

OPTIMASI KINERJA ABSENSI RFID SERVERLESS MENGGUNAKAN BATCH PROCESSING, QUEUEING DAN RATE LIMIT MANAGEMENT UNTUK INTEGRASI CLOUD

Ahmad Khaerudin¹, Martanto², Denni Pratama³, Indra Wiguna Marthanu⁴.

Program Studi Teknik Informatika¹
Program Studi Manajemen Informatika²
Program Studi Komputerisasi Akuntansi^{3,4}

STMIK IKMI Cirebon
<https://ikmi.ac.id/page/18/?lang=de>
ahmaddn.dina55@gmail.com

(*) Corresponding Author : ahmaddn.dina55@gmail.com
Published : 20 Juli 2026

Abstract—Fast, accurate, and cost-effective attendance recording is an essential need for educational institutions. At SMK Plus As-Salafiyah, the existing attendance systems, including manual recording and QR Code scanning, have proven to be slow and prone to incomplete data recapitulation. On the other hand, the implementation of Real-Time Internet of Things (IoT) using free serverless platforms (Google Apps Script) frequently fails due to API rate limits (HTTP 429 Error). This study aims to design an IoT architecture based on the ESP32 Dual-Core microcontroller to address these issues. The system applies a computational load-balancing (multitasking) strategy; Core 1 exclusively handles RFID sensor reading and local data storage into MicroSD memory (Non-Volatile Queueing), while Core 0 operates asynchronously to transmit the queued data to a Google Sheets database using the Batch Processing technique. Quantitative performance testing results show that separating these computational processes creates a highly ideal trade-off. Local scanning speed (latency) reaches 1 to 2 seconds per student, which is significantly more efficient than the previous systems. Furthermore, the batching method successfully eliminates API rejection, enabling the system to achieve a 100% Packet Delivery Ratio (PDR) with zero data loss. In conclusion, this Dual-Core and Batch Processing architecture is proven robust as a reliable Zero-Cost Serverless attendance infrastructure solution for implementation in school environments.

Keywords : RFID Attendance, ESP32 Dual-Core, Batch Processing, MicroSD, Zero-Cost Serverless.

Abstrak—Pencatatan absensi yang cepat, akurat, dan minim biaya merupakan kebutuhan esensial bagi institusi pendidikan. Di SMK Plus As-Salafiyah, sistem absensi eksisting berupa pencatatan manual dan pemindaian QR Code terbukti lambat dan rentan terhadap rekapitulasi data yang tidak utuh. Di sisi lain, implementasi *Internet of Things* (IoT) dengan pengiriman data *Real-Time* menggunakan platform *serverless* gratis (Google Apps Script) sering mengalami kegagalan akibat pembatasan laju panggilan API (*Error HTTP 429*). Penelitian ini bertujuan untuk merancang arsitektur perangkat IoT berbasis mikrokontroler ESP32 *Dual-Core* guna memecahkan masalah tersebut. Sistem menerapkan strategi pembagian beban komputasi (*multitasking*); Core 1 secara eksklusif menangani pembacaan sensor RFID dan menyimpan data ke dalam memori MicroSD (*Non-Volatile Queueing*), sementara Core 0 beroperasi secara asinkron untuk mengirimkan tumpukan data ke *database* Google Sheets menggunakan teknik *Batch Processing*. Hasil pengujian kinerja secara kuantitatif menunjukkan bahwa pemisahan proses komputasi ini menghasilkan titik kompromi (*trade-off*) yang sangat ideal. Kecepatan pemindaian lokal (*latency*) mencapai 1 hingga 2 detik per siswa, jauh lebih efisien dibandingkan sistem sebelumnya. Selain itu, metode *batching* berhasil mengeliminasi penolakan API sehingga sistem mencapai tingkat keberhasilan pengiriman atau *Packet Delivery Ratio* (PDR) sebesar 100% tanpa adanya kehilangan data (*data loss*). Kesimpulannya, arsitektur *Dual-Core* dan *Batch Processing* ini terbukti tangguh sebagai solusi infrastruktur absensi tanpa biaya *server* (*Zero-Cost Serverless*) yang andal untuk diimplementasikan di lingkungan sekolah.

Kata Kunci: Absensi RFID, ESP32 *Dual-Core*, *Batch Processing*, MicroSD, *Zero-Cost Serverless*.

INTRODUCTION

Otomatisasi data berbasis IoT terbukti meningkatkan efisiensi pelacakan kehadiran dan mengurangi kesalahan manusia dengan memungkinkan identifikasi pasif secara *Real-Time*, agregasi otomatis, dan integrasi aman dengan *cloud*. Hal ini diperkuat oleh berbagai penelitian terbaru yang bersifat terbuka. Kajian tentang sensor RFID menyoroti keunggulan *tag* yang murah, tanpa baterai, dan kemampuan pembacaan otomatis yang cepat sehingga menghilangkan kebutuhan tanda tangan manual maupun manipulasi foto/QR [1], [2]. Implementasi sistem IoT untuk absensi menunjukkan keunggulan seperti *check-in* tanpa sentuh, pencatatan waktu kehadiran berbasis *cloud*, serta pengurangan risiko *proxy* dan kecurangan dibandingkan metode QR atau manual [3].

Meskipun demikian, Penelitian-penelitian terbaru yang bersifat *open-access* menyoroti berbagai tantangan teknis dan operasional dalam penerapan sistem absensi IoT-RFID yang terintegrasi dengan platform *serverless*. Masalah utamanya meliputi *cold start* dan *warm start*, keterbatasan sumber daya di perangkat *edge*, variabilitas latensi dan *throughput*, manajemen antrian (*queueing/backpressure*), serta pembatasan tingkat permintaan (*rate-limit Management*) di layanan *cloud* (Benedetti et al., 2021; Decker et al., 2022; Kjorveziroski et al., 2021; Shojaee Rad & Ghobaei-Arani, 2024; Shwe & Aritsugi, 2024). Studi kasus di SMK Plus As-Salafiyah menunjukkan bahwa sistem absensi berbasis QR Code yang ada saat ini mengalami kegagalan implementasi, ditandai dengan aplikasi yang "lambat dan sering terhenti". Hal ini mengakibatkan guru piket malas melakukan pemindaian, dan puncaknya adalah "banyak data rekap bulanan yang kosong", yang menunjukkan kegagalan *pipeline* data dalam menjamin integritas pencatatan.

(Benedetti et al., 2021) secara eksperimental mengukur perbedaan latensi antara *warm* dan *cold start* pada lingkungan *edge*. [NO_PRINTED_FORM] [5] memetakan arsitektur *serverless edge* dan arah penelitian terbuka. [NO_PRINTED_FORM] [6] melakukan *benchmark* pada FaaS berbasis Kubernetes sumber terbuka dan menemukan hambatan pada *throughput* dan penjadwalan. Studi desain perangkat berbasis ESP32 dan *edge device* oleh [NO_PRINTED_FORM] [9] menunjukkan bahwa mikrokontroler berbiaya rendah dapat memfasilitasi pencatatan lokal yang andal, *buffering*, dan transmisi untuk kejadian dengan

frekuensi tinggi. Shojaee Rad & Ghobaei-Arani (2024) menyusun taksonomi pendekatan *serverless data pipeline* yang menekankan pentingnya orkestrasi, toleransi kesalahan, dan desain yang sadar terhadap batas *rate-limit*.

Berdasarkan identifikasi masalah teknis (*rate-limit, throughput*) dan masalah nyata (data absensi kosong) tersebut, penelitian ini memiliki tujuan yang signifikan. Temuan-temuan penelitian sebelumnya menjadi dasar untuk merumuskan solusi. "Tujuan utama penelitian ini adalah mengembangkan dan mengoptimalkan *pipeline* pencatatan data RFID berbasis **Batch Processing (Reliable-Time)** yang andal untuk mengatasi kegagalan sistem QR Code di SMK Plus As-Salafiyah. Signifikansi penelitian ini terletak pada penerapan strategi spesifik berupa antrean fisik (*MicroSD queueing*) untuk memangkas waktu *latency* lokal secara instan sekaligus meningkatkan keandalan sistem absensi *serverless* secara mutlak.

Untuk mencapai tujuan tersebut, penelitian ini menerapkan arsitektur *pipeline* data yang spesifik. Pendekatan ini menggunakan sinkronisasi data berbasis *cloud* yang tahan terhadap gangguan koneksi melalui sinkronisasi offline-to-online. Analisis sistematis (Benedetti et al., 2021; Cassel et al., 2022) menunjukkan bahwa pendekatan *serverless/FaaS* meningkatkan skalabilitas dan efisiensi biaya untuk kejadian absensi yang bersifat *bursty*, meskipun tetap menghadirkan tantangan *cold start* dan orkestrasi yang dapat diatasi melalui mekanisme antrian dan manajemen *rate-limit*. Oleh karena itu, penelitian ini mengimplementasikan strategi *batch processing, queueing Mechanism* (via *MicroSD buffer*), dan *rate-limit Management* dalam desain *microservice* berbasis *edge-cloud* (ESP32 ke Google Apps Script) untuk menjaga integritas data selama koneksi tidak stabil.

Meskipun teknologi IoT menawarkan otomatisasi dan saat ini telah banyak tersedia mesin absensi RFID komersial yang dijual bebas di pasaran, solusi komersial tersebut umumnya kurang ideal bagi institusi pendidikan menengah ke bawah. Mesin absensi komersial mayoritas memiliki ekosistem yang tertutup (*closed-source*), membutuhkan investasi tambahan untuk pengadaan server lokal, atau mewajibkan biaya langganan *cloud (subscription)* bulanan. Sebagai alternatif yang hemat biaya, institusi sering beralih pada platform *serverless* gratis bersumber daya rendah (seperti Google Apps Script/Sheets).

Namun, terdapat kesenjangan (Research Gap) yang signifikan ketika alternatif teknologi tersebut diterapkan. Penelitian terdahulu mayoritas berfokus pada perancangan arsitektur pengiriman data secara *Real-Time* murni ke platform *cloud* gratis. Sayangnya, arsitektur *Real-*

Time ini justru sering gagal ketika dihadapkan pada aturan batas laju API (rate-limit/HTTP 429) dan koneksi internet sekolah yang fluktuatif, yang berujung pada hilangnya data rekapitulasi (data loss).

Kondisi ini terbukti pada studi kasus di SMK Plus As-Salafiyah. Pihak sekolah enggan menggunakan mesin absensi komersial karena tingginya kendala biaya pengadaan, biaya langganan bulanan, dan ekosistem server yang tertutup. Di sisi lain, metode alternatif yang digunakan sebelumnya terbukti tidak efektif. Pencatatan manual di kelas sangat rentan human error, sementara aplikasi QR Code di gerbang dinilai sangat lambat, sering mengalami crash, dan menyisakan banyak data rekap bulanan yang kosong akibat gagal terkirim.

Oleh karena itu, urgensi penelitian ini adalah menemukan titik kompromi (trade-off) arsitektur yang tepat antara kecepatan respons pemindaian di gerbang (latency lokal) dengan keandalan pengiriman data ke cloud. Kegagalan pencatatan absensi sama sekali tidak bisa ditoleransi dalam administrasi sekolah. Jika arsitektur aliran data tidak segera dioptimasi dari metode Real-Time menjadi metode berbasis antrean fisik (MicroSD queueing) dan pengiriman kelompok (Batch Processing), maka sekolah akan terus terjebak dalam krisis integritas data kehadiran.

MATERIALS AND METHODS

Tahapan penelitian merupakan bagian penting dalam proses pengembangan sistem berbasis Internet of Things (IoT), khususnya pada implementasi sistem absensi RFID serverless yang terintegrasi dengan cloud computing. Setiap tahapan dirancang secara sistematis agar proses pengumpulan data, pengolahan, implementasi perangkat, hingga evaluasi sistem dapat berjalan secara optimal. Selain itu, alur penelitian yang terstruktur membantu peneliti dalam meminimalkan kesalahan selama proses pengembangan sistem serta memastikan bahwa solusi yang dihasilkan mampu menjawab permasalahan yang ada. Dalam penelitian ini, tahapan penelitian difokuskan pada optimasi pengiriman data absensi menggunakan teknik batch processing, queueing, dan rate limit management berbasis ESP32 Dual-Core. Adapun alur tahapan penelitian yang digunakan dapat dilihat pada gambar berikut.

Gambar tersebut menunjukkan alur tahapan penelitian dalam pengembangan sistem absensi RFID serverless berbasis Internet of Things (IoT) menggunakan mikrokontroler ESP32 Dual-Core. Tahapan pertama dimulai dari observasi lapangan dan analisis kebutuhan yang dilakukan di SMK Plus As-Salafiyah untuk mengidentifikasi permasalahan pada sistem absensi sebelumnya. Berdasarkan hasil observasi ditemukan bahwa metode absensi manual dan QR Code masih memiliki berbagai kendala seperti antrean panjang, keterlambatan pencatatan data, serta sering terjadinya data kosong akibat kegagalan pengiriman ke server cloud. Setelah permasalahan teridentifikasi, tahap berikutnya adalah perancangan sistem dual-core yang bertujuan untuk membagi beban kerja pemrosesan data pada ESP32 agar sistem dapat berjalan lebih cepat dan stabil.



Gambar 1. Alur penelitian

Tahap selanjutnya adalah implementasi perangkat keras dan dashboard web. Pada tahap ini dilakukan proses integrasi sensor RFID RC522, modul MicroSD, LCD, buzzer, serta pengembangan antarmuka berbasis Google Apps Script dan Google Sheets sebagai platform serverless. Setelah perangkat berhasil dirakit, proses dilanjutkan dengan integrasi cloud dan batch processing. Pada tahap ini data absensi tidak langsung dikirim satu per satu ke server, melainkan dikumpulkan terlebih dahulu di dalam antrean MicroSD kemudian dikirim

secara berkelompok menggunakan metode batch processing untuk menghindari error rate limit dari server Google Apps Script. Proses ini menjadi inti utama penelitian karena mampu meningkatkan keandalan pengiriman data dan meminimalkan risiko kehilangan data.

Tahap berikutnya adalah pengujian performa dan stress test yang dilakukan dengan mensimulasikan kondisi penggunaan nyata di lingkungan sekolah. Pengujian dilakukan untuk mengetahui kemampuan sistem dalam menangani banyak proses pemindaian kartu RFID secara beruntun dalam waktu singkat. Hasil pengujian kemudian dianalisis pada tahap evaluasi kinerja sistem dengan menggunakan parameter Packet Delivery Ratio (PDR) dan latency lokal. Melalui tahapan tersebut, penelitian ini berhasil menghasilkan sistem absensi RFID berbasis serverless yang memiliki respons cepat, stabil, serta mampu menjaga integritas data secara optimal.

RESULTS AND DISCUSSION

Tahap ini merupakan realisasi dari perancangan untuk menjawab Tujuan Penelitian pertama. Perangkat keras dirakit menggunakan ESP32 Dual-Core, sensor RFID, dan MicroSD, lalu dipasang di area gerbang sekolah. Sementara itu, antarmuka *database* diimplementasikan menggunakan Google Sheets.



Gambar 2 Purwarupa Perangkat Absensi RFID di lingkungan sekolah



Gambar 3 Dokumentasi hasil data di database (Google Sheets)

Pada gambar 4.2 adalah bukti visual dari *back-end* atau *database* mentah (dalam hal ini, *worksheet* "Rekap Bulanan" di Google Sheets). Dokumentasi ini memvalidasi integritas data, membuktikan bahwa data yang dikirim oleh perangkat IoT (ESP32) melalui mekanisme *Batch Processing* dan *Queueing* telah berhasil diterima, diproses, dan disimpan secara permanen di *cloud* (Google Sheets). Ini mengkonfirmasi keberhasilan alur data end-to-end dari perangkat keras ke *database cloud* yang andal.

Untuk memvalidasi bahwa setiap modul perangkat keras dan instruksi perangkat lunak pada arsitektur di atas berjalan sesuai skenario, dilakukan pengujian fungsionalitas. Hasil pengujian fungsionalitas sistem ditunjukkan pada Tabel 1 berikut:

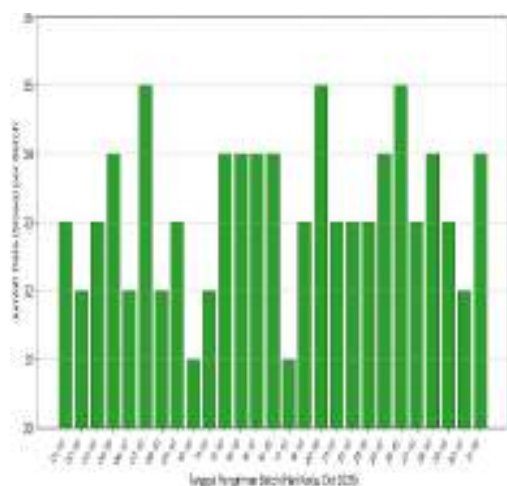
Tabel 1 Skenario Pengujian Fungsional

N o	Modul/Komponen	Skenario Pengujian	Hasil yang Diharapkan (ekspektasi)	Hasil
1	Sensor RFID	Menempelkan kartu RFID milik siswa ke area pemindai (reader).	UID kartu terbaca dalam hitungan kurang dari 2 detik dan teridentifikasi.	Valid
2	Modul MicroSD	Melakukan tap kartu saat perangkat ESP32	Data UID dan waktu lokal berhasil	Valid

		tidak terhubung ke jaringan Wi-Fi sekolah.	ditulis dan tersimpan dengan aman ke dalam file di MicroSD (Non-Volatile Queue).		Bagian ini memaparkan hasil dari tujuan penelitian kedua, yaitu mengevaluasi secara komparatif performa sistem usulan terhadap sistem eksisting sekolah (metode pencatatan manual dan sistem QR Code). Metrik utama yang dievaluasi adalah Latency Lokal (waktu yang dibutuhkan per siswa untuk melakukan absensi).															
3	Dual-Core Multitasking	Menempelkan kartu RFID secara beruntun (spamming) tepat saat alat sedang mengirim data ke Cloud.	Alat tidak freeze (membeku). Buzzer tetap berbunyi instan berkat pemisahan tugas Core 1 dan Core 0.	Valid	Berdasarkan pengujian pengiriman data absensi ke platform <i>serverless</i> (Google Apps Script dan Google Sheets), sistem berhasil membungkus data ke dalam format JSON <i>array</i> dan mengirimkannya secara berkala. Berdasarkan pengujian pengiriman data absensi ke platform <i>serverless</i> (Google Apps Script dan Google Sheets), sistem berhasil membungkus data ke dalam format JSON <i>array</i> (kelompok) dan mengirimkannya secara berkala. Hasil pengujian menunjukkan tingkat keberhasilan pengiriman atau Packet Delivery Ratio (PDR) mencapai 100%. Dari total 27 sesi pengiriman batch yang diuji selama pengujian lapangan, tidak ada satupun record data siswa yang hilang (data loss) atau gagal disimpan ke dalam database. Sistem antrean (queueing) pada MicroSD terbukti sangat efektif menahan data secara aman (non-volatile) ketika perangkat sedang luring, untuk kemudian dikirimkan seketika saat koneksi stabil.															
4	Koneksi Wi-Fi	Memutus daya router Wi-Fi lalu menyalakannya kembali.	Modul Wi-Fi pada ESP32 melakukan rekoneksi otomatis (auto-reconnect) tanpa perlu di-restart.	Valid	Tabel 2 Analisis Packet delivery ratio (PDR)															
<table><tr><th>Metrik</th><th>Nilai</th><th>Keterangan</th></tr><tr><td>Total Transaksi Dikirim</td><td>27</td><td>Total pengiriman <i>batch</i> selama sebulan.</td></tr><tr><td>Transaksi SUKSES</td><td>27</td><td>Jumlah <i>batch</i> yang diterima <i>server</i>.</td></tr><tr><td>Transaksi GAGAL</td><td>0</td><td>Jumlah <i>batch</i> yang gagal (misal: HTTP error).</td></tr><tr><td>PDR (Tingkat Keberhasilan)</td><td>100.00%</td><td>(27 / 27) * 100%</td></tr></table>						Metrik	Nilai	Keterangan	Total Transaksi Dikirim	27	Total pengiriman <i>batch</i> selama sebulan.	Transaksi SUKSES	27	Jumlah <i>batch</i> yang diterima <i>server</i> .	Transaksi GAGAL	0	Jumlah <i>batch</i> yang gagal (misal: HTTP error).	PDR (Tingkat Keberhasilan)	100.00%	(27 / 27) * 100%
Metrik	Nilai	Keterangan																		
Total Transaksi Dikirim	27	Total pengiriman <i>batch</i> selama sebulan.																		
Transaksi SUKSES	27	Jumlah <i>batch</i> yang diterima <i>server</i> .																		
Transaksi GAGAL	0	Jumlah <i>batch</i> yang gagal (misal: HTTP error).																		
PDR (Tingkat Keberhasilan)	100.00%	(27 / 27) * 100%																		
5	Batch Processing	Membiarkan 10 siswa melakukan tap, lalu mengecek transmisi data di serial monitor.	Data dikirim sekaligus dalam 1 paket JSON tunggal (1 kali panggilan API), bukan 10 kali panggilan terpisah.	Valid	Berdasarkan Tabel Analisis PDR pada Tabel 2 di atas, terlihat rincian secara konstan bahwa dari total 27 sesi pengujian batch harian, persentase keberhasilan pengiriman data selalu menyentuh angka 100%. Tidak terdapat satupun baris data yang gagal terkirim (0 data loss), yang membuktikan bahwa antrean fisik pada modul MicroSD berhasil secara mutlak menahan dan menjaga data absensi siswa agar tidak hilang ketika terjadi pemblokiran API oleh server.															

Tabel 3 Statistik deskriptif throughput (per Batch)

Metrik	Nilai	Keterangan
Rata-rata (Mean)	33.19 data/batch	Rata-rata jumlah data siswa yang dikirim per hari.
Min	31 data/batch	Jumlah data terendah dalam satu batch.
Max	35 data/batch	Jumlah data tertinggi.
Total Transaksi	27	Jumlah hari kerja/transaksi batch di bulan Oktober.

**Gambar 4 Analisis throughput per hari (Oktober 2025)**

Berdasarkan Gambar 4 di atas, grafik analisis throughput memvisualisasikan bagaimana mekanisme Batch Processing bekerja secara riil di lapangan. Sistem tidak lagi mengirimkan data absensi satu per satu setiap kali kartu dipindai, melainkan menumpuk puluhan data ke dalam satu paket (payload) JSON untuk dikirimkan sekaligus.

Grafik tersebut menunjukkan lonjakan pengiriman data (berkisar antara puluhan transaksi per batch) yang tereksekusi hanya dalam satu kali panggilan API ke Google Apps Script. Pendekatan ini secara drastis mengurangi beban lalu lintas jaringan dan menjadi kunci utama mengapa sistem ini berhasil terhindar dari pemblokiran rate-limit

(HTTP 429) yang sering dialami oleh platform serverless gratis.

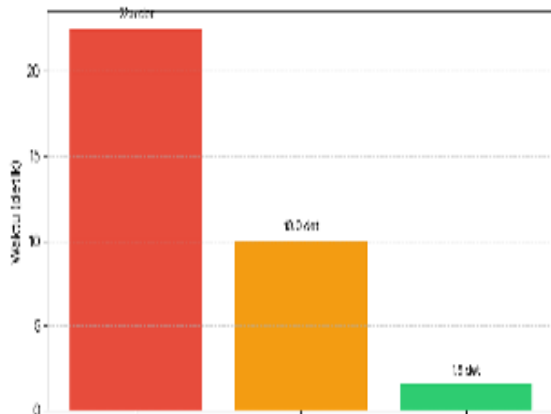
Berdasarkan hasil pengujian di lapangan, penerapan arsitektur Dual-Core terbukti membebaskan Core 1 dari beban koneksi internet, sehingga sistem dapat merespons pindaian kartu RFID secara instan. Hasil evaluasi menunjukkan bahwa waktu respons sistem usulan (Latency Lokal) hanya membutuhkan waktu 1 hingga 2 detik per siswa.

Untuk memvalidasi signifikansi dari arsitektur yang diusulkan, dilakukan perbandingan kinerja dengan sistem absensi metode lain (Manual dan QR Code) berdasarkan studi literatur dan observasi logis.

Tabel 5 Analisis perbandingan kinerja (sistem lama vs sistem baru)

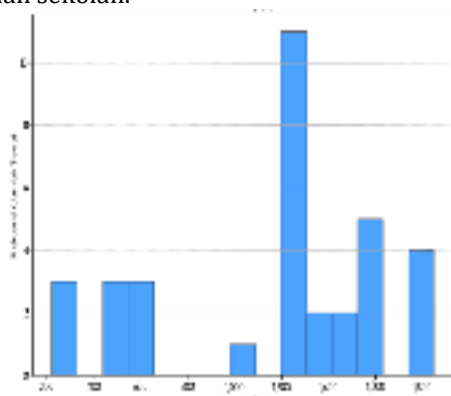
Metrik	Metode Manual (Buku)	Metode QR Code (Real-Time)	Sistem Peneliti (Batch-Queueing)
Latency Lokal	15 - 30 detik	5 - 15 detik	1 - 2 detik
(Waktu per 1 siswa tap)	(Mencari nama, ttd, antri)	(Buka HP/App, scan, tunggu)	(Tap & Go)
Latency Integrasi	2 Jam - 24 Jam	1 - 5 Detik	1166.7 Detik (Rata-rata)
(Waktu sampai data masuk rekap digital)	(Menuunggu Guru Piket/TU rekap manual ke excel)	(Jika Wifi lancar)	(Sengaja ditunda oleh sistem Batch)
Keandalan (PDR)	Rendah	Renah - Sedang	100% (Sangat Tinggi)
(Risiko data hilang saat proses)	(Human error, salah tulis, buku hilang)	(Sangat rentan Wifi. Jika 1 siswa gagal kirim, datanya hilang)	(Dibuktikan 100% PDR. Data aman di SD Card)
Throughput Lokal	Sangat Rendah	Renah	Sangat Tinggi

(Kemampuan menangani antrian)	(Sangat macet)	(Macet jika sinyal Wifi jelek)	(Anti- macet, siswa hanya tap 1-2 detik)
-------------------------------------	-------------------	---	--



**Gambar 5 Perbandingan Latency Lokal
(Waktu tap per siswa)**

Jika dikomparasikan, sistem usulan ini 7 kali lebih cepat dibandingkan sistem QR Code sebelumnya (membutuhkan 5-15 detik) dan 15 kali lebih cepat dibandingkan metode manual (15-30 detik). Secara praktis, peningkatan kecepatan yang sangat drastis ini berhasil menghilangkan masalah antrian panjang di gerbang sekolah. Efektivitas alat ini juga dibuktikan melalui statistik penggunaan nyata selama bulan Oktober 2025 di SMK Plus As-Salafiyah, di mana alat sukses mencatatkan akumulasi 896 transaksi absensi siswa Kelas X TBSM tanpa mengalami crash, Membuktikan bahwa sistem tidak hanya cepat di atas kertas, tetapi juga andal digunakan dalam rutinitas harian sekolah.



Gambar 6 Histogram Distribusi Latency (Uji coba okt 2025)

Untuk memvalidasi bahwa kecepatan respons alat bukanlah sebuah kebetulan, Gambar 6 menyajikan sebaran (distribusi)

latency lokal selama masa pengujian lapangan berlangsung. Grafik ini membuktikan bahwa waktu respons 1 hingga 2 detik tersebut merupakan angka yang konsisten dicapai pada hampir seluruh transaksi pindaian kartu RFID.

Tidak ditemukan adanya fluktuasi ekstrem atau lonjakan waktu tunda (*spike*) yang signifikan saat terjadi antrian panjang di gerbang sekolah. Konsistensi ini membuktikan bahwa pemisahan tugas menggunakan arsitektur *Dual-Core* bekerja dengan sempurna; di mana *Core 1* selalu dalam keadaan siaga penuh untuk merespons tap kartu tanpa terinterupsi oleh proses pencarian sinyal Wi-Fi atau komputasi pengiriman data yang sedang berjalan di *Core 0*.

Analisis ini didasarkan pada data uji coba tanggal 13 Oktober 2025, dimana semua 34 siswa yang hadir melakukan tap pada jam acak (sekitar 06:26 - 06:54), dan data dikirim serentak dalam satu *batch* pada pukul 06:58:00.

Tabel 4 Statistik deskriptif latency (Uji coba 13 Okt 2025)

Metrik	Nilai	Keterangan
Rata-rata (Mean)	1166.7 detik	(~19.4 menit) Rata-rata waktu tunggu data di SD Card.
Min	221.0 detik	(~3.7 menit) Waktu tunggu data tercepat (tap terakhir).
Max	1849.0 detik	(~30.8 menit) Waktu tunggu data terlama (tap pertama).
Std. Deviation	486.6 detik	Sebaran waktu tunggu data cukup bervariasi.
Jumlah Data	34	Total siswa yang hadir dan diukur.

CONCLUSION

Berdasarkan serangkaian tahapan perancangan, implementasi, dan pengujian kinerja yang telah dilakukan pada purwarupa sistem absensi RFID di SMK Plus As-Salafiyah, maka dapat ditarik kesimpulan yang secara langsung menjawab tujuan penelitian sebagai berikut:

Penelitian ini telah berhasil merancang dan mengimplementasikan arsitektur *Internet of Things* (IoT) pada perangkat *edge* menggunakan mikrokontroler ESP32 *Dual-Core*. Pembagian beban komputasi secara asinkron—di mana *Core 1*

didedikasikan murni untuk kecepatan pembacaan kartu dan penyimpanan lokal (MicroSD), sedangkan Core 0 menangani transmisi internet—terbukti efektif menjadi solusi atas kelemahan sistem absensi *real-time* tunggal. Strategi pengiriman data secara berkelompok (*Batch Processing*) berhasil memitigasi penolakan API (*Error HTTP 429*) dari platform *serverless* Google Apps Script.

Evaluasi sistem secara kuantitatif menunjukkan bahwa pemisahan proses lokal dan jaringan menghasilkan titik kompromi (*trade-off*) yang sangat ideal. Dari sisi kecepatan, sistem mencapai *latency* lokal 1 hingga 2 detik yang jauh lebih efisien dibandingkan pencatatan manual maupun pemindaian aplikasi *QR Code*. Dari sisi keandalan transmisi, pemanfaatan antrean fisik (*Queueing*) pada MicroSD berhasil mempertahankan *Packet Delivery Ratio* (PDR) di angka 100%, yang berarti tidak ada satu pun data presensi siswa yang hilang (*data loss*) meskipun infrastruktur beroperasi menggunakan *cloud* gratis (*Zero-Cost Serverless*).

REFERENCE

- [1] F. Costa, S. Genovesi, M. Borgese, and G. Manara, "A review of RFID sensors, the new frontier of Internet of Things," *Sensors*, vol. 21, no. 9, p. 3138, 2021, doi: 10.3390/s21093138.
- [2] Y. Duroc, "From identification to sensing: RFID is one of the key technologies in the IoT field," *Sensors*, vol. 22, no. 19, p. 7523, 2022, doi: 10.3390/s22197523.
- [3] V. D. Nguyen, H. V. Khoa, T. N. Kieu, and E.-N. Huh, "Internet of things-based intelligent attendance system: Framework, practice implementation, and application," *Electronics (Basel)*, vol. 11, no. 19, p. 3151, 2022, doi: 10.3390/electronics11193151.
- [4] P. Benedetti, M. Femminella, G. Reali, and K. Steenhaut, "Experimental analysis of the application of serverless computing to IoT platforms," *Sensors*, vol. 21, no. 3, p. 928, 2021, doi: 10.3390/s21030928.
- [5] V. Kjorveziroski, S. Filiposka, and V. Trajkovik, "IoT serverless computing at the edge: A systematic mapping review," *Computers*, vol. 10, no. 10, p. 130, 2021, doi: 10.3390/computers10100130.
- [6] J. Decker, P. Kasprzak, and J. M. Kunkel, "Performance evaluation of open-source serverless platforms for Kubernetes," *Algorithms*, vol. 15, no. 7, p. 234, 2022, doi: 10.3390/a15070234.
- [7] T. Shwe and M. Aritsugi, "Optimizing data processing: A comparative study of big data platforms in edge, fog, and cloud layers," *Applied Sciences*, vol. 14, no. 1, p. 452, 2024, doi: 10.3390/app14010452.
- [8] Z. Shojaei Rad and M. Ghobaei-Arani, "Data pipeline approaches in serverless computing: A taxonomy, review, and research trends," *J. Big Data*, vol. 11, p. 82, 2024, doi: 10.1186/s40537-024-00939-0.
- [9] D. Hercog, T. Lerher, M. Truntič, and O. Težak, "Design and implementation of ESP32-based IoT devices," *Sensors*, vol. 23, no. 15, p. 6739, 2023, doi: 10.3390/s23156739.
- [10] Z. Shojaei Rad and M. Ghobaei-Arani, "Data pipeline approaches in serverless computing: A taxonomy, review, and research trends," *J. Big Data*, vol. 11, no. 1, p. 82, 2024, doi: 10.1186/s40537-024-00939-0.
- [11] G. A. S. Cassel, V. F. Rodrigues, R. da Rosa Righi, M. R. Bez, A. C. Nepomuceno, and C. A. Costa, "Serverless computing for Internet of Things: A systematic literature review," *Future Generation Computer Systems*, vol. 128, pp. 299–316, 2022, doi: 10.1016/j.future.2021.10.020.
- [12] P. Benedetti, M. Femminella, G. Reali, and K. Steenhaut, "Experimental analysis of the application of serverless computing to IoT platforms," *Sensors*, vol. 21, no. 3, p. 928, 2021, doi: 10.3390/s21030928.