



**SISTEM KENDALI RUMAH PINTAR (*SMART HOME*)
DENGAN PERINTAH SUARA**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH

NAMA : LINDA YANTI
NPM : 1824370345
PROGRAM STUDI : SISTEM KOMPUTER

**PROGRAM STUDI SISTEM KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2020**

ABSTRAK
LINDA YANTI

**Sistem Kendali Rumah Pintar (*Smart Home*)
Dengan Perintah Suara**

2020

Tujuan pembuatan sistem kendali rumah pintar (*smart home*) dengan perintah suara ini adalah (1) untuk merancang atau membuat sistem kendali rumah pintar (*smart home*) dalam mengendalikan gorden dan lampu, (2) untuk menghubungkan antara motor DC, Arduino, *relay* dan *bluetooth module* dengan Android sebagai masukan perintah suaranya (3) mengetahui cara kerja Android dan *bluetooth module* ketika dihubungkan dengan Arduino.

Metode penelitian yang digunakan adalah dengan mengidentifikasi kebutuhan *hardware* dan *software*, menganalisis kebutuhan sistem, menganalisis sistem yang sedang berjalan, blok diagram rangkaian, rancangan penelitian, *flowchart* program, pengujian alat, dan hasil dari pengujian alat. Mikrokontroler Arduino Uno sebagai pengendali utama, *smartphone* Android sebagai media masukan perintah suara melalui aplikasi rumah pintar yang dirancang menggunakan *MIT App Inventor 2*, *relay module*, *bluetooth module* HC-05, serta *driver* motor IC L298N sebagai pemroses pendukung, *outputnya* adalah 1 buah lampu pijar 5 watt dan motor DC sebagai penggerak gorden.

Hasil pengujian pada *bluetooth* yang mempunyai jarak maksimal 1 hingga 10 meter. Hasil pengujian delay sistem perintah suara dengan tiga orang responden memiliki rata-rata waktu 3 detik. Hasil pengujian lampu dan motor DC sebagai penggerak gorden sudah seperti yang diharapkan ketika diberi perintah suara. Hasil pengujian dari keseluruhan sistem dapat bekerja sesuai dengan fungsinya, dengan mempunyai tingkat keberhasilan 100%.

Kata Kunci : *bluetooth*, perintah suara, rumah pintar, sistem kendali, *smart home*

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
ABSTRAK	iii
KATA PENGANTAR	iv
DAFTAR ISI	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	viii
DAFTAR LAMPIRAN	ix
DAFTAR ISTILAH	x
BAB I PENDAHULUAN	
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
BAB II LANDASAN TEORI	
2.1 Sistem	5
2.2 Sistem kendali.....	6
2.3 Rumah pintar (<i>smart home</i>).....	7
2.4 Arduino uno	7
2.4.1 Daya (<i>power</i>).....	9
2.4.2 Memori	10
2.4.3 Masukan dan keluaran (<i>input/output</i>).....	11
2.5 <i>Bluetooth module</i> HC-05.....	13
2.6 Motor DC.....	15
2.6.1 Komponen Utama Motor DC.....	16
2.7 <i>Driver</i> motor IC L298N	17
2.8 <i>Relay module</i>	19
2.9 Sistem operasi android	22
2.10 <i>MIT app inventor 2</i>	29
2.11 Lampu pijar.....	30

BAB III METODE PENELITIAN	
3.1 Tahapan penelitian	31
3.2 Metode pengumpulan data.....	33
3.3 Analisis sistem sedang berjalan	34
3.4 Rancangan penelitian	38
3.4.1 Perangkat keras (<i>hardware</i>)	38
3.4.2 Perangkat lunak (<i>software</i>)	41
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	
4.1 Kebutuhan Spesifikasi Minimum Hardware dan Software	55
4.2 Pengujian Aplikasi dan Pembahasan.....	56
4.2.1 Pengujian Aplikasi.....	56
4.2.2 Pembahasan.....	60
BAB V PENUTUP	
5.1 Simpulan.....	65
5.2 Saran.....	66
DAFTAR PUSTAKA	
BIOGRAFI PENULIS	
LAMPIRAN-LAMPIRAN	

KATA PENGANTAR

Alhamdulillahirabbil'alamin. Segala puji bagi Allah SWT yang telah memberikan kasih sayangNya, yang telah memberikan rahmat kesehatan dan kesempatan kepada penulis sehingga penulis dapat menyelesaikan skripsi yang berjudul "SISTEM KENDALI RUMAH PINTAR (*SMART HOME*) DENGAN PERINTAH SUARA".

Selama penulisan skripsi ini, penulis banyak menerima bantuan dan dukungan sehingga dapat menyelesaikan skripsi ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Rektor Universitas Pembangunan Panca Budi Medan.
2. Dekan Fakultas Sains dan Teknologi, Bapak Hamdani, ST., MT.
3. Ketua Program Studi Sistem Komputer Bapak Eko Hariyanto, S.Kom,M.Kom.
4. Dosen Pembimbing I Ibu Nadya Andhika Putri, S.Kom,M.Kom.
5. Dosen Pembimbing II Bapak Eka Putra, S.Kom,M.Kom.
6. Mamak tersayang yang telah memberikan dukungan doa dan waktunya untuk menjaga ketiga buah hati penulis, sehingga penulis bisa menyelesaikan skripsi ini.
7. Suami tercinta Candra Wijaya yang telah mendoakan dan memberikan dukungan sepenuh hati untuk membantu dan menjaga anak-anak ketika penulis sedang menyelesaikan skripsi ini.
8. Anak-anakku tersayang Azka Nafil Akbar, Nasyifa Zahrani, Arfan Kamil Faizan yang telah dengan rela ditinggal pergi mamanya untuk berdiskusi dengan teman mama dalam menyelesaikan skripsi ini.
9. Teman seperjuangan kelas Reg J/S L3A khususnya Sri Handayani yang setiap minggu rela didatangi untuk berdiskusi serta memberikan pengalaman dan pengetahuannya dalam membantu penulis.
10. Teman-teman Instalasi SIRS RSUP H.Adam Malik Medan yang telah memberikan dukungannya kepada penulis, terkhusus Ka.Instalasi SIRS RSUP H.Adam Malik Medan.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna karena adanya keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karena itu, kritik dan saran yang berifat membangun akan penulis terima dengan senang hati. Penulis berharap semoga skripsi dapat bermanfaat bagi semua pihak yang membutuhkan.

Medan, 05 Maret 2020

Linda Yanti
1824370345

BAB I

PENDAHULUAN

1. 1 Latar Belakang Masalah

Banyaknya aktivitas yang dilakukan di rumah tangga terkadang membuat kita harus *multitasking* dalam melakukan pekerjaan rumah tangga secara bersamaan. Terkadang sewaktu kita mengerjakan pekerjaan yang satu, pekerjaan yang lainnya lupa untuk kita kerjakan. Aktivitas yang biasa dilakukan di rumah tangga pada saat pagi dan malam hari yaitu membuka dan menutup gorden serta menyalakan atau mematikan lampu ruangan secara manual, dan biasanya seseorang sering lupa menutup gorden pada saat bepergian dan mematikan lampu ketika tidak digunakan. Kedua peralatan ini terkadang dianggap sebagai pekerjaan kecil yang sepele, padahal dampak dari terlupanya mematikan lampu merupakan pemborosan dalam penggunaan listrik sehingga menambah biaya dalam pembayaran listrik rumah tersebut. Begitu juga jika lupa membuka dan menutup gorden secara tepat waktu, bisa mengakibatkan lembabnya udara dalam ruangan karena tidak adanya cahaya yang cukup untuk masuk kedalam ruangan.

Oleh sebab itu penulis ingin memudahkan pekerjaan rumah tangga dalam mengendalikan peralatan yang akan digunakan. Salah satu kemudahan itu ialah dengan adanya rumah pintar (*smart home*). Rumah pintar (*smart home*)

merupakan sebuah aplikasi Android yang dirancang dengan bantuan komputer yang akan memberikan kenyamanan, keamanan dan penghematan energi yang berlangsung secara otomatis sesuai dengan kendali pengguna. Adapun peralatan rumah tangga yang ingin dijadikan sebagai bagian dari rangkaian rumah pintar (*smart home*) adalah gordena dan lampu ruangan. Karena gordena dan lampu ruangan merupakan peralatan yang sangat berperan penting dalam kehidupan sehari-hari.

Berdasarkan latar belakang tersebut diatas, maka dalam penelitian ini penulis mencoba membuat suatu rangkaian Sistem Kendali Rumah Pintar (*smart home*) dengan Perintah Suara untuk memudahkan penghuni rumah dalam mengendalikan peralatan rumah tersebut.

1.2 Perumusan Masalah

Beberapa rumusan masalah aplikasi rumah pintar (*smart home*) ini yaitu :

1. Bagaimana sistem kendali rumah pintar (*smart home*) untuk mengendalikan gordena dan lampu?
2. Bagaimana menghubungkan antara motor DC, Arduino dan *bluetooth module* dengan Android?
3. Bagaimana cara kerja Android dan *bluetooth module* ketika dihubungkan dengan Arduino?

1.3 Batasan Masalah

Adapun batasan masalah yang digunakan penulis dalam skripsi ini yaitu :

1. Rumah pintar (*smart home*) yang dibuat penulis bersifat *prototype* untuk membuka dan menutup gorden serta menghidupkan dan mematikan lampu ruangan.
2. Aplikasi yang digunakan adalah Arduino IDE, MIT App Inventor 2 dan *google assistant*. *Google assistant* hanya didukung oleh *smartphone* dengan OS Android 6.0 keatas.
3. Penghubung perintah suara dengan alat adalah *bluetooth module HC-05*. Penulis tidak membahas secara mendetail tentang aplikasi yang digunakan untuk perintah suara.

1.4 Tujuan Penelitian

Adapun tujuan dari skripsi ini adalah :

1. Untuk menerapkan ilmu pengetahuan yang didapat selama perkuliahan kedalam bentuk perancangan dan pembuatan alat.
2. Untuk mengetahui pengaruh perintah suara terhadap pengendalian peralatan rumah tangga yaitu gorden dan lampu ruangan.
3. Untuk mengetahui apakah aplikasi yang digunakan bisa berjalan seperti yang diharapkan.

1.5 Manfaat Penelitian

Adapun manfaat dari skripsi ini adalah :

1. Agar memudahkan penghuni rumah untuk membuka dan menutup gorden.
2. Agar memudahkan penghuni rumah untuk menyalakan dan mematikan lampu ruangan.
3. Terciptanya efektivitas dan efisiensi dalam penggunaan peralatan rumah tangga.

BAB II

LANDASAN TEORI

2.1 Sistem

Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) adalah suatu kesatuan yang terdiri dari beberapa komponen atau elemen yang dihubungkan secara bersama untuk memudahkan aliran informasi, materi ataupun energi untuk mencapai suatu tujuan tertentu. Sistem juga bisa berarti sekelompok komponen dan elemen yang digabungkan menjadi satu untuk mencapai tujuan tertentu.

Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan atau energi / barang (Murdick, R.G, 1991:27).

Sistem adalah kumpulan-kumpulan dari komponen-komponen yang memiliki unsur keterkaitan antara satu dengan yang lainnya (Indrajit, 2001:2). Sistem adalah himpunan dari bagian-bagian yang saling berhubungan, yang secara bersama mencapai tujuan-tujuan yang sama (Lani Sidharta, 1995:9).

2.2 Sistem Kendali

Dalam kehidupan sehari-hari, sadar atau tidak tanpa kita sadari kita terus bertemu dengan suatu perangkat atau peralatan yang cara kerjanya terkendali secara otomatis baik terkendali sebagian maupun terkendali secara keseluruhan seperti saat menghidupkan lampu, membuka dan menutup gorden, menghidupkan kipas angin, menghidupkan televisi dan masih banyak lagi contoh-contoh yang lainnya. Sistem kendali dibuat untuk mengendalikan perangkat sesuai dengan keinginan manusia itu sendiri.

Sistem kendali merupakan susunan komponen fisik yang terhubung atau terkait sedemikian rupa sehingga dapat memerintah, mengarahkan, atau mengatur diri sendiri atau sistem lain. Sistem kendali terdiri dari sub-sistem dan proses (*plants*) yang disusun untuk mendapatkan keluaran (*output*) dan kinerja yang diinginkan dari input yang diberikan.

Sistem kendali juga merupakan interkoneksi dari komponen-komponen yang membentuk suatu konfigurasi sistem yang akan menghasilkan suatu respon sistem sesuai dengan yang diinginkan, misalnya yaitu membuka dan menutup gorden serta menghidupkan atau mematikan lampu.

2.3 Rumah Pintar (*Smart Home*)

Rumah pintar (*smart home*) adalah sistem yang digunakan untuk mengintegrasikan berbagai sistem yang umumnya terdapat dalam suatu rumah (bangunan). Dengan terintegrasinya berbagai sistem ini diharapkan dapat meningkatkan kenyamanan, keamanan, kemudahan dan efisiensi. Sistem yang umum terdapat pada sebuah rumah konvensional adalah sistem penerangan (*lighting system*), sistem keamanan (*security system*), sistem pengatur temperatur dan ventilasi, sistem hiburan (*entertainment system*). Masing-masing dikendalikan secara terpisah baik dengan menggunakan *remote control*, saklar, dan lainnya.

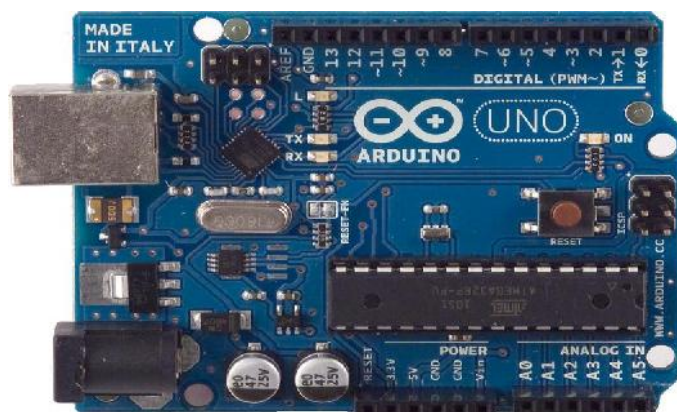
Pada tugas akhir ini penulis menggunakan aplikasi *open-source* berbasis web dari google yaitu MIT *App Inventor 2* untuk membangun aplikasi *smart home* dalam mengendalikan lampu dan gordena dengan perintah suara yang terhubung dengan *bluetooth*.

2.4 Arduino Uno

Arduino merupakan perangkat keras sekaligus perangkat lunak yang memungkinkan siapa saja melakukan pembuatan prototype suatu rangkaian elektronika yang berbasis mikrokontroler dengan mudah dan cepat. Karena sifatnya yang *open source* (terbuka) maka siapa saja dapat mengunduh skema hardware Arduino dan membangunnya (Abdul Kadir, 2016).

Papan Arduino berbasis mikrokontroler yang dikeluarkan oleh perusahaan Atmel. Sebagai contoh, Arduino Uno menggunakan mikrokontroler ATmega328P. Dari sisi perangkat lunak, Arduino IDE adalah *tool* yang bermanfaat untuk menuliskan program, mengompilasinya, dan sekaligus mengunggahnya ke papan Arduino.

Arduino Uno adalah sebuah *board* mikrokontroler yang didasarkan pada ATmega328 (*datasheet*). Arduino Uno mempunyai 14 pin digital *input / output* (6 di antaranya dapat digunakan sebagai output PWM), 6 *input* analog, sebuah osilator kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah ICSP *header*, dan sebuah tombol *reset*. Arduino Uno memuat semua yang dibutuhkan untuk menunjang mikrokontroler, mudah menghubungkannya ke sebuah komputer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk memulainya.



Gambar 2.1 Arduino Uno

Sumber : Febrianto (2014)

2.4.1 Daya (Power)

Arduino Uno dapat disuplai melalui koneksi USB atau dengan sebuah *power* suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2,1 mm ke *power jack* dari *board*. Kabel *lead* dari sebuah baterai dapat dimasukkan dalam *header* / kepala pin *Ground* (Gnd) dan pin Vin dari konektor *power*.

Board Arduino Uno dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt. Jika disuplai dengan yang lebih kecil dari 7 V, kiranya pin 5 Volt mungkin mensuplai kecil dari 5 Volt dan *board* Arduino Uno bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, *voltage* regulator bisa kelebihan panas dan membahayakan *board* Arduino Uno. *Range* yang direkomendasikan adalah 7 Volt sampai 12 Volt.

Adapun pin-pin dayanya adalah sebagai berikut:

1. VIN : Tegangan input ke Arduino *board* ketika *board* sedang menggunakan sumber suplai eksternal (seperti 5 Volt dari koneksi USB atau sumber tenaga lainnya yang diatur).

2. 5V : Pin *output* ini merupakan tegangan 5 Volt yang diatur dari regulator pada *board*. *Board* dapat disuplai dengan salah satu suplai dari DC *power jack* (7-12V), USB *connector* (5V), atau pin VIN dari *board* (7-12).
3. 3V3 : Sebuah suplai 3,3 Volt dihasilkan oleh regulator pada *board*. Arus maksimum yang dapat dilalui adalah 50 mA.
4. GND : Pin ground.

2.4.2 Memori

Pada mikrokontroler ATmega328 mempunyai memori berkapasitas 32 KB (dengan 0,5 KB digunakan untuk *bootloader*). ATmega328 juga mempunyai 2 KB SRAM (*Static Random Access Memory*) dan 1 KB EEPROM (*Electrically Erasable Programmable Read Only Memory*). Memori SRAM adalah tipe memori yang digunakan untuk menyimpan data. Data dalam memori ini akan hilang ketika daya ke mikrokontroler ditiadakan (*volatile*). Memori EEPROM (*Electrically Erasable Programmable Read Only Memory*) adalah tipe *nonvolatile* memori yang digunakan untuk menyimpan data yang sifatnya tidak sering dihapus, misalnya untuk menyimpan konfigurasi tertentu, *password*, atau data-data penting lainnya.

2.4.3 Masukan dan Keluaran (*Input / Output*)

Setiap 14 pin digital pada Arduino Uno dapat digunakan sebagai masukan dan keluaran, menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*. Fungsi-fungsi tersebut beroperasi di tegangan 5 Volt. Setiap pin dapat memberikan atau menerima suatu arus maksimum 40 mA dan mempunyai sebuah resistor *pull-up* (terputus secara *default*) 20-50 kOhm.

Beberapa pin pada Arduino Uno mempunyai fungsi-fungsi khusus yaitu sebagai berikut :

1. Serial : 0 (RX) dan 1 (TX). Pin 0 (RX) yaitu pin yang digunakan untuk menerima (RX) sedangkan pin 1 (TX) ialah pin yang digunakan untuk mengirimkan atau memancarkan (TX) serial data TTL (*Transistor-Transistor Logic*). Kedua pin ini dihubungkan ke pin-pin yang sesuai dari *chip* Serial Atmega8U2 USB-ke-TTL.
2. *External Interrupts* : 2 dan 3. Pin-pin ini dapat dikonfigurasi untuk dipicu sebuah *interrupt* (gangguan) pada sebuah nilai rendah, suatu kenaikan atau penurunan yang besar, atau suatu perubahan nilai.
3. PWM : 3, 5, 6, 9, 10, dan 11. Memberikan 8-bit PWM *output* dengan fungsi *analogWrite()*.
4. SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin-pin ini mendukung komunikasi SPI menggunakan SPI *library*.

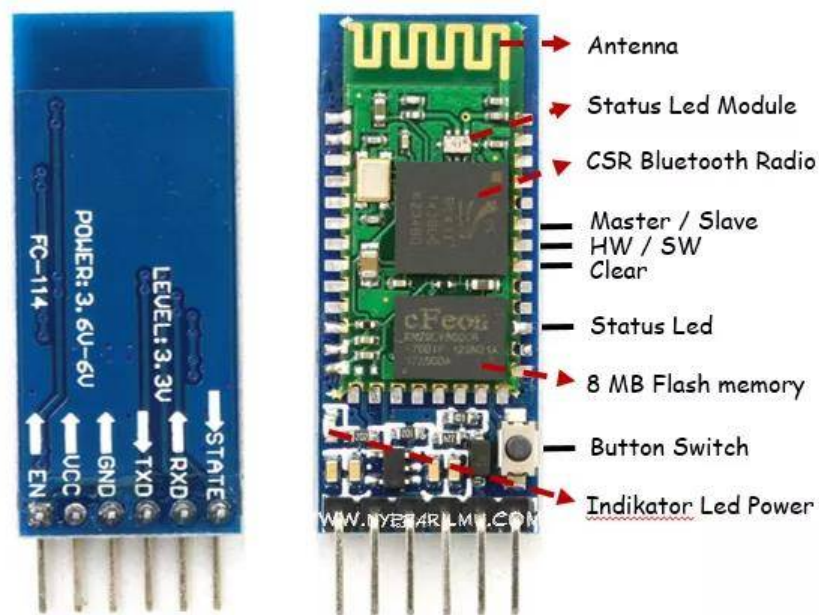
5. LED : 13. Ada sebuah lampu LED yang terpasang, LED tersebut terhubung ke pin digital 13. Ketika pin bernilai *high*, maka LED menyala, ketika pin bernilai *low*, maka LED mati.

Arduino Uno mempunyai 6 *input* analog, pin *input* analog tersebut diberi label A0 sampai A5. Setiap *input* analog tersebut memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara *default* 6 *input* analog tersebut mengukur dari *ground* sampai tegangan 5 Volt. Di sisi lain, beberapa pin mempunyai fungsi khusus yaitu :

1. I2C : A4 (SDA) dan A5 (SCL). Komunikasi yang digunakan pada I2C (TWI) yaitu menggunakan perpustakaan *wire*.
2. AREF : Referensi tegangan untuk *input* analog. Fungsi yang digunakan adalah *analogReference()*.
3. *Reset* : Membawa saluran ini *low* untuk *mereset* mikrokontroler. Secara khusus tombol *reset* untuk melindungi jika ada sesuatu yang menghalangi pada *board*.

2.5 Bluetooth Module HC-05

Bluetooth Module HC-05 adalah modul komunikasi nirkabel via *bluetooth* yang dimana beroperasi pada frekuensi 2.4GHz dengan pilihan dua mode konektivitas. Mode 1 berperan sebagai *slave* atau *receiver* data saja, mode 2 berperan sebagai master atau dapat bertindak sebagai *transceiver*. Modul *bluetooth* HC-05 terdiri dari 6 pin konektor, yang setiap pin konektor memiliki fungsi yang berbeda-beda.



Gambar 2.2 *Module Bluetooth HC-05*

Sumber : nyebarilmu.com (2017)

Tabel 2.1 Konfigurasi pin *Module Bluetooth HC-05*

No	Nama Pin	Nama	Fungsi
1.	Pin 1	<i>Key</i>	-
2.	Pin 2	VCC	Sumber tegangan 5V
3.	Pin 3	GND	<i>Ground</i> tegangan
4.	Pin 4	TXD	Mengirim data
5.	Pin 5	RXD	Menerima data
6.	Pin 6	<i>STATE</i>	-

Sumber : <http://eprints.polsri.ac.id/143/3/BAB%20II%20LA%20lusi.pdf>

Antarmuka yang dipergunakan untuk mengakses modul ini yaitu serial TXD, RXD, VCC serta GND. Pada modul *bluetooth* ini terdapat lampu LED (*built-in*) sebagai indikator koneksi *bluetooth* terhadap perangkat lainnya seperti sesama modul, dengan *smartphone* android, dan sebagainya.

Jangkauan jarak efektif module ini saat terkoneksi dalam *range* 10 meter, dan jika melebihi dari *range* tersebut maka kualitas konektivitas akan semakin kurang maksimal.

2.6 Motor DC

Motor DC merupakan jenis motor yang menggunakan tegangan searah (DC / *Direct Current*) yang secara otomatis akan berputar terus-menerus selama motor ini mendapatkan pasokan tegangan. Motor DC adalah suatu perangkat yang mengubah energi listrik menjadi energi kinetik atau gerakan (*motion*). Motor DC memiliki dua kabel. Kabel yang pertama dihubungkan ke tegangan yang positif dan kabel kedua dihubungkan ke *ground*. Kabel yang dihubungkan ke tegangan positif boleh yang mana saja. Arah putaran rotor (bagian motor yang berputar) ditentukan oleh kabel yang terhubung ke tegangan yang positif.

Motor DC ini menghasilkan sejumlah putaran per menit atau biasanya dikenal dengan istilah RPM (*Revolutions Per Minute*) dan dapat dibuat berputar searah jarum jam maupun berlawanan arah jarum jam apabila polaritas listrik yang diberikan pada motor DC tersebut dibalikkan.



Gambar 2.3 Motor DC

Sumber : Shopee.co.id

2.6.1 Komponen Utama Motor DC

Motor DC memiliki tiga komponen utama, yaitu :

1. Kutub Medan Magnet

Motor DC memiliki kutub medan yang *stasioner* dan kumparan motor DC yang menggerakkan *bearing* pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub medan yaitu kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi bukaan diantara kutub-kutub dari utara ke selatan.

2. Kumparan Motor DC

Ketika arus masuk menuju kumparan motor DC, maka arus ini akan menjadi elektromagnet. Kumparan motor DC yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Kumparan motor DC berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi.

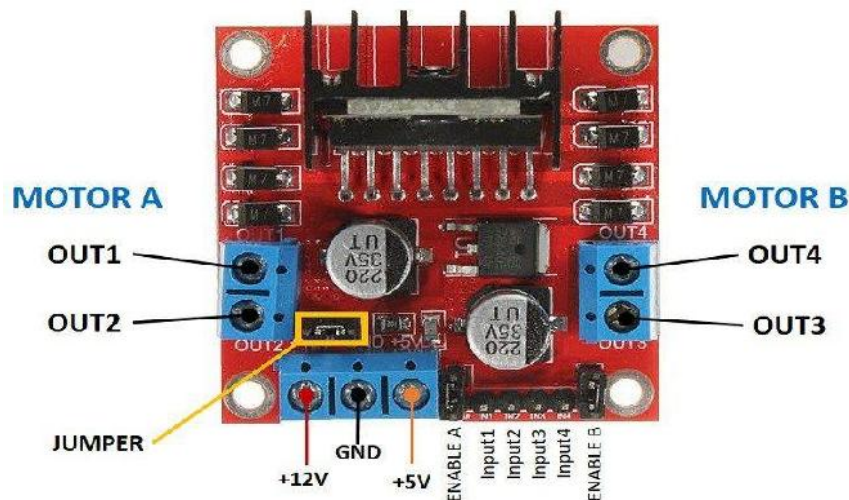
3. *Cummutator* Motor DC

Komponen ini terutama ditemukan dalam motor DC yang kegunaannya adalah untuk membalikkan arah arus listrik dalam kumparan motor DC. *Cummutator* juga membantu dalam transmisi arus antara kumparan motor DC dan sumber daya.

2.7 Driver Motor IC L298N

IC L298N adalah sebuah IC tipe *H-Bridge* yang mampu mengendalikan beban-beban induktif seperti *relay*, *solenoid*, motor DC dan motor *stepper*. Pada IC L298N terdiri dari transistor-transistor logik (TTL) dengan gerbang *nand* yang berfungsi untuk memudahkan dalam menentukan arah putaran suatu motor DC maupun motor *stepper*. IC L298N adalah jenis IC *driver* motor yang dapat mengendalikan arah putaran dan kecepatan motor DC.

Kelebihan dari *driver* motor L298N ini adalah cukup presisi dalam mengontrol motor. Selain itu, kelebihan *driver* motor L298N adalah mudah untuk dikontrol. Pada prinsipnya rangkaian *driver* motor L298N ini dapat mengatur tegangan dan arus sehingga kecepatan dan arah motor dapat diatur.



Gambar 2.4 Driver Motor IC L298N

Sumber : <https://randomnerdtutorials.com/esp32-dc-motor-l298n>

Bagian-bagian dari *driver* motor IC L298N yaitu :

1. Out1, Out2 : Motor A
2. *Jumper* 5V DC : sebagai mode pemilihan sumber tegangan 5V DC, jika tidak di jumper maka akan ke mode sumber tegangan 12V DC.
3. Out3, Out4 : Motor B
4. *Enable* A : berfungsi untuk mengaktifkan bagian output motor A. Hubungkan ke keluaran PWM untuk mengatur kecepatan motor.
5. *Enable* B : berfungsi untuk mengaktifkan bagian output motor B. Hubungkan ke keluaran PWM untuk mengatur kecepatan motor.
6. Input1, Input2, Input3, Input4 : sebagai kontrol pin untuk mengendalikan perputaran dan kecepatan motor yang dihubungkan ke mikrokontroler.
7. *12V jumper* : lepaskan jumper ini jika menggunakan sumber lebih dari 12 V DC. Hubungkan sumber tegangan motor disini, maksimum 35 V DC. Lepaskan 12V jumper jika >12V.
8. GND : *ground*
9. *5V output* : sumber tegangan yang ideal untuk mensuplai ke Arduino.

2.8 *Relay Module*

Relay module adalah saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*).

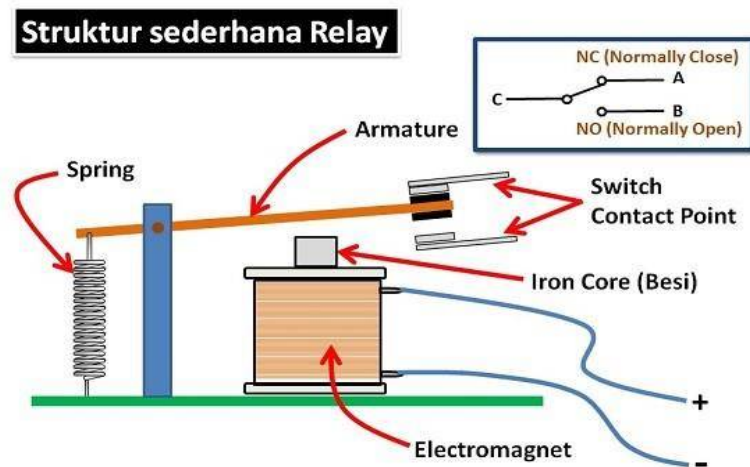


Gambar 2.5 *Module Relay 4 Channel*

Sumber : tokopedia.com

Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Kontak poin (*contact point*) *relay* terdiri dari 2 jenis yaitu :

1. *Normally Close* (NC) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *CLOSE* (tertutup).
2. *Normally Open* (NO) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *OPEN* (terbuka).



Gambar 2.6 Struktur Sederhana *Relay*

Sumber : teknikelektronika.com

Berdasarkan gambar diatas, sebuah besi (*iron core*) yang dililit oleh sebuah kumparan *coil* yang berfungsi untuk mengendalikan besi tersebut. apabila kumparan *coil* diberikan arus listrik, maka akan timbul gaya elektromagnet yang kemudian menarik *armature* untuk berpindah dari posisi sebelumnya (NC) ke posisi baru (NO) sehingga menjadi saklar yang dapat menghantarkan arus listrik di posisi barunya (NO).

Posisi dimana *armature* tersebut berada sebelumnya (NC) akan menjadi *OPEN* atau tidak terhubung. Pada saat tidak dialiri arus listrik, *armature* akan kembali lagi ke posisi Awal (NC). *Coil* yang digunakan oleh *relay* untuk menarik *Contact Poin* ke posisi *CLOSE* pada umumnya hanya membutuhkan arus listrik yang relatif kecil.

Karena *relay* merupakan salah satu jenis dari saklar, maka istilah *pole* dan *throw* yang dipakai dalam saklar juga berlaku pada *relay*. Berikut ini adalah penjelasan singkat mengenai istilah *pole* dan *throw* :

1. *Pole* : banyaknya kontak (*contact*) yang dimiliki oleh sebuah *relay*.
2. *Throw* : banyaknya kondisi yang dimiliki oleh sebuah kontak (*contact*).

Berdasarkan penggolongan jumlah *pole* dan *throw*-nya sebuah *relay*, maka *relay* dapat digolongkan menjadi :

1. *Single Pole Single Throw (SPST)* : *relay* golongan ini memiliki 4 terminal, 2 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
2. *Single Pole Double Throw (SPDT)* : *relay* golongan ini memiliki 5 terminal, 3 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
3. *Double Pole Single Throw (DPST)* : *relay* golongan ini memiliki 6 terminal diantaranya 4 terminal yang terdiri dari 2 pasang terminal saklar sedangkan 2 terminal lainnya untuk *coil*. *Relay* DPST dapat dijadikan 2 saklar yang dikendalikan oleh 1 *coil*.
4. *Double Pole Double Throw (DPDT)* : *relay* golongan ini memiliki terminal sebanyak 8 terminal, diantaranya 6 terminal yang merupakan 2 pasang *relay* SPDT yang dikendalikan oleh 1 (*single*) *coil*. Sedangkan 2 terminal lainnya untuk *coil*.

2.9 Sistem Operasi Android

Android merupakan sistem operasi berbasis *Linux* yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar (*smartphone*) dan komputer tablet. Android, *Inc.* didirikan di Palo Alto, California, pada bulan Oktober 2003 oleh Andy Rubin (pendiri *Danger*), Rich Miner (pendiri *Wildfire Communications, Inc.*), Nick Sears (mantan VP *T-Mobile*), dan Chris White (kepala desain dan pengembangan antarmuka *WebTV*) untuk mengembangkan "perangkat seluler pintar yang lebih sadar akan lokasi dan preferensi penggunanya".

Tujuan awal pengembangan Android adalah untuk mengembangkan sebuah sistem operasi canggih yang diperuntukkan bagi kamera digital. Namun, disadari bahwa pasar untuk perangkat tersebut tidak cukup besar, dan pengembangan Android lalu dialihkan bagi pasar telepon pintar untuk menyaingi *Symbian* dan *Windows Mobile* (*iPhone Apple* belum dirilis pada saat itu).

Tabel 2.2 Tingkatan Versi Android

Nama	Versi	Peluncuran
<i>Apple Pie</i>	1.0	23 September 2008
<i>Banana Bread</i>	1.1	09 Februari 2009
<i>Cupcake</i>	1.5	27 April 2009
<i>Donut</i>	1.6	15 September 2009
<i>Eclair</i>	2.0 – 2.1	26 Oktober 2009
<i>Froyo</i>	2.2 – 2.2.3	20 Mei 2010
<i>Gingerbread</i>	2.3 – 2.3.7	06 Desember 2010
<i>Honeycomb</i>	3.0 – 3.2.6	22 Februari 2011
<i>Ice cream sandwich</i>	4.0 – 4.0.4	19 Oktober 2011
<i>Jelly Bean</i>	4.1 – 4.3.1	09 Juli 2012
<i>KitKat</i>	4.4 – 4.4.4	31 Oktober 2013
<i>Lollipop</i>	5.0 – 5.1.1	25 Juni 2014
<i>Marshmallow</i>	6.0 – 6.0.1	19 Agustus 2015
<i>Nougat</i>	7.0	22 Agustus 2016
<i>Oreo</i>	8.0	21 Agustus 2017
<i>Pie</i>	9.0	06 Agustus 2018

Sumber : id.wikipedia.org/wiki/Daftar_versi_Android

1. Android 1.0 (*Apple Pie*)

Android versi pertama yaitu *Apple Pie*, yang dirilis pada 23 September 2008 dan hanya dilengkapi berbagai fitur seperti *Play Store*, kamera, *Web Browser*, Sinkronisasi antara *G-mail*, *Contacts* dan *Google Agenda*. Selain itu, diawal peluncurannya, Android juga sudah dilengkapi aplikasi *Google Maps* dan dukungan *streaming Youtube*.

2. Android 1.1 (*Banana Bread*)

Versi OS Android selanjutnya yaitu *Banana Bread*, rilis pada bulan Februari 2009. Dan fiturnya yaitu tidak jauh berbeda dengan versi sebelumnya. HTC merupakan salah satu *smartphone* Android pertama yang menggunakan versi ini.

3. Android 1.5 (*Cupcake*)

Dirilis pada awal bulan April 2009 dan juga tidak berbeda dengan versi Android sebelumnya. Hanya saja terdapat fitur tambahan seperti sudah *Support Bluetooth A2DP*, *AVRCP*, *Soft-keyboard* dengan prediksi teks dan *record* atau *watch videos*.

4. Android 1.6 (*Donut*)

Android *Donut* dirilis pada 15 September 2009, dan terdapat fitur tambahan seperti *Gesture Framework* hingga *Turn-by-turn navigation*. Kemudian, Android ini juga terlihat lebih sempurna pada saat itu. Dengan minimnya *bug*

ditambah lebih lengkapnya berbagai fitur yang disediakan oleh *Google*.

5. Android 2.0 (*Eclair*)

Android versi 2.0 ini bernama *Eclair* dan dirilis pada 26 Oktober 2009. Selain terdapat *Bluetooth*, versi ini juga mendapatkan fitur tambahan seperti *multi-touch*, *Live Wallpaper* dan juga *flash* kamera. Kemudian, beberapa fitur yang dapat anda nikmati dalam versi *Eclair* adalah *HTML*, *Digital zoom*, *Support Microsoft Exchange*, dan *Updated UI*.

6. Android 2.2 (*Froyo*)

Android 2.2 *Frozen Yoghurt (Froyo)* dirilis pertama kali pada tanggal 20 Mei 2010. Android 2.2 *Froyo* ini memberikan peningkatan pada kecepatan kerja, fitur *USB tethering*, *WiFi hotspot*, serta fitur keamanan.

7. Android 2.3 (*Gingerbread*)

Android 2.3 *Gingerbread* diluncurkan pada Desember 2010 dengan berbagai peningkatan yang cukup signifikan, terutama pada tampilan tatap muka (*interface*) yang digunakan. Versi sistem operasi ini banyak dilirik oleh banyak *brand smartphone* yang populer hingga saat ini.

8. Android 3.0 – 3.2 (*Honeycomb*)

Versi Android 3.0 – 3.2 *Honeycomb* yang dirilis pada 10 Mei 2011 memang diperuntukkan penggunaannya khusus untuk perangkat tablet. *User interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet.

Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis.

9. Android 4.0 (*Ice Cream Sandwich*)

Puncak kematangan Android yakni ketika pada versi Android 4.0 *Ice Cream Sandwich* yang dirilis pada bulan Oktober 2011. Operasi sistem ini mulai bekerja di semua jenis *smartphone* apapun. Selain bertambahnya fitur-fitur menarik, *Ice Cream Sandwich* juga merupakan versi Android paling banyak disukai pada waktu itu. Bahkan, Android *Ice Cream Sandwich* juga dilengkapi dengan fitur ekstra *multitasking* dan notifikasi yang lebih banyak.

10. Android 4.1.2 (*Jelly Bean*)

Jelly Bean rilis pada 09 Juli 2012 lewat konferensi I/O *Google*. Versi ini merupakan salah satu versi Android yang kerap mendapatkan *update* fitur-fitur yang berguna dan menarik, beberapa halnya adalah seperti memperbaiki rotasi layar, seperti *support* resolusi video 4K. *support* penulisan huruf *Hebrew and Arabic* dari kanan ke kiri, dan peningkatan kinerja, sistem keamanan dan masih banyak lainnya.

11. Android 4.4 (*KitKat*)

Android versi inilah yang saat ini banyak digunakan oleh mayoritas masyarakat Indonesia. *Kitkat* adalah versi Android yang rilis pada 2013 lalu. Pada versi ini, Android banyak mendapatkan pembaharuan fitur, seperti fitur

screen recording, untuk merekam kegiatan yang terjadi pada layar *smartphone* anda, *new translucent* sistem UI, peningkatan akses notifikasi, *system-wide settings* untuk *closed captioning*, peningkatan kinerja dan masih banyak yang lainnya.

12. Android 5.0 (*Lollipop*)

Rilis pada tahun 2014, Android yang satu ini lebih banyak menawarkan fitur tambahan untuk menyempurnakan fitur-fitur yang sudah ada. Dan Nexus 6 adalah salah satu ponsel yang paling pertama mencicipi Android versi ini. Selain itu, *Google* juga lebih menyempurnakan kinerja dari Android *Lollipop* sendiri.

13. Android 6.0 (*Marshmallow*)

Android versi 6.0 merupakan salah satu sistem operasi Android yang rilis pada tahun 2015, yang mana banyak membawa pembaharuan. Salah satunya adalah support *USB Type-C*. Sistem operasi ini secara jelas memberikan peningkatan pada sistem keamanan dengan dihadirkannya *fingerprint* sensor sebagai sistem keamanan biometrik yang digunakan dan daya baterai yang lebih meningkat.

14. Android 7.0 (*Nougat*)

Pada pertengahan 2016, *Nougat* adalah versi Android termutakhir yang baru diperkenalkan pada ajang kumpul *developer Google I/O*. Beberapa lama setelahnya, *Google* menghadirkan *Nougat* secara resmi untuk publik.

Pembaruan paling mendasar pada versi *Nougat* adalah kehadiran *Google Assistant* yang menggantikan *Google Now*. Asisten digital tersebut lebih bisa diandalkan untuk menjalankan berbagai fungsi. Fitur-fitur baru lainnya mencakup layar *split-screen* saat dipakai *multitasking*, serta fitur *Doze* yang telah dikenalkan di versi Android *Marshmallow* namun telah ditingkatkan. Android *Nougat* juga memiliki dukungan terhadap *platform virtual reality* terbaru *Google*.

15. Android 8.0 – 8.1 (*Oreo*)

Android 8.0 & 8.1 *Oreo* menjadi sistem operasi Android paling terbaru yang banyak digunakan saat ini. Sistem operasi ini dirilis secara stabil mulai Agustus 2017 sudah mengalami pembaruan lewat versi Android 8.1 *Oreo*. Sistem operasi ini menawarkan pengalaman *multitasking* yang makin mumpuni dibanding versi sebelumnya.

16. Android 9.0 (*Pie*)

Android 9.0 *Pie* secara resmi diperkenalkan pada Agustus 2018. Sistem operasi Android ini memberi banyak perubahan, terutama untuk *smartphone* dengan desain baru. Misal Android 9.0 *Pie* memberikan navigasi berupa *gesture* yang menggantikan tombol fisik *Home*, *Back*, dan *Recent Apps*. Fitur lainnya yang cukup berguna adalah sistem notifikasi, pengatur kecerahan, hingga sistem *screenshot* terbaru yang lebih memudahkan pengguna.

2.10 MIT App Inventor 2

MIT App Inventor 2 merupakan aplikasi *open source* berbasis web yang semulanya disediakan oleh *Google* dan sekarang dikelola oleh *Massachusetts Institute of Technology (MIT)*. *App Inventor 2* merupakan *platform* untuk memudahkan proses pembuatan aplikasi Android sederhana tanpa harus mempelajari atau menggunakan bahasa pemrograman yang terlalu banyak. *App Inventor 2* didesain dengan sederhana dan mudah dimengerti, kita tidak harus bersentuhan dengan dunia pemrograman komputer yang kompleks.

Blok-blok kode program dalam *MIT App Inventor 2* sudah disediakan sedemikian rupa dalam bentuk potongan *puzzle* (teka-teki). Kita hanya melakukan *drag and drop* (geser dan lepaskan) blok-blok kode program kedalam *Blocks Editor*. Kemudian kita menyusun blok-blok kode program tersebut seperti halnya ketika menyusun *puzzle* (teka-teki) sesuai dengan alur program yang dikehendaki.



Gambar 2.7 Tampilan *MIT App Inventor 2*

2.11 Lampu Pijar

Lampu pijar adalah sumber cahaya buatan yang dihasilkan melalui penyaluran arus listrik melalui filamen yang kemudian memanans dan menghasilkan cahaya. Komponen utama dari lampu pijar adalah bola lampu yang terbuat dari kaca, filamen yang terbuat dari wolfram, dasar lampu yang terdiri dari filamen, gas pengisi dan kaki lampu.

Pada dasarnya filamen pada sebuah lampu pijar adalah resistor. Saat dialiri arus listrik, filamen tersebut menjadi sangat panas, hal ini yang menyebabkan warna cahaya yang dipancarkan oleh lampu pijar biasanya berwarna kuning kemerahan.



Gambar 2.8 Lampu Pijar

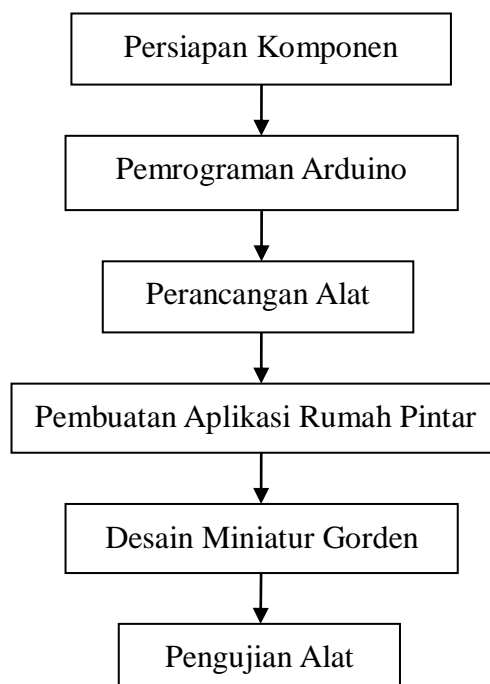
Sumber : id.wikipedia.org

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Sistem kendali rumah pintar (*smart home*) ini dirancang untuk mengendalikan lampu ruangan sekaligus mengendalikan gorden melalui perintah suara yang membutuhkan beberapa tahapan penelitian yaitu :



Gambar 3.1 Tahapan Penelitian

1. Komponen yang dibutuhkan untuk mendukung penelitian ini adalah sebagai berikut :
 - a. Arduino Uno
 - b. *Relay Module*
 - c. *Bluetooth Module* HC-05
 - d. *Breadboard*
 - e. *Driver Motor* IC L298N
 - f. Motor DC
 - g. Kabel *Jumper*
 - h. Adaptor AC/DC 12V
 - i. Baterai Kotak 9 V
2. Pemrograman Arduino
3. Melakukan perancangan alat
4. Pembuatan aplikasi rumah pintar (*smart home*). Aplikasi rumah pintar (*smart home*) dibutuhkan sebagai media masukan suara melalui *smartphone* Android yang dapat dibuat dengan *MIT App Inventor 2*.
5. Membuat desain rangkaian mekanik miniatur gorden dari *styrofoam*
6. Melakukan pengujian alat yang telah dirakit sehingga didapatkan hasil kinerja alat seperti yang diinginkan.

3.2 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam tugas akhir ini adalah dengan mempelajari beberapa sumber yang berkaitan dengan sistem kendali rumah pintar (*smart home*) dengan perintah suara. Dari beberapa sumber yang sudah dipelajari, maka aplikasi rumah pintar beserta rangkaian yang akan digunakan dalam proyek ini dapat dibuat *prototype* nya. Adapun metode penelitian yang digunakan adalah mengidentifikasi kebutuhan *hardware* dan *software*.

1. Identifikasi Perangkat Keras (*Hardware*)

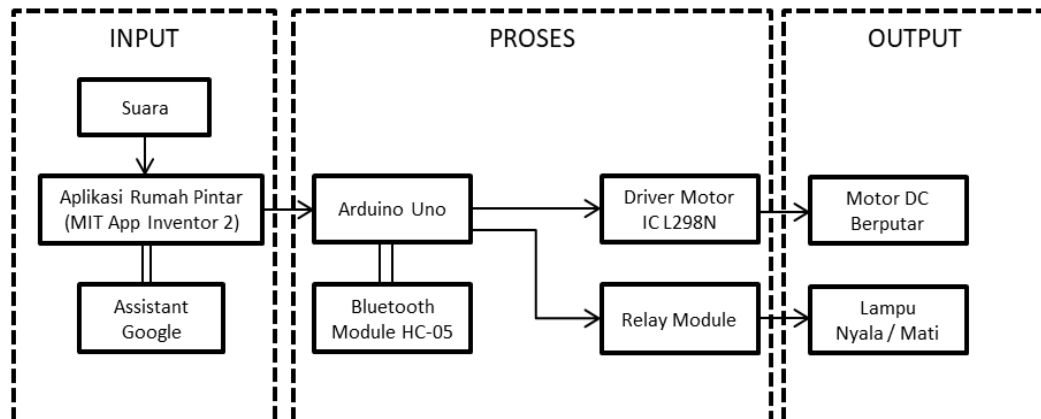
- a. Arduino Uno
- b. *Relay Module*
- c. *Bluetooth Module HC-05*
- d. *Driver Motor IC L298N*
- e. Motor DC
- f. 1 lampu pijar

2. Identifikasi Perangkat Lunak (*Software*)

- a. Arduino IDE
- b. Aplikasi *MIT App Inventor 2* sebagai media masukan perintah suara

3.3 Analisis Sistem Sedang Berjalan

Secara keseluruhan, sistem yang sedang berjalan dapat digambarkan pada blok diagram berikut ini :



Gambar 3.2 Blok Diagram Rangkaian

1. Proses Pemasukan Data

Pada proses pemasukan data dapat dijelaskan bahwa suara, aplikasi rumah pintar dan *assistant google* yang ada pada *smartphone* Android sebagai media masukan (*input*). Perintah suara yang diterima oleh *smartphone* Android selanjutnya akan diproses melalui aplikasi rumah pintar dan akan dicocokkan dengan *database google assistant*, setelah itu suara yang sudah diperoleh selanjutnya akan dikirimkan ke Arduino untuk diolah.

2. Proses Transaksi Data

Dari media input tersebut diproses kedalam rangkaian Arduino yang telah dipasangkan dengan *bluetooth module* HC-05 sebagai modul komunikasi antara

rangkaian dan *smartphone*. *Driver* motor IC L298N dan *relay module* juga dihubungkan ke Arduino. Setelah semua komponen terhubung, rangkaian tersebut diberikan *codingan* program Arduino yang sesuai sehingga rangkaian tersebut dapat berjalan sesuai dengan perintah yang kita berikan.

3. Proses Keluaran Data

Pada bagian *output* (keluaran) ini, *driver* motor IC L298N akan membuat motor DC berputar dan *relay module* yang telah dipasangkan lampu pijar akan mengendalikan lampu ruangan. Ketika diberi masukan perintah suara “nyalakan lampu” lampu akan menyala, ketika diberi perintah suara “matikan lampu” lampu akan mati. Begitu juga pada ketika diberi perintah suara “buka gorden” tirai gorden akan terbuka, ketika diberi perintah suara “tutup gorden” gorden akan tertutup.

4. Kelemahan-Kelemahan Proses Sistem Sedang Berjalan

Adapun beberapa kelemahan yang terdapat pada sistem kendali rumah pintar (*smart home*) dengan perintah suara ini adalah terdapatnya *delay* atau waktu tunggu ketika kita memberikan perintah suara dari *smartphone* ke rangkaian.

5. Analisis Kebutuhan Sistem

Pada tahapan ini, *output* atau keluaran yang ingin dihasilkan ialah menyalakan dan mematikan lampu serta membuka dan menutup gorden. Berdasarkan identifikasi kebutuhan akan perangkat keras (*hardware*) dan perangkat lunak

(*software*), maka diperoleh beberapa analisis kebutuhan terhadap sistem yang akan dirancang adalah sebagai berikut :

A. Perangkat Keras (*Hardware*)

a. Arduino Uno

Pada bagian proses, alat ini membutuhkan komponen yang dapat mengolah data dari masukan yang akan dikirim ke bagian keluaran. Pada proses ini Arduino Uno digunakan sebagai pengendali utama yang akan melakukan pemrosesan data dari masukan untuk selanjutnya mengatur keputusan dari jalannya alat yang akan dikirimkan ke bagian keluaran.

b. *Bluetooth Module* HC-05

Pada rangkaian rumah pintar (*smart home*) ini menggunakan *Bluetooth Module* HC-05 sebagai penghubung antara alat dan *smartphone Android* yang digunakan sebagai masukan perintah suara.

c. *Relay Module*

Dibutuhkan *relay* yang digunakan sebagai *switch* untuk 1 lampu ruangan yang berfungsi sebagai saklar lampu.

d. Lampu Pijar

Dibutuhkan 1 lampu ruangan untuk menjalankan rangkaian ini.

e. *Driver Motor* IC L298N

IC L298N adalah jenis IC driver motor yang dapat mengendalikan arah

putaran dan kecepatan motor DC ataupun motor stepper. IC L298N mampu mengeluarkan output tegangan untuk motor DC dan motor stepper sebesar 50 Volt. IC L298N terdiri dari Transistor-Transistor Logic (TTL) dengan gerbang nand yang memudahkan dalam menentukan arah putaran suatu motor DC dan motor stepper. IC L298N adalah driver motor berbasis H-Bridge yang mampu menangani beban hingga 4A pada tegangan 6V–46V.

f. Motor DC

Motor DC (*Direct Current*) digunakan untuk menggerakkan rangkaian gorden yang akan dipakai pada proyek tugas akhir ini. Motor DC ini menggunakan tegangan *input* 3V-6V DC.

Tabel 3.1 Kebutuhan Alat dan Bahan

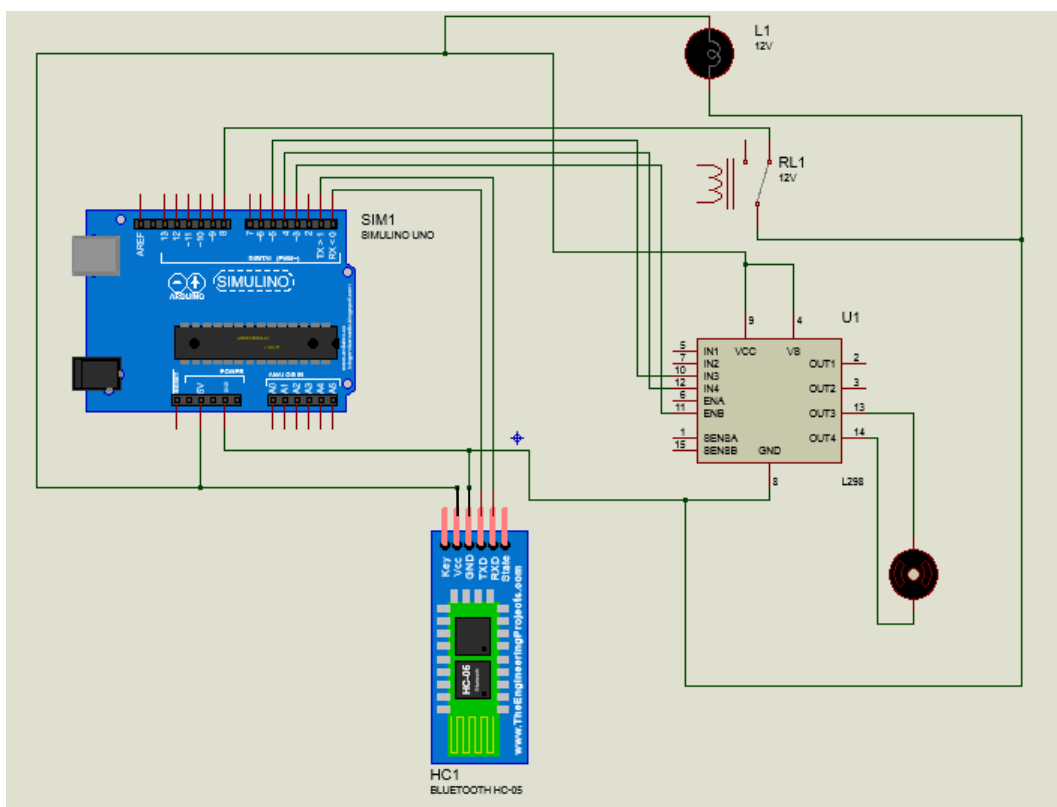
No	Rangkaian	Komponen	Spesifikasi
1	<i>System Minimum</i>	Arduino	Arduino Uno
		<i>Relay Module</i>	4 channel
2	<i>Input</i>	<i>Smartphone</i>	Android 9.0 (<i>Pie</i>)
		Koneksi Internet	Jaringan 3G/4G
		Adaptor	12 V AC/DC
		Baterai	9 V
3	<i>Output</i>	1 Lampu Pijar	220V AC
		Motor DC	3V-6V DC

3.4 Rancangan Penelitian

Rancangan penelitian sistem kendali rumah pintar (*smart home*) dengan perintah suara meliputi perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*).

3.4.1 Perangkat Keras (*Hardware*)

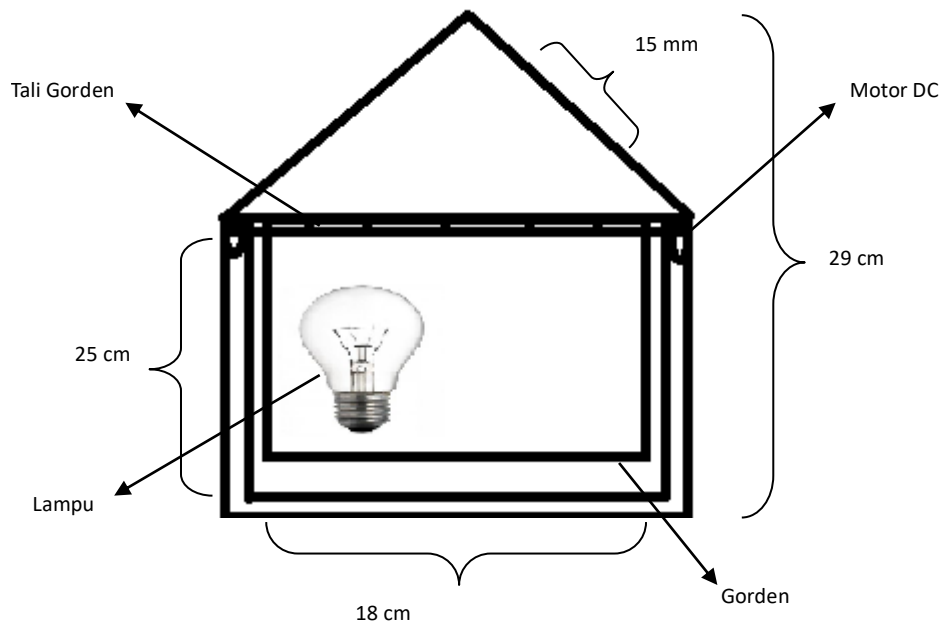
Pada tugas akhir ini dibutuhkan perancangan *hardware* yang meliputi rancangan komponen-komponen dan rancangan mekanik miniatur gorden sebagai tempat motor DC untuk menggerakkan tirai gorden.



Gambar 3.3 Rancangan Komponen

Gambar diatas merupakan gambar rancangan komponen-komponen yang dibutuhkan untuk menjalankan rangkaian lampu dan gorden. Adapun konfigurasi dari pin-pin diatas adalah :

- a. Pin 5V Arduino dipasangkan ke pin VCC *bluetooth module*
- b. Pin GND Arduino dipasangkan ke pin GND *bluetooth module*
- c. Pin 0 (RX) Arduino dipasangkan ke pin TXD *bluetooth module*
- d. Pin 1 (TX) Arduino dipasangkan ke pin RXD *bluetooth module*
- e. Pin 5V Arduino dipasangkan ke pin VCC *relay module*
- f. Pin GND Arduino dipasangkan ke pin GND *relay module*
- g. Pin 8 Arduino dipasangkan ke IN1 pada *relay module*
- h. Pin IN3 *driver motor IC L298N* dihubungkan ke pin 5 Arduino
- i. Pin IN4 *driver motor IC L298N* dihubungkan ke pin 4 Arduino
- j. Pin Enable B *driver motor IC L298N* dihubungkan ke pin 3 Arduino
- k. Pin GND *driver motor IC L298N* dihubungkan ke pin GND Arduino
- l. Pin 5V *driver motor IC L298N* dihubungkan ke pin 5V Arduino
- m. Pin 12V *driver motor IC L298N* dihubungkan ke adaptor agar menghasilkan daya sebesar 12V.
- n. Kabel positif pada baterai dihubungkan ke 12V pada driver motor IC L298N
- o. Kabel negatif pada baterai dihubungkan ke Ground pada driver motor IC L298N



Gambar 3.4 Rancangan Rangkaian Miniatur Gorden dan Lampu

Gambar diatas merupakan gambar miniatur gorden yang di desain menggunakan *styrofoam* setebal 15 mm dengan dimensi panjang 29 cm, lebar 18 cm dan tinggi 25 cm. Perancangan ini memerlukan ketelitian dan ketepatan dalam meletakkan posisi motor DC agar rel gorden bisa berjalan sempurna tanpa mengalami hambatan saat bergerak.

Begitu juga dengan lampu ruangan, diperlukan ketelitian dan ketepatan dalam memasang kabel kedalam saklar yang terdapat pada *relay*. *Relay* digunakan untuk mengkoneksikan sambungan lampu agar lampu tersebut bisa dikendalikan.

3.4.2 Perangkat Lunak (*Software*)

Pada perancangan perangkat lunak (*software*), penulis menggunakan perancangan program Arduino dan aplikasi rumah pintar yang dirancang menggunakan *MIT App Inventor 2*.

1. Perangkat Lunak (*Software*)

a. *Software* Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah *software* yang digunakan untuk memprogram di arduino, dengan kata lain Arduino IDE sebagai media untuk memprogram *board* Arduino. Dalam pemrograman Arduino ini sendiri menggunakan bahasa pemrograman C.

Arduino IDE ini berguna sebagai *text editor* untuk membuat, mengedit, dan juga memvalidasi kode program. Arduino IDE juga digunakan untuk meng-*upload* kodingan program ke board Arduino. Kode program yang digunakan pada Arduino disebut dengan istilah Arduino *sketch* atau disebut juga *source code* arduino, dengan ekstensi file *source code .ino*. Dalam setiap *sketch* memiliki dua buah fungsi penting yaitu “*void setup() {}*” dan “*void loop() {}*”. Adapun pembuatan program pada Arduino IDE ini dimulai dengan menginisialisasi pin-pin yang mana saja yang akan digunakan oleh sistem.

Input yang digunakan pada codingan Arduino adalah :

```
int lampu = 8;
int IN3 = 5;
int IN4 = 4;
int ENB = 3;
int SPEED = 100;
```

Inisialisasi pin yang digunakan untuk rangkaian ini adalah :

```
pinMode(lampu, OUTPUT);
pinMode(IN3, OUTPUT);
pinMode(IN4, OUTPUT);
pinMode(ENB, OUTPUT);
digitalWrite(lampu, HIGH);
digitalWrite(IN3, LOW);
digitalWrite(IN4, LOW);
```

Pengulangan (loop) yang ada pada program Arduino adalah :

```
void loop() {
  while (BT.available()) {
    delay(10);
    char c = BT.read();
    if (c == '#') {
      break;
    }
    voice += c;
  }

  if (voice.length() > 0) {
    Serial.println(voice);

    //-----perintah aktifkan-----//
    if (voice == "nyalakan lampu" ) {
      digitalWrite(lampu, LOW);
    }
    else if (voice == "tutup gorden" ) {
      tutup();
    }

    //-----perintah non-aktifkan-----//
    else if (voice == "Matikan lampu") {
      digitalWrite(lampu, HIGH);
    }
    else if (voice == "buka gorden") {
      buka ();
    }
    voice = "";
  }
}
```

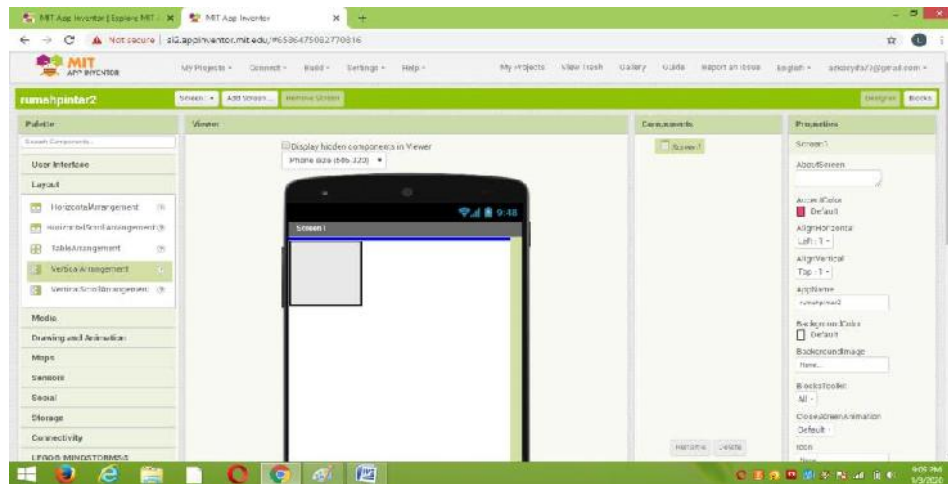
Gambar 3.5 Potongan *Listing* Program Arduino IDE

Keterangan :

Lampu menggunakan pin 8 yang terdapat pada modul *relay*.

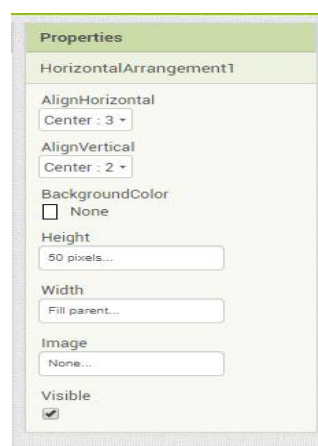
Pin-pin input pada driver motor IC L298N digunakan untuk menginisialisasi motor DC. IN3 dan ENB dihubungkan pada pin 3 dan pin 5 Arduino yang bertindak sebagai PWM. PWM (*Pulse Width Modulation*) memanipulasi keluaran digital sedemikian rupa sehingga menghasilkan sinyal analog. Arduino mengatur *output* digital ke *HIGH* dan *LOW* bergantian dengan porsi waktu tertentu untuk setiap nilai keluarannya. Durasi waktu untuk nilai *HIGH* disebut *pulse width* atau panjang pulsa. Variasi nilai *output* analog didapatkan dari perubahan panjang pulsa yang diberikan pada satu periode waktu dan dilakukan berulang-ulang. Sementara IN4 dihubungkan ke pin 4 pada Arduino. Cara mengganti arah putaran motor ini yaitu dengan cara mengganti arah kutub motor DC. Arah kutub motor dilambangkan dengan IN3 dan IN4. Untuk menggerakkan putaran motor, IN3 diberi logika *HIGH* dan IN4 diberi logika *LOW* ataupun sebaliknya. Jadi, IC L298N digunakan untuk membantu mengganti arah kutub dari motor DC.

6. Setelah tampilan awal terbuka, maka kita akan membuat *layout* untuk aplikasi Android. Kita tinggal melakukan drag and drop untuk *layout* yang kita inginkan.



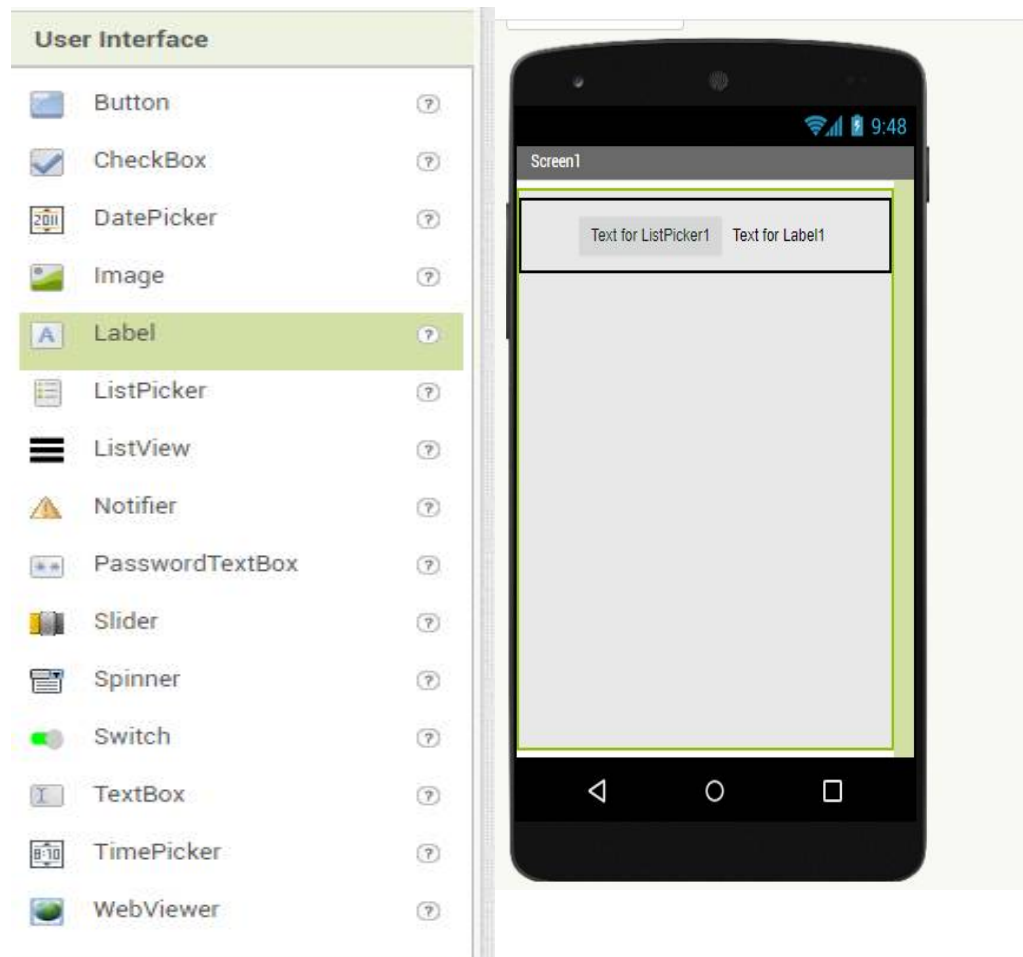
Gambar 3.7 Tampilan Proses Rancangan Layout

7. Tentukan nilai *properties layoutnya* agar tampilan aplikasinya seperti yang kita inginkan.



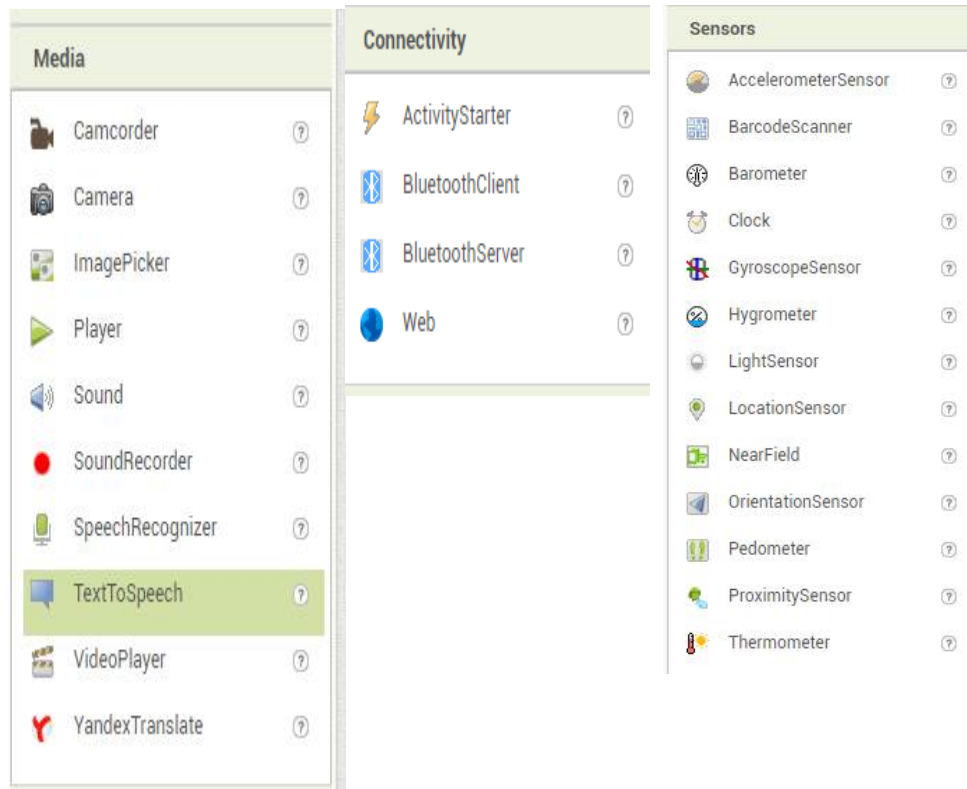
Gambar 3.8 Tampilan Properties

8. Tambahkan *user interface* dan atur nilai *propertiesnya*



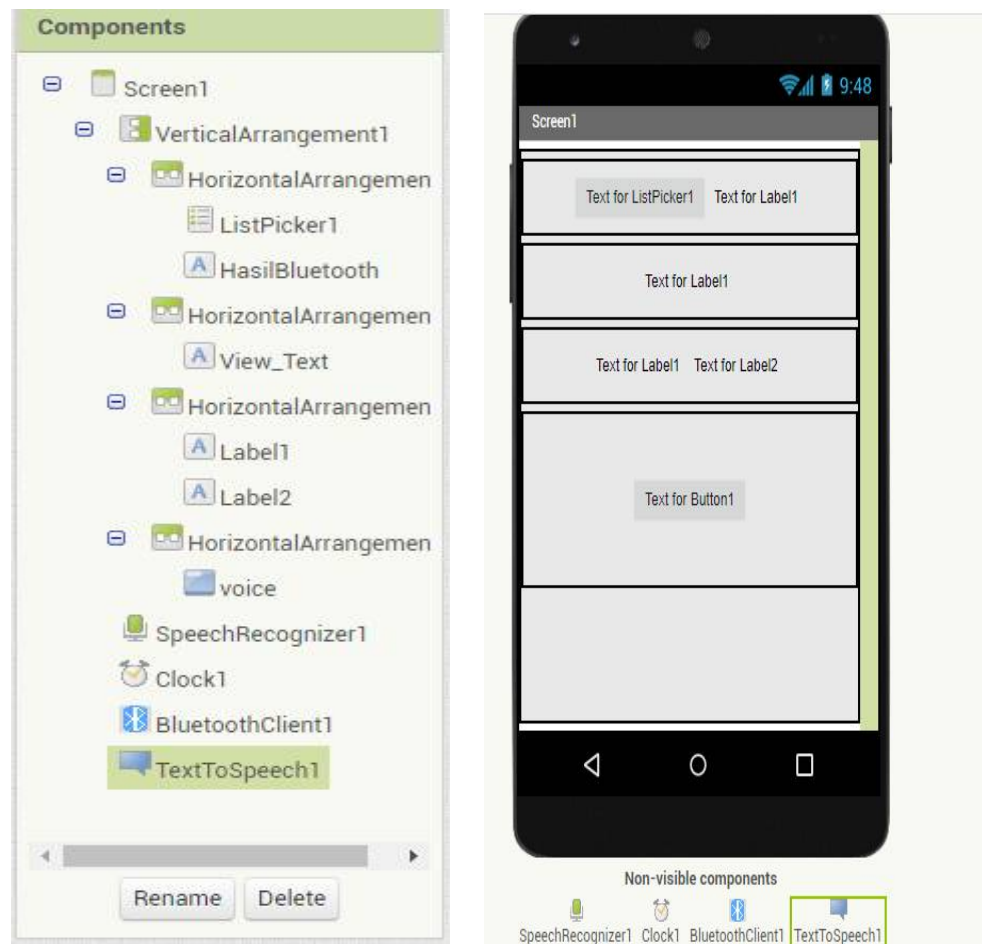
Gambar 3.9 Tampilan User Interface dan Layout

9. Tambahkan media, sensor dan *connectivity* yang diinginkan seperti *text to speech*, *voice recognizer*, *bluetoothclient*, *clock*, dll.



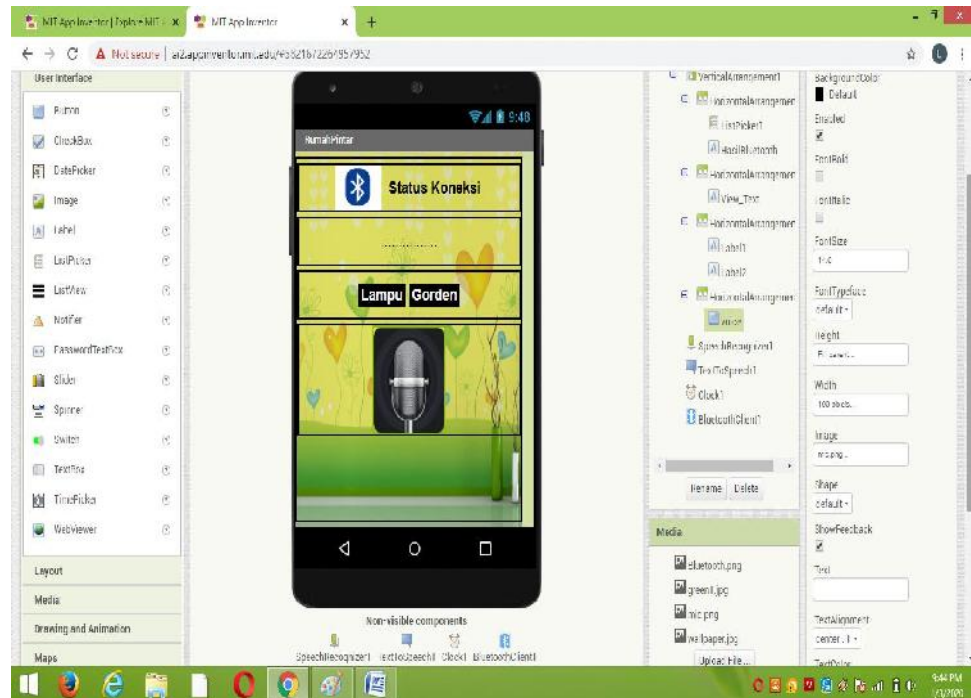
Gambar 3.10 Jenis-Jenis Media Pada *MIT App Inventor 2*

10. Ini adalah beberapa komponen yang dibutuhkan dalam membuat aplikasi rumah pintar.



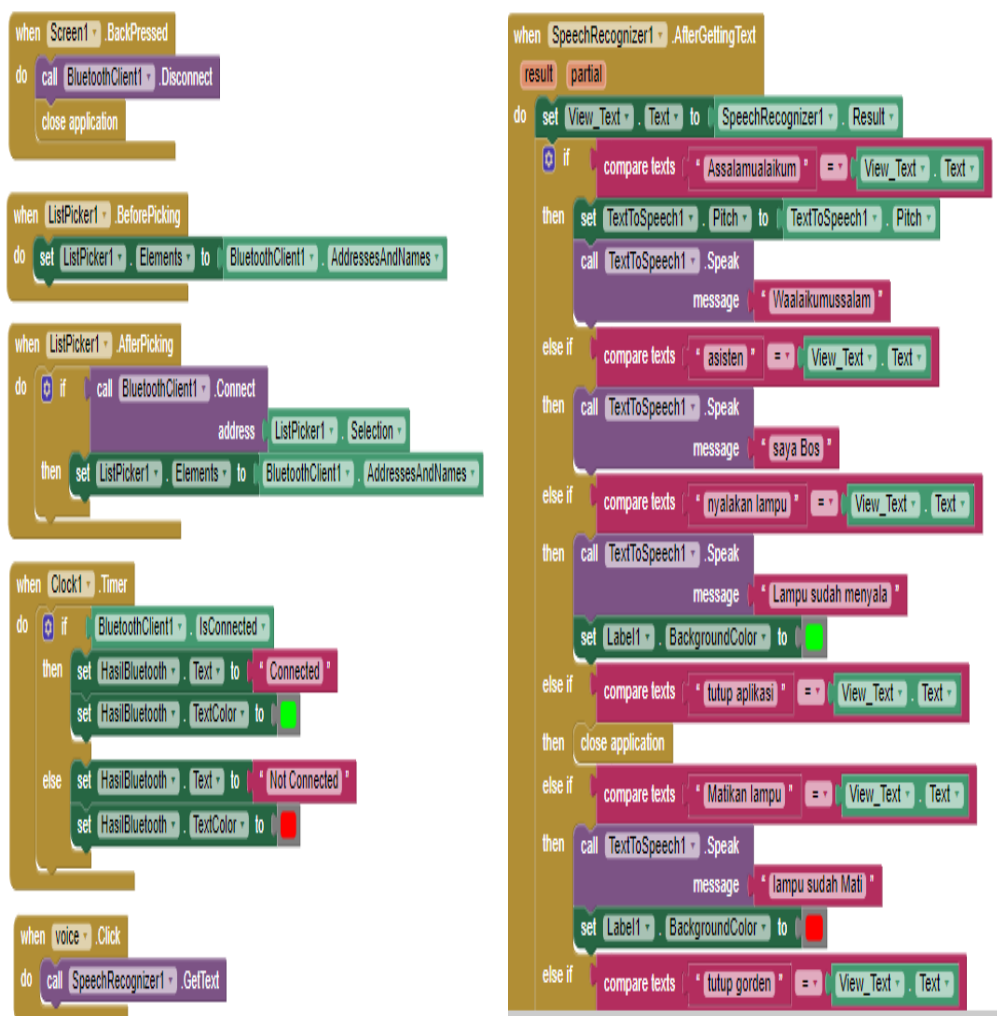
Gambar 3.11 Tampilan Komponen Aplikasi yang Digunakan

11. Setelah semua komponen disusun, kita tinggal memperindah tampilan Android. Ini adalah tampilan aplikasi rumah pintar yang diinginkan.



Gambar 3.12 Tampilan Aplikasi Rumah Pintar

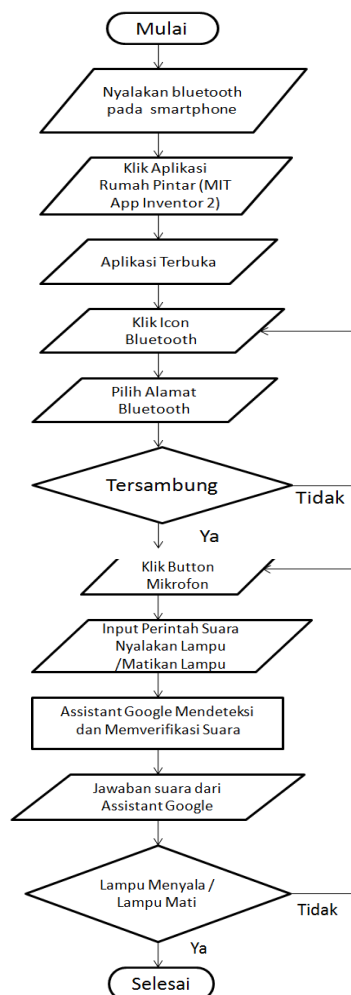
12. Setelah desain tampilan aplikasi sudah dibuat, langkah selanjutnya adalah menyusun blok-blok program. Blok-blok program ini berfungsi sebagai *codingan* di dalam *MIT App Inventor 2*. Kita tinggal menyusun blok-blok program yang berbentuk puzzle ini kedalam blok editor.



Gambar 3.13 Tampilan Blok Editor *MIT App Inventor 2*

2. Flowchart

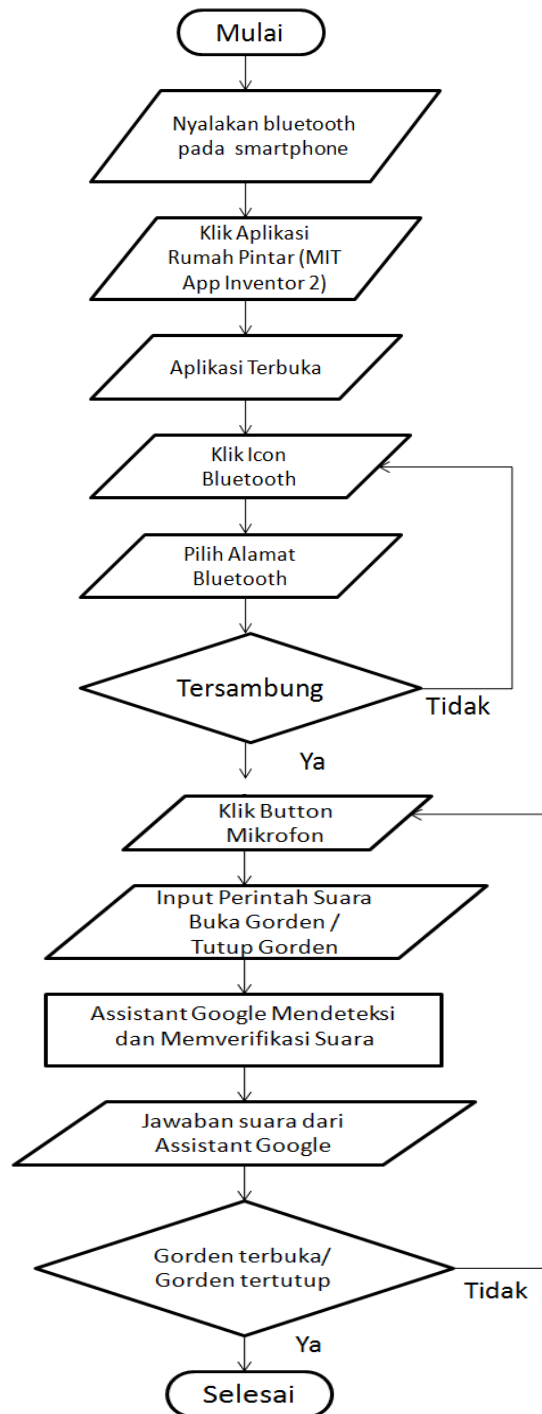
Pada pembuatan proyek tugas akhir ini, dibutuhkan suatu teknik perancangan struktur yang baik, biasanya diawali dengan pembuatan diagram alur (*flowchart*). Diagram alur digunakan untuk menggambarkan terlebih dahulu apa yang harus dikerjakan sebelum mulai merancang atau membuat suatu sistem seperti yang akan dijelaskan dibawah ini. Berikut adalah diagram alur (*flowchart*) dari aplikasi Android yang dibuat.



Gambar 3.14 Flowchart Menyalakan / Mematikan Lampu

Alur algoritma aplikasi untuk menyalakan / mematikan lampu :

1. Mulai
2. Hidupkan *bluetooth* pada *smartphone*
3. Buka aplikasi rumah pintar (*MIT App Inventor 2*)
4. Aplikasi rumah pintar terbuka
5. Klik *icon bluetooth* pada aplikasi
6. Pilih alamat *bluetooth*
7. Jika kedua *bluetooth* terpasang maka akan tersambung, jika tidak tersambung klik *icon bluetooth* kembali.
8. Setelah tersambung, klik *button* mikrofon untuk memberikan perintah suara.
9. *Input* perintah suara “nyalakan lampu” atau “matikan lampu”
10. *Assistant google* akan mendeteksi dan memverifikasi perintah suara yang diucapkan
11. Jika perintah suara yang diucapkan cocok dengan *database* yang dimasukkan, *assistant google* akan menjawab “lampu sudah menyala” atau “lampu sudah mati”. Jika perintah suara tidak cocok maka kembali ke *button* mikrofon.
12. Selesai



Gambar 3.15 Flowchart Buka / Tutup Gorden

Alur algoritma aplikasi untuk membuka dan menutup gorden :

1. Mulai
2. Hidupkan *bluetooth* pada *smartphone*
3. Buka aplikasi rumah pintar (*MIT App Inventor 2*)
4. Aplikasi rumah pintar terbuka
5. Klik *icon bluetooth* pada aplikasi
6. Pilih alamat *bluetooth*
7. Jika kedua *bluetooth* terpasang maka akan tersambung, jika tidak tersambung klik *icon bluetooth* kembali.
8. Setelah tersambung, klik *button* mikrofon untuk memberikan perintah suara.
9. Input perintah suara “buka gorden” atau “tutup gorden”
10. *Assistant google* akan mendeteksi dan memverifikasi perintah suara yang diucapkan
11. Jika perintah suara yang diucapkan cocok dengan *database* yang dimasukkan, *assistant google* akan menjawab “gorden sudah dibuka” atau “gorden sudah ditutup”. Jika perintah suara tidak cocok maka kembali ke *button* mikrofon.
12. Selesai

BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Spesifikasi Minimum *Hardware* dan *Software*

Pada proyek tugas akhir ini membutuhkan beberapa spesifikasi minimum *hardware* (perangkat keras) maupun *software* (perangkat lunak).

1. Spesifikasi *Hardware*

Spesifikasi *hardware* yang digunakan pada perancangan sistem kendali rumah pintar (*smart home*) ini adalah sebagai berikut :

- a. Arduino Uno
- b. *Relay Module*
- c. *Bluetooth Module* HC-05
- d. *Driver Motor* IC L298N
- e. Motor DC sebagai penggerak gorden
- f. 1 buah lampu pijar
- g. Rangkaian miniatur gorden yang dibuat menggunakan *styrofoam*
- h. *Smartphone* Android versi 9.0 (*Pie*) untuk menguji aplikasi rumah pintar

2. Spesifikasi *software*

Spesifikasi *software* yang digunakan pada perancangan sistem kendali rumah pintar (*smart home*) ini adalah sebagai berikut :

- a. Arduino IDE
- b. Aplikasi rumah pintar menggunakan *MIT App Inventor 2*

4.2 Pengujian Aplikasi dan Pembahasan

Pengujian pada aplikasi dilakukan untuk mengetahui pengaruh perintah suara terhadap kinerja alat apakah sudah bekerja atau berjalan sesuai dengan yang diharapkan.

4.2.1 Pengujian Aplikasi

Pada tahap pengujian ini seluruh rangkaian sistem kendali rumah pintar termasuk komponen *hardware* dan *software* dioperasikan. Seluruh komponen ini dirangkai dalam satu kesatuan sehingga dapat diperoleh aplikasi sesuai dengan yang diharapkan.

1. Proses pemasukan data

Pada proses pemasukan data menggunakan suara dan *smartphone* Android sebagai media untuk menangkap suara yang akan diucapkan sebagai perintah. Setelah suara ditangkap oleh *smartphone* Android, suara tadi dikirimkan melalui sinyal *bluetooth* yang sudah dipasangkan pada *bluetooth module* HC-05 rangkaian. Pastikan *bluetooth* pada *smartphone* sudah dihidupkan terlebih dahulu.

2. Proses transaksi data

Mikrokontroler Arduino Uno dapat bekerja dan memproses data yang dikirimkan dari aplikasi rumah pintar jika didalamnya sudah dimasukkan *listing* program. *Listing* program yang sudah dibuat di *upload* kedalam Arduino menggunakan *tools* pemrograman Arduino IDE. Fungsi program disini antara lain yaitu menginisialisasi pin-pin mana saja yang akan menjadi *input* maupun *output*, mengubah data yang dikirimkan melalui Android menjadi perintah logika “*HIGH*” atau “*LOW*” yang akan mengaktifkan atau menon-aktifkan *relay* dan *output-output* lainnya, serta menginisialisasi alamat IP *bluetooth* yang akan menjadi alamat tujuan pengiriman data dari Android.

Setelah *bluetooth* pada *smartphone* dan *bluetooth module* pada rangkaian terpasang atau tersambung, maka aplikasi rumah pintar pada *smartphone* siap digunakan untuk diberikan perintah suara. Setelah sinyal *bluetooth* menerima suara yang diucapkan, maka sinyal *bluetooth* akan diterima oleh Arduino, Arduino akan memproses suara tersebut dan mengirimkannya ke *output* lampu dan motor DC sebagai penggerak gorden.

3. Proses pelaporan data

Proses pelaporan data meliputi hasil pengujian dari berbagai jenis komponen yang digunakan dalam tugas akhir ini. Tujuan dari pelaporan data ini adalah untuk mengetahui apakah setiap bagian-bagian dari perangkat telah bekerja sesuai

dengan fungsinya atau tidak.

a. Pengujian Motor DC

Pengujian motor DC ini untuk mengetahui apakah kondisi perputaran motor sudah berjalan dan bekerja dengan baik atau tidak. Motor DC ini menggunakan IC L298N sebagai *driver*. Pengujian ini dilakukan dengan menghubungkan motor DC dengan *driver* dan memberikan tegangan serta memberi input data dari Arduino Uno. Tabel hasil pengujian dapat dilihat pada tabel berikut ini :

Tabel 4.1 Hasil Pengujian Motor DC

No	Kondisi Input	Kondisi Motor
1	IN3, HIGH	Berputar
	IN4, LOW	Diam
2	IN3, LOW	Diam
	IN4, HIGH	Berputar
3	IN3, LOW	Diam
	IN4, LOW	Diam

b. Pengujian *Bluetooth Module* HC-05

Pada pengujian *bluetooth module* HC-05 ini untuk mengetahui apakah sinyal diantara kedua *bluetooth* masih dapat terhubung atau tidak ketika diberi jarak tertentu, sehingga diharapkan dapat mengetahui jarak maksimal kinerja *bluetooth*. Berikut ini adalah hasil pengujian terhadap *bluetooth module* yang telah dilakukan :

Tabel 4.2 Hasil Pengujian *Module Bluetooth HC-05*

No	Kondisi	Jarak	Hasil Transmisi	
			Terhubung	Tidak Terhubung
1	Tanpa Penghalang	1-10 meter	√	
		11 meter	√	
		12 meter	√	
		13 meter		√
2	Ada Penghalang	1-10 meter	√	
		11 meter		√
		12 meter		√
		13 meter		√

c. Pengujian *Relay* dengan Lampu

Pengujian ini dilakukan untuk mengetahui apakah *relay* dan rangkaian lampu sudah tersambung dengan baik atau tidak. Berikut ini hasil pengujian *relay* ketika dihubungkan dengan lampu.

Tabel 4.3 Hasil Pengujian *Relay* dengan Lampu

No	Sinyal Input	Kondisi Relay	Kondisi Lampu	Tegangan (Volt)	Keterangan
1	<i>Low</i>	Aktif	Hidup	12	Benar
2	<i>High</i>	Aktif	Mati	12	Benar

d. Pengujian Perintah Suara

Tahap ini dilakukan untuk menguji pengaruh perintah suara apakah kata yang diucapkan terhadap rangkaian alat berfungsi dengan baik atau tidak.

Tabel 4.4 Hasil Pengujian Perintah Suara

No	Perintah yang diucapkan		Responden			Hasil Respon Perintah	Waktu Delay Sistem (detik)	Waktu Respon Rata-Rata
			1	2	3			
1	Lampu Ruangan	Nyalakan	√	√	√	Menyala	3	3
		Matikan	√	√	√	Mati	3	
2	Gorden	Buka	√	√	√	Buka	3	
		Tutup	√	√	√	Tutup	3	

4.2.2 Pembahasan

Berdasarkan hasil pengujian dari beberapa rangkaian dan komponen pada proyek tugas akhir ini, maka dapat diambil kesimpulan bahwa seluruh rangkaian dapat bekerja dengan baik dan sesuai dengan fungsi dari masing-masing komponen. Walaupun ada beberapa komponen yang terkadang memiliki waktu *delay* (waktu tunggu) sewaktu diberikan perintah suara, akan tetapi komponen tersebut berfungsi dengan baik.

Pada pengujian pengukuran rangkaian sistem terdapat beberapa perbedaan dengan adanya selisih dari hasil pengukuran dengan apa yang diperoleh dari teori komponen. Perbedaan tersebut dipengaruhi oleh beberapa faktor seperti alat ukur yang digunakan dan kurang telitinya dalam pengukuran.

1. Analisis Pengujian Motor DC

Motor DC yang digunakan untuk menggerakkan gordena ini menggunakan *driver* motor IC L298N. Motor DC yang digunakan dihubungkan dengan tali kemudian dipasangkan sepotong kain pada tali tersebut yang bertindak sebagai gordennya. Gordena akan bergerak ke kiri dan ke kanan (membuka dan menutup) tirai gordena secara otomatis sesuai dengan logika program yang telah dimasukkan kedalam Arduino Uno.

Pada saat motor diberikan input yang salah maka motor akan diam atau tidak bergerak dan tegangan pada motor 0V. Saat motor DC mendapat input yang benar maka motor akan berputar membuka atau menutup gordena. Untuk menggerakkan motor DC ini membutuhkan tegangan dari luar, yakni baterai 9 V. Kecepatan dari motor DC ini ditentukan oleh nilai yang dimasukkan ke pin PWM. Motor DC membutuhkan 2 pin untuk menggerakkan arah putaran motor. Kedua pin tersebut sebagai pengganti kutub dari motor DC. Arah kutub pada motor DC yang bergerak ke kiri dan ke kanan diatur oleh driver motor IC L298N. Cara mengganti arah putaran motor yaitu dengan memberi logika “*HIGH*” pada IN3 dan logika “*LOW*” pada IN4 ataupun sebaliknya.

2. Analisis Pengujian Jangkauan *Bluetooth Module* HC-05

Pengujian jangkauan *bluetooth* dilakukan dengan memberikan dua kondisi dimana kondisi pertama tanpa penghalang di dalam satu ruangan dan kondisi

kedua dengan penghalang. Penghalang yang dimaksudkan adalah ketika pintu ditutup dan kita memberikan perintah suara. Pengujian ini bertujuan untuk mengetahui jangkauan transmisi *bluetooth* dengan jarak tertentu.

Ketika *bluetooth* dioperasikan tanpa penghalang dengan jarak 1 sampai dengan 10 meter maka *bluetooth* masih bisa merespon perintah dari kata yang diucapkan, begitu juga dengan jarak 11-12 meter, *bluetooth* modul masih bisa merespon, ketika dengan jarak 13 meter *bluetooth* sudah terputus atau tidak tersambung lagi. Begitu juga ketika *bluetooth* dioperasikan dengan penghalang dengan jarak 1 sampai dengan 10 meter masih bisa merespon, tetapi ketika diberi jarak 11 hingga 13 meter, *bluetooth* sudah terputus atau tidak tersambung lagi.

3. Analisis Pengujian *Relay* dengan lampu

Pengujian *relay* dengan lampu bertujuan untuk mengetahui kinerja dari lampu dan *relay* sebagai *drivernya*. Lampu digunakan sebagai peralatan rumah tangga yang akan dikendalikan pada *smart home* ini. Lampu dihubungkan dengan *fitting* lampu yang telah tersambung dengan kabel. Kabel yang ada pada *fitting* lampu dikupas menjadi 2 bagian. Kedua bagian timah kabel dimasukkan pada bagian rangkaian *relay*. Lampu mengikuti logika *relay* yang *default* aktifnya adalah *LOW*. Ketika pin pada *relay* diberi logika “*LOW*” maka lampu akan menyala, begitu juga ketika diberi logika “*HIGH*” maka lampu akan mati.

4. Analisis Pengujian Aplikasi Rumah Pintar

Pengujian aplikasi rumah pintar ini dilakukan untuk mengetahui proses identifikasi suara yang ditangkap oleh *smartphone* dan juga untuk mengetahui kinerja dari perintah suara kepada respon alat. Sebagai contoh, kata perintah yang diucapkan oleh responden untuk menghidupkan lampu adalah “nyalakan lampu” maka lampu akan menyala. Begitu pula ketika responden mengucapkan kata perintah “matikan lampu” maka lampu akan mati atau padam.

Selain itu, pengujian juga dilakukan pada rangkaian gorden. Ketika diberi perintah suara “buka gorden” maka gorden akan terbuka, begitu pun ketika diberikan perintah suara “tutup gorden” maka gorden akan tertutup. Respon dari masing-masing alat sudah berjalan cukup baik dan bekerja sesuai dengan yang diharapkan.

5. Analisis Pengujian Keseluruhan Alat

Dari hasil pengujian keseluruhan alat didapatkan bahwa aplikasi dan rangkaian alat yang meliputi lampu dan rangkaian gorden sudah bekerja dan berjalan seperti yang diharapkan walaupun terdapat waktu tunggu (*delay*) dari aplikasi ke rangkaian alat rata-rata respon waktunya 3 detik. Hal tersebut mungkin dikarenakan kondisi ruangan yang *noise*, atau bisa juga disebabkan oleh jaringan pada operator seluler yang digunakan atau juga kesalahan dari pengguna yang memberi masukan perintah suara, ataupun kesalahan dalam mengoperasikan

rangkaian alat. Hasil pengujian keseluruhan alat dapat dilihat pada tabel berikut

ini :

Tabel 4.5 Hasil Pengujian Keseluruhan Alat

No	Kata Perintah	Tampilan Aplikasi	Tampilan Alat dan Respon Perintah	Keterangan
1.	Nyalakan lampu		 Lampu sudah menyala	Sesuai
2.	Matikan lampu		 Lampu sudah mati	Sesuai
3.	Buka gorden		 Gorden sudah dibuka	Sesuai
4.	Tutup gorden		 Gorden sudah ditutup	Sesuai

BAB V

PENUTUP

5.1 Simpulan

Dari hasil pengujian yang telah dilakukan terhadap sistem kendali rumah pintar (*smart home*) dengan perintah suara, maka simpulan yang dapat diperoleh antara lain :

1. Pembuatan *hardware* rangkaian gorden menggunakan *styrofoam* yang sudah dipasangkan dengan motor DC yang berada di sisi kanan dan sisi kiri bagian atas. Pada kedua bagian motor DC dipasangkan *gear* sebagai tempat rel atau tali, yang mana rel atau tali tersebut digantungkan sepotong kain yang berperan sebagai tirai / gorden. Motor DC tersebut sebagai penggerak dari tali yang akan membuka dan menutup gorden dari kanan ke kiri atau sebaliknya sesuai dengan perintah yang diberikan. Sedangkan lampu menggunakan *relay* sebagai saklar yang terhubung dengan Arduino.
2. Pembuatan aplikasi rumah pintar (*smart home*) menggunakan *MIT App Inventor 2* untuk merancang dan memprogram aplikasi secara *online*. Program yang digunakan untuk menjalankan mikrokontroler Arduino Uno menggunakan Arduino IDE sebagai pembuatan *codingan* program yang menggunakan bahasa pemrograman C.

3. Hasil kerja dari keseluruhan sistem kendali rumah pintar (*smart home*) dengan perintah suara bekerja sesuai dengan fungsinya seperti yang diharapkan.

5.2 Saran

Penulis menyadari masih banyak terdapat kekurangan dan kelemahan dalam pembuatan alat ini dikarenakan keterbatasan waktu, kemampuan dan dana, maka penulis menyarankan sebagai berikut :

1. Miniatur gorden seharusnya dipasangkan pada tempat yang kokoh seperti menggunakan acrylic sehingga ketika gorden diberikan perintah suara dapat berjalan lancar tanpa adanya halangan.
2. Menggunakan komponen yang lebih baik lagi agar rel maupun gear bisa berjalan optimal seperti menggunakan motor stepper dan tali binaan dari motor itu sendiri.
3. Untuk menjalankan aplikasi rumah pintar seharusnya tidak dalam kondisi yang berisik agar *google assistant* dapat mendengar perintah dengan baik untuk mengurangi terjadinya *delay* atau *noise* ketika diberikan perintah suara.

DAFTAR PUSTAKA

- Abdul Kadir. (2016). Simulasi Arduino. Jakarta : PT. Elex Media Komputindo
- Abdul Kadir. (2017). Pemrograman Arduino Menggunakan ArduBlock. Jakarta :PT. Elex Media Komputindo
- Agung Rismawan. (2015). Konsep Sistem Kendali, Sistem Kendali Terbuka & Tertutup dan Contoh Aplikasinya. Diakses dari <https://serbatelekomunikasi.wordpress.com/2015/02/12/8/>
- Agus Faudin. (2017). Tutorial Arduino mengakses module Bluetooth HC-05. Diakses dari <https://www.nyebarilmu.com/tutorial-arduino-module-bluetooth-hc-05/>
- Agus Faudin. (2017). Tutorial Arduino mengakses driver motor L298N. Diakses dari <https://www.nyebarilmu.com/tutorial-arduino-mengakses-driver-motor-l298n/>
- Andi Syofian. (2016). Pengendalian Pintu Pagar Geser Menggunakan Aplikasi Smartphone Android dan Mikrokontroler Arduino Melalui Bluetooth. Jurnal Teknik Elektro ITP. 5 (01). 2252-3472. Diakses dari <https://ejournal.itp.ac.id/index.php/telektro/article/viewFile/347/330>
- Azmi, Fadhillah, and Winda Erika. "Analisis keamanan data pada block cipher algoritma kriptografi rsa." ccess (Journal of Computer Engineering, System and Science) 2.1: 27-29
- Dickson Kho. (2017). Diakses dari <https://teknikelektronika.com/pengertian-relay-fungsi-relay/>
- Erika, winda. "analisis perbandingan metode tam (technology acceptance model) dan utaut (unified of acceptance and use of technology) terhadap persepsi pengguna sistem informasi digital library (studi kasus: universitas pembangunan panca budi medan)." Jurnal Mahajana Informasi 4.1 (2019): 78-83.
- Farisqi Panduardi, Endi Sailul Haq. (2016). Wireless Smart Home System Menggunakan Raspberry PI Berbasis Android. Jurnal Teknologi Informasi dan Terapan.03(01). 2354-838X. Diakses dari <https://pdfs.semanticscholar.org/402a/ce8d6629211519bc524830408a5c9c825574.pdf>
- Hamdi, Nurul. "Model Penyiraman Otomatis pada Tanaman Cabe Rawit Berbasis Programmable Logic Control." Jurnal Ilmiah Core IT: Community Research Information Technology 7.2 (2019).

- Hendrawan, J., & Perwitasari, I. D. (2019). Aplikasi Pengenalan Pahlawan Nasional dan Pahlawan Revolusi Berbasis Android. *JurTI (Jurnal Teknologi Informasi)*, 3(1), 34-40.
- Hasibuan, Alfiansyah. "Analisis Penggunaan Metode Algoritma Kohonen pada Jaringan Syaraf Tiruan Learning Vector Quantization (LVQ) pada Pengenalan Pola." (2019).
- Herdianto, H., & Anggraini, S. (2019, May). Perancangan sistem pendeteksi uang palsu untuk tuna netra menggunakan arduino uno. in seminar nasional teknik (semnastek) uisu (Vol. 2, No. 1, pp. 136-140).
- Harumy, T. H. F., & Sulistianingsih, I. (2016). Sistem penunjang keputusan penentuan jabatan manager menggunakan metode mfep pada cv. Sapo durin. In *Seminar Nasional Teknologi Informasi dan Multimedia* (pp. 6-7).
- Khairul, K., Haryati, S., & Yusman, Y. (2018). Aplikasi Kamus Bahasa Jawa Indonesia dengan Algoritma Raita Berbasis Android. *Jurnal Teknologi Informasi dan Pendidikan*, 11(1), 1-6.
- Muttaqin, Muhammad. "Analisa pemanfaatan sistem informasi e-office pada universitas pembangunan panca budi medan dengan menggunakan metode utaut." *Jurnal Teknik dan Informatika 5.1* (2018): 40-43.
- Nasution, M. Z. (2019). Penerapan principal component analysis (pca) dalam penentuan faktor dominan yang mempengaruhi pengidap kanker serviks (Studi Kasus: Cervical Cancer Dataset). *Jurnal Mantik*, 3(1), 204-210.
- Perwitasari, I. D. (2018). Teknik Marker Based Tracking Augmented Reality untuk Visualisasi Anatomi Organ Tubuh Manusia Berbasis Android. *Intecoms: Journal of Information Technology and Computer Science*, 1(1), 8-18.
- Rizal, Chairul. "Sistem pendukung keputusan penentuan guru dan pegawai terbaik menggunakan metode saw (simple additive weighting) studi kasus smas islam alulum terpadu medan." *Jurnal Teknik dan Informatika 6.2* (2019): 14-17.
- Ramadhani, S., Suherman, S., Melvasari, M., & Herdianto, H. (2018). Perancangan Teks Berjalan Online Sebagai Media Informasi Nelayan. *Jurnal Ilmiah Core IT: Community Research Information Technology*, 6(2).
- Saputra, Muhammad Juanda, and Nurul Hamdi. "Rancang bangun aplikasi sejarah kebudayaan aceh berbasis android STUDI kasus dinas kebudayaan dan pariwisata aceh." *journal of informatics and computer science 5.2* (2019): 147-157.

Sulistianingsih, I. (2019). Sistem Pendukung Keputusan Penentuan Menu Makanan Sehat untuk Pasien Rawat Inap. *Jurnal Teknik dan Informatika*, 6(1), 6-11.

<https://arifashkaf.wordpress.com/2015/10/14/pengertian-sistem-dan-contohnya-softskill/>

https://id.wikipedia.org/wiki/App_Inventor

[https://id.wikipedia.org/wiki/Android_\(sistem_operasi\)](https://id.wikipedia.org/wiki/Android_(sistem_operasi))

) https://id.wikipedia.org/wiki/Sistem_kendali

<https://id.wikipedia.org/wiki/Sistem>

https://id.wikipedia.org/wiki/Lampu_pijar

Zona Elektro. (2013). Diakses dari <http://zoniaelektro.net/motor-dc/>