



PROTOTYPE KEAMANAN GERBANG GESER BERBASIS ESP32 MENGUNAKAN TOMBOL SWITCH DAN NOTIFIKASI REAL TIME

ESP32-BASED SLIDING GATE SECURITY PROTOTYPE USING SWITCH BUTTONS AND REAL-TIME NOTIFICATION

Relly Dwi Septian Lubis¹, Efri Yandani², Dwi Winarti³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dharmas Indonesia

Email: 2102011025@undhari.ac.id

Article Info

Article history :

Received : 16-07-2025

Revised : 18-07-2025

Accepted: 20-07-2025

Pulished : 22-07-2025

Abstract

Security is an important aspect that requires special attention. One of the main access points that needs to be secured is the gate, which must be protected to prevent unwanted actions. This study aims to design and build a prototype of a sliding gate security system based on the ESP32 microcontroller that can provide real-time notifications. The system uses a switch button (limit switch) to detect the gate's condition (open or closed), a buzzer as a sound indicator, and an ESP32 microcontroller connected to a WiFi network. When the gate is opened, the system activates the buzzer and automatically sends a notification via Telegram to the user. Test results show that the system works effectively in detecting gate status and sending notifications quickly. This system is expected to enhance overall security and provide a greater sense of safety for users.

Keyword: Security, Sliding Gate, ESP32

Abstrak

Keamanan merupakan aspek penting yang perlu mendapatkan perhatian khusus. Salah satu akses utama yang perlu diamankan adalah gerbang, sehingga perlu dijaga agar terhindar dari tindakan yang tidak diinginkan. Penelitian ini bertujuan untuk merancang dan membangun prototype sistem keamanan gerbang geser berbasis ESP32 yang mampu memberikan notifikasi secara real-time. Sistem ini menggunakan tombol switch (limit switch) untuk mendeteksi kondisi gerbang (terbuka atau tertutup), buzzer sebagai indikator suara, dan ESP32 sebagai mikrokontroler utama yang terhubung ke jaringan WiFi. Ketika gerbang dibuka, sistem akan mengaktifkan buzzer dan secara otomatis mengirimkan notifikasi melalui Telegram kepada pengguna. Hasil pengujian menunjukkan bahwa sistem ini mampu bekerja dengan baik dalam mendeteksi status gerbang dan mengirimkan notifikasi dengan cepat. Dengan adanya sistem ini, diharapkan dapat meningkatkan sistem keamanan secara lebih luas dan memberikan rasa aman bagi penggunanya.

Kata Kunci: Keamanan, Gerbang Geser, ESP32

PENDAHULUAN

Keamanan merupakan kebutuhan dasar manusia yang sangat penting dalam menunjang kehidupan yang aman dan nyaman. Dalam skala yang lebih luas, keamanan tidak hanya berkaitan dengan perlindungan terhadap individu, tetapi juga terhadap properti dan aset lainnya. Seiring dengan perkembangan zaman, tantangan terhadap keamanan semakin kompleks. Data statistik menunjukkan bahwa angka kejahatan seperti pencurian, perampokan, dan tindakan kriminal lainnya mengalami peningkatan di berbagai belahan dunia. Kondisi ini menuntut adanya sistem keamanan yang lebih efektif, efisien, dan adaptif terhadap perkembangan teknologi.

Di era modern ini, teknologi telah menjadi bagian integral dalam berbagai aspek kehidupan, termasuk dalam bidang keamanan. Perkembangan teknologi *Internet of Things (IoT)* memungkinkan integrasi perangkat keras dan perangkat lunak untuk menciptakan sistem keamanan



yang pintar dan responsif. Sistem tradisional yang hanya mengandalkan pengawasan manual kini mulai tergantikan oleh sistem otomatis yang mampu mendeteksi, memantau, dan memberikan respons secara *real-time*. Penggunaan sensor, kamera, dan sistem alarm yang terhubung dengan internet merupakan contoh nyata penerapan teknologi ini, yang memungkinkan pengguna untuk mengakses dan mengontrol sistem keamanan dari jarak jauh.

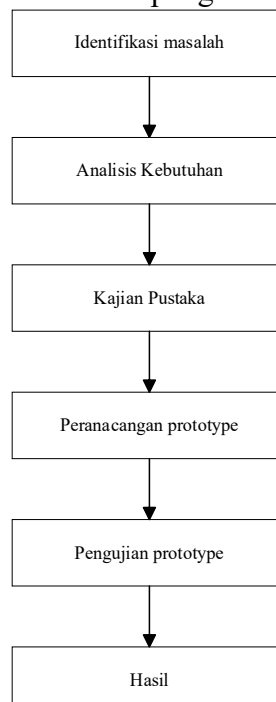
Salah satu implementasi nyata dari penerapan teknologi ini adalah pengembangan sistem keamanan pada gerbang geser otomatis. Sistem ini memanfaatkan mikrokontroler ESP32 sebagai inti pengendali, yang memiliki keunggulan dalam hal konektivitas dan fleksibilitas. Dalam prototipe yang dikembangkan, gerbang geser dilengkapi dengan tombol manual untuk membuka dan menutup, serta sistem notifikasi berbasis aplikasi *Telegram* yang memungkinkan pengguna mendapatkan informasi secara langsung mengenai status gerbang.

Keunggulan utama dari sistem ini tidak hanya terletak pada peningkatan keamanan, tetapi juga pada efisiensi operasional. Sistem otomatis mengurangi ketergantungan terhadap petugas keamanan, sehingga dapat menekan biaya tenaga kerja dan meningkatkan efektivitas pengawasan. Selain itu, pengguna juga memperoleh kenyamanan lebih dalam mengelola akses ke properti mereka.

Dengan adanya prototipe sistem keamanan gerbang geser berbasis ESP32 ini, diharapkan dapat memberikan kontribusi nyata dalam penerapan teknologi *IoT* untuk keamanan properti. Sistem ini tidak hanya menjadi solusi praktis bagi pengguna individu, tetapi juga dapat dijadikan acuan untuk pengembangan sistem keamanan di lingkungan yang lebih luas, seperti kawasan perumahan, perkantoran, maupun fasilitas industri. Integrasi teknologi modern dalam sistem keamanan menjadi langkah penting untuk menciptakan lingkungan yang lebih aman, nyaman, dan terkendali.

METODE PENELITIAN.

Metode yang digunakan dalam penelitian ini adalah metode pengembangan *Prototype*, yang bertujuan untuk membangun dan menguji *Prototype* keamanan gerbang geser Berbasis ESP32 dan Notifikasi *Real-time*. Metode ini dipilih karena memungkinkan pengembangan sistem secara bertahap dengan melakukan evaluasi dan perbaikan sebelum produk akhir digunakan. Kerangka kerja penelitian ini mengikuti tahapan dalam metode pengembangan *Prototype*, yang terdiri dari:



Gambar 1. Kerangka Penelitian



1. Identifikasi Masalah

Langkah awal dalam penelitian ini adalah mengidentifikasi permasalahan yang ada di lapangan, yaitu kebutuhan akan sistem keamanan gerbang yang praktis, otomatis, dan terintegrasi. Sistem keamanan konvensional memiliki berbagai keterbatasan, seperti ketergantungan pada tenaga manusia, kurangnya notifikasi *real-time*, serta tidak adanya kendali jarak jauh. Permasalahan ini menjadi dasar pengembangan sistem keamanan yang lebih modern dan berbasis teknologi.

2. Analisis Kebutuhan

Setelah masalah diidentifikasi, dilakukan analisis kebutuhan sistem. Dalam konteks ini, peneliti menentukan komponen apa saja yang dibutuhkan untuk membangun *Prototype*, seperti mikrokontroler ESP32, tombol *switch* sebagai kendali manual, dan sistem notifikasi *Telegram* sebagai media komunikasi *real-time*. Analisis juga mencakup kebutuhan perangkat keras (*hardware*) dan perangkat lunak (*software*), serta integrasi antara keduanya.

3. Kajian Pustaka

Penelitian ini dilanjutkan dengan kajian pustaka yang mencakup studi literatur dari jurnal, artikel, maupun proyek terdahulu yang relevan. Tujuannya adalah memperoleh pemahaman mendalam mengenai teknologi *IoT*, fungsi dan penggunaan ESP32, serta penerapan sistem notifikasi *real-time*. Kajian ini memberikan landasan teoritis untuk perancangan *Prototype*.

4. Perancangan *Prototype*

Pada tahap ini, peneliti mulai merancang sistem keamanan gerbang geser. Rancangan meliputi skema rangkaian elektronik, pemrograman mikrokontroler ESP32, serta integrasi antara tombol *switch* dengan sistem notifikasi *Telegram*. Perancangan juga mempertimbangkan aspek efisiensi, keandalan, dan kemudahan penggunaan.

5. Pengujian *Prototype*

Prototype yang telah dirancang kemudian diuji untuk memastikan semua komponen bekerja sesuai fungsinya. Pengujian mencakup respon tombol manual terhadap gerbang, keakuratan notifikasi *Telegram*, serta kestabilan koneksi internet. Tahapan ini penting untuk mengevaluasi apakah sistem dapat bekerja dengan baik dalam kondisi nyata.

6. Hasil

Tahap akhir dari metode ini adalah hasil pengujian. Hasil mencakup tingkat keberhasilan sistem dalam mendeteksi dan mengontrol status gerbang, kecepatan pengiriman notifikasi. Penelitian ini dilakukan di rumah peneliti yang dilaksanakan pada Januari 2025 sampai dengan Juni 2025, Pemilihan lokasi ini bertujuan untuk mempermudah proses perancangan, dan pengujian *prototype* secara langsung.

HASIL DAN PEMBAHASAN

Analisa Kebutuhan

Dalam perancangan *prototype* keamanan gerbang geser berbasis esp32 menggunakan tombol *switch* dan notifikasi *real time*, terdapat beberapa kebutuhan yang harus dipenuhi sistem dapat berfungsi dengan baik. kebutuhan tersebut meliputi:

1. Kebutuhan Perangkat Keras

- Esp32, sebagai mikrokontroler utama yang mengendalikan seluruh sistem

Tabel 1. ESP32

Mikrokontroler	Esp32
Operasi tegangan	3.3 Volt
Input tegangan	7-12V melalui Vin
Pin I/O Digital	25 Pin GPIO
Pin analog	6 Pin ADC
USB Controller	CP2102



<i>Wifi</i>	IEEE 802.11 b/g/n/e/i
Memori <i>flash</i>	4 MB
SRAM	520 KB
Kecepatan <i>clock</i>	240 MHz

- b. *Limit switch*, digunakan untuk mendeteksi kondisi gerbang (terbuka/tertutup).

Tabel 2. *Limit Switch*

Tipe	Mekanis
Tegangan maksimum	AC 125V-250V
<i>Curent Rating</i>	5A
Material kontak	Perak
Ukuran	Mini size
Mode kerja	On/off ketika ditekan atau dilepas

- c. *Buzzer*, berfungsi untuk memberikan peringatan suara saat gerbang terbuka dan tertutup.

Tabel 3. *Buzzer*

Jenis	Buzzer aktif
Tegangan operasi	5V
Frekuensi suara	2KHz
Diameter	± 14 mm

- d. *Adaptor*, untuk menghubungkan ke Listrik.

Tabel 4. *Adaptor*

Tegangan <i>output</i>	5V atau 12V
Arus <i>output</i>	1A – 2A
Tipe konektor	USB
Sumber listrik	PLN

- e. *Smartphone*.

Tabel 5. *Smartphone*

Model	Redmi 2312DRAABG
CPU	Mediatek Dimensity 6080
Penyimpanan	256 GB
Ram	8,0GB

2. Kebutuhan Perangkat Lunak

1. Arduino ide

Digunakan untuk menulis, mengedit dan mengunggah program ke mikrokontroler ESP32.

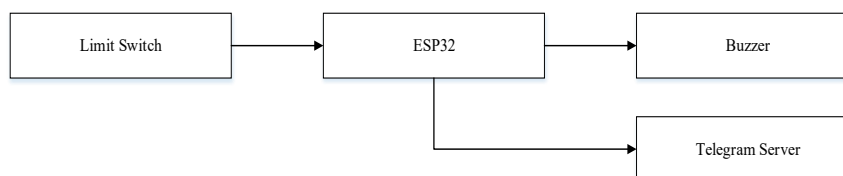
2. *Telegram*

Aplikasi yang digunakan untuk mengirim dan menerima notifikasi secara *real-time* dari alat melalui bot *telegram*.

Perancangan Sistem

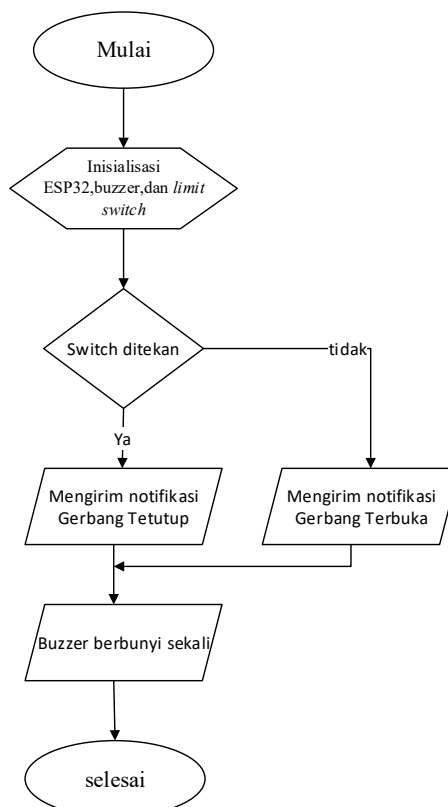
1. Diagram Blok Sistem

Berikut adalah diagram blok sistem dari prototype keamanan gerbang geser berbasis ESP32 menggunakan tombol limit switch dan notifikasi real time.



Gambar 1. Diagram Blok

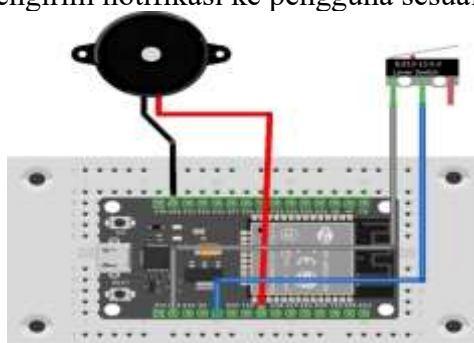
2. Flowchart



Gambar 2. Flowchart

Desain Perangkat Keras

1. Rangkaian Elektronik
 - a. *Limit switch* ditekan kondisi gerbang tertutup
 - b. *Limit switch* dilepas kondisi gerbang terbuka
 - c. Esp32 membaca status *limit switch* melalui pin digital
 - d. *Buzzer* berbunyi jika gerbang terbuka / tertutup
 - e. *Telegram* bot mengirim notifikasi ke pengguna sesuai status gerbang



2. Skematik Rangkaian

Gambar 3. Skematik rangkaian



Skematik rangkaian adalah diagram yang menjelaskan hubungan kelistrikan antar komponen, berikut komponen skema yang digunakan :

- ESP32 menjadi pusat kendali sistem menerima input dari *limit switch* dan mengirim output ke *buzzer* serta *telegram*.
- Limit switch* berfungsi untuk mendeteksi status gerbang tertutup atau terbuka.
- buzzer* berfungsi sebagai alarm lokal ketika status gerbang berubah.
- kabel *jumper* atau *breadboard* digunakan untuk menghubungkan komponen.

Testing Dan Implementasi

Testing

1. Pengujian Limit Switch

Pengujian dilakukan untuk memastikan limit switch dapat mendeteksi kondisi gerbang, prosedur pengujian:

- Pastikan gerbang dalam keadaan awal terbuka.
- Geser gerbang sehingga menekan *switch*, terbaca notifikasi gerbang tertutup.
- Lepas gerbang sehingga *switch* tidak menekan, terbaca notifikasi gerbang terbuka.

Kode Program

```

57 void loop() {
58   int state = digitalRead(Switch);
59
60   if (state == LOW && lastState == HIGH) {
61     digitalWrite(Buzzer, HIGH);
62     delay(200);
63     digitalWrite(Buzzer, LOW);
64     delay(100);
65     sendMessage("🔒 Gerbang Tertutup");
66   }
67   else if (state == HIGH && lastState == LOW) {
68     digitalWrite(Buzzer, HIGH);
69     delay(200);
70     digitalWrite(Buzzer, LOW);
71     delay(100);
72     sendMessage("🔓 Gerbang Terbuka");
73   }
74
75   lastState = state;
76   delay(100);
77 }

```

2. Pengujian Buzzer

Pengujian dilakukan untuk memastikan buzzer berbunyi sekali saat ada perubahan status gerbang.

Tabel 6. Pengujian Buzzer

Input	Output yang diharapkan
Gerbang ditekan	Buzzer bunyi sekali
Gerbang dilepas	Buzzer bunyi sekali

3. Pengujian Notifikasi Telegram

Memastikan pesan notifikasi dikirim saat terjadi perubahan Status.

- saat gerbang tertutup pesan notifikasi terkirim ke telegram.
- saat gerbang terbuka pesan notifikasi terkirim ke telegram.

Kode program



```

57 void loop() {
58   int state = digitalRead(Switch);
59
60   if (state == LOW && lastState == HIGH) {
61     digitalWrite(Buzzer, HIGH);
62     delay(200);
63     digitalWrite(Buzzer, LOW);
64     delay(100);
65     sendMessage("🚪 Gerbang Tertutup");
66   }
67   else if (state == HIGH && lastState == LOW) {
68     digitalWrite(Buzzer, HIGH);
69     delay(200);
70     digitalWrite(Buzzer, LOW);
71     delay(100);
72     sendMessage("🚪 Gerbang Terbuka");
73   }
74
75   lastState = state;
76   delay(100);
77 }

```

Implementasi Prototype

1. Instalasi Perangkat

Letakkan semua komponen di breadboard:

- ESP32.
- Lmit Switch.
- Buzzer.
- Kabel jumper.
- Adaptor.

2. Pengaturan Bot Telegram

Pastikan bot sudah memiliki token aktif dan kamu telah menyerahkan chat ID di kode.

3. Upload Program

Upload program utama ke ESP32 dan pastikan:

- Serial monitor menampilkan teks koneksi wifi.
- Gerbang menekan switch, buzzer berbunyi dan notifikasi terkirim.
- Gerbang dilepas, buzzer berbunyi dan notifikasi terkirim.

Kode Program

```

1 #include <WiFi.h>
2 #include <HTSPI.h>
3
4 #define Buzzer 5
5 #define Switch 4
6
7 const char ssid = "R";
8 const char password = "123456789";
9
10 String urlToken = "https://api.telegram.org/bot<token>";
11 String chatID = "<chatID>";
12
13 int lastState = HIGH;
14 bool isDisconnected = false;
15
16 void sendMessage(String message) {
17   if (HTTP_STATUS_OK == HTTP_STATUS_OK) {
18     HTTPClient http;
19     String url = "https://api.telegram.org/bot<token>sendMessage?chat_id=" + chatID + "&text=" + message;
20     http.begin(url);
21     int statusCode = http.GET();
22     if (statusCode != 200) {
23       Serial.println("Notifikasi terkirim " + message);
24       delay(1000);
25       Serial.println("❌ Gagal kirim notifikasi");
26     }
27     http.end();
28   } else {
29     Serial.println("Tidak terkoneksi ke WiFi. Notifikasi tidak terkirim.");
30   }
31 }

```




```

33
34 void setup() {
35   pinMode(Buzzer, OUTPUT);
36   pinMode(Switch, INPUT_PULLUP);
37   Serial.begin(115200);
38
39   WiFi.begin(ssid, password);
40   Serial.print("Menghubungkan ke WiFi");
41   int attempt = 0;
42
43   while (WiFi.status() != WL_CONNECTED && attempt < 20) {
44     delay(500);
45     Serial.print(".");
46     attempt++;
47   }
48
49   if (WiFi.status() == WL_CONNECTED) {
50     Serial.println("\nWiFi Terhubung: " + WiFi.localIP().toString());
51     wifiConnected = true;
52   } else {
53     Serial.println("\nGagal terhubung ke WiFi");
54     wifiConnected = false;
55   }
56 }
57
58 void loop() {
59   int state = digitalRead(Switch);
60
61   if (state == LOW && lastState == HIGH) {
62     digitalWrite(Buzzer, HIGH);
63     delay(200);
64     digitalWrite(Buzzer, LOW);
65
66     sendMessage("🚪 Gerbang Tertutup");
67   }
68   else if (state == HIGH && lastState == LOW) {
69     digitalWrite(Buzzer, HIGH);
70     delay(200);
71     digitalWrite(Buzzer, LOW);
72     delay(100);
73     sendMessage("🚪 Gerbang Terbuka");
74   }
75
76   lastState = state;
77   delay(100);
78 }

```

Hasil Implementasi

1. Tampilan *Prototype*



Gambar 4. *Prototype* Ketika Gerbang Dibuka

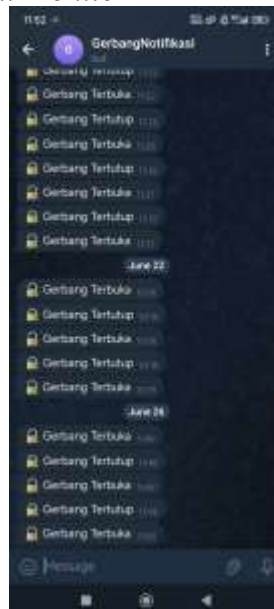
Gambar 4 menunjukkan kondisi *limit switch* tidak tertekan dan status gerbang yang terikirim di *telegram* “gerbang terbuka”.



Gambar 5. Prototype Ketika Gerbang Ditutup

Gambar 5 menunjukkan kondisi *limit switch* tertekan dan status gerbang yang terikirm di *telegram* “gerbang tertutup”.

2. Notifikasi Telegram Dan Serial Monitor



Gambar 6. Notifikasi Telegram.



Gambar 7. Serial Monitor

Gambar 6 dan 7 menampilkan notifikasi status gerbang di aplikasi *telegram* dan serial monitor pada aplikasi *arduino IDE*. status notifikasi berubah ketika *limit switch* tertekan atau tidak.



KESIMPULAN

Berdasarkan hasil perancangan, pengujian, dan implementasi yang telah dilakukan pada prototipe sistem keamanan gerbang geser berbasis ESP32, dapat disimpulkan bahwa, sistem yang dirancang mampu mendeteksi perubahan status pada gerbang (terbuka atau tertutup) menggunakan tombol *switch* (*limit switch*) secara akurat, *buzzer* berhasil memberikan peringatan bunyi singkat setiap kali status gerbang berubah, sebagai bentuk alarm lokal, sistem berhasil mengirimkan notifikasi real-time ke aplikasi *Telegram* menggunakan koneksi internet melalui ESP32, sehingga pengguna dapat menerima informasi status gerbang dari jarak jauh, sistem ini dapat digunakan sebagai solusi sederhana dan murah dalam meningkatkan keamanan akses gerbang, khususnya pada lingkungan rumah atau kantor kecil.

SARAN

Untuk pengembangan lebih lanjut, berikut beberapa saran yang dapat dipertimbangkan:

1. Menambahkan kamera atau sensor gerak (PIR) untuk mendeteksi aktivitas di sekitar gerbang agar sistem lebih komprehensif.
2. Menyimpan log data status gerbang ke database (*online/offline*) agar pengguna dapat memantau riwayat aktivitas.
3. Mengintegrasikan kontrol gerbang secara otomatis (menggunakan motor servo atau motor DC) yang bisa diakses dari aplikasi.
4. Menambahkan fitur autentikasi pengguna agar hanya orang tertentu yang bisa mengakses sistem atau menerima notifikasi.
5. Mengoptimalkan sumber daya (power supply) agar sistem tetap bisa beroperasi saat terjadi pemadaman listrik, misalnya dengan menambahkan baterai cadangan.

DAFTAR PUSTAKA

- Ardana, M., Panjaitan, B., & Budi Santoso, T. (2023). Rancang Bangun Sistem Keamanan Rumah Berbasis *Internet of Things* (IoT). *Jurnal Limits*, 20(2), 23–36. <https://doi.org/10.59134/jlmt.v20i2.605>
- DerIoTa.com. (2024). *Kekuatan dan Keterjangkauan: Memahami Perbedaan ESP8266 dan ESP32*. <https://derIoTa.com/news/read/1328/kekuatan-dan-keterjangkauan-memahami-perbedaan-esp8266-dan-esp32.html>
- Gunawan, I., Akbar, T., & Giyandhi Ilham, M. (2020). Prototipe Penerapan *Internet of Things* (IoT) Pada Monitoring Level Air Tandon Menggunakan Nodemcu Esp8266 Dan Blynk. *Infotek : Jurnal Informatika Dan Teknologi*, 3(1), 1–7. <https://doi.org/10.29408/jit.v3i1.1789>
- Hafit Setiawan, B., & Junianto, E. (2024). Sistem Pengendalian Pintu Gerbang Otomatis Berbasis IoT Menggunakan Aplikasi Blynk. *E-Prosidings Teknik Informatika*, 5(1), 1–8. <https://eprosiding.ars.ac.id/index.php/pti/article/view/1137>
- Hilman Aziz, & Imam Suharjo. (2024). Pengembangan Sistem Keamanan Gerbang Rumah Smart Home Berbasis IoT dengan Metode RnD. *JEKIN - Jurnal Teknik Informatika*, 4(3), 663–674. <https://doi.org/10.58794/jekin.v4i3.895>
- Nabawi, I. S., & Badarudin, R. (2024). Perancangan Sistem Kontrol Lampu Berbasis Mikrokontroler Arduino Uno R3 Dengan Sensor Suara Lm393, Electret Microphone. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3). <https://doi.org/10.23960/jitet.v12i3.4536>
- Nurul Hidayatullah, R., Ariesanto Ramdhan, N., & Khamid, A. (2024). Pengembangan Kendali Lampu Menggunakan Mikrokontroler Nodemcu Esp32 Dan Arduino IDE Berbasis *Internet of Things* (IoT). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 7762–7767. <https://doi.org/10.36040/jati.v8i4.10461>



- Pangestu, G. A., & Asyhari, M. Y. (2024). *Sistem Keamanan Rumah Berbasis Internet of Things (IoT) Menggunakan Notifikasi Bot Telegram untuk Pendeteksian Gerak*. 4(1), 1–14.
- Panggabean, J. H., & Gultom, M. V. (2023). Pengembangan Sistem Pengontrol Dan Monitoring Pintu Gerbang Menggunakan Smartphone Melalui WiFi Berbasis Arduino. *Einstein*, 11(3), 177. <https://doi.org/10.24114/eins.v11i3.34039>
- Prasetyanto, A. E., & Hadisusila, C. P. (2023). Aplikasi Arduino dalam Teknik I/O untuk Mengintegrasikan dan Mengendalikan Perangkat Elektronik. *Nusantara of Engineering (NOE)*, 6(2), 96–102. <https://doi.org/10.29407/noe.v6i2.21308>
- Raditya, W., Styawati, S., Surahman, A., Budiawan, A., Amanda, F., Putri, N. D., & Yudha, S. (2023). Penerapan Sistem Keamanan Gerbang Rumah Berbasis Telegram Menggunakan Esp8266. *Jurnal Teknik Dan Sistem Komputer*, 3(2), 93–103. <https://doi.org/10.33365/jtikom.v3i2.2353>
- Selokaton, A. P., Herwanto, A., Informatika, T., Unggul, U. E., & Lock, S. D. (2025). *PENERAPAN INTERNET OF THINGS (IOT) UNTUK MERANCANG SISTEM KEAMANAN PADA BRANKAS MENGGUNAKAN ESP32-CAM DAN KEYPAD*. 8(5), 10870–10878.
- Sepudin, D. M., & Abdullah, S. (2023a). Sistem keamanan pintu rumah berbasis aplikasi Telegram dan Internet of Things (IoT). *Jurnal RESTIKOM: Riset Teknik Informatika Dan Komputer*, 4(3), 93–99.
- Sepudin, D. M., & Abdullah, S. (2023b). Sistem Keamanan Pintu Rumah Berbasis Internet of Things Berbasis NodeMCU ESP32 dan Telegram. *Jurnal RESTIKOM: Riset Teknik Informatika Dan Komputer*, 4(3), 93–99. <https://doi.org/10.52005/restikom.v4i3.99>
- Setiawan, R. (2021). *Apa Itu Prototype? Kenapa Itu Penting?* Wwww.Dicoding.Com. <https://www.dicoding.com/blog/apa-itu-Prototype-kenapa-itu-penting/>
- Setiawan, R. (2022). *Apa itu Arduino? Pahami Lebih Mendalam*. <https://www.dicoding.com/blog/apa-itu-arduino/>
- Simarmata, S. D., Gunawan, I., Sari, I. P., Sumarno, S., & Kirana, I. O. (2021). Sistem Kendali Pintu Gerbang Otomatis Menggunakan Koneksi Wireless Module WiFi Berbasis Mikrokontroler Arduino Uno. *Jurnal Pendidikan Dan Teknologi Indonesia*, 1(7), 297–308. <https://doi.org/10.52436/1.jpti.67>
- Suriana, I. W., Setiawan, I. G. A., & Graha, I. M. S. (2022). Rancang Bangun Sistem Pengaman Kotak Dana Punia berbasis Mikrokontroler NodeMCU ESP32 dan Aplikasi Telegram. *Jurnal Ilmiah Telsinas Elektro, Sipil Dan Teknik Informasi*, 4(2), 75–84. <https://doi.org/10.38043/telsinas.v4i2.3198>
- Wasista Sigit, Setiawardana, Saraswati Ayu Delima, susanto E. (2019). *aplikasi Internet of Things (IoT) dengan arduino dan android “membangun smart home dan smart robot berbasis arduino dan android.”*
- Wicaksono Fajar Mochammad, S.Kom., M.Kom, Hidayat, S.Kom., M. . (2017). *Mudah Belajar Mikrokontroler Arduino uno*.