>Entitas</i>	Elemen yang bertanggung jawab menyimpan data atau informasi.
7.		<i>Activation</i>	Suatu titik dimana sebuah objek mulai berpartisipasi di dalam sebuah <i>sequence</i> yang menunjukkan kapan sebuah objek mengirim atau menerima objek.

No	Simbol	Nama	Keterangan
8.		<i>Message Entry</i>	Berfungsi untuk menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
9.		<i>Message to Self</i>	Simbol ini menggambarkan pesan/hubungan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi.
10.		<i>Message Return</i>	Menggambarkan hasil dari pengiriman <i>message</i> dan digambarkan dengan arah dari kanan ke kiri.

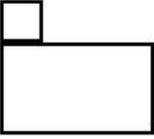
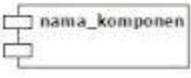
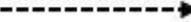
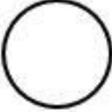
4. *class diagram*: merupakan gambaran struktur dan deskripsi dari *class*, *package*, dan objek yang saling berhubungan seperti diantaranya pewarisan, asosiasi dan lainnya.

Tabel 2.4 Simbol *Class Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Class</i>	Kelas pada struktur sistem.
2.		<i>Interface</i>	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.		<i>Association</i>	Relasi antar kelas dengan arti umum, asosiasi biasanya juga disertai dengan <i>Multiplicity</i> .
4.		<i>Directed Association</i>	Relasi antar kelas dengan makna kelas yang atau digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.		<i>Generalisasi</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
6.		<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas'.

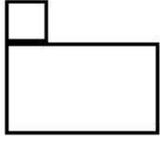
5. *component diagram*: diagram yang menunjukkan secara fisik komponen perangkat lunak pada sistem dan hubungannya. *Component Diagram* merupakan bagian dari sistem yang diuraikan menjadi subsistem atau modul yang lebih kecil.

Tabel 2.5 Simbol *Component Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Package</i>	<i>Package</i> merupakan sebuah simbol yang dipakai untuk tempat komponen.
2.		<i>Component</i>	<i>Component system</i> adalah simbol yang menjelaskan perangkat keras atau objek dalam sistem tersebut.
3.		<i>Dependency</i>	Simbol yang menjelaskan sebuah keterkaitan antara komponen satu dengan yang lain.
4.		<i>Link</i>	Simbol <i>link</i> ini dipakai untuk mengarahkan relasi antar komponen, jika suatu komponen memiliki relasi atau keterkaitan dengan komponen lainnya.
5.		<i>Interface</i>	Hal ini mirip dengan bahasa pemrograman berorientasi objek (PBO), dimana simbol ini dipakai untuk antar muka dengan fungsi supaya tidak langsung mengakses objek.

6. *deployment diagram*: mendeskripsikan arsitektur fisik dalam node untuk perangkat lunak dalam sistem. Komponen perangkat lunak, *processor*, dan peralatan lain yang membangun arsitektur sistem secara *runtime*.

Tabel 2.6 Simbol *Deployment Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Package</i>	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>node</i> .
2.		<i>Node</i>	Biasanya mengacu pada perangkat keras, perangkat lunak yang tidak dibuat sendiri.
3.		<i>Dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
4.		<i>Link</i>	Relasi antar <i>node</i> .

2.2.5 Diagram Blok

Diagram blok adalah diagram dari sistem dimana bagian utama atau fungsi yang diwakili oleh blok dihubungkan dengan garis yang menunjukkan hubungan dari blok. Diagram Blok banyak digunakan dalam bidang teknik dalam desain perangkat keras, desain elektronik, desain perangkat lunak, dan diagram alur proses.

Diagram blok biasanya digunakan untuk level yang lebih tinggi, deskripsi yang kurang mendetail yang dimaksudkan untuk memperjelas konsep keseluruhan tanpa memperhatikan detail implementasi. Bandingkan dengan diagram skema dan diagram tata letak yang digunakan dalam teknik kelistrikan, yang menunjukkan detail implementasi komponen listrik dan konstruksi fisik. [9]

2.2.6 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 pin *input* dari *output digital* dimana 6 pin *input* tersebut dapat digunakan sebagai *output PWM* dan 6 pin *input analog*, 16 MHz osilator kristal, koneksi *USB*, *jack power*, *ICSP header*, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board Arduino Uno* ke komputer dengan menggunakan kabel *USB* atau listrik dengan *AC* yang ke adaptor *DC* atau baterai untuk menjalankannya.

Arduino Uno berbeda dengan semua *board* sebelumnya dalam hal koneksi *USB-to-serial* yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter *USB-to-serial* berbeda dengan *board* sebelumnya yang menggunakan chip *FTDI driver USB-to-serial*. [1]



Gambar 2.1 Arduino Uno

2.2.7 Relay

Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan

Mekanikal (seperangkat Kontak Saklar/*Switch*). Relay menggunakan Prinsip Elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan relay yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan Armature Relay (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A. [3]



Gambar 2.2 *Module Relay*

2.2.8 **Sensor PIR**

Sensor *PIR* atau disebut juga dengan *Passive Infra Red* merupakan sensor yang digunakan untuk mendeteksi adanya pancaran sinar infra merah dari suatu objek. Sesuai dengan namanya sensor *PIR* bersifat pasif, yang berarti sensor ini tidak memancarkan sinar infra merah melainkan hanya dapat menerima radiasi sinar infra merah dari luar. Sensor *PIR* dapat mendeteksi radiasi dari berbagai objek dan karena semua objek memancarkan energi radiasi, sebagai contoh ketika terdeteksi sebuah gerakan dari sumber infra merah dengan suhu tertentu yaitu manusia mencoba melewati sumber infra merah yang lain misal dinding, maka sensor akan membandingkan pancaran infra

merah yang diterima setiap satuan waktu, sehingga jika ada pergerakan maka akan terjadi perubahan pembacaan pada sensor. Sensor *PIR* terdiri dari beberapa bagian yaitu, Lensa *Fresnel*, Penyangin Infra Merah, Sensor *Pyroelektrik*, Penguat Amplifier, Komparator.

Sensor *PIR* bekerja dengan cara menangkap pancaran infra merah, kemudian pancaran infra merah yang tertangkap akan masuk melalui Lensa *Fresnel* dan mengenai sensor *pyroelektrik*, sinar infra merah mengandung energi panas membuat sensor *pyroelektrik* dapat menghasilkan arus listrik. Arus listrik inilah yang akan menimbulkan tegangan dan dibaca secara *analog* oleh sensor. Kemudian komparator akan membandingkan sinyal yang sudah diterima dengan tegangan referensi tertentu yang berupa keluaran sinyal 1 bit. Sensor *PIR* hanya akan mengeluarkan logika 0 dan 1. 0 saat sensor tidak mendeteksi adanya perubahan pancaran infra merah dan 1 saat sensor mendeteksi infra merah. Sensor *PIR* hanya dapat mendeteksi pancaran infra merah dengan panjang gelombang 8-14 mikrometer. Manusia memiliki suhu badan yang dapat menghasilkan pancaran infra merah dengan panjang gelombang antara 9-10 mikrometer, panjang gelombang tersebut dapat terdeteksi oleh sensor *PIR* membuat sensor ini sangat efektif digunakan sebagai *human detektor*. Sensor *PIR* hanya akan mendeteksi jika objek bergerak atau secara teknis saat terjadi adanya perubahan pancaran infra merah. [10]



Gambar 2.3 Sensor *PIR*

2.2.9 Sensor *LDR*

LDR (*Light Dependent Resistor*) adalah salah satu jenis komponen resistor yang memiliki nilai resistansi berubah-ubah berdasarkan dari intensitas cahaya yang diterima. *LDR* berasal dari bahan *kadmium sulfide* yang merupakan bahan semikonduktor. Nilai hambatan/resistansi akan turun bila cahaya yang mengenainya semakin banyak/terang. Dan sebaliknya nilai resistansi akan semakin besar bila cahaya yang mengenainya sedikit/gelap dan hal ini menyebabkan terhambatnya arus listrik.

Sehingga fungsi sensor *LDR* adalah untuk menyalurkan arus listrik bila menerima sejumlah intensitas cahaya/terang dan juga menghambat arus listrik dalam kondisi gelap. Naik turunnya nilai resistansi akan sebanding dengan intensitas cahaya yang mengenainya. Biasanya sensor *LDR* memiliki nilai resistansi sebesar 200 Kilo Ohm ketika dalam kondisi gelap dan akan menurun hingga 500 Ohm ketika mendapat banyak cahaya/kondisi terang. [3]



Gambar 2.4 *Module Sensor LDR*

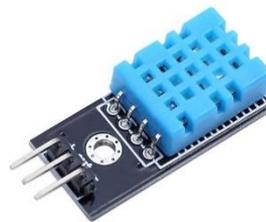
2.2.10 Sensor *DHT11*

Sensor *DHT11* (*Digital Humidity Temperature*) adalah salah satu jenis sensor yang banyak digunakan pada proyek berbasis Arduino. Sensor ini memiliki keunikan yaitu dapat membaca suhu (*temperature*) ruangan dan kelembapan udara (*humidity*). Sensor ini dikemas dalam bentuk kecil dan ringkas, serta harganya yang terjangkau. Kegunaan sensor *DHT11* ini biasanya dipakai pada *project* monitoring suhu ruangan maupun kelembapan udara pada ruangan oven.

Sensor *DHT11* merupakan serangkaian komponen sensor dan *IC Controller* yang dikemas dalam satu paket. Sensor ini ada yang memiliki 4 pin ada pula yang 3 pin. Tapi tidak menjadi masalah karena dalam penerapannya tidak ada perbedaan. Di dalam *body sensor* yang berwarna biru atau putih terdapat sebuah resistor dengan tipe *NTC* (*Negative Temperature Coefficient*). Resistor jenis ini memiliki karakteristik dimana nilai resistansinya berbanding terbalik dengan kenaikan suhu. Artinya, semakin tinggi suhu ruangan maka nilai

resistansi *NTC* akan semakin kecil. Sebaliknya nilai resistansi akan meningkat ketika suhu di sekitar sensor menurun.

Selain itu di dalamnya terdapat sebuah sensor kelembapan dengan karakteristik resistif terhadap perubahan kadar air di udara. Data dari kedua sensor ini diolah di dalam *IC Controller*. *IC Controller* ini akan mengeluarkan *output* data dalam bentuk *single wire bi-directional*. [11]



Gambar 2.5 Sensor *DHT11*

2.2.11 Kabel *Jumper*

Kabel *jumper* adalah kabel elektrik untuk menghubungkan antar komponen di *breadboard* tanpa memerlukan solder. Kabel *jumper* umumnya memiliki *connector* atau pin di masing-masing ujungnya. *Connector* untuk masuk disebut *male connector*, dan *connector* untuk ditusuk disebut *female connector*. Kabel *jumper* dibagi menjadi 3 yaitu: *male to male*, *male to female* dan *female to female*.

Kabel yang digunakan sebagai penghubung antar komponen yang digunakan dalam membuat perangkat *prototype*. Kabel *jumper* bisa dihubungkan ke *controller* seperti *Raspberry Pi*, Arduino melalui *breadboard*.

Karakteristik dari kabel *jumper* ini memiliki panjang antara 10 sampai 20 cm. Jenis kabel *jumper* ini jenis kabel serabut yang bentuk housingnya bulat. Dalam merancang sebuah desain rangkaian elektronik, maka dibutuhkan sebuah kabel yang digunakan untuk menghubungkannya. [12]



Gambar 2.6 Kabel *Jumper*

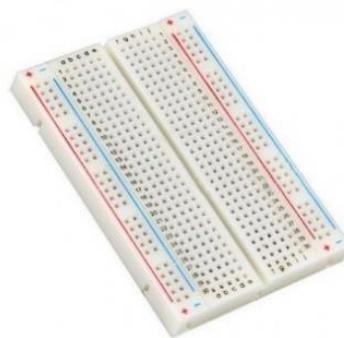
2.2.12 *Buzzer*

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loudspeaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya. Karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. *Buzzer* biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (*alarm*). [3]

Gambar 2.7 *Buzzer*

2.2.13 *Project Board*

Project Board atau yang sering disebut dengan *Breadboard* adalah dasar konstruksi sebuah sirkuit elektronik dan merupakan *prototype* dari suatu rangkaian elektronik. Di zaman *modern* istilah ini sering digunakan untuk merujuk pada jenis tertentu dari papan tempat merangkai komponen. Dimana papan ini langsung dapat digunakan tanpa melakukan proses penyolderan. *Project Board* digunakan untuk membuat suatu rangkaian yang tidak membutuhkan komponen elektronik dalam jumlah banyak. Jumlah lubang koneksi yang dimiliki kurang lebih 170 titik. [13]

Gambar 2.8 *Project Board*

2.2.14 AC Light Dimmer

Perangkat elektronik yang dirancang untuk mengubah daya listrik (*power regulator*). Biasanya digunakan untuk mengatur kecerahan cahaya yang dipancarkan oleh lampu pijar atau *LED*. Biasanya *dimmer* ditemukan sebagai saklar lampu, hanya saja tidak dengan tombol hidup dan mati. Dan dengan tombol penyesuaian kecerahan, dapat digunakan untuk menambah dan mengurangi kecerahan bola lampu. Tetapi penting untuk dipahami bahwa semua ini hanya akan bekerja dengan lampu pijar, tidak akan bekerja dengan lampu hemat energi. Menyesuaikan kecerahan itu bagus, tetapi harus terus-menerus mendekati saklar dan menyalakan kecerahan. Tetapi untuk mengatasi masalah ini, ada *dimmer* digital yang akan dikendalikan oleh mikrokontroler dari kejauhan. [14]



Gambar 2.9 AC Light Dimmer

2.2.15 Push Button

Push Button adalah perangkat/saklar sederhana yang berfungsi untuk menghubungkan atau memutuskan aliran arus listrik dengan sistem kerja tekan *unlock* (tidak mengunci). Sistem kerja *unlock* disini berarti saklar akan bekerja sebagai *device* penghubung atau pemutus

aliran arus listrik saat tombol ditekan, dan saat tombol tidak ditekan (dilepas), maka saklar akan kembali pada kondisi normal.

Sebagai *device* penghubung atau pemutus, *push button switch* hanya memiliki 2 kondisi, yaitu *On* dan *Off* (1 dan 0). Istilah *On* dan *Off* ini menjadi sangat penting karena semua perangkat listrik yang memerlukan sumber energi listrik pasti membutuhkan kondisi *On* dan *Off*.

Karena sistem kerjanya yang *unlock* dan langsung berhubungan dengan operator, *push button switch* menjadi *device* paling utama yang biasa digunakan untuk memulai dan mengakhiri kerja mesin di industri. Secanggih apapun sebuah mesin bisa dipastikan sistem kerjanya tidak terlepas dari keberadaan sebuah saklar seperti *push button switch* atau perangkat lain yang sejenis yang bekerja mengatur pengkondisian *On* dan *Off*. [15]



Gambar 2.10 *Push Button*

2.2.16 *Solenoid Door Lock*

Solenoid Door Lock adalah salah satu pengunci otomatis yang difungsikan khusus untuk pengunci pintu. *Door Lock* membutuhkan tegangan 12V. Sistem kerjanya yaitu *NC* (*Normally Close*) dan *NO*

(*Normally Open*). Cara kerja *NC* adalah apabila diberi tegangan, maka *solenoid* akan memanjang (tertutup). Begitu pula sebaliknya cara kerja *NO* adalah apabila diberi tegangan, maka *solenoid* akan memendek (terbuka). [16]



Gambar 2.11 *Solenoid Door Lock*

2.2.17 *Module Wifi ESP8266*

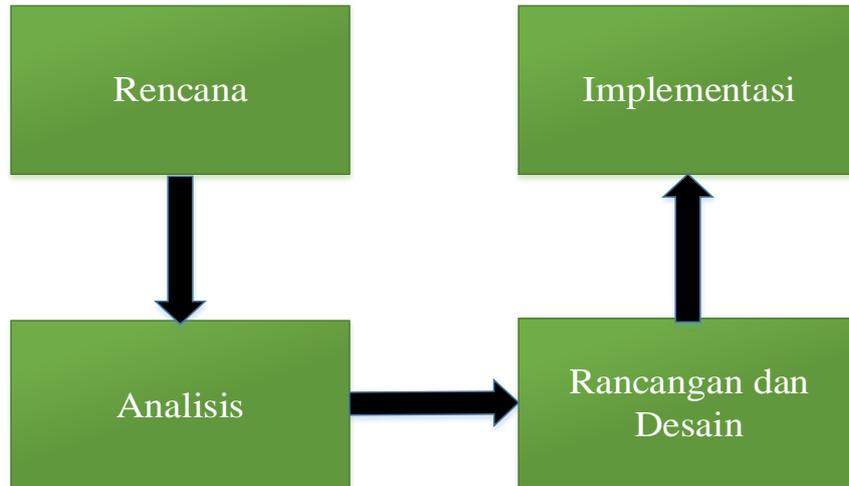
ESP8266 merupakan modul *wifi* yang berfungsi sebagai perangkat tambahan mikrokontroler seperti Arduino agar dapat terhubung langsung dengan *wifi* dan membuat koneksi *TCP/IP*. Modul ini membutuhkan daya sekitar 3.3v dengan memiliki tiga *mode wifi* yaitu *Station*, *Access Point* dan *Both* (Keduanya). Modul ini juga dilengkapi dengan prosesor, memori dan *GPIO* dimana jumlah pin bergantung dengan jenis ESP8266 yang digunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler. [17]



Gambar 2.12 *Module Wifi ESP8266*

BAB III METODOLOGI PENELITIAN

3.1 Prosedur Penelitian



Gambar 3.1 Alur Prosedur Penelitian

Prosedur penelitian dilakukan dengan beberapa tahapan yaitu Rencana, Analisis, Rancangan dan Desain, kemudian Implementasi. Berikut penjelasannya.

3.1.1 Rencana/Planning

Tahap awal pada penelitian ini adalah pencarian ide yaitu pembuatan aplikasi pada sistem kendali otomatis rumah pintar berbasis android serta pengumpulan data-data yang akan digunakan dalam pembuatan aplikasi ini.

3.1.2 Analisis

Analisis berisi langkah-langkah awal pengumpulan data, penyusunan pembuatan aplikasi pada sistem kendali otomatis rumah

pintar berbasis android serta penganalisaan data apa saja yang akan digunakan dalam pembuatan aplikasi ini.

3.1.3 Rancangan dan Desain

Rancangan penelitian adalah suatu cara yang akan digunakan dalam pelaksanaan penelitian dan menjelaskan setiap prosedur penelitian mulai dari tujuan penelitian sampai dengan analisis data. Adapun rancangan atau desain yang akan dibuat adalah meliputi rancangan *software*. *Software* yang digunakan untuk membuat aplikasi berbasis android yaitu *software App Inventor*.

3.1.4 Implementasi

Hasil dari penelitian ini akan diujicobakan secara *real* untuk menilai seberapa baik aplikasi pada sistem kendali otomatis rumah pintar berbasis android yang telah dibuat serta memperbaiki bila ada kesalahan yang terjadi. Kemudian hasil dari ujicoba tersebut akan diimplementasikan.

3.2 Metode Pengumpulan Data

3.2.1 Observasi

Dilakukan pengamatan pada objek terkait guna untuk mengumpulkan data yang diperlukan untuk pembuatan aplikasi. Dalam hal ini observasi dilakukan di Desa Wangandawa Kecamatan Talang Kabupaten Tegal. Meninjau secara langsung lokasi yang akan dibuat sistem kendali otomatis rumah pintar berbasis android.

3.2.2 Studi Literatur/Studi Pustaka Penelitian

Studi literatur adalah mencari referensi teori yang relevan dengan kasus atau permasalahan yang ditemukan, literatur yang didapat bersumber dari jurnal yang mengacu pada permasalahan.

3.3 Waktu dan Tempat Penelitian

Waktu dan tempat penelitian di rumah Bapak Suheri di Desa Wangandawa Kecamatan Talang Kabupaten Tegal. Alasan dipilihnya rumah tersebut sebagai tempat penelitian karena pemilik rumah sering meninggalkan rumahnya dalam keadaan kosong. Entah untuk kegiatan kuliah ataupun kegiatan lain di luar rumah. Jadi dibutuhkan suatu sistem keamanan rumah agar dapat membantu pemilik rumah dalam mengontrol keadaan rumah. Penelitian ini dilakukan pada bulan Oktober – Desember 2020.

BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Permasalahan

Seiring dengan perkembangan teknologi yang begitu pesat memunculkan berbagai macam inovasi dalam berbagai bidang salah satunya adalah rumah pintar. Rumah pintar adalah bagian dari inovasi *Internet of Things* dimana semua peralatan elektronik dapat dikendalikan oleh *smartphone*. Beberapa tahun kemudian rumah pintar merupakan hal yang biasa bagi setiap orang. Semua rumah nantinya sudah memiliki sistem rumah pintar yang dapat mengendalikan dan memonitoring keadaan rumah serta meningkatkan keamanan rumah hanya dengan menggunakan *smartphone*.

Rumah pintar berguna untuk memudahkan penghuni rumah dalam mengatur segala hal yang berhubungan dengan kenyamanan penghuni rumah. Mulai dari keamanan hingga dapat mengontrol dan mengendalikan peralatan elektronik melalui aplikasi berbasis android pada *smartphone*. Sehingga aplikasi android ini diharapkan dapat lebih mengoptimalkan kinerja manusia, karena dapat mengontrol keadaan rumah hanya dengan melalui aplikasi berbasis android.

4.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mengetahui spesifikasi dari kebutuhan aplikasi yang akan dibuat. Pada tahap ini akan membahas

mengenai perangkat keras (*hardware*) dan perangkat lunak (*software*) yang dibutuhkan dalam pembuatan Sistem Kendali Otomatis Rumah Pintar Berbasis Android.

4.2.1 Kebutuhan Perangkat Keras

Kebutuhan *hardware* yang dimaksud yaitu perangkat keras yang digunakan untuk membuat Sistem Kendali Otomatis Rumah Pintar Berbasis Android. Adapun perangkat keras yang dibutuhkan yaitu:

1. arduino uno r3 dip
2. kabel *jumper* (21 buah)
3. relay 2 *channel* (2 buah)
4. sensor *pir*
5. sensor *ldr*
6. sensor *dht11*
7. *buzzer*
8. *project board*
9. *ac light dimmer 1 channel*
10. *push button*
11. *solenoid door lock*
12. *module wifi esp8266*
13. lampu (2 buah)
14. *fan*

4.2.2 Kebutuhan Perangkat Lunak

Kebutuhan *software* yaitu perangkat lunak yang digunakan untuk membuat aplikasi dan program dari Sistem Kendali Otomatis Rumah Pintar Berbasis Android. Adapun perangkat lunak yang dibutuhkan yaitu:

1. *mit app inventor*
2. *arduino ide*

Selanjutnya terdapat komponen-komponen yang dibutuhkan untuk membuat aplikasi berbasis android pada Sistem Kendali Otomatis Rumah Pintar Berbasis Android.

Tabel 4.1 Komponen Aplikasi

Komponen	Jumlah
<i>Button</i>	17
Label	10
<i>Image</i>	6

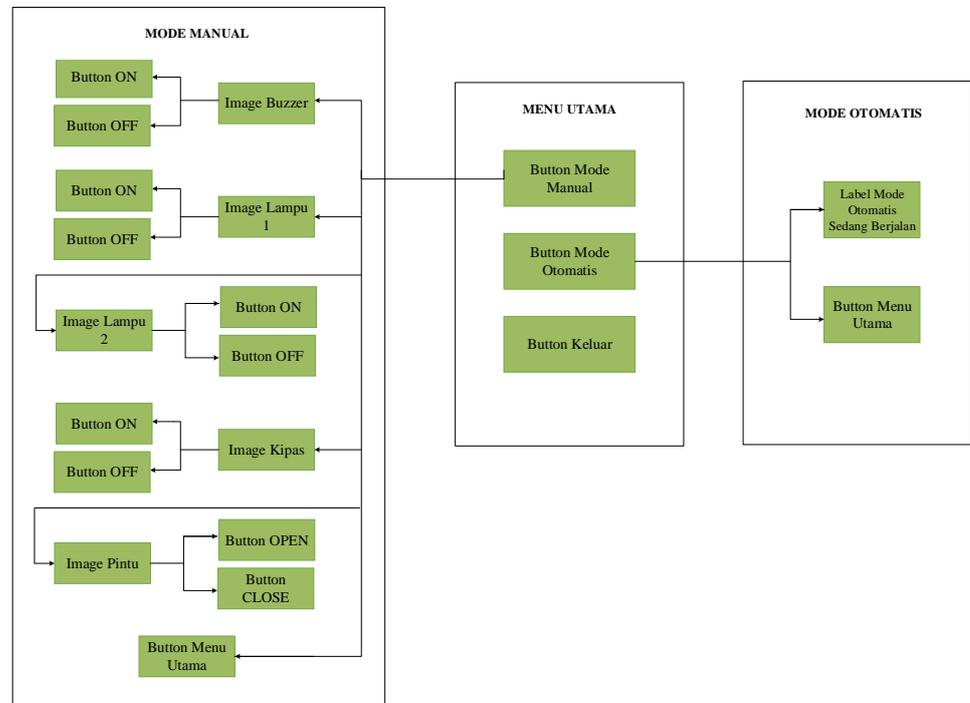
4.3 Perancangan Sistem

Perancangan sistem ini dilakukan dengan perencanaan sistem, implementasi sistem, dan ujicoba sistem. Untuk mempermudah dalam merancang dan membuat Sistem Kendali Otomatis Rumah Pintar Berbasis Android, maka dirancang sebuah diagram blok dan *UML*.

4.3.1 Diagram Blok

Diagram blok digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar dapat lebih memahami

sistem yang akan dibuat, maka perlu dibuatkan gambaran tentang sistem yang berjalan.



Gambar 4.1 Diagram Blok Aplikasi

Tiap-tiap blok dalam gambar memiliki fungsi masing-masing.

1. Mode Manual

Merupakan tombol yang digunakan untuk mengakses saat pemilik rumah ingin menyalakan atau mematikan lampu ataupun kipas dengan menggunakan aplikasi atau juga ketika pemilik rumah sedang berada di luar rumah. Berikut tombol-tombol yang ada di mode manual.

- a. *Buzzer ON*: merupakan tombol yang digunakan untuk menyalakan *Buzzer*.

- b. *Buzzer OFF*: merupakan tombol yang digunakan untuk mematikan *Buzzer*.
- c. Lampu 1 *ON*: merupakan tombol yang digunakan untuk menyalakan lampu 1.
- d. Lampu 1 *OFF*: merupakan tombol yang digunakan untuk mematikan lampu 1.
- e. Lampu 2 *ON*: merupakan tombol yang digunakan untuk menyalakan lampu 2.
- f. Lampu 2 *OFF*: merupakan tombol yang digunakan untuk mematikan lampu 2.
- g. Kipas *ON*: merupakan tombol yang digunakan untuk menyalakan kipas.
- h. Kipas *OFF*: merupakan tombol yang digunakan untuk mematikan kipas.
- i. Pintu *OPEN*: merupakan tombol yang digunakan untuk membuka pintu.
- j. Pintu *CLOSE*: merupakan tombol yang digunakan untuk menutup pintu.
- k. Menu Utama: merupakan tombol yang digunakan untuk keluar dari mode menuju ke menu utama.

2. Mode Otomatis

Merupakan tombol yang digunakan untuk mengakses saat pemilik rumah sedang sibuk melakukan sesuatu dan tidak sempat

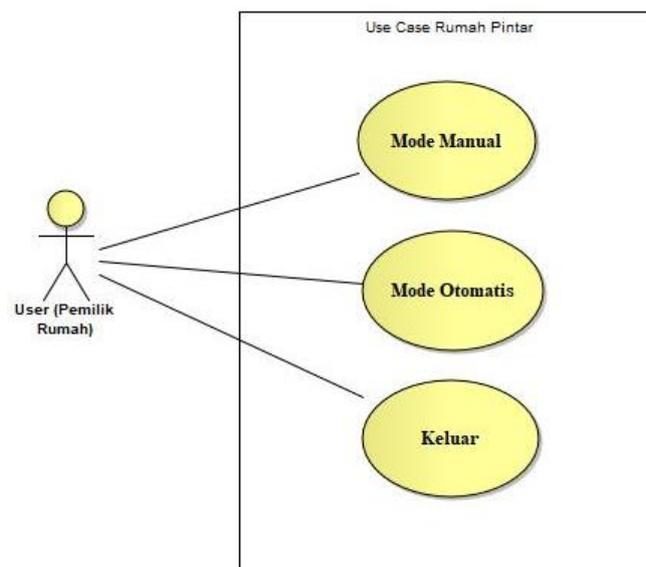
untuk menyalakan dan mematikan lampu ataupun kipas. Berikut tampilan dan tombol yang ada di mode otomatis.

- a. Label Mode Otomatis Sedang Berjalan: merupakan label yang digunakan untuk memberitahu pemilik rumah atau pengguna aplikasi bahwa alat sedang menjalankan mode otomatis.
- b. Menu Utama: merupakan tombol yang digunakan untuk keluar dari mode menuju ke menu utama.

4.3.2 UML (*Unified Modeling Language*)

UML merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. Berikut beberapa diagram *UML* yang digunakan dalam pembuatan sistem.

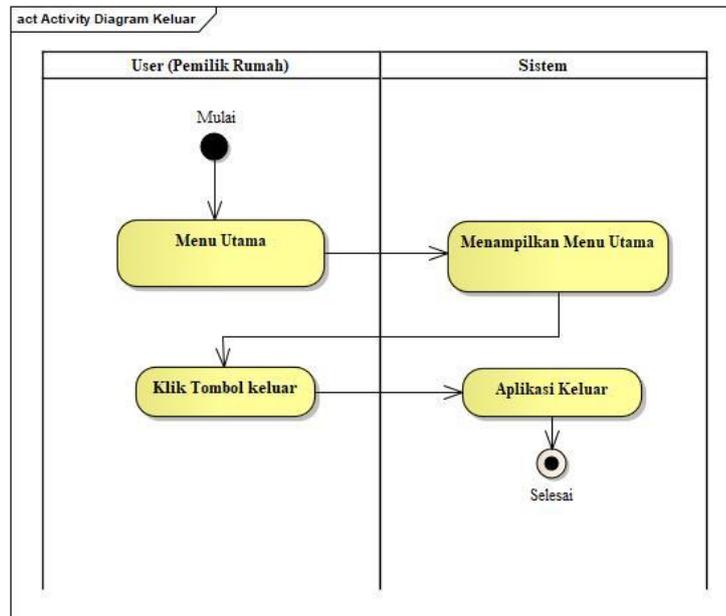
1. *Use Case Diagram*



Gambar 4.2 *Use Case* Rumah Pintar

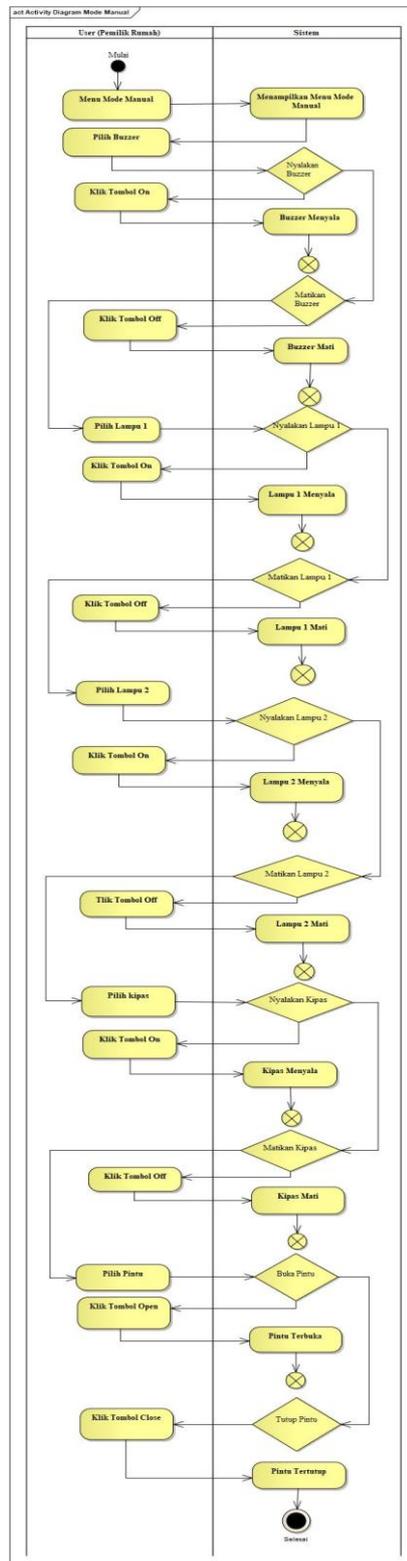
2. Activity Diagram Sistem Kendali Otomatis Rumah Pintar

a. Activity Diagram Keluar



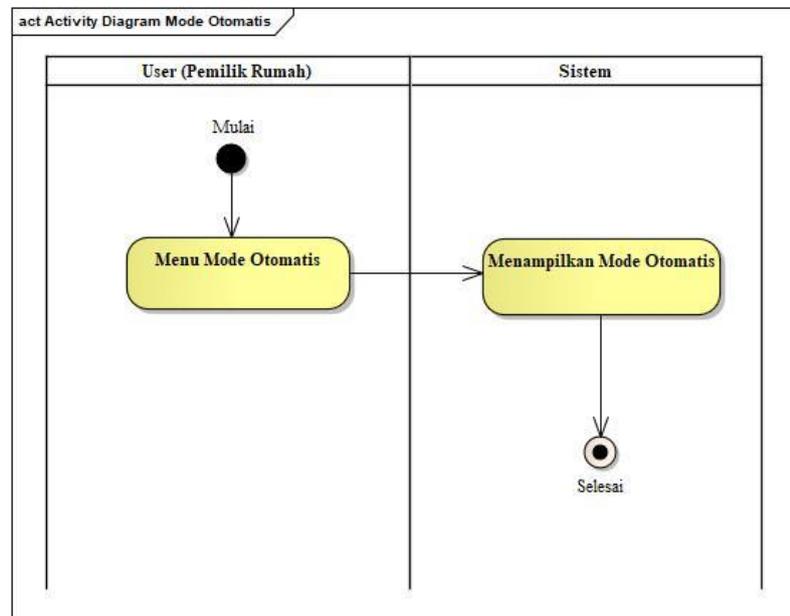
Gambar 4.3 Activity Diagram Keluar

b. *Activity Diagram Mode Manual*



Gambar 4.4 *Activity Diagram Mode Manual*

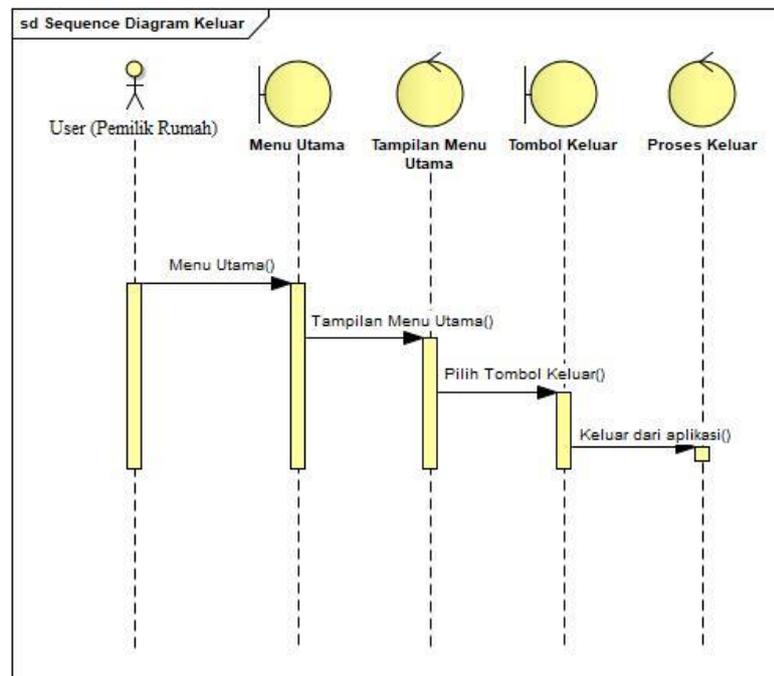
c. *Activity Diagram Mode Otomatis*



Gambar 4.5 *Activity Diagram Mode Otomatis*

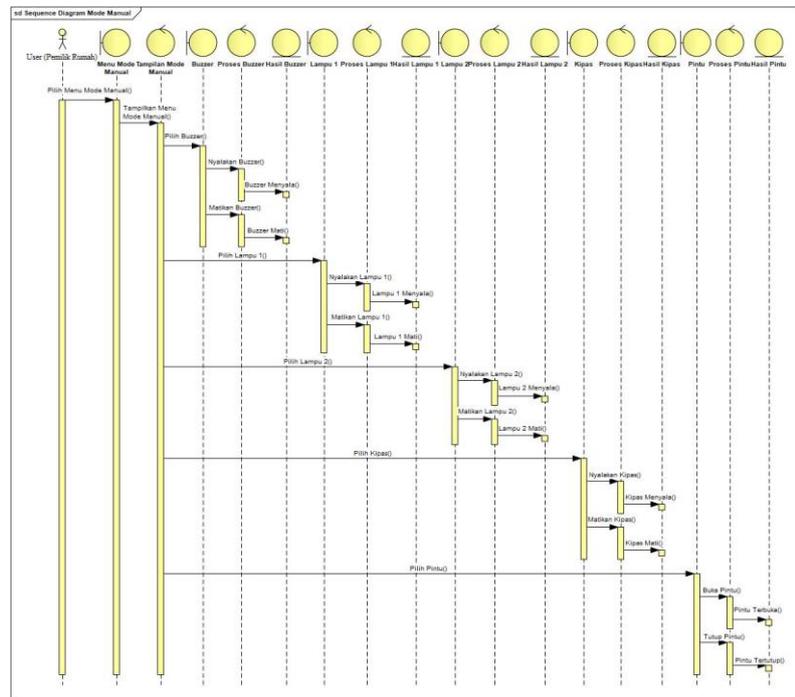
3. *Sequence Diagram Sistem Kendali Otomatis Rumah Pintar*

a. *Sequence Diagram Keluar*



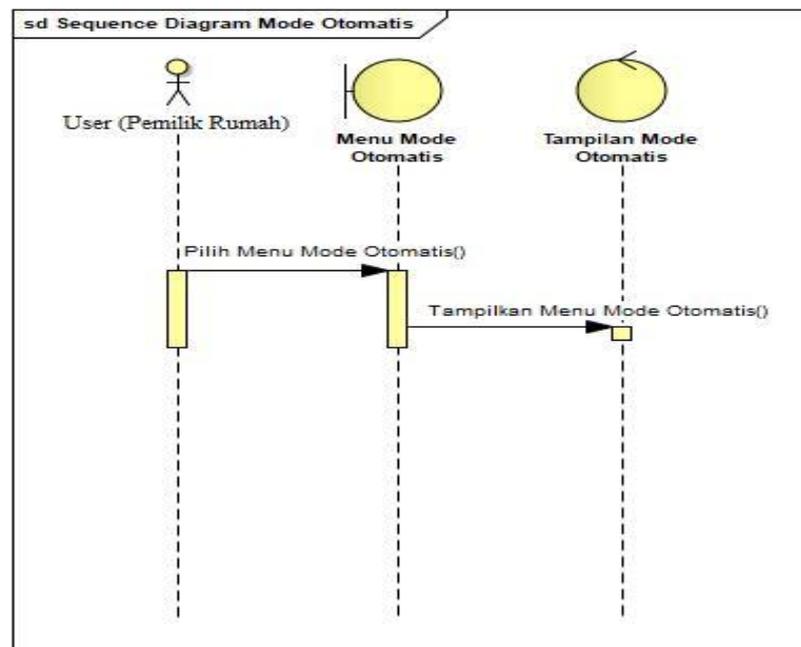
Gambar 4.6 *Sequence Diagram Keluar*

b. *Sequence Diagram Mode Manual*



Gambar 4.7 *Sequence Diagram Mode Manual*

c. *Sequence Diagram Mode Otomatis*



Gambar 4.8 *Sequence Diagram Mode Otomatis*

BAB V

HASIL DAN PEMBAHASAN

5.1 Implementasi Sistem

Setelah melakukan analisis dan perancangan, maka didapatkan analisis kebutuhan perangkat keras (*hardware*), dan analisis kebutuhan perangkat lunak (*software*) untuk membuat Sistem Kendali Otomatis Rumah Pintar Berbasis Android. Tahap selanjutnya yaitu tahap implementasi hasil dan pembahasan aplikasi berbasis android yang akan digunakan pada Sistem Kendali Otomatis Rumah Pintar Berbasis Android dengan menggunakan *software MIT App Inventor* untuk membuat aplikasi berbasis android dan *Arduino IDE* untuk membuat *coding*. Aplikasi berbasis android ini dibuat agar dapat digunakan dan diimplementasikan di rumah-rumah yang mudah untuk dilakukan tindak kejahatan karena kurangnya sistem keamanan.

5.1.1 Implementasi Perangkat Keras

Implementasi perangkat keras merupakan suatu proses perakitan alat yang digunakan dalam pembuatan Sistem Kendali Otomatis Rumah Pintar Berbasis Android.

Adapun perangkat keras yang digunakan untuk memenuhi kriteria dalam pengoperasian alat sebagai berikut:

1. arduino uno r3 dip
2. kabel *jumper* (21 buah)
3. relay 2 *channel* (2 buah)

4. sensor *pir*
5. sensor *ldr*
6. sensor *dht11*
7. *buzzer*
8. *project board*
9. *ac light dimmer 1 channel*
10. *push button*
11. *solenoid door lock*
12. *module wifi esp8266*
13. lampu (2 buah)
14. *fan*

Berikut Sistem Kendali Otomatis Rumah Pintar Berbasis Android yang telah dibuat:



Gambar 5.1 Implementasi Alat

Selanjutnya terdapat tabel penjelasan rangkaian yang telah dibuat pada Sistem Kendali Otomatis Rumah .Pintar Berbasis Android.

Tabel 5.1 Rangkaian Arduino Uno

Arduino Uno	Project Board
Pin 3,3 V	+
Pin 5 V	+
Pin GND	-

Tabel 5.2 Rangkaian Sensor *LDR*

Sensor <i>LDR</i>	Arduino Uno	Project Board
GND		-
In	Pin <i>Analog 0</i>	
VCC		+ (3,3 V)

Tabel 5.3 Rangkaian Sensor *DHT11*

Sensor <i>DHT11</i>	Arduino Uno	Project Board
GND		-
Data	Pin <i>Digital 7</i>	
VCC		+ (3,3 V)

Tabel 5.4 Rangkaian Sensor *PIR*

Sensor <i>PIR</i>	Arduino Uno	Project Board	Buzzer
GND		-	GND
<i>Input Data</i>	Pin <i>Digital 6</i>		
VCC		+ (3,3 V)	
	Pin <i>Digital 5</i>		<i>Input</i>

Tabel 5.5 Rangkaian *Module WiFi ESP8266*

Module <i>WiFi ESP8266</i>	Arduino Uno	Project Board
TX	Pin <i>Digital 8</i>	
CH-PD (EN)		-
VCC		+ (3,3 V)
RX	Pin <i>Digital 9</i>	
GND		-

Tabel 5.6 Rangkaian Kipas

Arduino Uno	Project Board	Relay	Adaptor	Kipas
Pin <i>Digital 10</i>		<i>Input Data</i>	VCC	+
			GND	-
	+ (5V)	VCC		

Arduino Uno	Project Board	Relay	Adaptor	Kipas
	-	GND		

Tabel 5.7 Rangkaian Lampu 1

Arduino Uno	Project Board	Relay	Adaptor	AC Dimmer	Lampu 1
Pin Digital 12		Input Data	VCC		+
			GND		-
	+ (5V)	VCC		VCC	
	-	GND		GND	
Pin Digital 3				PWM	
Pin Digital 2				Z-C	

Tabel 5.8 Rangkaian Lampu 2

Arduino Uno	Project Board	Relay	Adaptor	Lampu 2
Pin Digital 13		Input Data	VCC	+
			GND	-
	+ (5V)	VCC		
	-	GND		

Tabel 5.9 Rangkaian Pintu

Arduino Uno	Project Board	Relay	Adaptor	Pintu
Pin Digital 11		Input Data	VCC	+
			GND	-
	+ (5V)	VCC		
	-	GND		

5.1.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak merupakan proses penerapan pada aplikasi berbasis android sebagai media untuk mengendalikan alat yang telah dibuat. Dalam pengaplikasiannya, aplikasi berbasis android dibuat dengan menggunakan *MIT App Inventor*.

Perangkat lunak yang dapat digunakan untuk mengimplementasikan alat ini adalah:

1. *mit app inventor*
2. *arduino ide*

Berikut tampilan aplikasi berbasis android yang digunakan untuk mengendalikan Sistem Kendali Otomatis Rumah Pintar Berbasis Android.

1. Menu Utama

Menu utama merupakan tampilan awal aplikasi yang berisi pilihan mode yang akan digunakan. Di menu utama terdapat 2 mode yaitu mode manual dan mode otomatis. Selain itu di menu utama juga terdapat tombol keluar untuk keluar dari aplikasi.

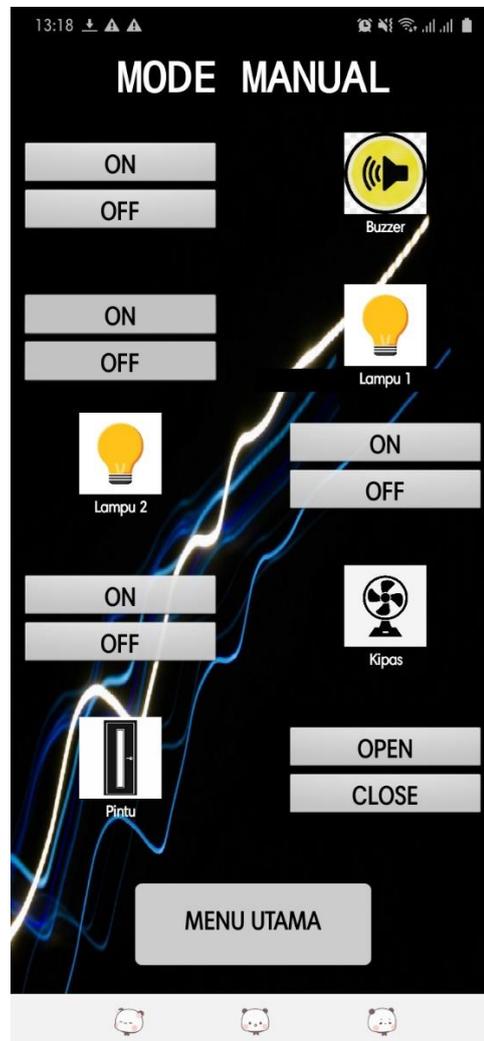


Gambar 5.2 Tampilan Menu Utama Aplikasi Rumah Pintar

2. Mode Manual

Mode Manual merupakan tombol yang digunakan untuk mengakses saat pemilik rumah ingin menyalakan atau mematikan lampu ataupun kipas menggunakan aplikasi saat pemilik rumah sedang berada di luar rumah. Selain itu mode manual juga dapat digunakan ketika pemilik rumah sedang berada di dalam rumah. Mode manual memiliki beberapa tombol yaitu *buzzer on*, *buzzer off*, *lampu1 on*, *lampu1 off*, *lampu2 on*, *lampu2 off*, *kipas on*,

kipas *off*, pintu *open*, pintu *close*, *buzzer*, dan yang terakhir tombol untuk keluar dari mode.



Gambar 5.3 Tampilan Mode Manual Aplikasi Rumah Pintar

3. Mode Otomatis

Mode otomatis merupakan tombol yang digunakan untuk mengakses saat pemilik rumah sedang sibuk dan tidak sempat menyalakan atau mematikan lampu maupun kipas dengan menggunakan aplikasi. Mode ini dapat digunakan ketika pemilik rumah sedang berada di dalam rumah. Mode otomatis hanya

memiliki satu tombol untuk keluar dari mode dan tampilan informasi bahwa pemilik sedang menggunakan mode otomatis.



Gambar 5.4 Tampilan Mode Otomatis Aplikasi Rumah Pintar

5.2 Hasil Pengujian

5.2.1 Pengujian Aplikasi

Pengujian pada Sistem Kendali Otomatis Rumah Pintar Berbasis Android dimaksudkan untuk menguji semua komponen yang

dipakai, baik dari sisi perangkat lunak (*software*) maupun perangkat keras (*hardware*) yang dibuat.

5.2.2 Hasil Pengujian Aplikasi

Tahap pengujian merupakan hal yang dilakukan untuk mengetahui apakah perangkat lunak (*software*) dan perangkat keras (*hardware*) telah berjalan dengan baik dan sesuai dengan apa yang telah diharapkan. Terdapat 2 mode pada tahap pengujian aplikasi yaitu pengujian aplikasi mode manual dan pengujian aplikasi mode otomatis. Berikut hasil pengujian yang telah dilakukan.

1. Pengujian Aplikasi Mode Manual

Pengujian aplikasi mode manual yaitu pengujian dengan menekan *button-button* yang terdapat pada tampilan aplikasi mode manual. Terdapat beberapa pengujian yang dilakukan pada aplikasi mode manual.

a. Pengujian Lampu

Tabel 5.10 Pengujian Lampu

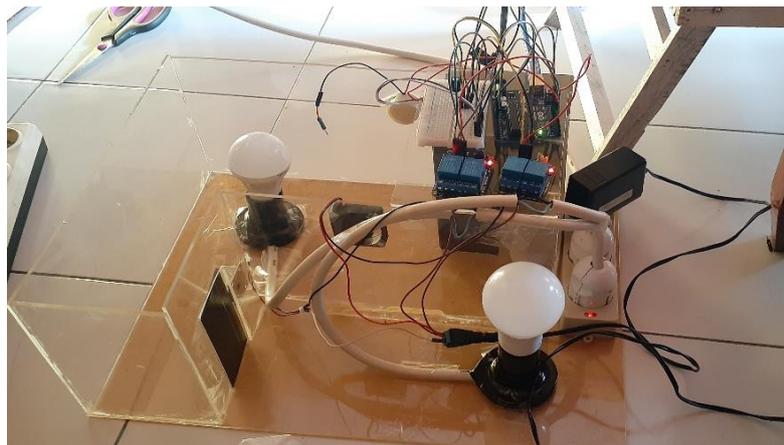
<i>Input</i>	<i>Output</i>	<i>Jarak</i>	<i>Koneksi</i>	<i>Keterangan</i>
<i>Button ON</i> pada Lampu 1	Lampu 1	1 meter dari rumah	Baik	Berhasil Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 1	Lampu 1	1 meter dari rumah	Buruk	Gagal Lampu <i>OFF</i>
<i>Button ON</i> pada Lampu 1	Lampu 1	10 meter dari rumah	Buruk	Gagal Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 1	Lampu 1	10 meter dari rumah	Baik	Berhasil Lampu <i>OFF</i>

<i>Input</i>	<i>Output</i>	Jarak	Koneksi	Keterangan
<i>Button ON</i> pada Lampu 1	Lampu 1	30 meter dari rumah	Baik	Berhasil Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 1	Lampu 1	30 meter dari rumah	Baik	Berhasil Lampu <i>OFF</i>
<i>Button ON</i> pada Lampu 2	Lampu 2	1 meter dari rumah	Baik	Berhasil Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 2	Lampu 2	1 meter dari rumah	Baik	Berhasil Lampu <i>OFF</i>
<i>Button ON</i> pada Lampu 2	Lampu 2	10 meter dari rumah	Baik	Berhasil Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 2	Lampu 2	10 meter dari rumah	Baik	Berhasil Lampu <i>OFF</i>
<i>Button ON</i> pada Lampu 2	Lampu 2	30 meter dari rumah	Baik	Berhasil Lampu <i>ON</i>
<i>Button OFF</i> pada Lampu 2	Lampu 2	30 meter dari rumah	Baik	Berhasil Lampu <i>OFF</i>

Dari hasil pengujian lampu di atas yaitu mendapat satu kegagalan pada saat akan menyalakan lampu 1 pada jarak 10 meter dari rumah. Kegagalan tersebut dikarenakan koneksi yang digunakan mengalami kendala. Berikut tampilan pengujian lampu yang berhasil dinyalakan.



Gambar 5.5 Tampilan Pengujian Lampu Menyala



Gambar 5.6 Tampilan Pengujian Lampu Mati

b. Pengujian Kipas

Tabel 5.11 Pengujian Kipas

<i>Input</i>	<i>Output</i>	<i>Jarak</i>	<i>Koneksi</i>	<i>Keterangan</i>
<i>Button ON</i> pada Kipas	Kipas	1 meter dari rumah	Baik	Berhasil Kipas <i>ON</i>
<i>Button OFF</i> pada Kipas	Kipas	1 meter dari rumah	Baik	Berhasil Kipas <i>OFF</i>

<i>Input</i>	<i>Output</i>	<i>Jarak</i>	<i>Koneksi</i>	<i>Keterangan</i>
<i>Button ON</i> pada Kipas	Kipas	10 meter dari rumah	Baik	Berhasil Kipas <i>ON</i>
<i>Button OFF</i> pada Kipas	Kipas	10 meter dari rumah	Baik	Berhasil Kipas <i>OFF</i>
<i>Button ON</i> pada Kipas	Kipas	30 meter dari rumah	Baik	Berhasil Kipas <i>ON</i>
<i>Button OFF</i> pada Kipas	Kipas	30 meter dari rumah	Baik	Berhasil Kipas <i>OFF</i>

Dari hasil pengujian kipas di atas yaitu *Button ON* dan *OFF* pada kipas yang terdapat pada aplikasi semua berhasil. Karena koneksi yang dipakai baik maka *Button ON* dan *OFF* berhasil menjalankan dan menghentikan kipas. Berikut tampilan pengujian kipas.



Gambar 5.7 Tampilan Pengujian Kipas Memutar



Gambar 5.8 Tampilan Pengujian Kipas Berhenti

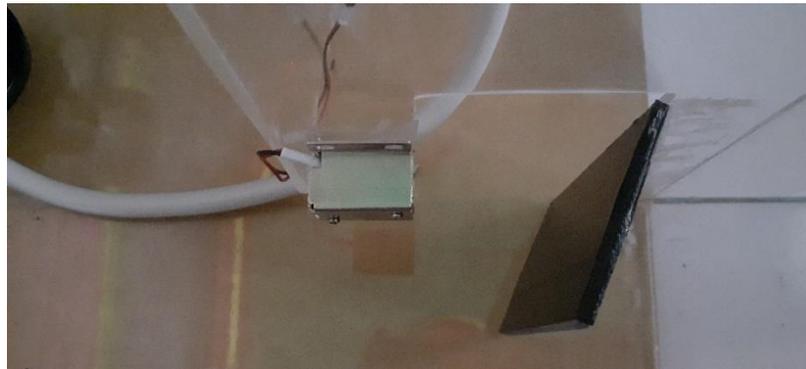
c. Pengujian Pintu

Tabel 5.12 Pengujian Pintu

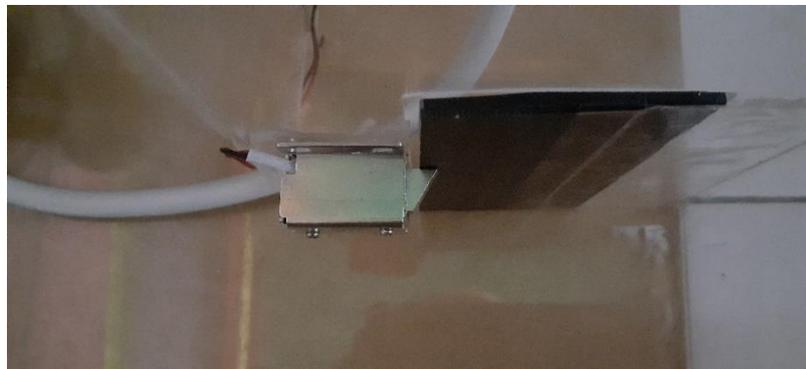
<i>Input</i>	<i>Output</i>	<i>Jarak</i>	<i>Koneksi</i>	<i>Keterangan</i>
<i>Button OPEN pada Pintu</i>	<i>Doorlock</i>	1 meter dari rumah	Buruk	Gagal Pintu <i>OPEN</i>
<i>Button CLOSE pada Pintu</i>	<i>Doorlock</i>	1 meter dari rumah	Baik	Berhasil Pintu <i>CLOSE</i>
<i>Button OPEN pada Pintu</i>	<i>Doorlock</i>	10 meter dari rumah	Baik	Berhasil Pintu <i>OPEN</i>
<i>Button CLOSE pada Pintu</i>	<i>Doorlock</i>	10 meter dari rumah	Baik	Berhasil Pintu <i>CLOSE</i>
<i>Button OPEN pada Pintu</i>	<i>Doorlock</i>	30 meter dari rumah	Baik	Berhasil Pintu <i>OPEN</i>
<i>Button CLOSE pada Pintu</i>	<i>Doorlock</i>	30 meter dari rumah	Baik	Berhasil Pintu <i>CLOSE</i>

Dari hasil pengujian pintu di atas yaitu berhasil atau tidaknya *Button OPEN* dan *CLOSE* pada Pintu yang terdapat pada

aplikasi terdapat kegagalan. Karena koneksi yang dipakai terkadang mengalami kendala, maka *Button OPEN* dan *CLOSE* terdapat satu kegagalan yaitu pada saat membuka pintu pada jarak 1 meter dari rumah. Berikut tampilan pengujian pintu yang berhasil.



Gambar 5.9 Tampilan Pengujian Pintu Terbuka



Gambar 5.10 Tampilan Pengujian Pintu Tertutup

d. Pengujian *Buzzer*

Tabel 5.13 Pengujian *Buzzer*

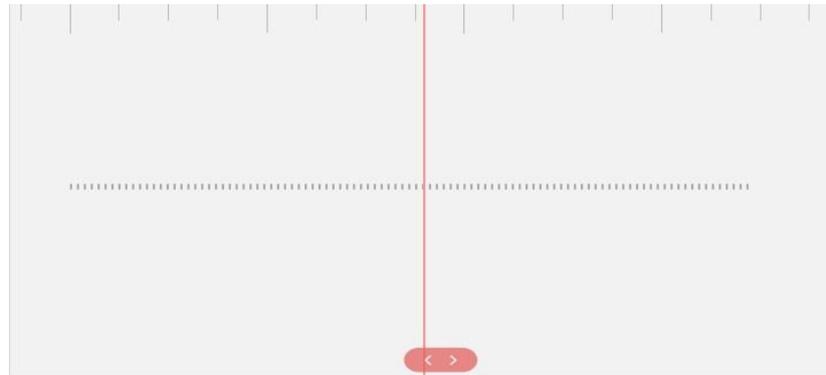
<i>Input</i>	<i>Output</i>	<i>Jarak</i>	<i>Koneksi</i>	<i>Keterangan</i>
<i>Button ON pada Buzzer</i>	<i>Buzzer</i>	1 meter dari rumah	Baik	Berhasil <i>Buzzer ON</i>

<i>Input</i>	<i>Output</i>	Jarak	Koneksi	Keterangan
<i>Button OFF</i> pada <i>Buzzer</i>	<i>Buzzer</i>	1 meter dari rumah	Baik	Berhasil <i>Buzzer OFF</i>
<i>Button ON</i> pada <i>Buzzer</i>	<i>Buzzer</i>	10 meter dari rumah	Baik	Berhasil <i>Buzzer ON</i>
<i>Button OFF</i> pada <i>Buzzer</i>	<i>Buzzer</i>	10 meter dari rumah	Baik	Berhasil <i>Buzzer OFF</i>
<i>Button ON</i> pada <i>Buzzer</i>	<i>Buzzer</i>	30 meter dari rumah	Baik	Berhasil <i>Buzzer ON</i>
<i>Button OFF</i> pada <i>Buzzer</i>	<i>Buzzer</i>	30 meter dari rumah	Baik	Berhasil <i>Buzzer OFF</i>

Dari hasil pengujian *buzzer* di atas yaitu *Button ON* dan *OFF* pada *buzzer* yang terdapat pada aplikasi semua berhasil. Karena koneksi yang dipakai baik maka *Button ON* dan *OFF* berhasil mengaktifkan dan menonaktifkan *buzzer*. Berikut tampilan pengujian *buzzer*.



Gambar 5.11 Tampilan Pengujian *Buzzer* Aktif



Gambar 5.12 Tampilan Pengujian *Buzzer* Nonaktif

2. Pengujian Aplikasi Mode Otomatis

Pengujian aplikasi mode otomatis yaitu pengujian dengan menekan *button* mode otomatis kemudian akan masuk ke tampilan menu otomatis. Pada tampilan menu otomatis terdapat pemberitahuan bahwa pemilik rumah sedang menggunakan mode otomatis.

Dari semua hasil pengujian di atas maka dapat disimpulkan bahwa Sistem Kendali Otomatis Rumah Pintar Berbasis Android dapat dikendalikan secara manual melalui aplikasi berbasis android pada tampilan mode manual dan secara otomatis melalui tampilan mode otomatis berdasarkan *input* dari berbagai macam sensor. Menurut hasil pengujian, Sistem Kendali Otomatis Rumah Pintar Berbasis Android dapat berjalan dengan baik tergantung pada koneksi internet yang dipakai. Apabila koneksi baik, maka sistem akan berjalan dengan lancar. Akan tetapi jika koneksi buruk, maka sistem akan sedikit terlambat dalam menerima instruksi.

BAB VI

SIMPULAN DAN SARAN

6.1 Simpulan

Dari hasil perancangan dan pembuatan Sistem Kendali Otomatis Rumah Pintar Berbasis Android mendapatkan beberapa simpulan, diantaranya:

1. telah dibuat implementasi Sistem Kendali Otomatis Rumah Pintar Berbasis Android dengan ukuran kurang lebih panjang 42 cm, lebar 30 cm, dan tinggi 15 cm.
2. sistem kendali otomatis rumah pintar berbasis android terdapat 2 mode yaitu mode manual dan mode otomatis.
3. pada mode manual sistem kendali otomatis rumah pintar berbasis android dapat dikendalikan dengan menggunakan aplikasi berbasis android berdasarkan instruksi yang diinginkan.
4. pada mode otomatis sistem kendali otomatis rumah pintar berbasis android dapat berjalan secara otomatis dengan inputan dari sensor mendeteksi keadaan yang ada di dalam rumah.
5. pada mode manual aplikasi dapat berjalan dengan baik tergantung dari koneksi internet yang dipakai.
6. menggunakan *module wifi esp8266* sebagai penghubung antara alat dan aplikasi.

7. alat dan aplikasi dapat berkomunikasi melalui *wifi* yang tersambung dengan *ip address* yang sama.

6.2 Saran

Adapun saran yang dapat disampaikan berdasarkan penelitian untuk meningkatkan implementasi kerja alat meliputi:

1. alat ini belum memiliki tampilan untuk menampilkan hasil yang dikeluarkan oleh sensor.
2. pada aplikasi belum terdapat notifikasi yang dapat memberitahu kepada pemilik rumah mengenai keadaan rumah.
3. aplikasi hanya memiliki satu *user* untuk mengendalikannya.
4. dibutuhkan koneksi yang lancar untuk menghubungkan antara alat dengan aplikasi.
5. alat ini belum memiliki baterai cadangan, jadi jika listrik rumah mati alat juga ikut mati.

DAFTAR PUSTAKA

- [1] M. Muslihudin, W. Renvilia, Taufiq, A. Andoyo, and F. Susanto, "Implementasi Aplikasi Rumah Pintar Berbasis Android Dengan Arduino Microcontroller," *J. Keteknikan dan Sains*, vol. 1, no. 1, pp. 23–31, 2018.
- [2] R. E. Putri and D. Yendri, "Sistem Pengontrolan Dan Keamanan Rumah Pintar (Smart Home) Berbasis Android," *J. Inf. Technol. Comput. Eng.*, vol. 2, no. 01, pp. 1–6, 2018, doi: 10.25077/jitce.2.01.1-6.2018.
- [3] A. S. Romoadhon and D. R. Anamisa, "Sistem Kontrol Peralatan Listrik pada Smart Home Menggunakan Android," *Rekayasa*, vol. 10, no. 2, p. 116, 2017, doi: 10.21107/rekayasa.v10i2.3613.
- [4] F. Masykur and F. Prasetyowati, "Perancangan Aplikasi Rumah Pintar," *SiTekin*, vol. 14, no. 1, pp. 93–100, 2016.
- [5] M. W. Sari and H. Hardyanto, "Implementasi Aplikasi Monitoring Pengendalian Pintu Gerbang Rumah Menggunakan App Inventor Berbasis Android," *Eksis*, vol. 09, no. 1, pp. 20–28, 2016.
- [6] A. R. Azka, E. D. Marindani, and R. D. Nyoto, "Rancang Bangun Sistem Pengendali Smarthome menggunakan Mikrokontroler dengan Speech Command pada Smarthome Android," *J. Sist. dan Teknol. Inf.*, vol. 6, no. 2, p. 82, 2018, doi: 10.26418/justin.v6i2.24643.
- [7] V. Vincent, J. V. Harryanto, A. M. Lubis, and J. W. Simatupang, "Kotak Kendali Perangkat Elektronik Nirkabel untuk Aplikasi Smart Home," *J. Telekomun. dan Komput.*, vol. 10, no. 2, p. 67, 2020, doi: 10.22441/incomtech.v10i2.8264.
- [8] J. T. Komputer, P. Harapan, and B. Tegal, "Unified Modeling Language (UML) Model Untuk Pengembangan Sistem Informasi Akademik Berbasis Web," vol. 03, no. 01, pp. 126–129, 2018.
- [9] . Z., . Z., and K. Amri, "Desain Sistem Kontrol Penyalaan Lampu dan Perangkat Elektronik untuk Meniru Keberadaan Penghuni Rumah," *J. Nas. Tek. Elektro*, vol. 5, no. 1, 2016, doi: 10.20449/jnte.v5i1.182.

- [10] M. I. KURNIAWAN, U. SUNARYA, and R. TULLOH, "Internet of Things : Sistem Keamanan Rumah berbasis Raspberry Pi dan Telegram Messenger," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 6, no. 1, p. 1, 2018, doi: 10.26760/elkomika.v6i1.1.
- [11] A. H. Saptadi, "Perbandingan Akurasi Pengukuran Suhu dan Kelembaban Antara Sensor DHT11 dan DHT22 Studi Komparatif pada Platform ATMEL AVR dan Arduino," *J. Inform. dan Elektron.*, vol. 6, no. 2, 2015, doi: 10.20895/infotel.v6i2.73.
- [12] Ματινα, "No TitleΕΛΕΝΗ," *Αγαη*, vol. 8, no. 5, p. 55, 2019.
- [13] R. Y. Endra, "Smart Room Menggunakan Internet Of Things Untuk Efisiensi Biaya dan Keamanan Ruangan," 2019, doi: 10.31219/osf.io/gz6mb.
- [14] M. T. Hidayat, "Rancang Bangun Pemanas Suhu Kandang Anak Ayam Broiler Secara Otomatis Berbasis Mikrokontroler Atmega 2560," *Sci. Electro*, vol. 10, no. 1, pp. 50–55, 2019.
- [15] A. Iskandar, M. Muhajirin, and L. Lisah, "Sistem Keamanan Pintu Berbasis Arduino Mega," *J. Inform. Upgris*, vol. 3, no. 2, pp. 99–104, 2017, doi: 10.26877/jiu.v3i2.1803.
- [16] P. S. Informatika, "Naskah publikasi rancang bangun smart home dengan konsep internet of things (iot) berbasis android," 2020.
- [17] S. Samsugi, A. Ardiansyah, and D. Kastutara, "Arduino dan Modul Wifi ESP8266 sebagai Media Kendali Jarak Jauh dengan antarmuka Berbasis Android," *J. Teknoinfo*, vol. 12, no. 1, p. 23, 2018, doi: 10.33365/jti.v12i1.42.

LAMPIRAN

Lampiran 1 Surat Kesiediaan Pembimbing TA

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Ida Afriliana, ST, M.Kom
NIDN : 0624047703
NIPY : 12.013.168
Jabatan Struktural : Koordinator Akademik DIII Teknik Komputer
Jabatan Fungsional : Asisten Ahli

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut :

Nama : Irma Luthfianalela
NIM : 18040151
Program Studi : DIII Teknik Komputer
Judul TA : Sistem Kendali Otomatis Rumah Pintar Berbasis Android

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, Januari 2021

Mengetahui,

Kepala Program Studi DIII Teknik Komputer



S.Pd, M.Kom
NIPY. 07.011.083

Dosen Pembimbing I,



Ida Afriliana, ST, M.Kom
NIPY. 12.013.168

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Nurohim, S.ST, M.Kom
NIDN : 0625067701
NIPY : 09.017.342
Jabatan Struktural : Koordinator Lab Hardware
Jabatan Fungsional : Asisten Ahli

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut :

Nama : Irma Luthfianalela
NIM : 18040151
Program Studi : DIII Teknik Komputer
Judul TA : Sistem Kendali Otomatis Rumah Pintar Berbasis Android

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, Januari 2021

Mengetahui,

Kepala Program Studi DIII Teknik Komputer



Ras, S.Pd, M.Kom
NIPY. 07.011.083

Dosen Pembimbing II,

Nurohim, S.ST, M.Kom
NIPY. 09.017.342

Lampiran 2 *Block Code* Aplikasi

The image displays 21 individual block codes for different buttons in an application. Each code block follows a similar structure: a 'when' trigger (e.g., 'when Button1 .Click'), a 'do' block containing a sequence of actions. The actions typically include setting a 'Web1' URI to a specific URL, calling 'Web1 .Get', and playing a specific sound. Some codes also include 'open another screen' or 'close screen' actions. The URLs used are: 'http://192.168.43.60/manual/1', 'http://192.168.43.60/otomats/1', 'http://192.168.43.60/buzzer/1', 'http://192.168.43.60/buzzer/0', 'http://192.168.43.60/lampu/1', 'http://192.168.43.60/lampu/0', 'http://192.168.43.60/lampu/2/1', 'http://192.168.43.60/lampu/2/0', 'http://192.168.43.60/kipas/1', and 'http://192.168.43.60/kipas/0'. The sounds played are labeled Sound9, Sound1, Sound3, Sound5, Sound4, Sound6, Sound7, and Sound8. Screen names mentioned are 'Screen2' and 'Screen3'. The buttons are labeled Button1, Button2, Button3, Button23, Button24, Button11, Button12, Button13, Button14, Button17, Button18, Button13, Button14, Button17, Button18, Button20, Button21, and Button4.

```
when Button1 .Click
do
  set Web1 .Uri to "http://192.168.43.60/manual/1"
  call Web1 .Get
  open another screen screenName "Screen2"

when Button2 .Click
do
  set Web1 .Uri to "http://192.168.43.60/otomats/1"
  call Web1 .Get
  open another screen screenName "Screen3"

when Button3 .Click
do
  close application

when Button23 .Click
do
  set Web1 .Uri to "http://192.168.43.60/buzzer/1"
  call Web1 .Get
  call Sound9 .Play

when Button24 .Click
do
  set Web1 .Uri to "http://192.168.43.60/buzzer/0"
  call Web1 .Get
  call Sound9 .Play

when Button11 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu/1"
  call Web1 .Get
  call Sound1 .Play

when Button12 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu/0"
  call Web1 .Get
  call Sound2 .Play

when Button13 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu/2/1"
  call Web1 .Get
  call Sound3 .Play

when Button14 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu/2/0"
  call Web1 .Get
  call Sound4 .Play

when Button17 .Click
do
  set Web1 .Uri to "http://192.168.43.60/kipas/1"
  call Web1 .Get
  call Sound5 .Play

when Button18 .Click
do
  set Web1 .Uri to "http://192.168.43.60/kipas/0"
  call Web1 .Get
  call Sound6 .Play

when Button13 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu2/1"
  call Web1 .Get
  call Sound3 .Play

when Button14 .Click
do
  set Web1 .Uri to "http://192.168.43.60/lampu2/0"
  call Web1 .Get
  call Sound4 .Play

when Button17 .Click
do
  set Web1 .Uri to "http://192.168.43.60/kipas/1"
  call Web1 .Get
  call Sound5 .Play

when Button18 .Click
do
  set Web1 .Uri to "http://192.168.43.60/kipas/0"
  call Web1 .Get
  call Sound6 .Play

when Button20 .Click
do
  set Web1 .Uri to "http://192.168.43.60/pintu/1"
  call Web1 .Get
  call Sound7 .Play

when Button21 .Click
do
  set Web1 .Uri to "http://192.168.43.60/pintu/0"
  call Web1 .Get
  call Sound8 .Play

when Button4 .Click
do
  set Web1 .Uri to "http://192.168.43.60/k.manual/1"
  call Web1 .Get
  close screen
```

Lampiran 3 Source Code

```
#include <DHT.h>
#include <DHT_U.h>
#include "WiFiEsp.h"
#include "SoftwareSerial.h"

SoftwareSerial esp8266(8, 9);
char NETWORK_SSID[] = "SamsungA20";
char NETWORK_PASSWORD[] = "selamatpagicikgu";

int status = WL_IDLE_STATUS;
WiFiEspServer server(80);

#define DHT1_PIN 7 //sensor dht
#define DHTTYPE DHT11
DHT dht(DHT1_PIN, DHTTYPE);
int ldr1 = A0; //sensor ldr
int ldr2 = A1; //sensor ldr
int pir = 5; //sensor pir
int bz = 6; //buzzer
int selenoid = 10; //relay untuk selenoid
int kipas = 11; //relay untuk kipas
int lampu1 = 12; //relay untuk lampu1
int lampu2 = 13; //relay untuk lampu2

int statuspir; //untuk sensor pir
int sv; //untuk sensor ldr1
int dimmer;
int mnl; //mode
int otmts; //mode

void setup() {
  Serial.begin(9600);
  esp8266.begin(9600);

  WiFi.init(&esp8266);
  Serial.println("Koneksi ke jaringan dilakukan.....");
  Serial.print("SSID: ");
  Serial.println(NETWORK_SSID);

  status = WiFi.begin(NETWORK_SSID, NETWORK_PASSWORD);
  if (status != WL_CONNECTED){
    Serial.println("gagal melakukan koneksi WiFi!");
    while(true);
  }
  server.begin();
  IPAddress ip = WiFi.localIP();
  Serial.print("Alamat IP: ");
  Serial.println(ip);

  Serial.println("siap melayani permintaan!");

  dht.begin();
  pinMode(ldr1, INPUT);
  pinMode(pir, INPUT);
```

```

    pinMode(kipas,OUTPUT);
    pinMode(lampu1, OUTPUT);
    pinMode(lampu2, OUTPUT);
    pinMode(bz, OUTPUT);
    pinMode(solenoid, OUTPUT);
}

void loop() {
    rumah();
}

void rumah(){
    delay(1000);
    WiFiEspClient klien = server.available();
    if(klien) {
        Serial.println("Permintaan baru...");
        boolean barisSekarangKosong = true;
    }
    String req = klien.readStringUntil('\r');
    Serial.println(req);
    klien.flush();
    if(req.indexOf("/manual/1") != -1){
        mnl = HIGH;
        while(mnl = HIGH){
            Serial.println("mode manual");
            WiFiEspClient klien = server.available();
            if(klien) {
                Serial.println("Permintaan baru...");
                boolean barisSekarangKosong = true;
            }
            String req = klien.readStringUntil('\r');
            Serial.println(req);
            klien.flush();
            int val = LOW;
            delay(250);
            if (req.indexOf("/lampu1/1") != -1)
            {
                Serial.println("Lampu = ON");
                digitalWrite(lampu1, HIGH);
                val = HIGH;
            }
            else if (req.indexOf("/lampu1/0") != -1)
            {
                Serial.println("Lampu = OFF");
                digitalWrite(lampu1, LOW);
                val = LOW;
            }
            if (req.indexOf("/lampu2/1") != -1)
            {
                Serial.println("Lampu = ON");
                digitalWrite(lampu2, HIGH);
                val = HIGH;
            }
            else if (req.indexOf("/lampu2/0") != -1)
            {
                Serial.println("Lampu = OFF");
            }
        }
    }
}

```

```

        digitalWrite(lampu2, LOW);
        val = LOW;
    }
    if (req.indexOf("/kipas/1") != -1)
    {
        Serial.println("Kipas = ON");
        digitalWrite(kipas, HIGH);
        val = HIGH;
    }
    else if (req.indexOf("/kipas/0") != -1)
    {
        Serial.println("Kipas = OFF");
        digitalWrite(kipas, LOW);
        val = LOW;
    }
    if (req.indexOf("/pintu/1") != -1)
    {
        Serial.println("pintu = ON");
        digitalWrite(solenoid, HIGH);
        val = HIGH;
    }
    else if (req.indexOf("/pintu/0") != -1)
    {
        Serial.println("pintu = OFF");
        digitalWrite(solenoid, LOW);
        val = LOW;
    }
    if (req.indexOf("/buzzer/1") != -1)
    {
        Serial.println("buzzer = ON");
        digitalWrite(bz, HIGH);
        val = HIGH;
    }
    else if (req.indexOf("/buzzer/0") != -1)
    {
        Serial.println("buzzer = OFF");
        digitalWrite(bz, LOW);
        val = LOW;
    }
    delay(250);
    klien.println("HTTP/43.60 200 OK");
    klien.println("Content-Type: text/html");
    klien.println(""); // IMPORTANT
    klien.println("<!DOCTYPE HTML>");
    klien.println("<html>");
    klien.print("Status of the gpio: ");
    if(val == LOW)
    {
        klien.print("ON");
    }
    else
    {
        klien.print("OFF");
    }
    delay(1500);
    if(req.indexOf("/k.manual/1") != -1)

```

```

        {
            break;
            mn1 = LOW;
            Serial.println("Kembali Ke Menu Utama");
        }
    else
    {
        Serial.println("masukan perintah");
    }
}
}
if(req.indexOf("/k.otomatis/1") != -1){
    otmts = HIGH;
    while(otmts = HIGH){
        Serial.println("mode otomatis");
        WiFiEspClient klien = server.available();
        if(klien) {
            Serial.println("Permintaan baru...");
            boolean barisSekarangKosong = true;
        }
        String req = klien.readStringUntil('\r');
        Serial.println(req);
        klien.flush();
        int val = LOW;
        delay(250);
        float temp = dht.readTemperature();
        sv=analogRead(ldr1);
        sv=analogRead(ldr1);
        statuspir=digitalRead(pir);
        Serial.print("Intensitas cahaya saat ini = ");
        Serial.println(sv);
        Serial.print("Suhu saat ini          = ");
        Serial.println(temp);
        if(sv<=160){
            digitalWrite(lampul, HIGH);
        }else if(sv<=161){
            digitalWrite(lampul, LOW);
        }
        if(temp>=32){
            digitalWrite(kipas, HIGH);
        }else if(temp>32){
            digitalWrite(kipas, LOW);
        }
        if(statuspir == HIGH){
            digitalWrite(bz, HIGH);
        }else if(statuspir == LOW){
            digitalWrite(bz, LOW);
        }
        if(req.indexOf("/k.otomatis/1") != -1)
        {
            otmts = LOW;
        }
    }
}
}
}

```

Lampiran 4 Dokumentasi Observasi

