

# KENDALI PAGAR RUMAH OTOMATIS BERBASIS ANDROID

Kent Bryan Gozali

Pembimbing I : Kusno Harianto, S.Kom., M.Kom Pembimbing II : Andi Yusika Rangan, S.Kom., M.Kom

Teknik Informatika, STMIK Widya Cipta Dharma  
Jl. M. Yamin No.25, Samarinda, 75123  
E-mail : Kent.bryangozali@gmail.com

## ABSTRAK

Bagaimana cara merancang dan membangun desain alat kendali pagar rumah otomatis berbasis android menggunakan perintah dengan aplikasi sederhana pada *smartphone* untuk membuka dan menutup pagar secara otomatis dengan jangkauan yang cukup luas agar dapat memudahkan dan membantu pemilik rumah. Penelitian dilaksanakan di STMIK Widya Cipta Dharma yang beralamatkan di Jl. Muhammad Yamin No. 25, Samarinda, Kalimantan Timur, waktu penelitian kurang lebih selama 1 bulan, pengumpulan data pada penelitian ini menggunakan pengamatan lapangan, studi dokumentasi, dan studi pustaka. Untuk pengembangan sistem menggunakan metode *prototype* dan pengujian yang digunakan adalah *black box testing* dan *white box testing*. Hasil dari pembuatan sistem kendali pagar rumah otomatis berbasis android, *prototype* ini memiliki kelebihan dan kekurangan, kelebihan alat ini dapat di operasikan menggunakan 2 koneksi yaitu koneksi Bluetooth dan koneksi jaringan WiFi. Alat ini juga mempunyai sensor yang dapat mendeteksi jika ada objek yang berada di depan pagar. Kekurangan Alat ini memerlukan 2 aplikasi yang berbeda untuk mengendalikan pagar berdasarkan dengan koneksi yang digunakan. Alat ini belum mempunyai sistem keamanan yang cukup. Jarak kendali pagar lebih jauh menggunakan koneksi jaringan WiFi dibandingkan dengan menggunakan koneksi Bluetooth. Penggunaan koneksi Bluetooth lebih responsif dibandingkan dengan koneksi jaringan WiFi yang memiliki sedikit *delay*. Pengujian alat menggunakan *black box* dan *white box*, tujuan dari pengujian ini yaitu untuk memastikan semua kesalahan masukan yang dilakukan oleh pengguna dapat ditangani oleh sistem.

**Kata Kunci :** Arduino, NodeMCU ESP8266 V3, Android, WiFi, Bluetooth

## 1. PENDAHULUAN

Kemajuan teknologi di bidang IT akhir-akhir ini berkembang dengan pesat dan berdampak banyak terhadap kehidupan manusia. Hal ini membuat beban pekerjaan sehari-hari menjadi lebih ringan dan efisien. Salah satu contohnya adalah kendali pagar rumah otomatis menggunakan Android. Terkadang manusia lupa untuk menutup pagar rumah saat terburu-buru dan ketika sadar bahwa gerbang belum ditutup, tidak sedikit orang yang malas serta enggan untuk menutupnya. Oleh karena itu, dibutuhkan alat pengendali pagar rumah otomatis yang mampu dikendalikan dalam jarak yang cukup jauh. Sejauh ini perkembangan pengendali pagar rumah biasanya di kendalikan dengan *remote control* atau *software* yang hanya dapat dijangkau dengan jarak tertentu. Hal itu masih menjadi kendala ketika pemilik rumah lupa menutup pagar pada jarak yang tidak terjangkau oleh *remote control* dan *software*.

Peneliti berusaha untuk mengatasi masalah di atas menggunakan pengendali yang tidak memiliki batas jarak. Sistem kerja aplikasi android memberi perintah terhadap perangkat NodeMCU untuk memerintahkan *Driver Motor Stepper* memberi gerak terhadap motor yang akan membuka dan menutup gerbang. Motor yang digunakan adalah motor *stepper*. Karena sudut rotasi motor proporsional dengan pulsa masukan sehingga lebih mudah diatur dan juga dapat langsung memberikan torsi penuh pada saat mulai bergerak. Pernyataan ini menjadi salah satu alasan peneliti menggunakan motor *stepper*. Menggunakan motor *stepper* sebagai penggerak pagar rumah, serta dikendalikan dengan NodeMCU ESP8266 V3 yang berupa aplikasi android sehingga dapat memberi perintah dalam jangkauan yang luas.

## 2. RUANG LINGKUP PENELITIAN

### 1. Cakupan permasalahan

Berdasarkan latar belakang yang diuraikan, maka dirumuskan permasalahan yaitu : “Bagaimana Cara Merancang Alat Kendali Pagar Otomatis Berbasis Andorid?”.

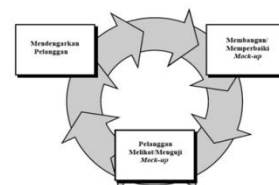
### 2. Batasan-batasan penelitian

Berdasarkan uraian diatas dan mengindari luasnya pokok permasalahan yang ada, maka diberikan batasan masalah sebagai berikut :

1. Menerapkan penggunaan aplikasi sederhana membuka dan menutup pagar di android sebagai pemberi perintah.
2. Membuat pagar rumah dengan pagar terbuka secara horizontal dan kendali pagar rumah otomatis menggunakan android yang berfungsi untuk menutup dan membuka secara otomatis dengan jangkauan yang cukup luas menggunakan koneksi jaringan WiFi dengan jarak 33m dan Bluetooth dengan jarak 42m.
3. Pengujian menggunakan *black box* dan *white box*.

## 3. BAHAN DAN METODE

Adapun bahan dan metode yang digunakan dalam membangun alat ini yaitu metode *Prototype*



Gambar 3.1 *Prototype*

### 3.1 Mendengarkan Pelanggan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

### 3.2 Membangun atau Memperbaiki *Mock-Up*

Membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*). Dalam tahap ini juga *prototyping* yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

### 3.3 Pelanggan Melihat dan Menguji *Mock-Up*

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan *White Box*, *Black Box*, *Basis Path*, pengujian arsitektur dan lain-lain. Lalu perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

### 3.4 Pengujian *Black Box*

*Black box* menyinggung uji coba yang dilakukan pada perangkat. Pengujian *black box testing* berfokus pada persyaratan fungsional perangkat keras. Pengujian ini memungkinkan analisis perangkat memperoleh kumpulan data *input* yang akan mengerjakan seluruh keperluan fungsional program. Untuk mempermudah dalam proses pengujian maka perlu dibuatnya suatu tabel pengujian yang digunakan sebagai tolak ukur atau acuan dalam pengujian perangkat keras tersebut.

Pada pengujian *black box testing* ini tahap pengujian yang dilakukan terhadap pengguna, dimana kesimpulan dari pengujian ini didapat dengan cara pengujian langsung yang dilakukan oleh peneliti program tersebut.

### 3.5 Pengujian *White Box*

Pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara *procedural* untuk membagi pengujian ke dalam beberapa kasus pengujian. Pengujian dilakukan berdasarkan bagaimana suatu *software* menghasilkan *output* dari *input*. Pengujian ini dilakukan berdasarkan kode program.

## 4. RANCANGAN SISTEM

Berikut ini adalah rancangan yang digunakan dalam membangun alat Kendali Pagar Rumah Otomatis Berbasis Android :

### 4.1 Analisis Data

Alat ini pada dasarnya mulai bekerja saat mengklik tombol buka atau tutup pada aplikasi sederhana android di *smartphone* lalu dari mikrokontroler mengirimkan sinyal ke *driver* ULN dan ke I2C LCD dengan koneksi bluetooth atau koneksi jaringan WiFi yang telah tertanam di dalam papan NodeMCU, dari *driver* ULN diteruskan ke motor *stepper* untuk menggerakkan pagar dan dari I2C LCD untuk menampilkan bahwa alat telah berkerja membuka ataupun menutup pagar. Sensor pir berfungsi untuk mendeteksi jika

ada tamu yang berdiri dalam jarak radius sensor pir di depan pagar, dan akan langsung mengirimkan sinyal ke aplikasi sederhana android.

### 4.2 Analisis Kebutuhan

Pada penelitian ini terdapat 2 analisis teknologi, yaitu :

#### 1) Perangkat Lunak

Dalam pembuatan kendali pagar rumah otomatis menggunakan Arduino Uno dan NodeMCU, maka dibutuhkan perangkat lunak sebagai berikut :

1. Windows 11 Home 64-bit, digunakan untuk menjalankan operasi komputer, sehingga dapat berinteraksi dengan mikrokontroler.
2. Program Arduino IDE, digunakan untuk membuat, mengubah dan *upload* kode program yang telah ditulis dengan bahasa pemrograman C.
3. Program App Inventor untuk membuat aplikasi android sederhana sebagai alat kendali pagar.
4. Program *fritzing*, digunakan untuk membuat rangkaian skematik.

#### 2) Perangkat Keras

Perangkat keras yang digunakan dalam perancangan kendali pagar otomatis ini, yaitu :

1. *Personal Computer* (PC) atau laptop yang digunakan untuk memprogram.
2. Arduino Uno dan NodeMCU, sebagai sistem minimum untuk alat kendali.
3. Sensor PIR, alat mendeteksi ini digunakan untuk mendeteksi pergerakan dari suatu objek yang bergerak mendekati pagar.
4. Buzzer, alat ini untuk memberikan sebuah suara peringatan ketika sensor pir mendeteksi.
5. I2C LCD, LCD berfungsi untuk memberi informasi jika pagar terbuka atau tertutup.
6. Driver ULN2003, sebagai pemberi perintah kepada motor *stepper* untuk menggerakkan pagar.
7. Motor *Stepper*, berfungsi untuk membuka dan menutup pagar.
8. Bluetooth HC-05, berfungsi untuk menghubungkan *smartphone* dan alat melalui koneksi Bluetooth.
9. *Smartphone*, sebagai alat untuk memberikan perintah kepada alat melalui aplikasi sederhana.  
Kotak untuk tempat rangkaian kontrol.

### 4.3 Analisis Sistem

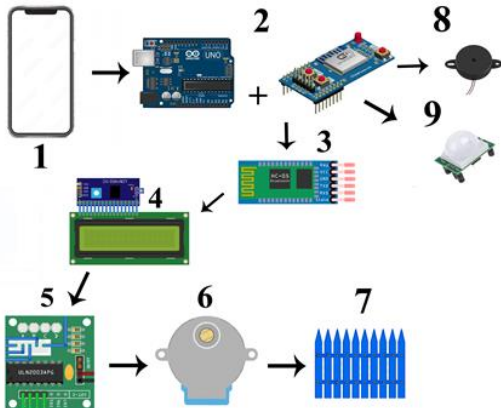
Pada penelitian ini terdapat beberapa analisis sistem yaitu :

1. Sensor PIR akan mendeteksi pergerakan melalui pancaran energi radiasi dari objek yaitu perubahan suhu.
2. Buzzer akan berbunyi ketika sensor pir mendeteksi pergerakan.
3. Mikrokontroler akan memproses ketika pada aplikasi android menekan tombol buka atau tutup.
4. I2C LCD akan memberikan informasi ketika alat sedang berkerja.
5. Driver ULN akan memberikan perintah ke motor *stepper* untuk berkerja ketika mendapatkan perintah.
6. Motor *Stepper* akan membuka atau menutup pagar ketika mendapatkan perintah.

7. Bluetooth HC-05 akan menerima perintah dari *smartphone* untuk mengoperasikan pagar.

#### 4.4 Desain

Pada tahapan desain, digambarkan gambaran umum pagar rumah otomatis, kerangka kerja untuk mempermudah pembuatan alat. Alat yang digunakan adalah 220V AC, Adaptor 5V DC, Arduino Uno, NodeMCU, Sensor PIR, Buzzer, I2C LCD 1602, *Driver* ULN2003, Bluetooth HC-05, Motor *Stepper* berikut gambaran umum pagar rumah otomatis berbasis android.



Gambar 4.1 Rangkaian Kendali Pagar Rumah Otomatis Berbasis Android

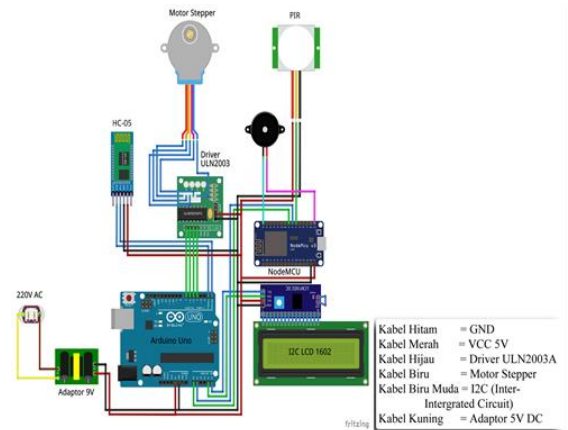
Pada gambar di atas menjelaskan gambaran umum pagar rumah otomatis berbasis android yaitu :

1. Arduino Uno dan NodeMCU digunakan sebagai kontroler dari semua komponen.
2. Sensor pir sebagai indra paling peka di antara semua komponen sebagai sensor.
3. Buzzer sebagai pemberi informasi suara.
4. I2C LCD sebagai pemberi informasi ketika alat sedang berkerja untuk membuka dan menutup pagar.
5. *Smartphone* digunakan sebagai perantara antara jaringan WiFi dan Bluetooth kepada pagar rumah otomatis ini untuk memberikan perintah.
6. *Driver* ULN2003 sebagai alat untuk memberikan aliran listrik kepada motor *stepper* untuk menggerakkan pagar.
7. Motor *Stepper* digunakan sebagai penggerak dari pagar rumah.
8. Bluetooth HC-05 digunakan sebagai penerima perintah dari *smartphone* untuk mengoperasikan alat tersebut.

#### 4.5 Wiring Fritzing

Pada gambar di bawah, Arduino Uno dan NodeMCU berkomunikasi dengan pemberi perintah yaitu *smartphone* melalui Bluetooth HC-05 atau koneksi WiFi yang telah tertanam di dalam papan NodeMCU sendiri. Lalu dari Bluetooth HC-05 melalui kabel *ground*, papan Arduino Uno dan NodeMCU mengirimkan perintah yang telah di berikan oleh *smartphone* melalui kabel *ground* ke I2C LCD dan ke *Driver* ULN , lalu *Driver* ULN meneruskan perintahnya ke Motor *Stepper*. Sensor pir akan berkerja secara otomatis jika mendeteksi suatu pergerakan yang melewati radius sensor tersebut dan buzzer akan berbunyi, lalu mengirimkan ke papan Arduino Uno dan NodeMCU

melalui kabel *ground* kemudian diteruskan kembali sampai ke *smartphone* dengan koneksi Bluetooth atau pun koneksi jaringan WiFi.

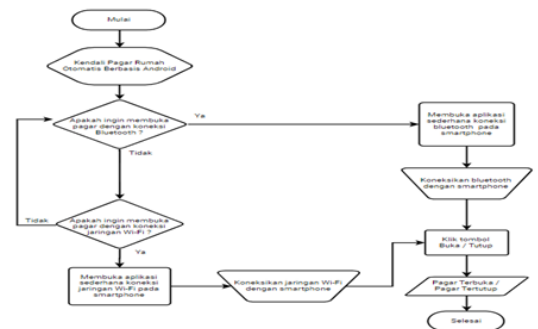


Gambar 4.2 Wiring Fritzing Dari Kendali Pagar Rumah Otomatis Berbasis Android

Komponen yang digunakan dalam pembuatan alat ini adalah komponen yang sudah disusun sedemikian rupa untuk menunjang pembuatan alat ini. Komponen pada alat ini berbentuk *module*.

#### 4.6 Flowchart Program

Pada bagian ini akan dibahas bagaimana algoritma kerja sistem pada alat yang akan di buat dengan menggunakan *flowchart*. Analisa ini bertujuan untuk membuat sketsa atau pengaturan cara kerja alat yang terpisah antara komponen satu dengan komponen yang lainnya. Berikut ini adalah gambaran alur algoritma dari kendali pagar rumah otomatis berbasis android menggunakan arduino uno dan nodeMCU.



Gambar 4.3 Flowchart Kendali Pagar Rumah Otomatis Berbasis Android

Pada simbol "Mulai" menandakan alat dalam keadaan menyala. Kemudian simbol "Inisialisasi" menandakan proses inisialisasi komponen yang terdapat pada kendali pagar rumah otomatis berbasis android, yaitu Arduino Uno, NodeMCU, Bluetooth HC-05, I2C LCD, *Driver* ULN2003, Motor *Stepper*, Sensor Pir, dan Buzzer. Ketika ingin membuka atau menutup pagar, pertama-tama yang dilakukan adalah membuka aplikasi pada *smartphone*. Lalu pilih salah satu koneksi untuk menghubungkan *smartphone* dengan alat kendali pagar otomatis yaitu dengan menggunakan koneksi bluetooth atau dengan

koneksi jaringan WiFi. setelah perangkat *smartphone* terhubung dengan alat kendali pagar otomatis, buka aplikasi pada *smartphone* dan klik tombol buka atau klik tombol tutup. Dengan begitu alat akan mengeluarkan sebuah *output* yaitu pagar terbuka atau pagar tertutup. Kemudian untuk sensor pir bisa mendeteksi apakah ada pergerakan di depan pagar jika ada seseorang yang berada di dalam jangkauan radius sensor pir, akan muncul sebuah notifikasi di dalam aplikasi sederhana bahwa sensor pir sedang mendeteksi sebuah pergerakan dan Buzzer akan berbunyi.

#### 4.7 Black Box Testing

##### 4.7.1 Black Box Tesing Pada Aplikasi Dengan Koneksi Jaringan WiFi

Tabel 4.1 Tabel Hasil Pengujian *Black Box Testing* Pada Aplikasi Dengan Koneksi Jaringan WiFi

N o.	Skenario Pengujian	Hasil Yang Di Harapkan	Hasil Pengujian	Jumlah Percobaan	
				✓	✗
1.	Terkoneksi dengan <i>hotspot</i> jaringan <i>smartphone</i>	Berhasil terkoneksi dengan jaringan WiFi	Berhasil	5x	3x
2.	Mengklik tombol buka pada aplikasi	Berhasil membuka pagar	Berhasil	5x	3x
3.	Mengklik tombol tutup pada aplikasi	Berhasil menutup pagar	Berhasil	5x	3x
4.	Menampilkan status pagar <i>Open</i> saat membuka pagar	Berhasil menampilkan status <i>Open</i>	Berhasil	5x	3x
5.	Menampilkan status pagar <i>Close</i> saat menutup pagar	Berhasil menampilkan status <i>Close</i>	Berhasil	5x	3x
6.	Sensor pir mendeteksi saat ada suatu objek berada di jangkauan sensor	Berhasil mendeteksi suatu objek	Berhasil	5x	3x

Tabel 4.2 Hasil Pengujian Jarak Kendali Pagar Menggunakan Koneksi Jaringan WiFi

Jarak Deteksi	Posisi Jangkauan Deteksi			Keterangan
	Lurus	Kanan (90°)	Kiri (90°)	
1 m	✓	✓	✓	Mendeteksi

5 m	✓	✓	✓	Mendeteksi
10 m	✓	✓	✓	Mendeteksi
15 m	✓	✓	✓	Mendeteksi
20 m	✓	✓	✓	Mendeteksi
25 m	✓	✓	✓	Mendeteksi
30 m	✓	✓	✓	Mendeteksi
31 m	✓	✓	✓	Mendeteksi
32 m	✓	✓	✓	Mendeteksi
33 m	-	-	-	Tidak Mendeteksi
34 m	-	-	-	Tidak Mendeteksi

##### 4.7.2 Black Box Testing Pada Aplikasi Dengan Koneksi Bluetooth

Tabel 4.3 Hasil Pengujian *Black Box Testing* Pada Aplikasi Dengan Koneksi Bluetooth

N o.	Skenario Pengujian	Hasil Yang Di Harapkan	Hasil Pengujian	Jumlah Percobaan	
				✓	✗
1.	Terkoneksi dengan Bluetooth	Berhasil terkoneksi dengan Bluetooth	Berhasil	5x	3x
2.	Mengklik tombol buka pada aplikasi	Berhasil membuka pagar	Berhasil	5x	3x
3.	Mengklik tombol tutup pada aplikasi	Berhasil menutup pagar	Berhasil	5x	3x
4.	Menampilkan status pagar <i>Open</i> saat membuka pagar	Berhasil menampilkan status <i>Open</i>	Berhasil	5x	3x
5.	Menampilkan status pagar <i>Close</i> saat menutup pagar	Berhasil menampilkan status <i>Close</i>	Berhasil	5x	3x

Tabel 4.4 Hasil Pengujian Jarak Kendali Pagar Menggunakan Koneksi Jaringan Bluetooth

Jarak Deteksi	Posisi Jangkauan Deteksi			Keterangan
	Lurus	Kanan (90°)	Kiri (90°)	
1 m	✓	✓	✓	Mendeteksi
5 m	✓	✓	✓	Mendeteksi
10 m	✓	✓	✓	Mendeteksi
15 m	✓	✓	✓	Mendeteksi
20 m	✓	✓	✓	Mendeteksi
25 m	✓	✓	✓	Mendeteksi
30 m	✓	✓	✓	Mendeteksi

Tabel 4.4 Lanjutan Hasil Pengujian Jarak Kendali Pagar Menggunakan Koneksi Jaringan Bluetooth

Jarak Deteksi	Posisi Jangkauan Deteksi			Keterangan
	Lurus	Kanan (90°)	Kiri (90°)	
40 m	√	√	√	Mendeteksi
41 m	√	√	√	Mendeteksi
42 m	√	√	√	Mendeteksi
43 m	-	-	-	Tidak Mendeteksi
44 m	-	-	-	Tidak Mendeteksi
45 m	-	-	-	Tidak Mendeteksi

#### 4.7.3 Black Box Testing Pada Sensor Pir

Tabel 4.5 Hasil Pengujian Respon Sensor Pir

Objek	Status Sensor	Keterangan	Sensitivitas Sensor	Jumlah penelitian
Manusia	Aktif	Mendeteksi	100%	3
Benda Mati	Tidak Aktif	Tidak Mendeteksi	0%	3

#### 4.8 White Box Testing

Untuk menguji *white box* di penelitian ini, maka dibutuhkan *flowgraph*. *White box* menguji tiap fungsi pernyataan yang ada di alat pada penelitian ini. Diantaranya adalah *basis path*. Pengujian ini diuji dengan menggunakan perhitungan *Cyclomatic Complexity* yang memiliki rumus :

$$V(G) = E - N + 2P$$

Keterangan :

E = Jumlah *edges* (anak panah pada *flowgraph*)

N = Jumlah *nodes* (lingkaran pada *flowgraph*)

P = Jumlah *predicate node* (koneksi antar *nodes*)

Tabel 4.6 Nilai Optimal *Cyclomatic Complexity* (CC)

CC	Kategori Modul	Resiko
1-10	Sederhana	Kecil
11-20	Kompleks	Sedang
21-50	Sangat Kompleks	Tinggi
>50	Tidak Stabil	Sangat Tinggi

Setelah menghitung nilai CC yang didapat, maka nilai tersebut dapat ditinjau berdasarkan tabel diatas. Telah disarankan bahwa nilai V(G) dari setiap modul tidak boleh melebihi 10. Melakukan hal itu akan membuat modul sulit dimengerti oleh manusia dan memiliki resiko terjadinya *loophole* dalam sebuah *pseudocode*. Jika ada modul yang ditemukan memiliki nilai V(G) lebih besar dari 10, maka modul tersebut harus dipertimbangkan untuk didesain ulang.

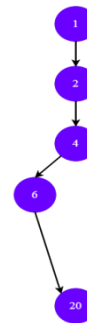
##### 1. Basic Path Koneksi Kontrol Pagar

```

1. if(firebase.get(firebaseData, "/control/" + (String)ROOM_NAME)) {
2.   if (firebaseData.dataType() == "string") {
3.     int statusPintu = firebaseData.stringData().toInt();
4.     if (statusPintu)
5.     {
6.       if(!statusOpen){
7.         espSerial.println(1);
8.         Serial.println((String)ROOM_NAME+" terbuka");
9.       }
10.      statusOpen=true;
11.    }
12.   else
13.   {
14.     if(statusOpen){
15.       espSerial.println(2);
16.       Serial.println((String)ROOM_NAME+" terkunci");
17.     }
18.     statusOpen=false;
19.   }
20. }
21. }

```

Gambar 4.4 Pseudocode dan line number Koneksi Kontrol Pagar



Gambar 4.5 Flowgraph Koneksi Kontrol Pagar

Berdasarkan *Flowgraph* tersebut dapat dihitung *Cyclomatic Complexity* berikut:

$$V(G) = E - N + 2P$$

$$V(G) = 7 - 7 + 2$$

$$V(G) = 2$$

Maka jalur independen yang didapat :

Path 1 : 1, 2, 4, 6, 20

Path 2 : 1, 2, 4, 12, 14, 20

Setelah menghitung CC maka jumlah tes yang dilakukan adalah sebanyak 4 kali dan berdasarkan jumlah CC yang ditemukan bahwa *pseudocode* Koneksi Kontrol Pagar masih dalam kategori modul sederhana dengan resiko kecil sehingga *pseudocode* dapat dikatakan aman.

Tabel 4.7 Skenario Jalur Independen Koneksi Kontrol Pagar

Path	Skenario yang diharapkan	Skenario yang sebenarnya	Validasi
1	Ketika aplikasi android sudah terkoneksi dengan pagar , dan klik tombol <i>open</i> maka pagar akan terbuka	Ketika aplikasi android sudah terkoneksi dengan pagar , dan klik tombol <i>open</i> maka pagar akan terbuka	Sesuai
2	Ketika aplikasi android sudah terkoneksi dengan pagar , dan klik tombol <i>close</i> maka pagar akan tertutup	Ketika aplikasi android sudah terkoneksi dengan pagar , dan klik tombol <i>close</i> maka pagar akan tertutup	Sesuai

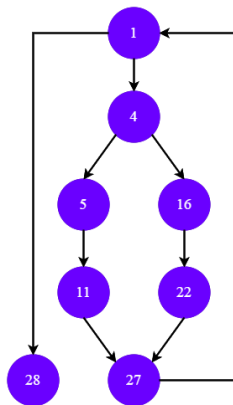
## 2. Basis Path Kontrol Pagar Koneksi WiFi Dan Informasi LCD

```

1. while (espSerial.available() > 0)
2. {
3.   int dataIn = espSerial.parseInt();
4.   if(dataIn==2){
5.     if(statusBuka){
6.       lcd.setCursor(0, 0);
7.       lcd.print(" Gerbang Tertutup");
8.       lcd.setCursor(0, 1);
9.       lcd.print(" By App Android ");
10.
11.       Direction=false;
12.       steps_left=5000;
13.     }
14.     statusBuka=false;
15.   } else if(dataIn==1){
16.     if(!statusBuka){
17.       lcd.setCursor(0, 0);
18.       lcd.print("Gerbang Terbuka");
19.       lcd.setCursor(0, 1);
20.       lcd.print(" By App Android ");
21.       Direction=true;
22.       steps_left=5000;
23.     }
24.     statusBuka=true;
25.   }
26. }
27. }
28. }

```

Gambar 4.6 Pseudocode dan line number Kontrol Pagar Koneksi WiFi Dan Informasi LCD



Gambar 4.7 Flowgraph Kontrol Pagar Koneksi WiFi Dan Informasi LCD

Berdasarkan *Flowgraph* tersebut dapat dihitung *Cyclomatic Complexity* berikut:

$$V(G) = E - N + 2P$$

$$V(G) = 9 - 8 + 2$$

$$V(G) = 3$$

Maka jalur independen yang didapat :

Path 1 : 1, 4, 5, 11, 27, 1, 28

Path 2 : 1, 4, 16, 22, 27, 1, 28

Path 3 : 1, 28

Setelah menghitung CC maka jumlah tes yang dilakukan adalah sebanyak 4 kali dan berdasarkan jumlah CC yang ditemukan bahwa *pseudocode* Kontrol Pagar Koneksi WiFi Dan Informasi LCD masih dalam kategori modul sederhana dengan resiko kecil sehingga *pseudocode* dapat dikatakan aman.

Tabel 4.8 Skenario Jalur Independen Kontrol Pagar Koneksi WiFi Dan Informasi LCD

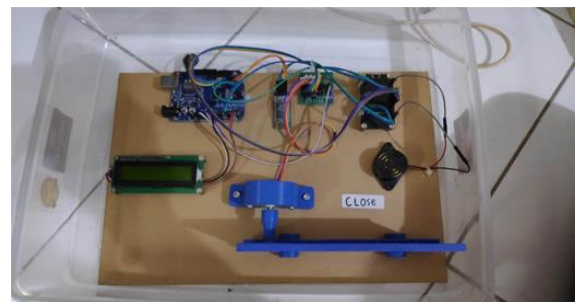
Path	Skenario yang diharapkan	Skenario yang sebenarnya	Validasi
1	Ketika aplikasi android sudah terkoneksi dengan	Ketika aplikasi android sudah terkoneksi dengan	Sesuai

	pagar dan klik tombol <i>close</i> , maka pagar akan tertutup dan LCD akan memberikan informasi bahwa "Gerang Tertutup By App Android", lalu pagar akan bergerak menutup	pagar dan klik tombol <i>close</i> , maka pagar akan tertutup dan LCD akan memberikan informasi bahwa "Gerang Tertutup By App Android", lalu pagar akan bergerak menutup	
2	Ketika aplikasi android sudah terkoneksi dengan pagar dan klik tombol <i>open</i> , maka pagar akan terbuka dan LCD akan memberikan informasi bahwa "Gerang Terbuka By App Android", lalu pagar akan bergerak membuka	Ketika aplikasi android sudah terkoneksi dengan pagar dan klik tombol <i>open</i> , maka pagar akan terbuka dan LCD akan memberikan informasi bahwa "Gerang Terbuka By App Android", lalu pagar akan bergerak membuka	Sesuai
3	Tidak ada input yang diberikan maka proses akan berhenti	Tidak ada input yang diberikan maka proses akan berhenti	Sesuai

## 5. Implementasi (*implementation*)

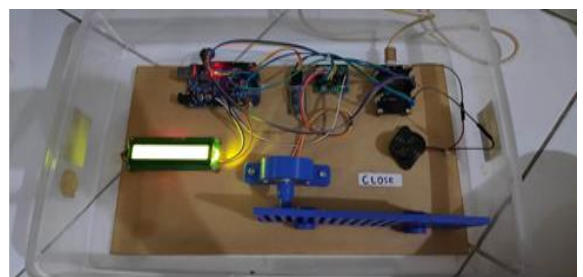
### 5.1 Alat Dalam Keadaan Mati

Pada kondisi pertama, pada gambar 5.1 semua komponen dalam keadaan tidak aktif atau *off*. Sensor pir tidak mendeteksi adanya pergerakan objek.



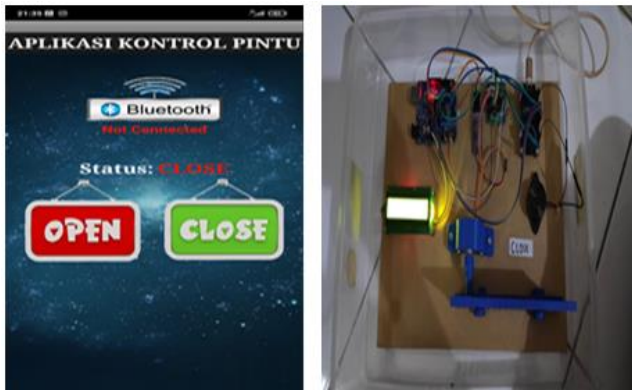
Gambar 5.1 Alat Dalam Keadaan Mati

### 5.2 Alat Dalam Keadaan Menyala Dan Belum Terkoneksi



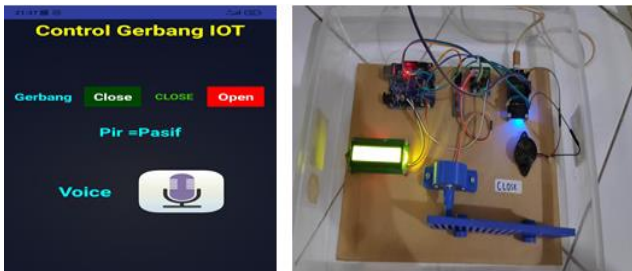
Gambar 5.2 Alat Dalam Kondisi Menyala Dan Belum Terkoneksi

Pada kondisi kedua, di gambar 5.2 alat dalam keadaan menyala atau *on*. Dan kondisi perangkat sedang tidak terkoneksi dengan Bluetooth ataupun koneksi jaringan WiFi. Hal pertama yang harus dilakukan adalah mengkoneksikan alat dengan *smartphone*, pilih salah satu koneksi yaitu dengan Bluetooth ataupun dengan jaringan WiFi. Untuk aplikasi sederhana pada *smartphone* terbagi menjadi 2, yaitu aplikasi kendali pagar menggunakan koneksi Bluetooth dan aplikasi kendali pagar menggunakan koneksi jaringan WiFi. Ini adalah tampilan ketika alat sedang terkoneksi dengan koneksi Bluetooth yang ada pada gambar 5.3.



Gambar 5.3 Aplikasi Koneksi Bluetooth Dan Alat Dalam keadaan Menyala

Pada kondisi ketiga di gambar 5.4, tampilan alat dengan koneksi jaringan WiFi dan menggunakan aplikasi sederhana yang berbeda dengan aplikasi sederhana yang menggunakan koneksi Bluetooth.



Gambar 5.4 Aplikasi Koneksi Jaringan WiFi Dan Alat Dalam Keadaan Menyala

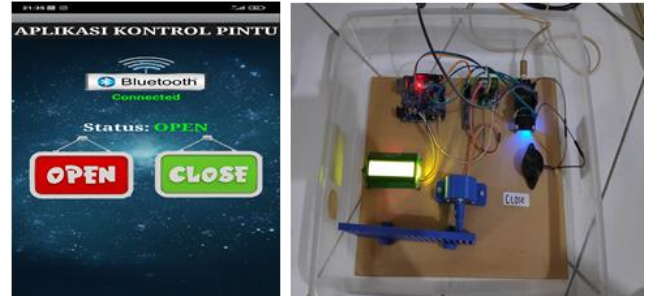
Pada kondisi ini di gambar 5.5, saat sensor pir sedang tidak ada mendeteksi pergerakan apapun dan keterangan di dalam aplikasi yang betuliskan PIR = Pasif jika sedang tidak mendeteksi adanya pergerakan.



Gambar 5.5 Sensor Pir Sedang Tidak Mendeteksi Pergerakan

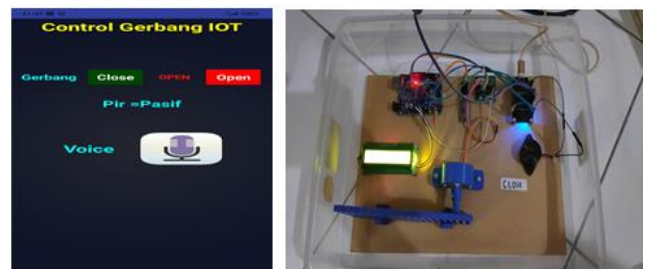
### 5.3 Alat Setelah Selesai Berkerja

Pada kondisi keempat, ini adalah tampilan aplikasi dan alat setelah selesai berkerja. Pada gambar 5.6 adalah gambar aplikasi dan alat yang sudah terkoneksi dengan perangkat Bluetooth dan pagar telah terbuka.



Gambar 5.6 Aplikasi Koneksi Bluetooth Dan Alat Setelah Pagar Terbuka

Pada gambar 5.7 adalah tampilan aplikasi dan alat setelah selesai berkerja yang terkoneksi dengan koneksi WiFi dan pagar telah terbuka, dan juga keterangan sensor pir yang belum ada mendeteksi pergerakan apapun.



Gambar 5.7 Aplikasi Koneksi Jaringan WiFi Dan Alat Setelah Pagar Terbuka

Pada gambar 5.8 adalah gambar aplikasi koneksi WiFi, terdapat keterangan Pir = Aktif yaitu bahwa sensor pir sedang mendeteksi adanya pergerakan yang berada di jangkauan sensor pir tersebut.



Gambar 5.8 Sensor Pir Mendeteksi Adanya Pergerakan

## 6. KESIMPULAN

Setelah melakukan perancangan dan pembuatan sistem, kemudian dilakukan pengujian dan mendapatkan hasilnya, maka dapat diambil beberapa kesimpulan tentang sistem kerja dari “Kendali Pagar Rumah Otomatis Berbasis Android” yang dibuat oleh penulis antara lain :

1. Jarak kendali pagar lebih jauh menggunakan koneksi jaringan WiFi dibandingkan dengan menggunakan koneksi Bluetooth.
2. Penggunaan koneksi Bluetooth lebih responsif dibandingkan dengan koneksi jaringan WiFi yang memiliki sedikit *delay*.
3. Pengujian alat menggunakan *black box* dan *white box*, tujuan dari pengujian ini yaitu untuk memastikan semua kesalahan masukan yang dilakukan oleh pengguna dapat ditangani oleh sistem.

## 7. SARAN

Agar dapat meningkatkan kualitas dari *prototype* ini penulis menyarankan untuk dapat merancang rangkaian sistem pada “Kendali Pagar Rumah Otomatis Berbasis Android”, lebih baik lagi sehingga sistem dapat bekerja lebih efektif. Saran-saran untuk pengembangan antara lain :

1. Untuk pengembangan konsep dari alat ini dapat ditambahkan sebuah sistem “*Live Streaming Video*” agar lebih mudah untuk mengetahui situasi yang ada di depan pagar.
2. Untuk meningkatkan keamanan dari pagar otomatis ini, perlu di tambahkan peningkatan keamanan dalam bentuk notifikasi secara langsung ketika pagar terbuka secara manual.
3. Disarankan untuk menggunakan mikrokontroler yang lebih baik untuk sistem yang lebih kuat dan akurat.
4. Ditambahkan alat komunikasi agar tamu yang berada di depan pagar dapat langsung berkomunikasi melalui alat tersebut tanpa keluar dari rumah.
5. Memperbaiki *voice command* agar lebih akurat dan tepat saat memberikan perintah.

## 8. DAFTAR PUSTAKA

- Andi Syofian, 2016, Pengendalian Pintu Pagar Geser Menggunakan Aplikasi *Smartphone* Android Dan Mikrokontroler, <https://docplayer.info/36550693-Pengendalian-pintu-pagar-geser-menggunakan-aplikasi-smartphone-android-dan-mikrokontroler-arduino-melalui-bluetooth.html>, Padang : Institut Teknologi Padang.
- Andry Bastian, 2015, Perancangan Sistem Keamanan Pada Sepeda Motor Dengan Menggunakan ArduinoUno Berbasis Mikrokontroller ATmega328 Melalui Media *Handphone*, Tangerang : Sekolah Tinggi Manajemen Dan Ilmu Komputer Raharja.
- Ardiansyah, 2019, Perancangan Alat Pendeteksi Hewan Pengganggu Tanaman Kebun Menggunakan Sensor Gerak PIR (*Passive Infra Red*) Berbasis Mikrokontroler, <http://repositori.uin-alauddin.ac.id/14994/1/Ardiansyah.pdf>, Makassar : Universitas Islam Negeri (UIN).
- Banud Wianggara, 2016, Rancang Bangun Pengaman Pintu Menggunakan Sidik Jari (*Fingerprint*) Dan Android Berbasis Mikrokontroler ATmega8, Pontianak : Universitas Tanjungpura Pontianak.

- Bayu, 2022, Urutan OS Android Terbaru, dari *Cupcake* hingga Android 11, <https://hpsultan.com/urutan-android-terbaru/>, Jakarta, Hp Sultan.
- Dayat Kurniawan, 2013, *Mikrokontroler dan Aplikasinya*, Jakarta : PT Penerbit Elex Media Komputindo.
- Debby Seftiani Putri, 2017, Sistem Kendali Tampilan Informasi Pada *Display Dot* Matriks *Input Karakter* Melalui Android, <http://eprints.polsri.ac.id/4540/>, Palembang : Politeknik Negeri Sriwijaya.
- Dewa De, 2019, Cara Program Modul Sensor *Infrared* FC-51 Dengan Arduino, <https://teknisibali.com/cara-program-modul-sensor-infrared-fc-51-dengan-arduino/>, Teknisibali.com.
- Dewi Masitoh, 2016, SISTEM PENDUKUNG KEPUTUSAN SELEKSI PENERIMAAN KARYAWAN BARU NABILA CAKE & BAKERY DENGAN METODE *SIMPLE ADDITIVE WEIGHTING* (SAW) BERBASIS WEB, <http://eprints.umpo.ac.id/2344/>, Ponorogo : Universitas Muhammadiyah Ponorogo.
- Franky Chandra, 2013, *Jago Elektronika Rangkaian Sistem Otomatis*, Yogyakarta : PT Penerbit Kawan Pustaka.
- Hendro Simanjuntak, 2019, Prototype Kendali Pintu Otomatis Menggunakan Motor DC Dan Sensor Pir Dengan Arduino R3 Berbasis Mikrokontroler, Samarinda : STMIK Widya Cipta Dharma.
- Khoirul Iman, 2016, LCD Dengan I2C *Module* Untuk Arduino, <https://khoiruliman.wordpress.com/2016/06/07/lcd-dengan-i2c-module-untuk-arduino/>, khoiruliman.wordpress.com.
- Lila Setiyani, 2018, REKAYASA PERANGKAT LUNAK [*Software Engineering*], [https://www.researchgate.net/publication/333209319\\_REKAYASA\\_PERANPERA\\_LUNAK\\_Software\\_Engineering](https://www.researchgate.net/publication/333209319_REKAYASA_PERANPERA_LUNAK_Software_Engineering), Pancawati, Jatayu Catra Internusa
- Malabay, 2016, Pemanfaatan *Flowchart* Untuk Kebutuhan Deskripsi Proses Bisnis, <https://ejournal.esaunggul.ac.id/index.php/JIK/article/download/1579/1426>, Jakarta : Universitas Esa Unggul Jakarta.
- Muhamad Rizki Maulana, 2018, Perancangan Sistem Pengendali Pintu Pagar Otomatis Menggunakan Android *Speech Recognition* Berbasis Arduino, <https://repositori.usu.ac.id/bitstream/handle/123456789/7236/121402037.pdf?sequence=1&isAllowed=y>, Medan : Universitas Sumatra Utara..
- Muhammad Renaldy, 2022, Rancang Bangun Aplikasi E-Arsip Berbasis Website Dengan Menerapkan Algoritma *Knuth-Morris-Pratt*, Samarinda : Sekolah Tinggi Manajemen Informatika Dan Komputer Widya Cipta Dharma.
- Nurul Huda, 2018, PENGISIAN BATERAI *ACCUMULATOR* BERBASIS *SOLAR CELL*

- TRACKING PADA TAMBAK AIR PAYAU*,  
<http://eprints.undip.ac.id/67145/>, Semarang :  
 Universitas Diponegoro.
- Ratih Dian Saraswati, Maria Damiana Nestri Kiswari,  
 2019, Pagar Dari Teritori Menjadi *Pride* dan  
 Identitas,  
[http://journal.unika.ac.id/index.php/tuturrupa/artic  
 le/download/1949/pdf](http://journal.unika.ac.id/index.php/tuturrupa/article/download/1949/pdf), Semarang : UNIKA  
 Soegijapranata Semarang.
- Riswandi, 2019, *Sistem Kontrol Vertical Garden*  
 Menggunakan NodeMCU ESP8266 Berbasis  
 Android,  
[http://repositori.uinalauddin.ac.id/15440/1/Sistem  
 %20Kontrol%20Vertical%20Garden%20Menggu  
 nakan%20NodeMCU%20ESP8266%20Berbasis  
 %20Android%20by%20Riswandi\\_602001150.pdf](http://repositori.uinalauddin.ac.id/15440/1/Sistem%20Kontrol%20Vertical%20Garden%20Menggunakan%20NodeMCU%20ESP8266%20Berbasis%20Android%20by%20Riswandi_602001150.pdf),  
 Makassar : Universitas Islam Negeri Alauddin  
 Makassar.
- Sufajar Butsianto, Eka Nur Arifin, 2020,  
 PENGEMBANGAN SISTEM INFORMASI  
 PENJUALAN BERBASIS WEB  
 MENGGUNAKAN METODE PROTOTYPING  
 PADA TOKO BAY STICKER,  
[https://www.jurnal.pelitabangsa.ac.id/index.php/si  
 gma/article/download/540/350#:~:text=Menurut  
 %20Rosa%20A.S%2C%20Prototyping%20adalah  
 %20dalam%20bentuk%20yang%20telah%20selesai%  
 20.,](https://www.jurnal.pelitabangsa.ac.id/index.php/sigma/article/download/540/350#:~:text=Menurut%20Rosa%20A.S%2C%20Prototyping%20adalah%20dalam%20bentuk%20yang%20telah%20selesai%20.,) Bekasi : Universitas Pelita Bangsa.
- Syahrul, 2016, *Motor Stepper: Teknologi, Metoda Dan*  
 Rangkaian Kontrol,  
[https://jurnal.unikom.ac.id/\\_s/data/jurnal/v06-  
 n02/vol-6-artikel-7.pdf/pdf/vol-6-artikel-7.pdf](https://jurnal.unikom.ac.id/_s/data/jurnal/v06-n02/vol-6-artikel-7.pdf/pdf/vol-6-artikel-7.pdf),  
 Bandung : Universitas Komputer Indonesia.
- Wahyu Rizki, 2019, *SMART TRASH BIN* BERBASIS  
 ARDUINO DENGAN SENSOR *ULTRASONIK*  
 DAN *SMS GATEAWAY*,  
[https://repository.bsi.ac.id/index.php/repo/viewite  
 m/21843](https://repository.bsi.ac.id/index.php/repo/viewitem/21843), Banyumas : Universitas Bina Sarana  
 Informatika.
- Yauma Patricia Anggelausia, 2019, *SISTEM*  
*MONITORING DAN ON-OFF* OTOMATIS  
 POMPA AIR PADA TANDON,  
<http://eprints.polsri.ac.id/6841/>, Palembang :  
 Politeknik Negeri Sriwijaya.
- Yoga Laksamana Jati, 2019, Pengembangan *Trainer*  
 Pembelajaran (*Voice Command Robotic*) Robot  
 Dengan Perintah Suara,  
<https://eprints.uny.ac.id/63926/>, Yogyakarta :  
 Universitas Negeri Yogyakarta.
- Yogie El Anwar, Noer Soedjarwanto, Ageng Sadnowo  
 Repelianto, 2015, *Prototype* Penggerak Pintu  
 Pagar Otomatis Berbasis Arduino Uno Atmega  
 328P Dengan Sensor Sidik Jari,  
[https://electrician.unila.ac.id/index.php/ojs/article/  
 view/158/pdf](https://electrician.unila.ac.id/index.php/ojs/article/view/158/pdf), Bandar Lampung : Universitas  
 Lampung.
- Yoki Firmansyah, Deasy Purwaningtias, Lisa Pratiwi,  
 2019, *PROTOTYPE SISTEM INFORMASI*  
*PENGOLAHAN DANA BOS (SIP BOS)*  
 BERBASIS WEB,  
[https://media.neliti.com/media/publications/31382  
 7-prototype-sistem-informasi-pengolahan-da-  
 b401366e.pdf](https://media.neliti.com/media/publications/313827-prototype-sistem-informasi-pengolahan-da-b401366e.pdf), Pontianak : Universitas Bina  
 Sarana Informatika Kampus Kota Pontianak.