

**PERANCANGAN SISTEM KENDALI PINTU PAGAR RUMAH DAN
PINTU GARASI PINTAR MENGGUNAKAN NODEMCU ESP32
BERBASIS INTERNET OF THINGS (IoT)**

PROYEK AKHIR

*Diajukan Sebagai Salah Satu Persyaratan Untuk Mendapatkan Gelar Ahli
Madya Pada Program Studi Diploma (III) Teknik Elektronika
Departemen Teknik Elektronika Fakultas Teknik
Universitas Negeri Padang*



OLEH :

RINDU LAHIRA DIESA

21066033

PROGRAM STUDI DIPLOMA III TEKNIK ELEKTRONIKA

DEPARTEMEN TEKNIK ELEKTRONIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI PADANG

2025

HALAMAN PERSETUJUAN PEMBIMBING

Judul : Perancangan Sistem Kendali Pintu Pagar Rumah Dan Pintu
Garasi Pintar Menggunakan NodeMCU ESP32 Berbasis
Internet Of Things (IoT)

Nama : Rindu Lahira Diesa

Nim : 21066033

Program Studi : Teknik Elektronika

Departemen : Elektronika

Fakultas : Teknik

Padang, Maret 2025

Disetujui Oleh:

Pembimbing



Ryan Fikri, S.Pd., M.T.
Nip. 19950106 202203 1 015

Mengetahui

Kepala Departemen



Dr. Dedv Irfan, S.Pd., M.Kom
NIP. 19760408 200501 1 002

HALAMAN PENGESAHAN PROYEK AKHIR

Nama : Rindu Lahira Diesa

NIM : 21066033

Dinyatakan Lulus Setelah Mempertahankan di Depan Tim Penguji Proyek Akhir
Program Studi Teknik Elektronika Departemen Teknik Elektronika
Fakultas Teknik Universitas Negeri Padang Dengan Judul:

**Perancangan Sistem Kendali Pintu Pagor Rumah Dan Pintu Garasi Pintar
Menggunakan NodeMCU ESP32 Berbasis *Internet Of Things* (IoT)**

Padang, Maret 2025

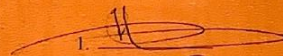

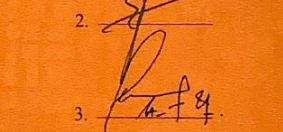
Tim penguji:

Tanda Tangan

1. Drs. Almasri, M.T

2. Dr. Yasdinul Huda, M.T

3. Ryan Fikri, S.Pd., M.T

1. 
2. 
3. 

SURAT TIDAK PLAGIAT

Saya yang bertanda tangan dibawah ini :

Nama : Rindu Lahira Diesa
TM/NIM : 2021/21066033
Program Studi : D3 Teknik Elektronika
Jurusan : Teknik Elektronika
Fakultas : Teknik

Dengan ini saya menyatakan, bahwa tugas akhir yang berjudul "**Perancangan Sistem Kendali Pintu Pagar Rumah Dan Pintu Garasi Pintar Menggunakan NodeMCU ESP32 Berbasis *Internet Of Things* (IoT).**" ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah lazim. Demikian saya buat dengan penuh kesadaran dan rasa tanggung jawab sebagai anggota masyarakat ilmiah.

Padang, Maret 2025
Yang Menyatakan,



Rindu Lahira Diesa
Rindu Lahira Diesa
NIM. 21066033

ABSTRAK

Rindu Lahira Diesa
(21066033)

Perancangan Sistem Kendali Pintu Pagar Dan Pintu
Garasi Pintar Menggunakan Nodemcu ESP32 Berbasis
Internet Of Things (IoT)

Proyek akhir ini bertujuan untuk merancang sistem kendali pintu pagar rumah dan pintu garasi pintar menggunakan NodeMCU ESP32 berbasis *Internet of Things* (IoT). Sistem ini dirancang untuk mempermudah pemilik rumah dalam mengontrol pintu pagar dan garasi serta lampu garasi melalui aplikasi *Blynk* yang dapat diakses via smartphone. Sistem ini menggunakan sensor ultrasonik untuk mendeteksi keberadaan kendaraan dan menggerakkan motor servo secara otomatis untuk membuka dan menutup pintu pagar dan garasi. Selain itu, sistem juga dilengkapi dengan kontrol lampu garasi yang dapat dihidupkan atau dimatikan melalui aplikasi *Blynk*. Hasil simulasi menunjukkan bahwa sistem dapat berfungsi dengan baik, meskipun respons sistem bergantung pada kualitas jaringan internet. Proyek ini diharapkan dapat meningkatkan efisiensi dan kenyamanan dalam pengelolaan pintu pagar dan garasi.

Key Words : NodeMCU ESP32, *Internet of Things* (IoT), *Blynk*, sensor ultrasonik, motor servo, kontrol pintu otomatis.

KATA PENGANTAR

Puji syukur penulis ucapkan kehadirat Allah Subhanahu Wata'ala karena berkat dan karunia-Nya yang begitu besar sehingga penulis dapat menyelesaikan Proyek Akhir ini yang berjudul “Perancangan Sistem Kendali Pintu Pagar Rumah Dan Pintu Garasi Pintar Menggunakan Nodemcu Esp32 Berbasis *Internet Of Things* (Iot)”.

Shalawat dan salam tak lupa penulis kirimkan kepada junjungan kita, yakninya Nabi Besar Muhammad SAW yang menjadi suri tauladan dalam setiap sikap dan tindakan sebagai seorang muslim dan muslimah.

Pembuatan Proyek Akhir ini merupakan salah satu syarat dalam menyelesaikan pendidikan Diploma Tiga (D3) Jurusan Elektronika Fakultas Teknik Universitas Negeri Padang. Penyelesaian Proyek Akhir ini tidak terlepas dari bantuan bimbingan serta dorongan dari berbagai pihak sehingga dapat diselesaikan segala hambatan dan rintangan yang dihadapi, untuk itu penulis mengucapkan terimakasih kepada :

1. Bapak Krismadinata, S.T., M.T., Ph.D. Selaku Rektor Universitas Negeri Padang.
2. Bapak Dr. Muhammad Anwar, S.Pd., M.T selaku Dekan Fakultas Teknik Universitas Negeri Padang.
3. Bapak Dr. Dedy Irfan, S.Pd,M.Kom selaku Kepala Departemen Teknik Elektronika Universitas Negeri Padang.
4. Bapak Dr. Zulwisli, S.Pd., M.Eng selaku Ketua Program Studi Diploma III Teknik Elektronika Fakultas Teknik Universitas Negeri Padang.

5. Bapak Ryan Fikri, S.Pd., M.T selaku dosen pembimbing yang telah memberikan masukan dan saran dalam penyelesaian proyek akhir ini.
6. Bapak Drs.Almasri,M.T selaku dosen penguji pada sidang proyek akhir ini.
7. Bapak Dr.Yasdinul Huda,M.T selaku dosen penguji pada sidang proyek akhir ini.
8. Seluruh Staf Pengajar, Pegawai beserta Teknisi Labor Departemen Teknik Elektronika.
9. Kepada orang tua saya Bapak Osfaldo Ardiles dan Ibuk Mailasri, adik-adik saya Nafsa Zikri Diesa, Nafiis Faruq Diesa, beserta keluarga besar Amak Nur Aini yang telah mendoakan dan memberikan dukungan baik moril maupun material sehingga saya selaku anak yang ayah dan ibun cintai ini dapat menyelesaikan proyek akhir ini.
10. Kepada Rendhi Fitra Yanna yang saat ini bersama saya, yang telah bersedia membantu, mendoakan dan mensupport saya serta mendengarkan dan menerima keluh kesah saya selama penyusunan proyek akhir ini.
11. Terimakasih juga untuk rekan dan sahabat saya Dzikra Adila, Nadia Dwi Afani, Nabila Kencana Wungu, Wini Zilfiani atas semua support dan semangatnya yang tak pernah pudar sedari kecil sampai saat ini. Semoga kita selalu dapat kebersamai dalam setiap momen berharga yang akan kita raih bersama.
12. Kepada seluruh pihak yang telah banyak membantu dalam menyusun laporan yang tidak bisa disebut namanya satu persatu.
13. Kepada diri sendiri Rindu Lahira Diesa yang telah berhasil berjuang dan melewati segala rintangan dan cobaan dalam masa perkuliahan sampai dengan penulisan proyek akhir ini, terimakasih sudah mampu bertahan sampai dengan

sejauh ini, semoga kamu lebih semangat dan lebih mampu untuk menyelesaikan semua rintangan yang lebih besar kedepannya

Semoga segala motivasi, dorongan dan bantuan serta bimbingan yang diberikan menjadi amal jariah dan mendapat balasan yang setimpal dari Allah SWT.

Padang, Maret 2025

Rindu Lahira Diesa
NIM. 21066033

MOTTO DAN PERSEMBAHAN

MOTTO :

Teruslah berusaha dan berjuang semaksimal mungkin jangan mudah menyerah dalam menghadapi segala rintangan dan cobaan dalam hidup.

PERSEMBAHAN :

Dengan menyebut nama ALLAH SWT Yang Maha Pengasih dan Maha Penyayang, segala puji bagi ALLAH atas segala nikmat dan rahmat serta karunia-Nya yang tak terhingga.

Sholawat serta salam semoga selalu tercurah kepada Nabi Muhammad saw, keluarga, sahabat dan para pengikutnya. Karya sederhana ini dipersembahkan Kepada-Mu yang tak terhingga. Khususnya atas kesempatan yang Engkau berikan untuk menyelesaikan studi ini.

Karya ini juga dipersembahkan kepada kedua orang tua tercinta Bapak Osfaldo Ardiles dan Ibuk Mailasri yang telah mendidik, membesarkan, dan memberi kasih sayang yang tak terhingga. Mereka adalah inspirasi terbesar dalam hidup saya untuk menjadi lebih baik kedepannya. Tak lupa juga dipersembahkan kepada keluarga besar Amak Nur Aini yang selalu memberikan dukungan dan doa. Semoga karya sederhana ini dapat bermanfaat bagi banyak pihak.

DAFTAR ISI

| | Halaman |
|---|-------------------------------------|
| HALAMAN JUDUL..... | i |
| HALAMAN PERSETUJUAN PEMBIMBING | Error! Bookmark not defined. |
| HALAMAN PENGESAHAN PROYEK AKHIR . | Error! Bookmark not defined. |
| KATA PENGANTAR..... | iii |
| MOTTO DAN PERSEMBAHAN..... | viii |
| DAFTAR ISI..... | ix |
| DAFTAR GAMBAR | xi |
| DAFTAR TABEL | xii |
| BAB I PENDAHULUAN..... | 1 |
| A. Latar Belakang | 1 |
| B. Identifikasi Masalah..... | 3 |
| C. Batasan Masalah | 3 |
| D. Rumusan Masalah..... | 4 |
| E. Tujuan Proyek Akhir | 4 |
| F. Manfaat Proyek Akhir..... | 4 |
| BAB II HASIL DAN PEMBAHASAN..... | 5 |
| A. Analisis Kebutuhan Proyek..... | 5 |
| B. Desain Proyek Akhir | 13 |
| C. Deskripsi Proyek Akhir..... | 20 |
| D. Hasil dan Pembahasan | 25 |
| BAB III KESIMPULAN DAN REKOMENDASI | 28 |
| A. Kesimpulan | 28 |
| B. Rekomendasi..... | 28 |
| C. Rangkain Pengembangan..... | 29 |

| | |
|---|-----------|
| D. Kesimpulan Rangkaian Pengembangan..... | 31 |
| E. Rekomendasi Rangkaian Pengembangan | 31 |
| DAFTAR PUSTAKA | 32 |
| LAMPIRAN..... | 33 |

DAFTAR GAMBAR

| | Halaman |
|--|---------|
| Gambar 1. Diagram Aliran Proses | 5 |
| Gambar 2. <i>Software Blynk</i> | 8 |
| Gambar 3. Wokwi | 8 |
| Gambar 4. <i>Smartphone</i> | 9 |
| Gambar 5. NodeMCU ESP32 | 10 |
| Gambar 6. LED | 11 |
| Gambar 7. Motor Servo..... | 11 |
| Gambar 8. Sensor Ultrasonik | 12 |
| Gambar 9. LCD | 13 |
| Gambar 10. Blok Diagram | 16 |
| Gambar 11. <i>Flowchart</i> Sistem | 19 |
| Gambar 12. Skematik Rangkain | 19 |
| Gambar 13. Tampilan Program Simulasi | 21 |
| Gambar 14. Tampilan Simulasi Pagar Terbuka Dengan Sensor | 21 |
| Gambar 15. Tampilan Simulasi Pintu Garasi Terbuka Dengan Sensor..... | 22 |
| Gambar 16. Tampilan Simulasi Lampu Dalam Keadaan Hidup Dengan <i>Blynk</i> ... | 23 |
| Gambar 17. Tampilan Simulasi Lampu Dalam Keadaan Hidup Dengan <i>Blynk</i> ... | 23 |
| Gambar 18. Tampilan Kontrol Pintu Pagar Terbuka Pada <i>Blynk</i> | 24 |
| Gambar 19. Tampilan Simulasi Pagar Terbuka Dengan <i>Blynk</i> | 24 |
| Gambar 20. Tampilan Kontrol Pintu Garasi Terbuka Pada <i>Blynk</i> | 25 |
| Gambar 21. Tampilan Simulasi Pintu Garasi Terbuka Dengan <i>Blynk</i> | 25 |
| Gambar 22. Skematik Rangkaian Pengembangan | 30 |

DAFTAR TABEL

Halaman

| | |
|--------------------------------|----|
| Tabel 1. Studi Literatur | 13 |
|--------------------------------|----|

BAB I

PENDAHULUAN

A. Latar Belakang

Revolusi industri 4.0 ditandai dengan perkembangan teknologi, salah satu contoh perkembangan teknologi ini adalah *Internet of Things*. IoT merupakan sebuah konsep di mana suatu benda atau objek ditanamkan teknologi-teknologi seperti sensor dan *software* dengan tujuan untuk berkomunikasi, mengendalikan, menghubungkan, dan bertukar data melalui perangkat lain selama masih terhubung ke internet. IoT memiliki hubungan yang erat dengan istilah *Machine-to-Machine* atau M2M. Seluruh alat yang memiliki kemampuan komunikasi M2M ini sering disebut dengan perangkat cerdas atau *smart devices*. Perangkat cerdas ini diharapkan dapat membantu kerja manusia dalam menyelesaikan berbagai urusan atau tugas yang ada.

Pagar merupakan keamanan paling terdepan dalam menjaga suatu bangunan. Pagar menggunakan kunci tambahan untuk mengatasi lemahnya perlindungan yang tidak luput dari pelaku tindakan kriminal. Kebutuhan yang meningkat membuat setiap orang bergantung pada teknologi, salah satunya dengan mengaplikasikan teknologi untuk mempermudah pekerjaan. Untuk mempermudah pemilik rumah membuka dan menutup pagar tanpa harus mengeluarkan tenaga yang besar, maka diperlukan sebuah rangkaian elektronik yang dapat bekerja secara otomatis untuk membuka dan menutup pagar yang dapat diakses melalui *smartphone* android untuk membuka pintu garasi dan menutup secara otomatis melalui jaringan *wifi*.

Internet of Things merupakan sebuah konsep di mana suatu benda atau objek ditanamkan teknologi-teknologi seperti sensor dan *software* dengan tujuan untuk berkomunikasi, mengendalikan, menghubungkan, dan bertukar data melalui perangkat lain selama masih terhubung ke internet. *Internet of Things* adalah suatu sistem yang digunakan oleh manusia bisa terhubung dengan internet, sehingga dengan sistem ini pengguna tersebut dapat memantau atau alat mengendalikan dari jarak jauh atau online (Sujono & Ramadhan, 2020).

Berdasarkan hasil studi Pustaka penelitian Alreza Putra, dkk (2018) berjudul “Rancang Bangun Prototipe Buka Tutup Pintu Bendungan Otomatis Berbasis Iot Menggunakan SMS Gateway” menggunakan sensor ketinggian, notifikasi melalui sms dengan modul SIM800L dan web untuk memonitoring ketinggian air didapatkan bahwa sistem yang dirancang mampu membuka dan menutup pintu air bendungan, data yang terbaca sensor ketinggian air oleh NodeMCU dikirim ke modul SIM800L untuk selanjutnya dikirimkan notifikasi berupa SMS dan tampilan pada website. Sedangkan penelitian lainnya oleh Ericson D. Situmorang, dkk. Tentang rancang bangun alat buka tutup pintu pagar dengan menggunakan *handphone* dan *keypad* juga memiliki kelemahan yaitu akses jarak harus dekat antara *keypad* dengan pagar pintu rumah.

Berdasarkan penelitian-penelitian tersebut, maka penulis merancang sebuah sistem pengontrol otomatis pagar rumah berbasis *Internet of Things*. Sistem ini menggunakan NodeMCU sebagai pengontrol pintu pagar rumah, motor DC dan motor servo sebagai penggerak pintu pagar serta *smartphone* android sebagai perangkat yang memerintahkan pagar terbuka atau tertutup. Semua *device* terhubung menggunakan internet dengan aplikasi *Blynk*.

Proyek akhir ini dibuat untuk mempermudah pemilik rumah membuka dan menutup pagar rumah serta pintu garasi dan mengontrol lampu dalam garasi dengan judul “Perancangan Sistem Kendali Pintu Pagar Rumah Dan Pintu Garasi Pintar Menggunakan NodeMCU ESP32 Berbasis *Internet Of Things* (Iot)”

B. Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan dapat diidentifikasi masalah yang akan penulis jadikan sebagai bahan proyek akhir yaitu sebagai berikut:

1. Pemanfaatan *Internet of Things* untuk kemudahan pemilik rumah dalam mengontrol pagar.
2. Efisiensi waktu pemilik rumah untuk membuka dan menutup pagar rumah dan pintu garasi serta kontrol lampu garasi.

C. Batasan Masalah

Dalam pembuatan proyek akhir ini, penulis membatasi ruang lingkup permasalahan untuk menghindari kerancuan pembahasan yang meluas dalam proyek akhir antara lain:

1. Proyek akhir ini hanya terbatas dalam perancangan dan uji coba sistem garasi pintar menggunakan wokwi untuk mensimulasikan sistem.
2. Antar muka menggunakan *Blynk* sebagai pusat kontrol pada sistem yang dirancang dan di uji cobakan.

D. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka dirumuskan masalah bagaimana merancang sebuah sistem kendali pagar rumah dan pintu garasi serta kontrol hidup atau mati lampu garasi melalui aplikasi *Blynk*?

E. Tujuan Proyek Akhir

Tujuan pembuatan proyek akhir ini adalah:

1. Membuat sistem pengendali pagar dan pintu garasi otomatis menggunakan NodeMCU ESP32 berbasis IoT.
2. Merancang lampu otomatis sebagai penerangan garasi pada malam hari pada saat pintu garasi terbuka.
3. Memahami prinsip kerja putaran motor servo dengan beban pagar dan panjang lintasan rel.

F. Manfaat Proyek Akhir

Adapun manfaat yang ingin dicapai dari pembuatan proyek akhir ini adalah sebuah sistem kendali pagar rumah dan pintu garasi serta kontrol hidup atau mati lampu garasi melalui aplikasi *Blynk* untuk mempermudah pemilik rumah melalui *smartphone* berbasis IoT.

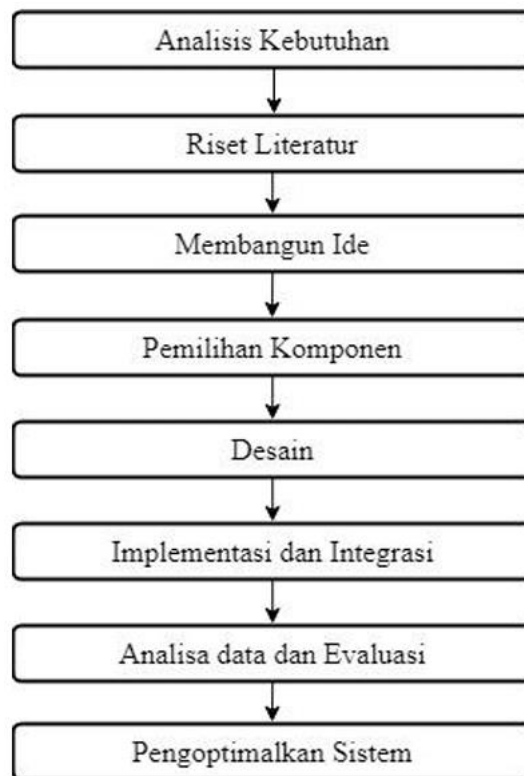
BAB II

HASIL DAN PEMBAHASAN

A. Analisis Kebutuhan Proyek

1. Diagram Aliran Proses

Diagram aliran proses merupakan representasi visual yang digunakan untuk menggambarkan langkah-langkah atau proses yang digunakan untuk menyelesaikan proyek akhir ini atau mencapai suatu tujuan dalam suatu sistem atau aktivitas.



Gambar 1. Diagram Aliran Proses

Tahapan-tahapan yang akan dilakukan pada saat perancangan alat dapat dilihat pada Gambar 1 dengan penjelasan sebagai berikut:

a. Analisis Kebutuhan Tahap awal

Dalam pengembangan sistem ini, memahami dengan jelas kebutuhan pengguna dan masalah yang ingin dipecahkan sangat perlu dilakukan untuk mendapatkan gambaran yang jelas tentang spesifikasi sistem.

b. Riset Literatur

Melakukan riset literatur dengan cara mencari informasi dari berbagai sumber untuk dilakukan perbandingan dan menemukan inspirasi baru untuk penelitian selanjutnya.

c. Membangun Ide

Membangun ide dengan judul perancangan sistem kendali pagar rumah dan garasi pintar menggunakan NodeMCU ESP32 berbasis *internet of things* (IoT) dikembangkan yang belum ada pada penelitian terdahulu.

d. Pemilihan Komponen

Komponen yang dipilih harus sesuai spesifikasi yang sudah ditentukan pada tahap analisis kebutuhan seperti sensor, motor servo, ESP32, LED dan komponen pendukung lainnya.

e. Desain

Tahap desain merupakan tahapan yang sangat penting karena pada tahap ini ide-ide yang telah dibangun akan diwujudkan dalam bentuk desain yang lebih detail. Desain yang baik akan mempermudah proses pembuatan dan memastikan bahwa alat yang dihasilkan berfungsi dengan baik.

f. Implementasi dan Integrasi

Pada tahap ini, akan mulai mengimplementasikan desain yang telah dirancang, komponen yang telah dipilih dihubungkan sesuai dengan desain yang telah dibuat dan juga melibatkan pemograman.

g. Analisa Data dan Evaluasi

Pada tahap analisa data dan evaluasi sistem akan dioperasikan dan data kinerja sistem akan dikumpulkan. Data ini kemudian akan dianalisis untuk mengetahui apakah sistem bekerja sesuai dengan yang diharapkan. Evaluasi juga dilakukan untuk mengidentifikasi kekurangan atau masalah yang perlu diperbaiki.

h. Optimasi Sistem

Melakukan perubahan-perubahan yang diperlukan untuk meningkatkan kinerja sistem monitoring kendali pagar rumah dan garasi pintar.

2. Kebutuhan Perangkat Lunak

Perancangan sistem monitoring kendali pagar rumah dan garasi pintar menggunakan NodeMCU ESP32 berbasis *Internet of Things* yang dijelaskan sebagai berikut:

a. Blynk

Blynk merupakan *platform* untuk aplikasi OS *Mobile* yang memiliki fungsi untuk mengendalikan atau memantau suatu modul seperti ESP32, NodeMCU, Motor Servo, dan modul lain yang bisa terkoneksi jaringan internet. *Blynk* memiliki pengaturan yang sederhana dan mudah tetapi memiliki layanan yang cukup lengkap seperti tombol pengendali, *chart*,

video streaming, notifikasi email dan lainnya. Di dalam pembuatan sistem ini menggunakan aplikasi *Blynk* yang berfungsi untuk mengaktifkan sistem pagar otomatis, pintu garasi serta lampu garasi. Berikut bentuk tampilan aplikasi *Blynk* dapat dilihat pada Gambar 2 dibawah ini:



Gambar 2. *Software Blynk*

b. Wokwi

Wokwi merupakan salah satu *platform* online simulator berbasis *website* yang mana kita bisa menjalankan simulasi mikrokontroler (ESP32, dll) didalamnya. Wokwi cocok untuk tujuan belajar karena penggunaannya simple dengan *interface* yang mudah dimengerti, bentuk tampilan *website* wokwi ini dapat dilihat pada Gambar 3 dibawah ini:



Gambar 3. Wokwi

3. Kebutuhan Perangkat Keras

Pada proses perancangan sistem kendali pagar rumah dan garasi pintar menggunakan NodeMCU ESP32 Berbasis *Internet of Things*

dibutuhkan beberapa perangkat keras yang akan menjadi satu kesatuan sistem yaitu sebagai berikut:

a. *Smartphone*

Secara umum, *smartphone* adalah perangkat elektronik genggam yang memiliki kemampuan komputasi dan koneksi internet yang canggih yang dirancang untuk memberikan berbagai fitur dan fungsionalitas yang lebih kompleks dari pada *handphone* biasa, seperti layar sentuh yang lebih besar, kamera yang lebih baik, penyimpanan yang lebih besar, pemrosesan yang lebih cepat, dan kemampuan untuk mengakses internet dan aplikasi yang lebih banyak.

Menurut David Wood, *smartphone* adalah *handphone* cerdas yang mempunyai kelebihan dibandingkan dengan alat telekomunikasi lainnya. Kelebihannya akan dapat terlihat dari proses pembuatannya serta proses dalam penggunaannya. Pada Gambar 4 dibawah ini dapat dilihat bentuk dari *Smartphone*.



Gambar 4. *Smartphone*

b. ESP32

ESP32 adalah mikrokontroler berharga rendah dan hemat energi dengan *wifi* dan dua-mode *bluetooth* terintegrasi. Generasi ESP32 menggunakan mikroprosesor *Tensilica Xtensa LX6* sebagai inti. Baik

dalam mode *single-core* maupun *dual-core*. ESP32 dibuat oleh *Espressif Systems*, Perusahaan berbasis di Shanghai, Tiongkok.

Keunggulan mikrokontroler ESP32 dibanding dengan mikrokontroler yang lain, terlihat dari pin out nya yang lebih banyak, pin analog lebih banyak, memori yang lebih besar, terdapat *bluetooth 4.0 low energy* serta tersedia *Wifi* yang memungkinkan untuk mengaplikasikan *Internet of Things* (IoT) dengan mikokontroler ESP32 (Imran, 2020). Bentuk rangkaian NodeMCU ESP32 dapat dilihat pada Gambar 5 dibawah ini:

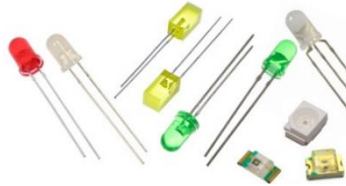


Gambar 5. NodeMCU ESP32

c. LED (*Light Emitting Dioda*)

Light Emitting Diode atau sering disingkat dengan LED adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga Dioda yang terbuat dari bahan semikonduktor. Warna-warna Cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang dipergunakannya. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti yang sering kita jumpai pada *Remote Kontrol* TV ataupun *Remote Kontrol* perangkat elektronik lainnya.

Dapat dilihat pada Gambar 6 dibawah ini bentuk dari LED yang digunakan.



Gambar 6. LED

d. Motor Servo

Motor servo adalah aktuator putar yang dirancang menggunakan sistem kontrol umpan balik *loop* tertutup, sehingga bisa diatur untuk menentukan serta memastikan posisi sudut dari poros keluaran motor. Motor servo terdiri dari motor DC, serangkaian roda gigi, rangkaian kontrol dan potensiometer. Rangkaian roda gigi yang melekat pada poros motor DC akan menghambat putaran poros dan menaikkan torsi motor servo, dan potensiometer dengan perubahan hambatannya saat motor bergerak berfungsi untuk penentu batas posisi putaran poros motor servo. Pada perancangan sistem penggerak pintu garasi otomatis ini menggunakan motor servo, dapat dilihat pada Gambar 7 dibawah ini bentuk tampilan dari motor servo yang digunakan.



Gambar 7. Motor Servo

e. Sensor Ultrasonik

Sensor ultrasonik adalah alat yang beroperasi dengan memanfaatkan pantulan gelombang suara untuk mengidentifikasi keberadaan suatu objek tertentu yang berada di depannya. Sensor ultrasonik terdiri dari dua komponen utama, yaitu unit pemancar dan unit penerima. Kedua unit ini bekerja sama untuk menghasilkan informasi yang diperlukan. Rentang frekuensi kerjanya berkisar antara 40KHz sampai dengan 400KHz yang berada di atas rentang gelombang suara. Bentuk dari sensor ultrasonik yang digunakan dapat dilihat pada Gambar 8 dibawah ini:

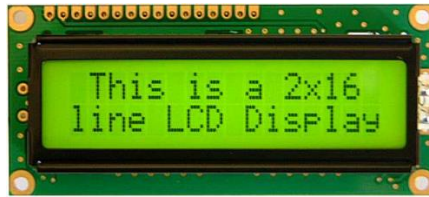


Gambar 8. Sensor Ultrasonik

f. LCD (*Liquid Crystal Display*)

LCD (*Liquid Crystal Display*) adalah suatu jenis media tampil yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan diberbagai bidang misalnya alal-alat elektronik seperti televisi, kalkulator, atau pun layar komputer. Pada postingan aplikasi LCD yang digunakan ialah LCD dot matrik dengan jumlah karakter 2 x 16. LCD sangat berfungsi sebagai penampil yang nantinya akan

digunakan untuk menampilkan status kerja alat. Bentuk dari LCD yang digunakan dapat dilihat pada gambar 9 dibawah ini:



Gambar 9. LCD

B. Desain Proyek Akhir

1. Tujuan Proyek Akhir

Tujuan dari perancangan sistem kontrol kendali pagar rumah dan garasi pintar ini untuk merancang dan mensimulasikan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas.

2. Metode Yang Digunakan

Metode yang digunakan penulis untuk mencapai tujuan adalah melalui studi literatur, dengan meninjau penelitian- penelitian terdahulu.

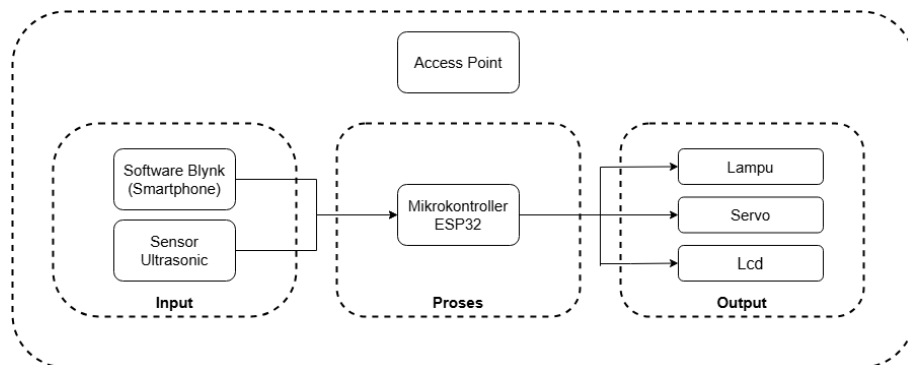
Tabel 1. Studi Literatur

| No | Nama Peneliti | Judul Penerbit | Hasil |
|----|--|--|--|
| 1 | Amalia Herlina, Moh. Irfan Syahbana, M. Adi Gunawan, Moh Miftahul Rizqi | Universitas Nurul Jadid, Probolinggo, Indonesia | Dari hasil pengujian NodeMCU ESP8266, Sistem Kendali Lampu secara keseluruhan dapat diketahui bahwa alat dapat berfungsi dengan baik. Sistem kendali berfungsi dengan baik pada jangkauan internet dengan jarak maksimal 25 meter. Sistem monitoring dapat membaca |

| No | Nama Peneliti | Judul Penerbit | Hasil |
|----|--|---|--|
| | | | tegangan dengan baik di mana saat input aplikasi <i>Blynk</i> 2.0 dimulai (on), aplikasi dapat membaca <i>output</i> tegangan sebesar 12V dan ditampilkan dilayar <i>smartphone</i> . |
| 2 | Cyntia Widiarsari1, Putra Abram Sianipar2, Muhammad Diono3 | Politeknik Caltex Riau, Teknologi Rekayasa Jaringan Telekomunik asi, | <i>Smartphone</i> berguna sebagai pengontrol NodeMCU yang terdapat pada modul <i>Wifi</i> ESP8266 melalui koneksi internet agar terhubung ke aplikasi <i>Blynk</i> . Relay digunakan sebagai <i>switch</i> yang akan memerintahkan motor untuk berputar ketika mendapat perintah dari NodeMCU. Perubahan kondisi relay akan menyebabkan motor DC 12V 200RPM berputar, sehingga pagar akan bergerak terbuka dan tertutup. Pergerakan motor servo juga menyebabkan pintu pagar terkunci setelah pagar tertutup. Agar sistem dapat bekerja dengan baik putaran per menit (RPM) dari motor membutuhkan <i>supply</i> tegangan sebesar 12V. Saat pagar terbuka, kecepatan motor 53RPM, sedangkan saat tertutup kecepatan motor 49RPM. Proses mengunci dan membuka pagar, motor servo membutuhkan waktu sekitar 2 detik. |

3. Desain Pembuatan Sistem

a. Blok diagram proses

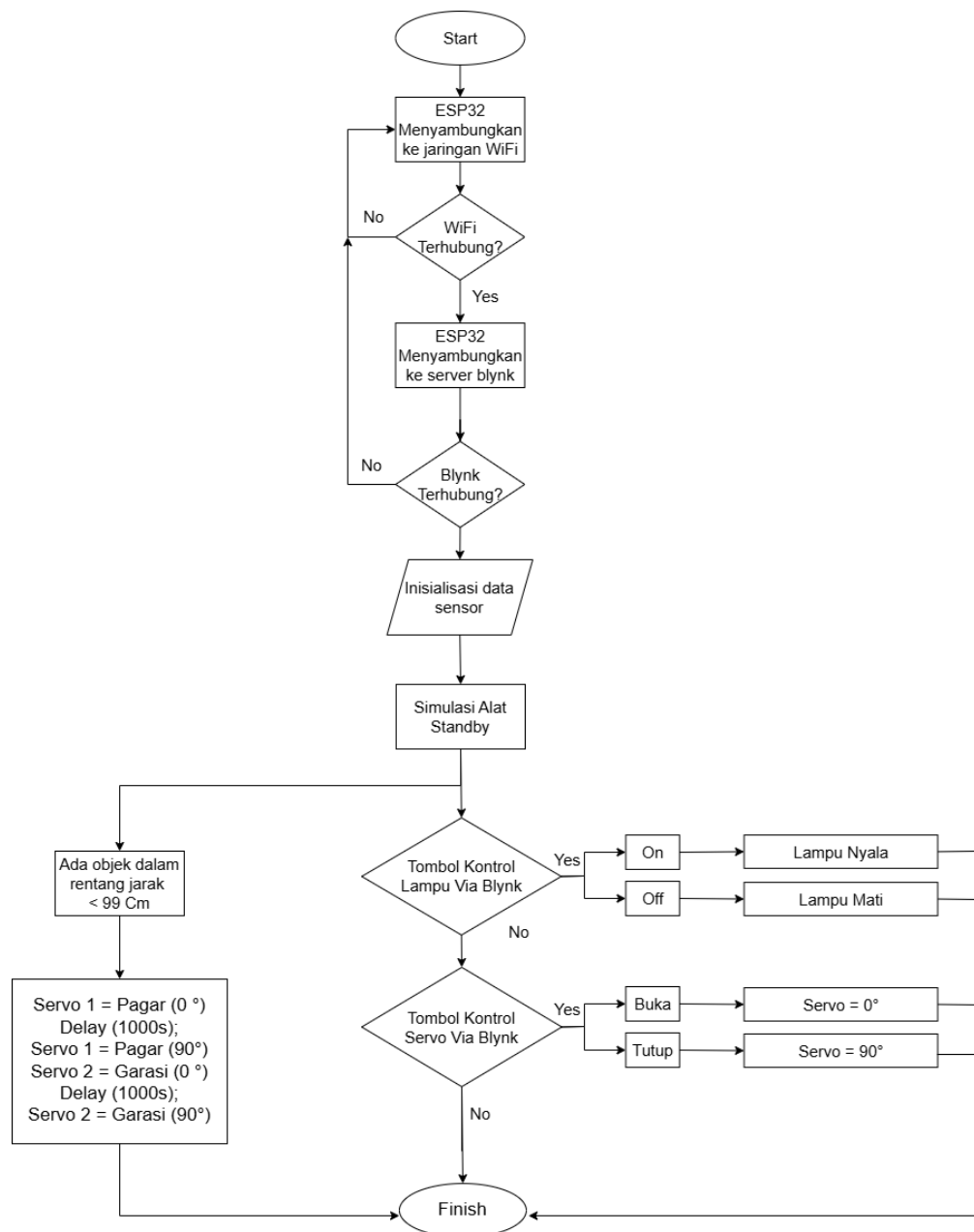


Gambar 10. Blok Diagram

- 1) *Access Point* berfungsi sebagai sumber jaringan internet untuk menghubungkan ESP32 dengan sever *Blynk*.
- 2) *Software Blynk (Smartphone)* berfungsi untuk mengirimkan perintah kepada ESP32 untuk mengaktifkan atau menonaktifkan Lampu dan Servo.
- 3) Sensor ultrasonik berfungsi sebagai pendeteksi objek yang akan men-*trigger* ESP32 untuk menggerakkan servo sebagai simulator untuk buka dan tutup pagar serta pintu garasi.
- 4) Mikrokontroler ESP32 berfungsi sebagai pengolah data yang didapat dari kontrol *Blynk* dan sensor ultrasonik.
- 5) Lampu berfungsi sebagai *output* rangkaian yang akan dikontrol melalui server *Blynk*.
- 6) Servo berfungsi sebagai *output* rangkaian yang akan dikontrol melalui server *Blynk* dan sensor ultrasonik.

b. Diagram aliran sistem (*Flowchart*)

Flowchart merupakan gambar yang memperlihatkan urutan langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataan pada sistem yang dirancang. *Flowchart* sistem dapat dijelaskan pada gambar di bawah ini:



Gambar 11. *Flowchart* sistem

Berikut adalah penjelasan alur kerja sistem kendali pagar rumah dan garasi pintar antara lain:

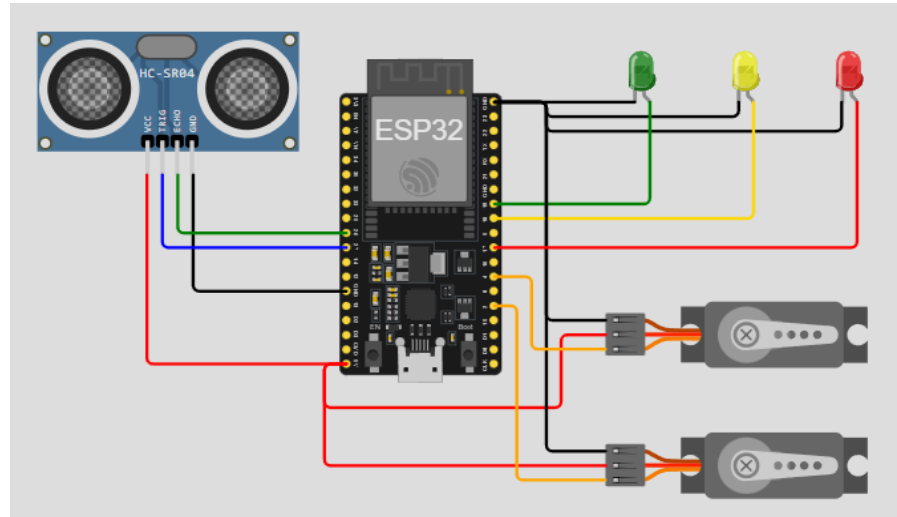
1. *Start* artinya adalah awal program dimulai atau dijalankan.
2. ESP32 menyambungkan ke jaringan *Wifi* artinya kondisi alat setelah dijalankan program yang dieksekusikan adalah program untuk menghubungkan ESP32 ke jaringan *Wifi*.
3. *Wifi* terhubung? artinya jika jaringan *Wifi* tidak terdeteksi oleh ESP32, program yang dieksekusikan diawal akan tetap diulang sampai jaringan terdeteksi oleh ESP32. Jika *Wifi* terhubung alat akan mengeksekusi program selanjutnya.
4. ESP32 menyambungkan ke server *Blynk*, artinya jika *Wifi* telah terhubung maka ESP32 akan terhubung ke *Blynk*.
5. *Blynk* terhubung? Artinya jika *Blynk* tidak terhubung, program yang dieksekusi adalah program untuk menghubungkan ESP32 ke jaringan *Wifi*.
6. Inisialisasi data sensor artinya adalah simulasi mulai membaca data dari sensor ultrasonik.
7. Simulasi alat *standby*, artinya simulasi telah *standby* dan siap menerima perintah dari aplikasi *Blynk*.
8. Tombol kontrol lampu *On/Off*, artinya adalah perintah yang diberikan oleh aplikasi *Blynk* berupa perintah *On/Off* dari tombol yang telah dibuat, jika perintahnya “*On*”, maka lampu akan menyala, jika perintah yang diberikan adalah “*Off*” maka lampu akan mati, jika tidak ada perintah, maka simulasi alat akan *standby*.

9. Tombol kontrol servo Buka/Tutup, artinya adalah perintah yang diberikan oleh aplikasi *Blynk* berupa perintah Buka/Tutup dari tombol yang telah dibuat, jika perintahnya “Buka”, maka servo akan rotasi ke posisi 0 derajat, jika perintah yang diberikan adalah “Tutup” maka servo akan kembali rotasi ke posisi 90 derajat, jika tidak ada perintah, maka simulasi alat akan *standby*.
10. Ada objek dalam jarak < 99 Cm artinya adalah sensor mendeteksi ada objek di depan sensor pada range kurang dari 99 Cm, pada kondisi ini, sensor akan mengirimkan perintah ke ESP32 untuk mentrigger servo 1 yang difungsikan sebagai pagar.
11. Servo 1 = *Open* (0°), *Delay* (1000s); Servo 1 = *Close* (90°), Servo 2 = *Open* (0°), *Delay* (1000s); Servo 2 = *Close* (90°) artinya adalah servo 1 atau pagar terbuka karena menerima perintah dari ESP32, setelah 1 menit, servo akan kembali ke posisi 90 derajat yang artinya pagar tertutup kembali. Setelah itu, servo 2 atau pintu garasi terbuka, setelah 1 menit servo akan kembali ke posisi 90 derajat yang artinya pintu garasi tertutup. Jika tidak ada perintah dari ESP32, simulasi alat akan *standby*.
12. Finish artinya semua program selesai dieksekusi atau dijalankan.

c. Skematik Rangkaian

Skema jaringan pintu garasi otomatis berbasis *Internet Of Things* menggunakan perangkat *Wifi* mikrokontroler NodeMCU ESP32, dimana perangkat tersebut akan memproses data dari sensor ultrasonik. Jaringan internet berfungsi untuk komunikasi alat dengan perangkat

Smartphone android yang sudah terinstal aplikasi *Blynk* untuk melakukan perintah buka dengan menampilkan jarak sensor ultrasonik. Untuk perancangan skematik rangkaian kontrol lampu dan buka tutup pagar serta pintu garasi di desain menggunakan *software* wokwi sebagai berikut:



Gambar 12. Skematik Rangkain

Pin I/O ESP32 yang digunakan:

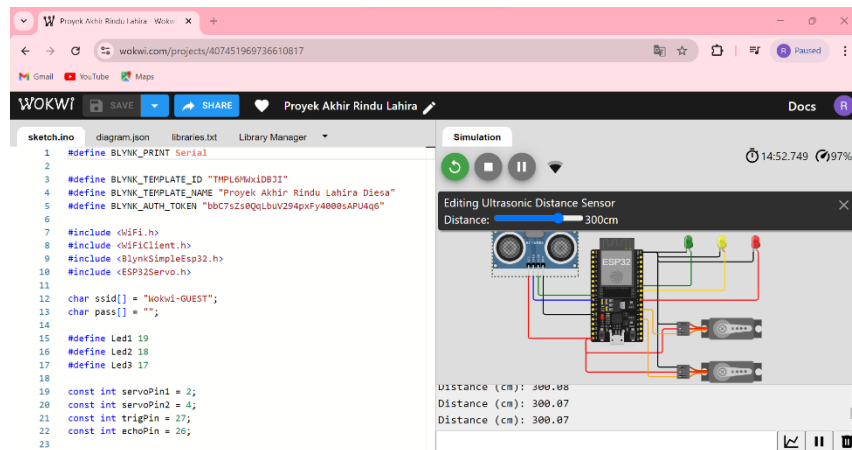
1. 5V terhubung ke pin pin VCC in motor servo, VCC sensor ultrasonik.
2. Pin GND terhubung ke pin katoda LED, pin GND motor servo dan pin GND sensor ultrasonik.
3. Pin D4 ESP32 terhubung ke pin kontrol PWM motor servo 1 yang difungsikan sebagai motor gerak pintu garasi.
4. Pin D2 ESP32 terhubung ke pin kontrol PWM motor servo 2 yang difungsikan sebagai motor gerak pagar rumah.
5. Pin D17 ESP32 terhubung ke pin anoda LED merah yang difungsikan sebagai lampu depan garasi.

6. Pin D18 ESP32 terhubung ke pin anoda LED kuning yang difungsikan sebagai lampu dalam garasi.
7. Pin D19 ESP32 terhubung ke pin anoda LED hijau yang difungsikan sebagai lampu samping garasi.
8. Pin D26 ESP32 terhubung ke pin *Echo* sensor ultrasonik.
9. Pin D27 ESP32 terhubung ke pin *Trig* sensor ultrasonik.

C. Deskripsi Proyek Akhir

1. Simulasi proyek akhir

Pada saat simulasi *standby* sensor akan mendeteksi objek sejauh 2 meter di depan sensor, Ketika ada objek yang berada di depan sensor dengan jarak kurang dari 1 meter maka pagar akan terbuka. Setelah objek masuk ke dalam pekarangan rumah pagar akan tertutup dengan *delay* waktu 1 menit, pintu garasi akan terbuka secara otomatis dan akan tertutup dengan *delay* 1 menit juga. Kondisi diatas terjadi pada saat server *Blynk* mengalami masalah, jika tidak terjadi masalah pada *Blynk*, pemilik rumah dapat mengontrol alat melalui *software Blynk*, yaitu dengan menekan tombol yang telah dibuat untuk mengontrol setiap alat yang ada pada simulasi atau implementasi alat. Jika pemilik rumah mengontrol alat menggunakan aplikasi *Blynk* ada beberapa tombol yang telah dibuat, diantaranya tombol untuk kontrol pagar, pintu garasi, dan lampu sekitar garasi.

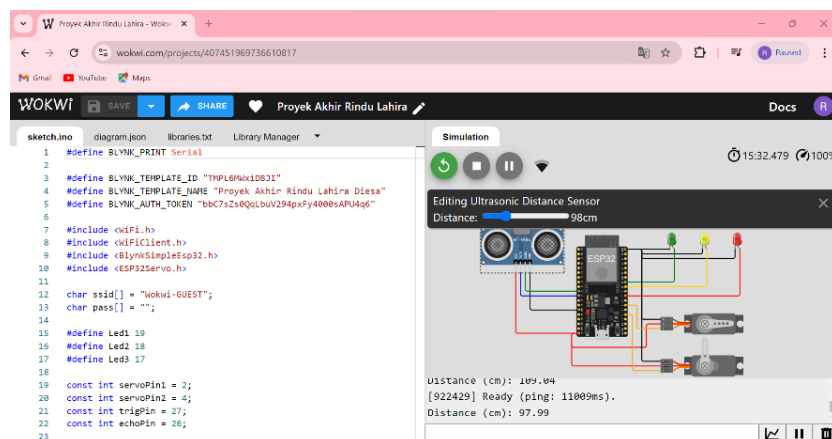


Gambar 13. Tampilan Program Simulasi

Pada gambar diatas dapat dilihat bahwa simulasi alat dalam kondisi *standby*, dimana motor servo pagar dan pintu garasi dalam keadaan tertutup, semua lampu dalam keadaan mati, serta sensor dalam keadaan *standby* untuk mendeteksi objek.

a. Tampilan simulasi pagar terbuka dengan sensor

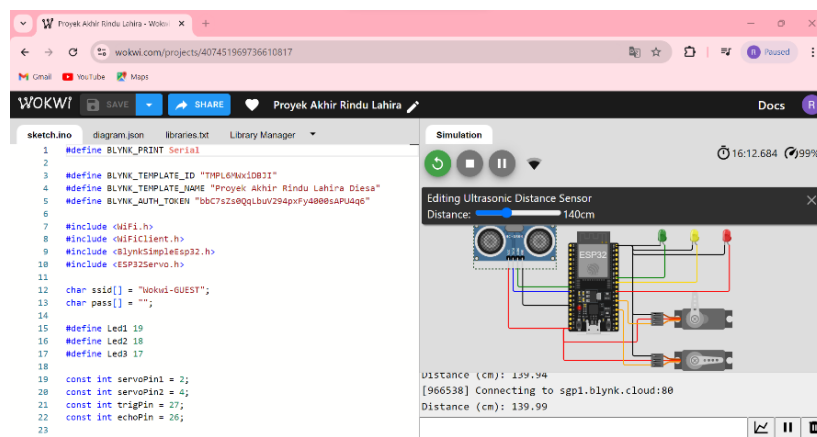
Pada saat sensor mendeteksi ada objek berupa mobil atau motor dalam jarak kurang dari 1 meter, maka pagar depan rumah akan terbuka secara otomatis yang dapat dilihat pada Gambar 14. di bawah ini.



Gambar 14. Tampilan Simulasi Pagar Terbuka Dengan Sensor

b. Tampilan simulasi pintu garasi terbuka dengan sensor

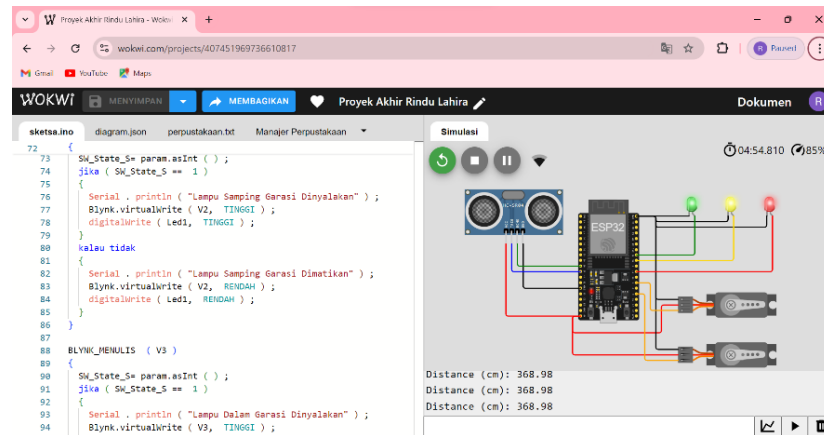
Pada Gambar 15. dapat dilihat saat sensor tidak mendeteksi ada objek lagi dalam jarak lebih dari 1 meter namun kurang dari 2 meter, dengan kata lain pada saat proses sensor menuju kondisi *standby* kembali, maka pagar depan rumah akan tertutup secara otomatis dan pintu garasi terbuka secara otomatis.



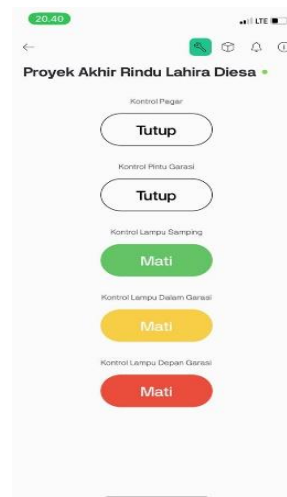
Gambar 15. Tampilan Simulasi Pintu Garasi Terbuka Dengan Sensor

c. Tampilan simulasi lampu dalam keadaan hidup dengan *Blynk*

Pada Gambar 16 dan Gambar 17 di bawah dapat dilihat bahwa simulasi lampu sedang menyala, perintah ini diberikan oleh pemilik rumah melalui aplikasi *Blynk*, pemilik rumah dapat menggunakan tombol yang telah dibuat pada aplikasi *Blynk* untuk menghidupkan dan mematikan lampu.



Gambar 16. Tampilan Simulasi Lampu Dalam Keadaan Hidup Dengan *Blynk*

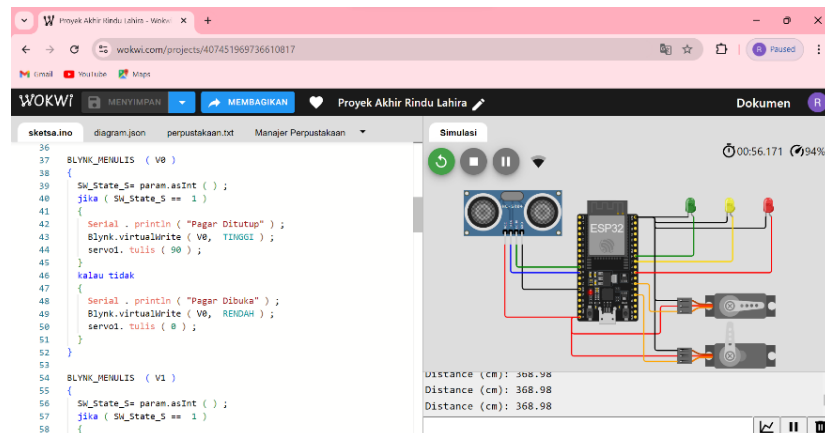


Gambar 17. Tampilan Simulasi Lampu Dalam Keadaan Hidup Dengan *Blynk*

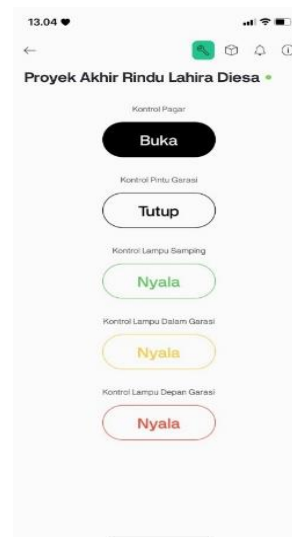
d. Tampilan simulasi pagar terbuka dengan *Blynk*

Pada Gambar 18 dan Gambar 19 di bawah ini adalah kondisi simulasi pada saat user atau pemilik rumah membuka pagar dengan menggunakan aplikasi *Blynk*, untuk gambar tampilan pada aplikasi terdapat pada Gambar 18. Dibawah ini, Dimana tombol untuk kontrol

pagar menampilkan kondisi “Tutup”, artinya jika tombol ditekan, maka pagar akan tertutup.



Gambar 19. Tampilan Simulasi Pagar Terbuka Dengan *Blynk*

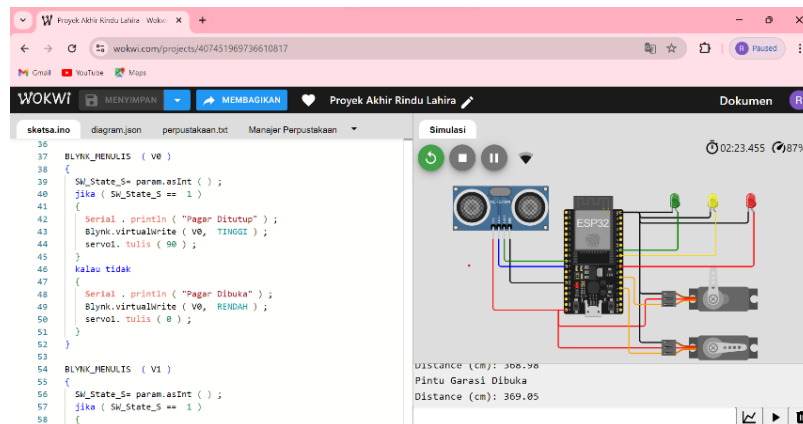


Gambar 18. Tampilan Kontrol Pintu Pagar Terbuka Pada *Blynk*

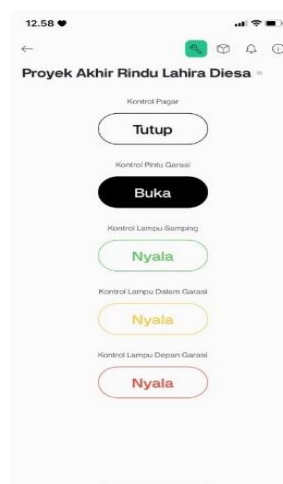
e. Tampilan simulasi pintu garasi terbuka dengan *Blynk*

Pada Gambar 20 dan Gambar 21 di bawah adalah kondisi simulasi pada saat user atau pemilik rumah membuka pintu garasi dengan menggunakan aplikasi *Blynk*, untuk gambar tampilan pada aplikasi terdapat pada gambar di bawah ini, Dimana tombol untuk kontrol pintu

garasi menampilkan kondisi “Tutup” artinya jika tombol ditekan, maka pintu garasi akan tertutup.



Gambar 21. Tampilan Simulasi Pintu Garasi Terbuka Dengan *Blynk*



Gambar 20. Tampilan Kontrol Pintu Garasi Terbuka Pada *Blynk*

D. Hasil dan Pembahasan

1. Analisis dan Evaluasi

Setelah melakukan proses perancangan, maka proses analisis dan evaluasi kinerja sistem, dengan informasi sebagai berikut:

- a. Pemilihan komponen untuk merancang proyek akhir yang penulis buat sudah memenuhi *spesifikasi* kebutuhan dan berjalan sesuai dengan yang diharapkan.
- b. Desain yang diperlukan seperti diagram aliran proses, blok diagram proses, *flowchart* sistem, skematik rangkaian dan desain simulasi sistem yang penulis buat sudah sesuai yang diharapkan.
- c. Penggunaan sensor ultrasonik pada simulasi yang penulis lakukan dapat bekerja secara otomatis tanpa harus bergantung sepenuhnya pada kontrol manual melalui aplikasi dan sudah berfungsi sesuai yang diharapkan.
- d. Pada simulasi di aplikasi *Blynk* untuk proses menjalankan program membutuhkan waktu respon 2-5 detik karena menggunakan jaringan internet, hal ini menunjukkan bahwa performa sistem sangat bergantung pada kualitas jaringan internet yang tersedia.

2. Pencapaian Dalam Proyek Akhir

Pencapaian dari proyek ini adalah *prototipe* telah berhasil dirancang dengan mikrokontroler NodeMCU ESP32 sebagai pengendali, android sebagai input perintah, *Blynk* sebagai media komunikasi, motor servo sebagai penggerak pintu garasi, sensor ultrasonik sebagai pendeteksi kendaraan.

Mencakup sebuah sistem kontrol membuka dan menutup pintu pagar berbasis internet. Sistem ini dirancang untuk mempermudah pemilik rumah dalam mengontrol akses pagar otomatis serta mengontrol hidup dan matinya

lampu pada garasi yang bertujuan untuk mengefesiensikan waktu lebih cepat.

3. Penggunaan Lebih Luas

Hasil proyek ini memiliki implikasi dalam kontek mempermudah dan mempersingkat waktu pemilik rumah untuk membuka pagar secara otomatis melalui sensor dan aplikasi pada android. Pada proyek ini sangat membutuhkan internet atau *wifi* yang tersambung pada program dan android. Oleh karena itu, harus dipastikan aplikasi terhubung dengan koneksi internet.

4. Rekomendasi untuk langkah selanjutnya

Untuk pengembangan selanjutnya, sistem perlu ditingkatkan agar dapat bekerja lebih baik dari pada saat ini agar sistem dapat terhubung dengan sistem keamanan lainnya, memperluas pengontrolan perangkat elektronik dalam rumah, serta meningkatkan algoritma agar lebih efisiensi dan mengurangi ketergantungan pada banyak aplikasi. Pemantauan dan pemeliharaan rutin harus menjadi agenda tetap untuk memastikan keandalan sistem.

BAB III

KESIMPULAN DAN REKOMENDASI

A. Kesimpulan

Proyek akhir ini “Perancangan Sistem Kendali Pagar Rumah Dan Garasi Pintar Menggunakan NodeMCU ESP32 Berbasis *Internet Of Things (Iot)*” berfungsi untuk mempermudah pemilik rumah membuka dan menutup pagar rumah serta pintu garasi dan mengontrol lampu dalam garasi.

Dengan jarak kendaraan kurang dari 1 meter sensor ultrasonik akan mendeteksi perintah untuk membuka pintu pagar, jika kendaraan sudah melewati pagar maka pintu garasi akan terbuka otomatis. Dengan jarak lebih dari 2 meter maka pagar akan tertutup secara otomatis dengan *delay* waktu 1 menit.

Program selanjutnya yaitu menghubungkan sistem dengan aplikasi *Blynk* menggunakan NodeMCU ESP32. Setelah melakukan pengujian tersebut diperoleh hasil bahwa pagar dapat diakses menggunakan aplikasi *Blynk* yang terdapat pada android atau ios dengan bantuan jaringan internet.

B. Rekomendasi

1. Tindakan perbaikan dalam pengembangan selanjutnya

Untuk meningkatkan kualitas sistem secara keseluruhan, perlu dilakukan pengujian yang lebih komperhensif untuk mengidentifikasi dan mengatasi kendala teknis. Selain itu, prioritas utama untuk mencegah resiko-resiko yang dapat mengancam keamanan rumah.

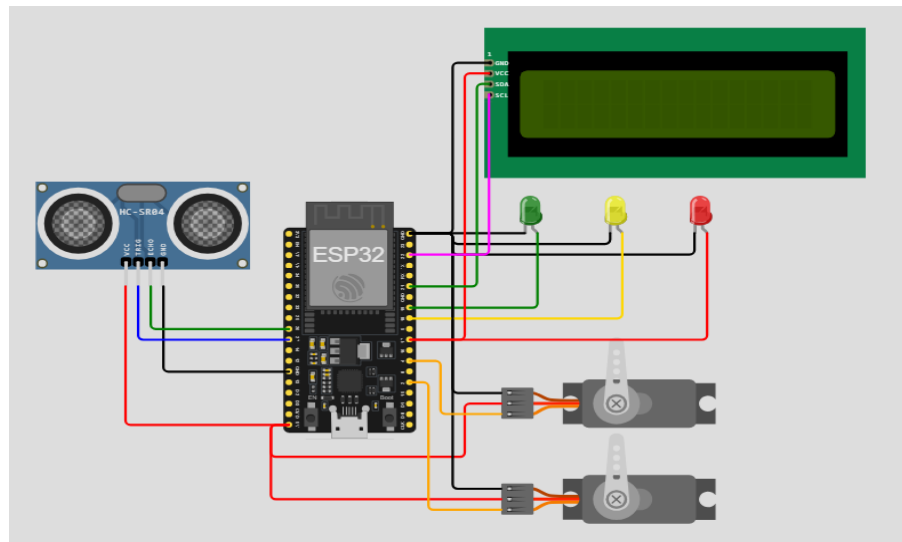
2. Penelitian lanjutan

- a. Sistem kendali pagar dan garasi dapat diintegrasikan dengan sistem keamanan rumah lainnya seperti CCTV, sensor gerak, dan alarm. Hal ini akan memberikan keamanan yang lebih *komprehensif* bagi pemilik rumah.
- b. Sistem dapat dikembangkan untuk mengirim notifikasi *real-time* ke *smartphone* pemilik rumah jika terdeteksi aktivitas mencurigakan di sekitar pagar atau garasi. Notifikasi ini dapat berupa pesan teks, email, atau pemberitahuan melalui aplikasi.
- c. Untuk mencegah peretasan, semua data yang dikirim antara perangkat dan aplikasi harus *dienkripsi* menambahkan *autentikasi* dua faktor akan meningkatkan keamanan akses ke sistem, sehingga hanya pengguna yang sah yang dapat mengontrol pagar dan garasi.
- d. Meskipun sistem dirancang untuk bekerja secara otomatis, penting untuk memiliki sistem backup manual yang dapat digunakan jika terjadi kegagalan sistem otomatis. Misalnya, dengan menyediakan tombol fisik untuk membuka dan menutup pagar atau garasi.

C. Rangkain Pengembangan

Skematik rangkaian pengembangan ini merupakan penyempurnaan dari sistem kendali pintu pagar dan pintu garasi pintar berbasis *Internet Of Things*. Penambahan komponen baru, yaitu LCD (*Liquid Crystal Display*), bertujuan untuk menampilkan informasi status sistem secara *real-time*, seperti jarak objek yang terdeteksi oleh sensor ultrasonik, status pintu garasi terbuka/tertutup.

Dengan adanya LCD pengguna dapat memantau sistem secara langsung tanpa harus mengandalkan aplikasi *Blynk* di *smartphone*. Hal ini dapat meningkatkan kemudahan penggunaan dan keandalan sistem. Untuk perancangan skematik rangkaian pengembangan dapat dilihat sebagai berikut:



Gambar 22. skematik rangkaian pengembangan

Pin I/O ESP32 yang digunakan:

1. Pin GND terhubung ke pin GND pada LCD yang difungsikan untuk menyamakan potensial listrik antara ESP32 dan LCD.
2. Pin D17 ESP32 terhubung ke pin VCC pada LCD yang difungsikan sebagai memberikan daya ke LCD untuk beroperasi.
3. Pin D21 ESP32 terhubung ke pin SDA pada LCD yang difungsikan sebagai mengirim data antara ESP32 dan LCD yang akan ditampilkan pada LCD.
4. Pin D22 ESP32 terhubung ke pin SCL pada LCD yang difungsikan sebagai mengirim sinyal *clock* yang menyinkronkan transmisi data antara ESP32 dan LCD.

D. Kesimpulan Rangkaian Pengembangan

Berdasarkan hasil rangkaian pengembangan dengan menambahkan LCD dapat disimpulkan bahwa meskipun LCD berhasil ditambahkan ke dalam rangkaian pengembangan sistem kendali pintu pagar dan pintu garasi pintar berbasis *Internet Of Things*, terdapat kendala teknis yang menyebabkan program tidak dapat berfungsi dengan baik.

Kendala ini disebabkan oleh error pada server Wokwi, yang merupakan platform simulasi online yang digunakan untuk menguji rangkaian dan program sebelum di implementasikan secara fisik.

E. Rekomendasi Rangkaian Pengembangan

1. Menggunakan platform simulasi lain selain Wokwi, seperti Proteus yang mungkin lebih stabil dan mampu menangani kompleksitas rangkaian dengan LCD.
2. Periksa kembali kode program untuk memastikan tidak ada kesalahan sintaks atau logika yang menyebabkan error.

Jika sistem berhasil di implementasikan, pertimbangkan untuk menambahkan fitur-fitur lain, seperti integrasi dengan sistem keamanan rumah (CCTV, sensor gerak, atau alarm).

DAFTAR PUSTAKA

- Fansyuri, M., & Yunita, D. (2024). Perancangan Internet of Things (IoT) untuk Sistem Kontrol Lampu Rumah.
- Herlina, A., Syahbana, M. I., Gunawan, M. A., & Rizqi, M. M. (2022). Sistem Kendali Lampu Berbasis Iot Menggunakan Aplikasi Blynk 2.0 Dengan Modul Nodemcu Esp8266. *INSANtek*, 3(2), 61-66.
- <http://digilib.unila.ac.id/64825/3/SKRIPSI%20TANPA%20BAB%20PEMBAHASAN.pdf>
- <https://www.empatpilar.com/pengertiansensorultrasonik/#:~:text=Sensor%20ultrasonik%20adalah>
- https://www.researchgate.net/publication/366142445_Sistem_Kendali_Lampu_Berbasis_Iot_Menggunakan_Aplikasi_Blynk_20_Dengan_Modul_Nodemcu_Esp8266
- https://www.researchgate.net/publication/366313989_Sistem_Kontrol_Otomatis_Pagar_Rumah_Berbasis_Internet_of_Things_IoT
- Muliadi, M., Imran, A., & Rasul, M. (2020). Pengembangan tempat sampah pintar menggunakan ESP32. *Jurnal Media Elektrik*, 17(2), 73-79.
- Pengertian Smartphone: Menurut Ahli, Sejarah, OS, dan Fiturnya! (rifqimulyawan.com)
- Widiasari, C., Sianipar, P. A., & Diono, M. (2022). Sistem Kontrol Otomatis Pagar Rumah Berbasis Internet of Things (IoT). *Jurnal ELEMENTER (Elektro dan Mesin Terapan)*, 8(2), 162-174.
- Widiasari, C., Sianipar, P. A., & Diono, M. (2022). Sistem Kontrol Otomatis Pagar Rumah Berbasis Internet of Things (IoT). *Jurnal ELEMENTER (Elektro dan Mesin Terapan)*, 8(2), 162-174.
- Widiasari, C., javascript:void(0) Sianipar, P. A., & Diono, M. (2022). Sistem Kontrol Otomatis Pagar Rumah Berbasis Internet of Things (IoT). *Jurnal ELEMENTER (Elektro dan Mesin Terapan)*, 8(2), 162-174.

LAMPIRAN

1. Source Code sebelum penambahan LCD

```
#define BLYNK_PRINT Serial

#define BLYNK_TEMPLATE_ID "TMPL6MWxiDBJI"
#define BLYNK_TEMPLATE_NAME "Proyek Akhir Rindu Lahira Diesa"
#define BLYNK_AUTH_TOKEN "bbC7sZs0QqLbuV294pxFy4000sAPU4q6"

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <ESP32Servo.h>

char ssid[] = "Wokwi-GUEST";
char pass[] = "";

//inisialisasi LED
#define Led1 19 // LED hijau lampu samping garasi
#define Led2 18 // LED kuning lampu dalam garasi
#define Led3 17 // LED merah lampu depan garasi

//inisialisasi servo
const int servoPin1 = 2;
const int servoPin2 = 4;
const int trigPin = 27;
const int echoPin = 26;

#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701

long duration;
float distanceCm;
float distanceInch;

int SW_State_S = 0;
BlynkTimer timer;
Servo servo1;
Servo servo2;

BLYNK_WRITE (V0)
{
  SW_State_S= param.asInt();
  if (SW_State_S == 1)
  {
    Serial.println("Pagar Ditutup"); // perintah pagar ditutup
  }
}
```

```

    Blynk.virtualWrite(V0, HIGH);
    servo1.write(90);
}
else
{
    Serial.println("Pagar Dibuka"); // perintah pagar dibuka
    Blynk.virtualWrite(V0, LOW);
    servo1.write(0);
}
}

BLYNK_WRITE (V1)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {
        Serial.println("Pintu Garasi Ditutup"); // perintah pintu
garasi ditutup
        Blynk.virtualWrite(V1, HIGH);
        servo2.write(90);
    }
    else
    {
        Serial.println("Pintu Garasi Dibuka"); // perintah pintu
garasi dibuka
        Blynk.virtualWrite(V1, LOW);
        servo2.write(0);
    }
}

BLYNK_WRITE (V2)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {
        Serial.println("Lampu Samping Garasi Dinyalakan");
        // perintah lampu samping garasi dinyalakan
        Blynk.virtualWrite(V2, HIGH);
        digitalWrite(Led1, HIGH);
    }
    else
    {
        Serial.println("Lampu Samping Garasi Dimatikan"); //lampu
samping garasi dimatikan
        Blynk.virtualWrite(V2, LOW);
        digitalWrite(Led1, LOW);
    }
}

```

```

}

BLYNK_WRITE (V3)
{
  SW_State_S= param.asInt();
  if (SW_State_S == 1)
  {
    Serial.println("Lampu Dalam Garasi Dinyalakan"); // perintah
    lampu dalam garasi dinyalakan
    Blynk.virtualWrite(V3, HIGH);
    digitalWrite(Led2, HIGH);
  }
  else
  {
    Serial.println("Lampu Dalam Garasi Dimatikan"); // perintah
    lampu dalam garasi dimatikan
    Blynk.virtualWrite(V3, LOW);
    digitalWrite(Led2, LOW);
  }
}

BLYNK_WRITE (V4)
{
  SW_State_S= param.asInt();
  if (SW_State_S == 1)
  {
    Serial.println("Lampu Depan Garasi Dinyalakan"); // perintah
    lampu depan garasi dinyalakan
    Blynk.virtualWrite(V4, HIGH);
    digitalWrite(Led3, HIGH);
  }
  else
  {
    Serial.println("Lampu Depan Garasi Dimatikan"); // perintah
    lampu depan garasi dimatikan
    Blynk.virtualWrite(V4, LOW);
    digitalWrite(Led3, LOW);
  }
}

// inisialisasi LED
void setup(){
  Serial.begin(9600);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(Led1, OUTPUT);
  pinMode(Led2, OUTPUT);
}

```

```

    pinMode(Led3, OUTPUT);

//inisialisasi servo
    servo1.attach(servoPin1);
    servo2.attach(servoPin2);
    servo1.write(90);
    servo2.write(90);
}

void loop() {
    Blynk.run();
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    duration = pulseIn(echoPin, HIGH);
    distanceCm = duration * SOUND_SPEED/2;
    distanceInch = distanceCm * CM_TO_INCH;
    Serial.print("Distance (cm): ");
    Serial.println(distanceCm);
    delay(1000);

    if(distanceCm < 99)
    {
        servo1.write(0);
        delay(10000);
        servo1.write(90);
        servo2.write(90);
    }
    else if (distanceCm >= 100 && distanceCm <= 200) //perintah
    pembacaan sensor ultrasonic
    {
        servo2.write(0);
        delay(10000);
        servo1.write(90);
        servo2.write(90);
    }
}

```

LAMPIRAN 2

2. Source Code rangkaian pengembangan

```
#define BLYNK_PRINT Serial

#define BLYNK_TEMPLATE_ID "TMPL6MWxiDBJI"
#define BLYNK_TEMPLATE_NAME "Proyek Akhir Rindu Lahir Diesa"
#define BLYNK_AUTH_TOKEN "bbC7sZs0QqLbuV294pxFy4000sAPU4q6"

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <ESP32Servo.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

char ssid[] = "Wokwi-GUEST";
char pass[] = "";

#define Led1 19
#define Led2 18
#define Led3 17

const int servoPin1 = 2;
const int servoPin2 = 4;
const int trigPin = 27;
const int echoPin = 26;

#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701

long duration;
float distanceCm;
float distanceInch;

int SW_State_S = 0;
BlynkTimer timer;
Servo servo1;
Servo servo2;

// Inisialisasi LCD (alamat I2C 0x27, ukuran 16x2)
LiquidCrystal_I2C lcd(0x27, 16, 2);

BLYNK_WRITE (V0)
{
```



```

SW_State_S= param.asInt();
if (SW_State_S == 1)
{
    Serial.println("Pagar Ditutup");
    i2c_lcd1602_write_string(lcd_info,"pagar Ditutup");
    Blynk.virtualWrite(V0, HIGH);
    servo1.write(90);
    lcd.setCursor(0, 0);
    lcd.print("Pagar Ditutup ");
}
else
{
    Serial.println("Pagar Dibuka");
    i2c_lcd1602_write_string (lcd_info, "Pagar Dibuka");
    Blynk.virtualWrite(V0, LOW);
    servo1.write(0);
    lcd.setCursor(0, 0);
    lcd.print("Pagar Dibuka ");
}
}

BLYNK_WRITE (V1)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {
        Serial.println("Pintu Garasi Ditutup");
        Blynk.virtualWrite(V1, HIGH);
        servo2.write(90);
        lcd.setCursor(0, 1);
        lcd.print("Pintu Garasi ");
    }
    else
    {
        Serial.println("Pintu Garasi Dibuka");
        Blynk.virtualWrite(V1, LOW);
        servo2.write(0);
        lcd.setCursor(0, 1);
        lcd.print("Pintu Garasi ");
    }
}

BLYNK_WRITE (V2)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {

```

```

        Serial.println("Lampu Samping Garasi Dinyalakan");
        Blynk.virtualWrite(V2, HIGH);
        digitalWrite(Led1, HIGH);
        lcd.setCursor(0, 0);
        lcd.print("Lampu Samping  ");
    }
    else

    {
        Serial.println("Lampu Samping Garasi Dimatikan");
        Blynk.virtualWrite(V2, LOW);
        digitalWrite(Led1, LOW);
        lcd.setCursor(0, 0);
        lcd.print("Lampu Samping  ");
    }
}

BLYNK_WRITE (V3)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {
        Serial.println("Lampu Dalam Garasi Dinyalakan");
        Blynk.virtualWrite(V3, HIGH);
        digitalWrite(Led2, HIGH);
        lcd.setCursor(0, 1);
        lcd.print("Lampu Dalam  ");
    }
    else
    {
        Serial.println("Lampu Dalam Garasi Dimatikan");
        Blynk.virtualWrite(V3, LOW);
        digitalWrite(Led2, LOW);
        lcd.setCursor(0, 1);
        lcd.print("Lampu Dalam  ");
    }
}

BLYNK_WRITE (V4)
{
    SW_State_S= param.asInt();
    if (SW_State_S == 1)
    {
        Serial.println("Lampu Depan Garasi Dinyalakan");
        Blynk.virtualWrite(V4, HIGH);
        digitalWrite(Led3, HIGH);
        lcd.setCursor(0, 1);
    }
}

```

```

        lcd.print("Lampu Depan      ");
    }
    else
    {
        Serial.println("Lampu Depan Garasi Dimatikan");
        Blynk.virtualWrite(V4, LOW);
        digitalWrite(Led3, LOW);
        lcd.setCursor(0, 1);
        lcd.print("Lampu Depan      ");
    }
}

void setup(){
    Serial.begin(9600);
    Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(Led1, OUTPUT);
    pinMode(Led2, OUTPUT);
    pinMode(Led3, OUTPUT);
    servo1.attach(servoPin1);
    servo2.attach(servoPin2);
    servo1.write(90);
    servo2.write(90);

    // Inisialisasi LCD dengan parameter jumlah kolom dan baris
    lcd.begin(16, 2); // 16 kolom dan 2 baris
    lcd.backlight();
    lcd.clear();
}

void loop() {
    Blynk.run();
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    duration = pulseIn(echoPin, HIGH);
    distanceCm = duration * SOUND_SPEED / 2;
    distanceInch = distanceCm * CM_TO_INCH;
    Serial.print("Distance (cm): ");
    Serial.println(distanceCm);
    delay(1000);

    // Menampilkan jarak di LCD

```

```
lcd.setCursor(0, 0);  
lcd.print("Distance: ");  
lcd.print(distanceCm);  
lcd.print(" cm");  
  
// Kontrol berdasarkan jarak  
if(distanceCm < 99)  
{  
    servo1.write(0);  
    delay(10000);  
    servo1.write(90);  
    servo2.write(90);  
}  
else if (distanceCm >= 100 && distanceCm <= 200)  
{  
    servo2.write(0);  
    delay(10000);  
    servo1.write(90);  
    servo2.write(90);  
}  
}
```