



Sistem Informasi Manajemen Logistik Kebencanaan Berbasis Web dengan Penerapan Algoritma First-In, First-Out (FIFO)

Muhammad Azir*, Irman Effendy, Leon Andretti Abdillah, Edi Supratman

Fakultas Sains Teknologi, Program Studi Sistem Informasi, Universitas Bina Darma, Palembang, Indonesia

Email: ^{1,*}221410112@student.binadarma.ac.id, ²irman.effendy@binadarma.ac.id, ³leon.abdillah@binadarma.ac.id,

⁴edi_supratman@binadarma.ac.id

Email Penulis Korespondensi: 221410112@student.binadarma.ac.id

Abstrak—Pengelolaan logistik kebencanaan pada BPBD Provinsi Sumatera Selatan masih bertumpu pada buku besar dan berkas *spreadsheet* lokal. Observasi dan wawancara dengan Kepala Bidang Logistik mengidentifikasi tiga permasalahan, yaitu pencatatan yang tidak terstandar antar-petugas, ketiadaan basis data terpusat dan *audit trail*, serta pembaruan data yang tidak *real-time* sehingga koordinasi distribusi terhambat. Penelitian ini membangun sistem informasi manajemen logistik berbasis *web* yang menerapkan algoritma *First-In, First-Out* (FIFO) sebagai kendali rotasi persediaan pada tingkat *batch*, sehingga setiap transaksi pengeluaran tertaut pada *batch* penerimaan asalnya dan risiko kedaluwarsa barang habis pakai dapat ditekan. Sistem dikembangkan dengan metode *Rapid Application Development* (RAD) dalam *timebox* 12 minggu menggunakan Laravel, Tailwind CSS, dan MySQL. Evaluasi dilakukan tiga lapis, yaitu pengujian fungsional *black box* atas sembilan fitur utama, evaluasi pengguna pada tiga iterasi prototipe bersama admin gudang, serta analisis kompleksitas dan skalabilitas distribusi FIFO. Seluruh fitur berfungsi sesuai spesifikasi; evaluasi pengguna menemukan sepuluh ketidaksesuaian data master yang tidak terjangkau pengujian fungsional; analisis kompleksitas menunjukkan waktu proses ditentukan oleh jumlah *batch* yang ditelusuri, bukan total *batch* tersimpan. Kontribusi penelitian ini mencakup tiga hal, yaitu (1) secara praktis, tersedianya sistem logistik kebencanaan yang menegakkan rotasi FIFO tingkat *batch* dan menautkan setiap pengeluaran ke *batch* penerimaan asalnya di dalam jejak audit yang siap diperiksa; (2) secara teoretis, perluasan penerapan FIFO dari persediaan sektor swasta ke domain logistik kebencanaan pemerintah dengan menempatkan tanggal penerimaan *batch* sebagai penanda pengganti tanggal kedaluwarsa; serta (3) secara metodologis, skema evaluasi tiga lapis beserta bukti empiris bahwa waktu respons distribusi FIFO ditentukan oleh jumlah *batch* aktif yang tersimpan, bukan oleh jumlah *batch* yang dikonsumsi.

Kata Kunci: Audit Trail; Logistik Kebencanaan; Algoritma FIFO; Sistem Informasi Manajemen; Rapid Application Development (RAD)

Abstract—Disaster logistics management at the Regional Disaster Management Agency (BPBD) of South Sumatra Province still relies on paper ledgers and local spreadsheet files. Observations and interviews with the Head of the Logistics Division identified three problems: non-standardized recording among officers, the absence of a centralized database and audit trail, and the lack of real-time updates that hampers distribution coordination. This study develops a web-based logistics management information system applying the First-In, First-Out (FIFO) algorithm as a batch-level inventory rotation control, so that every outgoing transaction is linked to its originating receipt batch, reducing the risk of consumable goods expiring. The system was developed using the Rapid Application Development (RAD) method within a 12-week timebox, built with Laravel, Tailwind CSS, and MySQL. Evaluation comprised three layers: black box functional testing of nine main features, user evaluation across three prototype iterations with the warehouse administrator, and a complexity and scalability analysis of FIFO distribution. All features performed according to specification; user evaluation revealed ten master-data mismatches beyond the reach of functional testing; and the complexity analysis shows that processing time depends on the number of batches traversed rather than the total stored. This study offers three contributions: (1) practically, it delivers a disaster logistics system that enforces batch-level FIFO rotation and links every outgoing transaction to its originating receipt batch within an audit-ready trail; (2) theoretically, it extends the application of FIFO from private-sector inventory to the government disaster logistics domain by positioning the batch receipt date as a proxy for the expiry date; and (3) methodologically, it provides a three-layer evaluation scheme together with empirical evidence that FIFO distribution response time is governed by the number of active batches stored rather than the number of batches consumed.

Keywords: Audit Trail; Disaster Logistics; FIFO Algorithm; Management Information System; Rapid Application Development (RAD)

1. PENDAHULUAN

Badan Penanggulangan Bencana Daerah (BPBD) Provinsi Sumatera Selatan mengelola logistik kebencanaan bersumber APBD dan BNPB dengan volume dan keragaman yang tinggi. Berdasarkan data gudang, komoditas habis pakai yang dikelola mencakup mie instan sebanyak 17.350 bungkus, sarden 6.940 kaleng, dan minyak goreng 1.735 kemasan, sedangkan perlengkapan non-habis pakai mencakup terpal lebih dari 400 lembar, kasur lipat 150 buah, jas hujan setelan 86 buah, dan alat pelindung diri pemadam kebakaran 20 set. Hingga penelitian ini dilaksanakan, seluruh data tersebut masih dicatat menggunakan buku besar dan berkas *spreadsheet* sederhana yang disimpan secara lokal pada komputer admin. Karakteristik logistik kebencanaan yang kritis terhadap waktu menuntut pencatatan yang cepat, akurat, dan terdokumentasi, terlebih karena sebagian besar komoditas habis pakai memiliki masa simpan terbatas sehingga rentan kedaluwarsa apabila rotasi stok tidak dikendalikan secara sistematis.

Observasi langsung pada gudang logistik dan wawancara dengan Kepala Bidang Logistik serta admin gudang BPBD Provinsi Sumatera Selatan mengidentifikasi tiga permasalahan operasional. Pertama, tidak tersedianya standar pencatatan antar-petugas menimbulkan kesalahan pengetikan, duplikasi entri, dan ketidakkonsistenan format ketika data dikelola secara bergantian oleh banyak pihak. Kedua, ketiadaan basis data terpusat dan jejak audit (*audit trail*) membuat data rentan hilang maupun rusak, serta memaksa admin merekapitulasi ulang catatan lapangan secara manual selama



berhari-hari menjelang audit. Ketiga, belum tersedianya sistem terpusat berbasis *web* menyebabkan pembaruan data persediaan tidak berlangsung secara *real-time* karena pelaporan petugas lapangan masih mengandalkan pesan teks WhatsApp. Ketiga kondisi tersebut berdampak langsung pada ketidakakuratan data persediaan dan lambatnya distribusi bantuan.

Kondisi empiris tersebut menjadi dasar peninjauan literatur pada bagian berikut. Secara teoretis, sistem informasi manajemen berperan sebagai penopang efisiensi operasional dan pengambilan keputusan organisasi [1], dan penerapannya pada instansi pemerintah terbukti mampu mengeliminasi duplikasi data, mempercepat pelaporan, serta menekan biaya operasional logistik [2]. Dalam manajemen persediaan dikenal beberapa metode pengendalian rotasi barang, yaitu *Last-In, First-Out* (LIFO), *First-Expired, First-Out* (FEFO), dan *First-In, First-Out* (FIFO). LIFO berisiko menyebabkan barang lama menumpuk hingga rusak, sedangkan FEFO menuntut pencatatan tanggal kedaluwarsa spesifik untuk setiap item, yang sulit dipenuhi ketika volume barang tinggi dan dokumen penyerta dari donor tidak selalu mencantumkan masa kedaluwarsa. FIFO memastikan barang yang lebih dahulu masuk dikeluarkan lebih dahulu sehingga rotasi persediaan berjalan teratur [3]. Penerapan FIFO secara konsisten pada lingkungan pergudangan terbukti meningkatkan akurasi data persediaan, mempercepat pengelolaan barang, dan menekan potensi kerugian finansial [4], sementara implementasinya pada sistem berbasis *web* mampu mengotomatisasi penentuan urutan pengeluaran berdasarkan tanggal penerimaan sehingga akurasi distribusi meningkat [5]. Perlu ditegaskan bahwa dalam penelitian ini FIFO diposisikan sebagai algoritma pengendalian rotasi persediaan yang diimplementasikan di dalam sistem, bukan sebagai metode pengembangan perangkat lunak.

Dari sisi rekayasa perangkat lunak, pemilihan metode pengembangan menentukan kecepatan penyelesaian sistem dan kesesuaiannya dengan kebutuhan pengguna. Metode *Waterfall* bersifat linier dan kaku sehingga perubahan kebutuhan di tengah pengembangan berbiaya tinggi, sedangkan *Agile* lebih tepat untuk produk perangkat lunak berskala besar dengan tim dan siklus rilis berkelanjutan. *Rapid Application Development* (RAD) yang digagas Martin [6] menekankan siklus prototipe iteratif, pembatasan waktu pengerjaan (*timeboxing*), dan keterlibatan intensif pengguna. Dibandingkan *Waterfall*, RAD terbukti lebih efektif memangkas waktu pengembangan tanpa menurunkan kualitas sistem [7], dan telah berhasil diterapkan pada pembangunan sistem informasi di lingkungan instansi publik yang kebutuhannya dinamis [8], [9]. Karakteristik tersebut sesuai dengan kondisi BPBD Provinsi Sumatera Selatan yang membutuhkan sistem operasional dalam waktu singkat dan memerlukan validasi berulang dari petugas gudang selaku pemilik pengetahuan operasional.

Sejumlah penelitian terdahulu telah membahas digitalisasi logistik maupun penerapan FIFO. Nizarudin dan Widodo [10] mendigitalisasi sistem logistik bencana pada BPBD Kabupaten Jepara menggunakan metode *Waterfall* dengan Laravel dan MySQL, serta berhasil menggantikan pencatatan manual dan menghasilkan laporan otomatis dengan tingkat kepuasan pengguna di atas 90%. Ardianti dkk. [11] membangun sistem informasi manajemen logistik dan peralatan kebencanaan multi-posko berbasis *web* dengan metode RAD yang dilengkapi notifikasi WhatsApp dan pemetaan visual posko. Mufti dkk. [12] menerapkan algoritma *Selection Sort* pada sistem informasi bantuan logistik korban bencana dan mencapai akurasi pengurutan data sebesar 100%. Pada konteks di luar kebencanaan, Lubis dkk. [13] menerapkan FIFO pada sistem persediaan barang jadi PT Rubberman Indonesia untuk mencegah penumpukan stok, sedangkan Pangastutik dkk. [14] menggunakan FIFO pada sistem laporan keuangan persediaan PT Usaha Maju sehingga akurasi nilai persediaan meningkat.

Apabila ditinjau dari kebutuhan pengendalian rotasi persediaan, kelima penelitian tersebut menyisakan celah yang berbeda-beda. Nizarudin dan Widodo [10] berhasil menggantikan pencatatan manual dengan basis data terpusat beserta pelaporan otomatis, namun sistem yang dibangun tidak memiliki kendali rotasi stok sehingga pengeluaran barang tidak terikat pada urutan penerimaan *batch*. Ardianti dkk. [11] memperluas cakupan koordinasi hingga tingkat multi-posko dan mempercepat penyampaian informasi melalui notifikasi WhatsApp serta pemetaan posko, tetapi tidak menerapkan algoritma FIFO dan tidak menautkan transaksi keluar ke *batch* asalnya. Mufti dkk. [12] mencapai akurasi pengurutan data sebesar 100%, namun pengurutan tersebut bekerja pada level tampilan data dan bukan sebagai pengendali urutan pengeluaran barang. Pada kelompok penelitian yang benar-benar menerapkan FIFO, Lubis dkk. [13] menempatkan FIFO pada persediaan barang jadi sektor manufaktur untuk mencegah penumpukan stok, tetapi konteksnya adalah perusahaan swasta dengan pemasok tetap serta tanpa kanal pengajuan publik maupun jejak audit instansi, sedangkan Pangastutik dkk. [14] mengarahkan FIFO pada penilaian akuntansi nilai persediaan, bukan pada pengendalian distribusi logistik kebencanaan.

Pemetaan tersebut memperlihatkan bahwa celah penelitian (*research gap*) tidak sekadar terletak pada ketiadaan fitur, melainkan pada tiga aspek yang lebih mendasar. Pertama, terdapat pemisahan domain: penelitian yang mengangkat konteks logistik kebencanaan pemerintah [10]-[12] berhenti pada digitalisasi pencatatan dan pengurutan tampilan data, tanpa mekanisme yang mengikat pengeluaran barang pada urutan penerimaannya, sehingga rotasi stok tetap bergantung pada kebijakan petugas. Kedua, penelitian yang benar-benar menerapkan FIFO [13], [14] berada pada sektor swasta dengan karakteristik persediaan bernilai jual dan pemasok tetap, berbeda dengan logistik kebencanaan yang penerimaannya bersifat sporadis, berasal dari banyak donor, dan tidak selalu disertai dokumen kedaluwarsa; konsekuensinya, pengendalian rotasi harus bersandar pada tanggal penerimaan batch, bukan pada tanggal kedaluwarsa. Ketiga, belum ada penelitian pada kedua kelompok tersebut yang menautkan setiap transaksi pengeluaran ke batch penerimaan asalnya sekaligus merekamnya dalam jejak audit, padahal keterlacakan tersebut merupakan syarat pertanggungjawaban keuangan negara pada instansi pemerintah. Ketiga aspek inilah yang membentuk kebaruan penelitian ini.

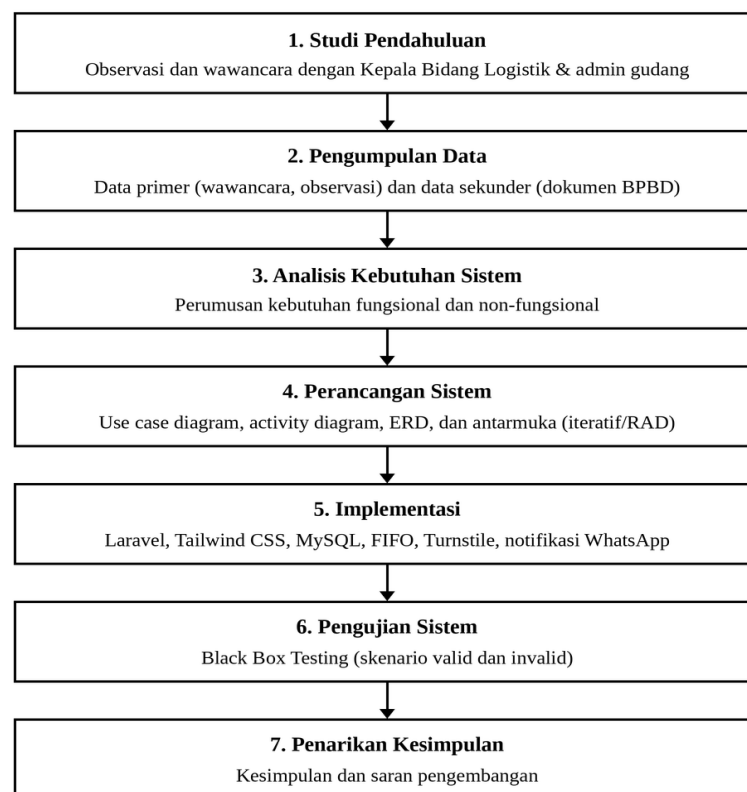
Berbeda dari kelima penelitian tersebut, penelitian ini memposisikan FIFO sebagai kendali rotasi persediaan pada tingkat *batch* yang menautkan setiap transaksi keluar ke *batch* penerimaan asalnya, kemudian mengintegrasikannya dengan jejak audit, kanal pengajuan publik terverifikasi, dan ekspor laporan siap audit di dalam satu arsitektur sistem yang dikembangkan melalui RAD ber-*timebox* pada instansi pemerintah. Kombinasi tersebut belum ditemukan pada penelitian terdahulu, baik pada kelompok yang mengangkat konteks logistik kebencanaan pemerintah maupun pada kelompok yang menerapkan FIFO di sektor swasta.

Berdasarkan celah tersebut, penelitian ini bertujuan untuk (1) membangun sistem informasi manajemen logistik berbasis *web* yang dirancang mengikuti alur kerja BPBD Provinsi Sumatera Selatan, mencakup pencatatan terstandar, jejak audit, kanal pengajuan logistik daring, dan pelaporan siap audit; (2) menerapkan algoritma FIFO berbasis *batch* pada proses pengeluaran barang sehingga setiap transaksi keluar tertaut pada *batch* penerimaan asalnya; serta (3) mengevaluasi sistem tidak hanya dari sisi kesesuaian fungsional, tetapi juga dari sisi evaluasi pengguna serta perilaku kompleksitas dan skalabilitas pada transaksi bervolume besar. Kebaruan penelitian ini terletak pada integrasi kendali rotasi FIFO tingkat *batch* dengan jejak audit dan kanal pengajuan publik dalam satu arsitektur sistem logistik kebencanaan pemerintah, yang belum ditemukan pada penelitian terdahulu sebagaimana diuraikan pada pemetaan di atas.

Sejalan dengan tujuan tersebut, kontribusi penelitian ini dirumuskan pada tiga tataran. Pada tataran praktis, penelitian ini menghasilkan sistem informasi manajemen logistik kebencanaan yang siap dioperasikan BPBD Provinsi Sumatera Selatan, yang menegakkan rotasi FIFO tingkat *batch*, merekam setiap perubahan data di dalam jejak audit, serta menghasilkan laporan siap audit tanpa rekapitulasi manual. Pada tataran teoretis, penelitian ini memperluas kajian penerapan FIFO dari persediaan sektor swasta ke domain logistik kebencanaan pemerintah dan menunjukkan bahwa tanggal penerimaan *batch* dapat berfungsi sebagai penanda pengganti (*proxy*) tanggal kedaluwarsa ketika dokumen penyerta dari donor tidak tersedia, sehingga keterlacakan *batch* merupakan prasyarat dan bukan pelengkap bagi penerapan FIFO yang dapat diaudit. Pada tataran metodologis, penelitian ini menyediakan skema evaluasi tiga lapis yang memadukan pengujian fungsional, evaluasi pengguna pada setiap iterasi RAD, serta analisis kompleksitas dan pengukuran skalabilitas; dari skema tersebut diperoleh bukti empiris bahwa waktu respons distribusi FIFO ditentukan oleh jumlah *batch* aktif yang tersimpan alih-alih jumlah *batch* yang dikonsumsi, yang menjadi dasar rekomendasi pengindeksan bagi implementasi berskala besar.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian



Gambar 1. Tahapan Penelitian

Penelitian ini dilaksanakan melalui tujuh tahapan sistematis sebagaimana ditunjukkan pada Gambar 1. Tahap pertama adalah studi pendahuluan melalui observasi langsung gudang logistik dan wawancara mendalam dengan Kepala



Bidang Logistik serta admin gudang BPBD Provinsi Sumatera Selatan untuk memetakan permasalahan operasional dan kebutuhan sistem. Tahap kedua adalah pengumpulan data, mencakup data primer dari wawancara dan observasi lapangan serta data sekunder dari dokumentasi operasional BPBD (buku gudang, laporan Excel, dan surat serah terima) dan studi literatur terhadap jurnal ilmiah maupun dokumentasi teknis *framework*. Tahap ketiga adalah analisis kebutuhan sistem untuk merumuskan kebutuhan fungsional dan non-fungsional, meliputi alur kerja pencatatan logistik, format laporan, dan mekanisme pengajuan publik. Tahap keempat adalah perancangan sistem yang meliputi pemodelan proses menggunakan *Unified Modeling Language* (UML) [15] berupa *use case diagram* [16] dan *activity diagram* [17], pemodelan basis data menggunakan *Entity Relationship Diagram* (ERD) [18], serta perancangan antarmuka yang dilakukan secara iteratif sesuai karakteristik RAD; setiap artefak perancangan divalidasi terlebih dahulu sebelum tahap konstruksi sebagaimana diuraikan pada Subbab 2.3. Tahap kelima adalah implementasi, yaitu penerjemahan rancangan ke dalam kode program, termasuk integrasi logika FIFO, Cloudflare Turnstile, dan notifikasi WhatsApp. Tahap keenam adalah pengujian sistem yang dilaksanakan dalam tiga lapis, yaitu pengujian fungsional *black box*, evaluasi pengguna pada setiap iterasi prototipe, serta analisis kompleksitas dan skalabilitas proses distribusi FIFO, sebagaimana dijelaskan pada Subbab 2.5. Tahap ketujuh adalah penarikan kesimpulan sebagai jawaban atas rumusan masalah beserta saran pengembangan selanjutnya.

2.2 Metode Pengembangan Sistem

Sistem dikembangkan menggunakan metode *Rapid Application Development* (RAD) [6] yang mengedepankan siklus iteratif, kolaborasi intensif dengan pengguna, dan pembatasan waktu pengerjaan [7]. Keseluruhan pengembangan dilaksanakan dalam *timebox* total 12 minggu yang terbagi ke dalam empat tahap dengan durasi tetap. Apabila suatu kebutuhan tidak dapat diselesaikan dalam *timebox* berjalan, kebutuhan tersebut tidak memperpanjang durasi tahap, melainkan digeser ke iterasi berikutnya atau dikeluarkan dari lingkup (*descoping*) setelah disepakati bersama admin gudang; aturan ini menjadi kendali agar iterasi RAD tidak melebar. Tahap pertama, *requirements planning* (2 minggu), dilakukan melalui observasi gudang logistik dan wawancara dengan Kepala Bidang Logistik serta admin gudang, dengan keluaran berupa identifikasi lima permasalahan sistem berjalan beserta solusi pada sistem usulan. Tahap kedua, *user design* (3 minggu), dilaksanakan dalam tiga iterasi prototipe yang masing-masing berdurasi satu minggu dengan pola empat hari perbaikan prototipe dan satu sesi demonstrasi-evaluasi bersama admin gudang pada akhir iterasi; pelibatan pengguna sejak dini menekan risiko kesalahan desain sebelum tahap pengkodean [9]. Tahap ketiga, *construction* (6 minggu), dibagi menjadi tiga *timebox* dua minggu berdasarkan prioritas fitur, yaitu modul data master dan pencatatan barang masuk; modul distribusi barang dengan algoritma FIFO beserta jejak audit; serta modul pengajuan publik, notifikasi, dan pelaporan. Konstruksi menggunakan *framework* Laravel [19] dan Tailwind CSS [20] dengan integrasi Cloudflare Turnstile [21]. Tahap keempat, *cutover* (1 minggu), mencakup pengujian akhir, pemindahan data persediaan awal, penyerahan sistem kepada BPBD Provinsi Sumatera Selatan, dan pendampingan penggunaan kepada admin gudang.

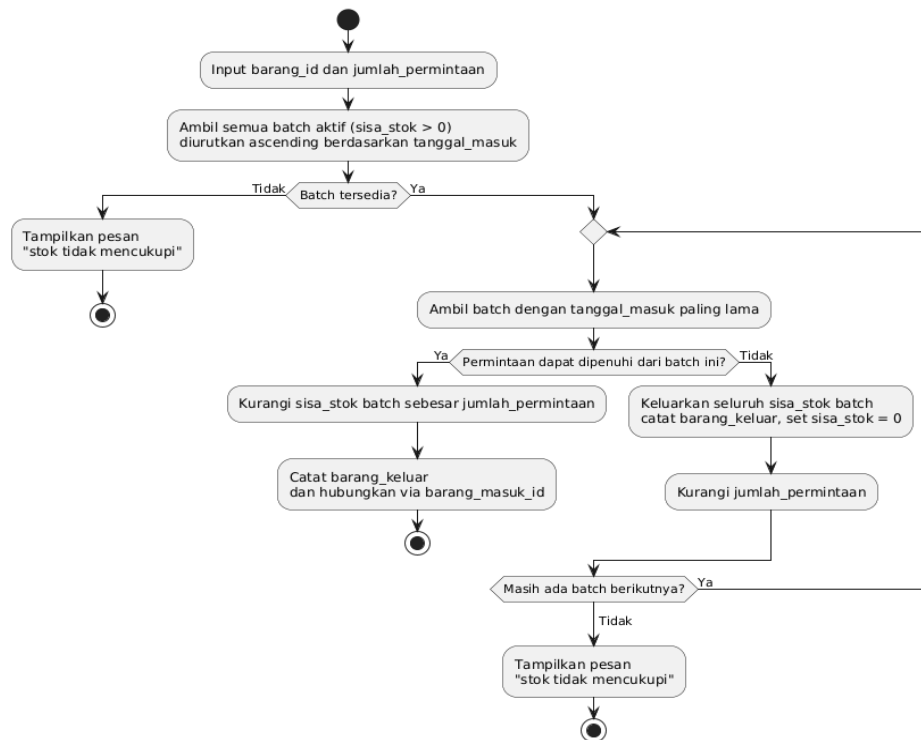
2.3 Validasi Model Perancangan

Agar model konseptual tidak berhenti pada tataran dokumentasi, setiap artefak perancangan divalidasi sebelum tahap konstruksi. Validasi *use case diagram* dan *activity diagram* dilakukan melalui penelusuran proses bisnis (*business process walkthrough*) bersama Kepala Bidang Logistik dan admin gudang, yaitu dengan menjalankan enam skenario proses bisnis riil berupa penerimaan bantuan dari donor, pencatatan batch baru, permintaan bantuan dari masyarakat, verifikasi pengajuan, pengeluaran barang beserta berita acara, dan penyusunan laporan periodik. Setiap langkah pada skenario diperiksa keberpasangannya dengan aktivitas pada model, dan ketidaksesuaian yang ditemukan dicatat sebagai temuan untuk diperbaiki pada iterasi berikutnya. Validasi ERD dilakukan melalui dua mekanisme, yaitu normalisasi hingga bentuk normal ketiga (3NF) untuk menghilangkan redundansi dan anomali pemuakhiran data, serta pemeriksaan keterlacakan (*traceability check*) yang memastikan setiap kebutuhan fungsional dapat dipetakan ke tabel dan relasi yang tersedia, khususnya keterkaitan tabel barang_keluars dengan barang_masuks yang menjadi dasar penerapan FIFO. Integritas referensial hasil rancangan selanjutnya diverifikasi pada tahap konstruksi melalui pengujian *foreign key constraint*, yaitu dengan mencoba menghapus data induk yang masih memiliki data anak dan memastikan sistem menolak operasi tersebut.

2.4 Penerapan Algoritma FIFO

Manajemen logistik merupakan proses strategis yang mengintegrasikan manajemen permintaan, manajemen persediaan, dan manajemen pergudangan dalam satu sistem terpadu untuk memastikan ketersediaan produk secara tepat waktu, tepat jumlah, dan tepat tempat [22]. Pada sistem yang dibangun, metode FIFO diimplementasikan pada dua level. Pada level basis data, setiap transaksi barang masuk dicatat dalam tabel barang_masuks sebagai satu *batch* dengan atribut tanggal_masuk sebagai kunci pengurutan dan sisa_stok untuk memantau sisa kuantitas *batch*.

Pada level logika aplikasi, fungsi distribusi barang melakukan iterasi terhadap seluruh *batch* aktif ($\text{sisa_stok} > 0$) suatu barang yang diurutkan secara *ascending* berdasarkan tanggal_masuk. Apabila jumlah permintaan dapat dipenuhi dari *batch* pertama, nilai *sisa_stok batch* tersebut langsung dikurangi dan transaksi keluar dihubungkan ke *batch* asal melalui kolom barang_masuk_id; apabila tidak mencukupi, sistem secara otomatis mengambil sisa dari *batch* berikutnya secara berurutan hingga jumlah permintaan terpenuhi atau stok habis. Alur lengkap implementasi algoritma FIFO pada sistem disajikan pada Gambar 2.



Gambar 2. Alur Implementasi Algoritma FIFO pada Sistem

2.5 Pengujian Sistem

Pengujian sistem dilaksanakan dalam tiga lapis agar evaluasi tidak berhenti pada kesesuaian fungsional. Lapis pertama adalah *Black Box Testing* [23], yaitu pengujian yang berfokus pada spesifikasi fungsional tanpa memperhatikan struktur kode program internal. Sembilan fitur utama diuji dengan skenario berhasil (*valid case*) dan skenario gagal (*invalid case*), meliputi *login*, manajemen data barang, pencatatan barang masuk, distribusi barang dengan FIFO, pengajuan logistik publik, verifikasi pengajuan, ekspor laporan PDF dan Excel, notifikasi stok menipis, serta log *audit trail*. Perlu digarisbawahi bahwa *Black Box Testing* hanya membuktikan kesesuaian keluaran terhadap spesifikasi dan tidak dapat dijadikan indikator kualitas internal maupun kinerja sistem. Lapis kedua adalah evaluasi pengguna yang melekat pada siklus iterasi RAD. Pada akhir setiap iterasi, prototipe didemonstrasikan kepada admin gudang selaku pengguna utama, kemudian pengguna menjalankan skenario operasional gudang pada prototipe tersebut dan menyampaikan penilaian atas kesesuaiannya dengan praktik kerja sehari-hari. Setiap masukan dicatat beserta tindak lanjutnya sehingga dapat ditelusuri kembali. Berbeda dengan lapis pertama, kasus uji pada lapis ini tidak ditentukan oleh pengembang, melainkan tumbuh dari pengetahuan operasional pengguna. Lapis ketiga adalah analisis kompleksitas dan skalabilitas terhadap proses distribusi FIFO, yang merupakan proses paling mahal secara komputasi karena melibatkan iterasi lintas *batch* serta penulisan banyak baris transaksi dalam satu rangkaian. Analisis dilaksanakan dalam dua bagian yang saling melengkapi. Bagian pertama bersifat analitis terhadap kompleksitas waktu algoritma, jumlah operasi tulis yang dibangkitkan, dan rencana eksekusi kueri pengambilan batch aktif. Bagian kedua berupa pengukuran empiris waktu respons pada empat tingkat volume batch aktif, yaitu 100, 500, 1.000, dan 5.000 batch, dengan 30 pengulangan pada setiap tingkat ditambah tiga pengulangan pemanasan yang tidak dicatat. Dua operasi diukur terpisah, yaitu seleksi batch aktif dan alokasi lengkap yang mencakup seleksi disertai pemutakhiran *sisa_stok* serta penyisipan baris barang keluar. Jumlah permintaan dikunci pada nilai tetap sehingga banyaknya batch yang dikonsumsi selalu tiga pada seluruh tingkat volume, dengan demikian satu-satunya variabel yang berubah adalah jumlah batch aktif yang tersimpan. Setiap pengulangan dijalankan di dalam transaksi yang dibatalkan (*rollback*) sehingga kondisi basis data kembali identik sebelum pengulangan berikutnya, dan waktu dicatat pada lapis kueri sehingga tidak mencakup waktu render antarmuka maupun latensi jaringan.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Requirements Planning

Hasil observasi pada tahap *requirements planning* menunjukkan bahwa pencatatan logistik di BPBD Provinsi Sumatera Selatan masih dilakukan secara manual. Barang bantuan yang diterima dicatat oleh petugas gudang ke dalam buku besar penerimaan, kemudian direkapitulasi ulang ke dalam berkas *spreadsheet* yang disimpan secara lokal pada komputer admin. Permintaan bantuan disampaikan melalui pesan WhatsApp atau surat permohonan tercetak, lalu diverifikasi secara lisan oleh Kepala Bidang Logistik. Pengeluaran barang dicatat pada buku pengeluaran disertai berita acara serah terima yang diketik ulang secara manual. Pada akhir periode, seluruh catatan tersebut direkapitulasi kembali untuk menyusun



laporan persediaan bagi keperluan audit. Berdasarkan kondisi tersebut, teridentifikasi lima permasalahan utama beserta solusi yang diterapkan pada sistem usulan sebagaimana disajikan pada Tabel 1. Kelima permasalahan tersebut menjadi dasar penyusunan kebutuhan fungsional sistem, yang meliputi pengelolaan data master barang dan kategori, pencatatan barang masuk sebagai batch, distribusi barang keluar dengan rekomendasi FIFO, pengajuan logistik publik beserta verifikasinya, pembuatan berita acara, ekspor laporan PDF dan Excel, notifikasi stok menipis, serta pencatatan audit trail. Adapun kebutuhan non-fungsional menekankan antarmuka yang responsif, kemudahan penggunaan bagi admin gudang, serta keamanan formulir publik dari pengiriman otomatis.

Tabel 1. Permasalahan Sistem Berjalan dan Solusi pada Sistem Usulan

No	Permasalahan Sistem Berjalan	Solusi pada Sistem Usulan
1	Pencatatan tidak terstandar karena tidak ada format baku dan diisi bergantian oleh beberapa petugas	Formulir <i>input</i> tervalidasi dengan data <i>master</i> barang dan satuan yang baku
2	Data tersimpan pada berkas <i>spreadsheet</i> lokal tanpa cadangan dan tanpa jejak perubahan	Basis data MySQL terpusat dilengkapi <i>audit trail</i> untuk setiap aktivitas tambah, ubah, dan hapus
3	Rekapitulasi laporan dilakukan manual sehingga memakan waktu sehari-hari	Laporan persediaan dan transaksi dapat diekspor ke PDF dan Excel secara langsung
4	Pengeluaran barang tidak mengacu pada tanggal penerimaan sehingga barang lama menumpuk	Pengeluaran otomatis mengikuti urutan <i>batch</i> penerimaan dengan metode FIFO
5	Permintaan bantuan disampaikan melalui pesan teks yang tidak terarsip	Formulir pengajuan daring terverifikasi Cloudflare Turnstile disertai notifikasi WhatsApp

3.2 Hasil User Design

Sesuai karakteristik metode RAD, rancangan sistem tidak disusun sekali jadi, melainkan melalui siklus demonstrasi prototipe dan evaluasi bersama admin gudang selaku pengguna utama sistem. Rangkuman iterasi yang dilaksanakan disajikan pada Tabel 2. Iterasi ketiga menunjukkan peran penting pelibatan pengguna dalam metode RAD, karena ketidaksesuaian satuan barang dan daftar komoditas merupakan pengetahuan operasional yang hanya dimiliki petugas gudang dan tidak dapat diidentifikasi melalui kajian dokumen semata. Pemodelan yang dihasilkan pada tahap ini mencakup use case diagram dengan tiga aktor (admin, atasan, dan masyarakat/petugas lapangan), activity diagram untuk masing-masing aktor, serta ERD yang menjadi dasar pembentukan sembilan tabel basis data. Pada iterasi tersebut, satuan mie instan diubah dari dus menjadi bungkus, sarden menjadi kaleng, gula pasir, minyak goreng, dan tepung gandum menjadi kilogram, kecap dan saus menjadi botol, serta *goodie bag* menjadi lembar. Komoditas beras dan susu dihapus melalui mekanisme *soft delete* karena tidak dikelola gudang BPBD, sehingga keduanya tidak lagi ditampilkan pada antarmuka namun riwayat transaksinya tetap terpelihara di dalam basis data.

Tabel 2. Iterasi Prototipe dan Evaluasi Pengguna

Iterasi	Masukan Pengguna	Tindak Lanjut pada Sistem
1	Pencatatan manual rawan kesalahan dan laporan audit harus dirapikan ulang, sehingga diperlukan format yang terstandar	Perumusan kebutuhan fungsional serta penetapan struktur data barang, <i>batch</i> penerimaan, dan <i>audit trail</i>
2	Rancangan antarmuka dinilai telah sesuai dengan alur kerja gudang	Rancangan dilanjutkan ke tahap konstruksi tanpa perubahan struktural
3	Satuan barang belum sesuai praktik gudang; komoditas beras dan susu tidak dikelola gudang BPBD	Nilai kolom satuan pada data <i>master</i> barang dimutakhirkan; beras dan susu dihapus melalui <i>soft delete</i>

3.3 Implementasi Sistem dan Basis Data

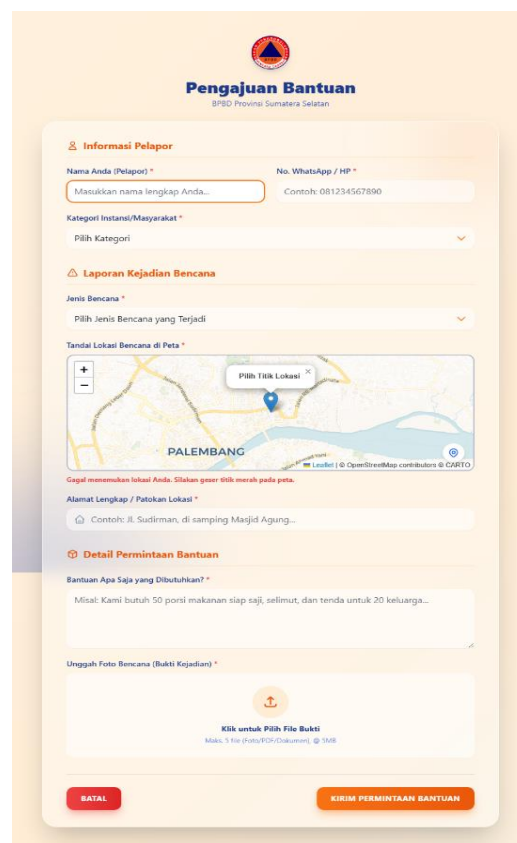
Implementasi sistem dilakukan berdasarkan hasil perancangan menggunakan metode RAD. Sistem dibangun berbasis web dengan framework Laravel pada sisi server (backend), Tailwind CSS untuk antarmuka pengguna (frontend), serta MySQL sebagai sistem manajemen basis data. Sistem dikembangkan pada Laravel 13.8 dan MySQL 8.0, sedangkan seluruh pengukuran kinerja yang dilaporkan pada Subbab 3.9 dilaksanakan setelah pemutakhiran dependensi, yaitu pada Laravel 13.15.0, MySQL 8.4.3, dan PHP 8.5.4. Lingkungan pengembangan menggunakan Apache melalui Laragon pada sistem operasi Windows 11 (64-bit) dengan perangkat keras Intel Core i5 Generasi ke-12, RAM 12 GB, dan penyimpanan SSD, dengan peramban pengujian Google Chrome. Layanan pendukung yang diintegrasikan meliputi Cloudflare Turnstile untuk perlindungan formulir publik dan *gateway* notifikasi WhatsApp. Laravel dipilih karena mampu menyederhanakan kompleksitas pengembangan berkat ekosistem *library* yang lengkap serta fitur keamanan bawaan [19], sedangkan Tailwind CSS mempercepat pembangunan antarmuka yang responsif dan konsisten tanpa memerlukan banyak kode CSS manual [20].

Basis data diimplementasikan pada MySQL sesuai rancangan ERD, yang terdiri atas sembilan tabel yang saling berelasi, yaitu *users*, *kategori*, *barang*, *barang_masuks*, *barang_keluars*, *pengajuans*, *master_data*, *audit_trail*, dan *notifications*. Seluruh tabel dibuat melalui mekanisme *migration* pada Laravel sehingga struktur basis data konsisten dengan rancangan. Tabel *barang_masuks* menjadi inti implementasi FIFO karena menyimpan atribut *tanggal_masuk* dan

sisa_stok yang menjadi dasar pengurutan pengeluaran barang. Relasi antar-tabel diterapkan menggunakan *foreign key* untuk menjaga integritas referensial data: tabel barangs berelasi ke kategoris, tabel barang_masuks dan barang_keluars berelasi ke barangs, sedangkan tabel barang_keluars juga terhubung langsung ke barang_masuks melalui kolom barang_masuk_id sehingga setiap transaksi pengeluaran dapat ditelusuri ke *batch* asalnya sesuai urutan FIFO. Beberapa tabel, seperti barangs, barang_masuks, barang_keluars, dan pengajuans, dilengkapi kolom *deleted_at* untuk mendukung mekanisme *soft delete* sehingga data yang dihapus tetap dapat dipulihkan dan tidak langsung hilang secara permanen.

3.4 Implementasi Antarmuka Halaman Publik

Antarmuka halaman publik dapat diakses tanpa autentikasi. Halaman beranda memuat logo dan nama BPBD Provinsi Sumatera Selatan, sambutan dan penjelasan ringkas layanan pengajuan bantuan logistik, serta tombol menuju formulir pengajuan; tampilannya responsif sehingga tetap tertata rapi ketika diakses melalui perangkat dengan ukuran layar berbeda. Halaman formulir pengajuan pada Gambar 3 menampilkan isian data pemohon (nama, instansi/desa, dan kontak), fitur unggah surat permohonan, serta pemilihan barang beserta jumlah kebutuhan. Sebelum data dikirim, pengguna wajib melewati verifikasi Cloudflare Turnstile yang tampil di atas tombol “Ajukan Bantuan”, sehingga formulir terlindungi dari pengiriman massal otomatis oleh *bot* tanpa mengganggu pengalaman pengguna yang sah [21]. Halaman *login* disediakan bagi pengguna internal (admin dan atasan) dengan kolom email dan kata sandi beserta pesan kesalahan apabila kredensial tidak valid.

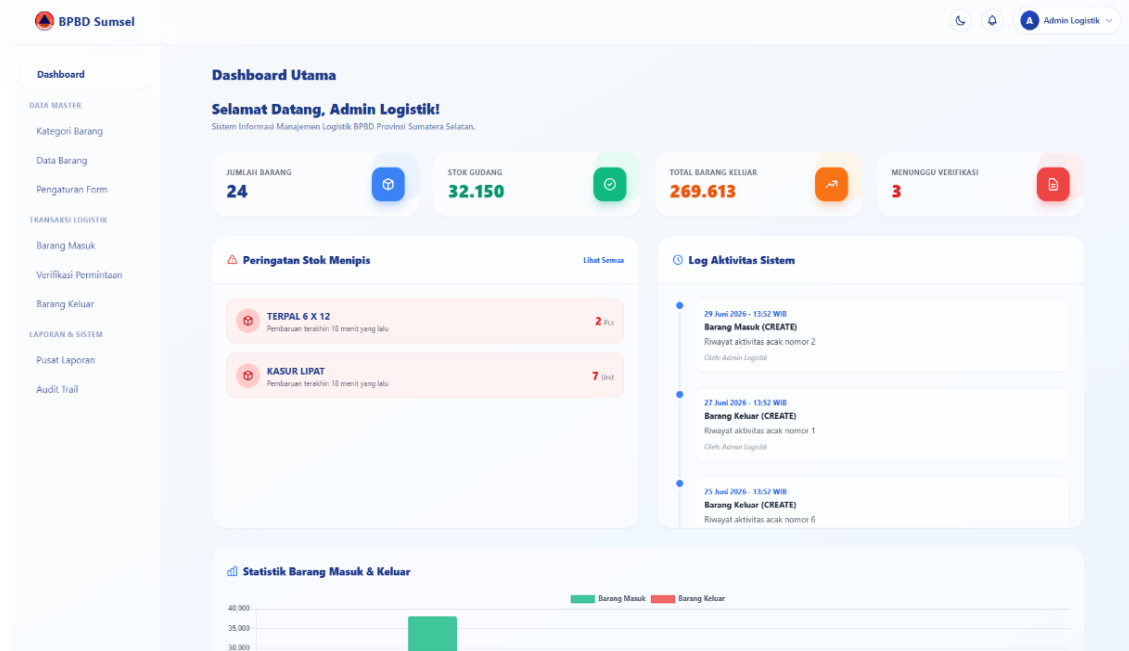


Gambar 3. Tampilan Halaman Formulir Pengajuan Publik

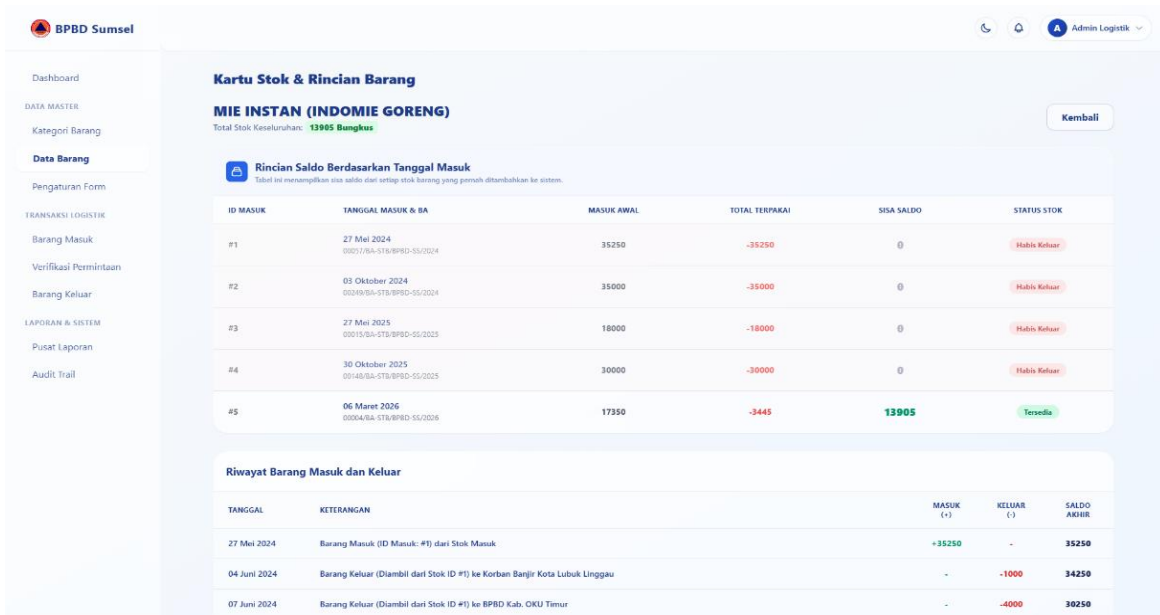
3.5 Implementasi Antarmuka Halaman Admin

Antarmuka halaman admin merealisasikan seluruh fitur operasional sistem dan hanya dapat diakses setelah *login*, dengan kerangka tata letak yang konsisten berupa *sidebar* navigasi, *header*, dan area konten utama. Halaman *dashboard* pada Gambar 4 menyajikan ringkasan informasi kritis secara *real-time* melalui kartu statistik, meliputi total barang, total stok, pengajuan tertunda, dan notifikasi stok menipis, sehingga admin memperoleh gambaran kondisi gudang secara sekilas begitu masuk ke sistem. Halaman notifikasi stok menipis menampilkan daftar barang dengan jumlah stok mendekati atau di bawah batas minimal, memuat nama barang, sisa stok, dan batas minimal, sehingga admin dapat segera menindaklanjuti dengan pengadaan. Halaman manajemen kategori dan manajemen barang menyajikan data dalam tabel yang dapat dicari dan diurutkan, lengkap dengan aksi tambah, ubah, dan hapus serta validasi masukan agar data yang tersimpan konsisten.

Halaman kartu stok pada Gambar 5 menampilkan rincian *batch* tiap barang beserta tanggal masuk dan sisa stok; urutan *batch* ditampilkan berdasarkan tanggal masuk sehingga admin dapat memantau langsung penerapan metode FIFO. Halaman daftar barang masuk menampilkan riwayat seluruh transaksi penerimaan barang, sedangkan formulir barang masuk mencatat setiap penerimaan sebagai satu *batch* baru yang memuat pemilihan barang, jumlah, tanggal masuk, asal barang, keterangan, dan unggah foto dokumentasi; setiap pencatatan tersebut menjadi dasar pengurutan FIFO.



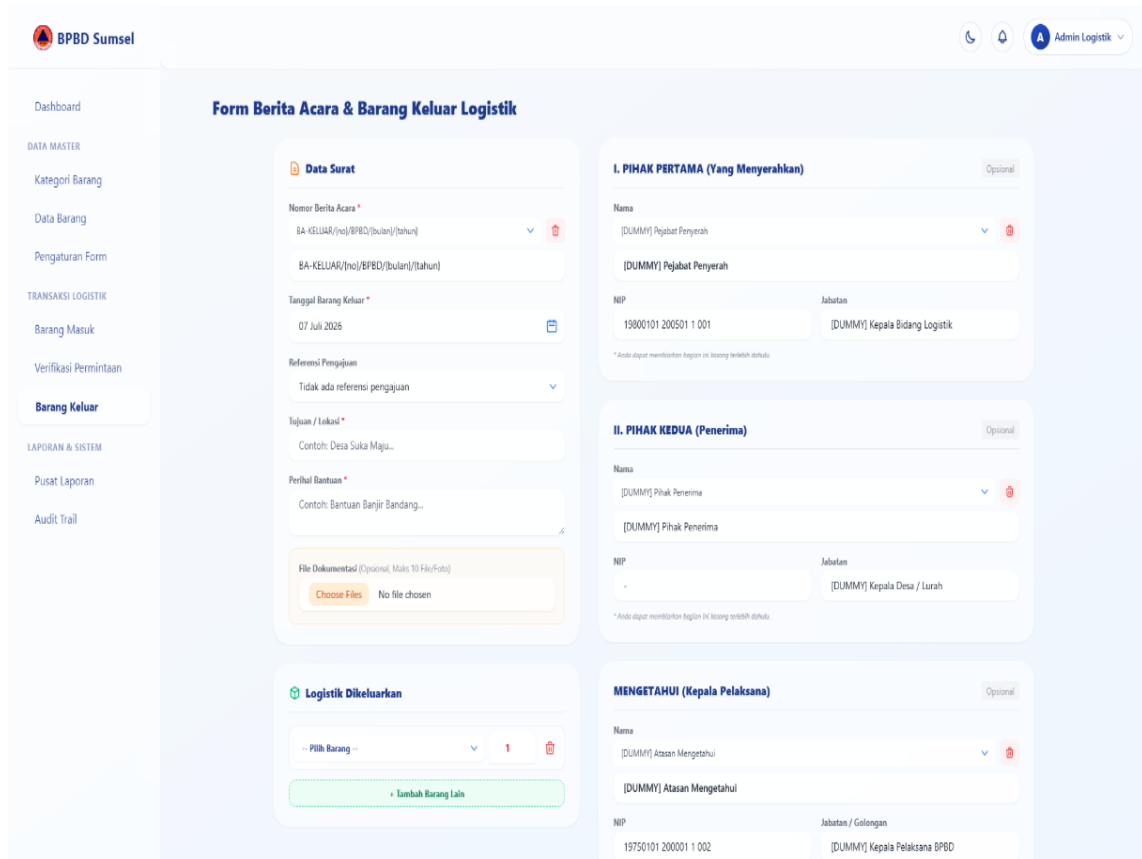
Gambar 4. Tampilan Halaman Dashboard



Gambar 5. Tampilan Halaman Kartu Stok (Batch FIFO)

Halaman formulir barang keluar pada Gambar 6 menyajikan rekomendasi urutan pengeluaran berdasarkan metode FIFO. Ketika admin memilih barang dan memasukkan jumlah, sistem menampilkan *batch* dengan tanggal masuk terlama sebagai prioritas, sehingga pengeluaran mengikuti kaidah *first-in, first-out*. Halaman daftar barang keluar kemudian menampilkan riwayat seluruh transaksi pengeluaran yang memuat nama barang, jumlah, tanggal keluar, dan tujuan sehingga distribusi barang mudah ditelusuri.

Halaman daftar pengajuan menampilkan seluruh pengajuan logistik dari masyarakat maupun petugas lapangan, memuat nama pengaju, lokasi, tanggal, dan status pengajuan sebagai titik awal tindak lanjut admin. Pada halaman detail pengajuan, admin dapat melihat rincian permintaan beserta bukti surat yang diunggah serta melakukan verifikasi berupa persetujuan atau penolakan; pengajuan yang disetujui diteruskan ke proses pembuatan berita acara pengeluaran barang. Halaman *master data* menyimpan identitas instansi dan pejabat penanda tangan sehingga dokumen resmi yang dihasilkan sistem selalu konsisten. Halaman laporan pada Gambar 7 menyediakan fungsi ekspor laporan persediaan dan transaksi; pengguna dapat memilih jenis laporan dan periode, lalu mengunduhnya dalam format PDF maupun Excel yang siap digunakan untuk keperluan audit. Setiap pengajuan yang masuk dari formulir publik juga diteruskan sistem sebagai notifikasi otomatis ke WhatsApp admin, sehingga admin dapat segera membuka halaman verifikasi tanpa harus memeriksa sistem secara berkala. Seluruh riwayat pengajuan, baik yang disetujui maupun yang ditolak, tetap tersimpan sebagai arsip digital yang dapat ditelusuri kembali kapan saja.



BPBD Sumsel

Form Berita Acara & Barang Keluar Logistik

Data Surat

Nomor Berita Acara *

BA-KELUAR/(no)/BPBD/(bulan)/(tahun)

BA-KELUAR/(no)/BPBD/(bulan)/(tahun)

Tanggal Barang Keluar *

07 Juli 2026

Referensi Pengajuan

Tidak ada referensi pengajuan

Tujuan / Lokasi *

Contoh: Desa Suka Maju...

Perihal Bantuan *

Contoh: Bantuan Barjir Bandang...

File Dokumentasi (Optional, Maks 10 File/Foto)

Choose Files No file chosen

I. PIHAK PERTAMA (Yang Menyerahkan)

Nama

[DUMMY] Pejabat Penyerah

[DUMMY] Pejabat Penyerah

NIP

19800101 200501 1 001

Jabatan

[DUMMY] Kepala Bidang Logistik

II. PIHAK KEDUA (Penerima)

Nama

[DUMMY] Pihak Penerima

[DUMMY] Pihak Penerima

NIP

-

Jabatan

[DUMMY] Kepala Desa / Lurah

Logistik Dikeluarkan

Pilih Barang

+ Tambah Barang Lain

MENGETAHUI (Kepala Pelaksana)

Nama

[DUMMY] Atasan Mengetahui

[DUMMY] Atasan Mengetahui

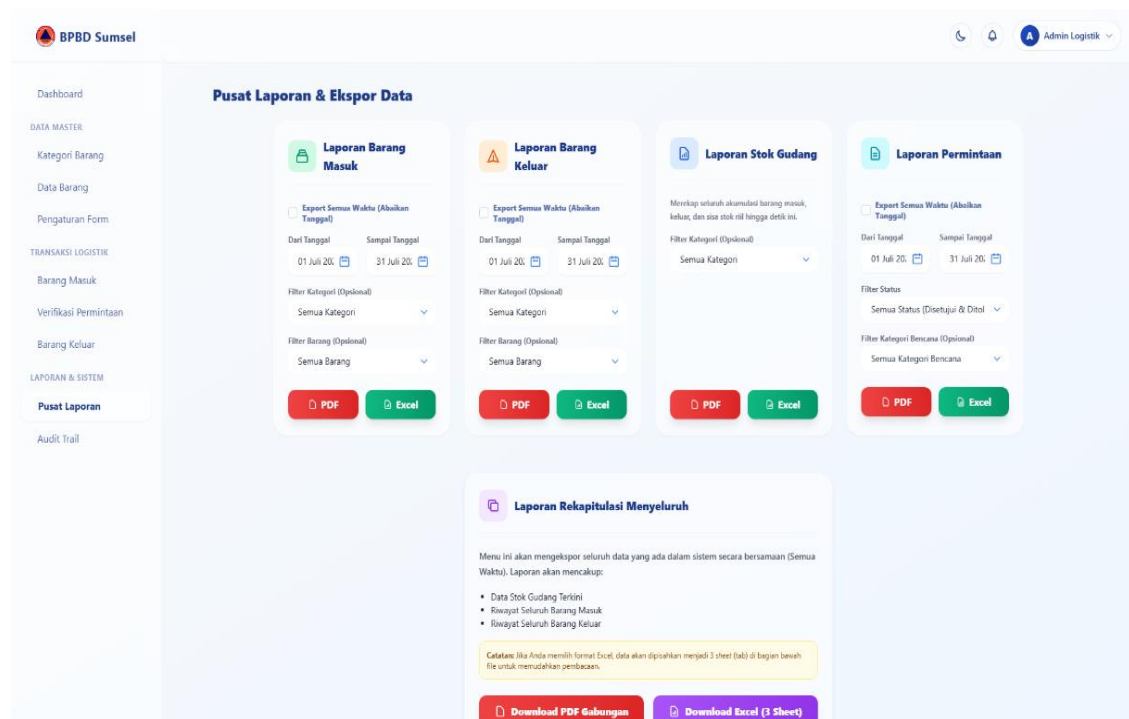
NIP

19750101 200001 1 002

Jabatan / Golongan

[DUMMY] Kepala Pelaksana BPBD

Gambar 6. Tampilan Halaman Formulir Barang Keluar (FIFO)



BPBD Sumsel

Pusat Laporan & Ekspor Data

Laporan Barang Masuk

Export Semua Waktu (Abaikan Tanggal)

Dari Tanggal

01 Juli 2026

Sampai Tanggal

31 Juli 2026

Filter Kategori (Opsional)

Semua Kategori

Filter Barang (Opsional)

Semua Barang

PDF Excel

Laporan Barang Keluar

Export Semua Waktu (Abaikan Tanggal)

Dari Tanggal

01 Juli 2026

Sampai Tanggal

31 Juli 2026

Filter Kategori (Opsional)

Semua Kategori

Filter Barang (Opsional)

Semua Barang

PDF Excel

Laporan Stok Gudang

Menyaji seluruh akumulasi barang masuk, keluar, dan sisa stok nil hingga detik ini.

Filter Kategori (Opsional)

Semua Kategori

PDF Excel

Laporan Permintaan

Export Semua Waktu (Abaikan Tanggal)

Dari Tanggal

01 Juli 2026

Sampai Tanggal

31 Juli 2026

Filter Status

Semua Status (Disetujui & Ditol)

Filter Kategori Bencana (Opsional)

Semua Kategori Bencana

PDF Excel

Laporan Rekapitulasi Menyeluruh

Menu ini akan mengeksport seluruh data yang ada dalam sistem secara bersamaan (Semua Waktu). Laporan akan mencakup:

- Data Stok Gudang Terkini
- Riwayat Seluruh Barang Masuk
- Riwayat Seluruh Barang Keluar

Catatan: Jika Anda memilih format Excel, data akan dipisahkan menjadi 3 sheet (tab) di bagian bawah file untuk memudahkan pembacaan.

Download PDF Gabungan Download Excel (3 Sheet)

Gambar 7. Tampilan Halaman Laporan

Terakhir, halaman log *audit trail* pada Gambar 8 menampilkan riwayat seluruh aktivitas perubahan data, memuat pengguna, jenis aksi (*create, update, delete*), tabel terkait, dan waktu perubahan, sehingga keterlacakan dan akuntabilitas data terjaga. Dengan adanya log ini, setiap perubahan data persediaan dapat ditelusuri kembali kepada pengguna yang melakukannya; apabila ditemukan ketidaksesuaian antara catatan sistem dan kondisi fisik gudang, admin maupun atasan dapat mengaudit riwayat perubahannya secara langsung tanpa memeriksa berkas-berkas terpisah seperti pada sistem berjalan.

WAKTU	PENGUNA	MODUL	AKSI	DETAIL PERUBAHAN
29 Juni 2026 13:52:34	Admin Logistik	Barang Masuk	CREATE	Riwayat aktivitas awal nomor 2
27 Juni 2026 13:52:34	Admin Logistik	Barang Keluar	CREATE	Riwayat aktivitas awal nomor 1
25 Juni 2026 13:52:34	Admin Logistik	Barang Keluar	CREATE	Riwayat aktivitas awal nomor 6
24 Juni 2026 13:52:34	Admin Logistik	Barang Keluar	CREATE	Riwayat aktivitas awal nomor 7
23 Juni 2026 13:52:34	Admin Logistik	Autentikasi	CREATE	Riwayat aktivitas awal nomor 8
23 Juni 2026 13:52:34	Admin Logistik	Autentikasi	CREATE	Riwayat aktivitas awal nomor 10
18 Juni 2026 13:52:34	Admin Logistik	Barang Keluar	CREATE	Riwayat aktivitas awal nomor 3
15 Juni 2026 13:52:34	Admin Logistik	Pengajuan	CREATE	Riwayat aktivitas awal nomor 5
14 Juni 2026 13:52:34	Admin Logistik	Autentikasi	CREATE	Riwayat aktivitas awal nomor 9
07 Juni 2026 13:52:34	Admin Logistik	Autentikasi	CREATE	Riwayat aktivitas awal nomor 4

Gambar 8. Tampilan Halaman Log Audit Trail

3.6 Implementasi Metode FIFO

Metode FIFO diimplementasikan pada proses pengeluaran barang dalam dua langkah utama sebagaimana ditunjukkan pada kode program. Langkah pertama adalah pengecekan stok sebelum eksekusi untuk mencegah *partial commit*: sistem mengakumulasi kebutuhan per barang dan membandingkannya dengan total sisa_stok seluruh *batch*; apabila jumlah yang diminta melebihi stok tersedia, transaksi ditolak dan admin menerima pesan kesalahan yang informatif. Langkah kedua adalah eksekusi pengurangan stok per *batch*: sistem mengambil seluruh *batch* aktif ($\text{sisa_stok} > 0$) dari suatu barang, mengurutkannya secara *ascending* berdasarkan tanggal_masuk (dengan id sebagai penentu urutan *batch* yang masuk pada tanggal yang sama), kemudian mengurangi stok mulai dari *batch* terlama hingga jumlah permintaan terpenuhi. Setiap pengambilan dari suatu *batch* dicatat sebagai transaksi barang keluar yang menautkan kolom barang_masuk_id ke *batch* asal, sehingga barang yang lebih lama tersimpan selalu dikeluarkan lebih dahulu dan risiko kerusakan atau kedaluwarsa dapat diminimalkan [4], [3].

```
// 1. Pengecekan stok sebelum eksekusi (mencegah partial commit)
$akumulasiKebutuhan = [];
foreach ($request->barang_id as $index => $bId) {
    $akumulasiKebutuhan[$bId] = ($akumulasiKebutuhan[$bId] ?? 0)
    + $request->jumlah[$index];
}
foreach ($akumulasiKebutuhan as $bId => $qtyRequested) {
    $totalTersedia = BarangMasuk::where('barang_id', $bId)
    ->sum('sisa_stok');
    if ($qtyRequested > $totalTersedia) {
        return back()->withErrors(['jumlah' =>
            "Stok tidak mencukupi."])->withInput();
    }
}

// 2. Eksekusi pengurangan stok per batch (FIFO)
foreach ($request->barang_id as $index => $bId) {
    $qtyNeeded = $request->jumlah[$index];
    $batches = BarangMasuk::where('barang_id', $bId)
    ->where('sisa_stok', '>', 0)
    ->orderBy('tanggal_masuk', 'asc')
    ->orderBy('id', 'asc')
    ->get();
    foreach ($batches as $batch) {
        if ($qtyNeeded <= 0) break;
        $diambil = min($qtyNeeded, $batch->sisa_stok);
        $batch->sisa_stok -= $diambil;
        $batch->save();
        BarangKeluar::create(array_merge($sharedData, [
            'barang_id' => $bId,
            'barang_masuk_id' => $batch->id,
            'jumlah' => $diambil,
        ]));
        $qtyNeeded -= $diambil;
    }
}
```



3.7 Hasil Pengujian Fungsional

Pengujian sistem dilakukan menggunakan metode *Black Box Testing* [23] terhadap sembilan fitur utama dengan skenario berhasil (*valid case*) dan skenario gagal (*invalid case*). Rangkuman hasil pengujian disajikan pada Tabel 3.

Tabel 3. Hasil Pengujian Black Box Testing

No	Fitur yang Diuji	Skenario Pengujian	Hasil Pengujian	Kesimpulan
1	Login	Memasukkan kredensial yang valid dan tidak valid	Sesuai harapan	Valid
2	Manajemen Data Barang	Menambah, mengubah, dan menghapus data barang	Sesuai harapan	Valid
3	Pencatatan Barang Masuk	Mengisi formulir dengan data lengkap dan tidak lengkap	Sesuai harapan	Valid
4	Distribusi Barang (FIFO)	Mendistribusikan barang di bawah dan melebihi stok tersedia	Sesuai harapan	Valid
5	Pengajuan Logistik Publik	Mengirim formulir dengan dan tanpa verifikasi Turnstile	Sesuai harapan	Valid
6	Verifikasi Pengajuan	Menyetujui dan menolak pengajuan yang masuk	Sesuai harapan	Valid
7	Ekspor Laporan	Mengekspor laporan stok dan transaksi ke PDF dan Excel	Sesuai harapan	Valid
8	Notifikasi Stok Menipis	Menurunkan stok barang hingga di bawah stok minimal	Sesuai harapan	Valid
9	Log Audit Trail	Melakukan perubahan data (create/update/delete)	Sesuai harapan	Valid

Berdasarkan Tabel 3, seluruh fitur utama memberikan keluaran yang sesuai dengan spesifikasi, baik pada skenario valid maupun invalid. Pengujian skenario gagal membuktikan mekanisme validasi berjalan sebagaimana mestinya: formulir barang masuk yang tidak lengkap ditolak disertai pesan validasi, distribusi yang melebihi total sisa stok ditolak sebelum transaksi dieksekusi sehingga tidak terjadi *partial commit*, dan pengajuan tanpa verifikasi Turnstile tidak dapat dikirim. Hasil ini perlu dimaknai secara proporsional. Tingkat kesesuaian yang tinggi menunjukkan bahwa fungsi-fungsi yang dispesifikasikan telah terimplementasi, bukan bahwa sistem bebas dari cacat, karena *Black Box Testing* hanya mengevaluasi kesesuaian keluaran terhadap kasus uji yang disusun berdasarkan spesifikasi yang sama. Kasus uji pada tahap ini dirancang oleh pengembang sehingga cakupannya terbatas pada jalur eksekusi yang telah diantisipasi dan berpotensi tidak menjangkau kondisi tak terduga di luar spesifikasi. Oleh sebab itu, kesimpulan kelayakan sistem tidak disandarkan pada pengujian fungsional semata, melainkan dilengkapi evaluasi pengguna pada Subbab 3.8 serta analisis kompleksitas dan skalabilitas pada Subbab 3.9.

3.8 Hasil Evaluasi Pengguna pada Iterasi Prototipe

Evaluasi pengguna dilaksanakan menyatu dengan siklus iterasi RAD sebagaimana dirangkum pada Tabel 2, dengan admin gudang sebagai pengguna utama yang menjalankan skenario operasional pada prototipe di akhir setiap iterasi. Berbeda dengan pengujian fungsional yang kasus ujinya disusun oleh pengembang berdasarkan spesifikasi, evaluasi pada tahap ini dikendalikan oleh pengetahuan operasional pengguna sehingga mampu menjangkau ketidaksesuaian yang tidak terwakili dalam spesifikasi awal. Iterasi pertama mengoreksi ruang lingkup kebutuhan dengan menegaskan perlunya format pencatatan terstandar beserta struktur *batch* penerimaan dan *audit trail*.

Iterasi kedua menyatakan rancangan antarmuka telah sesuai dengan alur kerja gudang sehingga konstruksi dapat dilanjutkan tanpa perubahan struktural. Iterasi ketiga menghasilkan temuan paling substantif, yaitu sepuluh ketidaksesuaian pada data master barang: delapan kesalahan satuan, meliputi mie instan dari dus menjadi bungkus, sarden menjadi kaleng, gula pasir, minyak goreng, dan tepung gandum menjadi kilogram, kecap dan saus menjadi botol, serta goodie bag menjadi lembar; dan dua komoditas yang ternyata tidak dikelola gudang BPBD, yaitu beras dan susu, yang kemudian dihapus melalui mekanisme *soft delete* agar riwayat transaksinya tetap terpelihara. Temuan tersebut penting secara metodologis karena seluruhnya lolos dari pengujian fungsional pada Subbab 3.7: satuan barang yang keliru tidak menimbulkan kegagalan fungsi apa pun, sebab sistem tetap menyimpan, menghitung, dan melaporkan angka secara benar menurut spesifikasi yang diberikan kepadanya. Ketidaksesuaian semacam ini hanya dapat terdeteksi oleh pengguna yang menguasai praktik gudang, sehingga hasil pada lapis ini menegaskan bahwa kelayakan sistem tidak dapat disandarkan pada tingkat kesesuaian fungsional semata.

3.9 Analisis Kompleksitas dan Skalabilitas Distribusi FIFO

Analisis pada subbab ini difokuskan pada proses distribusi barang karena proses tersebut menjalankan iterasi lintas *batch*, pemutakhiran kolom *sisa_stok*, dan penulisan baris transaksi barang keluar dalam satu rangkaian eksekusi, sehingga menjadi bagian sistem yang paling menentukan perilaku kinerja pada volume data yang besar.

Pengukuran empiris dilaksanakan menggunakan protokol yang diuraikan pada Subbab 2.5, pada lingkungan perangkat keras dan perangkat lunak yang sama dengan lingkungan pengembangan sebagaimana dirinci pada Subbab 3.3. Hasil pengukuran pada empat tingkat volume batch aktif disajikan pada Tabel 4.

**Tabel 4.** Hasil Pengukuran Waktu Respons Distribusi FIFO pada Empat Tingkat Volume Batch Aktif

Batch aktif	Operasi	Rata-rata (ms)	Median (ms)	Minimum (ms)	Maksimum (ms)	p95 (ms)	Simpangan baku (ms)
100	Seleksi	4,62	4,30	3,09	7,52	7,25	1,07
100	Alokasi	8,54	8,04	6,12	12,04	11,56	1,62
500	Seleksi	37,24	37,35	31,44	42,63	42,42	2,98
500	Alokasi	45,28	42,07	37,19	95,44	53,25	10,62
1.000	Seleksi	83,45	73,69	63,06	205,44	149,07	30,17
1.000	Alokasi	81,37	78,66	67,73	109,69	104,51	9,27
5.000	Seleksi	354,52	345,25	327,76	515,05	406,88	34,19
5.000	Alokasi	370,81	362,29	332,86	554,55	412,51	40,31

Perilaku kompleksitas proses distribusi FIFO ditinjau dari struktur algoritma yang diimplementasikan sekaligus dari hasil pengukuran pada Tabel 4. Secara analitis, algoritma penelusuran batch memiliki kompleksitas waktu $O(n)$ terhadap jumlah batch aktif, karena setiap batch dikunjungi paling banyak satu kali dan iterasi berhenti segera setelah jumlah permintaan terpenuhi. Namun pemeriksaan rencana eksekusi kueri menunjukkan bahwa penghentian dini tersebut hanya berlaku pada perulangan di lapis aplikasi, sedangkan basis data tetap mengembalikan seluruh baris batch aktif karena kueri pengambilan tidak membatasi jumlah baris yang diambil. Rencana eksekusi memperlihatkan akses melalui indeks pada kolom *barang_id* dengan tipe akses *ref*, sehingga pemindaian tabel penuh (*full table scan*) memang tidak terjadi; akan tetapi pengurutan menurut *tanggal_masuk* dilaksanakan melalui operasi *filesort* karena indeks yang tersedia hanya mencakup satu kolom.

Konsekuensinya, waktu respons ditentukan oleh jumlah batch aktif yang tersimpan, bukan semata oleh jumlah batch yang dikonsumsi. Pada seluruh pengukuran, jumlah batch yang dikonsumsi dikunci pada tiga batch, namun median waktu operasi seleksi meningkat dari 4,30 ms pada 100 batch aktif menjadi 345,25 ms pada 5.000 batch aktif, yaitu sekitar 80 kali lipat untuk volume data 50 kali lipat. Regresi linear terhadap nilai median menghasilkan hubungan $t \approx 1,45 + 0,0689n$ milidetik dengan koefisien determinasi $R^2 = 0,9996$, sedangkan model kuadratik hanya mencapai $R^2 = 0,9777$. Dengan demikian, pada rentang volume yang diuji, pertumbuhan waktu respons sebanding dengan jumlah batch aktif yang tersimpan di dalam tabel, bukan dengan jumlah batch yang benar-benar diambil.

Adapun beban operasi tulis terbukti kecil dan relatif tetap. Selisih median antara operasi alokasi dan operasi seleksi hanya 3,74 ms pada 100 batch, 4,72 ms pada 500 batch, 4,98 ms pada 1.000 batch, dan 17,05 ms pada 5.000 batch. Regresi terhadap operasi alokasi menghasilkan kemiringan 0,0717 ms per batch, hampir identik dengan kemiringan operasi seleksi sebesar 0,0689 ms per batch, sehingga selisih keduanya pada dasarnya berupa penambahan yang hampir konstan. Porsi biaya tulis terhadap waktu baca menurun dari 86,9% pada 100 batch aktif menjadi 4,9% pada 5.000 batch aktif. Artinya titik jenuh sistem bergeser: pada volume kecil beban didominasi operasi tulis, sedangkan pada volume besar beban didominasi pengambilan dan pengurutan data. Perlu dicatat bahwa setiap pengulangan diakhiri dengan pembatalan transaksi sehingga biaya penulisan permanen ke media penyimpanan tidak tercakup di dalam angka tersebut; biaya tulis pada kondisi operasional diperkirakan sedikit lebih tinggi daripada nilai yang dilaporkan.

Dari sisi keandalan waktu respons, persentil ke-95 pada 5.000 batch aktif mencapai 406,88 ms untuk operasi seleksi dan 412,51 ms untuk operasi alokasi, dengan nilai maksimum berturut-turut 515,05 ms dan 554,55 ms. Pengukuran diulang tiga kali pada mesin yang sama. Pada salah satu pengulangan, operasi alokasi tingkat 5.000 mencatat median 90,31 ms, jauh di bawah dua pengulangan lainnya. Asersi jumlah baris yang dijalankan di luar wilayah pengukuran memastikan bahwa ketiga pengulangan memproses 5.000 batch aktif dan mengonsumsi tepat tiga batch, sehingga perbedaan tersebut merupakan ragam pengukuran pada lingkungan pengembangan bersistem operasi tunggal dan bukan perbedaan beban kerja. Nilai yang dilaporkan pada Tabel 4 diambil dari pengulangan yang seluruh tingkat volumenya telah diverifikasi melalui asersi tersebut.

Pemeriksaan ketersediaan stok tetap dilakukan secara agregat sebelum eksekusi sehingga permintaan yang melampaui total sisa stok ditolak tanpa menimbulkan penulisan sebagian (*partial commit*). Berdasarkan temuan pengukuran ini, penambahan indeks komposit pada kolom *barang_id*, *sisa_stok*, dan *tanggal_masuk* berpotensi menghilangkan operasi *filesort* dan menekan laju pertumbuhan waktu respons; pengujian atas dugaan tersebut dirumuskan sebagai agenda penelitian lanjutan pada bagian kesimpulan.

3.10 Pembahasan

Hasil implementasi dan pengujian menunjukkan bahwa sistem telah memenuhi seluruh kebutuhan fungsional yang dirumuskan pada tahap analisis, dengan empat kontribusi utama. Pertama, algoritma FIFO bekerja otomatis pada level basis data dan logika aplikasi serta menautkan setiap transaksi keluar ke batch asalnya, sehingga kepatuhan terhadap kaidah rotasi stok tidak lagi bergantung pada kebijakan petugas; temuan ini sejalan dengan Titong [4] serta Ghani dkk. [5] bahwa FIFO meningkatkan akurasi data persediaan dan efisiensi pengelolaan barang. Kedua, pencatatan seluruh aktivitas melalui *audit trail* meningkatkan akuntabilitas dan keterlacakan data, menjawab permasalahan sistem berjalan yang sebelumnya tidak memiliki jejak perubahan. Ketiga, integrasi notifikasi WhatsApp dan perlindungan Cloudflare Turnstile pada formulir publik mempercepat respons terhadap pengajuan sekaligus menjaga sistem dari pengiriman



otomatis oleh bot. Keempat, ketersediaan ekspor laporan dalam format PDF dan Excel menjadikan laporan siap audit tanpa rekapitulasi manual sehari-hari. Dibandingkan dengan penelitian terdahulu sebagaimana diuraikan pada bagian pendahuluan, sistem yang dibangun memperluas hasil Nizarudin dan Widodo [10] serta Ardianti dkk. [11] pada konteks BPBD dengan menambahkan pengendalian rotasi stok berbasis FIFO tingkat batch, dan memindahkan penerapan FIFO dari sektor swasta [13], [14] ke konteks logistik kebencanaan pemerintah dengan tambahan kanal pengajuan publik tanpa *login*, *audit trail*, dan ekspor laporan siap audit. Berbeda dengan Mufti dkk. [12] yang menggunakan *Selection Sort* hanya untuk mengurutkan tampilan data, sistem ini menggunakan FIFO untuk benar-benar mengendalikan urutan pengeluaran barang berdasarkan tanggal penerimaan batch.

Di sisi lain, penelitian ini memiliki sejumlah keterbatasan. Dari sisi produk, sistem belum tersedia dalam versi aplikasi *mobile (native)*, pengiriman notifikasi WhatsApp masih bergantung pada layanan pihak ketiga, dan arsitektur belum mendukung pengelolaan lebih dari satu gudang (*multi-gudang*). Dari sisi evaluasi, kasus uji fungsional disusun oleh pengembang berdasarkan spesifikasi yang sama sehingga cakupannya terbatas pada jalur eksekusi yang telah diantisipasi; evaluasi pengguna berpusat pada admin gudang sebagai pengguna utama sehingga belum menjangkau seluruh peran pengguna sistem; serta pengukuran kinerja proses distribusi dilaksanakan pada satu mesin pengembangan dengan beban tunggal sehingga belum mencerminkan kondisi akses serentak saat tanggap darurat, dan ragam antarpengulangan yang teramati pada tingkat 5.000 batch aktif menunjukkan bahwa lingkungan pengukuran belum sepenuhnya terkendali. Selain itu, implementasi masih terbatas pada satu gudang di satu instansi sehingga generalisasi temuan perlu diuji pada BPBD lain. Setelah seluruh skenario pengujian dinyatakan berhasil, tahap *cutover* dilaksanakan melalui penyerahan akun administrator, pemindahan data persediaan awal ke dalam basis data sistem, serta pendampingan penggunaan kepada admin gudang, sehingga keempat tahapan RAD terpenuhi seluruhnya. Secara keseluruhan, sistem menjawab ketiga masalah yang diidentifikasi pada pendahuluan: pencatatan kini terstandarisasi melalui formulir tervalidasi, data tersimpan terpusat dengan *audit trail* otomatis, serta pembaruan data persediaan dan pengajuan berlangsung secara *real-time* pada satu sistem *web* yang sama.

Ditinjau dari proses pengembangan, penerapan RAD dengan *timebox* tetap terbukti mempersingkat waktu pembangunan sistem karena masukan pengguna langsung diproses pada siklus perancangan sehingga tahap konstruksi tidak lagi memerlukan perubahan struktural. Temuan ini konsisten dengan Haryanto dkk. [7] bahwa RAD efektif memangkas waktu pengembangan tanpa mengorbankan kualitas. Penjadwalan sesi evaluasi prototipe sebagai agenda tetap pada akhir setiap iterasi menjadi faktor penentu keberhasilan *timeboxing*, mengingat ketersediaan waktu petugas gudang terbatas oleh tugas operasional harian. Pelibatan admin gudang sejak tahap *user design* juga meningkatkan adopsi sistem karena pengguna telah memahami alur kerja sistem sebelum tahap *cutover* dilaksanakan.

4. KESIMPULAN

Penelitian ini menghasilkan sistem informasi manajemen logistik kebencanaan berbasis *web* pada BPBD Provinsi Sumatera Selatan yang menempatkan algoritma *First-In, First-Out* sebagai kendali rotasi persediaan pada tingkat batch, bukan sekadar sebagai urutan tampilan data. Melalui penautan setiap transaksi pengeluaran ke batch penerimaan asalnya, sistem mengubah kepatuhan terhadap kaidah rotasi stok dari praktik yang bergantung pada kebijakan petugas menjadi aturan yang ditegakkan oleh sistem dan terekam dalam jejak audit. Secara teoretis, penelitian ini memperluas penerapan FIFO yang selama ini dikaji pada persediaan sektor swasta ke dalam domain logistik kebencanaan pemerintah, yang memiliki karakteristik penerimaan sporadis dari banyak donor dan dokumen kedaluwarsa yang tidak selalu tersedia. Temuan penelitian menunjukkan bahwa pada kondisi tersebut tanggal penerimaan batch dapat berfungsi sebagai penanda pengganti (*proxy*) yang memadai bagi pengendalian rotasi, sekaligus menegaskan bahwa keterlaksanaan batch merupakan prasyarat, bukan pelengkap, bagi penerapan FIFO yang dapat diaudit. Penelitian ini juga menunjukkan bahwa *timeboxing* pada RAD dapat dijalankan di instansi publik dengan ketersediaan waktu pengguna yang terbatas, sepanjang evaluasi prototipe dijadwalkan sebagai sesi tetap pada akhir setiap iterasi. Secara manajerial, sistem memberikan tiga implikasi bagi BPBD Provinsi Sumatera Selatan. Pertama, penyusunan laporan persediaan untuk keperluan audit tidak lagi memerlukan rekapitulasi ulang dari berkas terpisah karena laporan PDF dan Excel dihasilkan langsung dari basis data transaksi. Kedua, jejak audit memungkinkan setiap selisih antara catatan sistem dan kondisi fisik gudang ditelusuri hingga ke pengguna dan waktu perubahannya, sehingga pemeriksaan internal tidak lagi bertumpu pada rekonstruksi dokumen. Ketiga, kanal pengajuan daring memindahkan permintaan bantuan dari percakapan pesan instan yang tidak terarsip menjadi entri terverifikasi yang berstatus dan dapat ditelusuri. Dengan demikian, kontribusi penelitian ini dapat dirangkum dalam tiga hal. Pertama, kontribusi praktis berupa sistem informasi manajemen logistik kebencanaan yang beroperasi di BPBD Provinsi Sumatera Selatan serta menegakkan rotasi FIFO tingkat batch beserta jejak audit dan pelaporan siap audit. Kedua, kontribusi teoretis berupa perluasan penerapan FIFO ke domain logistik kebencanaan pemerintah dengan tanggal penerimaan batch sebagai penanda pengganti tanggal kedaluwarsa, sekaligus penegasan keterlaksanaan batch sebagai prasyarat penerapan FIFO yang dapat diaudit. Ketiga, kontribusi metodologis berupa skema evaluasi tiga lapis dan bukti empiris bahwa waktu respons distribusi FIFO ditentukan oleh jumlah batch aktif yang tersimpan, bukan oleh jumlah batch yang dikonsumsi, yang menjadi dasar rekomendasi pengindeksan pada implementasi berskala besar. Keterbatasan penelitian ini mencakup kasus uji fungsional yang disusun oleh pengembang, evaluasi pengguna yang berpusat pada admin gudang, pengukuran kinerja distribusi yang dilaksanakan pada satu mesin pengembangan dengan beban tunggal, serta lingkup implementasi pada satu gudang di satu instansi. Berdasarkan



keterbatasan tersebut, penelitian lanjutan disarankan untuk: (1) menambahkan indeks komposit pada kolom *barang_id*, *siswa_stok*, dan *tanggal_masuk* tabel *barang_masuks*, kemudian mengukur ulang waktu respons dengan protokol yang sama guna menguji apakah penghapusan operasi *filesort* menekan laju pertumbuhan waktu respons terhadap jumlah batch aktif; (2) melaksanakan pengujian beban serentak dengan sekurang-kurangnya 50 pengguna simultan guna mengukur degradasi waktu respons pada kondisi tanggap darurat; (3) menambahkan pengujian *white box* dengan target cakupan pernyataan (*statement coverage*) minimal 80% pada modul distribusi FIFO; (4) memperluas arsitektur ke skema multi-gudang dengan pengendalian rotasi lintas lokasi dan mengujinya pada sekurang-kurangnya tiga kabupaten/kota; serta (5) mengembangkan modul peramalan kebutuhan logistik berbasis data historis distribusi dan mengevaluasinya menggunakan metrik kesalahan peramalan seperti *Mean Absolute Percentage Error* (MAPE).

REFERENCES

- [1] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 16th ed. New York, NY, USA: Pearson, 2020.
- [2] S. Sasmito Muslim, N. A. Wibowo, and F. Nofandi, "Analisis Penerapan Sistem Informasi Manajemen Pada Kegiatan Logistik Di Indonesia," *Dinamika Bahari*, vol. 2, no. 1, pp. 6–12, 2021, doi: 10.46484/db.v2i1.262.
- [3] T. T. Agustin, "Penerapan Metode FIFO (First In First Out) Dalam Pengendalian Persediaan Barang," *Jurnal Bisnis, Logistik dan Supply Chain (BLOGCHAIN)*, vol. 2, no. 2, pp. 92–102, 2022, doi: 10.55122/blogchain.v2i2.536.
- [4] F. S. Titong, "Penerapan Metode FIFO (First in First Out) dalam Menjaga Efektivitas Warehouse pada PT. Mutiara Luwuk Bintang Lestari," *Jurnal Syntax Admiration*, vol. 5, no. 10, pp. 3762–3768, 2024, doi: 10.46799/jsa.v5i10.1500.
- [5] A. Ghani, J. Veri, and Mardison, "Perancangan Sistem Informasi Inventory Menggunakan Metode FIFO pada Toko Grosir Ceria," *Jurnal KomtekInfo*, vol. 11, no. 1, pp. 18–24, 2024, doi: 10.35134/komtekinfo.v11i1.498.
- [6] J. Martin, *Rapid Application Development*, 3rd ed. New York, NY, USA: Macmillan, 1991.
- [7] T. Haryanto, T. Tanuwijaya, E. Y. Koentjoro, and R. Maharddhika, "Implementasi Metode Rapid Application Development Dalam Pengembangan Sistem Informasi Integrated-Sales," *Jurnal Ekonomi Manajemen Sistem Informasi*, vol. 6, no. 3, pp. 1602–1618, 2025, doi: 10.38035/jemsi.v6i3.4163.
- [8] N. Hidayat and K. Hati, "Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE)," *Jurnal Sistem Informasi STMIK Antar Bangsa*, vol. 10, no. 2, pp. 8–17, 2021, doi: 10.51998/jsi.v10i1.352.
- [9] M. Tabrani et al., "Penerapan Metode Rapid Application Development (RAD) pada Rancangan Sistem Informasi Pengelolaan Data Penduduk Desa Dawuan Kabupaten Karawang," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 13, no. 3, 2024, doi: 10.30591/smartcomp.v13i3.6184.
- [10] M. I. Nizarudin and A. Widodo, "Digitalisasi Sistem Logistik Bencana untuk Meningkatkan Layanan BPBD Kabupaten Jepara," *ABDIFORMATIKA: Jurnal Pengabdian Masyarakat Informatika*, vol. 5, no. 2, pp. 101–109, 2025, doi: 10.59395/abdiformatika.v5i2.283.
- [11] F. A. E. Ardianti, R. Setiawan, and M. Arifin, "Sistem Informasi Manajemen Logistik Dan Peralatan Kebencanaan Multi Posko Web-Based," *DECODE: Jurnal Pendidikan Teknologi Informasi*, vol. 5, no. 2, pp. 612–624, 2025, doi: 10.51454/decode.v5i2.1233.
- [12] A. Mufti, P. L. L. Belluano, and M. A. Mude, "Sistem Informasi Bantuan Logistik Korban Bencana Alam," *Buletin Sistem Informasi dan Teknologi Islam*, vol. 1, no. 2, pp. 92–99, 2020, doi: 10.33096/busiti.v1i2.830.
- [13] H. Lubis, A. Fitriyani, and M. H. Prayitno, "Sistem Informasi Persediaan Barang Jadi Menggunakan Metode FIFO Pada PT Rubberman Indonesia," *JSI (Jurnal Sistem Informasi) Universitas Suryadarma*, vol. 10, no. 2, pp. 1–9, 2023, doi: 10.35968/jsi.v10i2.1071.
- [14] H. A. Pangastutik, A. A. Aqham, and H. I. Huda, "Sistem Informasi Laporan Keuangan Persediaan Barang Menggunakan Metode FIFO Berbasis Web Pada PT. Usaha Maju," *Jurnal Mahasiswa Kreatif*, vol. 1, no. 5, pp. 119–130, 2023, doi: 10.59581/jmk-widyakarya.v1i5.1161.
- [15] K. Nistrina and L. Sahidah, "Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Penerimaan Siswa Baru di SMK Marga Insan Kamil," *J-SIKA: Jurnal Sistem Informasi Karya Anak Bangsa*, vol. 4, no. 1, pp. 17–23, 2022. [Online]. Available: <https://ejournal.unibba.ac.id/index.php/j-sika/article/view/839>
- [16] E. O. W. Susanti and I. Ummami, "Rancang Bangun Sistem Informasi Jurnal Perkuliahan Berbasis Web Guna Meningkatkan Efektivitas Pembelajaran," *Jurnal Teknologi dan Sistem Informasi Bisnis (JTEKSIS)*, vol. 4, no. 2, p. 386, 2022, doi: 10.47233/jteksis.v4i2.556.
- [17] N. Musthofa and M. A. Adiguna, "Perancangan Aplikasi E-Commerce Spare-Part Komputer Berbasis Web Menggunakan CodeIgniter Pada Dhamar Putra Computer Kota Tangerang," *OKTAL: Jurnal Ilmu Komputer dan Sains*, vol. 1, no. 3, pp. 199–207, 2022. [Online]. Available: <https://journal.mediapublikasi.id/index.php/oktal/article/view/37>
- [18] M. N. Arkan and U. A. S. Anandri, "Perancangan Entity-Relationship Diagram (ERD) Menggunakan SQL Server Database pada Platform Tautan Halaman Web Lynk.id," *TEKNOFILE: Jurnal Sistem Informasi*, vol. 3, no. 4, 2025. [Online]. Available: <https://jurnal.nawansa.com/index.php/teknofile/article/view/411>
- [19] M. I. Alifudin, "Analisis dan Perancangan Sistem Informasi Manajemen Tugas Multi-Peran Berbasis Web," *Jurnal Publikasi Ilmu Komputer dan Multimedia*, vol. 4, no. 3, pp. 74–85, 2025, doi: 10.55606/jupikom.v4i3.5228.
- [20] F. Rifandi, T. V. Adriansyah, and R. Kurniawati, "Website Gallery Development Using Tailwind CSS Framework," *Jurnal E-Komtek*, vol. 6, no. 2, pp. 205–214, 2022, doi: 10.37339/e-komtek.v6i2.937.
- [21] Cloudflare, "Cloudflare Turnstile: A CAPTCHA-free, privacy-preserving alternative," 2023. [Online]. Available: <https://developers.cloudflare.com/turnstile/>
- [22] A. Nagari et al., *Manajemen Logistik dan Rantai Pasokan*. Jakarta, Indonesia: Sada Kurnia Pustaka, 2024.
- [23] B. Hardika et al., "Pengujian Blackbox Testing Website Garuda Farm Menggunakan Teknik Equivalence Partitioning," *Jurnal Kridatama Sains dan Teknologi*, vol. 6, no. 2, pp. 740–753, 2024, doi: 10.53863/kst.v6i02.1420.