

**PERANCANGAN SISTEM KEAMANAN PADA *SMART HOME*
MENGUNAKAN *VOICE COMMAND* DENGAN
KONEKTIVITAS BLUETOOTH**

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Mohammad Kholili Adi Putra
NIM: 135150307111047



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

PERANCANGAN SISTEM KEAMANAN PADA *SMART HOME* MENGGUNAKAN *VOICE COMMAND* DENGAN KONEKTIVITAS BLUETOOTH

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Mohammad Kholili Adi Putra

NIM: 135150307111047

Skripsi ini telah diuji dan dinyatakan lulus pada
03 Agustus 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Sabriansyah Rizqika Akbar, S.T, M.Eng.

NIP. 19820809 201212 1 004

Gembong Edhi Setyawan, S.T, M.T.

NIK: 201208 761201 1 001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D.

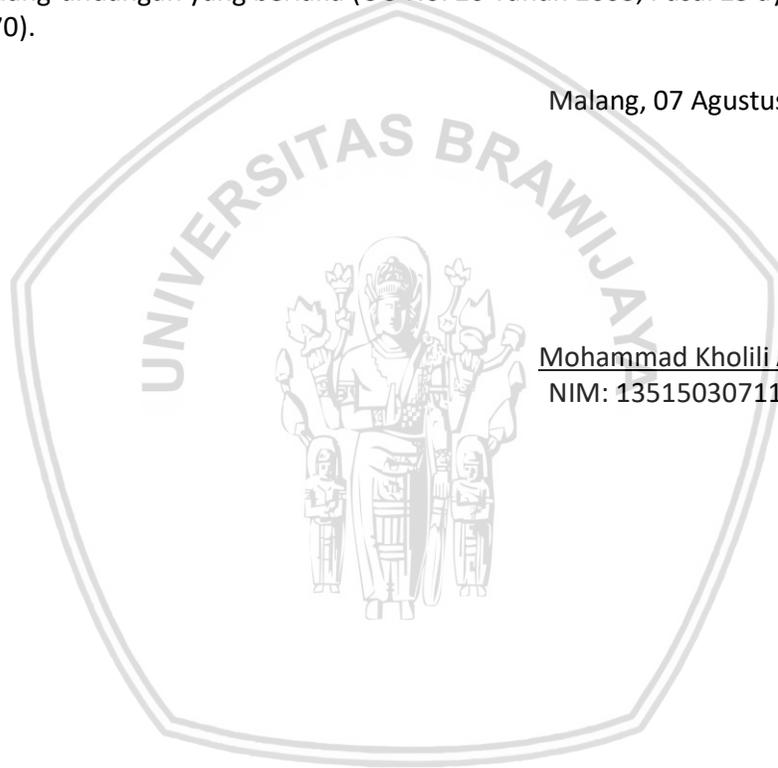
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 07 Agustus 2018



Mohammad Kholili Adi Putra
NIM: 135150307111047



KATA PENGANTAR

Alhamdulillah, puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik serta hidayah-Nya, sehingga peneliti dapat menyelesaikan laporan skripsi dengan judul “Perancangan sistem keamanan pada *smart home* menggunakan *voice command* dengan konektivitas bluetooth” dapat diselesaikan dengan baik.

Dalam pelaksanaan dan penyusunan laporan skripsi ini, tidak sedikit hambatan yang penulis hadapi. Namun penulis menyadari bahwa kelancaran dalam penyusunan laporan ini tidak lain berkat bantuan, dorongan dan bimbingan dari berbagai pihak, sehingga kendala-kendala yang penulis hadapi dapat teratasi. Penghargaan dan terima kasih yang sebesar-besarnya penulis sampaikan kepada:

1. Bapak Gatot priyanto, Ibu Siti Juwariyah selaku orang tua, nenek supini dan tante rohma yang tiada henti memberikan semangat dan dorongan baik berupa doa maupun materi selama penulis melakukan penelitian.
2. Bapak Sabriansyah Rizqika Akbar, S.T, M.Eng dan Bapak Gembong Edhi Setyawan, S.T, M.T selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dengan mengarahkan penulis sehingga dapat menyelesaikan penelitian.
3. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Informatika
4. Bapak Sabriansyah Rizqika Akbar, S.T., M.Eng. selaku Ketua Program Studi Teknik Komputer.
5. Teman-Teman kontrakan : Noor, Cholilulloh, Delta, Kresana, Bagus, Indra, Fajar, Willy, Zamroni, Alfian, Rizki, Adnan, Anang, Tegar, Yogi, Billy. Yang selalu memberikan semangat.
6. Teman-teman teknik komputer angkatan 2013 yang telah banyak memberi bantuan dan dorongannya selama penulis menempuh studi di fakultas teknik komputer universitas Brawijaya dan selama penyelesaian penelitian ini.

Akhir kata penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat banyak kekurangan yang perlu disempurnakan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat penulis harapkan. Semoga skripsi ini bermanfaat bagi semua pihak yang membacanya.

Malang, 07 Agustus 2018

Mohammad Kholili Adi Putra
29adiputra@gmail.com

ABSTRAK

Sistem keamanan rumah merupakan keharusan untuk setiap rumah, salah satu contoh sistem keamanan rumah yang masih banyak digunakan adalah alarm rumah. Tetapi seiring dengan perkembangan teknologi saat ini sistem keamanan rumah semakin banyak jenisnya dan akan terus berjalan perkembangannya. Kurangnya tingkat keamanan menjadikan sering terjadinya pencurian dan pembobolan pada rumah. Terkadang kita sudah yakin sudah mengunci rumah kita, namun pada kenyataannya kasus pembobolan rumah pada rumah saat ini dikarenakan dengan mudahnya para pencuri membuka pengunci pintu yang terpasang hanya dengan seutas kawat ataupun kunci tiruan lainnya. Berdasarkan masalah tersebut diperlukan adanya pengaman pintu rumah yang dapat dikontrol menggunakan *smartphone* android dengan menggunakan perintah suara. *Voice command recognition system* adalah salah satu konsep pengendalian perangkat elektronik dengan menggunakan perintah suara. Dengan menerapkan teknologi *Google Voice Command*, sistem ini dapat menghidupkan dan mematikan peralatan rumah tangga dengan bantuan dari mikrokontroler Arduino dan Bluetooth. Pada penelitian ini menjelaskan perancangan sistem keamanan rumah menggunakan voice command yang digunakan pada aplikasi android dengan konektivitas Bluetooth yang dapat mengontrol solenoid door lock, PIR, dan RTC. Dari hasil pengujian, sistem ini memiliki nilai keberhasilan 93,33% pada pengujian fungsional dan pengujian pengolahan masukan pengguna dalam mengontrol solenoid door lock. Pada pengujian sensor PIR, nilai keberhasilan mencapai 100%. Selanjutnya pengujian RTC, nilai keberhasilan mencapai 100%.

Kata Kunci : Sistem keamanan rumah, *Voice command recognition*, solenoid door lock, PIR, RTC.

ABSTRACT

Home security system is a must for every house, for instance home alarm that is still many used. But, along with the development of the present technology, home security system has various kinds and will keep moving for the development. The lack of security level has the impact of theft and burglary in house. Sometimes, the house owner thinks that he has locked his house, but in reality, the burglary case in house now is due to the easy thing for the burglar to open the installed lock door by using a wire or other imitated keys. Based on the problem, it is need to add the door safety that can be controlled by using Android smartphone with *voice command*. *Voice command recognition system* is one concept of controlling the electronic devices by using *voice command*. By applying the Google Voice Command technology, the system can turn on and turn of the home equipment by the support of microcontroller and Bluetooth. The research explains the design of home security system using voice command that is used by the android application with the Bluetooth connection to control the solenoid door lock, PIR, and RTC. From the test result, the system has value of success 93,33% in functional testing and processing testing the user input in controlling solenoid door lock. In sensor PIR testing, the value of success is up to 100%. The next test is RTC, the value of success is up to 100%.

Keywords : *Home Security system, Voice command recognition, solenoid door lock, PIR, RTC.*



DAFTAR ISI

PENGESAHAN	i
PERNYATAAN ORISINALITAS	ii
KATA PENGANTAR.....	iii
ABSTRAK	iv
ABSTRACT	v
DAFTAR ISI	vi
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	ix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan	3
1.4 Manfaat	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAN	5
2.1 Tinjauan Pustaka	5
2.2 Dasar Teori	6
2.2.1 Android.....	6
2.2.2 Bluetooth.....	7
2.2.3 <i>Voice command</i>	7
2.2.4 Arduino IDE	7
2.2.5 MIT Appinventor 2	8
2.2.6 <i>Smart home</i>	10
BAB 3 METODOLOGI	11
3.1 Metode Penelitian.....	11
3.2 Analisis Kebutuhan	11
3.3 Perancangan dan Implementasi	12
3.4 Pengujian dan Analisis.....	12

3.5 Kesimpulan dan Saran	12
BAB 4 ANALISIS KEBUTUHAN	13
4.1 Gambaran Umum Sistem	13
4.2 Kebutuhan Sistem	13
4.2.1 Kebutuhan Perangkat Keras	13
4.2.2 Kebutuhan Perangkat Lunak	19
4.3 Kebutuhan Fungsional	20
4.4 Kebutuhan Non-Fungsional	20
4.4.1 Karakteristik yang Berbeda tiap Individu	20
4.4.2 Lingkungan Pengoperasian Sistem	21
4.4.3 Asumsi Ketergantungan tiap Komponen	21
BAB 5 PERANCANGAN DAN IMPLEMENTASI	22
5.1 Gambaran Umum Sistem	22
5.2 Perancangan Sistem	23
5.2.1 Perancangan Perangkat Keras	23
5.2.2 Perancangan Perangkat Lunak	28
5.3 Implementasi Sistem	34
5.3.1 Implementasi Perangkat Keras	34
5.3.2 Implementasi Perangkat Lunak	36
BAB 6 PENGUJIAN DAN ANALISIS.....	45
6.1 Pengujian <i>voice command</i>	45
6.1.1 Tujuan	45
6.1.2 Prosedur	45
6.1.3 Hasil dan Analisis	46
6.1.4 Analisis Hasil Pengujian.....	46
6.2 Pengujian Bluetooth	47
6.2.1 Tujuan	47
6.2.2 Prosedur.....	47
6.2.3 Hasil dan Analisis	49
6.2.4 Analisis Hasil Pengujian.....	51
6.3 Pengujian solenoid door lock	51
6.3.1 Tujuan	51
6.3.2 Prosedur.....	52



6.3.3 Hasil Pengujian.....	53
6.3.4 Analisis Hasil Pengujian.....	62
6.4 Pengujian PIR.....	65
6.4.1 Tujuan.....	65
6.4.2 Prosedur.....	66
6.4.3 Hasil Pengujian.....	67
6.4.4 Analisis Hasil Pengujian.....	69
6.5 Pengujian RTC.....	70
6.5.1 Tujuan.....	70
6.5.2 Prosedur.....	71
6.5.3 Hasil dan Analisis.....	72
6.5.4 Analisis Hasil Pengujian.....	75
BAB 7 KESIMPULAN DAN SARAN.....	77
7.1 Kesimpulan.....	77
7.2 Saran.....	77
Daftar Pustaka.....	79



DAFTAR TABEL

Tabel 4. 1 Spesifikasi mikrokontroler.....	14
Tabel 4. 2 Spesifikasi sensor PIR	14
Tabel 4. 3 Spesifikasi Solenoid Door Lock	15
Tabel 4. 4 Spesifikasi Relay	16
Tabel 4. 5 Spesifikasi Modul Bluetooth.....	17
Tabel 4. 6 Spesifikasi RTC	17
Tabel 4. 7 Spesifikasi Android device	18
Tabel 4. 8 Spesifikasi komputer windows.....	18
Tabel 5. 1 Kabel hubungan RTC dengan Arduino	24
Tabel 5. 2 Kabel hubungan Bluetooth dengan Arduino	25
Tabel 5. 3 Kabel hubungan sensor PIR dengan Arduino	25
Tabel 5. 4 Kabel hubungan Relay, Solenoid Door Lock dan Adaptor dengan Arduino....	26
Tabel 6. 1 Hasil pengujian mengunci semua pintu selama waktu tertentu	47
Tabel 6.2 Hasil pengujian mengunci semua pintu selama waktu tertentu	51
Tabel 6. 5 Hasil pengujian menghidupkan semua sistem.....	63
Tabel 6.6 Hasil pengujian mematikan semua sistem.....	63
Tabel 6.7 Hasil pengujian membuka pintu depan	63
Tabel 6.8 Hasil pengujian membuka pintu kamar	64
Tabel 6.9 Hasil pengujian mengunci pintu depan.....	64
Tabel 6.10 Hasil pengujian mengunci pintu kamar	64
Tabel 6.11 Data pengujian solenoid door lock	65
Tabel 6.12 Hasil pengujian sensor PIR	70
Tabel 6.13 Data pengujian solenoid door lock	70
Tabel 6.14 Hasil pengujian mengunci semua pintu selama waktu tertentu	75
Tabel 6.15 Hasil pengujian mengunci semua pintu pada waktu tertentu.....	75
Tabel 6. 16 Data pengujian RTC	76

DAFTAR GAMBAR

Gambar 2. 1 Android.....	6
Gambar 2. 2 MIT App Inventor	8
Gambar 2. 3 Palette MIT App Inventor.....	8
Gambar 2. 4 Hello World Blocks	9
Gambar 3. 1 Alur metodologi	11
Gambar 5. 1 Gambaran umum sistem.....	22
Gambar 5. 2 Pohon perancangan sistem.....	23
Gambar 5. 3 Blok perancangan perangkat keras.....	24
Gambar 5. 4 Perancangan rangkaian perangkat keras pada aplikasi fritzing.....	27
Gambar 5. 5 Diagram alir fungsi menerima data melalui Bluetooth.....	28
Gambar 5. 6 Diagram alir fungsi menghidupkan semua sistem	29
Gambar 5. 7 Rancangan fungsi mematikan semua sistem.....	29
Gambar 5. 8 Rancangan fungsi menjalankan PIR	30
Gambar 5. 9 Rancangan fungsi membuka salah satu pintu	30
Gambar 5. 10 Rancangan fungsi mengunci salah satu pintu.....	31
Gambar 5. 11 Rancangan fungsi mengunci pintu waktu tertentu.....	31
Gambar 5. 12 Rancangan fungsi mengunci semua pintu selama periode waktu tertentu.....	32
Gambar 5. 13 Perancangan aplikasi Android.....	33
Gambar 5. 14 Pohon implementasi sistem.....	34
Gambar 5. 15 Rangkaian sistem pada blackbox	34
Gambar 5. 16 Blackbox rangkaian sistem	35
Gambar 5. 17 Rancangan pintu sistem	35
Gambar 5. 18 Implementasi fungsi konektivitas Bluetooth	40
Gambar 5. 19 Implementasi notifikasi konektivitas Bluetooth	40
Gambar 5. 20 Implementasi fungsi pengambilan data perintah suara	41
Gambar 5. 21 Implementasi fungsi pengiriman data perintah	41
Gambar 5. 22 Implementasi fungsi penerimaan feedback.....	42
Gambar 5. 23 Implementasi fungsi penerimaan feedback kedua	42
Gambar 5. 24 Implementasi fungsi penerimaan feedback ketiga	43
Gambar 5. 25 Implementasi fungsi penerimaan feedback keempat	44
Gambar 6. 1 Pohon pengujian dan analisis.....	45
Gambar 6. 2 Menu perintah suara.....	46



Gambar 6.3 hasil output aplikasi	46
Gambar 6.4 Menu Bluetooth	48
Gambar 6.5 Bluetooth <i>connected</i>	48
Gambar 6.6 Menu perintah suara	49
Gambar 6. 7 Hasil output pada aplikasi	49
Gambar 6. 8 Hasil output pada serial monitor	50
Gambar 6. 9 Hasil output aplikasi	50
Gambar 6. 10 Hasil output pada serial monitor	50
Gambar 6.11 Menu Bluetooth	52
Gambar 6.12 Bluetooth <i>Connected</i>	52
Gambar 6.13 Menu perintah suara	53
Gambar 6.14 Menghidupkan semua sistem	54
Gambar 6.15 Semua solenoid door lock mengunci	54
Gambar 6.16 <i>Feedback</i> menghidupkan semua sistem	55
Gambar 6.17 Menghidupkan semua sistem kedua	55
Gambar 6.18 <i>Feedback</i> menghidupkan semua sistem kedua	56
Gambar 6.19 Kesalahan perintah suara	56
Gambar 6.20 Kesalahan perintah suara kedua	57
Gambar 6.21 Mematikan semua sistem	58
Gambar 6.22 <i>Feedback</i> mematikan semua sistem	58
Gambar 6.23 Membuka pintu depan	59
Gambar 6.24 <i>Feedback</i> membuka pintu depan	59
Gambar 6.25 membuka pintu kamar	60
Gambar 6.26 <i>Feedback</i> membuka pintu kamar	60
Gambar 6.27 Mengunci pintu depan	61
Gambar 6.28 <i>Feedback</i> mengunci pintu depan	61
Gambar 6.29 Mengunci pintu kamar	62
Gambar 6.30 <i>Feedback</i> mengunci pintu kamar	62
Gambar 6.31 Menu Bluetooth	66
Gambar 6.32 Bluetooth Connected	66
Gambar 6.33 Menu perintah suara	67
Gambar 6.34 Menjalankan sensor PIR	68
Gambar 6.35 <i>Feedback</i> dari sensor PIR	68
Gambar 6.36 Menjalankan sensor PIR kedua	69



Gambar 6.37 *Feedback* kedua dari sensor PIR69

Gambar 6.38 Menu Bluetooth71

Gambar 6.39 Bluetooth *connected*71

Gambar 6.40 Menu perintah suara72

Gambar 6.41 Mengunci semua pintu 5 detik73

Gambar 6.42 *Feedback* mengunci pintu selama 5 detik73

Gambar 6.43 Kedua solenoid door lock terbuka74

Gambar 6.44 Mengunci pintu pada waktu tertentu.....74



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Smart Home adalah sebuah konsep rumah masa depan, yang salah satu konsepnya adalah pengendalian peralatan elektronik dengan menggunakan perintah suara (*Voice Command*), dimana pada saat itu manusia tidak lagi harus bergerak mendekati sebuah peralatan rumah tangga dan menekan tombol yang ada untuk menghidupkan atau mematikan sebuah alat tersebut, melainkan dapat dikendalikan melalui perintah suara dari penghuni rumah tersebut (Ali, et al., 2017). Konsep rumah ini bermaksud agar para penghuni rumah tersebut dapat memiliki pengalaman yang nyaman, menyenangkan, efisiensi atas semua pekerjaan rumah tangga. Dalam beberapa hal konsep rumah seperti ini sangat membantu bagi para orang tua dan orang cacat, sehingga diharapkan akan dapat memberikan kualitas peningkatan hidup bagi orang-orang yang dinyatakan mungkin memerlukan pengasuh atau perawatan secara khusus (Dani, et al., 2016).

Konsep *smart home* bertujuan agar masyarakat memiliki kenyamanan dan keamanan di dalam rumah. Salah satu fitur pada smart home adalah sistem keamanan rumah, salah satu contoh sistem keamanan rumah yang masih banyak digunakan adalah alarm rumah. Tetapi seiring dengan perkembangan teknologi saat ini sistem keamanan rumah semakin banyak jenisnya dan akan terus berjalan perkembangannya. Sistem keamanan rumah sudah menjadi salah satu fitur yang wajib dimiliki oleh setiap pemilik rumah yang ingin memiliki rumah yang aman dari pencurian dan gangguan keamanan yang lain yang tidak diinginkan. Kurangnya tingkat keamanan menjadikan sering terjadinya pencurian dan pembobolan pada rumah. Terkadang kita sudah yakin sudah mengunci rumah kita, namun pada kenyataannya kasus pembobolan rumah pada rumah saat ini dikarenakan dengan mudahnya para pencuri membuka pengunci pintu yang terpasang hanya dengan seutas kawat ataupun kunci tiruan lainnya (Shandy, et al., 2015).

Voice Command Recognition System adalah salah satu konsep pengendalian perangkat elektronik dengan menggunakan perintah suara. Dengan konsep tersebut maka semua peralatan elektronik dapat dikendalikan dengan perintah suara dari penggunaannya. Dengan menggunakan *voice command*, kita dapat memerintahkan sebuah *smartphone* untuk mengerjakan sesuatu melalui perintah suara, sehingga kita tidak perlu mengoperasikan tombol navigasi pada layer *smartphone*. *Speech recognition* sendiri merupakan teknologi komputer masa kini yang digunakan untuk mengidentifikasi suara yang diucapkan oleh seseorang tanpa mengetahui siapa yang berbicara. Implementasi *speech recognition* dapat digunakan untuk berbagai macam *platform*, misalnya perintah suara untuk menjalankan aplikasi komputer, perintah suara untuk mengoperasikan *hardware*, dan lain lain. Cara kerja yang dilakukan oleh sistem dengan mendeteksi tingkat penekanan suara, lalu dicocokkan dengan database

yang tersedia. Penggunaan speech recognition dapat digabungkan dengan berbagai macam mikrokontroler, seperti *Arduino*, *raspberry pi*, dan *AVR* (Pallas, et al., 2018).

Suara merupakan alat untuk berkomunikasi yang paling efektif dan efisien karena kemudahan dalam penggunaannya. Pemakaian sinyal ucapan untuk berkomunikasi mempunyai beberapa kelebihan, diantaranya informasi maupun perintah dapat diberikan oleh pemakai meskipun pemakai sedang sibuk melakukan aktivitas lain yang melibatkan tangan maupun mata. Dengan menggunakan *voice command* (perintah suara) akan membantu pengguna melakukan kontrol pada peralatan rumah tangga tanpa harus mendekati perangkat tersebut, karena pengguna hanya perlu mengucapkan sebuah perintah suara pada *smartphone* (Sischalhaci, et al., 2012).

Smartphone atau Android device dimanfaatkan untuk masukan suara dari user dan tempat pengolahan kalimat, sedangkan *Arduino* sebagai menerima dan mengolah data dari *Android device*. Hasil proses pada *Arduino* adalah sebuah system yang telah dijalankan, kemudian menghasilkan *feedback* berupa kondisi keamanan rumah pada saat ini. *Feedback* tersebut berupa suara yang diucapkan pada *smartphone* Android.

Parameter yang diuji berupa masukan suara dari individu yang berbeda beda, pengujian tingkat keberhasilan sistem dalam melakukan penguncian pintu menggunakan solenoid door lock, uji tingkat keberhasilan sistem dalam pengecekan keberadaan seseorang didalam rumah menggunakan sensor PIR (*Passive Infra Red*), uji tingkat keberhasilan sistem dalam melakukan pengecekan waktu saat ini menggunakan RTC(*Real Time Clock*) yang dilanjutkan dengan penguncian pintu,. Dengan adanya skripsi ini, penulis berusaha mengembangkan teknologi kontrol smarthome supaya lebih mudah penggunaannya dalam berbagai bidang khususnya melalui suara.

1.2 Rumusan Masalah

Berdasarkan latar belakang maka terdapat rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana akurasi dan efektivitas pengolahan teks terhadap masukan *voice command* atau perintah suara dari user?
2. Bagaimana akurasi dan efektivitas pengiriman data menggunakan Bluetooth?
3. Bagaimana tingkat keberhasilan solenoid door lock untuk sistem keamanan pada *smart home*?
4. Bagaimana tingkat keberhasilan sensor PIR untuk sistem keamanan pada *smart home*?
5. Bagaimana tingkat keberhasilan *Real Time Clock* untuk sistem keamanan pada *smart home*?

1.3 Tujuan

Sesuai dengan rumusan masalah yang telah disebutkan maka terdapat tujuan dari penelitian ini sebagai berikut:

1. Untuk mengetahui akurasi dan efektivitas pengolahan teks terhadap masukan *voice command* atau perintah suara dari user.
2. Untuk mengetahui akurasi dan efektivitas pengiriman data menggunakan Bluetooth.
3. Untuk mengetahui tingkat keberhasilan sistem keamanan pada *smart home* dalam penguncian pintu menggunakan solenoid door lock.
4. Untuk mengetahui tingkat keberhasilan sistem keamanan pada smart home dalam pengecekan keberadaan seseorang menggunakan sensor PIR.
5. Untuk mengetahui tingkat keberhasilan sistem keamanan pada smart home dalam pengecekan waktu saat ini menggunakan *Real Time Clock*.

1.4 Manfaat

Terdapat manfaat dalam penelitian ini antara lain:

1. Penulis dapat menerapkan suatu sistem kontrol *smart home* menggunakan *voice command*.
2. Bagi para peneliti, hasil penelitian dapat dikembangkan menjadi penelitian baru.
3. Mendapatkan pemahaman mengenai akurasi dan efektivitas pengolahan teks terhadap masukan *voice command* atau perintah suara
4. Mendapatkan pemahaman mengenai akurasi dan efektivitas pengiriman data menggunakan Bluetooth.
5. Mendapatkan pemahaman mengenai tingkat keberhasilan solenoid door lock dalam penguncian pintu.
6. Mendapatkan pemahaman mengenai tingkat keberhasilan pengecekan keberadaan seseorang menggunakan sensor PIR.
7. Mendapatkan pemahaman mengenai tingkat keberhasilan dalam pengecekan waktu menggunakan *Real Time Clock*.

1.5 Batasan Masalah

Penentuan batasan masalah dilakukan agar permasalahan yang sudah dirumuskan dapat lebih terfokus dan tidak meluas. Pada penelitian ini, batasan masalah tersebut antara lain:

1. Mikrokontroler yang digunakan adalah Arduino Uno.
2. Sensor yang digunakan adalah sensor PIR dengan jangkauan maksimum 7meter.
3. Aktuator yang digunakan adalah solenoid door lock dengan tipe LY-03 berjumlah 2 buah dan relay dengan 2 channel.
4. Menggunakan RTC atau *Real Time Clock* dengan tipe DS3231 sebagai referensi waktu saat ini, referensi waktu yang digunakan adalah jam, menit, detik.

5. Tools yang digunakan untuk membuat aplikasi android adalah MIT App inventor.
6. Smartphone berbasis Android yang digunakan untuk mengontrol smart home.
7. Implementasi sistem memanfaatkan komunikasi nirkabel, dalam implementasinya menggunakan modul Bluetooth HC-05.

1.6 Sistematika Pembahasan

Sistematika penulisan dalam skripsi ini meliputi sebagai berikut:

BAB 1 Pendahuluan

Bab ini menjelaskan tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

BAB 2 Landasan Kepustakaan

Bab ini membahas tinjauan pustaka dari penelitian-penelitian sebelumnya dan dasar teori yang mendukung dalam pembuatan sistem ini.

BAB 3 Metodologi

Bab ini menjelaskan tentang langkah-langkah dalam melakukan penelitian ini, antara lain metode penelitian, analisis kebutuhan sistem, perancangan dan implementasi sistem, pengujian dan analisis, dan penutup.

BAB 4 Analisis Kebutuhan

Pada bab ini berisi tentang kebutuhan terkait penelitian yaitu kebutuhan perangkat keras, kebutuhan perangkat lunak serta kebutuhan fungsionalitas dan kebutuhan non fungsionalitas.

BAB 5 Perancangan dan implementasi

Pada bab ini berisi tentang bagaimana perancangan dan implementasi perangkat keras dan lunak yang digunakan.

BAB 6 Pengujian dan analisis

Pada bab ini berisi pengujian dari implementasi yang dilakukan dan mengolah data yang di dapat untuk di analisa.

BAB 7 Kesimpulan dan saran

Bab ini berisi tentang penarikan kesimpulan penelitian dan saran dari pengguna untuk pengembangan sistem selanjutnya.

BAB 2 LANDASAN KEPUSTAKAN

Pada penelitian ini, penulis mengkaji dari jurnal tentang voice command yang berjudul “Perancangan Aplikasi *Voice Command Recognition* berbasis Android dan Arduino Uno” ditulis oleh Akhmad Wahyu Dani, dkk. Kemudian referensi selanjutnya “Desain interaksi aplikasi pengendali *smart home* menggunakan *voice command* berbasis Android” ditulis oleh Ananto Setyo Wicaksono, dkk. Sedangkan untuk beberapa referensi dasar teori yang diperlukan berdasarkan pembuatan sistem dan rumusan masalah yaitu konsep dasar dari sensor sensor PIR, solenoid door lock, serta modul Bluetooth.

2.1 Tinjauan Pustaka

Pada sub bab ini akan menjelaskan tentang perbandingan antara penelitian yang dilakukan dengan penelitian yang pernah dilakukan sebelumnya. Pada penelitian pertama telah melakukan penerapan *voice command recognition* pada beberapa perangkat elektronik yang diantaranya adalah kipas dc, solenoid door lock, dan lampu. Pada penelitian ini perangkat elektronik tersebut dikontrol menggunakan aplikasi *AMR_Voice* pada smartphone android serta dengan menggunakan media komunikasi Bluetooth (Dani, et al., 2016).

Pada rujukan kedua membahas tentang prototipe pengendali rumah menggunakan smartphone Android dengan konektivitas Bluetooth, pada penelitian tersebut menggunakan mikrokontroler Arduino Mega2560 yang terhubung dengan Bluetooth HC-05, driver motor, solenoid door lock, buzzer dan 6 LED, dimana semua perangkat tersebut dikontrol menggunakan aplikasi pada smartphone Android. (Wicaksono, et al., 2016).

Berdasarkan pada penelitian Akhmad Wahyu Dani, Andi Ardiansyah, dan Dodi Hermawan (2016) sistem dibuat lebih difokuskan pada seberapa efektif media komunikasi Bluetooth apabila digunakan sebagai media komunikasi Arduino dengan smartphone Android, sehingga pengujian yang dilakukan beberapa kali dengan perbedaan jarak antara sistem dengan smartphone Android serta dengan adanya perbedaan penghalang ataupun tidak ada penghalang pada komunikasi Bluetooth. Sedangkan pada penelitian Ananto Setyo Wicaksono, Fitri Utaminingrum, dan Hurriyatul Fitriyah (2016) merancang sebuah prototipe untuk pengendali rumah menggunakan smartphone Android, sistem dibuat lebih difokuskan pada aplikasi Android sehingga pada aplikasi tersebut terdapat fitur-fitur yang akan mempermudah pengguna dalam menggunakannya. Fitur-fitur tersebut diantaranya adalah penyediaan kata kunci pada setiap menu, kemudian fitur bantuan untuk mencoba menyalakan lampu. Pada tiap menu aplikasi terdiri dari beberapa kata kunci yang berbeda, misalkan pada menu utama terdiri dari kata kunci lampu, pintu, penyiram, dan elektronik, kemudian setelah masuk pada menu lampu terdiri dari kata kunci depan, tengah, kamar depan, kamar belakang, dan kamar mandi, setelah itu apabila sudah masuk pada menu lampu depan terdiri dari kata kunci hidup, mati, menu, dan lampu, serta terdapat indikator kondisi lampu depan pada saat ini sehingga pada aplikasi dapat diketahui bahwa kondisi lampu depan saat ini mati atau hidup.

Berdasarkan rujukan dari kedua penelitian tersebut, maka penelitian ini akan merancang sistem keamanan pada smarthome dengan menggunakan konektivitas Bluetooth yang dimana sistem terdiri dari 2 solenoid door lock, sensor PIR, RTC, dan Bluetooth HC-05. Sistem dilengkapi dengan beberapa fitur diantaranya fitur untuk menghidupkan semua sistem, mematikan semua sistem, membuka atau mengunci pada salah satu pintu, melakukan scan keberadaan seseorang didalam rumah, mengunci semua pintu selama periode waktu tertentu, serta mengunci semua pintu pada waktu tertentu.

2.2 Dasar Teori

Pada bagian ini akan dibahas teori-teori yang berkaitan dengan penelitian yang akan dilakukan diantaranya Android, Bluetooth, *Voice command*, Arduino IDE, dan MIT App Inventor 2.

2.2.1 Android

Android adalah sistem operasi yang saat ini banyak dipakai pada smartphone. Android adalah sebuah sistem operasi mobile yang *open-source* dan dikembangkan oleh Google. Android banyak dipakai sebagai sistem operasi smartphone, karena bersifat *open source* yang membuat sistem operasi tersebut dapat dipasang pada smartphone manapun. Berbeda dengan sistem operasi lain yang bersifat *closed source* misalnya, IOS yang hanya dapat digunakan pada smartphone keluaran Apple saja. Untuk mengembangkan aplikasi android, kita dapat memakai berbagai program misalnya, java dan app inventor (Wicaksono, et al., 2016).



Gambar 2. 1 Android

Sumber : (Raphael, 2017)

Gambar 2.1 merupakan tampilan dasar android pada smartphone, dimana pada android terdapat beberapa aplikasi yang dapat digunakan untuk keperluan sehari - hari.

2.2.2 Bluetooth

Bluetooth adalah suatu peralatan media komunikasi yang dapat digunakan untuk menghubungkan sebuah perangkat komunikasi dengan perangkat komunikasi lainnya. Bluetooth umumnya digunakan pada smartphone, komputer atau personal komputer dan tablet. Fungsi Bluetooth yaitu sebagai media komunikasi untuk mempermudah berbagi data sebagai pengganti kabel.

Bluetooth merupakan teknologi komunikasi *wireless* atau tanpa kabel yang beroperasi dalam pita frekuensi 2.4GHz (antara 2.02 GHz sampai 2.480 GHz) dengan menggunakan sebuah *frequency hopping transceiver* yang mampu menyediakan layanan komunikasi data dan juga suara *secara real-time* antara host-host bluetooth dengan jarak jangkauan yang terbatas (Wicaksono, et al., 2016).

2.2.3 Voice command

Voice command adalah fitur perintah suara untuk menjalankan suatu aplikasi yang terdapat pada smartphone atau perangkat lainnya untuk mempermudah pengguna dalam menggunakan fitur tersebut. Sebagian besar *smartphone* punya fasilitas perintah program dengan suara (*voice command*), tetapi ternyata fitur ini belum banyak dimanfaatkan secara maksimal. *Voice command* ini adalah fitur dimana kita bias memerintahkan sebuah *smartphone* untuk mengerjakan sesuatu melalui perintah suara, sehingga kita tidak perlu mengoperasikan tombol navigasi pada layar (Wicaksono, et al., 2016).

Google speech recognition adalah salah satu fitur yang ada di dalam smartphone android, dengan fitur tersebut, kita dapat mengenali kata yang diucapkan oleh pengguna. Berbeda dengan *Sound recognition*, *Speech recognition* lebih berorientasi kepada apa yang diucapkan daripada siapa yang mengucapkan. Dimana terdapat perbedaan yang signifikan apabila dibandingkan dengan *Sound recognition*, dilihat dari hasilnya *sound recognition* lebih berorientasi kepada manusianya, sedangkan *speech recognition* lebih berorientasi kepada hasil dari ucapan manusianya. Maka kedua fitur tersebut dapat digunakan sesuai dengan kebutuhan sebuah sistem.

2.2.4 Arduino IDE

Merupakan aplikasi pengompile kode Arduino berbasis *open-source* yang dapat dipasang pada Windows, Mac OS X, maupun Linux. Pada Arduino, bahasa pemrograman yang dipakai merupakan C/ C++ yang dapat dipanggil dari kode programmer kemudian *di-compile* dan *di-upload*. Pemograman C pada Arduino ini merupakan pemograman tingkat rendah, yang artinya pemograman dilakukan langsung ke *hardware*. Sehingga, pemograman Arduino langsung ditunjukkan kepada pin yang digunakan pada *source code*. Semua perintah pemograman C dan C ++ yang dibangun harus sesuai dengan standar avr-g++ agar dapat mendukung dan bekerja dengan baik di Arduino (Pallas, et al., 2018).

2.2.5 MIT Appinventor 2

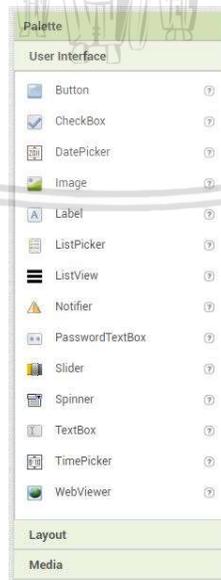
App inventor adalah tool yang digunakan untuk pembuatan aplikasi android berbasis web. Salah satu fitur dari app inventor adalah dalam membuat program androidnya seperti menyusun sebuah puzzle. Kita hanya perlu untuk memasang blok – blok program tersebut agar saling terhubung. Sehingga akan mempermudah dalam proses pembuatan aplikasi pada Android. Kemudian setelah program sudah jadi kita dapat *build* program tersebut menjadi file.apk agar dapat digunakan di *smartphone* android.



Gambar 2. 2 MIT App Inventor

Sumber : (MITAppInventor, 2018)

Gambar 2.2 diatas menunjukkan tampilan awal web app inventor yang digunakan untuk membuat sebuah aplikasi android. disana terdapat beberapa menu diantaranya, *palette*, *viewer*, *component* dan *properties*. Untuk membuat program menggunakan app inventor setidaknya butuh dua tahap. Tahap yang pertama adalah menentukan komponen yang digunakan dalam sebuah *screen*. Tahap kedua adalah menentukan perilaku atau apa yang dilakukan komponen dalam screen yang dibuat dengan menggunakan blok-blok program. Tahap menentukan komponen, yaitu kita memilih komponen yang ada dalam tampilan menu *palette*.



Gambar 2. 3 Palette MIT App Inventor

Sumber : (MITAppInventor, 2018)

Pada Gambar 2.3 diatas merupakan menu *palette* tempat komponen tersedia. Pada Gambar 2.3 tersebut menunjukkan beberapa komponen yang dapat digunakan diantaranya pada menu *user interface* terdapat *button, textbox, DatePicker, Image, Label, ListPicker, ListView, Notifier, PassowrdTextBox, Slider, Spinner, TextBox, TimePicker, dan WebView*. Dengan menggunakan komponen-komponen tersebut maka akan mempermudah dalam proses pembuatan aplikasi pada smartphone Android. kemudian pada bagian menu *layout*, menu *media* dan menu *drawing and animation* juga terdapat komponen yang dapat digunakan untuk pembuatan antarmuka pada aplikasi. Apabila hanya meletakkan komponen yang digunakan pada screen, maka aplikasi masih belum bisa berjalan karena komponen tersebut hanyalah komponen yang belum ada perintah atau perilakunya. Jadi meskipun tombol tersebut ditekan, tidak ada perubahan apapun. Karena itu tahap kedua, yaitu menentukan perilaku masing - masing komponen dengan blok - blok program. Untuk mengatur perilakunya, buka bagian *block* pada app inventor.



Gambar 2. 4 Hello World Blocks

Sumber : (MITAppInventor, 2018)

Pada Gambar 2.4 diatas terdapat beberapa menu perintah yang dapat digunakan pada menu block yaitu *Control, Logic, Math, Text, List, Colors, Variabels*, dan *Procedures*. Dimana pada menu Control terdapat beberapa blok fungsi diantaranya fungsi *if else*, fungsi *for*, fungsi *while*, fungsi *do* dan beberapa fungsi lainnya. Kemudian pada menu Logic terdapat blok program diantaranya *true, false, not, and, dan or*. Pada menu Math terdapat beberapa blok program diantaranya *+, -, x, /, ^* dan beberapa blok program lainnya. Pada menu Text terdapat blok program untuk *upcase, downcase, input teks*, dan beberapa blok program lainnya. Begitupula pada menu *Text, List, Colors, Variabels*, dan *Procedures* terdapat blok-blok program yang digunakan agar suatu komponen dapat melakukan tindakan seperti yang diinginkan.

Pada blok program tersebut menjelaskan bahwa apabila tombol1 ditekan maka label1 berganti tulisan menjadi Hello World. Kelebihan yang dimiliki oleh app inventor adalah pemrograman yang dilakukan tidak menggunakan *script* seperti pemrograman pada java, melainkan menggunakan blok-blok program. Kekurangannya adalah kita harus mengikuti aturan yang ada pada app inventor, sehingga kita sulit untuk memodifikasi atau membuat program yang kompleks.

2.2.6 Smart home

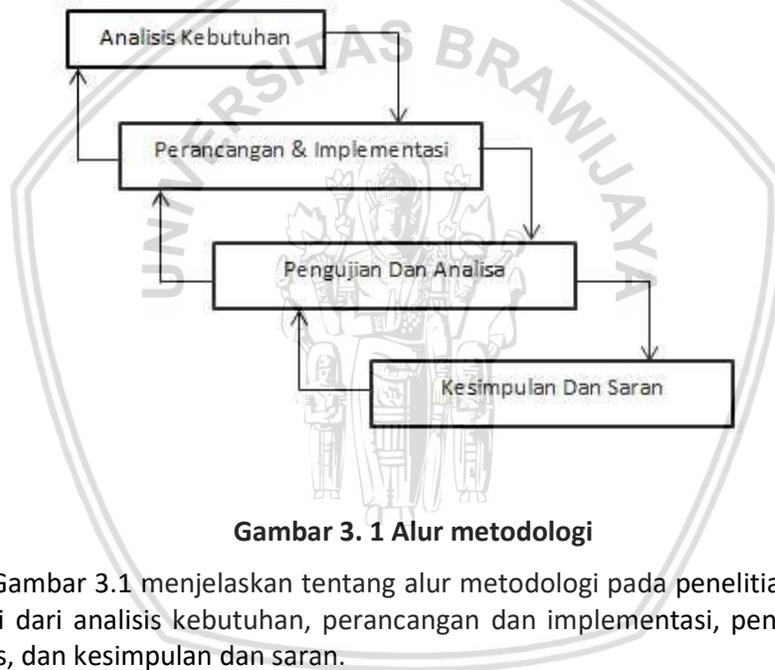
Smart home atau rumah pintar adalah aplikasi gabungan antara teknologi dan pelayanan yang dikhususkan pada lingkungan rumah dengan fungsi tertentu yang bertujuan meningkatkan keamanan, efisiensi dan kenyamanan penghuninya. Sistem rumah pintar (*smart home*) biasanya terdiri dari perangkat monitoring , perangkat kontrol dan otomatis ada beberapa perangkat yang dapat di akses menggunakan komputer. Rumah Pintar (*Smart Home*) merupakan sebuah aplikasi yang dirancang dengan berbantuan komputer yang akan memberikan kenyamanan, keamanan dan penghematan energi yang berlangsung secara otomatis sesuai dengan kendali pengguna dan terprogram melalui komputer pada gedung atau tempat tinggal kita. Teknologi yang dirancang untuk rumah pintar ini bertujuan untuk memudahkan pemilik rumah dalam memantau kondisi peralatan elektronik yang terhubung dari *gadget* yang dimiliki. (Masykur & Prasetyowati, 2016).



BAB 3 METODOLOGI

3.1 Metode Penelitian

Pada bab ini menjelaskan tentang metodologi penelitian yang digunakan dalam melakukan penelitian maupun dalam penulisan skripsi. Adapun langkah-langkah yang dilakukan secara berurutan yaitu analisis kebutuhan, perancangan sistem, implementasi, pengujian dan analisis, serta pengambilan kesimpulan dan saran sebagai referensi untuk pengembangan penelitian selanjutnya. Jika pada tahap analisis ditemui kesalahan maka dilakukan pengulangan terhadap tahapan perancangan dan implementasi sistem. Berikut adalah gambaran diagram alir metodologi yang digunakan dalam penelitian dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Alur metodologi

Gambar 3.1 menjelaskan tentang alur metodologi pada penelitian ini, yang diawali dari analisis kebutuhan, perancangan dan implementasi, pengujian dan analisis, dan kesimpulan dan saran.

3.2 Analisis Kebutuhan

Pada bagian ini dibahas mengenai dasar-dasar teori yang digunakan untuk mendukung penelitian tentang perancangan pengendali *smart home* menggunakan suara melalui *smartphone* dan Arduino dengan menggunakan *voice command*. Terdapat beberapa sub bab yang dibahas, diantaranya gambaran umum sistem yang menjelaskan tentang bagaimana sistem ini dibuat, analisis kebutuhan sistem yang membahas kebutuhan-kebutuhan yang digunakan untuk penelitian dan dibagi menjadi kebutuhan perangkat keras dan kebutuhan perangkat lunak. Lalu kebutuhan fungsional dan kebututhan non fungsional. Penjelasan lebih rinci dibahas pada bab 4 analisis kebutuhan.

3.3 Perancangan dan Implementasi

Perancangan sistem merupakan tahapan bagaimana membangun sebuah sistem dari penelitian yang dilakukan. Tahapan ini dilakukan setelah melakukan tahapan analisis kebutuhan. Dengan adanya tahap ini maka Sistem dapat digambarkan secara sistematis dan terstruktur. Perancangan sistem dibahas dari segi perancangan sistem pada Android, perancangan sistem pada Arduino. Penjelasan lebih rinci dibahas pada bab 5 perancangan dan implementasi.

Implementasi sistem dilaksanakan sesuai dengan perancangan yang telah ditentukan sebelumnya, mulai dari analisis kebutuhan hingga perancangan sistem. Langkah pertama yang dilakukan dalam implementasi sistem yaitu Pengguna memberikan masukan berupa perintah suara melalui program yang telah dibuat menggunakan *Android platform*. Langkah selanjutnya pada Arduino merangkai *hardware* untuk komunikasi dengan memanfaatkan HC-05. Setelah komponen tersebut saling terhubung, dirangkai perangkat lunak pada Arduino melakukan pengolahan kalimat untuk melakukan tindakan sesuai dengan perintah yang telah dimasukkan oleh *user* melalui program pada *Android platform*. Penjelasan lebih rinci dibahas pada bab 5 perancangan dan implementasi.

3.4 Pengujian dan Analisis

Tahap pengujian dan analisis sistem untuk menguji apakah sistem yang dibuat sudah sesuai seperti yang diharapkan oleh peneliti. Pengujian dan analisis menggunakan beberapa parameter. Berikut ini adalah parameter yang digunakan dalam pengujian sistem :

1. Pengujian fungsional dari sistem. Hal ini ditunjukkan untuk mengetahui apakah sistem sudah berjalan sesuai dengan keinginan peneliti. Tolak ukur dalam hal ini dapat dilihat melalui serial monitor pada Arduino, serta aktif atau tidaknya sensor dan aktuator.
2. Pengujian sensor PIR sebagai pendeteksi manusia dengan cara scanning keadaan didalam rumah untuk mengetahui apakah di dalam rumah masih ada manusia atau tidak, yang kemudian memberikan *feedback* keadaan di dalam rumah berupa suara pada android.
3. Pengujian solenoid door lock dengan cara mengunci atau membuka pintu kamar dan pintu rumah bagian depan secara bersamaan, mengunci atau membuka pada salah satu pintu, mengunci selama periode waktu tertentu, dan mengunci pada waktu tertentu. kemudian *feedback* berupa suara pada android.

3.5 Kesimpulan dan Saran

Penarikan kesimpulan merupakan tahap yang dilakukan setelah melakukan seluruh kegiatan pengujian sistem yang telah dirancang sebelumnya. Tujuan penarikan kesimpulan agar penelitian ini dapat digunakan sebagai tolak ukur dan dapat dilanjutkan menjadi penelitian yang lebih baik serta tidak berhenti sampai kegiatan penulis selesai. Pengambilan saran bertujuan agar penelitian ini dapat dikembangkan menjadi penelitian yang lebih baik ke depannya.

BAB 4 ANALISIS KEBUTUHAN

Bab ini berisi tentang penjelasan gambaran umum sistem bekerja. Kemudian penjelasan tentang analisis kebutuhan sistem meliputi kebutuhan fungsional dan kebutuhan non fungsional yang terdiri dari kebutuhan perangkat lunak serta penjelasan mengenai Batasan desain sistem.

4.1 Gambaran Umum Sistem

Gambaran umum pada sistem yang dibuat, Arduino mampu bekerja sesuai dengan perintah suara yang dimasukkan oleh user melalui *Android device*. Adapun sistem yang digunakan berupa *Android device* yang berfungsi untuk masukan suara sekaligus tempat untuk *text processing*. Metode yang digunakan untuk *input* perintah dari *smartphone* Android ke Arduino adalah menggunakan perintah suara (*voice command*). Dimana pengguna akan menginput perintah pada *Android device* menggunakan aplikasi yang sudah ada dengan perintah suara. Kemudian aplikasi Android tersebut memproses data suara untuk diubah kedalam bentuk teks, yang kemudian teks tersebut dikirim ke Arduino melalui konektivitas Bluetooth HC-05. Setelah Arduino menerima input melalui Bluetooth yang berupa teks, maka Arduino menjalankan sensor PIR ataupun solenoid door lock berdasarkan perintah yang telah diterima. Setelah Arduino menjalankan sensor PIR atau solenoid door lock, maka Arduino mengirimkan *feedback* dari pemrosesan tersebut yang berupa teks ke aplikasi android menggunakan konektivitas Bluetooth. Kemudian aplikasi android menerima *feedback* dari arduino tersebut, *feedback* tersebut berupa teks yang kemudian teks tersebut diproses menjadi suara pada aplikasi android.

4.2 Kebutuhan Sistem

Kebutuhan sistem berisi penjelasan kebutuhan fungsional dan non-fungsional, kebutuhan komunikasi, kebutuhan perangkat keras dan kebutuhan perangkat lunak dianalisis sesuai dengan kebutuhan sistem sehingga mempermudah dalam mendesain dan mengimplementasikan sistem.

4.2.1 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras merupakan kebutuhan perangkat yang digunakan untuk mewujudkan fisik sistem alat ini. Perangkat keras pada penelitian ini berupa :

1. Arduino Uno

Arduino Uno adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari Wiring platform, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*-nya memiliki prosesor Atmel AVR dan *software*-nya memiliki Bahasa pemrograman sendiri. Saat ini Arduino sangat populer di seluruh dunia. Banyak pemula yang belajar mengenal robotika dan elektronika lewat Arduino karena mudah dipelajari. Bahasa yang dipakai dalam Arduino bukan *assembler* yang relatif sulit, tetapi Bahasa C yang

disederhanakan dengan bantuan pustaka – pustaka (*libraries*) Arduino. Dengan menggunakan *libraries* yang ada maka akan mempermudah untuk membuat kode program pada Arduino Uno, karena pada saat membuat kode program hanya dengan menggunakan suatu fungsi untuk memanggil *libraries* tersebut maka suatu komponen akan dapat dijalankan dengan normal.

Tabel 4. 1 Spesifikasi mikrokontroler

Kategori	Spesifikasi
Mikrokontroller	Arduino Uno
Tegangan Pengoprasian	5V
Tegangan Input yang disarankan	7 – 12V
Batas Tegangan Input	6 – 20V
Jumlah pin I/O digital	14pin
Jumlah pin input Analog	6pin
Memori Flash	32KB
SRAM	2KB
EPROM	1KB
Clock Speed	16MHz

Berdasarkan spesifikasi pada Tabel 4.1 pada penelitian ini menggunakan Arduino uno karena memiliki pin-pin yang sesuai dengan kebutuhan dimana pin-pin tersebut yang digunakan untuk melakukan komunikasi antar perangkat keras yang terhubung. Pada sistem ini peneliti membutuhkan jumlah pin yang digunakan sebanyak 7 pin diantaranya adalah 5 pin I/O digital dan 2 pin I/O analog yang dapat dipenuhi kebutuhannya dengan menggunakan mikrokontroler Arduino Uno. Dibandingkan menggunakan mikrokontroler Arduino mega yang memiliki pin-pin yang lebih banyak, maka akan terjadi pemborosan pin pada mikrokontroler yang digunakan, karena pada sistem ini pin yang digunakan tidak terlalu banyak sehingga lebih baik apabila menggunakan Arduino Uno yang memiliki jumlah pin I/O digital sebanyak 14 pin serta pin input analog sebanyak 6 pin, dimana pin-pin tersebut telah mencukupi kebutuhan sistem.

2. Sensor PIR

PIR (*Passive Infrared Receiver*) merupakan sebuah sensor berbasis infrared. Akan tetapi, tidak seperti sensor infrared kebanyakan yang terdiri dari IR LED dan fototransistornya. PIR tidak memancarkan apapun seperti IR LED. Sesuai dengan Namanya "*Passive*", sensor ini hanya merespon energi dari inframerah pasif yang dimiliki oleh setiap benda yang terdeteksi olehnya. Benda yang bisa dideteksi oleh sensor ini biasanya adalah tubuh manusia.

Tabel 4. 2 Spesifikasi sensor PIR

Kategori	Spesifikasi
Tipe produk	HC-SR501 Body sensor module
Range tegangan pengoperasian	4.5V-20V
Level output	High 3.3V/LOW 0V
Trigger	L trigger satu kali/H pengulangan trigger
Waktu delay	0.3detik-5menit

Sudut deteksi	<110°
Jarak maksimum deteksi	7meter

Berdasarkan spesifikasi pada Tabel 4.2 pada penelitian ini menggunakan sensor HC-SR501 Body sensor module atau biasa disebut dengan sensor PIR, dimana sensor tersebut memiliki range tegangan untuk pengoperasian antara 4.5V-20V, kemudian pada trigger sensor dapat diatur apakah sensor akan bekerja satu kali saja atau secara terus menerus setelah adanya pendeteksian keberadaan seseorang, serta sudut deteksi untuk sensor PIR adalah kurang dari 110° dan jarak maksimum deteksi pada sensor adalah 7 meter. Pada sistem ini sensor PIR sangat sesuai untuk mendeteksi keberadaan seseorang yang ada didalam rumah.

3. Solenoid door lock

Solenoid door lock adalah alat yang dapat mengubah energi listrik menjadi energi mekanik atau linear. Solenoid yang paling umum biasanya menggunakan medan magnet yang dibuat dari arus listrik ditrigger sebagai aksi kerja dorong atau tarik pada sebuah objek sebagai *starter, valve, switch, latches*. Jenis paling sederhana dari selenoida mengandalkan dua aspek utama untuk fungsi solenoid tersebut, yaitu sebuah kawat atau berenamel terisolasi yang dibentuk menjadi gulungan ketat dan batang yang terbuat dari besi atau baja. Batang besi atau baja merupakan feromagnetik, sebuah properti yang dapat berfungsi sebagai elektromagnetik ketika diberikan arus listrik. Ketika arus listrik kawat yang dibentuk menjadi koil menerima arus. Medan magnet yang dihasilkan menarik besi atau batang baja dengan kuat. Batang yang dihubungkan pada sebuah pegas bergerak ke kumparan dan akan tetap pada posisinya sampai arus dihentikan, kondisi pegas saat ini menjadi tertekan. Ketika arus dimatikan, pegas kembali ke posisi semula dan menarik batang besi atau baja pada posisi awalnya.

Tabel 4. 3 Spesifikasi Solenoid Door Lock

Kategori	Spesifikasi
Tipe produk	Solenoid Door Lock LY-03
Tegangan pengoperasian	12V DC
Waktu penguncian	1detik

Berdasarkan spesifikasi pada Tabel 4.3 penelitian ini menggunakan solenoid door lock ly-03 yang digunakan sebagai pengunci pintu pada sistem ini. Solenoid yang digunakan memiliki tegangan pengoperasian 12V DC yang berarti apabila tegangan yang masuk kurang dari 12V DC maka solenoid door lock tidak dapat bekerja. Kemudian waktu yang dibutuhkan oleh solenoid door lock untuk mengunci atau membuka adalah 1 detik, sehingga setelah mendapatkan perintah dari sistem maka solenoid door lock membutuhkan waktu 1 detik untuk bekerja mengunci atau membuka.

4. Relay

Relay adalah sebuah saklar yang dikendalikan oleh arus. Relay memiliki sebuah kumparan tegangan rendah yang dililitkan pada sebuah inti. Terdapat sebuah armatur besi yang akan tertarik menuju inti apabila dialiri arus mengalir melewati kumparan. Relay dibutuhkan dalam rangkaian elektronika sebagai eksekutor sekaligus *interface* antara beban dan system kendali elektronik yang berbeda sistem *power supply*-nya. Secara fisik antara saklar atau kontaktor dengan elektromagnet relay terpisah sehingga antara beban dan sistem kontrol terpisah.

Tabel 4. 4 Spesifikasi Relay

Kategori	Spesifikasi
Tipe produk	Relay 2channel JQC-3FF
Tegangan pengoperasian	5V, 10A/250VAC
Jumlah pin I/O	4pin

Berdasarkan spesifikasi pada Tabel 4.4 penelitian ini menggunakan relay 2 channel JQC-3FF yang digunakan sebagai saklar untuk mengatur tegangan yang masuk dari adaptor. Relay yang digunakan adalah 2 channel karena solenoid door lock yang digunakan pada sistem ini adalah 2 buah, sehingga lebih baik menggunakan relay 2 channel untuk menyesuaikan dengan kebutuhan sistem. Untuk lebih jelas mengenai relay 2 channel dapat dilihat pada gambar dibawah. Pada relay 2 channel, kita dapat menggunakan salah satu channel saja ataupun menggunakan keduanya. Relay memiliki beberapa pin yang digunakan sebagai input dari Arduino yaitu pin GND, IN1, IN2, dan VCC, kemudian beberapa pin yang digunakan sebagai input dari adaptor serta output dari sistem, pin-pin tersebut diantaranya NO1, COM1, NC1, NO2, COM2, NC2.

5. Modul Bluetooth HC-05

Modul Bluetooth HC-05 adalah sebuah modul Bluetooth SPP (*Serial Port Protocol*) yang banyak digunakan untuk komunikasi *nirkabel*, prinsip kerjanya yaitu mengkonversi port serial ke Bluetooth. HC-05 memiliki 2 mode konfigurasi, yaitu AT mode dan Communication mode. AT mode berfungsi untuk melakukan pengaturan konfigurasi dari HC-05, sedangkan Communication mode berfungsi untuk melakukan komunikasi dengan Bluetooth pada perangkat lain. Jarak sinyal dari HC-05 adalah 30 meter, dengan kondisi tanpa penghalang.

Mode komunikasi yang digunakan berupa mode slave, mode ini memungkinkan Android device untuk terhubung dengan modul ini. Tegangan yang digunakan adalah 5V agar performa dalam penerimaan data menjadi maksimal.

Tabel 4. 5 Spesifikasi Modul Bluetooth

Kategori	Spesifikasi
Tipe produk	Bluetooth Module HC-05
Tegangan pengoperasian	4V-6V
Jumlah pin I/O	6pin
Jangkauan	30m
Mode operasi	Master/Slave

Berdasarkan spesifikasi pada Tabel 4.5 penelitian ini menggunakan Modul Bluetooth HC-05 yang digunakan sebagai modul komunikasi antara Arduino dengan smartphone android. dengan menggunakan modul Bluetooth maka akan mempermudah pengguna untuk melakukan kontrol terhadap sistem karena pengguna dapat menggunakan media komunikasi Bluetooth yang terdapat pada *smartphone* android untuk melakukan kontrol pada sistem. Modul Bluetooth HC-05 memiliki 6 pin yang terdiri dari pin Enable/Key, VCC, GND, TX, RX, dan State. Pada pin Enable/Key digunakan untuk mengatur modul akan digunakan sebagai Data Mode atau AT command mode, tetapi secara default akan berada pada Data Mode, kemudian pada pin VCC dihubungkan ke sumber tegangan +5V, pada pin GND dihubungkan ke sumber GND, pada pin TX dihubungkan pada serial data transmit atau pin TX pada Arduino, pada pin RX dihubungkan pada receive serial data atau pin RX pada Arduino, pada pin state dapat dihubungkan pada board Arduino sebagai feedback untuk mengetahui bahwa Bluetooth telah bekerja.

6. Modul RTC DS3231

Modul RTC DS3231 adalah salah satu jenis modul yang dimana berfungsi sebagai RTC (*Real Time Clock*) yang dapat menghitung waktu mulai dari detik, menit, jam, tanggal, hari, dan tahun dengan akurat dan menyimpan data waktu tersebut secara real time. Karena modul tersebut bekerja secara *real time*, maka proses hitung waktu dilakukan *output* dan data tersebut langsung disimpan atau dapat dikirim ke perangkat lain melalui sistem antarmuka.

Tabel 4. 6 Spesifikasi RTC

Kategori	Spesifikasi
Tipe produk	RTC DS3231
Tegangan pengoperasian	2.3V-5.5V
Jumlah pin I/O	6pin

Berdasarkan spesifikasi pada Tabel 4.6 penelitian ini menggunakan Modul RTC DS3231 yang digunakan untuk mengetahui waktu pada saat ini. Tegangan untuk operasi pada RTC adalah antara 2.3V-5.5V sehingga modul ini dapat dihubungkan pada 3.3V atau 5V pada pin Arduino. Modul RTC DS3231 memiliki 6 pin yang terdiri dari pin GND, VCC, SDA, SCL, SQW, 32K, dimana GND dihubungkan dengan sumber – atau GND, VCC dihubungkan dengan sumber + atau VCC/5V, SDA (Serial Data) dihubungkan dengan pin analog, SCL (*Serial*

Clock) dihubungkan dengan pin analog, sedangkan SQW (*Square Wave output*) dan 32K (*32K oscillator output*) tidak digunakan.

7. Android device

Smartphone atau bisa disebut *Android device* merupakan kebutuhan hardware yang berfungsi sebagai pengolahan serta masukan suara dari user. *Android device* yang digunakan harus memiliki Bluetooth device, tidak ada syarat minimal spesifikasi *Android device* yang digunakan. Namun dalam penelitian ini spesifikasi *Android device* dapat dilihat pada tabel 7 dibawah ini.

Tabel 4. 7 Spesifikasi Android device

Kategori	Spesifikasi
Merek	Xiaomi Redmi 5 Plus
Sistem Operasi	Android 7.1.2 Nougat
Memori	RAM : 4GB , Internal : 64GB
Chipset	Chipset Snapdragon 625 + CPU Octa-core 2.0GHz Cortex-A53 + GPU Adreno 506

Pada Tabel 4.7 *Android device* yang digunakan berjenis Xiaomi Redmi 5 Plus, dengan sistem operasi yang digunakan adalah Android 7.1.2 Nougat, serta kapasitas RAM 4 GB dan kapasitas Internal storage 64 GB. Chipset yang tertanam pada *smartphone* Xiaomi Redmi 5 Plus adalah Qualcomm Snapdragon 625, kemudian CPU pada Xiaomi Redmi 5 Plus adalah Octa-core 2.0 GHz Cortex-A53, kemudian GPU pada Xiaomi Redmi 5 Plus adalah Adreno 506, dengan menggunakan *Android device* berspesifikasi seperti pada tabel 4.7 tersebut dapat menjalankan Aplikasi voice command dengan normal tanpa adanya kendala.

Hardware yang dibutuhkan pada android device di antaranya :

- Mic untuk memasukan suara dari user.
- Bluetooth *device* digunakan untuk komunikasi dengan Arduino.

8. Komputer windows

Komputer merupakan sebuah perangkat yang digunakan untuk memprogram sistem ke mikrokontroler Arduino Uno. Spesifikasi komputer yang digunakan untuk penelitian dapat dilihat pada tabel 4.8 dibawah ini.

Tabel 4. 8 Spesifikasi komputer windows

Kategori	Spesifikasi
Merek	ASUS K46CB
Sistem Operasi	Windows 10 Pro 64 Bit
Memori	RAM : 4GB , Harddisk : 256GB
Port USB	USB 2.0 atau USB 3.0
Prosesor	Intel Core i7-3537U

Berdasarkan Tabel 4.8 spesifikasi komputer yang digunakan menggunakan Sistem operasi windows 10-64 Bit dengan ram 4GB, Harddisk 256GB dan prosesor Intel Core i7-3537U. Usb Port yang digunakan bisa menggunakan USB 2.0 maupun

repository.ub.ac.id

USB 3.0. Port USB digunakan untuk menghubungkan mikrokontroler Arduino uno dengan komputer agar dapat memasukkan kode program ke mikrokontroler.

4.2.2 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak merupakan kebutuhan perangkat yang digunakan untuk membangun program pada sistem alat ini. Perangkat lunak pada penelitian ini berupa :

1. **Arduino IDE versi 1.8.4**

Arduino IDE versi 1.8.4 merupakan perangkat lunak yang berfungsi sebagai editor untuk menuliskan kode program, *compile* serta *upload* program ke Arduino uno pada sistem. Bahasa yang digunakan adalah Bahasa khusus Arduino IDE yang mirip dengan Bahasa C. juga dilengkapi dengan berbagai macam fungsi dan juga library yang mendukung dalam perancangan sistem diantaranya library DS3231 RTC Library, Software Serial Bluetooth Library. Arduino IDE dapat dioperasikan pada sistem operasi Windows 7, Windows 8, Windows 10, dan juga Linux.

2. **MIT App Inventor 2**

App Inventor 2 yang merupakan sebuah program aplikasi yang berjalan pada web atau biasa disebut aplikasi web. Aplikasi ini dibuat oleh pengembang MIT, sehingga sering disebut MIT App Inventor. Aplikasi ini berfungsi sebagai alat untuk membangun sebuah aplikasi yang berjalan pada perangkat android. App Inventor 2 dapat diakses pada alamat web: "ai2.appinventor.mit.edu" melalui peramban internet dan cocok digunakan untuk pengembang aplikasi pemula karena tampilannya yang sederhana dan mudah dimengerti.

Pada sistem ini menggunakan MIT App Inventor 2 sebagai *tool* untuk pembuatan aplikasi pada Android karena kemudahan dalam penggunaan *tool* tersebut dalam pembuatan aplikasi, serta fitur-fitur yang ada pada *tool* tersebut sudah memenuhi untuk pembuatan aplikasi. Fitur-fitur tersebut diantaranya adalah menu pembuatan tampilan antarmuka aplikasi yang terdiri dari pembuatan *Button*, *Label*, dan *TextBox*, serta fitur *blocks* yang digunakan untuk memberikan blok-blok perintah pada setiap menu yang sudah dibuat pada antarmuka aplikasi.

3. **Windows 10 Pro**

Windows 10 pro merupakan perangkat lunak yang berfungsi sebagai sistem operasi pada perangkat PC atau Laptop yang dipakai dalam penelitian untuk melakukan perancangan sistem yaitu untuk menjalankan Arduino IDE. Juga terdapat beberapa sistem operasi yang *didukung* diantaranya windows 7, Windows 8, Windows 10 dan juga Linux. Pada penelitian ini peneliti menggunakan Windows 10 sebagai sistem operasi pada perangkat laptop yang digunakan karena dapat menjalankan aplikasi Arduino IDE tanpa adanya kendala.

4.3 Kebutuhan Fungsional

Kebutuhan yang harus dipenuhi pada penelitian ini antara lain :

1. Aplikasi pada smartphone Android dapat menerima input berupa *voice command* atau perintah suara yang kemudian perintah suara tersebut diubah ke dalam bentuk teks.
2. Aplikasi pada smartphone Android dapat mengirimkan data melalui Bluetooth pada smartphone ke Bluetooth pada sistem, serta aplikasi dapat menerima data melalui Bluetooth.
3. Sistem dapat menjalankan sensor PIR untuk melakukan pengecekan kondisi di dalam rumah, apabila sensor PIR mendeteksi adanya pergerakan manusia maka sistem akan memberikan *feedback* ke aplikasi android bahwa di dalam rumah ada seseorang. Kemudian sistem akan menjalankan fungsi untuk mengunci semua pintu yang ada di rumah tersebut, serta memberikan *feedback* ke aplikasi android bahwa semua pintu sudah dikunci.
4. Sistem dapat menjalankan fungsi untuk membuka semua kunci pada pintu yang ada di rumah, diantaranya membuka pintu depan rumah dan membuka pintu kamar, kemudian sistem akan memberikan *feedback* ke aplikasi android bahwa semua pintu sudah dibuka.
5. Sistem dapat menjalankan sensor PIR untuk melakukan pengecekan kondisi di dalam rumah, apabila ada pergerakan manusia maka sistem akan memberikan *feedback* ke aplikasi android bahwa didalam rumah ada seseorang, atau sistem dapat menjalankan fungsi untuk mengunci salah satu pintu yang ada di rumah serta memberikan *feedback* ke aplikasi android bahwa pintu sudah dikunci atau dibuka.
6. Sistem dapat menjalankan fungsi untuk mengunci semua pintu selama beberapa waktu sesuai dengan perintah yang didapatkan, kemudian sistem akan memberikan *feedback* ke aplikasi android bahwa pintu telah dikunci selama beberapa waktu. Setelah mencapai batas waktu yang telah ditentukan sistem akan menjalankan solenoid door lock untuk membuka semua pintu.
7. Sistem dapat menjalankan fungsi untuk mengunci semua pintu yang ada di rumah apabila waktu yang ditentukan sudah tercapai, hal ini dilakukan untuk mengatasi adanya kelalaian pengguna dalam hal mengunci pintu rumah, sehingga sistem akan otomatis mengunci semua pintu apabila sudah mencapai waktu tertentu.

4.4 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional meliputi beberapa kebutuhan yang harus dipenuhi agar sistem berjalan dengan baik. Kebutuhan non-fungsional meliputi kebutuhan perangkat keras dan perangkat lunak yang digunakan untuk membangun sistem.

4.4.1 Karakteristik yang Berbeda tiap Individu

Sistem ini dibuat untuk *user* secara umum dengan harapan akan mempermudah penggunaan *quadcopter* dalam berbagai bidang khususnya

kontrol dan bidang pengolahan suara. Dalam sistem ini *user* yang berbeda mampu mengontrol *quadcopter* tanpa membutuhkan keahlian khusus apapun.

4.4.2 Lingkungan Pengoperasian Sistem

Pada dasarnya lingkungan pengoperasian sistem ini dapat dilakukan pada lingkungan manapun baik di dalam ruangan maupun luar ruangan, namun untuk pengujian digunakan ruangan tertutup agar mempermudah dalam proses pengujian sistem.

4.4.3 Asumsi Ketergantungan tiap Komponen

Tiap komponen yang digunakan memiliki ketergantungan satu sama lain agar memiliki hasil yang sesuai dengan harapan.

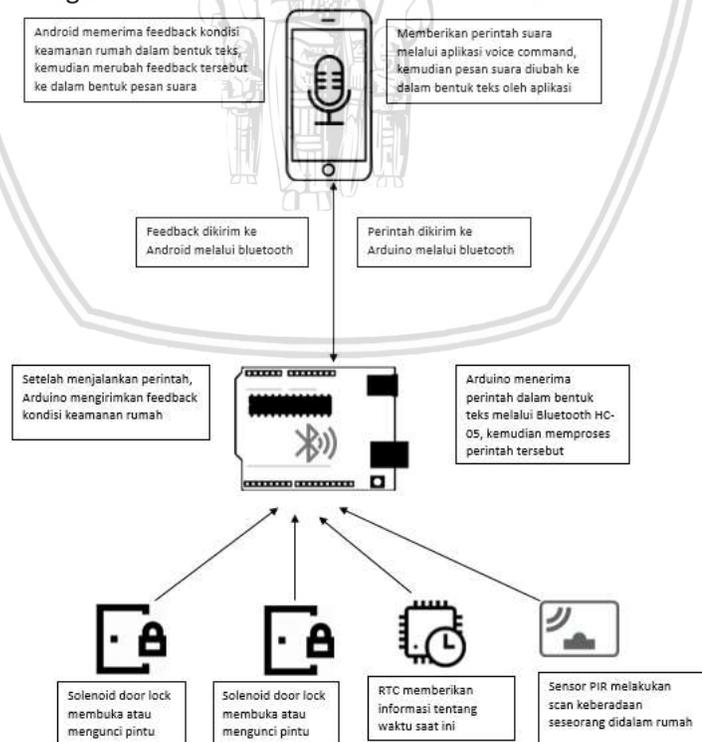
- a. Sistem akan aktif berfungsi apabila mikrokontroler telah terhubung ke sumber daya.
- b. Solenoid door lock pada sistem akan aktif berfungsi apabila telah terhubung ke sumber daya 12V DC dan Ground.
- c. Aplikasi Android membutuhkan koneksi Bluetooth yang dihubungkan dengan Arduino sebagai inialisasi bahwa program dimulai. Di dalamnya terdapat button dengan icon microphone untuk memberikan masukan suara.
- d. Menggunakan module Bluetooth HC-05 sebagai media komunikasi antara aplikasi android dengan mikrokontroler.
- e. Letak posisi sensor PIR harus berada pada atas pintu dan menghadap ke bawah, dikarenakan sensor PIR melakukan scanning orang di bawahnya.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Pada bab ini menjelaskan tentang mekanisme perancangan sistem dari penelitian, serta implementasi sistem yang meliputi implementasi perangkat keras, perangkat lunak, dan implementasi pendukung sistem lainnya.

5.1 Gambaran Umum Sistem

Pada penelitian ini sistem yang dibuat terdiri dari beberapa komponen, yaitu aplikasi *voice command* yang sudah terinstal pada smartphone android, Arduino Uno sebagai mikrokontroler yang digunakan pada sistem ini, kemudian Modul Bluetooth HC-05 komponen yang digunakan sebagai media komunikasi secara *wireless* dalam proses pertukaran data antara mikrokontroler Arduino Uno dengan Aplikasi pada *smartphone* Android, sensor PIR sebagai pendeteksi keberadaan seseorang di dalam rumah, RTC untuk memberikan informasi waktu tentang detik, menit, jam, tanggal, hari, dan tahun pada saat ini, dimana informasi tersebut akan digunakan sebagai referensi waktu pada sistem, kemudian 2 solenoid door lock sebagai pengunci pintu yang dibagi menjadi dua bagian penempatan, solenoid door lock yang pertama akan ditempatkan pada bagian depan rumah kemudian solenoid door lock yang kedua akan ditempatkan di pintu kamar. Kedua solenoid door lock tersebut dihubungkan dengan Arduino Uno sebagai mikrokontroler, dimana mikrokontroler akan mekontrol apakah solenoid tersebut akan membuka ataupun mengunci. Untuk gambaran sistem secara umum adalah sebagai berikut :

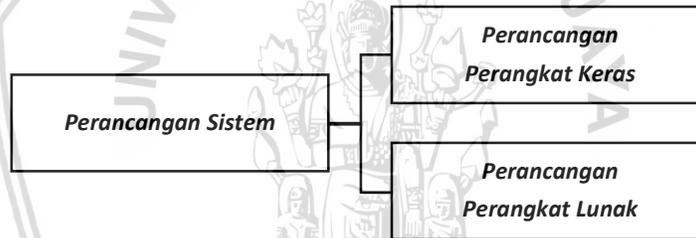


Gambar 5. 1 Gambaran umum sistem

Dari Gambar 5.1 diatas dapat dilihat bahwa pada awalnya pengguna memasukkan perintah dalam bentuk suara menggunakan aplikasi *voice command*, kemudian perintah suara tersebut diubah ke dalam bentuk teks sebelum dikirim ke Arduino. Setelah perintah suara diubah ke dalam bentuk teks, perintah tersebut dikirim ke Arduino uno menggunakan konektivitas Bluetooth menggunakan Bluetooth. Kemudian Arduino menyesuaikan perintah tersebut dengan daftar perintah yang ada pada kode program Arduino, apabila perintah tersebut telah sesuai dengan salah satu perintah maka Arduino akan menjalankan sensor atau aktuator sesuai dengan perintah tersebut. Setelah perintah selesai dijalankan Arduino akan mengirimkan *feedback* berupa teks tentang kondisi keamanan rumah ke *smartphone* android menggunakan konektivitas Bluetooth. Kemudian *smartphone* Android akan merubah *feedback* teks dari Arduino tersebut ke dalam bentuk pesan suara.

5.2 Perancangan Sistem

Proses perancangan sistem dibagi menjadi beberapa tahapan yang dimulai dari gambaran sistem secara umum. Kemudian dilanjutkan dengan perancangan perangkat keras dan perangkat lunak serta peralatan pendukung *sistem* ini. Perancangan sistem dijelaskan pada pohon perancangan sistem sebagai berikut :



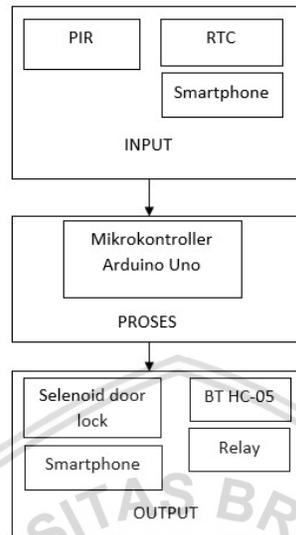
Gambar 5. 2 Pohon perancangan sistem

Gambar 5.2 merupakan pohon perancangan sistem yang terbagi menjadi dua bagian, yaitu : perancangan perangkat keras dan perancangan perangkat lunak.

5.2.1 Perancangan Perangkat Keras

Perancangan perangkat keras merupakan perancangan sistem yang ditunjukkan untuk merangkai fisik sistem alat, pada Gambar 5.3 dibawah merupakan blok perancangan yang terdiri dari bagian *input*, bagian proses dan bagian output pada sistem. Bagian input terdapat sensor PIR dan RTC, sensor PIR akan melakukan pengecekan keberadaan seseorang yang ada didalam rumah, kemudian RTC memberikan informasi tentang waktu yang terdiri dari waktu detik, menit, jam, tanggal, hari, dan tahun pada saat ini yang nantinya akan digunakan oleh Arduino untuk pemrosesan selanjutnya. Bagian proses terdapat mikrokontroler Arduino Uno sebagai komponen perangkat keras yang berfungsi mengolah data. Dan bagian output berupa modul Bluetooth yang digunakan untuk mengirimkan data informasi ke Android, solenoid door lock sebagai *output* untuk membuka atau mengunci pada pintu rumah, relay sebagai saklar atau *switch* arus

listrik dari adaptor, kemudian adaptor sebagai sumber arus listrik untuk menjalankan solenoid door lock.



Gambar 5. 3 Blok perancangan perangkat keras

Pada Gambar 5.3 merupakan blok perancangan perangkat keras, dimana pemasangan komponen dengan mikrokontoler pada sistem ini merupakan hal penting dalam merangkai perangkat keras sistem alat. Di sini dilakukan pendataan setiap pin dari masing masing komponen yang digunakan untuk menghubungkan antar perangkat keras. Penggunaan setiap pin pada mikrokontroler juga harus sesuai dan tepat terhubung dengan pin yang berada pada perangkat keras lainnya agar sistem dapat berjalan dengan normal. Pada konfigurasi yang pertama akan dijelaskan tentang konfigurasi pin antara RTC dengan Arduino Uno. Untuk lebih jelas tentang hubungan di setiap pin komponen-komponen yang digunakan pada sistem ini dapat dilihat pada tabel 5.1 sebagai berikut.

Tabel 5. 1 Kabel hubungan RTC dengan Arduino

Pin RTC	Warna Kabel	Pin Arduino	Deskripsi
VCC	Merah	5V	5v DC power
GND	Hitam	GND	Ground
SDA	Hijau	A4	Serial clock
SCL	Ungu	A5	Serial data

Pada Tabel 5.1 merupakan konfigurasi Modul RTC DS3231 dengan Arduino uno, pada kabel merah merupakan sumber + atau VCC dari RTC yang dihubungkan dengan sumber + atau VCC/+5V pada breadboard yang terhubung dengan pin 5V pada Arduino , kemudian pada RTC terdapat kabel hitam yang merupakan GND yang dihubungkan pada sumber – atau Ground pada breadboard yang juga terhubung dengan Ground pada Arduino, kemudian pin SDA yang merupakan pin serial clock pada RTC dihubungkan dengan pin A4 yaitu pin Analog pada Arduino, serta pin SCL yang merupakan pin serial data pada RTC dihubungkan dengan pin A5 yaitu pin Analog pada Arduino, data yang diterima dari RTC merupakan data

analog maka pin yang digunakan pada Arduino sebagai pemrosesan data dari RTC juga harus menggunakan pin analog.

Pada modul RTC DS3231 harus dilakukan konfigurasi waktu terlebih dahulu dengan cara memasukkan kode program pada Arduino IDE ke Arduino. Kode program tersebut berisi tentang konfigurasi detik, menit, jam, tanggal, hari, dan tahun, yang kemudian waktu tersebut akan secara real time disimpan pada modul RTC DS3231 dengan menggunakan baterai sebagai daya input untuk RTC.

Tabel 5. 2 Kabe hubungan Bluetooth dengan Arduino

Pin BT HC-05	Warna Kabel	Pin Arduino	Deskripsi
VCC	Merah	5V	<i>5v DC power</i>
GND	Hitam	GND	<i>Ground</i>
TX	Orange	10	<i>Transmit</i>
RX	Ungu	11	<i>Receive</i>

Pada Tabel 5.2 merupakan konfigurasi Modul Bluetooth HC-05 dengan Arduino uno, pada kabel merah merupakan sumber + atau VCC dari Bluetooth HC-05 yang dihubungkan dengan sumber + atau VCC/+5V pada *breadboard* yang terhubung dengan pin 5V pada Arduino , kemudian pada Bluetooth HC-05 terdapat kabel hitam yang merupakan GND yang dihubungkan pada sumber – atau Ground pada *breadboard* yang juga terhubung dengan Ground pada Arduino, kemudian pin TX merupakan pin Transmit dari Bluetooth HC-05 yang dihubungkan dengan pin 10 pada Arduino, serta pin RX yang merupakan pin Receive dari Bluetooth HC-05 yang dihubungkan dengan pin 11 pada Arduino, TX berfungsi untuk mengirim data atau jalur pengiriman data antara modul Bluetooth HC-05 ke komponen lain, sedangkan RX berfungsi sebagai jalur penerimaan data antara komponen lain ke modul Bluetooth HC-05.

Tabel 5. 3 Kabel hubungan sensor PIR dengan Arduino

Pin PIR HC-SR501	Warna Kabel	Pin Arduino	Deskripsi
VCC	Merah	5V	<i>5v DC power</i>
GND	Hitam	GND	<i>Ground</i>
OUT	Hijau	2	<i>Output</i>

Pada Tabel 5.3 merupakan konfigurasi sensor PIR HC-SR501 dengan Arduino uno, pada kabel merah merupakan sumber + atau VCC dari sensor PIR HC-SR501 yang dihubungkan dengan sumber + atau VCC/+5V pada *breadboard* yang terhubung dengan pin 5V pada Arduino , kemudian pada PIR HC-SR501 terdapat kabel hitam yang merupakan GND yang dihubungkan pada sumber – atau Ground pada *breadboard* yang juga terhubung dengan Ground pada Arduino, kemudian pin OUT merupakan pin Output dari sensor PIR HC-SR501 yang dihubungkan dengan pin 2 pada Arduino, pin OUT atau Output berfungsi sebagai pin yang untuk pengiriman data hasil dari sensor PIR HC-SR501 yang akan diproses lebih lanjut pada Arduino.

Tabel 5. 4 Kabel hubungan Relay, Solenoid Door Lock dan Adaptor dengan Arduino

Pin Relay	Warna Kabel	Pin Arduino	Deskripsi
VCC	Merah	5V	5v DC Power
GND	Hitam	GND	Ground
IN1	Merah muda	13	Digital Input 1
IN2	Orange	12	Digital Input 2
Pin SDL LY-03 (1)	Warna Kabel	Pin Relay	Deskripsi
VCC	Merah	COM1	Common Pin 1
Pin SDL LY-03 (2)	Warna Kabel	Pin Relay	Deskripsi
VCC	Merah	COM2	Common Pin 2
Pin SDL LY-03 (1)	Warna Kabel	Pin Adaptor	Deskripsi
GND	Hitam	GND	Ground
Pin SDL LY-03 (2)	Warna Kabel	Pin Adaptor	Deskripsi
GND	Hitam	GND	Ground
Pin Relay	Warna Kabel	Pin Adaptor	Deskripsi
NO1	Biru	VCC	12v DC Power
NO2	Biru	VCC	12v DC Power

Pada Tabel 5.4 merupakan konfigurasi Relay 2 Channel, Solenoid Door Lock (1), Solenoid Door Lock (2), dan Adaptor dengan Arduino Uno, pada Relay terdapat 4 pin yang akan digunakan yaitu kabel merah merupakan sumber + atau VCC dari sensor PIR HC-SR501 yang dihubungkan dengan sumber + atau VCC pada *breadboard* yang terhubung dengan pin 5V Arduino, kemudian pada PIR HC-SR501 terdapat kabel hitam yang merupakan pin GND yang dihubungkan ke sumber - atau Ground pada *breadboard* yang terhubung dengan pin Ground pada Arduino, kemudian pin IN1 yang merupakan pin yang digunakan sebagai Digital Input 1 dihubungkan dengan pin 13 pada Arduino, pin IN2 yang merupakan pin yang digunakan sebagai Digital Input 2 dihubungkan dengan pin 12 pada Arduino, pin 13 dan pin 12 digunakan sebagai Digital Input untuk perintah dari Arduino yang apabila Arduino memberikan input HIGH/LOW maka Relay bekerja untuk membuka atau menutup kaki yang ada didalam di dalam relay.

Pada konfigurasi pin Solenoid Door Lock (1) dengan Relay 2 Channel terdapat kabel merah dari Solenoid Door Lock (1) yang merupakan sumber + atau VCC dihubungkan dengan COM1 atau Common Pin 1, kemudian pada konfigurasi pin Solenoid Door Lock (2) dengan Relay 2 Channel terdapat kabel merah dari Solenoid Door Lock (2) yang merupakan sumber + atau VCC dihubungkan dengan COM2 atau Common Pin 2, pin COM1 dan pin COM2 akan bekerja apabila pin IN1 dan pin IN2 mendapat input HIGH/LOW dari Arduino yang akan membuat kaki dari COM1 dan COM2 di dalam relay akan membuka atau menutup.

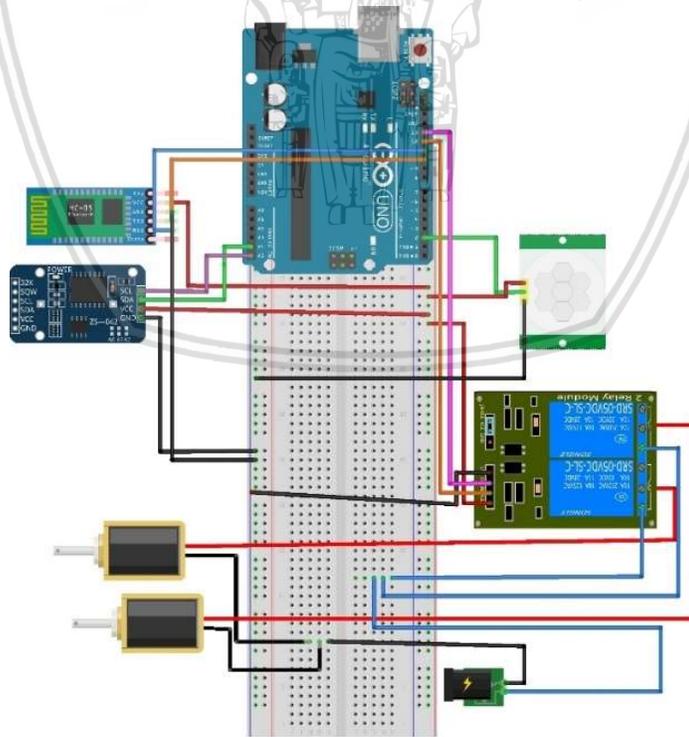
Pada konfigurasi pin Solenoid Door Lock (1) dengan Adaptor terdapat kabel hitam dari Solenoid Door Lock (1) yang merupakan sumber - atau Ground dihubungkan dengan sumber - atau Ground pada *breadboard* yang terhubung dengan pin Ground pada Adaptor, kemudian pada konfigurasi pin Solenoid Door Lock (2) terdapat kabel hitam dari Solenoid Door Lock (2) yang merupakan sumber - atau Ground dihubungkan dengan ke sumber - atau Ground pada *breadboard*



yang terhubung dengan pin Ground pada Adaptor, sehingga Ground dari kedua Solenoid Door Lock akan terhubung dengan Ground pada Adaptor.

Pada konfigurasi pin Relay 2 Channel dengan Adaptor terdapat kabel biru atau NO1 dari Relay 2 Channel yang merupakan sumber + atau VCC dihubungkan dengan sumber + atau VCC pada breadboard yang terhubung dengan 12V DC Adaptor, kemudian pada pin NO2 dari Relay 2 Channel terdapat kabel yang merupakan sumber + atau VCC dihubungkan dengan sumber + atau VCC pada *breadboard* yang terhubung dengan 12V DC Adaptor, sehingga pin NO1 dan pin NO2 pada Relay 2 Channel akan terhubung dengan 12V DC Adaptor.

Semua konfigurasi pin pada Relay 2 Channel, Solenoid Door Lock(2), Solenoid Door Lock(2) dan Adaptor harus sesuai dengan tabel 5.4 agar sistem dapat berjalan dengan normal. Pada proses konfigurasi pin komponen-komponen pada sistem, perlu diperhatikan juga penempatan dari setiap komponen tersebut agar tidak terjadi error pada sistem atau bahkan terjadi kerusakan pada komponen sistem. Untuk perancangan rangkaian perangkat keras pada sistem ini, dilakukan perancangan skematik perangkat keras dengan menggunakan aplikasi fritzing. Pada skematik tersebut terdiri dari *breadbord*, Arduino Uno, RTC DS3231, Bluetooth HC-05, sensor PIR HC-SR501, dua buah Solenoid Door Lock LY-03, Relay 2 Channel, serta Adaptor, dimana semua komponen tersebut dihubungkan dengan pin-pin yang sesuai dengan tabel 5.4 dengan menggunakan kabel jumper, untuk mempermudah dalam pembacaan alur kabel, maka kabel yang digunakan dibedakan antara kabel VCC, GND, dan kabel yang digunakan sebagai I/O data Digital atau I/O data Analog.



Gambar 5. 4 Perancangan rangkaian perangkat keras pada aplikasi fritzing

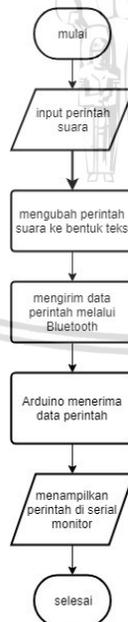
Pada Gambar 5.4 merupakan skematik rangkaian RTC DS3231, Bluetooth HC-05, sensor PIR HC-SR501, Relay 2 Channel, Solenoid Door Lock (1), Solenoid Door Lock (2), serta Adapter yang dibuat menggunakan aplikasi fritzing dengan penggunaan warna kabel sesuai dengan tabel hubungan antar komponen yang sudah dijelaskan pada tabel diatas.

5.2.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada sistem ini mencakup perangkat lunak untuk aplikasi android *voice command* dan upload *source code* program pada mikrokontroller Arduino Uno serta menjelaskan sub sistem yang ada pada sistem ini. Untuk bahasa yang digunakan yaitu adalah bahasa C Arduino. Pada bagian perancangan aplikasi *voice command* pada smartphone Android menggunakan tool MIT App Inventor, kemudian pada bagian pembuatan *source code* program menggunakan Arduino IDE yang sudah terinstal pada komputer windows. Kemudian pada bagian ini juga dijelaskan mengenai diagram alir pada tiap-tiap fungsi sistem, mulai dari fungsi untuk menghidupkan semua sistem, fungsi untuk mematikan semua sistem, fungsi untuk menjalankan PIR, fungsi untuk membuka salah satu pintu, fungsi untuk mengunci salah satu pintu, mengunci semua pintu selama periode waktu tertentu, dan fungsi untuk mengunci semua pintu pada waktu tertentu.

5.2.2.1 Perancangan source code

A. Rancangan fungsi menerima data melalui Bluetooth

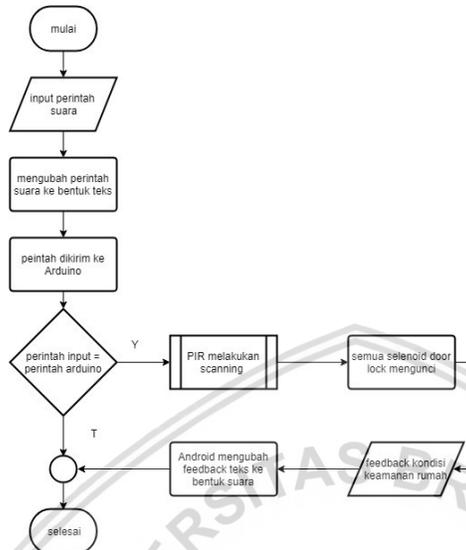


Gambar 5. 5 Diagram alir fungsi menerima data melalui Bluetooth

Pada Gambar 5.5 diatas dapat dilihat bahwa pada awal proses dilakukan input perintah suara yang kemudian perintah suara diubah ke dalam bentuk teks dan dilanjutkan dengan pengiriman data perintah suara melalui media komunikasi

Bluetooth. Selanjutnya Arduino akan menerima data perintah tersebut dan menampilkan data perintah ke serial monitor.

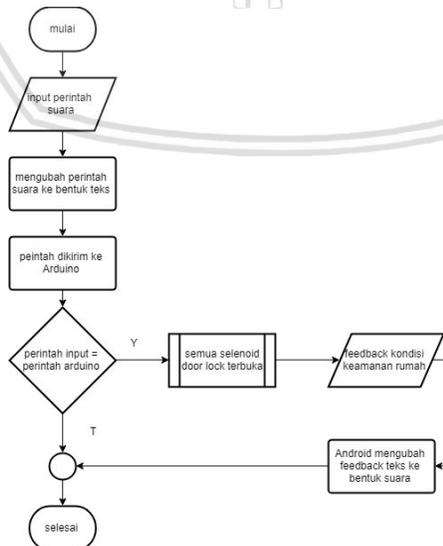
B. Rancangan fungsi untuk menghidupkan semua sistem



Gambar 5. 6 Diagram alir fungsi menghidupkan semua sistem

Pada Gambar 5.6 diatas dapat dilihat bahwa mikrokontroler menerima data melalui Bluetooth kemudian menyesuaikan data perintah suara tersebut dengan data perintah yang ada pada kode program mikrokontroler Arduino Uno, apabila data perintah sudah sesuai maka sensor PIR akan melakukan scanning untuk memeriksa keberadaan seseorang di dalam rumah, serta pada saat yang sama sistem akan menjalankan semua solenoid untuk mengunci semua pintu yang ada dirumah, kemudian *feedback* dari hasil fungsi tersebut akan dikirimkan oleh Arduino Uno ke aplikasi Android melalui konektivitas Bluetooth.

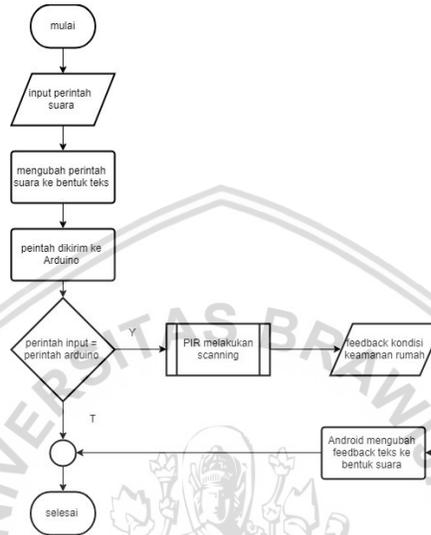
C. Rancangan fungsi untuk mematikan semua sistem



Gambar 5. 7 Rancangan fungsi mematikan semua sistem

Gambar 5.7 merupakan rancangan fungsi mematikan semua sistem, dimana pada awal proses data perintah diterima melalui media komunikasi Bluetooth, kemudian perintah disesuaikan dengan data perintah yang ada pada mikrokontroler, apabila data perintah sesuai maka mikrokontroler akan membuka semua solenoid door lock. Kemudian mikrokontroler akan mengirimkan *feedback* ke Android melalui Bluetooth.

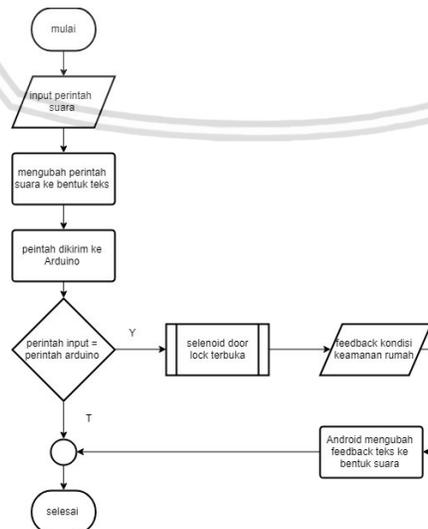
D. Rancangan fungsi untuk menjalankan PIR



Gambar 5. 8 Rancangan fungsi menjalankan PIR

Dari gambar 5.8 dapat dilihat bahwa data perintah diterima melalui Bluetooth kemudian diperiksa kesesuaian perintah dengan data perintah pada Arduino, ketika data perintah sudah sesuai maka PIR melakukan scanning keberadaan seseorang didalam rumah, kemudian mengirim *feedback* ke Android melalui bluetooth.

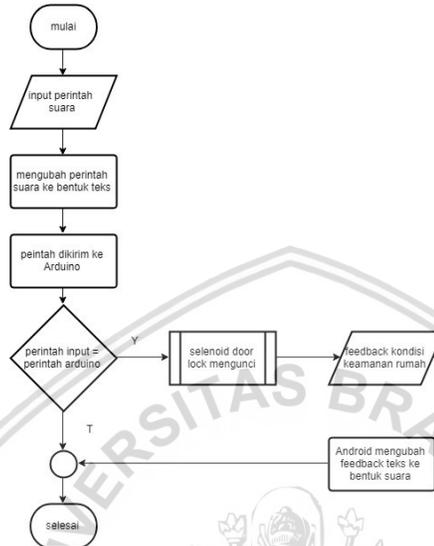
E. Rancangan fungsi untuk membuka salah satu pintu



Gambar 5. 9 Rancangan fungsi membuka salah satu pintu

Dari Gambar 5.9 dapat dilihat bahwa data perintah diterima melalui Bluetooth kemudian diperiksa kesesuaian perintah dengan data perintah pada Arduino, ketika data perintah sudah sesuai maka solenoid door lock akan terbuka, kemudian mengirim *feedback* ke Android melalui bluetooth.

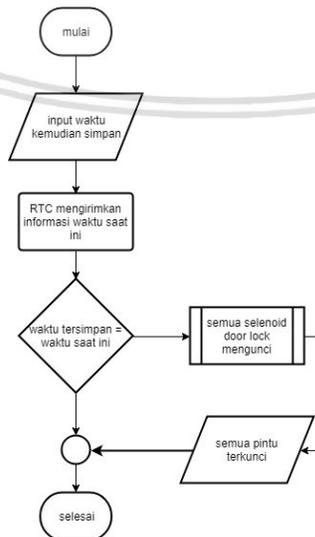
F. Rancangan fungsi untuk mengunci salah satu pintu



Gambar 5. 10 Rancangan fungsi mengunci salah satu pintu

Pada Gambar 5.9 diatas merupakan fungsi untuk mengunci salah satu pintu, dimana pada awal proses data perintah diterima melalui Bluetooth kemudian diperiksa kesesuaian perintah yang diterima oleh Arduino dengan data perintah yang sudah ada pada Arduino, ketika data perintah yang diterima sudah sesuai maka solenoid door lock akan mengunci, kemudian *feedback* dari hasil proses tersebut akan dikirimkan *feedback* ke Aplikasi *smartphone* Android melalui modul Bluetooth yang ada pada Arduino.

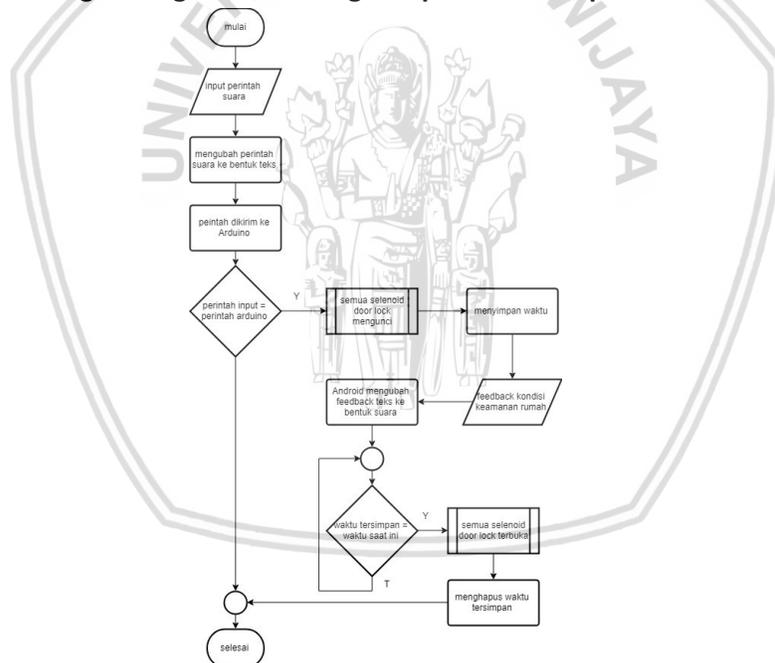
G. Rancangan fungsi untuk mengunci semua pintu pada waktu tertentu



Gambar 5. 11 Rancangan fungsi mengunci pintu waktu tertentu

Dari Gambar 5.11 dapat dilihat bahwa pada awalnya kita mengatur waktu yang meliputi detik, menit, dan jam pada kode program Arduino, kemudian secara berkala sistem akan melakukan pengecekan apakah waktu yang tersimpan sudah sama dengan waktu saat ini, apabila waktu tersimpan sudah sama dengan waktu saat ini maka sistem akan bekerja untuk menjalankan solenoid door lock untuk mengunci semua pintu di dalam rumah. Proses pengecekan sistem ini akan dilakukan terus menerus karena waktu yang tersimpan pada sistem tidak dilakukan penghapusan pada data waktu tersebut, sehingga meskipun waktu yang tersimpan sudah terpenuhi dan sistem sudah bekerja untuk menjalankan solenoid door lock untuk mengunci pintu, maka sistem akan terus menerus melakukan pengecekan waktu tersimpan. Pada sistem ini proses penguncian pintu hanya akan terjadi satu kali dalam sehari, karena pada sistem ini waktu yang diatur adalah jam malam hari dikarenakan untuk mengantisipasi apabila pengguna lupa untuk mengunci pintu rumah pada malam hari. Sehingga sistem ini akan membantu pengguna apabila pintu rumah belum terkunci pada malam hari maka secara otomatis solenoid door lock akan mengunci pintu apabila data waktu yang tersimpan sudah sesuai.

H. Rancangan fungsi untuk mengunci pintu selama periode waktu tertentu

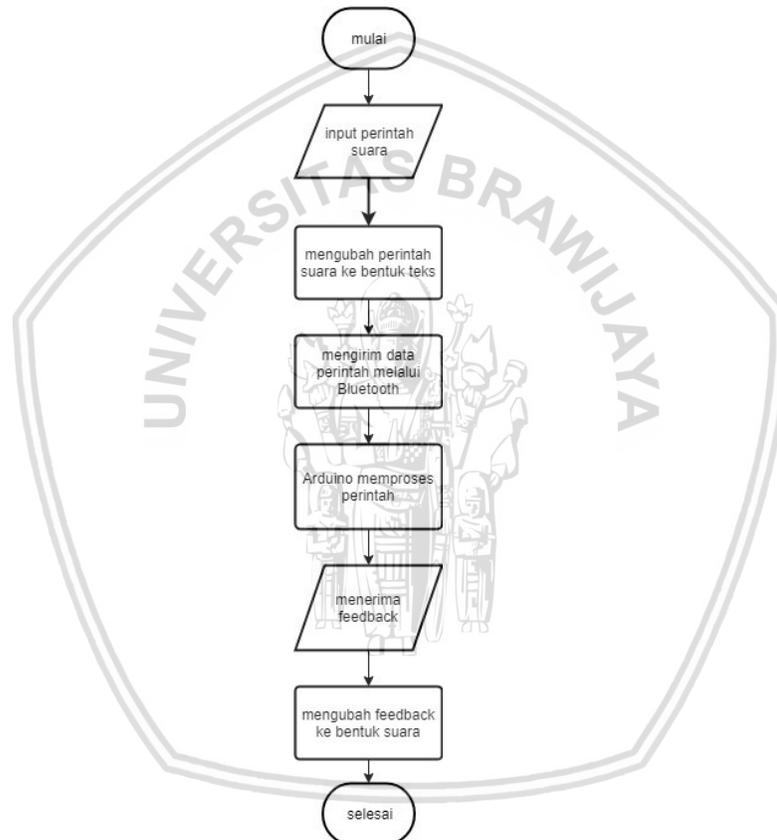


Gambar 5. 12 Rancangan fungsi mengunci semua pintu selama periode waktu tertentu

Dari Gambar 5.12 dapat dilihat bahwa pada awal proses adalah data perintah yang diterima melalui modul Bluetooth pada mikrokontroler Arduino akan disesuaikan dengan data perintah yang sudah ada pada Arduino, ketika data perintah sudah sesuai maka solenoid door lock akan bekerja dan mengunci pintu, kemudian mengirim *feedback* ke aplikasi smartphone Android melalui modul

bluetooth, kemudian sistem akan bekerja untuk menyimpan data waktu pada saat ini. Secara berkala sistem akan melakukan pengecekan waktu yang tersimpan tadi dan akan dibandingkan dengan waktu saat ini, apabila waktu tersimpan sudah sesuai dengan waktu saat ini maka sistem akan bekerja menjalankan solenoid door lock untuk membuka pintu. setelah waktu yang ditentukan telah sesuai dan solenoid door lock membuka pintu, maka data waktu yang telah tersimpan pada Arduino akan dihapus agar tidak terjadi pengulangan pada proses pengecekan waktu tersimpan sehingga tidak terjadi solenoid door lock yang membuka pintu dengan sendirinya.

5.2.2.2 Perancangan Aplikasi Android

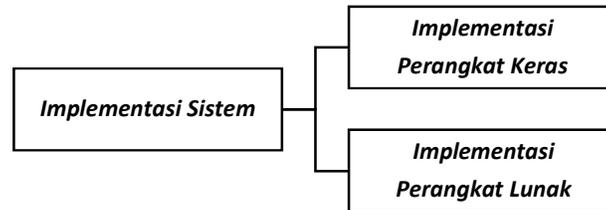


Gambar 5. 13 Perancangan aplikasi Android

Pada Gambar 5.13 dapat dilihat bahwa pada awal sistem pengguna harus memasukkan perintah suara melalui aplikasi voice command, kemudian aplikasi mengubah perintah suara tersebut ke dalam bentuk teks, setelah itu perintah tersebut dikirim ke Arduino melalui media komunikasi Bluetooth, setelah sistem selesai memproses perintah maka aplikasi akan menerima *feedback* berupa pesan teks, kemudian aplikasi akan mengubah *feedback* yang berupa pesan teks tersebut ke dalam bentuk pesan suara.

5.3 Implementasi Sistem

Implementasi sistem pada penelitian ini merupakan proses yang dilakukan setelah perancangan sistem. Implementasi sistem dijelaskan pada pohon implementasi sistem sebagai berikut:

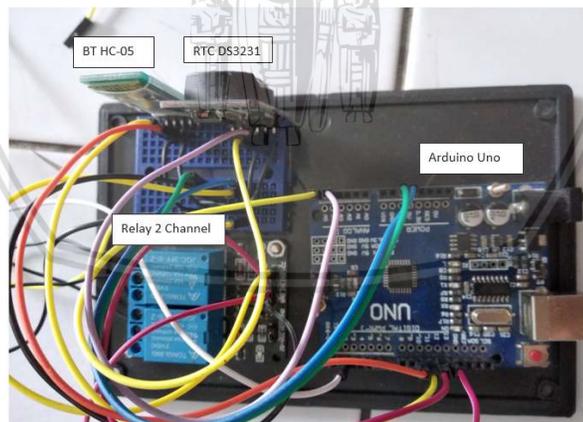


Gambar 5. 14 Pohon implementasi sistem

Gambar 5.14 merupakan implementasi sistem terdiri dari implementasi perangkat keras dan implementasi perangkat lunak.

5.3.1 Implementasi Perangkat Keras

Implementasi perangkat keras pada sistem ini meliputi perangkat keras sensor yang digunakan untuk mengambil nilai *input* yang didapatkan dari pembacaan sensor PIR, serta pembacaan waktu menggunakan RTC. Kemudian perangkat keras pemrosesan digunakan untuk memproses data menggunakan mikrokontroler Arduino. Dan perangkat keras *output* yang meliputi solenoid door lock dan relay, serta modul Bluetooth HC-05 sebagai media komunikasi *wireless*. Untuk lebih jelasnya mengenai implementasi perangkat keras telah dirancang dalam blackbox yang dapat dilihat pada gambar dibawah ini.

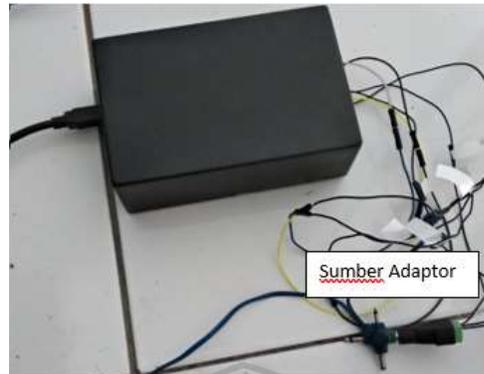


Gambar 5. 15 Rangkaian sistem pada blackbox

Pada Gambar 5.15 menampilkan rangkaian sistem pada blackbox dari rancangan perangkat keras yang telah dibangun, dimana terdapat sebuah mikrokontroler Arduino Uno yang terhubung dengan breadboard dan terdapat komponen perangkat yang terhubung yaitu RTC DS3231, Bluetooth HC-05, Relay 2 Channel, dimana semua komponen tersebut dirancang menjadi satu ke dalam blackbox agar mempermudah dalam penempatan komponen sistem, serta agar

repository.ub.ac.id

sistem terlihat lebih rapi. Untuk lebih jelasnya mengenai penempatan blackbox pada sistem dapat dilihat pada gambar dibawah ini.



Gambar 5. 16 Blackbox rangkaian sistem

Pada Gambar 5.16 diatas dapat dilihat blackbox yang didalamnya terdapat rangkaian perangkat sistem yang diantaranya RTC DS3231, Bluetooth HC-05, Relay 2 Channel, kemudian dibagian luar blackbox terdapat sumber dari Adaptor + atau VCC/12VDC serta sumber – atau Ground, dimana sumber 12V DC tersebut akan dihubungkan dengan relay sehingga relay dapat menggerakkan Solenoid Door Lock.



Gambar 5. 17 Rancangan pintu sistem

Pada Gambar 5.16 diatas menampilkan implmentasi alat secara langsung pada sebuah pintu dengan panjang dan lebar pintu sesuai dengan aslinya. Pada implementasi ini sensor PIR DS3231 ditempatkan pada bagian atas pintu, dimana PIR akan melakukan pengecekan keberadaan seseorang yang ada dibawah pintu tersebut ataupun berada didekat sensor tersebut karena sensor PIR memiliki jarak jangkauan 110° dan 7meter. Kemudian Solenoid Door Lock (1) dan Solenoid Door Lock (2) ditempatkan pada bagian samping kiri pintu, dimana pada Solenoid Door Lock (1) diimplementasikan sebagai pengunci pintu rumah pada bagian depan dan Solenoid Door Lock (2) diimplementasikan sebagai pengunci pintu kamar, kedua Solenoid Door Lock ditempatkan pada bagian samping sebagai implementasi pengunci pintu yang sebenarnya.

5.3.2 Implementasi Perangkat Lunak

5.3.2.1 Implementasi Program Arduino

A. Implementasi fungsi menerima data melalui Bluetooth

Algoritma 1 : fungsi menerima data melalui Bluetooth	
1	while (BT.available()) {
2	delay(10);
3	char c = BT.read();
4	perintah = c;
5	}

Pada baris pertama sampai ke 5 merupakan fungsi *while*, yaitu ketika Bluetooth telah terhubung maka akan dilakukan penerimaan data dengan sebuah perintah *BT.read*, yang kemudian data yang diterima melalui Bluetooth tersebut dimasukkan ke dalam variabel perintah.

B. Implementasi fungsi menghidupkan semua sistem

Algoritma 2 : fungsi menghidupkan semua sistem	
1	if (perintah == "hidupkan semua sistem") {
2	//fungsi scan PIR
3	data = digitalRead(inpir);
4	//kondisi ada orang
5	if ((data==HIGH) && (statpir==LOW) (data==HIGH) &&
6	(statpir==HIGH)){
7	statpir=HIGH;
8	BT.print(1);
9	}
10	//kondisi tidak ada orang
11	else if ((data==LOW) && (statpir==HIGH) (data==LOW) &&
12	(statpir==LOW)) {
13	statpir=LOW;
14	BT.print(2);
15	}
16	//fungsi kunci pintu depan
17	digitalWrite(13, HIGH);
18	//fungsi kunci pintu kamar
19	digitalWrite(12, HIGH);
20	}

Pada kode program diatas merupakan implementasi fungsi untuk menghidupkan semua sistem yang ditulis pada *source code* program Arduino, pada tabel diatas dapat dilihat bahwa apabila data perintah sama dengan "hidupkan semua sistem" maka program akan dijalankan, tetapi apabila data perintah tidak sesuai maka program tidak akan berjalan. Pada fungsi ini sistem

akan menjalankan sensor PIR untuk melakukan pengecekan keberadaan seseorang di dalam rumah, apabila terdeteksi adanya keberadaan seseorang, sistem akan memberikan mengirimkan sebuah data dengan perintah BT.print(1) dimana data yang dikirimkan melalui Bluetooth tersebut adalah sebuah data berisi angka 1, sedangkan apabila sensor PIR tidak mendeteksi adanya keberadaan seseorang di dalam rumah maka sistem akan mengirimkan sebuah data dengan perintah BT.print(2) dimana data yang dikirimkan melalui Bluetooth tersebut adalah sebuah data berisi angka 2.

Setelah sistem memberikan feedback atau dengan cara mengirim data BT.print(1) atau data BT.print(2) melalui perangkat Bluetooth HC-05, maka sistem akan memberikan nilai HIGH pada pin 13 dan pin 12 yang akan membuat relay untuk membuat Solenoid Door Lock (1) dan Solenoid Door Lock (2) mengunci.

C. Implementasi fungsi mematikan semua sistem

Algoritma 3 : fungsi mematikan semua sistem	
1	if (perintah == "matikan semua sistem") {
2	digitalWrite(13, LOW);
3	digitalWrite(12, LOW);
4	//BT Print
5	BT.print(3);
6	}

Pada kode program diatas dapat dilihat bahwa apabila data perintah sama dengan “matikan semua sistem”, maka pin 13 dan 12 akan bernilai LOW sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) dan Solenoid Door Lock (2) akan membuka. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan perintah BT.print(3) dimana data yang dikirimkan tersebut adalah sebuah data dengan nilai 3.

D. Implementasi fungsi menjalankan PIR

Algoritma 4 : fungsi menjalankan PIR	
1	if (perintah == "cek kondisi rumah") {
2	data = digitalRead(inpir);
3	//kondisi ada orang
4	if ((data==HIGH) && (statpir==LOW) (data==HIGH) &&
5	(statpir==HIGH)) {
6	statpir=HIGH;
7	//BT Print
8	BT.print(4);
9	}
10	//kondisi tidak ada orang
11	else if ((data==LOW) && (statpir==HIGH) (data==LOW) &&
12	(statpir==LOW)) {
13	statpir=LOW;
14	//BT Print
15	BT.print(5);}
16	}

Pada kode program diatas dapat diketahui bahwa data perintah harus sama dengan “cek kondisi rumah” agar program dapat dijalankan. Pada fungsi ini sistem akan menjalankan sensor PIR untuk melakukan pengecekan keberadaan seseorang di dalam rumah, apabila terdeteksi adanya keberadaan seseorang, sistem akan memberikan mengirimkan sebuah data dengan perintah BT.print(4) dimana data yang dikirimkan melalui Bluetooth tersebut adalah sebuah data berisi

angka 4, sedangkan apabila sensor PIR tidak mendeteksi adanya keberadaan seseorang di dalam rumah maka sistem akan mengirimkan sebuah data dengan perintah BT.print(5) dimana data yang dikirimkan melalui Bluetooth tersebut adalah sebuah data berisi angka 5.

E. Implementasi fungsi membuka salah satu pintu

Algoritma 5 : fungsi membuka salah satu pintu	
1	if (perintah == "Buka pintu depan") {
2	digitalWrite(13, LOW);
3	//BT Print
4	BT.print(6);
5	}
6	if (perintah == "Buka pintu kamar") {
7	digitalWrite(12, LOW);
8	//BT Print
9	BT.print(8);
10	}

Pada kode program diatas dapat dilihat bahwa apabila data perintah sama dengan "buka pintu depan", maka pin 13 diberi nilai LOW sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) akan membuka. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan perintah BT.print(6) dimana data yang dikirimkan tersebut adalah sebuah data dengan nilai 6. Kemudian pada fungsi kedua apabila data perintah sama dengan "buka pintu kamar", maka pin 12 akan diberi nilai LOW sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (2) akan membuka. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan perintah BT.print(8) dimana data yang dikirimkan tersebut merupakan sebuah data dengan nilai 8.

F. Implementasi fungsi mengunci salah satu pintu

Algoritma 6 : fungsi mengunci salah satu pintu	
1	if (perintah == "kunci pintu depan") {
2	digitalWrite(13, HIGH);
3	//BT Print
4	BT.print(7);
5	}
6	if (perintah == "kunci pintu kamar") {
7	digitalWrite(12, HIGH);
8	//BT Print
9	BT.print(9);
10	}

Pada kode program diatas dapat dilihat bahwa apabila data perintah sama dengan "kunci pintu depan", maka pin 13 diberi nilai HIGH sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) akan mengunci. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan perintah BT.print(7) dimana data yang dikirimkan tersebut adalah sebuah data dengan nilai 7. Kemudian pada fungsi kedua apabila data perintah sama dengan "kunci pintu kamar", maka pin 12 akan diberi nilai HIGH sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (2) akan mengunci. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan

perintah `BT.print(9)` dimana data yang dikirimkan tersebut merupakan sebuah data dengan nilai 9.

G. Implementasi fungsi mengunci semua pintu selama periode waktu tertentu

Pada implementasi fungsi untuk mengunci semua pintu selama periode waktu tertentu dibutuhkan 2 fungsi source code, yang pertama adalah fungsi untuk proses mengunci semua pintu selama periode waktu tertentu, yang kedua source code program untuk melakukan pengecekan apabila waktu yang tersimpan sudah sesuai dengan waktu saat ini. Untuk *source code* program pengunci pintu dapat dilihat dibawah ini.

Algoritma 7 : fungsi mengunci semua pintu selama periode waktu tertentu	
1	<code>if (perintah == "saya pergi 5 detik") {</code>
2	<code>digitalWrite(13, HIGH);</code>
3	<code>digitalWrite(12, HIGH);</code>
4	<code>dataDetiksimpan=dataDetik;</code>
5	<code>dataDetikmati=dataDetiksimpan+5;</code>
6	<code>//BT Print</code>
7	<code>BT.print(10);}</code>

Pada kode program diatas dapat dilihat bahwa apabila perintah sama dengan "saya pergi 5 detik", maka pin 13 dan 12 akan diberi nilai HIGH sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) dan Solenoid Door Lock (2) akan mengunci. Kemudian data waktu saat ini yang terdiri dari dataDetik akan disimpan pada variable dataDetiksimpan, kemudian pada variabel dataDetikmati dimasukkan nilai dari dataDetiksimpan ditambahkan dengan angka 5 atau 5 detik. Setelah itu sistem akan memberikan *feedback* kepada aplikasi Android tentang kondisi rumah saat ini melalui Bluetooth HC-05 dengan perintah `BT.print(10)` dimana data yang dikirimkan tersebut adalah sebuah data dengan nilai 10. Kemudian pada fungsi selanjutnya dilakukan pengecekan apakah waktu saat ini sudah sesuai dengan nilai waktu pada dataDetikmati. Untuk *source code* program pengecekan waktu yang tersimpan dapat dilihat pada tabel dibawah.

Algoritma 8 : fungsi pengecekan waktu	
1	<code>if (dataDetikmati==dataDetik){</code>
2	<code>digitalWrite(13, LOW);</code>
3	<code>digitalWrite(12, LOW);</code>
4	<code>dataDetikmati=""; }</code>

Pada kode program diatas dapat dilihat bahwa fungsi tersebut akan melakukan pengecekan data waktu yang telah tersimpan pada variable dataDetikmati sudah sesuai dengan data waktu detik saat ini pada variable dataDetik, apabila data waktu sudah sesuai maka pin 13 dan pin 12 akan diberi nilai HIGH sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) dan Solenoid Door Lock (2) membuka. Setelah itu sistem akan menghapus data waktu yang telah tersimpan pada variabel dataDetikmati dengan perintah `dataDetikmati=""`, sehingga pada variabel dataDetikmati sudah tidak ada data waktu. Hal tersebut dilakukan dengan tujuan agar Arduino tidak menjalankan Solenoid Door Lock (1) dan Solenoid Door Lock (2) lagi.

H. Implementasi fungsi mengunci pintu pada waktu tertentu

Algoritma 9 : fungsi mengunci pintu pada waktu tertentu	
1	if (dataJam==21 && dataMenit==00 && dataDetik==00){
2	digitalWrite(13, HIGH);
3	digitalWrite(12, HIGH);
4	}

Pada kode program diatas dapat dilihat bahwa fungsi tersebut akan melakukan pengecekan data waktu yang terdiri dari dataJam, dataMenit, dan dataDetik yang telah didapatkan dari hasil pembacaan RTC. Kemudian apabila nilai pada variable dataJam sudah sama dengan nilai 21 yaitu sama dengan jam 9 malam, nilai variabel dataMenit sudah sama dengan nilai 00, dan nilai variabel dataDetik sudah sama dengan 00, maka pin 13 dan pin 12 akan diberi nilai HIGH sehingga membuat relay bekerja untuk membuat Solenoid Door Lock (1) dan Solenoid Door Lock (2) mengunci.

5.3.2.2 Implementasi Aplikasi Android

A. Implementasi fungsi konektivitas Bluetooth HC-05

Gambar 5. 18 Implementasi fungsi konektivitas Bluetooth

Gambar 5.18 pada bagian atas adalah blok program *BeforePicking* atau sebelum memilih Bluetooth yang akan dihubungkan, maka saat menekan tombol Bluetooth pada aplikasi akan ditampilkan nama Bluetooth dan alamat Bluetooth yang sedang aktif disekitar perangkat *smartphone*, kemudian pada bagian bawah adalah blok program *AfterPicking* atau setelah memilih Bluetooth yang akan dihubungkan, maka nama Bluetooth dan alamat Bluetooth yang sudah dipilih akan disambungkan dengan *smartphone*. Kemudian dilanjutkan pada blok kode tentang keterangan Bluetooth sudah terhubung atau tidak, untuk lebih jelasnya dapat dilihat pada gambar dibawah.

Gambar 5. 19 Implementasi notifikasi konektivitas Bluetooth

Pada Gambar 5.19 berisi blok program Timer dengan menggunakan fungsi if else, dimana apabila BluetoothClient1 sama dengan *IsConnected* maka pada Label1 akan diisi dengan data yang berisi sebuah kata “Connected” dengan warna hijau, kemudian pada fungsi selanjutnya ditambahkan dengan “not” sehingga apabila not BluetoothClient1 sama dengan *IsConnected* maka pada Label1 akan diisi dengan data yang berisi sebuah kata “Disconnected” dengan warna merah. Label1 tersebut akan mempermudah pengguna dalam menggunakan aplikasi, karena pengguna dapat mengetahui apabila Bluetooth pada *smartphone* Android sudah terhubung dengan Bluetooth pada Arduino dengan adanya teks “Connected” ataupun “Disconnected” pada aplikasi Android.

B. Implementasi fungsi pengambilan data perintah suara

The image shows two Scratch code blocks. The first block is a 'when SpeechRecognizer1 BeforeGettingText' block with a 'do' sub-block containing 'set Label2 Text to' followed by an empty text field. The second block is a 'when Button1 TouchDown' block with a 'do' sub-block containing 'call SpeechRecognizer1 GetText'.

Gambar 5. 20 Implementasi fungsi pengambilan data perintah suara

Berdasarkan Gambar 5.20 pada bagian atas berisi block program *BeforeGettingText* atau sebelum mendapatkan teks maka label 2 masih kosong, kemudian pada dibawahnya berisi blok program *TouchDown* atau ditekan, maka apabila Button1 ditekan akan dipanggil SpeechRecognizer1 dan akan dilakukan pengambilan data suara yang langsung diubah ke dalam bentuk teks, dimana teks tersebut akan diproses lebih lanjut pada blok program selanjutnya.

C. Implementasi fungsi pengiriman data perintah

The image shows a Scratch code block 'when SpeechRecognizer1 AfterGettingText' with a 'do' sub-block. Inside the 'do' block, there are four lines of code: 'set Label2 Text to' with 'downcase' and 'SpeechRecognizer1 Result'; 'call BluetoothClient1 SendText' with 'text' set to 'downcase' and 'SpeechRecognizer1 Result'; 'set Label3 Text to' with 'downcase' and 'SpeechRecognizer1 Result'; and 'set Label3 TextColor to' with a blue color swatch.

Gambar 5. 21 Implementasi fungsi pengiriman data perintah

Pada Gambar 5.21 berisi block program *AfterGettingText* atau setelah mendapatkan teks, maka hasil teks dari SpeechRecognizer1 akan diubah setiap hurufnya ke downcase atau ke bentuk huruf kecil, kemudian memasukkan teks tersebut ke dalam Label2. Setelah itu BluetoothClient1 atau Bluetooth pada *smartphone* akan mengirimkan teks hasil dari SpeechRecognizer 1 yang tiap hurufnya telah diubah ke bentuk huruf kecil. Kemudian pada Label3 akan diisi dengan data dari SpeechRecognizer 1 yang tiap hurufnya telah diubah ke bentuk

huruf kecil dan memberikan teks tersebut warna biru dan ditampilkan pada aplikasi Android.

D. Implementasi fungsi penerimaan feedback

Pada Gambar 5.22 dibawah merupakan blok program inialisasi variabel glocal terima yang kemudian diisi data berupa kata “Hello” sebagai inialisasi awal, kemudian bagian bawah terdapat blok program Pembacaan Timer dengan tambahan fungsi if else di dalamnya, pada fungsi *if else* pertama adalah apabila BluetoothClient1 isConnected dan data yang diterima dari Bluetooth terserbut lebih besar atau sama dengan 0 maka variabel global terima akan memasukkan data tersebut untuk disimpan, kemudian data global terima dimasukkan pada variabel Status. Pada fungsi else selanjutnya adalah ketika data dari variabel global terima sama dengan 1 maka pada variabel Status akan dimasukkan kata “Ada orang di dalam rumah dan semua pintu sudah dikunci” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Kemudian pada fungsi if else selanjutnya ketika data dari variabel global terima sama dengan 2 maka pada variabel Status akan dimasukkan kata “Tidak ada orang di dalam rumah dan semua pintu sudah dikunci”.

Gambar 5. 22 Implementasi fungsi penerimaan feedback

Gambar 5.22 implementasi blok program tentang fungsi penerimaan feedback. Kemudian pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Nilai 1 dan 2 dari variabel global tersebut didapat dari feedback sistem dengan perintah program BT.print(1) serta BT.print(2).

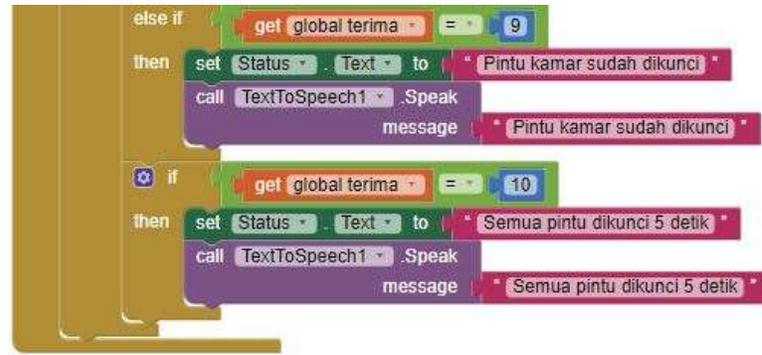
Gambar 5. 23 Implementasi fungsi penerimaan feedback kedua

Gambar 5.23 pada bagian atas adalah blok program fungsi else ketika data dari variabel global terima sama dengan 3 maka pada variabel Status akan dimasukkan kata “Semua sistem sudah dimatikan” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Pada fungsi if else selanjutnya ketika data dari variabel global terima sama dengan 4 maka pada variabel Status akan dimasukkan kata “Ada orang” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Pada fungsi else ketika data dari variabel global terima sama dengan 5 maka pada variabel Status akan dimasukkan kata “Tidak ada orang” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Nilai 3, 4 dan 5 dari variabel global tersebut didapat dari feedback sistem dengan perintah program `BT.print(3)`, `BT.print(4)` serta `BT.print(5)`.



Gambar 5. 24 Implementasi fungsi penerimaan feedback ketiga

Gambar 5.24 pada bagian atas adalah blok program fungsi else ketika data dari variabel global terima sama dengan 6 maka pada variabel Status akan dimasukkan kata “Pintu depan sudah dibuka” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Pada fungsi else ketika data dari variabel global terima sama dengan 7 maka pada variabel Status akan dimasukkan kata “Pintu depan sudah dikunci” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Kemudian pada fungsi if else selanjutnya ketika data dari variabel global terima sama dengan 8 maka pada variabel Status akan dimasukkan kata “Pintu kamar sudah dibuka” dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Nilai 6, 7 dan 8 dari variabel global tersebut didapat dari feedback system dengan perintah program `BT.print(6)`, `BT.print(7)`, serta `BT.print(8)`.

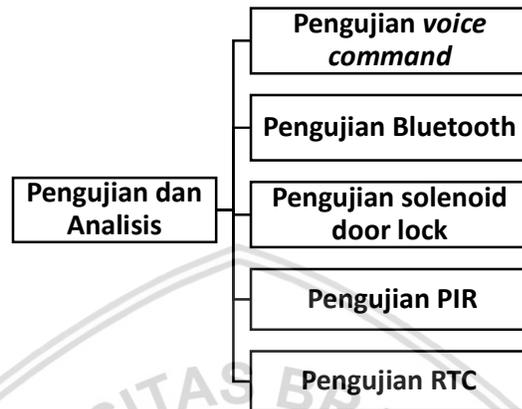


Gambar 5. 25 Implementasi fungsi penerimaan feedback keempat

Gambar 5.25 pada bagian atas adalah blok program merupakan lanjutan dari fungsi if else sebelumnya di mana ketika data dari variabel global terima sama dengan 9 maka pada variabel Status akan dimasukkan kata "Pintu kamar sudah dikunci" dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Pada fungsi if else yang terakhir adalah ketika data dari variabel global terima sama dengan 10 maka pada variabel Status akan dimasukkan kata "Semua pintu dikunci 5 detik" dan pada blok program berikutnya merupakan program TextToSpeech1 untuk pengucapan kata tersebut pada aplikasi Android. Nilai 9 dan 10 dari variabel global tersebut didapat dari feedback sistem dengan perintah program `BT.print(9)` serta `BT.print(10)`.

BAB 6 PENGUJIAN DAN ANALISIS

Bab ini membahas tentang tahapan pengujian yang dilakukan untuk menguji sistem alat pada penelitian ini. Serta analisis yang disimpulkan berdasarkan hasil pengujian.



Gambar 6. 1 Pohon pengujian dan analisis

Pada Gambar 6.1 merupakan pohon pengujian dan analisis yang dijelaskan bahwa pengujian sistem meliputi pengujian solenoid door lock, pengujian PIR dan pengujian RTC.

6.1 Pengujian voice command

Pengujian voice command merupakan bagian dari pengujian sistem dengan tujuan dan prosedur yang telah ditentukan untuk memperoleh hasil serta analisisnya.

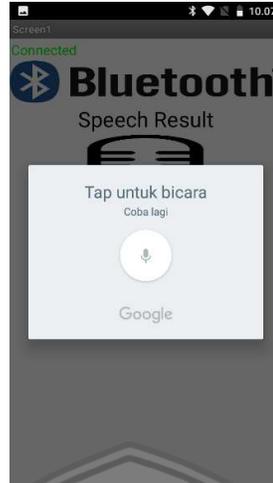
6.1.1 Tujuan

Tujuan dari pengujian *voice command* digunakan untuk menguji fungsionalitas pada sistem dalam proses input perintah suara pada android yang kemudian perintah suara tersebut diubah ke dalam bentuk teks.

6.1.2 Prosedur

Pengujian *voice command* dilakukan dengan beberapa tahapan sebagai berikut :

1. Menyiapkan sistem alat dan bahan serta sebuah *smartphone* yang telah terinstal aplikasi untuk sistem alat.
2. Pengguna menekan tombol mikrofon pada aplikasi, kemudian mengucapkan perintah suara. Untuk gambar tampilan aplikasi android saat memasukkan perintah suara dapat dilihat pada Gambar 6.2 yang ada dibawah ini.



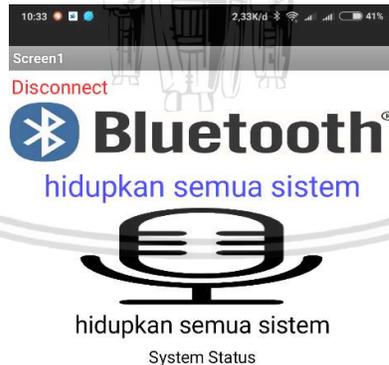
Gambar 6. 2 Menu perintah suara

Pada Gambar 6.2 terdapat sebuah menu untuk memasukkan perintah suara, dengan menekan tombol mikrofon pengguna dapat langsung mengucapkan perintah suara yang nantinya perintah tersebut akan diproses oleh sistem.

3. Mengamati kesesuaian perintah suara dengan teks yang muncul pada aplikasi.
4. Kesimpulan.

6.1.3 Hasil dan Analisis

Hasil pengujian pada sistem ini berupa sebuah teks yang berasal dari input perintah suara dari pengguna yang telah diubah ke dalam bentuk teks. Pada proses ini pengguna mengucapkan perintah suara berupa “hidupkan semua sistem”.



Gambar 6.3 hasil output aplikasi

Gambar 6.3 dapat dilihat bahwa hasil teks tersebut adalah “hidupkan semua sistem”, sehingga dapat dipastikan bahwa fungsi *voice command* dapat berjalan sesuai tujuan.

6.1.4 Analisis Hasil Pengujian

Setelah pengujian voice command telah dilakukan, didapatkan hasil pengujian sistem seperti yang terlihat pada Tabel 6.10. Keberhasilan pengujian

diberi tanda *checklist* (\checkmark) sedangkan pengujian yang tidak berhasil diberi tanda *cross* (\times). Berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi sesuai dengan tujuannya karena perintah suara yang dimasukkan sesuai dengan teks yang muncul pada aplikasi.

Tabel 6. 1 Hasil pengujian mengunci semua pintu selama waktu tertentu

No	Perintah	Sesuai	Total pengujian	Total sukses
1	Hidupkan semua sistem	\checkmark	5	3
2	Matikan semua sistem	\checkmark	5	3
3	Buka pintu depan	\checkmark	5	5
4	Buka pintu kamar	\checkmark	5	5
5	Kunci pintu depan	\checkmark	5	5
6	Kunci pintu kamar	\checkmark	5	5
7	Saya pergi 5 detik	\checkmark	5	5
Total			35	31
Persentase keberhasilan				88,57

Hasil perhitungan tingkat keberhasilan *voice command* sebagai berikut.

$$\text{Nilai Persentase Keberhasilan} = \frac{31}{35} \times 100\% = 88,57\%$$

6.2 Pengujian Bluetooth

Pengujian fungsi pengiriman data menggunakan Bluetooth merupakan bagian dari pengujian sistem dengan tujuan dan prosedur yang telah ditentukan untuk memperoleh hasil serta analisisnya.

6.2.1 Tujuan

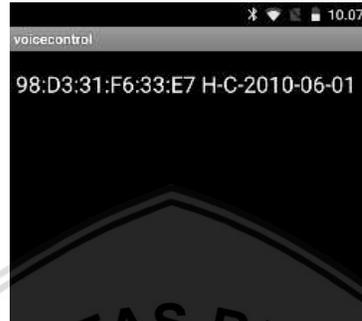
Tujuan dari pengujian Bluetooth digunakan untuk menguji fungsionalitas pada sistem dalam fungsi untuk pengiriman data menggunakan bluetooth, serta hasil data yang telah dikirimkan ditampilkan pada serial monitor Arduino IDE.

6.2.2 Prosedur

Pengujian Bluetooth dilakukan dengan beberapa tahapan sebagai berikut :

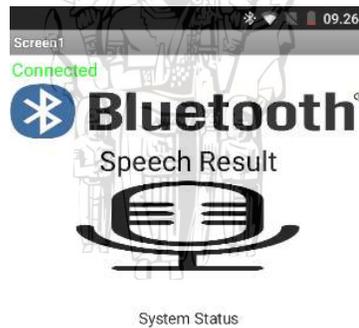
1. Merancang Bluetooth HC-05 yang dihubungkan dengan Arduino Uno sesuai dengan pin-nya menggunakan kabel jumper sehingga Bluetooth dapat bekerja sesuai apa yang diinginkan.
2. Menulis *source code* untuk menjalankan Bluetooth pada Arduino IDE.

3. Menyiapkan sistem alat dan bahan serta sebuah smartphone yang telah terinstal aplikasi untuk sistem alat.
4. Pengguna menekan tombol bertuliskan Bluetooth, kemudian memilih Bluetooth HC-05 untuk menghubungkan aplikasi Android dengan sistem. kemudian amati apabila pada aplikasi sudah tertulis bahwa Bluetooth "*Connedted*", maka Bluetooth pada *smartphone* sudah terhubung dengan HC-05 pada sistem alat.



Gambar 6.4 Menu Bluetooth

Pada Gambar 6.4 dapat dilihat bahwa terdapat Bluetooth HC-05 yaitu modul Bluetooth yang digunakan sebagai media komunikasi antara aplikasi Android dengan mikrokontroler, pilih Bluetooth HC-05 kemudian tunggu hingga pada aplikasi bertuliskan "*Connected*".

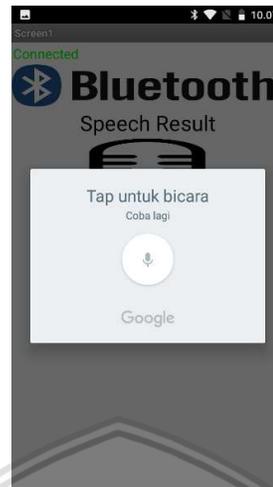


Gambar 6.5 Bluetooth *connected*

Pada Gambar 6.5 dapat diketahui bahwa Bluetooth pada aplikasi Android telah terhubung dengan Bluetooth HC-05 pada mikrokontroler, sehingga aplikasi dapat dijalankan untuk pertukaran data.

5. Pengguna menekan tombol mikrofon pada aplikasi, kemudian mengucapkan perintah suara. Untuk gambar tampilan aplikasi android saat memasukkan perintah suara dapat dilihat pada gambar yang ada dibawah ini.

Pada Gambar 6.6 dibawah terdapat sebuah menu untuk memasukkan perintah suara, dengan mekan tombol mikrofon pengguna dapat langsung mengucapkan perintah suara yang nantinya perintah tersebut akan diproses oleh sistem.

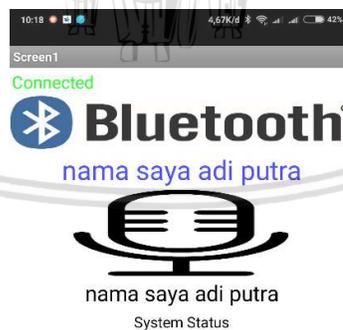


Gambar 6.6 Menu perintah suara

6. Mengamati hasil dari perintah suara yang diucapkan, pada serial monitor Arduino IDE.
7. Kesimpulan.

6.2.3 Hasil dan Analisis

Hasil pengujian pada sistem ini berupa data teks yang ditampilkan pada aplikasi, dimana data teks tersebut merupakan hasil dari input perintah suara yang telah diubah ke dalam bentuk teks. Kemudian data teks tersebut dikirimkan ke Arduino menggunakan media komunikasi Bluetooth dan ditampilkan pada serial monitor. Pada proses ini pengguna memasukkan perintah suara berupa “nama saya adi putra”.



Gambar 6. 7 Hasil output pada aplikasi

Pada Gambar 6.7 diatas dapat dilihat bahwa data teks pada aplikasi sesuai dengan masukan dari pengguna yaitu “nama saya adi putra”. Selanjutnya data akan dikirimkan dari aplikasi smartphone Android ke Arduino Uno dengan menggunakan media komunikasi Bluetooth. Kemudian data yang diterima pada Arduino akan ditampilkan pada serial monitor Arduino IDE.



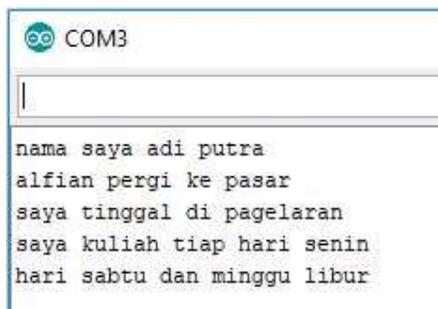
Gambar 6. 8 Hasil output pada serial monitor

Pada Gambar 6.8 dapat dilihat bahwa data yang diterima pada Arduino adalah “nama saya adi putra”, dimana data tersebut telah sesuai dengan data yang dikirimkan oleh aplikasi pada smartphone Android. Kemudian dilakukan pengujian lagi hingga 5 kali pengujian dengan pengiriman data teks yang berbeda, untuk hasil pengujian dapat dilihat pada Gambar 6.9 dibawah ini.



Gambar 6. 9 Hasil output aplikasi

Pada Gambar 6.9 diatas dapat dilihat bahwa data teks pada aplikasi sesuai dengan masukan dari pengguna yaitu “hari sabbtu dan minggu libur”. Selanjutnya data akan dikirimkan dari aplikasi smartphone Android ke Arduino Uno dengan menggunakan media komunikasi Bluetooth. Kemudian data yang diterima pada Arduino akan ditampilkan pada serial monitor Arduino IDE.



Gambar 6. 10 Hasil output pada serial monitor

Pada Gambar 6.10 dapat dilihat bahwa data yang diterima pada serial monitor Arduino adalah “hari sabtu dan minggu libur”, dimana data tersebut telah sesuai dengan data yang dikirimkan oleh aplikasi pada smartphone Android.

6.2.4 Analisis Hasil Pengujian

Setelah pengujian Bluetooth telah dilakukan, didapatkan hasil pengujian sistem seperti yang terlihat pada Tabel 6.10. Keberhasilan pengujian diberi tanda *checklist* (\checkmark) sedangkan pengujian yang tidak berhasil diberi tanda *cross* (\times). Berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi sesuai dengan tujuannya karena perintah suara yang dimasukkan pada aplikasi sesuai dengan data yang muncul pada serial monitor Arduino IDE.

Tabel 6.2 Hasil pengujian mengunci semua pintu selama waktu tertentu

No	Perintah	Sesuai	Total pengujian	Total sukses
1	nama saya adi putra	✓	1	1
2	alfian pergi ke pasar	✓	1	1
3	saya tinggal di pagelaran	✓	1	1
4	saya kuliah tiap hari senin	✓	1	1
5	hari sabtu dan minggu libur	✓	1	1
Total			5	5
Persentase keberhasilan				100

Hasil perhitungan tingkat keberhasilan *voice command* sebagai berikut.

$$\text{Nilai Persentase Keberhasilan} = \frac{5}{5} \times 100\% = 100\%$$

6.3 Pengujian solenoid door lock

Pengujian fungsi menghidupkan semua sistem merupakan bagian dari pengujian sistem dengan tujuan dan prosedur yang telah ditentukan untuk memperoleh hasil serta analisisnya.

6.3.1 Tujuan

Tujuan dari pengujian solenoid door lock digunakan untuk menguji fungsionalitas pada sistem dalam fungsi menghidupkan semua sistem, mematikan semua sistem, membuka pintu depan, membuka pintu kamar, mengunci pintu depan, dan mengunci pintu kamar adalah untuk mengetahui bagaimana sistem bekerja mulai dari proses input perintah suara, pengiriman perintah ke Arduino, proses menghidupkan semua sistem, hingga *feedback* yang didapatkan pada aplikasi Android tentang kondisi keamanan rumah saat ini.

6.3.2 Prosedur

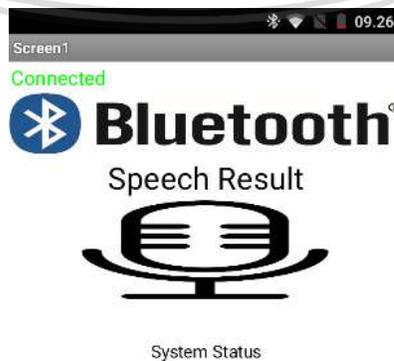
Pengujian solenoid door lock dilakukan dengan beberapa tahapan sebagai berikut :

1. Merancang sensor PIR, relay, dan solenoid door lock yang dihubungkan dengan Arduino Uno sesuai dengan pin-nya menggunakan kabel jumper sehingga sensor PIR, relay dan solenoid door lock dapat bekerja sesuai apa yang diinginkan.
2. Menulis source code untuk menghidupkan semua sistem pada Arduino IDE.
3. Menyiapkan sistem alat dan bahan serta sebuah smartphone yang telah terinstal aplikasi untuk sistem alat.
4. Pengguna menekan tombol bertuliskan Bluetooth, akan muncul nama Bluetooth dan alamat Bluetooth yang terdeteksi disekitar *smartphone*. kemudian memilih Bluetooth HC-05 untuk menghubungkan aplikasi Android dengan Bluetooth yang ada pada Arduino. kemudian amati apabila pada aplikasi sudah tertulis bahwa Bluetooth "*Connedted*", maka Bluetooth pada *smartphone* sudah terhubung dengan HC-05 pada sistem alat.



Gambar 6.11 Menu Bluetooth

Pada Gambar 6.11 dapat dilihat bahwa terdapat Bluetooth HC-05 yaitu modul Bluetooth yang digunakan sebagai media komunikasi antara aplikasi Android dengan mikrokontroler, kemudian pilih nama Bluetooth dan tunggu hingga ada tulisan "*Connected*" dengan warna hijau pada aplikasi.

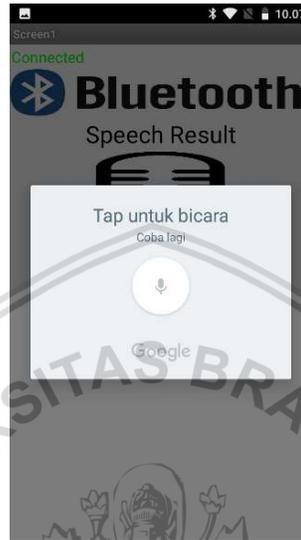


Gambar 6.12 Bluetooth *Connected*

repository.ub.ac.id

Pada Gambar 6.12 dapat diketahui bahwa Bluetooth pada aplikasi Android telah terhubung dengan Bluetooth HC-05 pada mikrokontroler, sehingga aplikasi dapat dijalankan untuk pertukaran data.

5. Pengguna menekan tombol mikrofon pada aplikasi, kemudian mengucapkan perintah suara.



Gambar 6.13 Menu perintah suara

Pada Gambar 6.13 terdapat sebuah menu untuk memasukkan perintah suara, dengan menekan tombol mikrofon pengguna dapat langsung mengucapkan perintah suara yang nantinya perintah tersebut akan diproses oleh sistem.

6. Mengamati hasil dari perintah suara yang diucapkan, apabila perintah suara belum sesuai dengan data perintah suara pada Arduino maka pengguna harus menekan tombol mikrofon kembali dan mengucapkan perintah suara hingga tiap kata tersebut sesuai.
7. Mengamati bagaimana sistem menanggapi suatu perintah tersebut apabila perintah suara yang diberikan salah, serta bagaimana sistem menanggapi suatu perintah apabila perintah suara yang diberikan sudah sesuai dengan data perintah pada Arduino.
8. Mengamati bagaimana *feedback* dari sistem setelah alat bekerja untuk menjalankan solenoid door lock.
9. Kesimpulan.

6.3.3 Hasil Pengujian

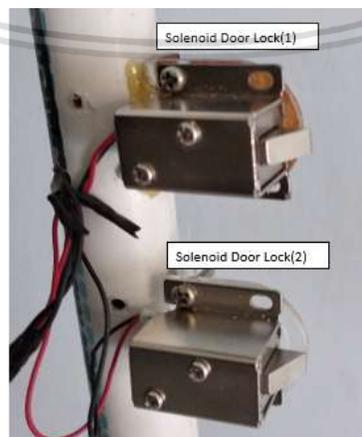
Hasil pengujian pada sistem ini berupa semua sistem telah dihidupkan, dimana solenoid door lock(1) dan solenoid door lock(2) akan mengunci serta sensor PIR akan dijalankan untuk mendeteksi keberadaan seseorang yang ada di dalam rumah. Kemudian setelah semua sistem selesai dijalankan, maka Arduino akan mengirimkan *feedback* ke Aplikasi Android melalui media komunikasi Bluetooth bahwa semua sistem telah dihidupkan, dan mengirimkan kondisi

keamanan rumah saat ini. Untuk gambar sistem secara detail dapat dilihat pada Gambar 6.5 dibawah ini.



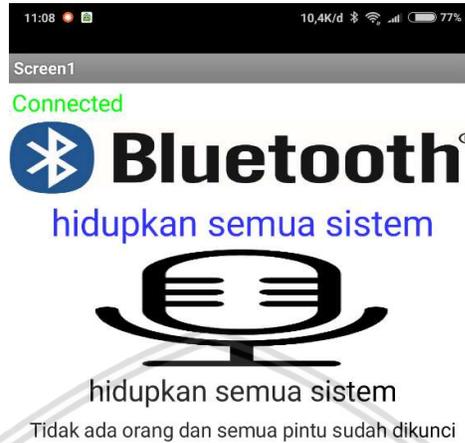
Gambar 6.14 Menghidupkan semua sistem

Pada Gambar 6.14 diatas dapat dilihat kondisi pir berada pada bagian atas pintu yang digunakan untuk melakukan pengecekan tentang keberadaan seseorang, serta solenoid door lock(1) berada pada bagian kiri atas dan solenoid door lock (2) berada pada bagian kiri bawah. Pada pengujian fungsi ini sistem menjalankan sensor PIR serta menjalankan kedua solenoid door lock untuk mengunci, karena dibawah sensor PIR tidak ada seseorang maka seharusnya sensor PIR tidak mendeteksi keberadaan seseorang. Untuk kondisi solenoid door lock yang sudah mengunci dapat dilihat langsung pada perangkat, tetapi untuk hasil pengecekan sensor PIR hanya dapat dilihat pada aplikasi pada smartphone Android bersamaan dengan *feedback* sistem yang memberitahukan bahwa semua pintu sudah dikunci. Untuk lebih jelasnya mengenai kondisi solenoid door lock yang mengunci dapat dilihat pada Gambar 6.6 dibawah ini.



Gambar 6.15 Semua solenoid door lock mengunci

Berdasarkan Gambar 6.15 dapat dilihat bahwa semua solenoid door lock pada pintu tersebut telah mengunci, kemudian untuk hasil *feedback* pada aplikasi Android dapat dilihat pada Gambar 6.16 dibawah ini.



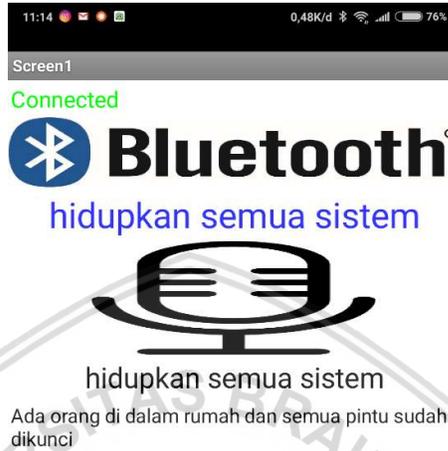
Gambar 6.16 Feedback menghidupkan semua sistem

Berdasarkan Gambar 6.16 dapat dilihat bahwa feedback yang diterima setelah semua sistem dihidupkan adalah “Tidak ada orang di dalam rumah dan semua pintu sudah dikunci”, menandakan bahwa sensor PIR tidak mendeteksi keberadaan seseorang di dalam rumah, serta semua solenoid door lock yang sudah mengunci. Kemudian dilakukan juga pengujian dengan kondisi dimana dibawah sensor PIR ada seseorang, untuk lebih jelasnya dapat dilihat pada Gambar 6.8 dibawah ini.



Gambar 6.17 Menghidupkan semua sistem kedua

Gambar 6.17 menjelaskan bahwa pada kondisi saat ini dibawah sensor PIR terdapat seseorang dan seharusnya sensor PIR mendeteksi keberadaan seseorang, kemudian hasil dari pengecekan tersebut dapat dilihat melalui feedback aplikasi Android. untuk lebih jelasnya mengenai *feedback* pada aplikasi dapat dilihat pada Gambar 6.18 dibawah.



Gambar 6.18 Feedback menghidupkan semua sistem kedua

Berdasarkan Gambar 6.18 dapat dilihat bahwa *feedback* yang didapat pada aplikasi Android adalah “Ada orang di dalam rumah dan semua pintu sudah dikunci”, yang menandakan dimana sensor PIR mendeteksi keberadaan seseorang dan semua solenoid door lock telah mengunci, sehingga dapat diketahui bahwa pada pengujian ini sistem berjalan dengan benar.

Kemudian dilakukan pengujian lagi untuk menghidupkan semua sistem, tetapi sistem tidak dapat berjalan karena perintah suara yang dimasukkan pada aplikasi Android tidak sesuai dengan data perintah suara yang ada pada kode program Arduino, hal tersebut terjadi karena adanya *noise* suara dari luar pada saat pengguna memasukkan perintah suara, untuk lebih jelasnya dapat dilihat pada Gambar 6.19 dibawah ini.



Gambar 6.19 Kesalahan perintah suara

Pada Gambar 6.19 dapat dilihat bahwa perintah suara yang diperoleh aplikasi adalah “hidupkan semua system” sedangkan perintah pada data kode program Arduino adalah “hidupkan semua sistem” sehingga Arduino tidak dapat memproses perintah tersebut karena tidak sesuai dengan data perintah pada kode program. Kemudian pada pengujian selanjutnya juga terjadi *noise* pada saat memasukkan perintah suara yang menyebabkan hasil dari perintah suara tersebut tidak sesuai dengan data perintah yang ada. Untuk lebih jelasnya dapat dilihat pada Gambar 6.11 dibawah ini.

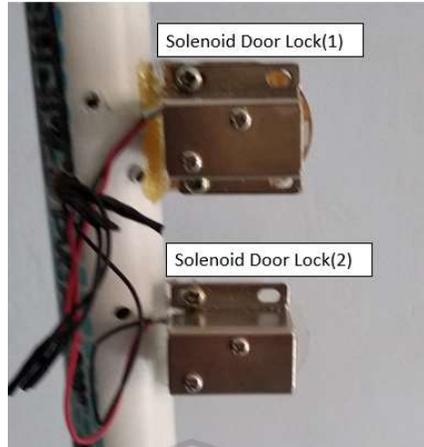


Gambar 6.20 Kesalahan perintah suara kedua

Pada Gambar 6.20 dapat dilihat bahwa perintah suara yang diperoleh aplikasi adalah “hidupkan cuma sistem” sedangkan perintah pada data kode program Arduino adalah “hidupkan semua sistem” sehingga Arduino tidak dapat memproses perintah tersebut karena tidak sesuai dengan data perintah pada kode program, sedangkan pada bagian feedback yang bertuliskan “Ada orang di dalam rumah dan semua pintu sudah dikunci” ada *feedback* dari perintah suara yang sebelumnya, diaman feedback tersebut tidak dihapus dan hanya akan berganti apabila ada feedback berbeda.

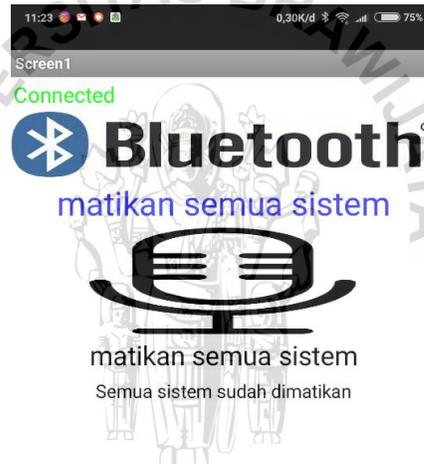
Kemudian pengujian solenoid door lock selanjutnya adalah untuk mematikan semua sistem, dimana pengguna memasukkan perintah suara berupa “matikan semua sistem” pada aplikasi Android. Hasil dari pengujian ini berupa semua sistem yang telah dimatikan, dimana solenoid door lock(1) dan solenoid door lock(2) akan membuka. Kemudian sistem akan mengirimkan *feedback* menggunakan media komunikasi Bluetooth menggunakan modul blueooth dari Arduino kepada aplikasi Android bahwa sistem telah bekerja, dan mengirimkan kondisi keamanan rumah saat ini.

Pada Gambar 6.15 dibawah dapat dilihat bahwa semua solenoid door lock yang ada di dalam rumah membuka, menandakan bahwa sistem telah bekerja sesuai yang tujuannya.



Gambar 6.21 Mematikan semua sistem

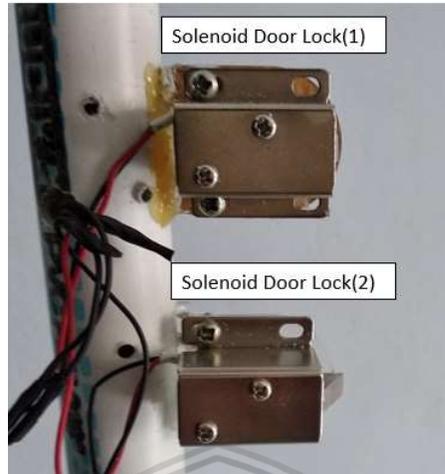
Gambar 6.21 mematikan semua sistem, dimana semua solenoid door lock mengunci. Lalu untuk *feedback* yang didapat pada aplikasi Android dapat dilihat pada gambar 6.22 dibawah



Gambar 6.22 Feedback mematikan semua sistem

Berdasarkan Gambar 6.22 dapat dilihat bahwa *feedback* yang diterima setelah sistem dijalankan adalah “Semua sistem sudah dimatikan”, menandakan semua solenoid door lock yang ada di dalam rumah sudah terbuka.

Kemudian pengujian solenoid door lock selanjutnya adalah untuk membuka pintu depan, dimana pengguna memasukkan perintah suara berupa “buka pintu depan” pada aplikasi Android. Hasil dari pengujian ini berupa solenoid door lock(1) atau pintu depan akan terbuka, dimana solenoid door lock(1) berada pada bagian atas serta solenoid door lock(2) berada pada bagian bawah. Pada Gambar 6.27 dibawah dapat dilihat bahwa solenoid door lock(1) telah membuka kunci rumah bagian depan, menandakan bahwa sistem telah bekerja sesuai dengan tujuan. Pada fungsi ini perintah suara yang diberikan adalah “buka pintu depan” sehingga seolenoid yang terbuka hanyalah solenoid door lock(1) sedangkan solenoid door lock(2) tetap pada posisi terkunci.



Gambar 6.23 Membuka pintu depan

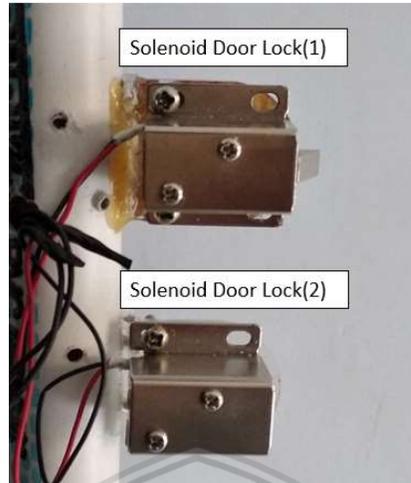
Gambar 6.23 membuka pintu depan, dimana solenoid door lock(1) membuka dan solenoid door lock(2) mengunci atau tetap pada kondisi sebelumnya. Kemudian sistem akan memberikan *feedback* pada aplikasi Android, untuk lebih jelasnya dapat dilihat pada Gambar 6.15 dibawah.



Gambar 6.24 Feedback membuka pintu depan

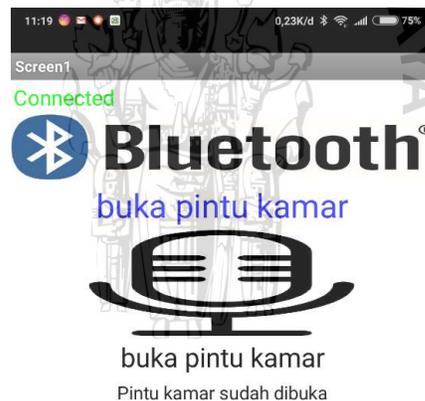
Berdasarkan Gambar 6.24 dapat diketahui bahwa *feedback* yang didapatkan pada aplikasi adalah sebuah pesan bertuliskan “Pintu depan sudah dibuka”, yang menandakan solenoid door lock(1) telah membuka kunci pada pintu depan.

Kemudian pengujian solenoid door lock selanjutnya adalah untuk membuka pintu kamar, dimana pengguna memasukkan perintah suara berupa “buka pintu kamar” pada aplikasi Android. Hasil dari pengujian ini berupa solenoid door lock(2) atau pintu kamar akan terbuka, untuk lebih jelasnya dapat dilihat pada Gambar 6.16 dibawah.



Gambar 6.25 membuka pintu kamar

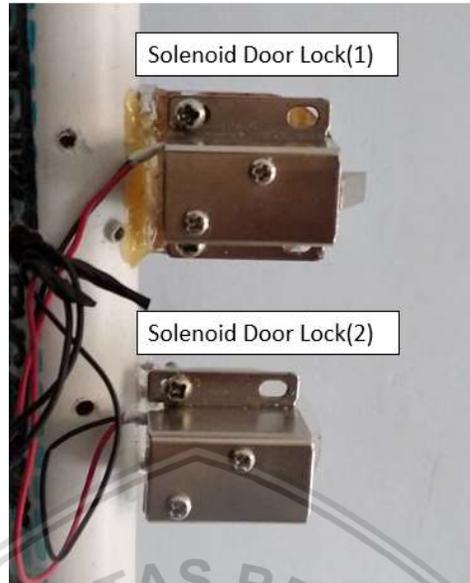
Gambar 6.25 membuka pintu kamar, dimana solenoid door lock(2) membuka dan solenoid door lock(1) mengunci atau tetap pada kondisi sebelumnya. Kemudian sistem akan memberikan *feedback* pada aplikasi Android, untuk lebih jelasnya dapat dilihat pada Gambar 6.17 dibawah.



Gambar 6.26 Feedback membuka pintu kamar

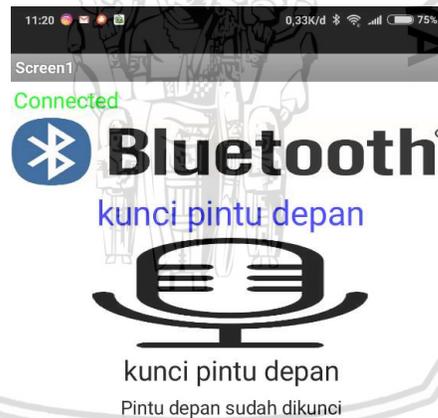
Berdasarkan Gambar 6.26 dapat diketahui bahwa *feedback* yang didapatkan pada aplikasi adalah sebuah pesan bertuliskan “Pintu kamar sudah dibuka”, yang menandakan solenoid door lock(2) telah membuka kunci pada pintu kamar.

Kemudian pengujian solenoid door lock selanjutnya adalah untuk mengunci pintu depan, dimana pengguna memasukkan perintah suara berupa “kunci pintu depan” pada aplikasi Android. Hasil dari pengujian ini berupa solenoid door lock(1) atau pintu depan akan mengunci. Pada Gambar 6.34 dibawah menjelaskan tentang solenoid door lock(1) yang dalam posisi mengunci dan solenoid door lock(2) yang dalam posisi yang sama seperti sebelumnya, untuk lebih jelasnya dapat dilihat pada Gambar 6.18 dibawah.



Gambar 6.27 Mengunci pintu depan

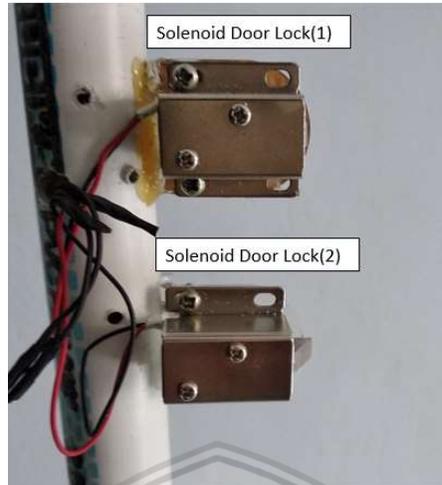
Gambar 6.27 mengunci pintu depan, dimana solenoid door lock(1) mengunci dan solenoid door lock(2) terbuka atau tetap pada kondisi sebelumnya. Kemudian sistem akan memberikan *feedback* pada aplikasi Android, untuk lebih jelasnya dapat dilihat pada Gambar 6.19 dibawah.



Gambar 6.28 Feedback mengunci pintu depan

Berdasarkan Gambar 6.28 dapat diketahui bahwa *feedback* yang didapatkan pada aplikasi adalah sebuah pesan bertuliskan “Pintu depan sudah dikunci”, yang menandakan solenoid door lock(1) telah mengunci pintu rumah bagian depan.

Kemudian pengujian solenoid door lock selanjutnya adalah untuk mengunci pintu kamar, dimana pengguna memasukkan perintah suara berupa “kunci pintu kamar” pada aplikasi Android. Hasil dari pengujian ini berupa solenoid door lock(2) atau pintu depan akan mengunci, untuk lebih jelasnya dapat dilihat pada Gambar 6.20 dibawah.



Gambar 6.29 Mengunci pintu kamar

Gambar 6.29 mengunci pintu kamar, dimana solenoid door lock(2) mengunci dan solenoid door lock(1) terbuka atau tetap pada kondisi sebelumnya. Kemudian sistem akan memberikan *feedback* pada aplikasi Android, untuk lebih jelasnya dapat dilihat pada Gambar 6.30 dibawah.



Gambar 6.30 Feedback mengunci pintu kamar

Berdasarkan Gambar 6.30 dapat diketahui bahwa *feedback* yang didapatkan pada aplikasi adalah sebuah pesan bertuliskan “Pintu kamar sudah dikunci”, yang menandakan solenoid door lock(2) telah membuka kunci pada pintu kamar.

6.3.4 Analisis Hasil Pengujian

Setelah semua pengujian solenoid door lock telah dilakukan, didapatkan hasil pengujian menghidupkan semua sistem seperti yang terlihat pada Tabel 6.1. Keberhasilan pengujian diberi tanda *checklist* (✓) sedangkan pengujian yang tidak berhasil diberi tanda *cross* (×). Berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi sesuai dengan tujuannya apabila perintah suara yang dimasukkan pada aplikasi sesuai dengan data perintah yang ada di kode program Arduino. Pada pengujian ke-3 dan ke-5 perintah suara tidak sesuai dikarenakan adanya noise dari luar saat proses input perintah suara.

Tabel 6. 3 Hasil pengujian menghidupkan semua sistem

Nomor Pengujian	Ada Orang /Tidak	Perintah sesuai	SDL (1)	SDL (2)	PIR	Fungsi Berjalan	Feedback
1	tidak	ya	√	√	√	√	√
2	tidak	ya	√	√	√	√	√
3	tidak	tidak	X	X	X	X	X
4	ada	ya	√	√	√	√	√
5	ada	tidak	X	X	X	X	X

Berikutnya pada Tabel 6.2 merupakan hasil pengujian mematikan semua sistem, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi untuk mematikan semua sistem dikarenakan semua perintah suara telah sesuai.

Tabel 6.4 Hasil pengujian mematikan semua sistem

Nomor Pengujian	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	X	X	√	√
2	ya	X	X	√	√
3	ya	X	X	√	√
4	ya	X	X	√	√
5	ya	X	X	√	√

Berikutnya pada Tabel 6.3 merupakan hasil pengujian membuka pintu depan, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi untuk membuka pintu depan dikarenakan semua perintah suara telah sesuai.

Tabel 6.5 Hasil pengujian membuka pintu depan

Nomor Pengujian	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	√	X	√	√
2	ya	√	X	√	√
3	ya	√	X	√	√
4	ya	√	X	√	√
5	ya	√	X	√	√



Berikutnya pada Tabel 6.4 merupakan hasil pengujian membuka pintu kamar, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi untuk membuka pintu depan dikarenakan semua perintah suara telah sesuai.

Tabel 6.6 Hasil pengujian membuka pintu kamar

Nomor Pengujian	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	X	√	√	√
2	ya	X	√	√	√
3	ya	X	√	√	√
4	ya	X	√	√	√
5	ya	X	√	√	√

Berikutnya pada Tabel 6.5 merupakan hasil pengujian mengunci pintu depan, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi untuk membuka pintu depan dikarenakan semua perintah suara telah sesuai.

Tabel 6.7 Hasil pengujian mengunci pintu depan

Nomor Pengujian	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	√	X	√	√
2	ya	√	X	√	√
3	ya	√	X	√	√
4	ya	√	X	√	√
5	ya	√	X	√	√

Berikutnya pada Tabel 6.6 merupakan hasil pengujian mengunci pintu kamar, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi untuk membuka pintu depan dikarenakan semua perintah suara telah sesuai.

Tabel 6.8 Hasil pengujian mengunci pintu kamar

Nomor Pengujian	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	X	√	√	√
2	ya	X	√	√	√
3	ya	X	√	√	√

4	ya	X	√	√	√
5	ya	X	√	√	√

Dengan semua pengujian solenoid door lock yang telah dilakukan, maka didapatkan tingkat keberhasilan sistem seperti pada Tabel 6.6 dibawah.

Hasil perhitungan tingkat keberhasilan pada sistem sebagai berikut.

$$\text{Nilai Persentase Keberhasilan} = \frac{28}{30} \times 100\% = 93,33\%$$

Tabel 6.9 Data pengujian solenoid door lock

No	Jenis Pengujian	Total Pengujian	Total Sukses
1	Menghidupkan semua sistem	5	3
2	Mematikan semua sistem	5	5
3	Membuka pintu depan	5	5
4	Membuka pintu kamar	5	5
5	Mengunci pintu depan	5	5
6	Mengunci pintu kamar	5	5
Total		30	28
Persentase keberhasilan			93,33%

6.4 Pengujian PIR

Pengujian fungsi mematikan semua sistem merupakan bagian dari pengujian sistem dengan tujuan dan prosedur yang telah ditentukan untuk memperoleh hasil serta analisisnya.

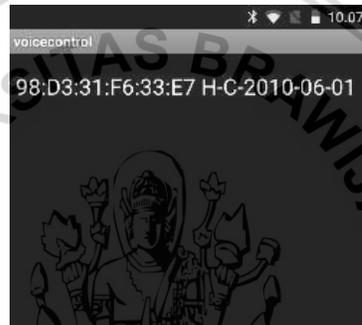
6.4.1 Tujuan

Tujuan dari pengujian PIR adalah untuk mengetahui bagaimana sistem bekerja mulai dari proses input perintah suara, pengiriman perintah ke Arduino, proses menjalankan sensor PIR, hingga *feedback* yang didapatkan pada aplikasi Android tentang kondisi keamanan rumah saat ini. Fungsi ini akan mempermudah pengguna apabila pengguna hanya ingin mendeteksi keberadaan seseorang di dalam rumah saja tanpa menggunakan fungsi untuk mengunci pintu.

6.4.2 Prosedur

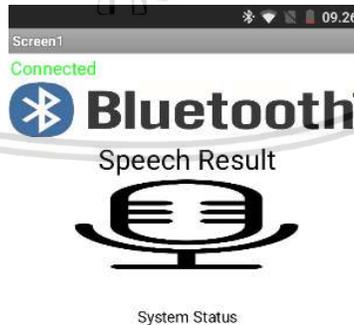
Pengujian PIR dilakukan dengan beberapa tahapan sebagai berikut :

1. Merancang sensor relay, dan solenoid door lock yang dihubungkan dengan Arduino Uno sesuai dengan pin-nya menggunakan kabel jumper sehingga sensor relay dan solenoid door lock dapat bekerja sesuai apa yang diinginkan.
2. Menulis *source code* untuk menghidupkan semua sistem pada Arduino IDE.
3. Menyiapkan sistem alat dan bahan serta sebuah *smartphone* yang telah terinstal aplikasi untuk sistem alat.
4. Pengguna menekan tombol bertuliskan Bluetooth, kemudian memilih Bluetooth HC-05 untuk menghubungkan aplikasi Android dengan sistem. kemudian amati apabila pada aplikasi sudah tertulis bahwa Bluetooth “*Connedted*”, maka Bluetooth pada *smartphone* sudah terhubung dengan HC-05 pada sistem alat.



Gambar 6.31 Menu Bluetooth

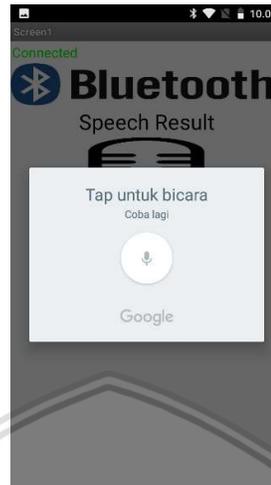
Pada Gambar 6.31 dapat dilihat bahwa terdapat Bluetooth HC-05 yaitu modul Bluetooth yang digunakan sebagai media komunikasi antara aplikasi Android dengan mikrokontroler, pilih Bluetooth HC-05 kemudian tunggu hingga pada aplikasi bertuliskan “*Connected*”.



Gambar 6.32 Bluetooth Connected

Pada Gambar 6.32 dapat diketahui bahwa Bluetooth pada aplikasi Android telah terhubung dengan Bluetooth HC-05 pada mikrokontroler, sehingga aplikasi dapat dijalankan untuk pertukaran data.

5. Pengguna menekan tombol mikrofon pada aplikasi, kemudian mengucapkan perintah suara.



Gambar 6.33 Menu perintah suara

Pada Gambar 6.33 terdapat sebuah menu untuk memasukkan perintah suara, dengan menekan tombol mikrofon pengguna dapat langsung mengucapkan perintah suara yang nantinya perintah tersebut akan diproses oleh sistem.

6. Mengamati hasil dari perintah suara yang diucapkan, apabila perintah suara belum sesuai dengan data perintah suara pada Arduino maka pengguna harus menekan tombol mikrofon kembali dan mengucapkan perintah suara hingga tiap kata tersebut sesuai.
7. Mengamati bagaimana sistem menanggapi suatu perintah tersebut apabila perintah suara yang diberikan salah, serta bagaimana sistem menanggapi suatu perintah apabila perintah suara yang diberikan sudah sesuai dengan data perintah pada Arduino.
8. Mengamati bagaimana *feedback* dari sistem setelah alat bekerja untuk menjalankan sensor PIR.
9. Kesimpulan.

6.4.3 Hasil Pengujian

Hasil pengujian pada sistem ini berupa sensor PIR yang akan bekerja untuk melakukan pengecekan tentang keberadaan seseorang yang ada di dalam rumah, kemudian sistem akan mengirimkan *feedback* ke Aplikasi Android melalui media komunikasi Bluetooth tentang kondisi keamanan rumah saat ini.

Pada Gambar 6.34 dibawah dapat dilihat kondisi pir berada pada bagian atas pintu yang digunakan untuk melakukan pengecekan tentang keberadaan seseorang, serta solenoid door lock(1) berada pada bagian kiri atas dan solenoid door lock (2) berada pada bagian kiri bawah. Pada pengujian fungsi ini sistem menjalankan sensor PIR, dikarenakan dibawah sensor PIR tidak ada seseorang maka seharusnya sensor PIR tidak mendeteksi keberadaan seseorang.



Gambar 6.34 Menjalankan sensor PIR

Pada Gambar 6.34 diatas dapat dilihat kondisi pir berada pada bagian atas pintu yang digunakan untuk melakukan pengecekan tentang keberadaan seseorang, kemudian untuk hasil *feedback* pada aplikasi Android dapat dilihat pada Gambar 6.35 dibawah ini.



Gambar 6.35 Feedback dari sensor PIR

Berdasarkan Gambar 6.35 dapat dilihat bahwa *feedback* yang diterima setelah sistem dijalankan adalah “Tidak ada orang”, menandakan bahwa sensor PIR tidak mendeteksi keberadaan seseorang di dalam rumah. Kemudian dilakukan juga pengujian dengan kondisi dimana dibawah sensor PIR ada seseorang, untuk lebih jelasnya dapat dilihat pada Gambar 6.36 dibawah.



Gambar 6.36 Menjalankan sensor PIR kedua

Gambar 6.36 menjelaskan bahwa pada kondisi saat ini dibawah sensor PIR terdapat seseorang dan seharusnya sensor PIR mendeteksi keberadaan seseorang.



Gambar 6.37 Feedback kedua dari sensor PIR

Berdasarkan Gambar 6.37 dapat dilihat bahwa *feedback* yang didapat pada aplikasi Android adalah “Ada orang”, yang menandakan dimana sensor PIR mendeteksi keberadaan seseorang di bawah pintu, sehingga dapat diketahui bahwa pada pengujian ini sistem berjalan sesuai dengan tujuan.

6.4.4 Analisis Hasil Pengujian

Setelah pengujian sensor PIR telah dilakukan, didapatkan hasil pengujian sistem seperti yang terlihat pada Tabel 6.8. Keberhasilan pengujian diberi tanda *checklist* (✓) sedangkan pengujian yang tidak berhasil diberi tanda *cross* (×). Berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi

sesuai sesuai dengan tujuannya karena perintah suara yang dimasukkan pada aplikasi sesuai dengan data perintah yang ada di kode program Arduino.

Tabel 6.10 Hasil pengujian sensor PIR

No	Ada Orang /Tidak	Perintah sesuai	PIR	Fungsi Berjalan	Feedback
1	tidak	ya	√	√	√
2	tidak	ya	√	√	√
3	tidak	ya	√	√	√
4	ada	ya	√	√	√
5	ada	ta	√	√	√

Dengan pengujian sensor PIR yang telah dilakukan, maka didapatkan tingkat keberhasilan sistem seperti pada Tabel 6.9 dibawah.

Hasil perhitungan tingkat keberhasilan pada sistem sebagai berikut.

$$\text{Nilai Persentase Keberhasilan} = \frac{5}{5} \times 100\% = 100\%$$

Tabel 6.11 Data pengujian solenoid door lock

Jenis Pengujian	Total Pengujian	Total Sukses
Menjalankan sensor PIR	5	5
Total	5	5
Persentase keberhasilan		100%

6.5 Pengujian RTC

Pengujian fungsi menjalankan PIR merupakan bagian dari pengujian sistem dengan tujuan dan prosedur yang telah ditentukan untuk memperoleh hasil serta analisisnya.

6.5.1 Tujuan

Tujuan dari pengujian RTC digunakan untuk menguji fungsionalitas pada sistem dalam fungsi mengunci semua pintu selama waktu tertentu, dan mengunci semua pintu pada waktu tertentu adalah untuk mengetahui bagaimana sistem bekerja mulai dari proses input perintah suara, pengiriman perintah ke Arduino, proses menghidupkan semua sistem, hingga *feedback* yang didapatkan pada aplikasi Android tentang kondisi keamanan rumah saat ini

6.5.2 Prosedur

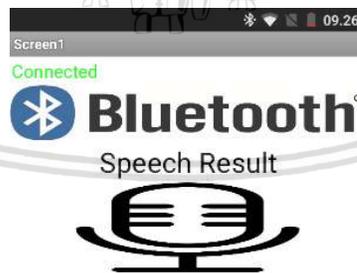
Pengujian RTC dilakukan dengan beberapa tahapan sebagai berikut :

8. Merancang sensor PIR yang dihubungkan dengan Arduino Uno sesuai dengan pin-nya menggunakan kabel jumper sehingga sensor PIR dapat bekerja sesuai apa yang diinginkan.
9. Menulis source code untuk menjalankan sensor PIR pada Arduino IDE.
10. Menyiapkan sistem alat dan bahan serta sebuah smartphone yang telah terinstal aplikasi untuk sistem alat.
11. Pengguna menekan tombol bertuliskan Bluetooth, kemudian memilih Bluetooth HC-05 untuk menghubungkan aplikasi Android dengan sistem. kemudian amati apabila pada aplikasi sudah tertulis bahwa Bluetooth "Connedted", maka Bluetooth pada *smartphone* sudah terhubung dengan HC-05 pada sistem alat.



Gambar 6.38 Menu Bluetooth

Pada Gambar 6.38 dapat dilihat bahwa terdapat Bluetooth HC-05 yaitu modul Bluetooth yang digunakan sebagai media komunikasi antara aplikasi Android dengan mikrokontroler, pilih Bluetooth HC-05 kemudian tunggu hingga pada aplikasi bertuliskan "Connected".



System Status

Gambar 6.39 Bluetooth *connected*

Pada Gambar 6.39 dapat diketahui bahwa Bluetooth pada aplikasi Android telah terhubung dengan Bluetooth HC-05 pada mikrokontroler, sehingga aplikasi dapat dijalankan untuk pertukaran data.

12. Pengguna menekan tombol mikrofon pada aplikasi, kemudian mengucapkan perintah suara. Untuk gambar tampilan aplikasi android saat memasukkan perintah suara dapat dilihat pada gambar yang ada dibawah ini.



Gambar 6.40 Menu perintah suara

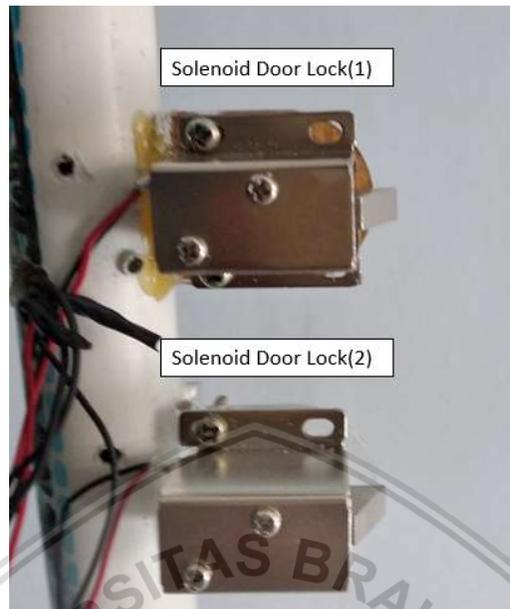
Pada Gambar 6.40 terdapat sebuah menu untuk memasukkan perintah suara, dengan menekan tombol mikrofon pengguna dapat langsung mengucapkan perintah suara yang nantinya perintah tersebut akan diproses oleh sistem.

13. Mengamati hasil dari perintah suara yang diucapkan, apabila perintah suara belum sesuai dengan data perintah suara pada Arduino maka pengguna harus menekan tombol mikrofon kembali dan mengucapkan perintah suara hingga tiap kata tersebut sesuai.
14. Mengamati bagaimana sistem menanggapi suatu perintah tersebut apabila perintah suara yang diberikan salah, serta bagaimana sistem menanggapi suatu perintah apabila perintah suara yang diberikan sudah sesuai dengan data perintah pada Arduino.
15. Mengamati bagaimana *feedback* dari sistem setelah alat bekerja untuk menjalankan RTC.
16. Kesimpulan.

6.5.3 Hasil dan Analisis

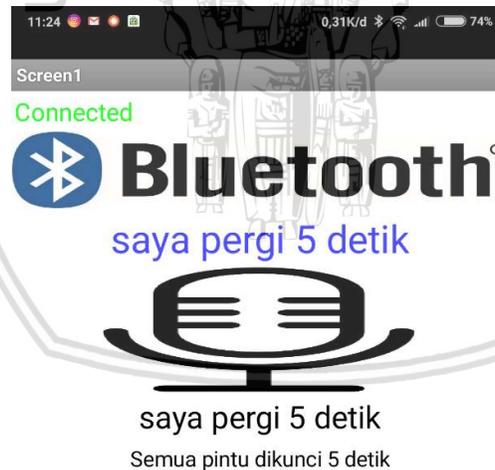
Hasil pengujian pada sistem ini berupa semua pintu akan terkunci selama 5 detik, dimana semua solenoid door lock akan bekerja dan mengunci pintu rumah serta mencatat waktu saat ini yang akan digunakan sebagai acuan untuk 5 detik kedepan, kemudian sistem akan mengirimkan *feedback* ke Aplikasi Android melalui media komunikasi Bluetooth tentang kondisi keamanan rumah saat ini.

Pada Gambar 6.32 dibawah ini dapat dilihat bahwa solenoid door lock(1) sebagai pintu depan dan solenoid door lock(2) sebagai pintu kamar sudah dalam kondisi mengunci, yang menandakan bahwa fungsi tersebut telah berjalan sesuai dengan tujuan.



Gambar 6.41 Mengunci semua pintu 5 detik

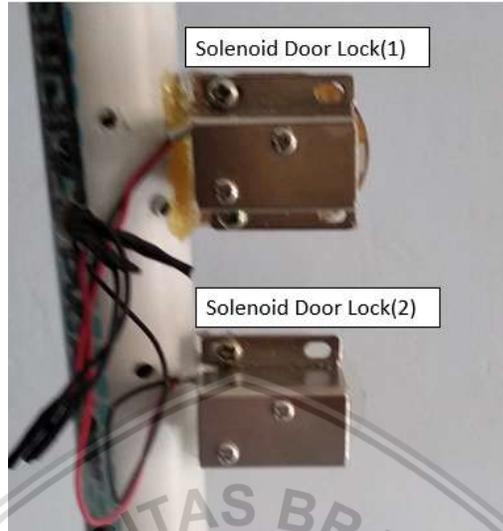
Gambar 6.41 kedua solenoid dalam keadaan mengunci, pada fungsi ini kedua solenoid akan mengunci selama 5 detik, kemudian sistem akan mengirimkan *feedback* bahwa semua pintu dikunci 5 detik, untuk lebih jelasnya mengenai *feedback* pada aplikasi Android dapat dilihat pada Gambar 6.42 dibawah.



Gambar 6.42 Feedback mengunci pintu selama 5 detik

Berdasarkan Gambar 6.42 dapat diketahui bahwa *feedback* yang didapatkan pada aplikasi Android bertuliskan “Semua pintu dikunci 5 detik” yang menandakan semua solenoid door lock telah mengunci pintu rumah selama 5 detik. Selanjutnya adalah proses semua solenoid door lock terbuka kembali, dimana semua solenoid door lock akan bekerja dan membuka semua pintu apabila waktu yang ditentukan telah sesuai dengan waktu saat ini, untuk gambar kedua

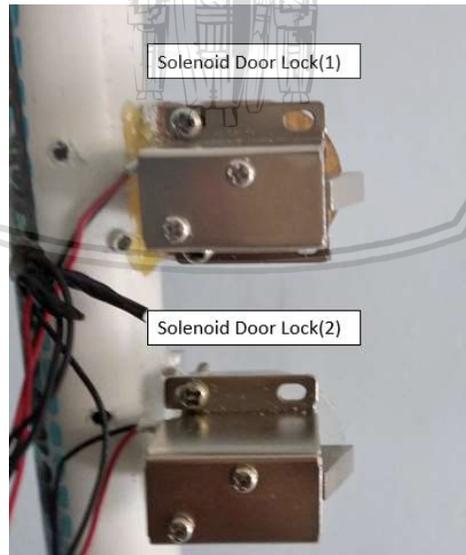
solenoid door lock saat terbuka kembali dapat dilihat pada Gambar 6.43 dibawah ini.



Gambar 6.43 Kedua solenoid door lock terbuka

Pada Gambar 6.43 diatas dapat dilihat bahwa solenoid door lock(1) sebagai pintu depan dan solenoid door lock(2) sebagai pintu kamar sudah dalam kondisi terbuka kembali setelah mengunci selama 5 detik, yang menandakan bahwa fungsi tersebut telah berjalan sesuai dengan tujuan.

Kemudian pengujian RTC selanjutnya adalah untuk mengunci semua pintu pada waktu tertentu, dimana apabila waktu yang telah ditentukan telah sama dengan waktu saat ini, maka semua solenoid door lock akan mengunci, untuk lebih jelasnya dapat dilihat pada Gambar 6.44 dibawah.



Gambar 6.44 Mengunci pintu pada waktu tertentu

Pada Gambar 6.44 diatas dapat dilihat bahwa semua solenoid door lock telah mengunci pintu rumah apabila waktu yang telah ditentukan sudah sama dengan

waktu saat ini. Hal ini dilakukan untuk mengantisipasi apabila pengguna lupa untuk mengunci pintu rumah pada malam hari.

6.5.4 Analisis Hasil Pengujian

Setelah pengujian RTC telah dilakukan, didapatkan hasil pengujian sistem seperti yang terlihat pada Tabel 6.10. Keberhasilan pengujian diberi tanda *checklist* (✓) sedangkan pengujian yang tidak berhasil diberi tanda *cross* (x). Berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi sesuai dengan tujuannya karena perintah suara yang dimasukkan pada aplikasi sesuai dengan data perintah yang ada di kode program Arduino.

Tabel 6.12 Hasil pengujian mengunci semua pintu selama waktu tertentu

No	Perintah sesuai	SDL(1)	SDL(2)	Fungsi Berjalan	Feedback
1	ya	✓	✓	✓	✓
2	ya	✓	✓	✓	✓
3	ya	✓	✓	✓	✓
4	ya	✓	✓	✓	✓
5	ya	✓	✓	✓	✓

Berikutnya pada Tabel 6.11 merupakan hasil pengujian mengunci semua pintu pada waktu tertentu, berdasarkan hasil tersebut terlihat bahwa sistem berhasil menjalankan fungsi tersebut menggunakan RTC.

Tabel 6.13 Hasil pengujian mengunci semua pintu pada waktu tertentu

No	Waktu Terpenuhi	SDL(1)	SDL(2)	Fungsi Berjalan
1	ya	✓	✓	✓
2	ya	✓	✓	✓
3	ya	✓	✓	✓
4	ya	✓	✓	✓
5	ya	✓	✓	✓

Dengan semua pengujian RTC yang telah dilakukan, maka didapatkan tingkat keberhasilan sistem seperti pada Tabel 6.6 dibawah.

Hasil perhitungan tingkat keberhasilan pada sistem sebagai berikut.

$$\text{Nilai Persentase Keberhasilan} = \frac{10}{10} \times 100\% = 100\%$$

Tabel 6. 14 Data pengujian RTC

No	Jenis Pengujian	Total Pengujian	Total Sukses
1	Mengunci pintu selama waktu tertentu	5	5
2	Mengunci pintu pada waktu tertentu	5	5
Total		10	10
Persentase keberhasilan			100%



BAB 7 KESIMPULAN DAN SARAN

7.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis yang telah dilakukan pada penelitian ini, maka dapat diambil kesimpulan sebagai berikut :

1. Android device dapat digunakan untuk pengolahan suara menggunakan voice command untuk nantinya diproses menjadi perintah untuk menjalankan sistem. Berdasarkan pengujian yang dilakukan pada voice command dalam proses untuk mengubah perintah suara ke dalam bentuk teks, hasil pengujian fungsional dinyatakan berhasil dengan tingkat keberhasilan 88,57%.
2. Berdasarkan pengujian yang dilakukan pada Bluetooth dalam proses pengiriman data teks dari aplikasi Android ke Arduino Uno dengan menggunakan media komunikasi Bluetooth, hasil pengujian fungsional dinyatakan berhasil dengan tingkat keberhasilan 100%.
3. Berdasarkan pengujian yang dilakukan pada solenoid door lock, solenoid door lock diprogram menjadi beberapa cara kerja, yaitu semua solenoid door lock mengunci secara bersamaan, semua solenoid door lock membuka secara bersamaan, solenoid door lock membuka pintu depan, solenoid door lock mengunci pintu depan, semua solenoid door lock mengunci selama 5 detik, serta solenoid door lock mengunci pada jam 9 malam, kemudian setelah membuka atau mengunci pintu solenoid door lock akan memberikan feedback tentang kondisi pintu sudah dikunci atau dibuka ke aplikasi Android. Hasil pengujian fungsional dinyatakan berhasil dengan tingkat keberhasilan 93,33%.
4. Berdasarkan pengujian yang dilakukan pada sensor PIR, sensor PIR bekerja untuk pengecekan keberadaan seseorang serta secara langsung memberikan feedback ke aplikasi Android tentang kondisi rumah saat itu. Hasil pengujian fungsional dinyatakan berhasil dengan tingkat keberhasilan 100%.
5. Berdasarkan pengujian yang dilakukan pada RTC, RTC bekerja untuk memberikan informasi tentang waktu saat ini yang kemudian digunakan sebagai referensi waktu untuk menjalankan solenoid door lock, serta secara langsung memberikan feedback ke aplikasi Android tentang kondisi rumah saat itu. Hasil pengujian fungsional dinyatakan berhasil dengan tingkat keberhasilan 100%.

7.2 Saran

Setelah mengambil kesimpulan dari penelitian ini, penulis memiliki beberapa saran untuk proses pengembangan penelitian selanjutnya. Saran dari penulis antara lain sebagai berikut :

1. Mengembangkan aplikasi voice command pada Android dengan cara meningkatkan kualitas tampilan pada aplikasi.

2. Menambahkan beberapa sensor atau aktuator lain pada sistem, maka sistem akan lebih membantu kegiatan sehari – hari.
3. Menambahkan metode *text processing*, sehingga data perintah suara dapat saring menggunakan kata kunci.
4. Menerapkan sistem voice command secara online, dimana pengguna dapat mengontrol sistem dari mana saja.



Daftar Pustaka

- Ali, A. T., Eltayeb, E. B. & Abusail, E. A. A., 2017. Voice Recognition Based Smart Home Control System. *International Journal of Engineering Inventions*.
- Android, 2018. *Android 8.0 Oreo*. [Online] Available at: <https://www.android.com/versions/oreo-8-0/> [Accessed 9 Juli 2018].
- Arduino, 2018. *What is Arduino*. [Online] Available at: <https://www.arduino.cc/en/Guide/Introduction> [Accessed 9 Juli 2018].
- Cloud, G., 2018. *Speech to text*. [Online] Available at: <https://cloud.google.com/speech-to-text/> [Accessed 9 juli 2018].
- Components101, 2017. *DS3231 RTC Module*. [Online] Available at: <https://components101.com/modules/ds3231-rtc-module-pinout-circuit-datasheet> [Accessed 9 juli 2018].
- Components101, 2017. *HC-05 Bluetooth module*. [Online] Available at: <https://components101.com/wireless/hc-05-bluetooth-module> [Accessed 9 juli 2018].
- Components101, 2017. *HC-SR501 PIR Sensor*. [Online] Available at: <https://components101.com/hc-sr501-pir-sensor> [Accessed 9 juli 2018].
- Dani, A. W., Ardiansyah, A. & Hermawan, D., 2016. Perancangan aplikasi voice command recognition berbasis android dan arduino uno. *Teknologi Elektro*.
- Masykur, F. & Prasetyowati, F., 2016. Aplikasi rumah pintar (smart home) pengendali peralatan elektronik rumah tangga berbasis web. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, Volume 3.
- MITAppInventor, 2018. *MIT App Inventor*. [Online] Available at: <http://ai2.appinventor.mit.edu>
- Pallas, F. A., Setyawan, G. E. & Prasetyo, B. H., 2018. Sistem kendali navigasi Quadcopter menggunakan suara melalui Smartphone dan Arduino dengan metode Text Processing. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Raphael, J., 2017. *Computerworld*. [Online] Available at: <https://www.computerworld.com/article/3219126/android/oreo-android-8.html> [Accessed 05 08 2018].
- Sadewo, A. D. B., Widasari, E. R. & Muttaqin, A., 2017. Perancangan pengendali rumah menggunakan smartphone Android dengan konektivitas Bluetooth. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Shandy, Y. D., Rakhmatsyah, A. & Suwastika, N. A., 2015. Implementasi sistem kunci pintu otomatis untuk smart home menggunakan sms gateway. *e-Proceeding of Engineering*.

- Sischalhaci, R., Usman, K. & Hadiyoso, S., 2012. *Aplikasi speech recognition pada sistem otomatis pengatur lampu menggunakan metode sliding window dan FFT (Fast Fourier Transform)*. s.l.:s.n.
- Vyskovsky, A., 2015. *Any Object Tracking and Following*. Prague, IEEE.
- Wicaksono, A. S., Utaminingrum, F. & Fitriyah, H., 2016. *Desain interaksi aplikasi pengendali smart home menggunakan voice command berbasis Android*. s.l.:s.n.

