



Implementasi dan Analisis Algoritma FIFO, FEFO, dan LIFO pada Sistem Automated Storage and Retrieval System (ASRS) Berbasis Internet of Things untuk Produk Minuman Kemasan

Muhammad Daffa Fauzan*, Tatyantoro Andrasto, Mario Norman Syah

Fakultas Teknik, Program Studi Teknik Elektro, Universitas Negeri Semarang, Semarang, Indonesia

Email: ^{1,*}daffafauzan723@students.unnes.ac.id, ²TatyantoroAndrasto@mail.unnes.ac.id, ³marionormansyah@mail.unnes.ac.id

Email Penulis Korespondensi: daffafauzan723@students.unnes.ac.id

Abstrak—Pengelolaan gudang produk minuman kemasan menghadapi risiko serius berupa produk kedaluwarsa dan human error dalam penerapan rotasi stok. Penelitian ini mengimplementasikan sistem Automated Storage and Retrieval System (ASRS) berbasis Internet of Things yang mengintegrasikan tiga algoritma rotasi stok, yaitu FIFO (First In First Out), FEFO (First Expired First Out), dan LIFO (Last In First Out), dalam satu platform backend control. Sistem dikembangkan menggunakan PHP, MySQL, mikrokontroler ESP32, dan QR Code scanner. Metode penelitian yang digunakan adalah Research and Development (R&D) dengan model pengembangan Waterfall. Pengujian dilakukan di Laboratorium Teknik Elektro Universitas Negeri Semarang menggunakan produk susu UHT 100 ml sebanyak 60 kali percobaan (20 per algoritma). Backend control mengelola logika algoritma melalui query SQL deterministik: FIFO menggunakan *ORDER BY created_at ASC*, FEFO menggunakan *ORDER BY expired_date ASC* dengan *tiebreaker* FIFO, dan LIFO menggunakan *ORDER BY created_at DESC*. Hasil pengujian menunjukkan ketiga algoritma mencapai akurasi 100% dan identifikasi QR Code menghasilkan akurasi 93,3% dengan rata-rata response time 287 ms. Pengujian edge case membuktikan sistem mampu menangani kondisi *expired date* sama, waktu input sama, stok kosong, dan slot penuh secara deterministik. Algoritma FEFO terbukti paling sesuai untuk produk minuman kemasan karena selalu memprioritaskan produk dengan tanggal kedaluwarsa terdekat, meminimalkan risiko produk kedaluwarsa. Kontribusi utama penelitian ini adalah Mengintegrasikan tiga algoritma rotasi stok (FIFO, FEFO, LIFO) dalam satu platform backend control IoT secara simultan, mekanisme tiebreaker deterministik berbasis query SQL untuk mengeliminasi ambiguitas pada kondisi data identik, dan validasi empiris keunggulan FEFO dibandingkan FIFO dan LIFO untuk produk minuman kemasan melalui analisis komparatif sisa stok.

Kata Kunci: ASRS; Backend Control; FEFO; FIFO; Internet of Things; LIFO; QR Code

Abstract—Warehouse management for packaged beverage products faces critical risks of expired products and human error in stock rotation. This research implements an IoT-based Automated Storage and Retrieval System (ASRS) integrating three stock rotation algorithms, FIFO, FEFO, and LIFO, in a single backend control platform using PHP, MySQL, ESP32 microcontroller, and QR Code scanner. Research and Development (R&D) method with Waterfall model was employed. Testing was conducted at the Electrical Engineering Laboratory of Semarang State University using 100 ml UHT milk products across 60 trials (20 per algorithm). Results show all three algorithms achieved 100% accuracy, and QR Code identification yielded 93.3% accuracy with an average response time of 287 ms. Edge case testing proved deterministic handling of identical *expiry dates*, simultaneous *timestamps*, empty stock, and full slots. FEFO proved most suitable for packaged beverage management by consistently prioritizing products with the nearest *expiry date*, minimizing spoilage risk. The main contributions of this research simultaneous integration of three stock rotation algorithms (FIFO, FEFO, LIFO) within a single IoT-based backend control platform, a deterministic SQL-based tiebreaker mechanism to eliminate ambiguity in identical-data conditions, and empirical validation of FEFO's superiority over FIFO and LIFO for packaged beverage products through comparative residual stock analysis.

Keywords: ASRS; Backend Control; FEFO; FIFO; Internet of Things; LIFO; QR Code

1. PENDAHULUAN

Industri minuman kemasan merupakan salah satu sektor industri dengan tingkat perputaran stok yang tinggi dan sensitivitas kritis terhadap tanggal kadaluarsa produk (Samuel et al., 2023). Kerugian akibat produk kadaluarsa yang tidak terdeteksi tepat waktu, kesalahan urutan pengambilan barang, serta sistem penyimpanan yang tidak terorganisasi masih menjadi permasalahan nyata yang berdampak langsung pada kerugian finansial pelaku usaha (Lestari et al., 2023). melaporkan bahwa penerapan FIFO yang belum optimal terbukti menyebabkan penurunan kualitas bahan dan meningkatkan risiko kerusakan produk (spoilage). Selain itu, penempatan barang yang tidak terstruktur meningkatkan risiko kesalahan distribusi (Widowati & Ningtiyas, 2022). (Kurtubi & Noviandi, 2022) serta (Sadiah, Purnama, et al., 2024) membuktikan bahwa FIFO mampu mengurangi produk kedaluwarsa dan meningkatkan efisiensi persediaan.

Urgensi masalah ini semakin nyata karena metode rotasi stok seperti FIFO, LIFO, dan FEFO masih diterapkan secara manual oleh sebagian besar pelaku industri, sehingga rentan terhadap human error dan inkonsistensi eksekusi. Berbagai penelitian terdahulu mengusulkan solusi otomatisasi: (Alhilal et al., 2023) mengembangkan ASRS berbasis robot 3-axis; S et al. (2025) memanfaatkan ESP32 dengan fingerprint dan HMI (Olugbenga Ayomide et al., 2024). mengintegrasikan IoT dengan sensor berat, RFID, dan computer vision. Namun sebagian besar penelitian masih berfokus pada aspek mekanik, keamanan, atau pencatatan data secara terpisah tanpa mengintegrasikan logika rotasi stok secara penuh.

Keterbatasan kritis yang teridentifikasi adalah belum adanya sistem ASRS yang mengintegrasikan ketiga algoritma rotasi stok (FIFO, LIFO, FEFO) secara terpadu dalam satu platform backend control berbasis IoT. Sistem backend pada penelitian terdahulu umumnya hanya berfungsi sebagai pencatat data, belum berperan aktif dalam mengendalikan logika rotasi stok secara *real-time* (Mohd Fadzli & Nawawi, 2024). Penelitian oleh (Ganesan et al.,



2025) mengarah pada integrasi IoT dan AI namun menuntut biaya implementasi tinggi yang kurang sesuai untuk skala menengah.

Beberapa penelitian terdahulu telah mengusulkan solusi otomasi untuk manajemen gudang. Alhilal et al. (2023) mengembangkan robot ASRS 3-axis untuk peminjaman instrumen laboratorium, namun sistem tersebut tidak dilengkapi dengan logika rotasi stok (FIFO/FEFO/LIFO) sehingga tidak dapat mengelola urutan pengeluaran barang berdasarkan kaidah manajemen inventori. Mohd Fadzli & Nawawi (2024) melakukan kajian komprehensif terhadap sistem ASRS dan menyimpulkan bahwa mayoritas implementasi hanya memanfaatkan backend sebagai pencatat data (data logger), belum berperan sebagai pengambil keputusan rotasi stok secara real-time. Olugbenga Ayomide et al. (2024) mengintegrasikan IoT dengan sensor berat, RFID, dan computer vision, namun fokus penelitiannya terbatas pada pelacakan inventori tanpa implementasi algoritma pemilihan urutan pengambilan barang yang terstruktur. Ganesan et al. (2025) mengarahkan solusi pada integrasi IoT dan AI, namun pendekatan ini menuntut infrastruktur komputasi tinggi yang tidak ekonomis untuk skala industri menengah.

Berdasarkan celah penelitian tersebut, penelitian ini mengembangkan sistem ASRS yang mengintegrasikan FIFO, LIFO, dan FEFO secara otomatis dalam satu backend control berbasis IoT. Kebaruan penelitian meliputi identifikasi barang menggunakan QR Code yang di-generate sistem secara otomatis, proses input barang sepenuhnya otomatis melalui konveyor setelah validasi QR Code, proses output barang dengan empat mode (FIFO, LIFO, FEFO, dan manual) yang ditentukan backend serta integrasi penuh antara algoritma, backend control, QR Code scanner, dan ASRS dalam satu ekosistem IoT terpadu.

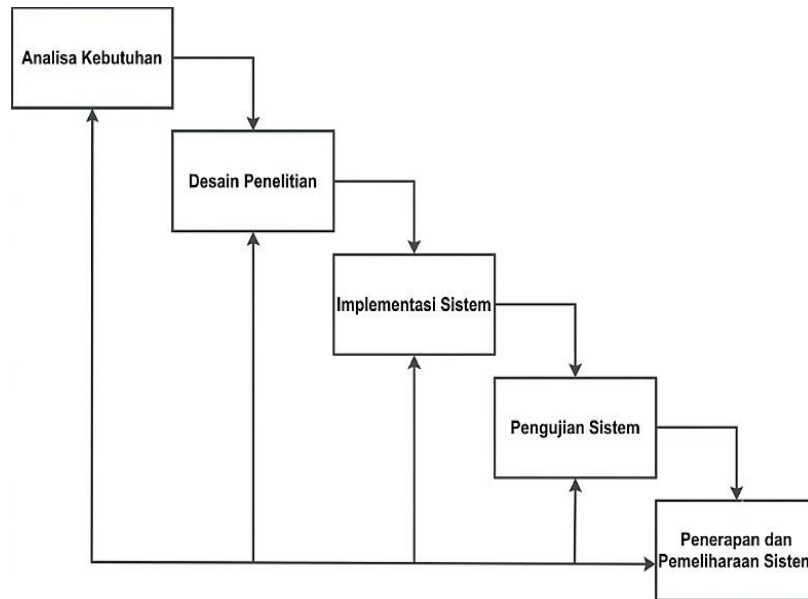
Penelitian ini dirancang untuk mengisi ketiga kesenjangan tersebut. Kebaruan penelitian meliputi identifikasi barang menggunakan QR Code yang di-generate sistem secara otomatis, proses input barang sepenuhnya otomatis melalui konveyor setelah validasi QR Code, proses output barang dengan empat mode (FIFO, LIFO, FEFO, dan manual) yang ditentukan backend, serta integrasi penuh antara algoritma, backend control, QR Code scanner, dan ASRS dalam satu ekosistem IoT terpadu.

2. METODOLOGI PENELITIAN

2.1 Pendekatan dan Prosedur Penelitian

Metode penelitian yang digunakan adalah *Research and Development* (R&D) dengan model pengembangan perangkat lunak *Waterfall*. Model ini dipilih karena kebutuhan *hardware* dan *software* sistem telah terdefinisi sejak awal, sehingga pendekatan linier lebih sesuai dibandingkan model iteratif. Penelitian dilaksanakan di Laboratorium Teknik Elektro, Fakultas Teknik, Universitas Negeri Semarang, pada periode Agustus 2025 hingga Juni 2026. Tahapan pengembangan sistem dilaksanakan sebagai berikut:

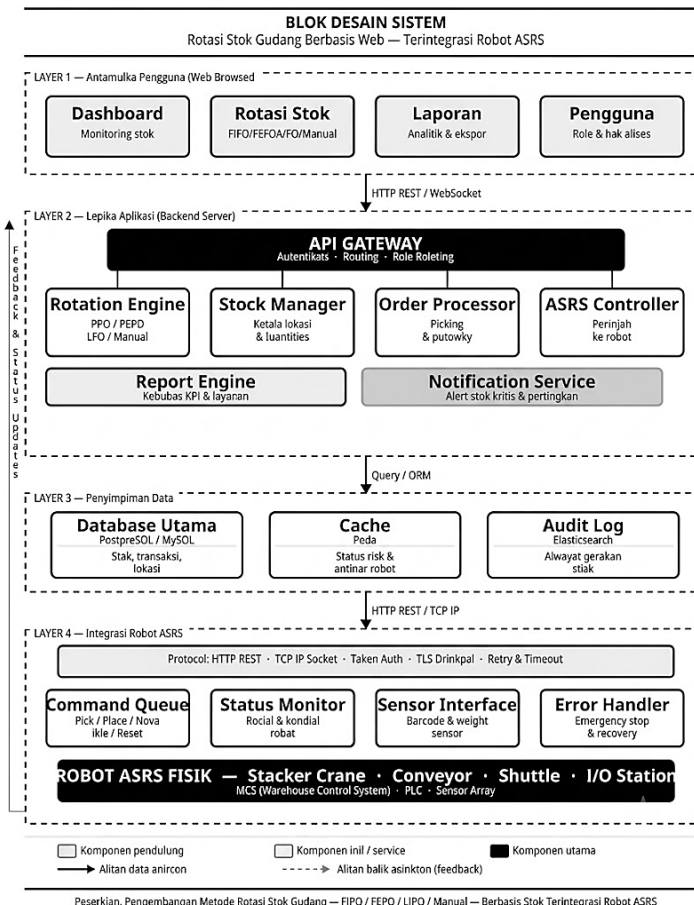
- Analisis Kebutuhan:** Pada tahap ini dilakukan identifikasi kebutuhan fungsional sistem, meliputi kebutuhan tiga mode rotasi stok (FIFO, FEFO, LIFO), identifikasi produk via QR Code, kendali robot ASRS melalui ESP32, serta antarmuka web berbasis PHP/MySQL. Survei pendahuluan dilakukan terhadap kondisi pengelolaan gudang manual untuk mengonfirmasi permasalahan *human error* dan produk kadaluarsa.
- Perancangan Sistem (Design):** Dirancang arsitektur sistem lengkap mencakup skema basis data MySQL (tabel *barang*, *slot*, *queue_robot*, *log_aktivitas*), diagram alur ketiga algoritma, protokol komunikasi HTTP antara *backend* dan ESP32, serta mekanisme *tiebreaker* deterministik. Pada tahap ini pula ditetapkan spesifikasi *hardware*: mikrokontroler ESP32, motor stepper Nema17, servo *gripper*, dan QR Code scanner.
- Implementasi (Coding):** Dikembangkan dua modul utama secara paralel. Modul *backend* dikembangkan menggunakan PHP dengan arsitektur MVC, mencakup *BarangController.php* dan *SlotController.php* beserta fungsi *query* dinamis *get_slot_by_mode.php* yang mengimplementasikan ketiga algoritma. Modul *firmware* ESP32 dikembangkan dalam bahasa C++ menggunakan Arduino IDE, mengimplementasikan mekanisme *polling* HTTP setiap 2 detik dan kendali motor stepper.
- Pengujian (Testing):** Dilakukan pengujian terstruktur dalam empat aspek: (a) pengujian akurasi algoritma FIFO, FEFO, dan LIFO masing-masing 20 kali menggunakan produk *Hydro Coco* dan susu UHT 200 ml; (b) pengujian identifikasi QR Code sebanyak 25 kali dengan variasi jarak 3–25 cm; (c) pengujian *edge case* pada kondisi *expired date* sama, waktu *input* sama, stok kosong, dan *slot* penuh; serta (d) analisis efisiensi komposisi sisa stok setelah 5 kali pengambilan.
- Pemeliharaan (Maintenance):** Sistem di-*deploy* di *server* Domainsia dan diuji dalam kondisi operasional nyata di laboratorium. Dokumentasi teknis disusun mencakup panduan penggunaan, spesifikasi API, dan daftar *endpoint* untuk memudahkan pengembangan lebih lanjut.



Gambar 1. Diagram Alir Tahapan Penelitian

2.2 Desain Sistem

Sistem ASRS yang dikembangkan merupakan platform manajemen gudang berbasis web yang mengintegrasikan tiga metode rotasi stok (FIFO, LIFO, dan FEFO) dengan robot ASRS fisik. Input sistem berupa data barang yang masuk ke gudang meliputi identitas barang, jumlah, waktu penerimaan (*timestamp*), dan tanggal kedaluwarsa (*expiry date*), yang dimasukkan melalui antarmuka web beserta pemilihan metode rotasi. Gambar 2 mengilustrasikan arsitektur desain sistem ASRS secara keseluruhan, mencakup lapisan *frontend*, *backend*, dan *hardware* ESP32 yang saling terhubung melalui protokol HTTP.

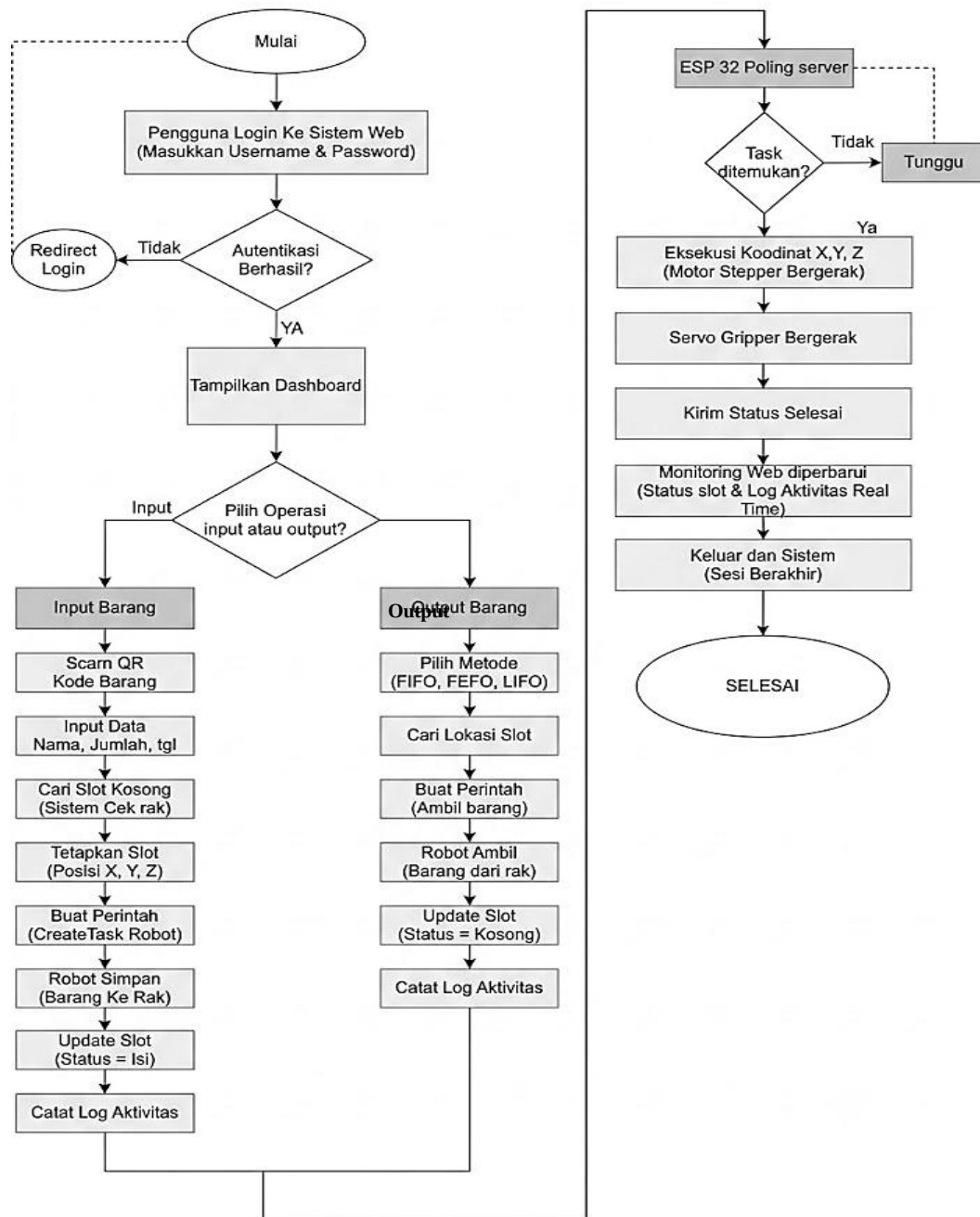


Gambar 2. Desain Sistem ASRS

Proses inti terjadi pada *Rotation Engine* di backend server, di mana algoritma rotasi yang dipilih mengolah data stok untuk menentukan urutan *picking*: FIFO memprioritaskan barang dengan *timestamp* penerimaan tertua, LIFO memprioritaskan barang dengan *timestamp* terbaru, dan FEFO memprioritaskan barang dengan *expiry date* paling dekat. Hasil keputusan diterjemahkan menjadi perintah gerakan yang dikirimkan ke robot ASRS melalui komunikasi HTTP secara otomatis.

2.2.1 Alur Kerja Sistem Keseluruhan

Sistem ASRS diawali ketika operator login ke aplikasi berbasis web. Setelah autentikasi berhasil, operator memilih jenis operasi: input barang untuk menyimpan ke rak, atau output barang untuk mengambil dari rak. Pada operasi input, sistem memandu operator melalui serangkaian tahapan mulai dari pemindaian QR Code, pengisian data barang, pencarian slot kosong secara otomatis, penetapan koordinat X, Y, Z, hingga pembuatan perintah tugas (*createTask*) yang diteruskan kepada robot. Gambar 3 menampilkan diagram alur cara kerja sistem secara keseluruhan, mulai dari proses *login* operator hingga eksekusi robot dan konfirmasi penyelesaian tugas.



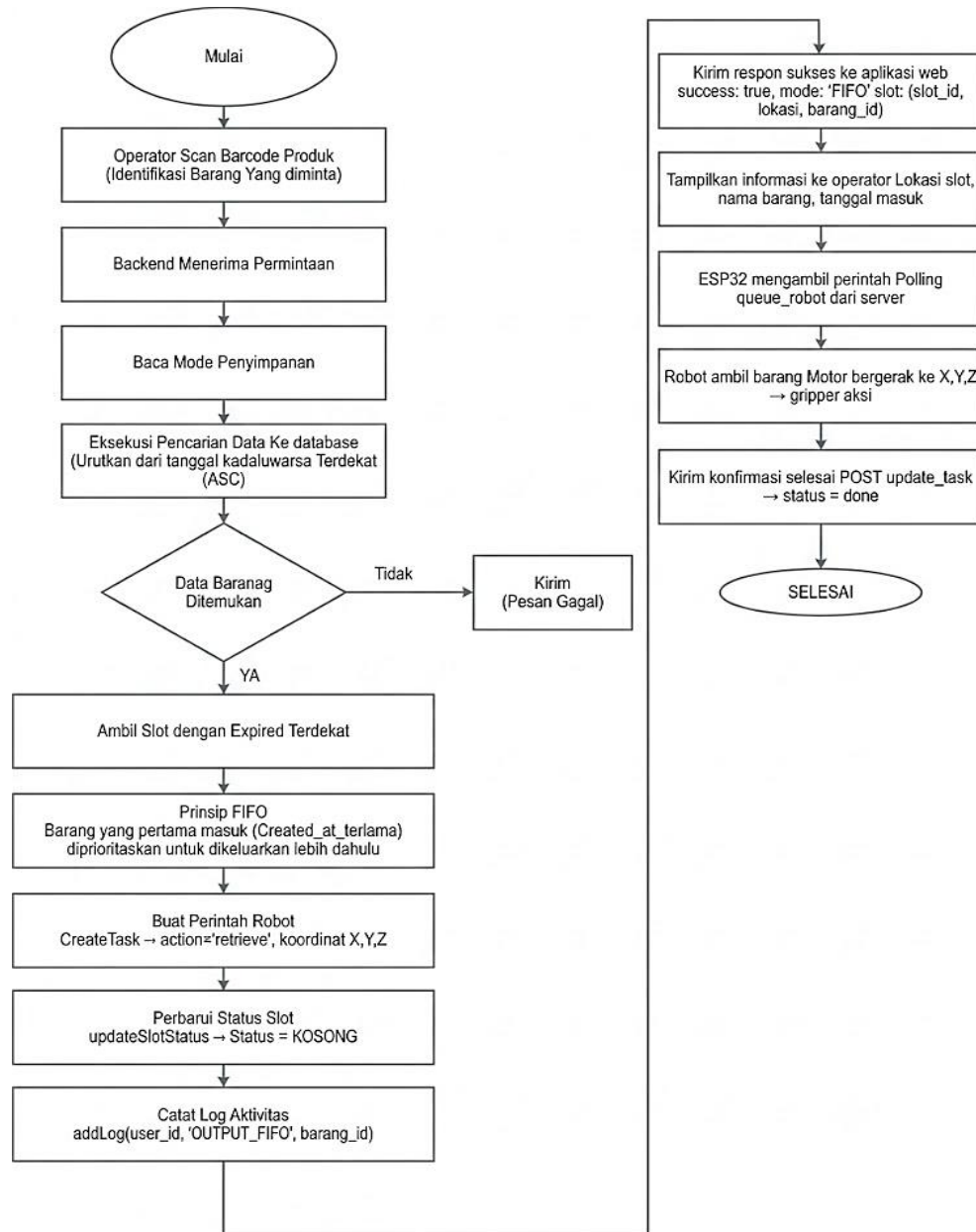
Gambar 3. Diagram Alur Cara Kerja Sistem Keseluruhan

Mikrokontroler ESP32 secara berkala melakukan polling ke server setiap dua detik melalui permintaan GET `/api/get_task.php`. Apabila tugas ditemukan, ESP32 memproses instruksi koordinat dengan mengaktifkan motor stepper untuk menggerakkan lengan robot menuju titik X, Y, Z yang telah ditentukan. Setelah robot mencapai posisi tujuan,

servo gripper diaktifkan untuk menjepit atau melepas barang. ESP32 kemudian mengirimkan konfirmasi penyelesaian ke server melalui POST `update_task` dengan status `done`.

2.2.2 Diagram Alur Algoritma FIFO

Gambar 4 menunjukkan diagram alur (*flowchart*) proses pengambilan barang menggunakan algoritma FIFO, dari awal perintah operator hingga pencatatan *log* transaksi. Diagram disajikan dalam format hitam-putih sesuai ketentuan metodologi.

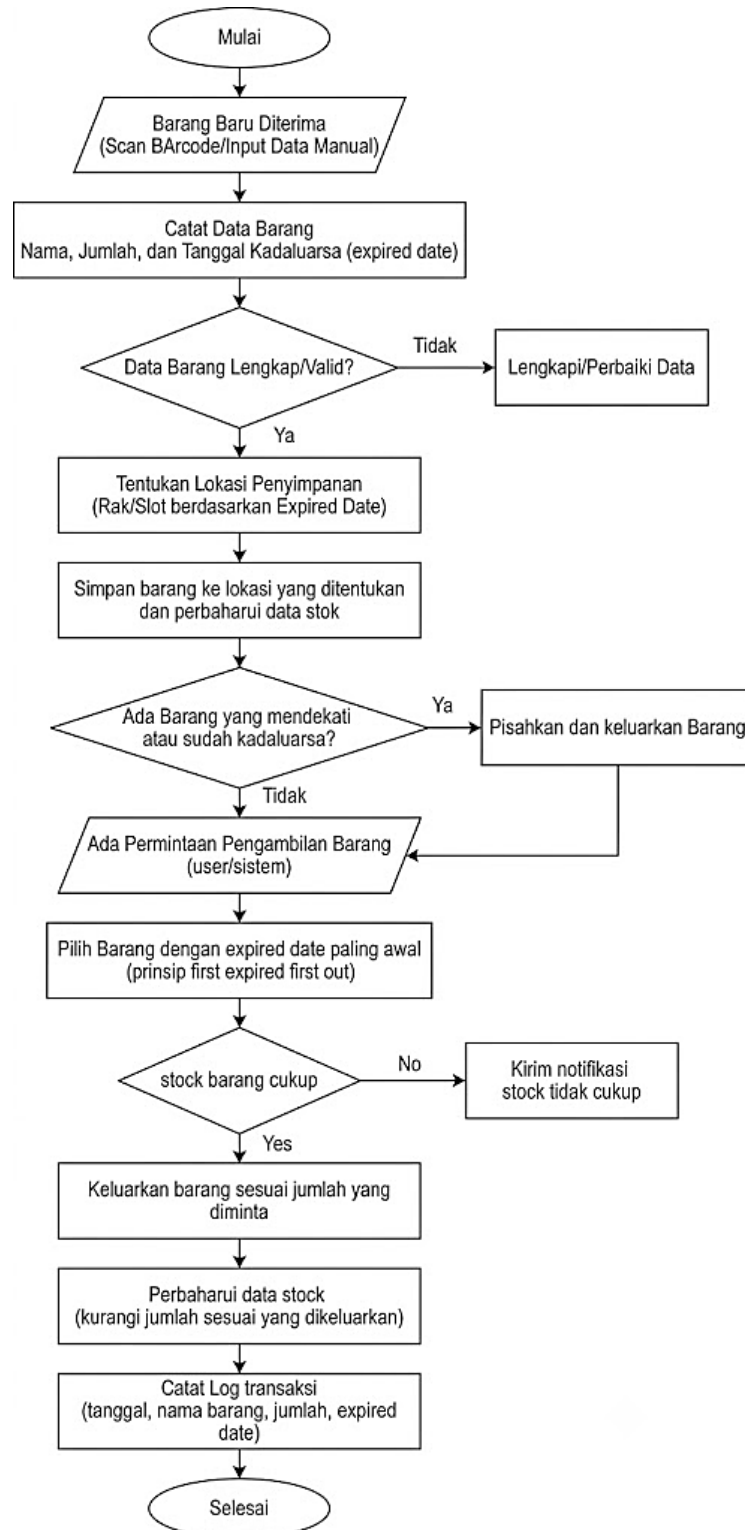


Gambar 4. Diagram Alur FIFO

Proses pengambilan barang dengan algoritma FIFO dimulai ketika operator memberikan perintah melalui antarmuka web. Sistem mengeksekusi pencarian data ke dalam database dengan mengurutkan seluruh barang berdasarkan waktu masuk terlama (*created_at* ASC), sehingga barang yang pertama kali disimpan menjadi prioritas utama untuk dikeluarkan. Setelah slot ditentukan, sistem membuat perintah tugas robot (*createTask*) dengan aksi *retrieve*, memperbarui status slot menjadi KOSONG, dan mencatat aktivitas ke log sistem dengan kode *OUTPUT_FIFO*.

2.2.3 Diagram Alur Algoritma FEFO

Gambar 5 menunjukkan diagram alur algoritma FEFO. Berbeda dengan FIFO, urutan pengambilan pada FEFO ditentukan berdasarkan nilai *expired_date* terkecil dengan *tiebreaker* FIFO apabila terdapat nilai *expired_date* yang sama.

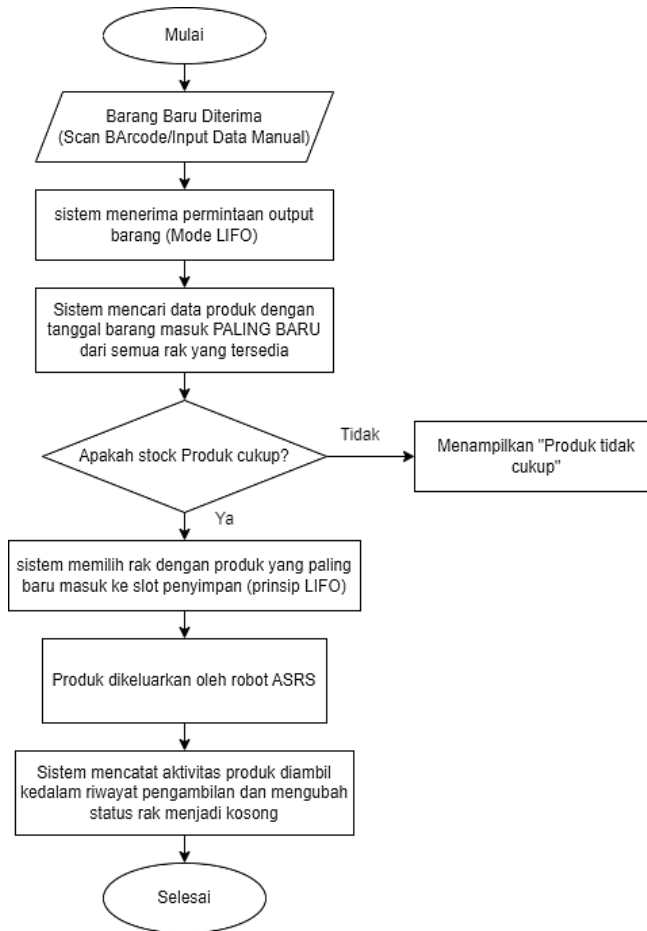


Gambar 5. Diagram Alur FEFO

Algoritma FEFO diterapkan dengan memilih barang yang memiliki *expired date* paling awal sebagai prioritas utama untuk dikeluarkan. Sistem memvalidasi kelengkapan data barang termasuk nama, jumlah, dan tanggal kedaluwarsa. Apabila terdapat produk dengan *expired date* yang sama, sistem menerapkan mekanisme *tiebreaker* menggunakan metode FIFO (urutan *created_at* ASC). Seluruh transaksi dicatat ke dalam log sistem yang memuat informasi tanggal transaksi, nama barang, jumlah, dan *expired date*.

2.2.4 Diagram Alur Algoritma LIFO

Gambar 6 menunjukkan diagram alur algoritma LIFO, di mana barang dengan *timestamp* masuk terbaru menjadi prioritas pertama untuk dikeluarkan. Sistem juga memeriksa kecukupan stok sebelum eksekusi.



Gambar 6. Diagram Alur LIFO

Algoritma LIFO melakukan pencarian data ke seluruh rak yang tersedia dengan mengurutkan berdasarkan tanggal masuk paling baru (*created_at DESC*), sehingga barang yang terakhir kali disimpan menjadi kandidat utama untuk dikeluarkan. Sebelum proses pengambilan dieksekusi, sistem memeriksa kecukupan stok. Apabila stok tidak mencukupi, sistem menampilkan notifikasi kepada operator. Setelah pengambilan selesai, sistem mencatat seluruh aktivitas transaksi dan memperbarui status slot secara otomatis.

2.3 Implementasi Algoritma FIFO, FEFO, dan LIFO

Ketiga algoritma diimplementasikan melalui fungsi query dinamis pada file *get_slot_by_mode.php*. Pada mode FIFO, query menggunakan *ORDER BY b.created_at ASC LIMIT 1* sehingga barang dengan *timestamp* penerimaan tertua diprioritaskan. Pada mode FEFO, query menggunakan *ORDER BY b.expired_date ASC, b.created_at ASC LIMIT 1*, memprioritaskan barang dengan tanggal kedaluwarsa terdekat dengan *tiebreaker* FIFO. Pada mode LIFO, query menggunakan *ORDER BY b.created_at DESC LIMIT 1*, memprioritaskan barang paling baru masuk. Mode MANUAL tidak menggunakan LIMIT sehingga seluruh slot ditampilkan untuk dipilih bebas.

Modul backend dikembangkan dalam dua kelas utama: *BarangController.php* mengelola *CRUD* data barang, pembuatan antrian tugas robot (*createTask()*), dan pencatatan log aktivitas (*addLog()*); serta *SlotController.php* mengelola status slot penyimpanan melalui *getAllSlots()*, *getEmptySlots()*, *getSlotStats()*, dan *updateSlotStatus()*. Seluruh operasi database menggunakan prepared statement PDO untuk keamanan query.

2.4 Metode Pengujian

Pengujian sistem mencakup empat aspek utama: (1) pengujian akurasi algoritma masing-masing 20 kali percobaan menggunakan produk Susu UHT 100 ml; (2) pengujian identifikasi QR Code 25 kali percobaan dengan variasi jarak baca; (3) pengujian edge case pada kondisi operasional khusus; dan (4) analisis efisiensi pengelolaan sisa stok. Akurasi dihitung menggunakan rumus (1)

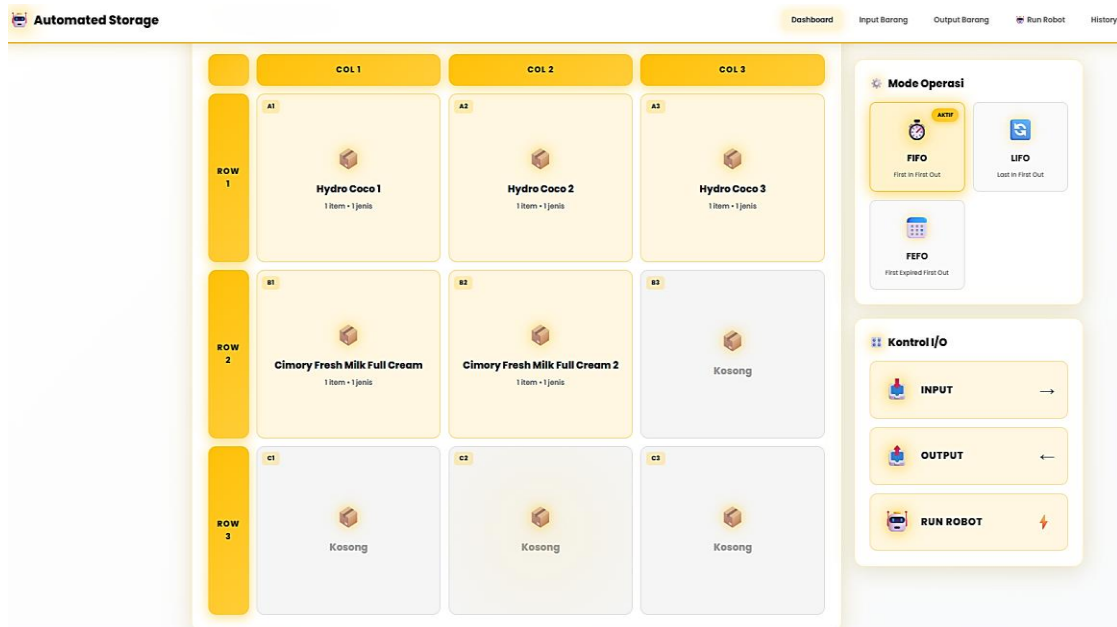
$$\text{Akurasi (\%)} = \frac{\text{Jumlah Output Benar}}{\text{Total Output}} \times 100\% \quad (1)$$

Variabel penelitian terdiri dari variabel bebas (mode algoritma: FIFO/FEFO/LIFO/MANUAL), variabel terikat Y1 (akurasi pemilihan slot, %), variabel terikat Y2 (response time API, ms), dan variabel kontrol (jumlah slot tersedia, spesifikasi hardware ESP32/Nema17/Servo).

3. HASIL DAN PEMBAHASAN

3.1 Deskripsi Implementasi Sistem

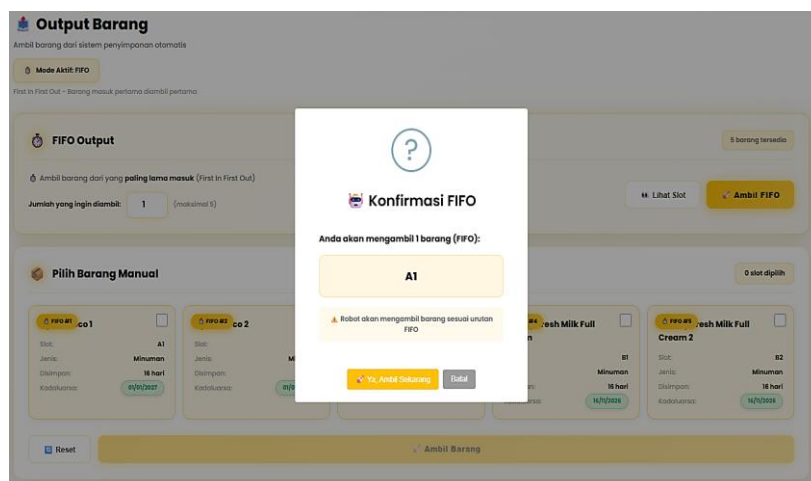
Sistem ASRS berbasis IoT dengan backend control yang mengintegrasikan tiga algoritma rotasi stok (FIFO, LIFO, FEFO) telah berhasil diimplementasikan dan diuji. Sistem terdiri dari aplikasi web yang di-deploy di server Domainsia sebagai backend sekaligus antarmuka monitoring, dan alat ASRS fisik yang dikendalikan oleh ESP32 melalui protokol HTTP. Seluruh pengujian dilakukan di Laboratorium Teknik Elektro UNNES menggunakan susu UHT 200 ml sebagai objek uji fisik. Antarmuka web menyediakan dashboard *real-time* dengan tampilan Storage Grid 3x3 yang menampilkan status 9 slot (A1–C3), panel mode operasi (FIFO/FEFO/LIFO), panel Kontrol I/O (INPUT/OUTPUT/RUN ROBOT), dan riwayat transaksi *real-time*. Seluruh pengujian menggunakan susu kemasan 200 ml dan hydro coco 200 ml sebagai objek uji fisik. Gambar 7 menampilkan tampilan *dashboard* utama sistem ASRS yang menunjukkan kondisi seluruh *slot* penyimpanan secara *real-time* beserta panel kendali algoritma.



Gambar 7. Tampilan Dashboard Sistem Automated Storage

3.2 Pengujian Akurasi Algoritma FIFO

Skenario pengujian FIFO dilakukan dengan mengisi seluruh 9 slot menggunakan produk dengan tanggal masuk berbeda: Hydro Coco 1-3 (*created_at*: 2026-05-16, slot A1–A3), Cimory Fresh Milk Full Cream 1-2 (*created_at*: 2026-05-17, slot B1–B2), Cimory Fresh Milk Cashew 1-2 (*created_at*: 2026-05-18, slot B3–C1), dan Cimory Fresh Milk Strawberry 1-2 (*created_at*: 2026-05-19, slot C2–C3). Berdasarkan prinsip FIFO, pengambilan seharusnya dimulai dari Hydro Coco yang masuk paling awal. Gambar 8 memperlihatkan tampilan konfirmasi pada antarmuka *web* saat sistem akan mengeksekusi pengambilan barang dengan metode FIFO, menampilkan detail *slot*, nama produk, dan *timestamp* yang dipilih berdasarkan *created_at* terlama.



Gambar 8. Tampilan Konfirmasi Pengambilan Barang Metode FIFO

Tabel 1 menyajikan kondisi awal seluruh 9 slot sebelum pengujian FIFO dilakukan, mencakup nama produk, tanggal masuk (*created_at*), tanggal kedaluwarsa (*expired_date*), dan posisi slot masing-masing produk.

Tabel 1. Kondisi Slot Awal Pengujian FIFO

No	Nama Produk	<i>created_at</i>	<i>Expired Date</i>	Slot
1	Hydro Coco 1	2026-05-16	2027-01-01	A1
2	Hydro Coco 2	2026-05-16	2027-01-01	A2
3	Hydro Coco 3	2026-05-16	2027-01-01	A3
4	Cimory Fresh Milk Full Cream	2026-05-17	2026-11-16	B1
5	Cimory Fresh Milk Full Cream 2	2026-05-17	2026-11-16	B2
6	Cimory Fresh Milk Cashew	2026-05-18	2026-11-15	B3
7	Cimory Fresh Milk Cashew 2	2026-05-18	2026-10-05	C1
8	Cimory Fresh Milk Strawberry	2026-05-19	2026-11-23	C2
9	Cimory Fresh Milk Strawberry 2	2026-05-19	2026-11-23	C3

Pengujian 20 kali percobaan menunjukkan sistem selalu memilih barang dengan *created_at* terlama sebagai prioritas. Pada kondisi *created_at* identik (contoh: ketiga Hydro Coco memiliki *created_at* sama), sistem menggunakan ID barang terkecil sebagai *tiebreaker* deterministik. Antarmuka web menampilkan pop-up konfirmasi FIFO yang memperlihatkan slot dan barang yang akan diambil sebelum eksekusi robot. Tabel 2 merangkum rekapitulasi hasil pengujian algoritma FIFO dari 20 kali percobaan, menunjukkan bahwa seluruh *output* barang sesuai dengan urutan FIFO yang diharapkan.

$$\text{Akurasi FIFO} = \left(\frac{20}{20}\right) \times 100\% = 100\%$$

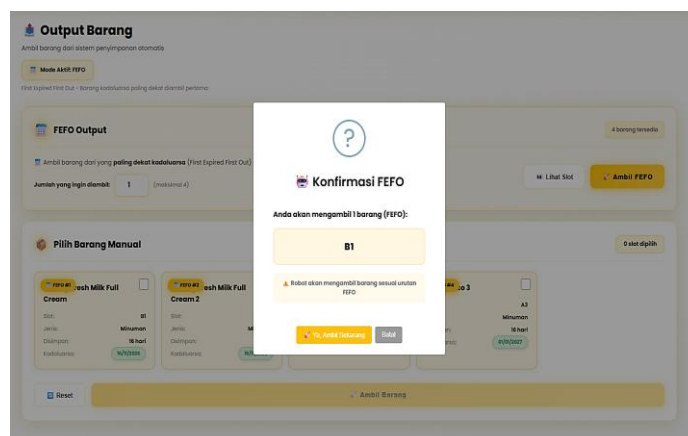
Tabel 2. Rekapitulasi Hasil Pengujian Algoritma FIFO

Parameter	Nilai Target	Nilai Hasil	Status
Total Percobaan	20	20	—
Output Benar (urutan FIFO)	20	20	Tercapai
Output Salah	0	0	Tidak ada error
Akurasi FIFO (%)	≥ 95%	100%	LULUS
Error Rate (%)	≤ 5%	0%	LULUS

Setelah 5 percobaan, sisa stok mencakup Cimory Cashew (2026-05-18) dan Cimory Strawberry (2026-05-19). Kondisi ini menunjukkan FIFO berhasil mengeluarkan barang terlama namun tidak mempertimbangkan *expired_date* dalam pengambilan keputusan.

3.3 Pengujian Akurasi Algoritma FEFO

Pengujian FEFO menggunakan kondisi slot yang sama dengan pengujian FIFO. Sistem memprioritaskan Cimory Fresh Milk Cashew 2 (slot C1, *expired_date*: 2026-10-05) sebagai barang pertama karena memiliki tanggal kedaluwarsa terdekat. Pada percobaan ke-3 dan ke-4 terjadi kondisi edge case di mana dua produk memiliki *expired_date* identik (2026-11-16), sehingga sistem menggunakan *created_at* ASC sebagai *tiebreaker* secara otomatis. Gambar 9 menampilkan tampilan konfirmasi pengambilan barang dengan metode FEFO, di mana sistem memilih produk dengan *expired_date* terdekat (2026-10-05) sebagai kandidat pertama.



Gambar 9. Tampilan Konfirmasi Pengambilan Barang Metode FEFO

Tabel 3 menggambarkan urutan pengambilan barang pada pengujian FEFO beserta analisis mekanisme *tiebreaker* yang diterapkan pada kondisi *expired_date* yang sama.

Tabel 3. Urutan Pengambilan FEFO dan Analisis *Tiebreaker*

No	Nama Produk	Expired Date	Algoritma	Keterangan
1	Cimory Cashew 2	2026-10-05	FEFO	Expired date terdekat, tidak perlu <i>tiebreaker</i>
2	Cimory Cashew	2026-11-15	FEFO	Expired date lebih dekat, tidak perlu <i>tiebreaker</i>
3	Cimory Full Cream	2026-11-16	FEFO	Expired date sama → <i>created_at</i> → ID terkecil (B1)
4	Cimory Full Cream 2	2026-11-16	FEFO	Expired date sama → <i>created_at</i> → ID lebih besar (B2)
5	Cimory Strawberry	2026-11-23	FEFO	Expired date lebih jauh, diambil kelima

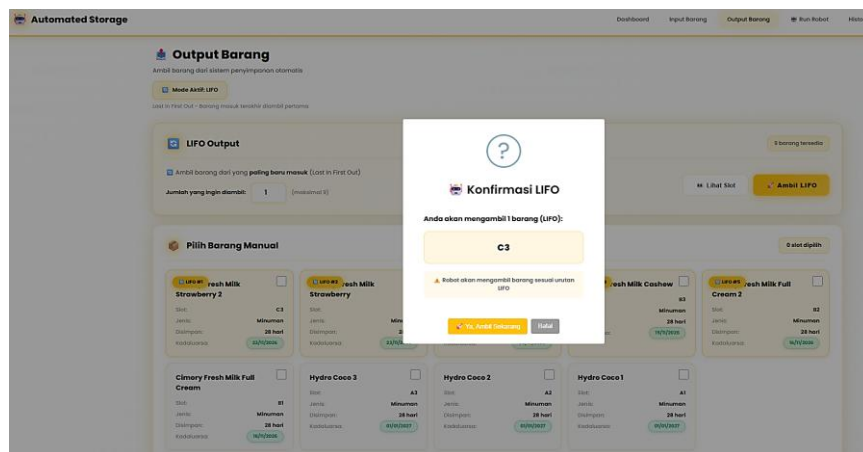
Setelah 5 percobaan FEFO, sisa stok mencakup Cimory Strawberry 2 (2026-11-23) dan ketiga Hydro Coco (2027-01-01) yang memiliki masa simpan paling panjang, membuktikan FEFO secara aktif mengeluarkan barang berisiko tinggi terlebih dahulu. Tabel 4 merekapitulasi hasil keseluruhan pengujian algoritma FEFO dari 20 kali percobaan.

Tabel 4. Rekapitulasi Hasil Pengujian Algoritma FEFO

Parameter	Nilai Target	Nilai Hasil	Status
Total Percobaan	20	20	—
Output Benar (urutan LIFO)	20	20	Tercapai
Output Salah	0	0	Tidak ada error
Akurasi LIFO (%)	≥ 95%	100%	LULUS
Error Rate (%)	≤ 5%	0%	LULUS

3.4 Pengujian Akurasi Algoritma LIFO

Pengujian LIFO menggunakan kondisi slot yang sama. Sistem memprioritaskan Cimory Strawberry 2 (slot C3, *created_at*: 2026-05-19, ID terbesar) sebagai barang pertama. Pada kondisi *created_at* identik, sistem menggunakan ID terbesar sebagai *tiebreaker*. Setelah 5 percobaan, Hydro Coco (masuk 2026-05-16) tetap tertahan, kondisi ini merepresentasikan risiko kritis penumpukan stok lama yang merupakan kelemahan inheren LIFO untuk produk pangan. Gambar 10 memperlihatkan tampilan konfirmasi pengambilan barang metode LIFO, di mana sistem memilih produk dengan *created_at* terbaru (2026-05-19) sebagai prioritas pertama.



Gambar 10. Tampilan Konfirmasi Pengambilan Barang Metode LIFO

Tabel 5 menyajikan rekapitulasi hasil pengujian algoritma LIFO dari 20 kali percobaan.

Tabel 5. Rekapitulasi Hasil Pengujian Algoritma LIFO

Parameter	Nilai Target	Nilai Hasil	Status
Total Percobaan	20	20	—
Output Benar (urutan LIFO)	20	20	Tercapai
Output Salah	0	0	Tidak ada error
Akurasi LIFO (%)	≥ 95%	100%	LULUS
Error Rate (%)	≤ 5%	0%	LULUS

3.5 Rekapitulasi Perbandingan Tiga Algoritma

Dari 60 total pengujian (20 per algoritma), seluruhnya menunjukkan hasil yang benar sesuai logika masing-masing metode. Ketiga algoritma mencapai akurasi 100% karena implementasinya bersifat deterministik berbasis query SQL, mengeliminasi human error secara menyeluruh. Tabel 6 merangkum perbandingan hasil akurasi ketiga algoritma (FIFO, FEFO, LIFO) dari total 60 kali pengujian.

**Tabel 6.** Rekapitulasi Perbandingan Tiga Algoritma

Algoritma	Total Uji	Benar	Error	Akurasi (%)
FIFO (First In First Out)	20	20	0	100%
FEFO (First Expired First Out)	20	20	0	100%
LIFO (Last In First Out)	20	20	0	100%
Rata-Rata Keseluruhan	60	60	0	100%

3.6 Pengujian Kondisi Khusus (Edge Case)

Pengujian edge case dilakukan untuk membuktikan ketangguhan sistem pada kondisi-kondisi operasional kompleks yang sering terjadi di gudang nyata. Kondisi *Expired Date* Sama: Pada FEFO, *tiebreaker* menggunakan *created_at* ASC. Pada FIFO, *tiebreaker* menggunakan ID terkecil. Pada LIFO, *tiebreaker* menggunakan ID terbesar. Seluruh mekanisme berjalan otomatis tanpa intervensi operator, membuktikan ketangguhan sistem dalam kondisi data ambigu.

Kondisi Waktu Input Sama: Ketika *created_at* identik, FIFO menggunakan ID terkecil; FEFO membandingkan *expired_date* terlebih dahulu, jika sama menggunakan ID terkecil; LIFO menggunakan ID terbesar. Seluruh mekanisme bersifat deterministik. Tabel 7 menyajikan seluruh skenario *edge case* yang diuji beserta mekanisme *tiebreaker* yang digunakan dan hasil pengujiannya.

Tabel 7. Hasil Pengujian Kondisi Edge Case

Kondisi	Algoritma	<i>Tiebreaker</i>	Hasil
<i>Expired date</i> sama	FEFO	<i>created_at</i> ASC	BERHASIL
<i>Expired date</i> sama	FIFO	ID terkecil	BERHASIL
<i>Expired date</i> sama	LIFO	ID terbesar	BERHASIL
Waktu input sama	FIFO	ID terkecil	BERHASIL
Waktu input sama	FEFO	<i>expired_date</i> lalu ID	BERHASIL
Waktu input sama	LIFO	ID terbesar	BERHASIL
Stok kosong	FIFO/FEFO/LIFO	Notifikasi otomatis	BERHASIL
Slot penuh	FIFO/FEFO/LIFO	Blokir input otomatis	BERHASIL

Pada kondisi stok kosong, sistem menghentikan proses output dan menampilkan notifikasi tanpa terjadi pengambilan fiktif. Sistem FEFO secara proaktif mendeteksi kondisi di mana semua stok telah melewati *expired date*, mencegah distribusi produk kadaluarsa secara otomatis. Pada kondisi slot penuh, sistem FEFO secara tidak langsung mendorong operator untuk melakukan output barang paling kritis terlebih dahulu, menciptakan siklus pengelolaan stok yang optimal

3.7 Analisis Efisiensi Pengelolaan Stok

Tabel 8 membandingkan kondisi sisa stok dan tingkat risiko kadaluarsa yang dihasilkan oleh masing-masing algoritma setelah 5 kali percobaan pengambilan barang.

Tabel 8. Perbandingan Kondisi Sisa Stok dan Risiko Kadaluarsa

Algoritma	Sisa <i>Expired Date</i>	Risiko Kadaluarsa	Kesesuaian untuk Minuman
FIFO	2026-10-05; 2026-11-15; 2026-11-23 (×2)	TINGGI	Tidak optimal — barang dengan <i>expired</i> dekat dapat tertahan
FEFO	2026-11-23 (×1); 2027-01-01 (×3)	RENDAH	PALING OPTIMAL — semua barang berisiko tinggi telah dikeluarkan
LIFO	2026-11-16 (×1); 2027-01-01 (×3)	TINGGI	Tidak optimal — barang masuk pertama terus tertahan

Algoritma FIFO meninggalkan Cimory Fresh Milk Cashew 2 (*expired*: 2026-10-05) di slot meskipun memiliki masa simpan paling pendek, karena bukan yang paling lama masuk. Algoritma LIFO menyebabkan Hydro Coco (masuk 2026-05-16) tertahan selama barang baru terus datang, berpotensi menyebabkan dead stock. Algoritma FEFO terbukti mengeluarkan semua barang dengan *expired date* terdekat lebih dahulu sehingga sisa stok memiliki masa simpan paling panjang (2026-11-23 hingga 2027-01-01) dan risiko kadaluarsa paling rendah. Temuan ini mengonfirmasi penelitian Mendes et al. (2020) yang menyatakan FEFO lebih sesuai untuk produk dengan variasi umur simpan kompleks, dan sejalan dengan prinsip keamanan pangan (food safety) yang mengharuskan produk dengan masa simpan lebih pendek untuk didistribusikan lebih dahulu.

3.8 Pengujian Identifikasi QR Barcode

QR Code di-generate otomatis oleh backend saat input data awal dengan encoding mencakup ID barang, nama produk, batch, *expired_date*, dan checksum validasi. Pengujian mencakup 25 percobaan dengan jarak baca 3 cm hingga 25 cm



menggunakan QR scanner yang terpasang di konveyor input. Tabel 9 menyajikan rekapitulasi lengkap hasil pengujian identifikasi QR Code mencakup akurasi pembacaan, *response time*, dan jarak efektif pembacaan scanner.

$$\text{Akurasi QR} = \left(\frac{23}{25}\right) \times 100\% = 93,3\%$$

Tabel 9. Rekapitulasi Hasil Pengujian QR Barcode

Parameter	Target	Hasil	Keterangan
Total Percobaan	25	25	—
QR Terbaca Benar	≥ 20	23	Percobaan ke-24 & 25 gagal (jarak 3 cm)
Kegagalan Baca	≤ 3	2	Di luar jangkauan efektif minimum
Akurasi Pembacaan QR	$\geq 90\%$	93,3%	LULUS
Response Time Rata-rata	< 500 ms	287 ms	Sangat Baik
Response Time Min	—	120 ms	Jarak 10 cm, cahaya optimal
Jarak Efektif QR Scanner	5–25 cm	5–25 cm	Optimal, di bawah 5 cm gagal

Dua kegagalan pada percobaan ke-24 dan ke-25 (jarak 3 cm) disebabkan keterbatasan jarak efektif minimum QR scanner. Response time rata-rata 287 ms jauh di bawah batas toleransi 500 ms. Akurasi 93,3% dinilai memadai untuk operasional konveyor dengan jarak terkontrol (5–25 cm) dan lebih ekonomis dibandingkan teknologi RFID yang digunakan (Olugbenga Ayomide et al., 2024).

3.9 Perbandingan dengan Penelitian Terdahulu

Hasil penelitian ini dikomparasikan dengan empat penelitian terdahulu yang relevan untuk mengukur posisi dan kontribusi penelitian secara kontekstual. Pertama, dibandingkan dengan Alhilal et al. (2023) yang mengembangkan robot ASRS 3-axis untuk peminjaman instrumen laboratorium, penelitian ini memiliki perbedaan mendasar pada lapisan *software*. Sistem Alhilal et al. (2023) tidak dilengkapi algoritma rotasi stok sehingga urutan pengeluaran barang bergantung sepenuhnya pada operator. Sebaliknya, sistem yang dikembangkan dalam penelitian ini mengeksekusi urutan pengambilan barang secara deterministik melalui *query* SQL tanpa keterlibatan operator, menghasilkan akurasi 100% pada seluruh 60 pengujian. Hal ini membuktikan bahwa penambahan lapisan *backend control* berbasis algoritma pada arsitektur ASRS secara signifikan meningkatkan keandalan manajemen inventori.

Kedua, dibandingkan dengan Mohd Fadzli & Nawawi (2024) yang menyimpulkan bahwa sistem ASRS pada umumnya hanya memanfaatkan *backend* sebagai *data logger*, penelitian ini secara langsung mengisi celah tersebut dengan menjadikan *backend control* sebagai mesin pengambil keputusan aktif. Setiap perintah pengambilan barang diproses oleh *backend* melalui fungsi *get_slot_by_mode.php* yang menentukan *slot* tujuan secara otomatis, memperbarui status inventori, dan meneruskan perintah koordinat ke robot ESP32 dalam satu siklus terpadu. Dengan demikian, temuan penelitian ini mengonfirmasi sekaligus melampaui rekomendasi Mohd Fadzli & Nawawi (2024) untuk mengembangkan ASRS dengan *backend* yang lebih aktif.

Ketiga, dibandingkan dengan Olugbenga Ayomide et al. (2024) yang menggunakan RFID sebagai teknologi identifikasi barang, penelitian ini memilih QR Code yang menghasilkan akurasi 93,3% dengan *response time* rata-rata 287 ms. Meskipun RFID umumnya memiliki akurasi lebih tinggi karena tidak memerlukan *line of sight*, teknologi QR Code dalam penelitian ini terbukti memadai untuk skenario konveyor dengan jarak terkontrol (5–25 cm) dengan biaya implementasi yang jauh lebih rendah. Selain itu, berbeda dengan Olugbenga Ayomide et al. (2024) yang fokus pada pelacakan inventori, QR Code dalam penelitian ini juga berfungsi sebagai *trigger* otomatisasi alur input barang secara menyeluruh, mulai dari validasi hingga penempatan di rak.

Keempat, dibandingkan dengan Mandalangi et al. (2025) yang mengimplementasikan algoritma FIFO tunggal pada sistem inventori berbasis *web* dan melaporkan peningkatan akurasi pengelolaan barang, penelitian ini memperluas cakupan dengan mengimplementasikan tiga algoritma secara simultan dalam satu platform. Lebih lanjut, penelitian ini menambahkan dimensi komparatif yang tidak ada pada Mandalangi et al. (2025): analisis kondisi sisa stok membuktikan bahwa pemilihan algoritma berdampak signifikan terhadap risiko kadaluarsa. Algoritma FEFO terbukti menghasilkan sisa stok dengan *expired date* paling jauh (2026-11-23 hingga 2027-01-01), dibandingkan FIFO yang menyalakan produk dengan *expired date* 2026-10-05 (78 hari lebih dekat) dan LIFO yang menyalakan produk dengan *expired date* 2026-11-16 (37 hari lebih dekat). Temuan ini memberikan dasar empiris yang lebih kuat untuk rekomendasi pemilihan algoritma pada industri minuman kemasan, melampaui sekadar pengukuran akurasi eksekusi algoritma.

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem ASRS berbasis IoT yang mengintegrasikan algoritma FIFO, FEFO, dan LIFO dalam satu platform *backend control*. Seluruh 60 pengujian (20 per algoritma) mencapai akurasi 100% melalui implementasi *query* SQL deterministik, mengeliminasi human error secara menyeluruh dan melampaui target minimal 95%. Sistem *backend control* PHP/MySQL berhasil dikembangkan sebagai pusat kendali logika rotasi stok sekaligus pengelola data slot penyimpanan secara terintegrasi, mencakup modul *BarangController*, *SlotController*,



sistem autentikasi, dan log transaksi *real-time*. Integrasi backend control dengan ASRS fisik melalui ESP32 berjalan andal, di mana identifikasi QR Code mencapai akurasi 93,3% dengan response time rata-rata 287 ms yang jauh di bawah batas toleransi 500 ms. Pengujian edge case pada kondisi *expired date* sama, waktu input sama, stok kosong, dan slot penuh membuktikan ketangguhan dan konsistensi sistem dalam skenario operasional kompleks. Analisis efisiensi sisa stok secara empiris membuktikan bahwa algoritma FEFO adalah pilihan paling tepat untuk produk minuman kemasan karena menghasilkan komposisi sisa stok dengan risiko kadaluarsa paling rendah (*expired date*: 2026-11-23 hingga 2027-01-01), secara aktif mencegah penumpukan barang kadaluarsa dibandingkan FIFO maupun LIFO. Untuk penelitian selanjutnya, disarankan memperluas pengujian dengan variasi produk lebih beragam, mengevaluasi teknologi identifikasi alternatif seperti RFID untuk meningkatkan jangkauan pembacaan, serta mengembangkan kapasitas slot yang lebih besar guna mengevaluasi skalabilitas sistem pada beban kerja tinggi. Penelitian ini memberikan kontribusi dalam tiga aspek. Pertama, dari sisi teknis, penelitian ini membuktikan bahwa perangkat berbasis PHP/MySQL/ESP32 yang terjangkau dapat menjalankan tiga algoritma rotasi stok sekaligus dengan tingkat akurasi sempurna, melebihi kemampuan sistem ASRS sebelumnya yang hanya berfungsi sebagai pencatat data. Kedua, dari sisi algoritma rotasi stok, penelitian ini memperkenalkan mekanisme penentuan prioritas berlapis (*expired_date* → *created_at* → ID) sebagai solusi atas kondisi di mana dua barang atau lebih memiliki data yang identik atau waktu input sama sebuah permasalahan yang belum dibahas dalam penelitian ASRS sebelumnya. Ketiga, dari sisi manajemen, hasil pengujian secara empiris menunjukkan bahwa algoritma FEFO lebih unggul dibandingkan FIFO dan LIFO, sehingga dapat menjadi rekomendasi praktis bagi pelaku industri minuman kemasan dalam memilih strategi rotasi stok yang tepat untuk mengurangi kerugian akibat produk kadaluarsa.

REFERENCES

- Alhilal, M., Hajah, M. S., & Kurdianto, A. A. (2023). Rancang Bangun Robot ASR (Automated Storage And Retrieval) 3 Axis Untuk Peminjaman Instrumen Di Laboratorium Electric Drive. *Techno Bahari*, 9(1), 27–33. <https://doi.org/10.52234/tb.v9i1.155>
- Bedoya-Vargas, F., & Del-Carpio-Lagones, D. A. (2024). *Developing a Lean Warehousing Storage Model: A Case Study in Peruvian Retail SMEs*. <https://doi.org/https://doi.org/10.46254/WC01.20240028>
- Bukhamseen, A., Alabdullah, M., Bin Gaufan, K., & Mysorewala, M. (2023). A Warehouse Storage and Retrieval System Using IoT and Autonomous Vehicle. *2023 9th International Conference on Automation, Robotics and Applications, ICARA 2023*, 346–350. <https://doi.org/10.1109/ICARA56516.2023.10125658>
- Deepika, M., & Vinoline, I. A. (2025). An Integrated Warehouse System using AS/RS (Automated Storage and Retrieval System) Technology. *Indian Journal Of Science And Technology*, 18(13), 1038–1045. <https://doi.org/10.17485/ijst.v18i13.178>
- Devega, M., Yuhelmi, Y., & Darmayunata, Y. (2024). Pembangunan Sistem Inventori Apotek Menggunakan Metode Fifo Dan Fefo. *ZONasi: Jurnal Sistem Informasi*, 6(1), 159–172. <https://doi.org/10.31849/zn.v6i1.17318>
- Fadzli, M. A. F. B. M., & Nawawi, and S. W. Bin. (2024). Review on Automated Storage and Retrieval System for Warehouse. *Journal of Informatics and Web Engineering*, 3(3), 77–97. <https://doi.org/10.33093/jiwe.2024.3.3.5>
- Ganesan, R. T., Mdhanush Raja, M., Sjeisurya, M., & Ngkamala Kannan, M. (2025). An Advanced Inventory Management System Powered by IoT and AI for Real-Time Tracking and Optimization. *International Journal for Research Trends and Innovation (Www.Ijrti.Org)*, 10(4), 173.
- Haikal, A. A., Indra, Z., Arfansyah, A., Kamil, A., Abror, A., Bastio, A., Tuan, S., & Serdang, D. (2025). *Jurnal Publikasi Ilmu Komputer dan Multimedia Implementasi Algoritma First In First Out (FIFO) Berbasis Web untuk Perhitungan dan Simulasi Berbagai Kasus Antrean*. <https://doi.org/10.55606/jupikom.v5i1.5615>
- Khan, Z., & Ahmed, M. B. (2025). *Advancements in Automated Storage and Retrieval Systems : A Comprehensive Review*. 6(2). <https://doi.org/10.19080/RAEJ.2025.06.555684>
- Kurtubi, A., & Noviandi, R. (2022). Penerapan metode first in first out (FIFO) dalam pengendalian persediaan produk makanan dan minuman. *Jurnal Manajemen Industri*, 14(2), 97–105. <https://doi.org/10.55122/blogchain.v2i2.536>
- Lestari, S. N., Gusnadi, D., Raharjo, T. P., Terapan, F. I., Barat, J., Makanan, P. B., & Kitchen, C. (2023). Analisis Penerapan Metode Fifo (First In First Out) Pada Penyimpanan Bahan Makanan Di Cold Kitchen The Papandayan Hotel. 8(2), 119–124. https://doi.org/https://doi.org/10.31970/pangan.v8i2.122?utm_source=chatgpt.com
- Mandalangi, R. J., Sara, K., & Mando, L. B. F. (2025). Web-Based Inventory Information System Using First-In First-Out (FIFO) Algorithm at CV Dewangga. *Jurnal Ilmiah Sistem Informasi*, 4(3), 523–543. <https://doi.org/10.51903/qmw5sk61>
- Naufal, I. A., Choiron, A., & Santoso, B. (2025). Aplikasi Inventory dengan Metode FIFO dan LIFO Berbasis Web pada PT. Affas Inti Selaras. *Jutisi : Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 14(1), 655. <https://doi.org/10.35889/jutisi.v14i1.2711>
- Olugbenga Ayomide, M., Abosede, D. O., Gabriel, A. K., Akintoba Emmanuel, A., & Timothy, A. O. (2024). A Framework for an IoT-enabled Intelligent Inventory Management System. *International Journal of Advances in Engineering and Management*, 6(10), 234–243. <https://doi.org/10.35629/5252-0610234243>
- Pratama, E. W., & Kiswantono, A. (2023). Electrical Analysis Using ESP-32 Module In Realtime. *JEECS (Journal of Electrical Engineering and Computer Sciences)*, 7(2), 1273–1284. <https://doi.org/10.54732/jeeecs.v7i2.21>
- Roziqin, Z., & Kusuma, A. P. (2021). Information System of Goods in and Out Using Fifo Method. *JOSAR (Journal of*



- Students Academic Research*), 6(1), 120–127. <https://doi.org/10.35457/josar.v6i1.1461>
- Sadih, H., Purnama, D., & Ishlah, M. (2024). Implementation of the First In First Out (FIFO) algorithm in the sandal and shoe product inventory (stock) application. *International Journal of Quantitative Research and Modeling*, 5(1), 31–39. <https://doi.org/10.46336/ijqrm.v5i1.552>
- Samuel, A. I., Jan, A. B. H., Palandeng, I. D., Penerapan, A., Pergudangan, M., Gudang, P., Trakindo, P. T., Samuel, A. I., Bin, A., Jan, H., Palandeng, I. D., Samuel, A. I., Jan, A. B. H., & Palandeng, I. D. (2023). Utama Manado Analysis of The Application of Warehousing Management at PT Trakindo. *Jurnal EMBA Vol . 11 No . 4 November 2023 , Hal . 677-685. 11(4)*, 677–685. <https://doi.org/https://doi.org/10.35794/emba.v11i4.51036>
- Setiawan, D., Nur, A., Purnamasari, R., & Dianta, I. A. (2023). Web-Based Inventory Management System Utilizing the First in First Out (Fifo) Method: a Case Study of Cv Berkah Foam Furniture. *Jurnal Ilmiah Teknologi Informasi Dan Komunikasi (Jtik)*, 14(2), 370–383. <https://doi.org/https://doi.org/10.51903/jtikp.v14i2.774>
- , D. W., & Ningtiyas, L. M. W. (2022). Analisis Penyimpanan dan Penempatan Barang Pada PT Bintang Dagang Internasional (Haistar) Surabaya Dalam Mendukung Ketersediaan Barang di Era Pandemi. *Jurnal Ilmiah Administrasi Bisnis Dan Inovasi*, 6(2), 100–114. <https://doi.org/10.25139/jiabi.v6i2.5212>