

# Sistem Pintu Cerdas dengan *QR Code* berbasis *Internet of Things* sebagai Penerapan *Edge Computing*

MADAJABBAR PALAPA HAKIM, URAY RISTIAN, SUHARDI

Rekayasa Sistem Komputer, Universitas Tanjungpura, Indonesia  
Email: madajabbar@student.untan.ac.id

*Received* 11 Juli 2023 | *Revised* 9 Agustus 2023 | *Accepted* 11 September 2023

## ABSTRAK

*Sistem pintu cerdas telah merevolusi kontrol akses dengan metode seperti pengenalan sidik jari, kata sandi, dan QR Code. Namun, tantangan seperti ketergantungan pada internet, keterlambatan pengiriman data, dan pembatasan penjadwalan tetap ada. Penelitian ini menerapkan Edge Computing pada sistem pintu cerdas berbasis IoT dengan QR Code dan menambahkan fitur penjadwalan. Tujuannya adalah meningkatkan respons, kinerja, dan fleksibilitas dengan meminimalkan keterlambatan dan pembatasan jaringan, serta memungkinkan akses pengguna sesuai jadwal yang ditentukan. Hasil pengujian menunjukkan variasi waktu akses dan status pintu dalam kondisi normal dan offline, dengan rata-rata keterlambatan 3,35 detik (normal) dan 3,01 detik (offline). Penelitian ini memiliki potensi untuk pengembangan sistem pintu cerdas yang lebih efisien dan responsif menggunakan Edge Computing, memberikan kontribusi pada sistem keamanan canggih dan terpercaya dengan kemampuan penjadwalan terintegrasi di berbagai lingkungan.*

**Kata kunci:** *Edge Computing, Sistem Pintu Cerdas, QR Code, Internet of Things*

## ABSTRACT

*Smart door systems have revolutionized access control with methods like fingerprint recognition, passwords, and QR Codes. However, challenges such as internet dependency, data transmission delays, and limited scheduling persist. This study implements Edge Computing in an IoT-based QR Code smart door system and introduces a scheduling feature. The objective is to enhance responsiveness, performance, and flexibility by minimizing delays and network limitations, while allowing user access based on predetermined schedules. Test results show variable access times and door statuses under normal and offline conditions, with average delays of 3.35 seconds (normal) and 3.01 seconds (offline). This research offers potential for a more efficient and responsive smart door system using Edge Computing, contributing to advanced and reliable security systems with integrated scheduling capabilities across diverse environments.*

**Keywords:** *Edge Computing, Smart Door System, QR Code, Internet of Things*

## 1. PENDAHULUAN

Sistem pintu cerdas adalah sebuah kunci pintu yang pengoperasiannya dapat dilakukan dengan cara yang tidak biasa (**Lonika & Hariyanto, 2019**). Pengoperasian dapat menggunakan berbagai metode seperti sidik jari, *password*, ketukan, komunikasi *Bluetooth*, bahkan jaringan *internet*. Fungsi utama dari sistem pintu cerdas adalah membatasi akses pintu hanya kepada orang-orang yang memiliki izin tertentu. Selain metode tradisional seperti sidik jari dan *password* (**Iqbal, dkk, 2021**) (**Safitri, 2022**), sistem pintu cerdas juga memanfaatkan teknologi *QR Code* sebagai salah satu metode pengoperasian. *QR Code* adalah perkembangan dari *barcode* yang dikembangkan oleh Denso Wave dan dapat menyimpan berbagai informasi seperti URL, nomor telepon, pesan SMS, atau teks apapun (**Bayu, dkk, 2021**). Tujuan utama penggunaan *QR Code* adalah untuk menyampaikan informasi secara cepat dan mendapatkan respons yang cepat dan tepat (**Wasito, dkk, 2020**). *QR Code* dapat berperan sebagai kunci pada sistem pintu cerdas, memungkinkan akses terbatas hanya kepada orang-orang yang memiliki izin tertentu.

Pemilihan antara *QR Code* dan *password* angka konvensional memiliki pertimbangan tersendiri. *QR Code* memungkinkan akses dengan cepat dan mudah hanya dengan pemindaian kode, tanpa perlu menghafal *password* yang mungkin bisa ditemukan atau dicuri oleh pihak yang tidak berwenang. Selain itu, *QR Code* juga dapat diterapkan dalam situasi di mana pengguna seringkali lupa dengan *password* pada sistem. Selain itu, penggunaan sistem pintu cerdas dapat diatur melalui penjadwalan yang memungkinkan pengaturan akses terbatas berdasarkan jadwal yang telah ditentukan sebelumnya.

Penjadwalan sistem adalah proses untuk menentukan waktu dan orang yang berwenang melakukan tugas tertentu (**Bimo, dkk, 2020**). Penjadwalan juga menjadi salah satu komponen penting dalam sistem pintu cerdas. Dalam sistem ini, penjadwalan digunakan untuk mengatur waktu akses pengguna ke pintu, sehingga memungkinkan akses terbatas sesuai dengan jadwal yang telah ditentukan sebelumnya. Dengan adanya penjadwalan, sistem pintu cerdas dapat mengoptimalkan penggunaan ruangan dan memastikan bahwa hanya pengguna yang memiliki izin atau otorisasi yang dapat membuka pintu pada waktu-waktu tertentu. Namun, sistem pintu cerdas memiliki ketergantungan terhadap koneksi internet yang dapat menyebabkan gangguan pada beberapa fitur saat terjadi masalah pada internet atau *server*.

Sistem pintu cerdas memiliki ketergantungan pada koneksi internet yang dapat menyebabkan terjadinya *delay* dalam operasinya. *Delay* dapat terjadi saat sistem harus mengirimkan permintaan ke *server* pusat melalui koneksi internet, mempengaruhi responsivitas sistem pintu cerdas. Selain itu, dengan meningkatnya penggunaan sistem pintu cerdas, beban permintaan pada *server* juga meningkat, yang dapat memperparah masalah *delay*. Pada tahun 2022, (**Murti, dkk, 2022**) menggunakan sensor *Radio Frequency Identification (RFID)* pada sistem pintu cerdas. *Delay* yang didapatkan dari pada penelitian tersebut rata-rata sebesar 24,235 dan cenderung membutuhkan waktu yang lama untuk menunggu respons dari *cloud server*. Untuk mengatasi masalah tersebut, solusi yang dapat diimplementasikan adalah dengan menggunakan *Edge Computing*, di mana sebagian pemrosesan data dan pengambilan keputusan dilakukan secara lokal di perangkat *edge* (**Taheri & Deng, 2020**) (**Baktir, dkk, 2017**) yang berdekatan dengan pintu. Dengan menerapkan *Edge Computing*, *delay* akibat ketergantungan pada koneksi internet dapat dikurangi, meningkatkan responsivitas (**Elbamby, dkk, 2019**) (**Abouaomar, dkk, 2022**) sistem pintu cerdas.

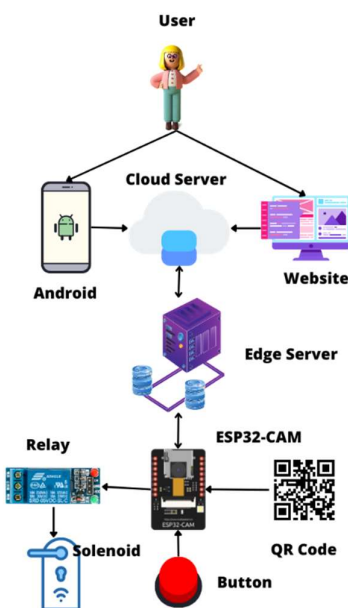
Tujuan dari penelitian ini adalah tiga aspek penting. Pertama, untuk mengetahui pengaruh penerapan *Edge Computing* terhadap penggunaan sistem pintu cerdas. Kedua, untuk menerapkan sistem pintu cerdas dalam menjadwalkan penggunaan ruangan. Ketiga, untuk

menganalisis *delay* yang dihasilkan dari penerapan *Edge Computing* pada sistem pintu cerdas. Dengan demikian, penelitian ini bertujuan untuk meningkatkan performa sistem pintu cerdas melalui penerapan teknologi *Edge Computing* serta memahami dampaknya terhadap penggunaan dan kinerja sistem pintu cerdas.

## 2. METODE

### 2.1 Deskripsi Sistem

Penelitian ini melibatkan pembuatan sistem pintu cerdas yang mengimplementasikan teknologi *Edge Computing* pada pintu cerdas dengan penguncian menggunakan *QR Code*. Sistem ini dirancang untuk memberikan kemudahan dalam mengunci pintu dan memberikan akses kepada pengguna yang memiliki izin. Dengan *QR Code* sebagai teknologi yang digunakan, pengguna hanya perlu memindai kode *QR* untuk membuka pintu, sehingga mempermudah proses akses masuk dan keluar ruangan. Selain itu, dengan menerapkan teknologi *Edge Computing*, sistem ini dapat memproses data secara cepat dan efisien di perangkat lokal, mengurangi ketergantungan pada pengiriman data ke *server* pusat, dan meningkatkan keamanan sistem secara keseluruhan. Gambaran deskripsi sistem pintu cerdas dapat dilihat pada Gambar 1.

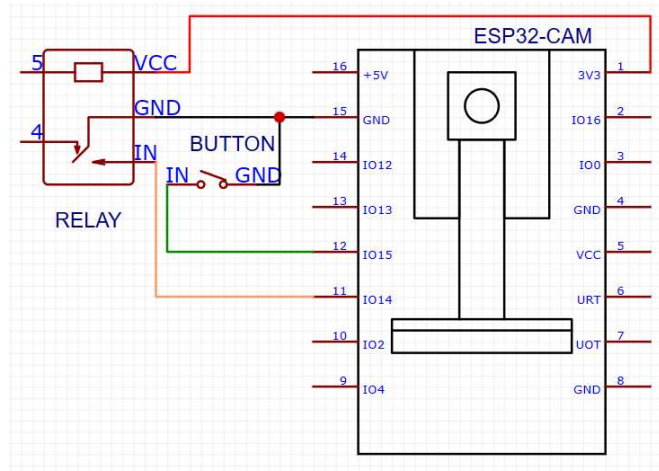


Gambar 1. Deskripsi Sistem

### 2.2 Perancangan Perangkat Keras

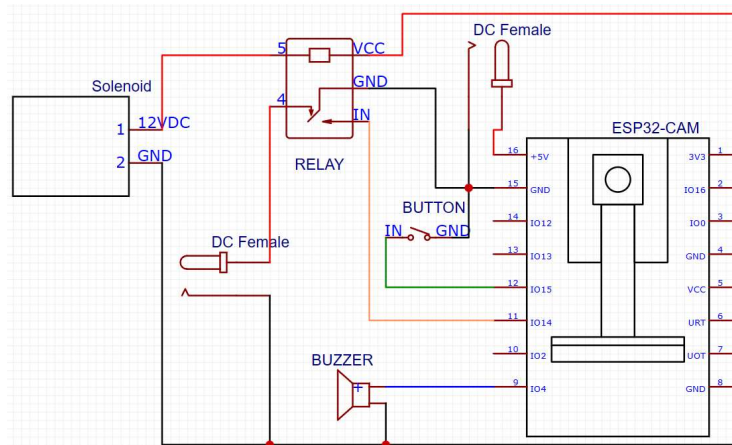
Perancangan perangkat keras dalam sistem pintu cerdas berbasis *QR Code* dan *Internet of Things* melibatkan beberapa komponen utama. Komponen tersebut meliputi ESP32-CAM sebagai perangkat pengambil gambar dan pengirim data ke *server*, modul relai sebagai pengontrol pintu, tombol sebagai perangkat pengendali manual, dan *buzzer* sebagai notifikasi suara. Sistem ini juga menggunakan komponen *Edge* sebagai perantara antara ESP32-CAM dan *server*. Perancangan sistem pemindaian *QR Code* melibatkan beberapa tahapan. Pengguna melakukan pemindaian *QR Code* menggunakan aplikasi *mobile* di *smartphone* pengguna (Widayati, 2017). Informasi dari *QR Code* dikirimkan ke *server* melalui *edge*, kemudian diteruskan ke mikrokontroler pada perangkat keras. Mikrokontroler memeriksa

keaslian informasi dan memberikan akses kepada pengguna untuk membuka pintu. Sistem juga memastikan keamanan dengan menggunakan *QR Code* yang unik dan tidak mudah dipalsukan. Integrasi dengan *database* pengguna memungkinkan sistem untuk mengenali *QR Code* yang telah diberikan akses dan memberikan izin akses tersebut. Gambaran perancangan sistem pemindaian *QR Code* dan sistem kendali tombol dapat dilihat pada Gambar 2.



**Gambar 2. Perancangan Sistem Pemindaian *QR Code* dan Sistem Kendali Tombol**

Perancangan keseluruhan perangkat keras mengintegrasikan sistem pemindaian *QR Code* dan sistem kendali tombol. ESP32-CAM berfungsi sebagai otak utama yang menghubungkan semua komponen dan menghubungkan perangkat keras ke *server* melalui *edge*. Pengguna dapat menggunakan *QR Code* atau tombol untuk mengendalikan pintu, dan buzzer memberikan notifikasi suara. Gambaran perancangan keseluruhan perangkat keras dapat dilihat pada Gambar 3.

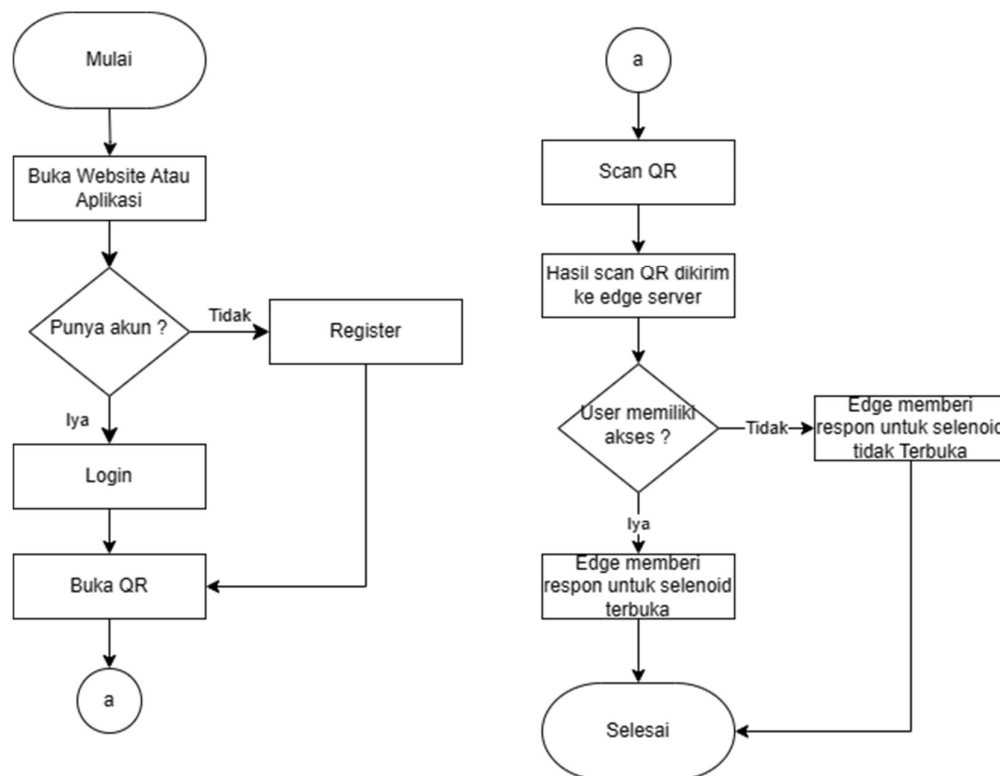


**Gambar 3. Perancangan Keseluruhan Perangkat Keras**

### 2.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak dalam penelitian ini berfungsi sebagai perantara antara mikrokontroler, *edge*, dan *server*. Berdasarkan perancangan perangkat keras, program perangkat lunak dikembangkan untuk mengolah informasi pengguna, memberikan akses, dan menghasilkan *QR Code*. Informasi dari mikrokontroler akan dikirimkan terlebih dahulu ke *edge*, kemudian diteruskan ke *server* jika tidak ada kendala dalam pengiriman data. Jika terjadi

kesalahan saat pengiriman data, data akan berstatus "pending" dan akan dikirimkan kembali setelah masalah teratasi. Data yang terkumpul di *server* akan ditampilkan melalui antarmuka berupa *website* dan aplikasi *Android*. Diagram alir perangkat lunak dapat dilihat pada Gambar 4.



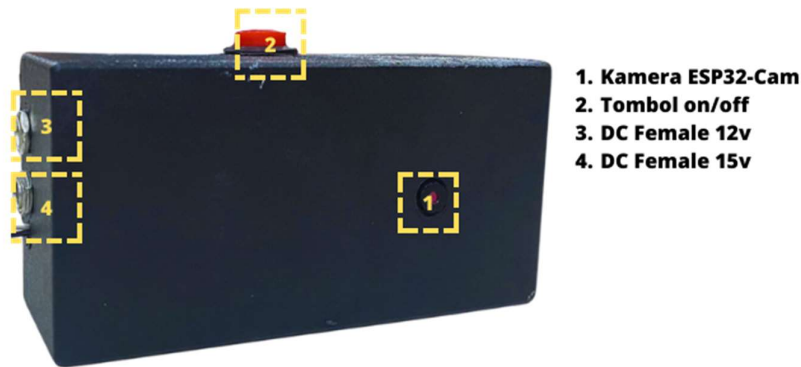
**Gambar 4. Diagram Alir Perangkat Lunak**

Perancangan sistem komunikasi antara *Edge Server* dan *Cloud Server* menggunakan protokol HTTP (**Daharmi, dkk, 2022**) untuk mengirimkan data dari *Edge Server* ke *Cloud Server*. Data yang diperoleh di *Edge Server* akan dikirimkan ke *Cloud Server* jika tidak ada gangguan (**Raj, 2021**). Jika terjadi gangguan, *Edge Server* akan menyimpan data tersebut dan akan dikirimkan setelah gangguan teratasi. Selain itu, perancangan sistem komunikasi antara *Edge Server* dan ESP32-CAM juga menggunakan protokol HTTP. ESP32-CAM akan mengirimkan data ke *Edge Server* melalui koneksi Wi-Fi. Data yang dikirimkan meliputi informasi pengguna yang memindai *QR Code*, status pintu, dan status pengiriman data. *Edge Server* akan memproses data tersebut dan mengirimkannya ke *Cloud Server*. Jika terjadi gangguan koneksi atau kesalahan saat mengirimkan data ke *Cloud Server*, data akan ditandai sebagai "pending" dan akan dikirimkan kembali setelah koneksi normal. Hal ini dilakukan untuk memastikan data tidak hilang dan dapat dikirimkan dengan sempurna. Selain itu, di *Edge Server* terdapat API yang berfungsi sebagai penghubung antara sistem pintu cerdas dan aplikasi *web* dan *mobile*. API ini digunakan untuk mengambil data dari *Cloud Server* dan menampilkannya pada aplikasi *web* dan *mobile*, seperti log pengguna, status pintu, dan informasi lainnya. Secara keseluruhan, perancangan sistem komunikasi berbasis *Edge Computing* ini memastikan keamanan dan integritas data, serta mempermudah pengguna dalam melihat informasi dan mengendalikan sistem pintu cerdas melalui aplikasi *web* atau *mobile* (**Pangestu, dkk, 2020**).

### 3. HASIL DAN PEMBAHASAN

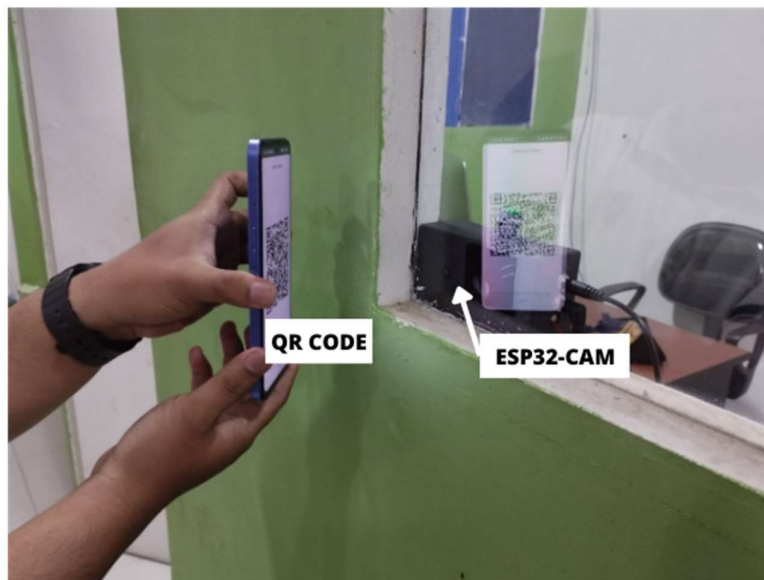
#### 3.1 Hasil Implementasi

Pada Gambar 5 ditampilkan hasil prototipe alat beserta keterangan.



**Gambar 5. Implementasi Purwarupa Alat**

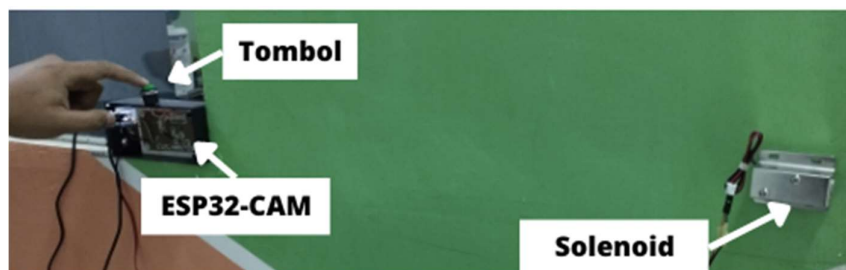
Dalam sistem ini, proses implementasi perangkat keras mencakup dua komponen utama, yaitu sistem pemindaian *QR Code* dan sistem kendali solenoid. Pada sistem pemindaian *QR Code*, dilakukan penunjukan *QR Code* pada ESP32-CAM dengan jarak 10 sampai 20 cm (Bayu, dkk, 2021). ESP32-CAM akan membaca *QR Code* tersebut dan memverifikasi otoritas untuk membuka pintu. Jika *QR Code* memiliki otoritas yang valid, solenoid pintu akan terbuka. Hasil implementasi sistem pemindaian *QR Code* pada sistem pintu cerdas dapat dilihat pada Gambar 6.



**Gambar 6. Implementasi Sistem Pemindaian *QR Code* pada Sistem Pintu Cerdas**

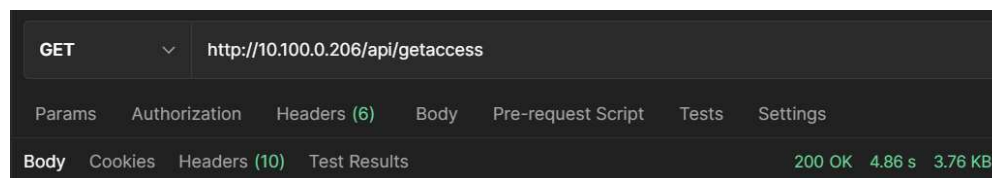
Sementara itu, dalam sistem kendali solenoid, dilakukan penghubungan solenoid pada pintu cerdas dengan sistem kendali tombol pada ESP32-CAM. Pengujian dilakukan dengan memberikan sinyal pada sistem kendali solenoid dan memastikan bahwa solenoid dapat

membuka atau menutup pintu dengan baik. Hasil implementasi sistem kendali tombol pada sistem pintu cerdas dapat dilihat pada Gambar 7.



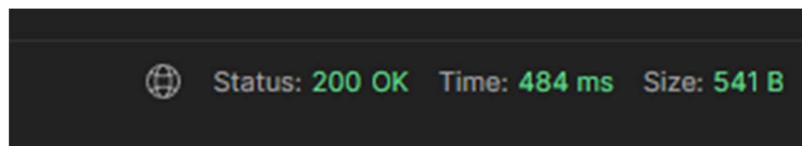
**Gambar 7. Implementasi Sistem Kendali Tombol pada Sistem Pintu Cerdas**

Dalam implementasi perangkat lunak, dilakukan pengembangan tampilan antarmuka berbasis *website* atau *android* dan integrasi dengan perangkat keras. Proses ini juga melibatkan sistem komunikasi antara *Edge Server* dan *Cloud Server*, serta antara *Edge Server* dan ESP32-CAM. Dalam implementasi sistem komunikasi antara *Edge Server* dan *Cloud Server*, dilakukan pengujian dengan melakukan *request API* pada *endpoint* yang telah disediakan. Tujuan pengujian ini adalah memastikan bahwa data yang diambil dari database dapat ditampilkan dengan benar pada halaman *website*. Hasil pengiriman data ke *Cloud Server* dapat dilihat pada Gambar 8.



**Gambar 8. Hasil Pengiriman Data ke Cloud Server**

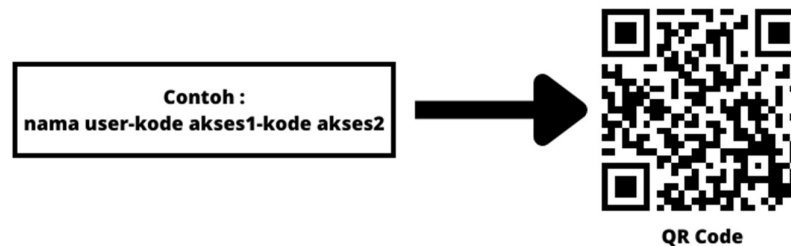
Pada Gambar 8, terlihat bahwa pengiriman data ke *Cloud Server* berhasil dengan status kode 200, yang menunjukkan berhasilnya permintaan. Waktu yang diperlukan untuk mengirim data adalah 4.86 detik, dan ukuran data yang terkirim adalah 3.76 KB. Hasil ini menunjukkan bahwa sistem komunikasi antara *Edge Server* dan *Cloud Server* berjalan dengan baik, dan data dapat dikirim melalui koneksi yang tersedia. Selanjutnya, pada implementasi sistem komunikasi antara *Edge Server* dan ESP32-CAM, dilakukan pengiriman data sensor dari ESP32-CAM ke *server edge*, serta menerima perintah dari *website* untuk mengontrol pintu cerdas. Tujuan dari pengujian ini adalah memastikan bahwa data dapat dikirimkan dan diterima dengan baik, serta pintu cerdas dapat dikendalikan melalui sistem ini. Hasil pengiriman data ke *Edge Server* dapat dilihat pada Gambar 9.



**Gambar 9. Hasil Pengiriman Data ke Edge Server**

Pada Gambar 9, terlihat bahwa pengiriman data ke *Edge Server* berhasil dengan status kode 200, yang menunjukkan berhasilnya permintaan. Waktu yang diperlukan untuk mengirim data adalah 484 milidetik, dan ukuran data yang terkirim adalah 541 B. Hasil ini menunjukkan

bahwa sistem komunikasi antara ESP32-CAM dan *Edge Server* berjalan dengan baik, dan data sensor dapat dikirim dan diterima melalui koneksi yang tersedia. Terakhir, implementasi pembuatan *QR Code* dilakukan untuk menghasilkan *QR Code* yang digunakan sebagai kunci untuk membuka pintu. *QR Code* ini dibuat dengan mengambil nama pengguna dan akses yang dimiliki pengguna sebagai isi dari *QR Code*. Pengguna yang telah memiliki akun akan diberikan akses oleh admin, dan sistem secara otomatis menambahkan kode akses pada *QR Code* pengguna. Pengguna yang baru membuat akun awalnya tidak memiliki akses ke pintu manapun, sehingga tampilan halaman pengguna akan kosong. Namun, setelah diberikan akses oleh admin, tampilan halaman pengguna akan memuat *QR Code* yang dapat digunakan untuk membuka pintu. Proses kode akses ke *QR Code* dapat dilihat pada Gambar 10.



**Gambar 10. Proses Kode Akses ke *QR Code***

*QR Code* yang mengalami degenerasi akan selalu bervariasi setiap kali terjadi perubahan akses pada sistem. Setiap kali ada penambahan atau pengurangan akses pengguna, *QR Code* yang dihasilkan akan berbeda. Hal ini dikarenakan informasi yang terkandung dalam *QR Code* mencerminkan kondisi akses terbaru, termasuk informasi tentang pengguna yang memiliki izin serta waktu akses yang diberikan. Oleh karena itu, jika terjadi perubahan dalam akses pengguna, *QR Code* akan mengalami perubahan sesuai dengan kondisi akses yang baru.

### 3.2 Hasil Pengujian

Tabel 1 merupakan hasil pengujian sistem pintu cerdas dengan koneksi internet. Tabel tersebut menampilkan data pengujian yang dilakukan pada berbagai hari dalam seminggu. Setiap baris dalam tabel mencakup informasi seperti hari, ruangan, pengguna, waktu pengujian, akses (ada/tidak ada), kondisi pintu (terbuka/tidak terbuka), keterangan, dan *delay* (penundaan) dalam detik. Dalam pengujian ini, terdapat 20 kasus pengujian yang dilakukan pada hari Senin, Selasa, Rabu, Kamis, dan Jumat. Setiap pengujian melibatkan pengguna tertentu yang mencoba mengakses pintu dengan kondisi akses (ada/tidak ada). Selanjutnya, dilakukan pengecekan terhadap kondisi pintu, apakah terbuka atau tidak terbuka, dan dicatat keterangan hasil pengujian. Selain itu, tabel juga mencantumkan nilai *delay* (penundaan) dalam detik untuk setiap pengujian. *Delay* ini menggambarkan waktu yang diperlukan oleh sistem pintu cerdas dalam merespons dan mengubah kondisi pintu sesuai dengan instruksi pengguna. Nilai *delay* yang tercatat bervariasi antara 1,12 detik hingga 6,23 detik, dengan rata-rata *delay* total sebesar 3,9 detik.



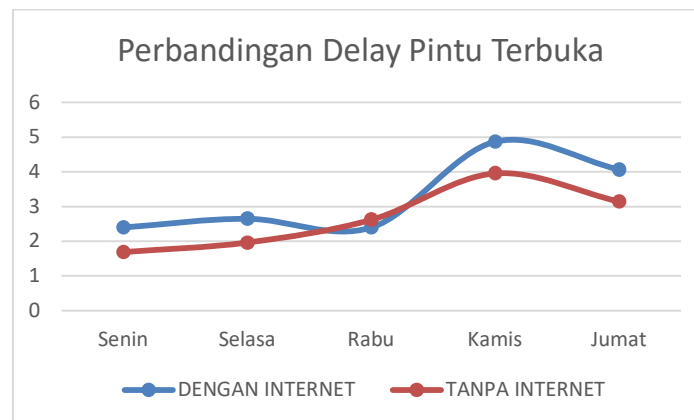
**Tabel 1. Pengujian Sistem Pintu Cerdas dengan Internet**

No	Hari	Ruangan	User	Waktu Pengujian	Akses	Kondisi Pintu	Keterangan	Delay (detik)
1	Senin	Pintu 1	User1	12:32	Ada	Terbuka	Berhasil	2,6
2	Senin	Pintu 1	User2	16:57	Ada	Terbuka	Berhasil	2,2
3	Senin	Pintu 1	User4	11:02	Tidak Ada	Tidak Terbuka	Berhasil	2,9
4	Senin	Pintu 1	User5	13:24	Tidak Ada	Tidak Terbuka	Berhasil	3,3
5	Selasa	Pintu 1	Admin	14:17	Ada	Terbuka	Berhasil	2,1
6	Selasa	Pintu 1	User3	16:23	Ada	Terbuka	Berhasil	3,2
7	Selasa	Pintu 1	User3	12:09	Tidak Ada	Tidak Terbuka	Berhasil	4,39
8	Selasa	Pintu 1	User4	14:46	Tidak Ada	Tidak Terbuka	Berhasil	3,7
9	Rabu	Pintu 1	Admin	15:57	Ada	Terbuka	Berhasil	2,44
10	Rabu	Pintu 1	User5	11:12	Ada	Terbuka	Berhasil	2,35
11	Rabu	Pintu 1	User1	12:00	Tidak Ada	Tidak Terbuka	Berhasil	2,59
12	Rabu	Pintu 1	User5	17:02	Tidak Ada	Tidak Terbuka	Berhasil	2,33
13	Kamis	Pintu 1	Admin	13:20	Ada	Terbuka	Berhasil	5,23
14	Kamis	Pintu 1	User3	08:51	Ada	Terbuka	Berhasil	4,51
15	Kamis	Pintu 1	User3	10:29	Tidak Ada	Tidak Terbuka	Berhasil	1,12
16	Kamis	Pintu 1	User7	16:09	Tidak Ada	Tidak Terbuka	Berhasil	1,78
17	Jumat	Pintu 1	User2	15:32	Ada	Terbuka	Berhasil	4,61
18	Jumat	Pintu 1	User5	09:20	Ada	Terbuka	Berhasil	3,52
19	Jumat	Pintu 1	User1	16:39	Tidak Ada	Tidak Terbuka	Berhasil	6,23
20	Jumat	Pintu 1	User2	16:47	Tidak Ada	Tidak Terbuka	Berhasil	5,66

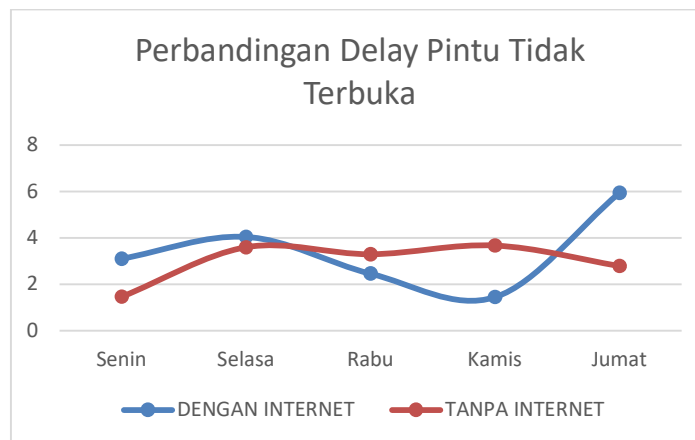
Tabel 1 merupakan hasil pengujian sistem pintu cerdas tanpa koneksi internet. Tabel ini menampilkan data pengujian yang dilakukan pada hari-hari yang sama seperti dalam Tabel 1. Setiap baris dalam tabel mencakup informasi seperti hari, ruangan, pengguna, waktu pengujian, akses (ada/tidak ada), kondisi pintu (terbuka/tidak terbuka), keterangan, dan *delay* (penundaan) dalam detik. Pengujian dilakukan dengan mengambil sampel dari 20 kasus yang melibatkan pengguna dengan akses dan kondisi pintu yang berbeda. Pada pengujian ini, tidak ada koneksi internet yang tersedia, sehingga sistem pintu cerdas diuji dalam kondisi *offline*. Data dalam Tabel 2 memberikan gambaran tentang performa sistem pintu cerdas ketika tidak terhubung dengan internet. Dalam setiap kasus pengujian, waktu respons sistem terhadap instruksi pengguna dicatat sebagai nilai *delay*. Nilai *delay* tersebut berkisar antara 1,29 detik hingga 4,23 detik, dengan rata-rata *delay* total sebesar 2,8 detik.

**Tabel 2. Pengujian Sistem Pintu Cerdas Tanpa Internet**

No	Hari	Ruangan	User	Waktu Pengujian	Akses	Kondisi Pintu	Keterangan	Delay (detik)
1	Senin	Pintu 1	Admin	10:20	Ada	Terbuka	Berhasil	1,49
2	Senin	Pintu 1	User2	16:13	Ada	Terbuka	Berhasil	1,88
3	Senin	Pintu 1	User1	16:17	Tidak Ada	Tidak Terbuka	Berhasil	1,29
4	Senin	Pintu 1	User4	10:33	Tidak Ada	Tidak Terbuka	Berhasil	1,64
5	Selasa	Pintu 1	Admin	15.03	Ada	Terbuka	Berhasil	1,77
6	Selasa	Pintu 1	User3	14.41	Ada	Terbuka	Berhasil	2,15
7	Selasa	Pintu 1	User2	13:56	Tidak Ada	Tidak Terbuka	Berhasil	3,87
8	Selasa	Pintu 1	User4	14:46	Tidak Ada	Tidak Terbuka	Berhasil	3,32
9	Rabu	Pintu 1	User1	08:02	Ada	Terbuka	Berhasil	3,07
10	Rabu	Pintu 1	User6	16.43	Ada	Terbuka	Berhasil	2,17
11	Rabu	Pintu 1	User1	12:00	Tidak Ada	Tidak Terbuka	Berhasil	3,33
12	Rabu	Pintu 1	User4	11:20	Tidak Ada	Tidak Terbuka	Berhasil	3,27
13	Kamis	Pintu 1	User3	08:51	Ada	Terbuka	Berhasil	4,23
14	Kamis	Pintu 1	User7	10:32	Ada	Terbuka	Berhasil	3,68
15	Kamis	Pintu 1	User1	10:41	Tidak Ada	Tidak Terbuka	Berhasil	4,08
16	Kamis	Pintu 1	User3	10:29	Tidak Ada	Tidak Terbuka	Berhasil	3,27
17	Jumat	Pintu 1	User1	13.33	Ada	Terbuka	Berhasil	2,62
18	Jumat	Pintu 1	User5	09:20	Ada	Terbuka	Berhasil	3,67
19	Jumat	Pintu 1	User1	16:39	Tidak Ada	Tidak Terbuka	Berhasil	3,82
20	Jumat	Pintu 1	User5	10:35	Tidak Ada	Tidak Terbuka	Berhasil	1,76

**Gambar 11. Perbandingan *Delay* Pintu Terbuka**

Gambar 11 menunjukkan perbandingan *delays* saat pintu terbuka antara pengujian dengan dan tanpa koneksi internet. Dalam sistem ini, pengujian menunjukkan bahwa pintu berhasil terbuka dengan rata-rata *delay* sekitar 2-6 detik ketika terdapat koneksi internet, sedangkan pada pengujian tanpa internet rata-rata *delay* sekitar 1-4 detik.



**Gambar 12.** Perbandingan *Delay* Pintu Tidak Terbuka

Gambar 12 menunjukkan perbandingan *delay* saat pintu tidak terbuka antara pengujian dengan dan tanpa koneksi internet. Dalam sistem ini, pengujian menunjukkan bahwa pintu berhasil tidak terbuka dengan rata-rata *delay* sekitar 2-6 detik ketika terdapat koneksi internet, sedangkan pada pengujian tanpa internet rata-rata *delay* sekitar 1-4 detik.

#### 4. KESIMPULAN

Penelitian ini bertujuan menerapkan *Edge Computing* pada sistem pintu cerdas untuk mengurangi ketergantungan pada koneksi internet. Dengan menggunakan *Edge Server* lokal, sistem ini dapat berfungsi secara optimal bahkan dalam situasi gangguan jaringan. Selain itu, sistem ini juga memiliki kemampuan penjadwalan penggunaan ruangan. Hasil pengujian menunjukkan bahwa rata-rata *delay* respon sistem pintu cerdas adalah 3,43 detik untuk respons terbuka dan 3,26 detik untuk respons tertutup dalam kondisi jaringan yang baik. Ketika terjadi masalah internet, rata-rata *delay* untuk respons terbuka adalah 3,30 detik dan 2,73 detik untuk respons tertutup. Meskipun ada potensi *delay* sebesar 3,9 detik, sistem berbasis QR code memiliki keunggulan dalam kemudahan penggunaan dan keamanan yang lebih tinggi. Pengguna hanya perlu memindai *QR code* untuk mengakses pintu tanpa khawatir lupa sandi atau menunggu kode OTP. *QR code* juga bisa diperbaharui berkala sehingga dapat meningkatkan keamanan sistem pintu cerdas.

#### DAFTAR RUJUKAN

- Abouaomar, A., Cherkaoui, S., Mlika, Z., & Kobbane, A. (2022). *Resource Provisioning in Edge Computing for Latency Sensitive Applications*. <http://arxiv.org/abs/2201.11837>
- Baktir, A. C., Ozgovde, A., & Ersoy, C. (2017). How Can Edge Computing Benefit from Software-Defined Networking: A Survey, Use Cases, and Future Directions. Dalam *IEEE Communications Surveys and Tutorials* (Vol. 19, Nomor 4, hlm. 2359–2391). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/COMST.2017.2717482>

- Bayu, R. B. S., Astutik, R. P., & Irawan, D. (2021). Rancang Bangun Smarthome Berbasis Qr Code Dengan Mikrokontroler Module Esp32. *JASEE Journal of Application and Science on Electrical Engineering*, 2(01), 47–60. <https://doi.org/10.31328/jasee.v2i01.60>
- Bimo Pratama, A., Ryana Suchendra, D., & Indah Hapsari, G. S. (2020). Sistem Penjadwalan Dan Riwayat Keamanan Pintu Pada Lab Ens Fit Scheduling And Log Of Door Security System In Lab Ens Fit. *e-Proceeding of Applied Science*, 6(2), 3585–3593.
- Daharmi, R. R., Bhawiyuga, A., & Basuki, A. (2022). *Pengembangan IoT Cloud Platform berbasis pada Layanan Serverless Computing* (Vol. 6, Nomor 12). <http://j-ptiik.ub.ac.id>
- Elbamby, M. S., Perfecto, C., Liu, C.-F., Park, J., Samarakoon, S., Chen, X., & Bennis, M. (2019). *Wireless Edge Computing with Latency and Reliability Guarantees*. <http://arxiv.org/abs/1905.05316>
- Iqbal, M., Hardyanto, R. H., & Ciptadi, P. W. (2021). Sistem Keamanan Ganda Menggunakan Fingerprint dan Keypad Pada Pintu Rumah (Smart Security System). *Seri Prosiding Seminar Nasional Dinamika Informatika*, 5(1).
- Lonika, T., & Hariyanto, S. (2019). Simulasi Smart Door Lock Berbasis QR Code Menggunakan Arduino Uno Pada Penyewaan Apartemen Online. *Jurnal ALGOR*, 1(1), 9–15.
- Murti, T. N., Ruslianto, I., & Ristian, U. (2022). Implementasi Sistem Kendali dan Monitoring Keamanan Pintu Berbasis IoT Menggunakan Perangkat Mobile. *JURIKOM (Jurnal Riset Komputer)*, 9(6), 1760–1769. <https://doi.org/10.30865/jurikom.v9i6.5032>
- Pangestu, A., Ziky Iftikhor, A., & Bakri, M. (2020). Sistem Rumah Cerdas Berbasis Iot Dengan Mikrokontroler Nodemcu Dan Aplikasi Telegram. Dalam *JTIKOM* (Vol. 1, Nomor 1).
- Raj, J. S. (2021). Optimized Mobile Edge Computing Framework for IoT based Medical Sensor Network Nodes. *Journal of Ubiquitous Computing and Communication Technologies*, 3(1), 33–42. <https://doi.org/10.36548/jucct.2021.1.004>
- Safitri, F. E. (2022). Rancang Bangun Pengaman Pintu Otomatis Menggunakan Sidik Jari (Fingerprint) dan Password Berbasis Arduino. *JTEIN: Jurnal Teknik Elektro Indonesia*, 3(2). <https://doi.org/10.24036/jtein.v3i2.269>
- Taheri, J., & Deng, S. (2020). Edge computing: Models, technologies and applications. *Edge Computing, March 2021*, 1–453. <https://doi.org/10.1049/PBPC033E>
- Wasito, B., Novian, H., Program, S. P., & Informatika, S. T. (2020). Pemanfaatan Quick Response Code Untuk Pencarian Informasi Produk Berbasis Mobile. *Jurnal Informatika dan Bisnis*, 9(2).
- Widayati, Y. T. (2017). Aplikasi Teknologi Qr ( Quick Response ) Code Implementasi Yang Universal. *KOMPUTAKI*, 3(1).