

RANCANG BANGUN PINTU GERBANG MENGUNAKAN WEB SERVER BERBASIS IOT

Farhan Juliansyah¹, Yonal Supit², Andi Muhammad Islah^{*3}

^{1,2,3}Program Studi Sistem Komputer, STMIK Catur sakti Kendari

e-mail:¹farhanjuliansyahms@gmail.com, ²yonalsupit@gmail.com ³Andiislah@gmail.com

Penelitian ini bertujuan untuk merancang dan membangun sistem pintu gerbang berbasis Internet of Things (IoT) menggunakan Web Server. Sistem ini memanfaatkan ESP32 sebagai mikrokontroler utama yang terhubung ke jaringan WiFi, sehingga memungkinkan pengguna untuk mengendalikan pintu gerbang melalui antarmuka web. Pengguna dapat membuka dan menutup gerbang secara nirkabel melalui perintah yang dikirimkan dari halaman web yang diakses melalui browser. Motor stepper digunakan untuk menggerakkan pintu, dan sensor ultrasonik digunakan untuk mendeteksi hambatan saat gerbang sedang ditutup, untuk menghindari kerusakan atau kecelakaan. Sistem ini juga menyediakan notifikasi status gerbang secara real-time. Hasil pengujian menunjukkan bahwa sistem berfungsi dengan baik dalam mengontrol pintu gerbang.

Kata kunci: IoT, web server, pintu gerbang otomatis, ESP32, motor stepper, sensor ultrasonik.

I. PENDAHULUAN

Pagar merupakan bagian utama dalam sebuah rumah dari segi keamanan. pagar konvensional untuk membuka dan menutupnya harus dilakukan secara manual oleh pemilik rumah. Pada kondisi ini hal tersebut sangat tidak efektif dan tidak efisien, karna pemilik rumah harus turun dari kendaraan untuk membuka pagar. Permasalahan ini yang mendasari adanya ide untuk pembuatan pintu gerbang menggunakan Web Server yang memanfaatkan internet of things (IoT).

Selama ini secara umum proses buka tutup pintu gerbang masih dilakukan secara manual dengan cara pemilik rumah harus menggeser pintu gerbang[1] Dengan adanya teknologi yang berkembang saat ini membuat manusia ingin melakukan sesuatu dengan mudah, salah satunya yaitu dengan pemanfaatan IoT (*Internet of Things*), IoT merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dan konektivitas internet secara terus menerus dengan adanya teknologi yang berkembang saat ini membuat manusia ingin melakukan sesuatu dengan mudah, salah satunya yaitu dengan pemanfaatan IoT (*Internet of Things*)[2]. Dengan dibuatnya pintu gerbang ini dapat mempermudah pemilik rumah dalam memasukan dan mengeluarkan kendaraannya

menuju gerbang tanpa harus turun dari kendaraan, dan memberikan keamanan lebih maka dapat dilakukan dengan cara pengendalian melalui *smartphone* untuk membuka pintu gerbang dan menutup melalui jaringan *wifi*[3].

Wifi merupakan kependekan dari *Wireless Fidelity* yaitu sebuah media penghantar komunikasi data tanpa kabel yang biasa digunakan untuk komunikasi atau transfer data dengan kemampuan yang sangat cepat[4]. Pengendalian ini akan dilakukan menggunakan Web Server sebagai input untuk membuka pintu gerbang dengan memanfaatkan jaringan *wifi* untuk pengendaliannya dan pintu menutup sehingga dapat dilakukan dimanapun selama masih berada dalam jangkauan sinyal jaringan *wifi* dan mikrokontroler arduino NodeMcu Esp32 sebagai pemrosesan serta motor servo sebagai penggerak pintu gerbang mendorong dan menarik pintu gerbang agar bisa membuka dan menutup[5].Maka penelitian ini mengambil judul “Rancang Bangun Pintu Gerbang Menggunakan Web Server Berbasis IoT”.

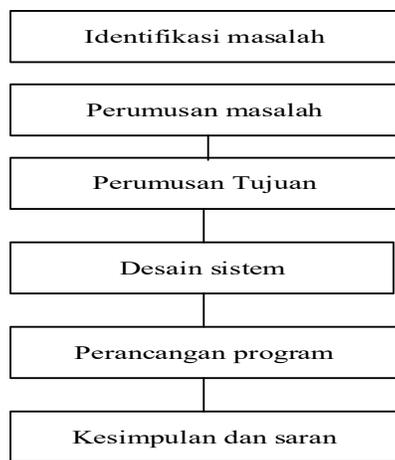
II. METODE PENELITIAN

A. Tahapan Penelitian

Untuk penyusunan penelitian ini, maka perlu adanya tahapan metode penelitian. Tahapan ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas, berikut gambaran tahap tahap yang harus dilakukan demi tercapainya tujuan dari penelitian ini.

Pada gambar 1 tahapan dari penelitian langkah pertama kali adalah:

- Identifikasi Masalah : Langkah pertama ini berfokus pada pengidentifikasian masalah atau kebutuhan yang ada. Ini adalah tahap di mana semua permasalahan yang memerlukan solusi dikumpulkan dan dianalisis.
- Perumusan Masalah : Setelah masalah diidentifikasi, langkah berikutnya adalah merumuskan masalah secara jelas dan rinci. Tahap ini bertujuan untuk mendefinisikan masalah agar dapat dipahami dengan lebih baik dan lebih mudah dicari solusinya.



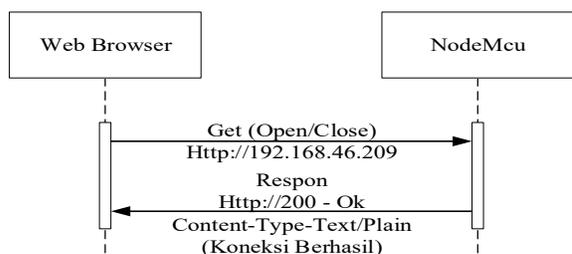
Gambar 1 Tahapan Penelitian

- Perumusan Tujuan : Pada tahap ini, tujuan dari solusi yang diinginkan dirumuskan. Ini adalah pernyataan tentang apa yang ingin dicapai melalui solusi yang akan dirancang.
- Desain Sistem : Langkah ini melibatkan pembuatan desain atau rancangan sistem yang akan digunakan untuk menyelesaikan masalah dan mencapai tujuan yang telah ditetapkan. Desain sistem biasanya mencakup model, skema, dan arsitektur dari solusi yang akan diimplementasikan.
- Perancangan Program : Tahap ini adalah pembuatan program atau aplikasi berdasarkan desain sistem yang telah dibuat. Perancangan program bertujuan untuk mengimplementasikan solusi yang konkret.
- Kesimpulan dan Saran : Setelah seluruh proses berjalan, dilakukan evaluasi dan penarikan kesimpulan. Selain itu, saran-saran untuk perbaikan atau pengembangan lebih lanjut juga diberikan pada tahap ini.

B. Perancangan Sistem

Dalam perancangan sistem ini menggunakan metode *Unified Modeling Language* (UML), adalah bahasa pemodelan visual yang digunakan untuk merancang. UML menyediakan notasi grafis standar untuk menggambarkan berbagai aspek sistem, seperti struktur, fungsi, dan interaksi antar komponen.

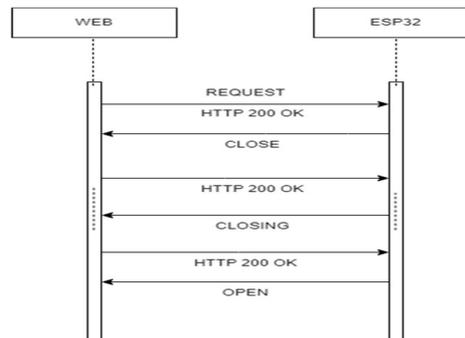
1. Perancangan Perangkat Lunak Ke Nodemcu



Gambar 2. Sequence Diagram Web browser ke NodeMcu

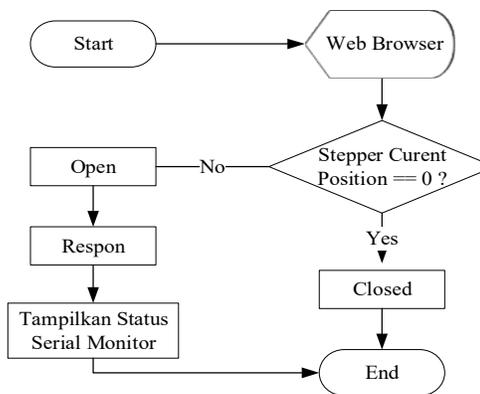
Diagram urutan pada gambar diatas menggambarkan proses komunikasi antara Browser web dan NodeMCU menggunakan permintaan HTTP GET. Ketika web

browser mengirimkan permintaan ke NodeMCU, NodeMCU merespons dengan kode status HTTP 200, yang menunjukkan bahwa permintaan tersebut telah berhasil diproses dan koneksi berhasil terjalin.



Gambar 3 Sequence Diagram Aplikasi dan Nodemcu Dalam Merespon Status.

Gambar 3 menunjukkan Ketika *Client* mengirimkan permintaan HTTP GET ke URL `"/status"`, NodeMCU menerima permintaan tersebut dan mengevaluasi status pagar saat ini. Status pagar ditentukan berdasarkan apakah pagar sedang menutup (*isClosing*), apakah motor stepper berada di posisi 0 (pagar tertutup), atau di posisi lain (pagar terbuka). NodeMCU kemudian membentuk teks respons yang menunjukkan status pagar (`"CLOSING"`, `"CLOSED"`, atau `"OPEN"`) dan menambahkan pesan tambahan jika ada objek yang terdeteksi menghalangi pagar. Respon HTTP 200 OK dikirimkan kembali ke *Client* dengan pesan status pagar, dan informasi ini juga dicatat di Serial Monitor dengan pesan seperti `"Status request received."` dan `"Gate status: [status]"`.



Gambar 4 Flowchart Status Gerbang

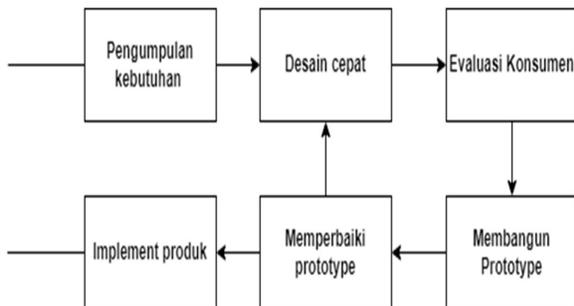
Flowchart di atas menggambarkan alur logika program untuk menjalankan program buka/tutup pintu gerbang. Proses dimulai dengan menerima permintaan dari *Client*. Jika posisi adalah 0, status gerbang ditetapkan sebagai `"CLOSE"`. Jika posisi stepper adalah 1, maka gerbang ditetapkan sebagai `"OPEN"`. Apabila *client* memberikan permintaan untuk membuka pagar (*Open*) maka status pada stepper motor akan berubah menjadi 1, begitupun ketika *client* memberikan perintah untuk menutup pagar (*Close*) maka status stepper motor akan berubah menjadi 0

dan hal tersebut juga ditampilkan di Serial Monitor. Proses ini kemudian selesai.

Flowchart ini memberikan gambaran yang jelas tentang bagaimana program menentukan dan merespons gerbang berdasarkan kondisi internal sistem.

C. Pengembangan Sistem

Dalam perancangan system ini menggunakan, metode *prototype* digunakan dalam system untuk memfasilitasi kalsifikasi kebutuhan pengguna, menguji desain secara dini.



Gambar 5 Metode *prototype*

D. Pengumpulan Kebutuhan

Pada tahap ini, fokus utama untuk memahami terhadap kebutuhan perangkat keras dan perangkat lunak yang akan dikembangkan. Setelah mengumpulkan informasi dari berbagai sumber, yang telah di analisis evaluasi secara cermat untuk memastikan pemahaman yang tepat dan keselarasan dengan tujuan perancangan sistem.

Adapun perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan untuk membuat rancang bangun pintu gerbang menggunakan Web Server berbasis iot yaitu:

1. Kebutuhan Perangkat keras

Dalam perancangan ini dibutuhkan komponen perangkat keras seperti berikut:

Tabel 1 Perangkat keras

No	Komponen	Jumlah
1	Motor stepper 28BYJ-48	1
2	NodeMCU ESP32	1
3	ULN2003	1
5	Sensor ultra sonic	1

2. Kebutuhan Perangkat Lunak

Dalam perancangan ini dibutuhkan perangkat lunak seperti berikut:

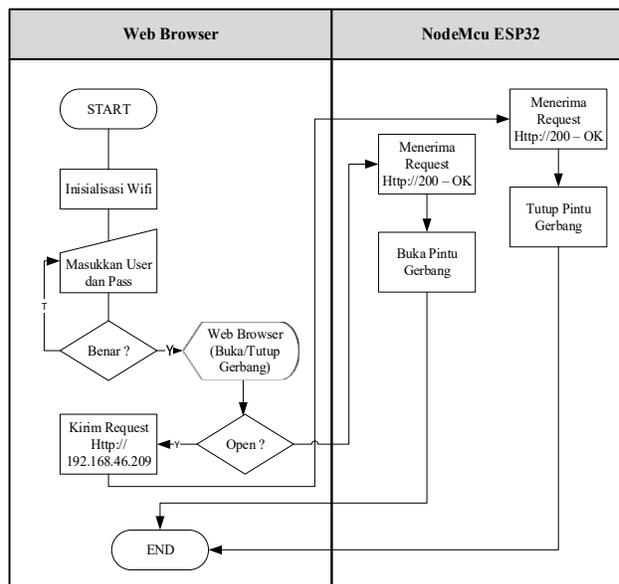
Tabel 1 perangkat lunak

No	Perangkat Lunak	Deskripsi
1	ESP32 IDE	Untuk memprogram ESP32
2	Fritzing	Untuk skema perangkat
3	Web Browser	Untuk memonitoring pintu gerbang

3. Desain Cepat

Tahapan selanjutnya ialah presentasi atau menggambarkan model sistem yang akan dikembangkan seperti proses dengan perancangan menggunakan *Unified Modeling Language (UML)*.

Flowmap Sistem Kendali Pintu Gerbang

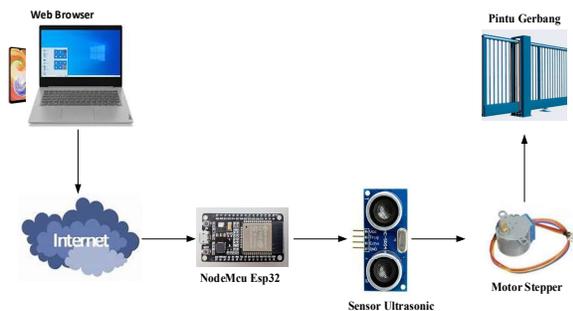


Gambar 6 Flowmap Sistem Kontrol

E. Pembuatan Prototype

Setelah desain quick desain disetujui oleh user, tahap selanjutnya yaitu pembangunan *prototype*. Dalam tahap ini, *Prototype* yang dibangun dengan sistem rancangan sementara kemudian evaluasi terhadap customer apakah sesuai dengan yang diinginkan atau masih perlu evaluasi kembali.

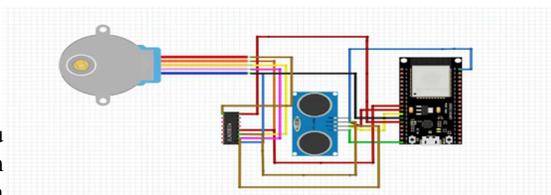
F. Arsitektur Sistem



Gambar 7. Arsitektur Sistem

1. Perancangan Perangkat Keras

Perangkat keras atau dalam bahasa Inggris hardware, akan digunakan dalam perancangan sistem rancang bangun pintu gerbang berbasis IoT NodeMCU ESP32 sebagai kontrol utama dalam pintu gerbang dengan sebuah Browser web yang telah dirancang. NodeMCU ESP32 juga sebagai penghubung ke jaringan internet menggunakan wifi.



Gambar 8. Perancangan Perangkat Keras

Adapun untuk penghubungan antara komponen-komponen sebagai berikut:

Tabel 3 Uln 2003A ke MCU ESP32

Uln 2003A	MCU ESP32
VCC	5V
GND	GND
IN 1	GPIO12
IN 2	GPIO14
IN 3	GPIO27
IN 4	GPIO26

Tabel 3. menunjukkan koneksi pin antara driver ULN2003A dan ESP32 MCU yang digunakan untuk mengendalikan motor stepper. Berikut penjelasan dari tiap baris pada tabel tersebut:

- VC VCC ke 5V:
Pin VCC pada ULN2003A terhubung ke pin 5V pada ESP32, yang menyediakan tegangan untuk driver ULN2003A agar dapat beroperasi. Tegangan 5V ini digunakan untuk memberikan daya pada motor stepper.
- GND ke GND:
Pin GND pada ULN2003A dihubungkan ke pin GND pada ESP32 untuk melengkapi rangkaian listrik, memastikan bahwa keduanya memiliki referensi tegangan yang sama.
- IN1 ke GPIO12:
Pin IN1 dari ULN2003A dihubungkan ke pin GPIO12 pada ESP32, yang digunakan untuk mengontrol salah satu kumparan motor stepper.
- IN2 ke GPIO14:
Pin IN2 dari ULN2003A dihubungkan ke pin GPIO14 pada ESP32, yang mengontrol kumparan motor stepper kedua.
- IN3 ke GPIO27:
Pin IN3 dari ULN2003A dihubungkan ke pin GPIO27 pada ESP32, yang mengontrol kumparan motor stepper ketiga.
- IN4 ke GPIO26:
Pin IN4 dari ULN2003A dihubungkan ke pin GPIO26 pada ESP32, yang mengontrol kumparan motor stepper keempat.

Dengan menggunakan empat pin dari ESP32 (GPIO12, GPIO14, GPIO27, dan GPIO26), motor stepper dapat dikendalikan melalui ULN2003A dengan mengatur urutan sinyal ke kumparan motor stepper, sehingga menghasilkan gerakan yang diinginkan.

Tabel 4 Motor stepper 28BYJ-48 ke ULN 2003A

Motor stepper 28BYJ-48	ULN 2003A
PIN Merah	5V
PIN Biru	Out 1
PIN Pink	Out 2
PIN Kuning	Out 3
PIN Orange	Out 4

Tabel 4 menunjukkan koneksi antara motor stepper 28BYJ-48 dengan ULN2003A, yang merupakan driver

untuk motor stepper tersebut. Berikut penjelasan dari setiap kolom dalam tabel:

- Motor Stepper 28BYJ-48: Motor stepper ini memiliki lima pin yang terhubung ke kumparan internalnya. Motor ini biasa digunakan untuk aplikasi yang membutuhkan pergerakan presisi dalam sudut-sudut kecil.
 - PIN Merah (5V): Pin ini harus terhubung ke suplai daya 5V untuk motor stepper.
 - PIN Biru (Out 1): Pin ini terhubung ke output 1 dari ULN2003A, yang akan mengontrol kumparan pertama motor.
 - PIN Pink (Out 2): Terhubung ke output 2 dari ULN2003A, untuk kumparan kedua.
 - PIN Kuning (Out 3): Terhubung ke output 3 dari ULN2003A, untuk kumparan ketiga.
 - PIN Orange (Out 4): Terhubung ke output 4 dari ULN2003A, untuk kumparan keempat.
- ULN2003A: Ini adalah IC driver yang digunakan untuk menggerakkan motor stepper dengan arus lebih tinggi dari yang bisa disediakan oleh microcontroller seperti ESP32. ULN2003A memiliki output yang sesuai untuk menggerakkan setiap kumparan pada motor stepper. Pin Out 1 hingga Out 4 terhubung ke kumparan motor stepper untuk mengontrol gerakannya sesuai urutan yang diperlukan.

Koneksi ini memungkinkan ESP32 atau microcontroller lain untuk mengendalikan motor stepper dengan sinyal kontrol, sementara ULN2003A menyediakan daya yang cukup untuk menggerakkan motor.

Tabel 5 Ultra Sonic ke MCU ESP32

	MCU ESP32
VCC	3V3
GND	GND
TRIGGER	GPIO PIN 12
ECHO	GPIO PIN 14

Tabel 5 menunjukkan koneksi antara MCU ESP32 dengan sensor ultrasonik (seperti HC-SR04), yang digunakan untuk mengukur jarak dengan memanfaatkan gelombang ultrasonik. Berikut penjelasan dari setiap koneksi:

- MCU ESP32:
 - VCC (3V3): Pin VCC pada ESP32 terhubung ke pin daya sensor ultrasonik. Karena ESP32 bekerja pada tegangan 3.3V, VCC-nya memberikan daya 3.3V ke sensor ultrasonik.
 - GND (Ground): Pin GND pada ESP32 terhubung ke pin ground sensor ultrasonik untuk memastikan adanya sirkuit tertutup.
- Sensor Ultrasonik:
 - TRIGGER (GPIO PIN 12): Pin TRIGGER pada sensor ultrasonik dihubungkan ke pin GPIO 12 pada ESP32. Ini adalah pin yang digunakan untuk memicu sensor mengirimkan sinyal ultrasonik.
 - ECHO (GPIO PIN 14): Pin ECHO pada sensor ultrasonik dihubungkan ke pin GPIO 14 pada ESP32. Pin ini menerima sinyal yang dipantulkan dari objek yang berada di depan sensor dan menghitung waktu antara

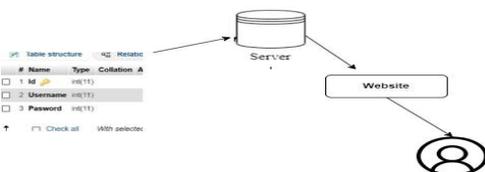
pengiriman dan penerimaan sinyal untuk mengukur jarak.

Secara keseluruhan, koneksi ini memungkinkan ESP32 untuk mengontrol sensor ultrasonik dan membaca data jarak dari objek di sekitarnya, yang kemudian dapat digunakan untuk berbagai aplikasi seperti deteksi objek atau pengukuran jarak dalam sistem IoT.

G. Perancangan Perangkat Lunak

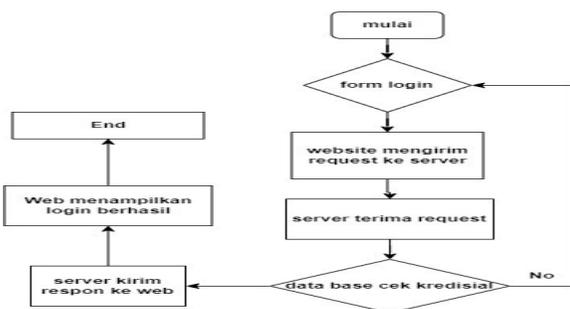
Berikut adalah struktur atau rancangan perangkat lunak yang akan digunakan:

1. Desain Alur Data Base



Gambar 9 Desain Alur Program Database (Database Design)

Tahap desain database adalah proses perencanaan dan pengorganisasian struktur database untuk menyimpan dan mengelola data secara efisien. Berikut adalah perencanaan struktur database yang akan digunakan. Berikut adalah langkah-langkah dalam tahap perencanaan desain database dalam bentuk flowchart:



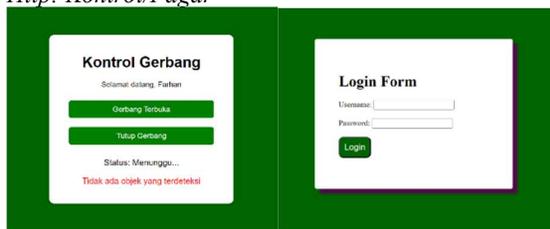
Gambar 10. Flochart Web Server Ke Database

Flowchart tersebut menggambarkan alur proses login dimana pengguna memasukkan kredensial, server memverifikasinya melalui database, dan jika valid, pengguna diberi akses, jika tidak, pengguna diminta untuk mengulangi login.

2. Desain Web UI

Dalam perancangan “Rancang Bangun Pintu Gerbang Menggunakan Web Server Berbasis IoT” terdapat rancangan Perangkat lunak yang nantinya akan digunakan untuk memonitoring gerbang yang terhubung. Sistem dituliskan dalam bahasa PHP pada HTML dan CSS.

Http: Kontrol/Pagar



Gambar 11. Perancangan Perangkat lunak

Pada perancangan sistem ini, peneliti melakukan input dari Browser kendali yang berada pada Web Server, input yang digunakan berupa Button (Buka dan Tutup Pagar).

4. Notifikasi

Pada Web ini terdapat pesan notifikasi dari sensor ultrasonik ke website dilakukan dengan cara sensor mengukur jarak dan mengirim data ke mikrokontroler (ESP32). Jika jarak yang terdeteksi berada dalam batas kritis, mikrokontroler mengirimkan informasi ini ke web server melalui koneksi Wi-Fi. Web server kemudian memproses data dan mengirimkan notifikasi secara real-time ke website pengguna, biasanya menggunakan teknologi seperti WebSocket Hasilnya, pengguna menerima notifikasi langsung di website tanpa perlu melakukan refresh.

Status: Menunggu...

Tidak ada objek yang terdeteksi

Gambar 12. Notifikasi

H. Pengujian

Pengujian sistem kontrol alat eltronik terkait dengan spesifikasi terstruktur dan Browser secara keseluruhan. Proses pengujian perlu dilakukan untuk memastikan bahwa Browser yang telah di buat sudah benar, sesuai dengan karakteristik yang di tetapkan dan tidak ada kesalahan.

Pengujian NodeMCU Pada bagian ini dilakukan pengujian Mikrokontroler NodeMCU 1.0 apakah NodeMCU 1.0 dapat berkerja dengan baik seperti memproses datadigital yang diinginkan, lalu dapat mengkoneksikan kedalam wifi yang diinginkan

Pengujian, pada bagian ini dilakukan pengujian apakah Browser mampu untuk mengontrol buka atau tutup Pintu Gerbang. selain itu apakah pengontrolan melalui browser dapat dilakukan di Pc/Laptop dan di handphone.

I. Implementasi

Setelah melakukan tahapan pengujian, implementasi pintu gerbang menggunakan Web Server berbabsis IoT dilakukan dengan menyelaraskan pintu gerbang untuk memberikan pengguna kemampuan untuk mengontrolnya melalui Browser yang terhubung. Proses implementasi melibatkan penyiapan perangkat keras dan lunak yang terintegrasi dengan baik.

J. Kesimpulan

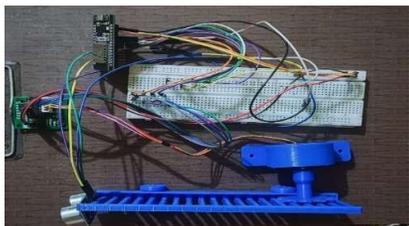
Dalam mengimplentasikan sistem pintu gerbang berbasis IoT penting untuk menjalani serangkaian tahap yang melibatkan pengujian dan optimalisasi kinerja sebelum memasuki fase implementasi.

III. HASIL DAN PEMBAHASAN

A. Implementasi Perangkat Keras

Implementasi perangkat keras meliputi pembuatan rangkaian yang diperlukan untuk sistem. Langkah-langkah yang dilakukan mencakup menghubungkan sensor ultrasonik dan stepper motor ke ESP32 menggunakan

kabel jumper sesuai dengan pin yang telah ditentukan pada tahap perancangan perangkat keras.



Gambar 13 Rangkaian Perangkat Keras

Pada gambar 13 adalah rangkaian perangkat keras yang sudah dihubungkan satu sama lain sesuai dengan pin yang ditentukan.



Gambar 14 Rangkaian Sensor Ultrasonic

Gambar 14 adalah sensor ultrasonik yang sudah dihubungkan dengan NodeMCU ESP32 agar dapat mengirim data berupa jarak objek dengan menggunakan pin trig dan echo pada sensor ultrasonik untuk mengukur jarak objek di depan gerbang yang kemudian dikirimkan ke NodeMCU ESP32.



Gambar 15 Rangkaian Stepper Motor

Gambar 15 adalah stepper motor yang sudah dihubungkan dengan ESP32 agar dapat mengontrol gerakan motor. Stepper motor ini dihubungkan ke driver motor yang bertugas menerima sinyal dari ESP32 untuk mengatur pergerakan langkah motor. Setiap langkah motor dikontrol melalui pin kontrol yang terhubung ke driver motor dari ESP32, memungkinkan motor untuk membuka atau menutup gerbang dengan presisi tinggi.

B. Implementasi Perangkat Lunak

1. Perangkat Lunak Esp32

```

1 #include <WiFi.h>
2 #include <ESPAsyncWebServer.h>
3 #include <AccelStepper.h>
4 #include <NewPing.h>
5
6 // Definisi pin untuk stepper motor
7 #define IN1 34
8 #define IN2 27
9 #define IN3 26
10 #define IN4 25
11
12 // Definisi pin untuk sensor ultrasonik
13 #define TRIGGERS_PIN 9
14 #define ECHO_PIN 18
15 #define MAX_DISTANCE 5
16
17 // Parameter sistem
18 #define STEPS_PER_REV 100 // Sesuai dengan parameter mekanis sistem Anda
19
20 // Replace with your network credentials
21 const char* ssid = "Ayam";
22 const char* password = "12345678";
23
24 // Create AsyncWebServer object on port 80
25 AsyncWebServer server(80);
    
```

Gambar 16 Library, Set SSID, Set HTTP Dan Verifikasi Perangkat.

Berikut adalah program yang telah disesuaikan dengan penggunaan sensor ultrasonik dan stepper motor dengan NodeMCU ESP32. Program ini menginisialisasi beberapa library yang diperlukan untuk menghubungkan ESP32 ke jaringan WiFi, melakukan permintaan HTTP, dan mengontrol gerakan stepper motor serta membaca data dari sensor ultrasonik. SSID dan kata sandi WiFi didefinisikan untuk menghubungkan ESP32 ke jaringan, serta URL server untuk mengirim data. Kode ini juga mengatur pin untuk mengontrol stepper motor dan sensor ultrasonik.

```

void setup() {
  // Start Serial
  Serial.begin(115200);

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
    Serial.println("Alamat IP: ");
    Serial.println(WiFi.localIP());
  }
  Serial.println("Connected to WiFi");

  // Konfigurasi stepper motor
  stepper.setMaxSpeed(500); // Sesuaikan kecepatan maksimal sesuai dengan motor stepper
  stepper.setAcceleration(100); // Sesuaikan akselerasi sesuai dengan motor stepper

  // Route for opening the gate
  server.on("/open", HTTP_GET, [] (AsyncWebServerRequest *request) {
    Serial.println("Open command received.");
    stepper.moveTo(30 * STEPS_PER_REV); // Gerakkan motor sejauh 30 mm (3 cm)
    isClosing = false;
    obstacleDetected = false; // Reset obstacleDetected saat membuka gerbang
  });
}
    
```

Gambar 17 void setup

Pada gambar 17 di atas adalah fungsi setup () yang menginisialisasi berbagai komponen yang digunakan dalam sistem. Pertama, komunikasi serial diatur dengan baud rate 115200 untuk debugging. Kemudian, fungsi connect WiFi () dipanggil untuk menghubungkan ESP32 ke jaringan WiFi. Selanjutnya, sensor ultrasonik

diinisialisasi dengan mengatur pin trig sebagai output dan pin echo sebagai input. Stepper motor juga diinisialisasi dengan mengatur jumlah langkah per revolusi dan pin yang digunakan.

```

void loop() {
  // Update stepper motor state
  stepper.run();

  // Cek jarak dari sensor ultrasonik
  static unsigned long lastPingTime = 0;
  unsigned long currentTime = millis();
  if (currentTime - lastPingTime > 1000) { // Pengecekan jarak setiap 1 detik
    lastPingTime = currentTime;
    unsigned int distance = sonar.ping_cm();
    Serial.print("Distance: ");
    Serial.println(distance);
    Serial.println(" cm");
  }

  if (isClosing) {
    if (distance > 0 && distance <= 2) {
      stepper.stop();
      obstacleDetected = true;
      wasClosingBeforeObstacle = true; // Menyimpan status penutupan sebelum mendeteksi objek
      Serial.println("Object detected within 4 cm, stopping gate.");
    } else if (obstacleDetected && (distance == 0 || distance > 4)) {
      obstacleDetected = false;
      if (wasClosingBeforeObstacle) {
        // Jika objek tidak terdeteksi dan pintu sebelumnya dalam proses penutupan
        stepper.moveTo(0); // Lanjutkan menutup
      }
    }
  }
}
    
```

Gambar 18 void loop ()

Gambar 18 menampilkan kode program fungsi loop () yang berfungsi untuk menjalankan logika utama yang diulang terus menerus. Pertama, jika koneksi WiFi terputus, fungsi connect WiFi () dipanggil untuk menghubungkan kembali. Kemudian, sensor ultrasonik digunakan untuk mengukur jarak objek dengan mengirimkan pulsa melalui pin trig dan mengukur durasi pulsa yang kembali melalui pin echo, yang kemudian dikonversi menjadi jarak objek dalam cm. Jika jarak yang terukur kurang dari atau sama dengan 4 cm, maka fungsi open Gate () dipanggil untuk membuka gerbang dengan menggerakkan stepper motor. Jika tidak, fungsi close Gate () dipanggil untuk menutup gerbang. Status gerbang dikirim ke server untuk memperbarui informasi.

C. Pengujian

Pengujian dilakukan untuk mengevaluasi apakah sistem yang telah dibangun sesuai dan berfungsi dengan baik. Pengujian mencakup semua komponen utama, mulai dari, stepper motor, dan sensor ultrasonik, serta pengujian sistem secara keseluruhan.

1. Pengujian Sensor Ultra Sonic

Pengujian ini bertujuan untuk mengevaluasi akurasi sensor ultrasonik pada jarak 1 cm hingga 4 cm. Setiap jarak diuji 10 kali untuk memeriksa konsistensi pengukuran.

Tabel 6 Pengujian jarak objek

No	Jarak Aktual(cm)	Jarak Terukur(cm)	Error(%)	Keterangan	Tegangan (V)
1	1	1	0	Akurat	2.0 V
2	2	2	0	Akurat	2.0 V
3	3	3	0	Akurat	2.0 V
4	4	4	0	Akurat	2.0 V
5	4	3.5	12.5	Kurang Dari Batas	4.2- 4.4 V
6	4	4.5	12.5	Melebihi Batas	4.2- 4.4 V
7	4	4	0	Akurat	2.0 V
8	4	4	0	Akurat	2.0 V
9	4	4	0	Akurat	2.0 V
10	4	3.5	12.5	Kurang Dari Batas	4.2- 4.4 V

Dari hasil percobaan pada tabel 6 Pengujian sensor ultrasonik menunjukkan bahwa sensor bekerja dengan akurat pada sebagian besar kasus, namun terdapat beberapa pengukuran yang kurang dari atau melebihi batas yang diharapkan seperti nomor 5 dengan jarak aktual 4 cm dan jarak terukur 3.5 cm

Kesimpulan dari tabel pengukuran jarak aktual, jarak terukur, error, dan tegangan yang digunakan menunjukkan beberapa hal penting terkait kinerja sensor ultrasonik:

- Tegangan Stabil: Pada jarak pendek (1-3 cm) dan jarak tepat 4 cm, tegangan sensor ultrasonik stabil pada 2.0 V, dengan error 0% yang menandakan hasil akurat.
- Perubahan Tegangan pada Error: Pada jarak 4 cm yang mengalami selisih dengan jarak terukur (baik lebih besar atau lebih kecil), tegangan naik menjadi sekitar 4.2-4.4 V. Hal ini terjadi saat ada ketidakakuratan pengukuran, dengan error sekitar 12.5%.
- Korelasi antara Tegangan dan Error: Ketidakakuratan pengukuran jarak terjadi ketika tegangan sensor naik. Ini menunjukkan bahwa perubahan tegangan pada sensor dapat memengaruhi hasil pengukuran, terutama ketika jarak terukur tidak sesuai dengan jarak aktual.

Dari kesimpulan ini, bisa dikatakan bahwa tegangan sensor berperan penting dalam menentukan akurasi pengukuran, terutama pada jarak batas tertentu.

2. Pengujian Stepper Motor

Berikut adalah tabel yang menggambarkan hasil pengujian kecepatan dan akselerasi stepper motor serta pengamatan ketika melebihi batas yang telah ditentukan:

Tabel 7 Pengujian Langkah Stepper motor

No	Parameter	Nilai	Pengamatan	Tegangan (V)
1	Kecepatan (RPM)	500	Motor bekerja dengan stabil pada kecepatan ini	5 V
2	Akselerasi (RPM/s)	100	Akselerasi menghasilkan transisi yang mulus hingga kecepatan maksimum.	5 V
3	Kecepatan > 500 RPM	1000	Motor mengalami penurunan performa, getaran, dan motor berhenti secara tiba-tiba	5,5 V
4	Akselerasi > 100 RPM/s	500	Motor mengalami getaran yang signifikan, ketidakstabilan, dan kesalahan posisi.	5,5 V

Dari pengujian yang dilakukan, didapatkan beberapa hasil penting terkait kecepatan motor dan akselerasi motor stepper berdasarkan tegangan yang diberikan:

1. Kecepatan 500 RPM dengan Tegangan 5 V: Motor bekerja stabil pada kecepatan ini. Ini menunjukkan bahwa pada kecepatan yang relatif standar (500 RPM), tegangan 5 V sudah cukup untuk menjaga kinerja motor tetap stabil tanpa masalah.
2. Akselerasi 100 RPM/s dengan Tegangan 5 V: Transisi kecepatan dari kondisi diam hingga mencapai kecepatan maksimum berjalan mulus. Ini menunjukkan bahwa akselerasi yang moderat tidak menyebabkan gangguan performa pada motor saat menggunakan tegangan 5 V.
3. Kecepatan Melebihi 500 RPM dengan Tegangan 5,5 V: Ketika kecepatan melebihi 500 RPM (hingga 1000 RPM), terjadi penurunan performa motor, getaran yang lebih kuat, dan akhirnya motor berhenti secara tiba-tiba. Ini menandakan bahwa kecepatan yang terlalu tinggi, meskipun diberikan tegangan yang lebih besar (5,5 V), tidak dapat dipertahankan oleh sistem, menyebabkan ketidakstabilan.
4. Akselerasi Lebih dari 100 RPM/s dengan Tegangan 5,5 V: Pada akselerasi yang lebih tinggi (500 RPM/s), motor mengalami getaran signifikan dan kesalahan posisi. Meskipun tegangan dinaikkan menjadi 5,5 V, motor tetap tidak dapat mengatasi akselerasi yang cepat, yang menyebabkan ketidakstabilan dan masalah performa.

Kesimpulan: Tegangan standar 5 V cukup efektif pada kecepatan dan akselerasi yang moderat. Namun, ketika kecepatan dan akselerasi meningkat, motor membutuhkan lebih banyak tegangan (5,5 V), meskipun ini tidak selalu menjamin stabilitas. Ketidakstabilan mulai terjadi pada

kecepatan yang lebih tinggi dan akselerasi yang lebih cepat, bahkan dengan tegangan tambahan.

3. Pengujian Koneksi

Hasil dari pengujian NodeMCU ESP32 dapat terkoneksi dengan database.

```
ets Jul 29 2019 12:21:46
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1298
load:0x40078000,len:13872
load:0x40080400,len:4
ho 8 tail 4 room 4
load:0x40080404,len:3048
entry 0x40080590
Connecting to WiFi...
Alamat IP: 192.168.17.209
Connected to WiFi
***** 0 ****
```

Gambar 19 Koneksi NodeMCU ESP32

Pada gambar 19 menunjukkan NodeMCU ESP32 dapat terhubung ke Wi-Fi dan berkomunikasi dengan web server menggunakan protokol HTTP untuk mengirim atau menerima data, untuk kontrol jarak jauh seperti membuka atau menutup pintu gerbang berbasis IoT.

4. Pengujian Sistem web

Pengujian sistem dilakukan untuk menguji keseluruhan sistem yang telah dibuat dengan cara menguji kontrol dan monitoring pagar di lokasi tertentu. Pada Browser, menampilkan data status pagar, kontrol pagar, dan notifikasi sensor ultrasonik. Pengujian sistem web dapat dilihat pada tabel di bawah ini.

Tabel 8 Pengujian System Web

NO	Fitur	Kredensial yang benar	Hasil	Keterangan
1	Login	Tombol "Open Gate" dan "Close Gate" berfungsi dengan baik	Berhasil login dan diarahkan ke halaman kontrol	Sistem menerima dan memverifikasi dengan benar
2	Kontrol	Status pagar ditampilkan dan diperbarui secara real-time	Tombol berfungsi dan mengirim perintah ke sistem	Sistem merespons perintah kontrol pagar dengan cepat dan akurat
3	Status pagar	Status pagar ditampilkan dan diperbarui secara real-time	Status ditampilkan dan diperbarui setiap 5 detik	Informasi status pagar selalu up-to-date
4	Notifikasi	Notifikasi ditampilkan saat sensor ultrasonik mendeteksi objek	Notifikasi muncul di antarmuka web	Sistem notifikasi bekerja dengan baik

Tabel 8 merangkum pengujian fitur utama dari sistem web yang digunakan untuk mengontrol dan memonitor pagar. Hasil pengujian menunjukkan bahwa semua fitur berfungsi sesuai dengan yang diharapkan, memberikan kontrol yang efisien dan informasi yang diperlukan kepada pengguna.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari penelitian ini didapatkan kesimpulan bahwa sistem yang dirancang dan dibangun berjalan sesuai dengan yang diharapkan, yaitu merancang dan membangun sistem kontrol gerbang berbasis *Internet of Things* (IoT) menggunakan ESP32, stepper motor, dan sensor ultrasonik. Sistem ini memungkinkan kontrol jarak jauh melalui Browser web yang terhubung ke server, serta memberikan informasi real-time tentang status gerbang.

Dari hasil pengujian, diketahui bahwa sistem berfungsi dengan baik, namun terdapat beberapa kelemahan pada kinerja motor stepper pada kecepatan tinggi dan saat mendeteksi objek menggunakan sensor ultrasonik. Error pada sensor ultrasonik dalam program yang digunakan bervariasi dengan rata-rata error sebesar 12.5%. Selain itu, sistem juga berhasil menampilkan status gerbang secara real-time melalui web server, serta mendeteksi dan merespons halangan yang terdeteksi di jalur gerbang.

Dengan demikian, penelitian ini telah berhasil mencapai tujuannya, meskipun masih terdapat ruang untuk perbaikan dalam hal optimasi kinerja motor stepper dan akurasi deteksi sensor ultrasonik.

B. Saran

Selain memastikan kehandalan sensor *ultrasonic*, saran untuk penelitian selanjutnya yaitu Menambahkan keamanan. Hal ini dapat meningkatkan fungsionalitas sistem untuk pintu gerbang

DAFTAR PUSTAKA

- [1] C. Widiyari, P. Abram Sianipar, and M. Diono, "Sistem Kontrol Otomatis Pagar Rumah Berbasis Internet of Things (IoT)," *Jurnal Elektro dan Mesin Terapan*, vol. 8, no. 2, pp. 162–174, Nov. 2022, doi: 10.35143/elementer.v8i2.5748.
- [2] A. Junaidi, "INTERNET OF THINGS, SEJARAH, TEKNOLOGI DAN PENERAPANNYA : REVIEW," Bandung, 2015.
- [3] T. Sukendar, M. Saputro, A. Ishaq, and A. Sumbaryadi, "Perancangan Pintu Gerbang Otomatis Menggunakan Arduino Dengan Koneksi Bluetooth Berbasis Android," *Jurnal Teknologi Informatika dan Komputer*, vol. 8, pp. 164–175, Sep. 2022, doi: 10.37012/jtik.v8i2.1131.
- [4] A. F. Silvia, E. Haritman, and Y. Muladi, "RANCANG BANGUN AKSES KONTROL PINTU GERBANG BERBASIS ARDUINO DAN ANDROID," 2014. [Online]. Available: <http://jurnal.upi.edu/>
- [5] S. Adinata, K. Budi Utomo, T. Informasi, and P. Negeri Samarinda, "JIP (Jurnal Informatika Polinema) Halaman| 83 RANCANG BANGUN MONITORING PAGAR OTOMATIS BERBASIS IOT DENGAN KONTROL DAN KEAMANAN," Nov. 2024.