

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Sejumlah penelitian terdahulu telah mengembangkan cara untuk menghitung jumlah pengunjung menggunakan berbagai metode. Penelitian terkait tersebut merupakan sumber utama dan juga menunjukkan perbedaan pada penelitian ini, karena berfokus pada integrasi ESP32, sensor infrared proximity, dan mekanisme penguncian secara otomatis untuk mencegah overkapasitas ruangan secara real-time. Penelitian tersebut antara lain:

Table 1. Penelitian terkait

No	Nama (Tahun)	Judul
(1)	(2)	(3)
1.	Irham Ramadana Putra, Alamsyah, Mery S, Ardi Amir, Tan Suryani S[8]	Judul: Rancang Bangun Sistem Informasi Penghitung Jumlah Orang pada Ruangan Tertutup berbasis <i>Internet of Things</i> (IoT) Hardware: ESP32 CAM, Sensor IR Proximity Software: Arduino IDE, Database Fitur: Streaming kamera, Web interface Hasil penelitian: Sistem penghitung orang berbasis ESP32-CAM dan sensor IR mampu mendeteksi dan menghitung jumlah orang secara real-time. Data diproses oleh ESP32-CAM, disimpan ke database, dan ditampilkan di LCD serta web berupa jumlah orang dan streaming kamera.
2.	Mega Fahmawaty, Muhammad Royhan, Mahmudin[1]	Judul: Perancangan Alat Penghitung Jumlah Pengunjung Di Perpustakaan Unis Tangerang Menggunakan Sensor Pir Berbasis IoT Hardware: Wemos D1 ESP8266, sensor PIR HC-SR501, LED Software:

No	Nama (Tahun)	Judul
		Arduino IDE, <i>ThinkSpeak</i> Fitur: <i>Cloud monitoring (ThinkSpeak), LED Indicator</i> Hasil Penelitian: Sistem ini mendeteksi kehadiran pengunjung menggunakan sensor PIR, mengolah data melalui Wemos, dan mengaktifkan LED sebagai indikator masuk/keluar. Ketika kapasitas mencapai batas 70 orang, LED peringatan menyala. Pemrograman menggunakan Arduino IDE berhasil diimplementasikan dengan baik, dan data jumlah pengunjung tersimpan di platform ThingSpeak untuk pemantauan online.
3.	Bustanul Arifin, Eka Nuryanto Budisusila, Amir Cahyadi[9]	Judul: Penghitung Jumlah Orang Dalam Ruang Dengan Sensor Inframerah Dan Modul Lcd Tft Sebagai Display Hardware: Arduino Mega 2560, sensor IR Sharp CP2Y0A02, LCD TFT Software: Arduino IDE Hasil Penelitian: Hasil penelitian ini menunjukkan Sensor infrared Sharp GP2Y0A02 berfungsi optimal pada jarak 15-150 cm. Data deteksi objek berhasil diproses Arduino Mega 2560 dan ditampilkan di LCD TFT dengan waktu penyimpanan 59 detik. Sistem bekerja stabil untuk objek berwarna terang.
4.	Rizky Dwi Saputra, Affan Bachri, Arif Budi Laksono[10]	Judul: Rancang Bangun Alat Penghitung Jumlah Pengunjung Di Restoran Menggunakan Sensor Pir Hc-Sr501 Dan Pintu Otomatis Untuk Penerapan Social Distancing Berbasis Arduino Hardware: Arduino Uno, sensor PIR HC-SR501, sensor IR, servo motor, LCD 16x2, <i>DFPlayer Mini</i>, Buzzer Software: Arduino IDE Fitur: Kontrol pintu otomatis, <i>Audio Alert, Social Distancing</i> Hasil Penelitian:

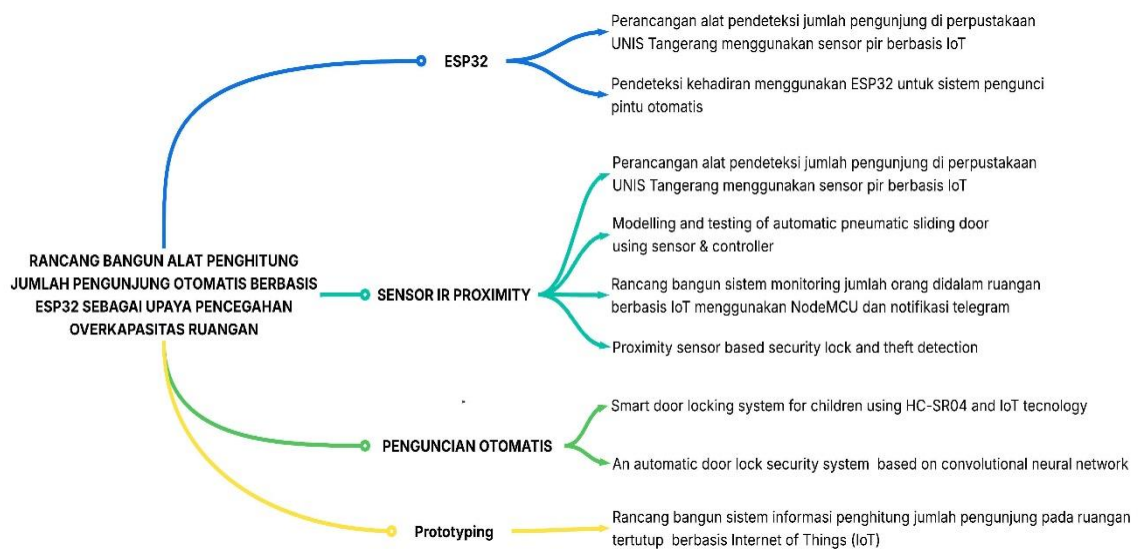
No	Nama (Tahun)	Judul
		Ketika sensor PIR mendeteksi objek (logika 1), pintu terbuka otomatis via motor servo. Sensor infrared berfungsi sebagai penghitung pengunjung dan mengaktifkan buzzer, dengan data ditampilkan di LCD. Saat kapasitas penuh, sistem menampilkan "sudah penuh" di LCD dan mengeluarkan suara peringatan melalui <i>DFPlayer</i> .
5.	Anton Prafanto, Edy Budiman, Putut Pamilih Widagdo, Gubtha Mahendra Putra, Reza Wardhana[4]	Judul: Pendeteksi Kehadiran menggunakan ESP32 untuk Sistem Pengunci Pintu Otomatis Hardware: ESP32, Solenoid Lock Software: Arduino IDE Teknologi: <i>Bluetooth LOW Energy (BLE)</i>, RSSI Fitur: <i>Access Control</i> berbasis Proximity Hasil Penelitian: Hasil penelitian ini menunjukkan bahwa Sistem deteksi berbasis RSSI dengan ESP32 berhasil mendeteksi keberadaan individu dan memperkirakan jarak antar-perangkat berdasarkan kekuatan sinyal.
6.	Rocky T. Detaq, Molina O. Odja, Almido H. Ginting[11]	Judul: Sistem Kontrol dan Monitoring Keamanan Pintu Menggunakan Mikrokontroler dan Smartphone Hardware: Arduino Uno, sensor <i>Fingerprint</i>, Solenoid lock, kamera Software: Arduino IDE, Aplikasi Telegram Fitur: <i>Biometric Access</i>, <i>Remote Monitoring</i>, notifikasi Telegram Hasil Penelitian: Hasil penelitian ini menunjukkan bahwa sistem kontrol dan monitoring keamanan pintu berbasis mikrokontroler dan smartphone berhasil bekerja sesuai rancangan, mampu membuka kunci pintu dalam waktu 5 detik ketika sidik jari terdeteksi valid, serta mengaktifkan fitur keamanan berupa

No	Nama (Tahun)	Judul
		pengambilan foto dan pengiriman notifikasi melalui Telegram ketika sidik jari tidak terdaftar.
7.	Ardi Susanto, Rizki Wijayatun Pratiwi[12]	Judul: Alat Kendali Perangkat Ruangan Otomatis Dengan Sistem Penghitung Menggunakan Sensor Infrared Berbasis Arduino Hardware: Arduino UNO R3, Sensor Infrared E18-D50NK, LCD Karakter 16x2 modul 12C Software: Arduino IDE, Proteus Fitur: Occupancy-based Automation, Dual IR <i>Beam Counting</i>, Monitoring Status Sistem Secara Real Time Hasil Penelitian: Hasil penelitian ini menunjukkan bahwa sistem kendali perangkat ruangan otomatis berbasis mikrokontroler Arduino berhasil bekerja sesuai rancangan, mampu menghitung jumlah orang yang masuk dan keluar ruangan melalui dua sensor infrared secara akurat, serta mengaktifkan atau menonaktifkan perangkat elektronik secara bertingkat berdasarkan jumlah penghuni ruangan yaitu hanya lampu pinggir atau seri menyala ketika terdapat satu orang, lampu utama, lampu pinggir, dan televisi menyala ketika terdapat dua orang, serta seluruh perangkat termasuk kipas angin menyala ketika jumlah penghuni mencapai enam orang atau lebih dengan seluruh status sistem ditampilkan secara real-time pada LCD karakter 16x2.

No	Nama (Tahun)	Judul
8.	Arini Murdhiani, Habibullo[13]	Judul: Pengembangan Sistem Penghitung Pengunjung Ruang Baca Departemen Teknik Elektro Menggunakan Teknologi Mikrokontroler Hardware: Arduino Nano, NodeMCU ESP8266, Sensor Ultrasonik HC-SR04, Line Sensor, Buzzer, LCD Software: Arduino IDE, Ubidots Fitur: Real Time Visitor Counting dengan dual Sensor (Ultrasonik dan Line Sensor), Monitoring Jarak Jauh via Ubidots, Occupancy-Based Trigger Hasil Penelitian: Hasil penelitian menunjukkan bahwa Sistem mampu mendeteksi pergerakan masuk dan keluar pengunjung secara akurat melalui kombinasi dua sensor ultrasonik HC-SR04 dan satu sensor garis yang dipasang pada satu pintu, dengan rata-rata kesalahan pengukuran jarak sensor ultrasonik sebesar 0,1 cm pada pengujian pengunjung masuk dan 0,2 cm pada pengujian pengunjung keluar. Data jumlah pengunjung yang masuk, keluar, dan berada di dalam ruangan diproses oleh Arduino Nano dan dikirim secara real-time ke LCD untuk tampilan lokal serta ke platform Ubidots melalui NodeMCU ESP8266 untuk monitoring jarak jauh. Pengujian keseluruhan sistem membuktikan bahwa output pada LCD konsisten dengan data yang ditampilkan pada dashboard Ubidots, dan sistem mampu memicu buzzer sebagai indikator audio ketika jumlah pengunjung mencapai batas maksimal yang telah ditentukan (10 orang), sekaligus menampilkan notifikasi “ruangan penuh” pada LCD.
9.	Rindang Alamsyah, Ika Karlina Laila Nur Suciningtyas, Nanta Fakih Pebrianto, Iman Fahruzi, Imam Sholihuddin, Abdurrahman Dwijotomo, Lalu Kaisar Wisnu	Judul: Alat Pemantau dan Penghitung Jumlah Pengunjung berbasis Aplikasi Telegram Hardware: ESP32-CAM, Sensor Infrared Software: Arduino IDE, Firmware ESP32-CAM, Telegram BOT Fitur:

No	Nama (Tahun)	Judul
	Kita, Indra Daulay[14]	Pengambilan Gambar Otomati Sebagai Bukti Fisik Pengunjung, Pengiriman Data Real Time ke Telegram, Remote Control Via Telegram BOT Hasil Penelitian: Hasil penelitian menunjukkan bahwa sensor infrared mampu mendeteksi keberadaan pengunjung dan memberikan sinyal pemicu kepada ESP32-CAM untuk mengambil gambar sebagai bukti fisik, kemudian mengirimkan data berupa jumlah pengunjung dan gambar tersebut secara otomatis ke aplikasi Telegram. Pengujian terhadap jarak deteksi menunjukkan bahwa sistem berfungsi secara optimal pada rentang jarak 15 hingga 60 cm dari sensor, sedangkan pada jarak 75 cm hingga 90 cm sistem tidak mampu mendeteksi objek dan tidak menghasilkan respons. Waktu rata-rata yang dibutuhkan sistem untuk mendeteksi pengunjung adalah 8,7 detik, sedangkan waktu rata-rata pengiriman gambar ke aplikasi Telegram adalah 6,1 detik, sehingga total delay rata-rata antara deteksi hingga penerimaan data di aplikasi adalah 2,5 detik. Selain itu, waktu tercepat sistem dalam membaca perpindahan antara satu pengunjung ke pengunjung berikutnya adalah 15,4 detik, yang mengindikasikan bahwa sistem paling ideal digunakan di lokasi dengan akses masuk terkendali dan pengunjung datang secara bergantian satu per satu, seperti pada pintu masuk perpustakaan dengan sistem tap kartu.
10.	Timbo Faritcan Siagallan, Tita[15]	Judul: Rancang Bangun Sistem Keamanan terhadap Kunci Ruangan Berbasis BOT Telegram Menggunakan Mikrokontroler ESP8266 Hardware: Wemos ESP8266, Solenoid Door Lock 12V, Modul Relay Software: Arduino IDE, Bascom AVR, Telegram BOT Fitur: Kontrol Kunci Pintu Jarak Jauh, Notifikasi real time saat pintu dibuka atau dibuka paksa, integrasi pengunci otomatis dengan ESP8266, Sistem Berbasis Internet Hasil Penelitian: Hasil penelitian menunjukkan bahwa sistem mampu mengontrol kunci elektrik (<i>solenoid door lock 12 V</i>) secara jarak jauh melalui perintah yang dikirim melalui Telegram, serta memberikan notifikasi kepada pengguna ketika kunci diaktifkan atau dinonaktifkan. Pengujian menunjukkan bahwa ESP8266 mampu terhubung ke jaringan Wi-Fi

No	Nama (Tahun)	Judul
		secara stabil dan merespons perintah dari Telegram dengan tepat, sedangkan modul relay berfungsi sebagai saklar elektronik yang menghubungkan dan memutus arus listrik ke kunci elektrik sesuai sinyal logika dari mikrokontroler. Sistem juga dilengkapi buzzer sebagai indikator suara saat terjadi aksi penguncian atau pembukaan. Meskipun dalam dokumen disebutkan penggunaan XAMPP, MySQL, dan MIT App Inventor dalam analisis kebutuhan, tidak ada bukti pengujian atau implementasi nyata terhadap komponen tersebut dalam bagian implementasi maupun pengujian. satu-satunya output yang diverifikasi adalah kendali dan notifikasi melalui Telegram, sehingga sistem terbukti berfungsi sebagai solusi keamanan pintu berbasis internet dengan jangkauan tak terbatas, berbeda dengan sistem berbasis Bluetooth atau RF yang terbatas jaraknya.



Pendekatan *Research and Development* (R&D) banyak digunakan dalam penelitian terdahulu sebagai metode utama dalam perancangan dan pengujian prototipe sistem berbasis *Internet of Things* (IoT), termasuk alat penghitung jumlah pengunjung otomatis berbasis ESP32. Pada tahap pengujian, pendekatan eksperimental umumnya diterapkan untuk mengevaluasi kinerja prototipe yang telah dikembangkan. Metode ini memungkinkan peneliti untuk secara sistematis mengidentifikasi kebutuhan sistem, merancang arsitektur perangkat keras dan perangkat lunak, mengimplementasikan komponen pendukung, serta menguji kinerja prototipe dalam kondisi yang terkontrol, mulai dari akurasi deteksi sensor hingga respons mekanis sistem penguncian pintu saat kondisi overkapasitas. Pendekatan eksperimental tersebut sejalan dengan standar pengembangan sistem

embedded berbasis IoT, di mana validasi melalui pengujian empiris menjadi aspek utama dalam memastikan keandalan dan fungsionalitas sistem. Alur umum pengembangan sistem berdasarkan penelitian terdahulu dapat digambarkan dalam bentuk mind map sebagai berikut.

Gambar 1. Mind Map Penelitian

B. Landasan Teori

1. Konsep dasar IoT

Internet of Things (IoT) adalah sebuah konsep komputasi di mana benda-benda fisik dilengkapi dengan sensor, aktuator, dan kemampuan untuk terhubung dengan jaringan dapat saling berinteraksi, mengumpulkan informasi, serta membuat keputusan secara mandiri tanpa perlu campur tangan manusia secara langsung[16]. Dalam pengelolaan ruang akademis seperti Laboratorium IoT, penggunaan IoT bukan hanya ditujukan untuk meningkatkan efisiensi operasional, tetapi juga untuk mengatasi masalah nyata seperti kelebihan kapasitas yang sering mengganggu kenyamanan dan efektivitas dalam proses belajar mengajar. Sistem IoT yang ideal dalam situasi ini perlu mampu memantau jumlah orang secara langsung, memberikan umpan balik baik yang terlihat maupun yang terdengar, dan juga dapat melakukan tindakan fisik seperti mengunci pintu secara otomatis ketika batas jumlah kapasitas sudah terlampaui. Arsitektur dasar dari IoT yang meliputi sensor (*perception layer*), mikrokontroler (*processing layer*), dan lapisan antarmuka (*application layer*) menjadi dasar penting dalam menciptakan sistem yang tidak hanya terkoneksi, tetapi juga benar-benar cerdas dan tanggap terhadap perubahan lingkungan[17]. Penerapan sistem seperti ini sejalan dengan hasil penelitian sebelumnya di laboratorium yang menggunakan IoT, di mana pemantauan secara langsung dapat meningkatkan efektivitas pengawasan, meskipun sebagian besar belum memiliki mekanisme closed-loop control untuk mencegah overkapasitas secara otomatis[18].

Keberhasilan sistem IoT tidak dinilai dari seberapa rumit infrastruktur atau banyaknya perangkat yang tersambung, tetapi dari seberapa efektifnya dalam mengatasi masalah yang nyata[19]. Dalam contoh ini, penggunaan sensor IR Proximity untuk deteksi di garis depan, ESP32 sebagai pengolah data di tepi yang mengambil keputusan secara lokal tanpa bergantung pada cloud[20], serta kunci solenoid sebagai alat eksekusi fisik, menunjukkan penerapan IoT yang praktis dan tetap berpegang pada prinsip. Sistem ini membentuk closed-loop automation yakni, ketika sensor mendeteksi

keberadaan pengunjung, ESP32 langsung memperbarui hitungan dan membandingkannya dengan batas kapasitas. Jika melewati batas, maka sistem tidak hanya memberikan peringatan, tetapi juga benar-benar mengunci akses sebagai langkah pencegahan berdasarkan data, bukan hanya mencatat setelah kejadian terjadi. Metode yang sama telah diuji coba pada sistem pengendalian pintu yang menggunakan ESP32 dengan BLE untuk mendeteksi keberadaan orang[1]. Namun dalam penelitian ini, logika pengambilan keputusan tidak ditentukan oleh ada tidaknya smartphone pengguna, melainkan oleh peristiwa fisik (masuk atau keluar ruangan), sehingga lebih dapat diandalkan dan lebih inklusif. Oleh karena itu, prototipe ini bukan hanya sekadar alat untuk menghitung, tetapi juga representasi nyata dari filosofi IoT dan mengubah ruang yang pasif menjadi entitas yang aktif yang dapat merasakan, berpikir, dan bertindak untuk meningkatkan kenyamanan, keamanan, dan efisiensi pengguna terutama di laboratorium yang seharusnya menjadi contoh dalam penerapan teknologi ini.

2. Prinsip *fail-secure* dalam sistem pengamanan



Gambar 2. Logo fail-secure

Dalam desain sistem keamanan fisik berbasis aktuator elektromekanis, prinsip *fail-secure* merupakan pendekatan rekayasa yang menjamin sistem tetap dalam kondisi terkunci atau aman dari akses tidak sah ketika terjadi gangguan daya, kegagalan komponen, atau pemadaman sistem. Menurut Tsankov, *fail-secure* didefinisikan sebagai mekanisme di mana akses ke area terbatas secara otomatis diblokir saat sistem tidak beroperasi, sehingga memprioritaskan integritas keamanan (*security*) dibandingkan kemudahan evakuasi (*safety*)[21]. Penerapan prinsip ini umum ditemukan pada ruang-ruang yang memerlukan kontrol akses ketat, seperti laboratorium riset, ruang server, atau fasilitas dengan aset bernilai tinggi[22]. Dalam konteks prototipe yang dikembangkan, prinsip *fail-secure* diimplementasikan melalui penggunaan solenoid lock tipe *normally closed*, yang secara mekanis akan tetap mengunci pintu ketika tidak dialiri arus listrik.

Keputusan desain ini didasarkan pada karakteristik Laboratorium IoT UNUGHA Cilacap yang bukan merupakan area evakuasi darurat publik, melainkan ruang terbatas yang digunakan untuk praktikum dan penelitian, sehingga prioritas utama adalah mencegah overkapasitas dan menjaga integritas akses. Meskipun konfigurasi ini meningkatkan keamanan, Kathage menegaskan bahwa penerapan *fail-secure* dalam lingkungan terbatas tetap harus disertai dengan mekanisme *emergency override* manual seperti tombol fisik atau tuas mekanis sebagai lapisan mitigasi risiko keselamatan[23]. Prototipe ini memenuhi pertimbangan tersebut melalui penyediaan mode darurat ganda (fisik dan web server) yang memungkinkan pembukaan pintu secara manual saat diperlukan, sejalan dengan prinsip *defense in depth* dalam rekayasa sistem keamanan. Dengan demikian, penerapan prinsip *fail-secure* dalam penelitian ini bukanlah kelemahan, melainkan pilihan teknis yang disengaja berdasarkan analisis konteks operasional, kebutuhan fungsional, dan pertimbangan rekayasa sistem modern.

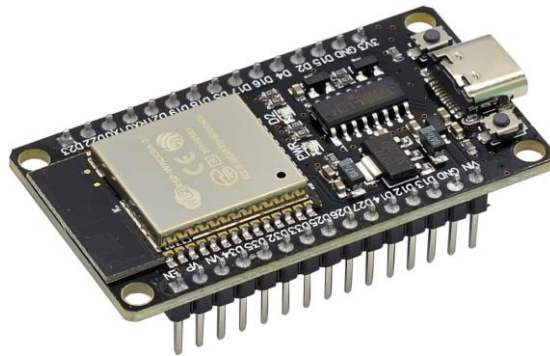
3. Mekanisme *emergency override* dalam sistem keamanan



Gambar 3. Logo emergency

Dalam sistem keamanan berbasis aktuator elektromekanis, mekanisme *emergency override* merupakan komponen penting untuk memastikan keselamatan pengguna saat terjadi kondisi darurat, seperti kebakaran, gempa, atau kegagalan daya[23]. Meskipun sistem dirancang dalam mode *fail-secure* untuk menjaga integritas akses, standar keselamatan modern mewajibkan adanya jalur evakuasi manual yang dapat diaktifkan tanpa bergantung pada sistem utama[21]. Pendekatan ini sejalan dengan prinsip *defense in depth*, di mana lapisan perlindungan tidak hanya mengandalkan otomatisasi, tetapi juga menyediakan intervensi manusia sebagai *last resort*. Dalam konteks IoT, *emergency override* dapat diimplementasikan melalui antarmuka fisik (tombol mekanis) maupun digital (web/mobile), asalkan tetap terisolasi dari logika pencacahan utama agar tidak mengganggu integritas data[24].

4. ESP32 development board



Gambar 4. ESP32 Development Board

Pada dasarnya, ESP32 adalah sebuah System on a Chip (SoC) yang dirancang khusus untuk aplikasi Internet of Things (IoT) dan sistem embedded[3]. Secara fisik, ESP32 sering diimplementasikan dalam bentuk modul terintegrasi seperti ESP32-WROOM-32 yang kemudian dipasang pada sebuah development board untuk memudahkan proses prototipe.

ESP32 menggabungkan kemampuan pemrosesan tinggi dengan konsumsi daya rendah, menjadikannya sempurna untuk aplikasi yang memerlukan operasi berkelanjutan dan efisien.

ESP32 memiliki berbagai fitur unggulan, seperti prosesor dual-core, berbagai GPIO (General Purpose Input/Output), ADC (Analog to Digital Converter), dan kemampuan PWM (Pulse Width Modulation). Fitur-fitur ini memungkinkan ESP32 untuk menangani berbagai tugas, mulai dari membaca sensor hingga mengendalikan aktuator, serta mengirim dan menerima informasi melalui koneksi Wi-Fi atau Bluetooth. Penggunaan ESP32 dalam proyek ini didasarkan pada beberapa keunggulan utama. konektivitas nirkabel melalui Wi-Fi memungkinkan ESP32 untuk berkomunikasi dengan server atau stage cloud untuk mengirimkan informasi secara real-time. fitur prosesor dual-core memungkinkan mikrokontroler ini memproses informasi dengan cepat dan efisien. ESP32 memiliki berbagai pin GPIO yang fleksibel, memudahkan campur tangan dengan komponen lain seperti sensor IR Proximity, modul Relay, dan Solenoid lock . konsumsi daya yang rendah membuat ESP32 cocok untuk aplikasi yang memerlukan operasi terus-menerus tanpa sumber daya besar. Oleh karena itu, ESP32 adalah pilihan yang ideal untuk sistem pendeteksi pengunjung otomatis ini.

1) Prinsip kerja ESP32

ESP32 dikembangkan oleh Espressif Systems dengan kemampuan dual-mode yang mendukung konektivitas Wi-Fi dan Bluetooth. Mikrokontroler ini dirancang untuk aplikasi Internet of

Things (IoT) dan sistem otomatisasi yang memerlukan operasi berdaya rendah serta performa tinggi. Sebagai platform mikrokontroler yang banyak digunakan dalam pengembangan IoT, ESP32 memiliki arsitektur internal yang kompleks namun terintegrasi dengan baik. Untuk memahami bagaimana modul ini dapat menjalankan berbagai fungsi secara efisien, perlu dipelajari mekanisme kerja dasar yang melibatkan koordinasi antar komponen utamanya.

Dengan memahami prinsip kerja ESP32, kita dapat memanfaatkan mikrokontroler ini secara optimal untuk berbagai aplikasi yang memerlukan konektivitas nirkabel, pengolahan data cepat, dan konsumsi daya rendah. ESP32 menawarkan solusi yang fleksibel dan efisien untuk membangun sistem IoT yang inovatif dan andal. Berikut adalah prinsip kerja ESP32 yang mencakup berbagai komponen internal agar bekerja secara sinergis untuk menjalankan fungsinya dengan efisien:

a) Inisialisasi sistem

Ketika ESP32 dihidupkan, langkah pertama yang dilakukan adalah inisialisasi sistem. Bootloader yang berada dalam memori ROM akan dijalankan terlebih dahulu. Bootloader ini bertugas untuk menginisialisasi hardware dan memuat firmware yang berada dalam memori flash. Firmware adalah perangkat lunak yang berisi instruksi yang harus dijalankan oleh mikrokontroler. Proses inisialisasi mencakup pengaturan pin GPIO (General Purpose Input/Output), konfigurasi modul komunikasi, serta inisialisasi timer dan periferal lainnya.

b) Pengolahan prosesor ganda

ESP32 dilengkapi dengan dua inti prosesor Xtensa LX6 yang dapat beroperasi secara bersamaan atau independen. Penggunaan dual-core ini memungkinkan mikrokontroler untuk menjalankan beberapa tugas secara simultan, sehingga meningkatkan efisiensi. Satu inti prosesor dapat digunakan untuk menangani tugas-tugas utama seperti pengolahan data sensor, sementara inti lainnya dapat menangani tugas jaringan atau komunikasi. Prosesor ini juga mendukung operasi floating point untuk perhitungan matematis yang kompleks.

c) Koneksi WiFi

Setelah inisialisasi, modul Wi-Fi diaktifkan, ESP32 melakukan pemindaian jaringan Wi-Fi yang tersedia dan mencoba terhubung

ke jaringan yang telah dikonfigurasi. Proses koneksi melibatkan otentikasi dengan titik akses (AP) dan mendapatkan alamat IP melalui protokol DHCP. Setelah terhubung, ESP32 dapat berkomunikasi dengan server atau perangkat lain melalui protokol komunikasi seperti HTTP, MQTT, atau WebSocket. Koneksi Wi-Fi memungkinkan ESP32 untuk mengirim dan menerima data secara real-time.

d) Koneksi bluetooth

Selain Wi-Fi, ESP32 juga mendukung Bluetooth v4.2 BR/EDR dan BLE (Bluetooth Low Energy). Modul Bluetooth memungkinkan ESP32 untuk berkomunikasi dengan perangkat lain yang mendukung Bluetooth. Proses pairing dan komunikasi Bluetooth melibatkan penemuan perangkat, otentikasi, dan pertukaran data melalui protokol komunikasi Bluetooth. BLE sangat cocok untuk aplikasi yang memerlukan konsumsi daya rendah.

e) Pengolahan data sensor

ESP32 memiliki banyak pin GPIO (General Purpose Input/Output) yang dapat dikonfigurasi sebagai input atau output. Pin GPIO digunakan untuk berinteraksi dengan sensor dan aktuator. Sebagai contoh, ketika sensor gerak IR Proximity mendeteksi gerakan, sinyal dari sensor diterima oleh pin GPIO yang dikonfigurasi sebagai input. ESP32 kemudian mengolah data ini sesuai dengan instruksi yang ada dalam firmware. Jika data menunjukkan adanya gerakan, ESP32 dapat mengirim sinyal ke pin GPIO yang dikonfigurasi sebagai output untuk mengaktifkan perangkat lain seperti alarm atau lampu.

f) Konversi analog ke digital

ESP32 dilengkapi dengan ADC (Analog to Digital Converter) yang dapat mengubah sinyal analog menjadi data digital. Ini memungkinkan ESP32 untuk membaca sinyal dari sensor analog seperti potensiometer atau sensor suhu. Data analog yang diterima oleh pin ADC dikonversi menjadi nilai digital yang dapat diproses oleh mikrokontroler. Konversi ini penting untuk aplikasi yang memerlukan pembacaan data analog dengan presisi tinggi.

g) Manajemen daya

ESP32 memiliki beberapa mode hemat daya yang dapat diaktifkan sesuai dengan kebutuhan aplikasi. Mode deep sleep dan light sleep digunakan untuk mengurangi konsumsi daya saat perangkat tidak aktif. Dalam mode deep sleep, hampir semua komponen ESP32 dimatikan kecuali Real-Time Clock (RTC) yang menjaga waktu. Mode ini memungkinkan ESP32 untuk beroperasi dalam jangka

waktu yang lama dengan sumber daya terbatas seperti baterai. ESP32 dapat dibangunkan dari mode sleep oleh timer atau interrupt eksternal.

h) Pengembangan dan pemrograman

ESP32 dapat diprogram menggunakan berbagai platform dan bahasa pemrograman, termasuk Arduino IDE, MicroPython, dan ESP-IDF (Espressif IoT Development Framework). Pengembangan perangkat lunak melibatkan penulisan kode, kompilasi, dan pengunggahan firmware ke mikrokontroler. Firmware menentukan bagaimana ESP32 berinteraksi dengan sensor, memproses data, dan berkomunikasi dengan perangkat lain. Penggunaan platform yang beragam memberikan fleksibilitas dalam pengembangan aplikasi yang kompleks dan canggih.

Table 2. Spesifikasi ESP32 Devkit V1

kategori	Spesifikasi
Chip Utama	ESP32-D0WD-V3 (dual-core Xtensa 32-bit LX6 microprocessor)
Frekuensi Clock	Hingga 240 MHz
Memori Flash	4 MB (terintegrasi pada modul DOIT ESP32 DevKitC V1)
RAM	520 KB SRAM
Konektivitas Nirkabel	Wi-Fi 802.11 b/g/n (2.4 GHz), Bluetooth v4.2 BR/EDR & BLE
GPIO	36 pin (dengan fungsi multiplex seperti PWM, I ² C, SPI, UART, ADC, DAC, dll.)
ADC	12-bit, hingga 18 channel (input analog)
DAC	2 channel, 8-bit
PWM	Hingga 16 channel
Antarmuka Komunikasi	UART (3 buah), I ² C (2), SPI (4), I ² S (2), CAN, SDIO
Mode Daya Rendah	Light-sleep, Deep-sleep, Hibernation (konsumsi arus hingga ~10 μ A di deep sleep)
Tegangan Operasi	3.0 - 3.6 V
Arus Maksimum per Pin	40 mA
Suhu Operasional	- 40°C hingga +125°C

Pemrograman	Arduino IDE, ESP-IDF, MicroPython, PlatformIO
Fitur Keamanan	Secure boot, Flash encryption, Cryptographic hardware acceleration (AES, SHA, RSA)

2) Kelebihan dan kekurangan ESP32

Dibandingkan dengan banyak mikrokontroler lain, ESP32 menawarkan keunggulan konektivitas yang sangat baik, berkat modul Wi-Fi dan Bluetooth yang terintegrasi. Konektivitas ini memungkinkan ESP32 untuk terhubung ke jaringan nirkabel dan berkomunikasi dengan server atau platform cloud, yang sangat penting untuk aplikasi IoT (Internet of Things)[25]. Salah satu keunggulan utama lainnya adalah dukungan untuk dual-core processor Xtensa LX6, yang memungkinkan ESP32 untuk melakukan multitasking dengan efisien. Penggunaan prosesor ganda ini memungkinkan berbagai tugas dapat dijalankan secara simultan tanpa mengurangi kinerja keseluruhan.

ESP32 juga memiliki konsumsi daya rendah, yang membuatnya ideal untuk aplikasi yang membutuhkan operasi jangka panjang dengan sumber daya terbatas. Mode hemat daya seperti deep sleep dan light sleep memungkinkan perangkat ini untuk tetap siaga dengan konsumsi energi minimal. Namun, keunggulan lain yang patut disebutkan adalah fleksibilitas GPIO yang dimilikinya. ESP32 memiliki banyak pin General Purpose Input/Output (GPIO) yang dapat dikonfigurasi untuk berbagai fungsi seperti input, output, PWM, I2C, SPI, dan UART, memberikan kemampuan untuk berinteraksi dengan berbagai sensor dan aktuator[26]. Tidak hanya itu, ESP32 dilengkapi dengan ADC (Analog to Digital Converter) dan DAC (Digital to Analog Converter), yang memungkinkan mikrokontroler ini untuk membaca sinyal analog dari sensor serta menghasilkan sinyal analog dari data digital. Kehadiran komunitas pengembang yang besar dan aktif juga menjadi salah satu kelebihan ESP32. Dokumentasi yang luas dan banyaknya contoh kode yang tersedia memudahkan pengembangan dan debugging proyek, serta mempercepat proses pembelajaran bagi pemula.

Namun, selain banyak kelebihan yang telah disebutkan, ada juga beberapa kekurangan yang perlu diperhatikan. Meskipun ESP32 memiliki mode hemat daya yang sangat baik, konsumsi daya dalam mode aktif masih cukup tinggi dibandingkan dengan beberapa mikrokontroler lain seperti STM32 atau SAMD21. Hal ini dapat menjadi masalah dalam aplikasi yang sangat memperhitungkan

efisiensi energi. Kompleksitas konfigurasi juga menjadi salah satu tantangan dalam menggunakan ESP32. Karena mikrokontroler ini memiliki banyak fitur dan kemampuan, konfigurasi awal dan pengaturan bisa cukup kompleks, terutama bagi pemula. Mengelola berbagai modul dan periferan ESP32 memerlukan pemahaman yang mendalam tentang arsitektur perangkat.

Waktu boot yang lebih lama juga bisa menjadi kelemahan. ESP32 cenderung memiliki waktu boot yang lebih lama dibandingkan dengan mikrokontroler lain, yang disebabkan oleh proses inisialisasi yang kompleks dan pengaturan berbagai modul pada saat startup[27]. Keandalan konektivitas Bluetooth juga perlu diperhatikan. Meskipun ESP32 mendukung Bluetooth, terkadang ada masalah dengan stabilitas dan keandalan koneksi Bluetooth, terutama dalam lingkungan dengan banyak interferensi sinyal. Terbatasnya pin GPIO juga dapat menjadi kendala. Meskipun ESP32 memiliki banyak pin GPIO, beberapa pin memiliki fungsi khusus yang tidak bisa diubah, yang dapat membatasi fleksibilitas dalam beberapa aplikasi yang membutuhkan banyak pin I/O.

Table 3. Kelebihan dan kekurangan ESP32

Kelebihan	Kekurangan
1. Dual processor/Prosesor ganda multitasking yang efisien, menjalankan berbagai tugas tanpa mengurangi kinerja	1. Mengkonfigurasi dan mengatur ESP32 bisa cukup rumit terutama bagi pemula, memerlukan pemahaman mendalam tentang arsitektur perangkat
2. Mendukung Wi-Fi dan Bluetooth, ideal untuk aplikasi IoT yang memerlukan konektivitas jaringan dan komunikasi jarak dekat	2. Mengkonfigurasi dan mengatur ESP32 bisa cukup rumit terutama bagi pemula, memerlukan pemahaman mendalam tentang arsitektur perangkat
3. Mode hemat daya seperti deep sleep dan light sleep memungkinkan operasi jangka panjang dengan sumber daya terbatas	3. Konsumsi daya dalam mode aktif cukup tinggi dibandingkan dengan beberapa mikrokontroler lain seperti STM32 dan SAMD21
4. Banyak pin GPIO yang dapat di konfigurasi untuk berbagai fungsi seperti input, output, PWM, I2C, SPI, dan UART	4. Beberapa pin memiliki fungsi khusus yang tidak bisa diubah, membatasi fleksibilitas pada beberapa aplikasi
5. Dilengkapi dengan ADC dan DAC untuk membaca sinyal analog dan	5. Proses inisialisasi yang kompleks menyebabkan waktu boot yang

menghasilkan data analog dari data digital	lebih lama dibandingkan dengan mikrokontroler lain
6. Dokumentasi yang luas dan banyaknya contoh kode memudahkan pengembangan dan debugging proyek	

5. Sensor Infrared Proximity

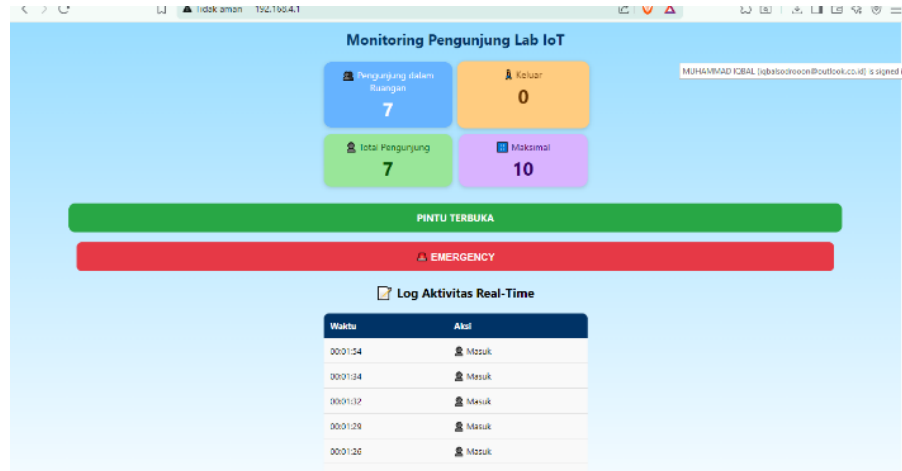


Gambar 5. sensor IR Proximity

Sensor proximity merupakan komponen elektronik yang mampu mendeteksi keberadaan objek tanpa kontak fisik dengan mengonversi perubahan gerakan atau posisi objek menjadi sinyal listrik. Jangkauan deteksinya bervariasi tergantung pada teknologi yang digunakan, seperti optik, ultrasonik, inframerah (IR), atau medan elektromagnetik. Setiap jenis sensor memiliki karakteristik teknis berbeda meliputi akurasi, sensitivitas, dan jarak deteksi, sehingga pemilihannya harus disesuaikan dengan kebutuhan aplikasi tertentu[5]. Sensor ini memiliki kemampuan utama untuk mendeteksi keberadaan objek di sekitarnya dengan tingkat akurasi yang baik. Seluruh komponen pada sensor berfungsi secara optimal selama proses uji coba, menunjukkan respons yang konsisten terhadap objek yang terdeteksi. Jangkauan deteksi sensor ini relatif pendek, yaitu berkisar antara 1 mm hingga beberapa sentimeter, tergantung pada jenis dan spesifikasinya.

Dari segi kebutuhan daya, sensor proximity umumnya beroperasi pada tegangan DC antara 10–30 volt. Namun, terdapat juga varian sensor yang dirancang untuk bekerja dengan tegangan AC lebih tinggi, yaitu dalam kisaran 100–200 volt[28]. Perbedaan tegangan kerja ini disesuaikan dengan kebutuhan aplikasi dan lingkungan penggunaannya, sehingga pemilihan sensor harus mempertimbangkan kesesuaian dengan sistem yang akan diimplementasikan.

6. Antarmuka Pemantauan Berbasis Web Server



Gambar 6. Tampilan monitor via web server

Pada penelitian ini, tampilan informasi jumlah pengunjung tidak lagi menggunakan perangkat keras eksternal seperti panel LED, melainkan diimplementasikan dalam bentuk antarmuka berbasis web yang di-host langsung oleh ESP32 dalam mode *Access Point* (AP). Pendekatan ini sejalan dengan prinsip pengembangan sistem IoT yang menekankan aksesibilitas universal dan antarmuka yang dapat dijangkau melalui perangkat umum tanpa ketergantungan pada modul tampilan khusus[29]. Sistem ini memanfaatkan kemampuan built-in web server ESP32 untuk menyajikan halaman HTML/CSS ringan yang dapat diakses melalui perangkat klien, baik smartphone maupun komputer yang terhubung ke jaringan WiFi yang dibuat oleh ESP32. Ketika pengguna membuka alamat <http://192.168.4.1> pada browser, sistem menampilkan informasi real-time berupa jumlah pengunjung saat ini, status kapasitas ruangan seperti normal, peringatan, atau overkapasitas, serta riwayat deteksi terakhir, sehingga memungkinkan pemantauan yang fleksibel tanpa ketergantungan pada lokasi fisik tertentu, sebagaimana diimplementasikan pula pada sistem keamanan ruangan berbasis ESP32 oleh Sofiyanto pada tahun 2025[30] dan Ziad pada tahun 2024[31].

Antarmuka web dirancang untuk responsif dan ringan, dengan pembaruan data dilakukan secara dinamis melalui AJAX polling ringan tanpa memerlukan refresh halaman manual. Strategi ini dipilih mengingat keterbatasan sumber daya memori pada ESP32, di mana mekanisme polling interval rendah terbukti lebih stabil dan hemat daya dibandingkan koneksi persisten berbasis WebSocket[32]. Seluruh logika tampilan dan pengolahan data berjalan secara lokal pada ESP32, tanpa ketergantungan pada cloud, basis data eksternal, atau perangkat antara, sesuai dengan prinsip edge computing dalam pengembangan sistem IoT yang andal dan

hemat sumber daya, sebagaimana ditegaskan oleh Ndruru pada tahun 2024[29]. dalam sistem pemantauan ruang puskesmas yang menempatkan seluruh proses pengolahan di sisi perangkat (*on-device processing*) untuk meminimalkan latensi dan meningkatkan privasi data. Pendekatan ini tidak hanya mengurangi biaya dan kompleksitas perangkat keras, tetapi juga meningkatkan kemudahan dalam pemeliharaan dan modifikasi tampilan, karena perubahan dapat dilakukan hanya dengan memperbarui kode HTML/CSS dalam firmware, tanpa perlu modifikasi hardware. Dalam implementasinya, antarmuka ini menjadi sarana utama untuk memantau jumlah pengunjung di Laboratorium IoT secara real-time dan intuitif.

7. Software Arduino IDE



Gambar 7. software arduino IDE

Menurut Deni Ahmad Jakaria dan Muhammad Rifki Fauzi, Arduino IDE (*integrated development environment*) merupakan sebuah software berbasis open source yang berfungsi sebagai platform pengembangan untuk menulis dan mengunggah kode program ke papan mikrokontroler Arduino. Perangkat lunak ini dikembangkan dengan bahasa pemrograman Java, sehingga bersifat cross-platform dan dapat dioperasikan pada berbagai sistem operasi, termasuk Windows, macOS, dan Linux[33].

8. Solenoid Lock



Gambar 8. Solenoid Lock

Penelitian ini berfokus pada penggunaan solenoid lock sebagai komponen utama dalam sistem pengamanan pintu ESP32. Solenoid lock merupakan aktuator elektromagnetik yang bekerja berdasarkan prinsip konversi energi listrik menjadi gerak mekanik linear untuk mengaktifkan mekanisme penguncian dan pembukaan pintu secara otomatis. Ketika arus listrik dialirkan melalui kumparan solenoid, medan magnet yang dihasilkan akan menarik plunger (batang feromagnetik) sehingga terjadi perpindahan posisi mekanis yang menggerakkan latch atau bolt pada sistem kunci. Karakteristik utama solenoid lock seperti respon cepat, konsumsi daya rendah dalam mode statis, serta kompatibilitas tinggi dengan sinyal logika mikrokontroler—menjadikannya pilihan dominan dalam implementasi sistem keamanan berbasis Internet of Things[22].

Integrasi solenoid lock dengan sensor magnetik dan mikrokontroler NodeMCU ESP8266 memungkinkan realisasi sistem keamanan rumah cerdas yang dapat dikendalikan melalui aplikasi Blynk berbasis smartphone, sekaligus memberikan notifikasi real-time terkait status pintu (terbuka/terkunci)[34]. Pendekatan serupa dikembangkan oleh Fakhrudin pada tahun 2024, yang mengimplementasikan ESP32 sebagai pengendali utama dengan aplikasi Blynk sebagai antarmuka pengguna, menunjukkan peningkatan stabilitas koneksi dan kemampuan multitasking dibandingkan penggunaan ESP8266[35].

Di sisi lain, Walunj dkk melakukan penelitian yang membuktikan bahwa Arduino Uno mampu mengendalikan solenoid lock secara andal melalui input keypad, meski terbatas dalam kapasitas konektivitas jaringan dan skalabilitas sistem[36]. Penelitian ini menggaris bawahi kebutuhan akan mikrokontroler dengan kemampuan pemrosesan paralel dan dukungan protokol komunikasi ganda (Wi-Fi dan Bluetooth) dalam aplikasi keamanan kritis seperti ruang sterilisasi, di mana ESP32 menunjukkan kinerja superior dalam hal latensi respon dan integrasi multi-sensor. Dengan demikian, pemanfaatan ESP32 dalam pengendalian solenoid lock merepresentasikan evolusi logis dari sistem keamanan konvensional menuju solusi yang adaptif, terukur, dan siap untuk integrasi dalam ekosistem smart environment.