

Perancangan dan Implementasi Sistem *Inventory* Berbasis *Desktop* Menggunakan Model *Waterfall* pada CV. Permata Printing

Abdul Hamdani Eko Saputro¹, Bagas Mishbahuddin¹, Bayu Santoso¹,
Maulana Ardhiansyah^{1*}

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46,
Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ¹abdulhamdaniekosaputro@gmail.com, ²mishbahuddin349@email.com,

³bayusomething02@gmail.com, ^{4*}dosen00374@unpam.ac.id

(* : coressponding author)

Abstrak— Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem informasi inventori berbasis desktop pada CV. Permata Printing guna mengatasi permasalahan pengelolaan stok barang yang belum terkomputerisasi. Saat ini, proses pendataan barang di CV. Permata Printing masih dilakukan secara manual, yang mengakibatkan tingginya risiko kehilangan data, ketidaksesuaian jumlah fisik barang dengan catatan, serta kesulitan dalam memantau ketersediaan stok secara *real-time* saat terjadi lonjakan pesanan. Pengembangan sistem ini menggunakan model *Waterfall* yang mencakup tahapan analisis, desain, pengkodean, pengujian, dan pemeliharaan, dengan mengimplementasikan bahasa pemrograman Java. Hasil dari penelitian ini adalah sebuah aplikasi inventori desktop yang mampu mengintegrasikan data barang secara sistematis. Implementasi sistem ini terbukti meningkatkan akurasi pendataan stok barang, meminimalisir kesalahan perhitungan, serta memudahkan pegawai dalam melakukan pemantauan inventori secara lebih cepat dan efisien dibandingkan metode konvensional sebelumnya.

Kata Kunci: *Sistem Inventori, Desktop, Model Waterfall, Java, CV. Permata Printing*

Abstract—This research aims to design and implement a desktop-based inventory information system at CV. Permata Printing to address the problem of non-computerized inventory management. Currently, the inventory data collection process at CV. Permata Printing is still done manually, resulting in a high risk of data loss, discrepancies between the physical quantity of goods and records, and difficulties in monitoring stock availability in real-time when there is a surge in orders. The development of this system uses the *Waterfall* model which includes the stages of analysis, design, coding, testing, and maintenance, by implementing the Java programming language. The result of this research is a desktop inventory application that is able to integrate inventory data systematically. The implementation of this system has been proven to increase the accuracy of inventory data collection, minimize calculation errors, and make it easier for employees to monitor inventory more quickly and efficiently compared to previous conventional methods.

Keywords: *Inventory System, Desktop, Waterfall Model, Java, CV. Permata Printing*

1. PENDAHULUAN

Perkembangan teknologi informasi saat ini telah memberikan pengaruh signifikan terhadap efektivitas operasional dalam sebuah perusahaan. Salah satu aspek krusial dalam menunjang keberlangsungan usaha adalah manajemen inventori yang efektif (Sundara et al., 2023). CV. Permata Printing merupakan perusahaan yang bergerak di bidang jasa percetakan dengan tingkat volume transaksi barang yang cukup dinamis setiap harinya.

Namun, sistem pendataan dan manajemen inventori (stok barang) di CV. Permata Printing hingga saat ini belum terkomputerisasi atau belum memiliki sistem khusus. Kondisi ketiadaan sistem inventori ini memicu berbagai kendala administratif yang menghambat alur kerja, di antaranya risiko hilangnya data pencatatan, terjadinya selisih perhitungan antara data administratif dengan fisik barang di gudang, serta kesulitan pegawai dalam memantau ketersediaan stok secara *real-time*, terutama saat frekuensi pesanan pelanggan sedang tinggi (Nurfi, 2020).

Mengingat pentingnya akurasi data dalam operasional perusahaan, maka diperlukan sebuah sistem informasi yang mampu mengelola data stok barang secara terintegrasi (Prasetyo, 2025). Oleh karena itu, penelitian ini mengambil judul "Perancangan dan Implementasi Sistem *Inventory* Berbasis *Desktop* Menggunakan Model *Waterfall* pada CV. Permata Printing". Pemilihan model

pengembangan *Waterfall* didasarkan pada pendekatannya yang sistematis dan terstruktur, sementara penggunaan bahasa pemrograman Java dipilih untuk membuat aplikasi desktop yang stabil, responsif, serta terkomputerisasi yang terhubung dengan data persediaan barang dengan database yang update untuk mempermudah pengguna dalam memantau jumlah persediaan barang (Adam et al., 2023).

Pemilihan model pengembangan perangkat lunak *Waterfall* didasarkan pada pendekatannya yang sistematis dan terstruktur dalam mendokumentasikan setiap tahapan pengembangan, mulai dari analisis hingga pemeliharaan (Hidayat & Ardhiansyah, 2022). Pengembangan sistem ini diimplementasikan menggunakan bahasa pemrograman Java dengan arsitektur antarmuka modern (Nauval Abrari et al., 2025), serta dipadukan dengan implementasi manajemen basis data. Hal ini diharapkan dapat memberikan solusi operasional yang komprehensif, stabil, terukur, dan responsif bagi permasalahan inventori di lingkungan CV. Permata Printing.

2. METODOLOGI PENELITIAN

2.1 Metode Pengembangan Perangkat Lunak

Penelitian ini menggunakan model *Waterfall* sebagai metodologi pengembangan sistem. Model ini dipilih karena alurnya yang sistematis dan terstruktur, yang sangat sesuai untuk pengembangan aplikasi inventori berbasis desktop agar setiap tahapannya dapat terdokumentasi dengan baik. Menurut (Manurung & Rizkina, 2021; Setiawan & Imam Suharjo, 2025) pengembangan menggunakan model *waterfall*, menunjukkan sistem mampu memberikan kemudahan dalam distribusi informasi secara cepat dan akurat dibanding inventarisasi manual mempunyai keunggulan-keunggulan berdasarkan kegunaan, kemudahan mempelajari, keefisiensi sistem dan keakuratan data. Tahapan *Waterfall* yang dilakukan meliputi:

1. Analisis Kebutuhan

Pada tahap ini, dilakukan pengumpulan data mengenai proses bisnis inventory di CV. Permata Printing serta mengidentifikasi masalah utama terkait sistem manual yang sedang berjalan. Analisis ini mencakup wawancara dengan stakeholder, observasi proses bisnis, dan dokumentasi requirement spesifik dari pengguna akhir. Output dari tahap ini adalah spesifikasi kebutuhan yang tertulis dan telah disetujui oleh pengguna.

2. Perancangan Sistem

Tahap ini mencakup perancangan *database* serta antarmuka (interface) aplikasi menggunakan bahasa pemrograman Java agar sistem dapat beroperasi sesuai kebutuhan pengguna. Perancangan dilakukan menggunakan tools modeling seperti Entity Relationship Diagram (ERD) untuk struktur database dan User Interface Mockup untuk tampilan aplikasi. Desain yang dibuat harus memenuhi requirement yang telah ditentukan sebelumnya dan mempertimbangkan aspek usability serta scalability.

3. Pengodean (*Coding*)

Implementasi desain ke dalam bentuk kode program menggunakan bahasa Java untuk membangun fitur-fitur utama inventori. Pada tahap ini, developer mengikuti coding standard yang telah ditetapkan untuk memastikan konsistensi kode dan memudahkan maintenance di masa depan. Penggunaan design pattern dan best practice dalam Java programming diterapkan untuk menghasilkan kode yang berkualitas tinggi.

4. Pengujian (*Testing*)

Melakukan pengujian sistem untuk memastikan seluruh fitur pendataan stok barang berjalan dengan akurat dan meminimalisir kesalahan input. Pengujian mencakup functional testing (apakah fitur bekerja sesuai spesifikasi) dan non-functional testing (performa, security, usability). Setiap bug yang ditemukan didokumentasikan dan diperbaiki sebelum release ke pengguna.

5. Pemeliharaan (*Maintenance*)

Tahap akhir di mana sistem yang telah diimplementasikan dipantau untuk memastikan kinerjanya tetap stabil dalam membantu aktivitas inventori perusahaan. Pemeliharaan mencakup bug fixing, performance monitoring, dan user support untuk memastikan sistem terus berjalan optimal.

2.2 Alat dan Bahan Penelitian

Dalam melaksanakan penelitian dan pengembangan sistem inventory berbasis desktop ini, diperlukan beberapa perangkat dan bahan pendukung yang memadai untuk memastikan pengembangan sistem berjalan dengan optimal:

1. Perangkat Keras (*Hardware*)

Perangkat komputer dengan prosessor minimal Intel Core i3 atau i5 untuk memastikan processing power yang cukup, RAM 6 GB atau lebih untuk memastikan performa tetap optimal saat menjalankan aplikasi

2. Perangkat Lunak (*Software*)

Bahasa pemrograman Java sebagai bahasa utama pengembangan sistem serta *tools* pendukung untuk pengelolaan basis data.

2.3 Teknik Pengumpulan Data

1. Observasi

Melakukan pengamatan langsung terhadap proses manajemen stok barang yang ada saat ini di CV. Permata Printing. Observasi difokuskan pada identifikasi operasional, error dalam pencatatan manual, hambatan yang dialami pegawai, dan kesulitan dalam monitoring stok secara real-time.

2. Studi Pustaka

Mencari referensi dan teori pendukung mengenai sistem inventori dan bahasa pemrograman Java untuk memperkuat dasar perancangan sistem

3. Wawancara Semi-Terstruktur (*Semi-Structured Interview*)

Melakukan sesi wawancara dengan stakeholder kunci di CV. Permata Printing termasuk pimpinan perusahaan, staff gudang, dan bagian administrasi. Wawancara dirancang untuk menggali lebih dalam tentang requirement sistem, ekspektasi pengguna, prioritas fitur, dan hambatan yang dirasakan dalam sistem manual. Hasil wawancara didokumentasikan dan dianalisis untuk mengidentifikasi common themes dan requirement yang konsisten dari berbagai responden.

3. ANALISA DAN PEMBAHASAN

3.1 Analisis Sistem yang Sedang Berjalan

Berdasarkan hasil observasi pada CV. Permata Printing, sistem pendataan barang saat ini masih dilakukan secara manual menggunakan catatan fisik atau buku besar. Analisis terhadap sistem manual tersebut menunjukkan beberapa kelemahan utama:

1. Resiko Data

Pencatatan manual rentan terhadap kerusakan, kehilangan, data, atau kesalahan penulisan oleh staff. Ketiadaan backup data sistematis mengakibatkan tingginya resiko kehilangan informasi apabila terjadi kerusakan pada dokumen fisik atau buku catatan. Selain itu, tidak ada mekanisme kontrol akses sehingga siapa saja dapat mengubah catatan tanpa audit trail (Jayema et al., 2024).

2. Ketidaksinkronan Data

Sering terjadi selisih antara jumlah stok di buku catatan dengan jumlah fisik barang yang tersedia di gudang. Penyebabnya seperti kesalahan pencatatan saat proses barang masuk atau keluar, tidak ada rekonsiliasi data secara berkala dan sistematis untuk memverifikasi akurasi, ketidaksesuaian timing antara waktu fisik barang keluar gudang dengan waktu pencatatan di dokumen, dan adanya kerusakan atau kehilangan barang selama penyimpanan yang tidak tercatat dengan baik

3. Informasi Tidak *Real-time*

Pegawai kesulitan mengetahui ketersediaan stok secara instan, terutama saat volume pesanan di CV. Permata Printing sedang meningkat. Pengecekan stok memerlukan waktu yang cukup lama karena harus datang ke lokasi gudang atau membuka dan membaca buku catatan secara manual,

yang dapat menghambat proses pengambilan keputusan cepat dan memperlambat response time terhadap customer inquiry. Informasi inventory yang tidak readily available juga mengurangi responsivitas perusahaan dalam menanggapi permintaan pelanggan.

3.2 Perancangan Sistem Inventori Berbasis *Desktop*

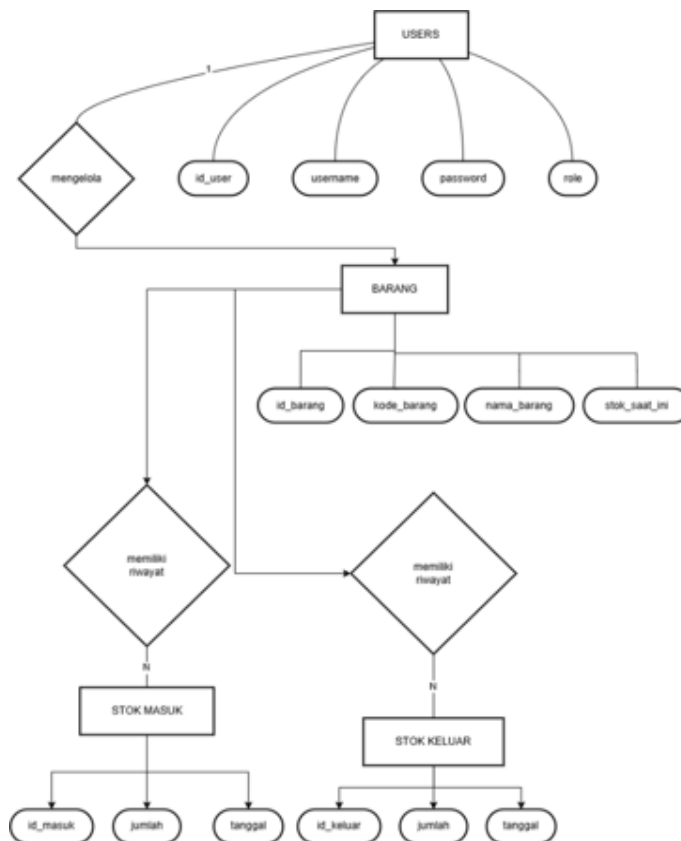
Untuk mengatasi masalah tersebut, sistem yang dibangun dirancang dengan alur kerja yang terintegrasi menggunakan bahasa pemrograman Java dengan fokus pada aksesibilitas pengguna, kemudahan operasional, dan data integrity yang kuat. Perancangan dilakukan dengan menitikberatkan pada kemudahan operasional bagi pengguna :

1. Arsitektur Sistem Berlapis (*Layered Architecture*)

Sistem dirancang dengan arsitektur layered yang memisahkan tanggung jawab antara presentation layer (user interface yang user-facing), business logic layer (proses bisnis dan validasi data), dan data access layer (interaksi dengan database backend). Pemisahan concern ini tidak hanya memudahkan maintenance dan testing dari setiap komponen secara independent, tetapi juga memungkinkan future development dan modifikasi fitur tanpa mengganggu layer lainnya. Arsitektur ini juga meningkatkan code reusability dan testability.

2. Struktur Data

Pengelolaan basis data yang terpusat untuk menghindari duplikasi data barang.



Gambar 1. *Entity Relationship Diagram (ERD)*

3. Antarmuka Pengguna

Desain aplikasi desktop menggunakan Java Swing dibuat sederhana, clean, dan intuitif agar pegawai dapat melakukan input barang masuk, keluar, dan pelaporan tanpa memerlukan pelatihan teknis yang rumit. Penggunaan komponen visual yang familiar (buttons, textfields, combobox, tables) dan konsisten di seluruh aplikasi membantu user experience yang optimal. Navigation yang logical dan straightforward memandu user melalui workflow operasional dengan efisien.

4. Implementasi Security Features yang Komprehensif

Sistem dilengkapi dengan fitur authentication robust berbasis username-password dengan enkripsi untuk memastikan hanya user yang authorized yang dapat mengakses sistem. Password disimpan dengan hashing algorithm (seperti bcrypt) bukan plain text untuk menjaga keamanan data sensitif. Implementasi role-based access control memastikan bahwa setiap user hanya dapat mengakses fitur dan data yang relevan dengan tanggung jawab mereka dalam organisasi.

5. Validasi Input dan Error Handling yang Robust

Sistem dilengkapi dengan validasi input yang comprehensive pada setiap form untuk memastikan data yang diinput sudah sesuai format, tipe data, dan business rule yang berlaku sebelum disimpan ke database. Error handling yang baik memberikan feedback yang jelas kepada user tentang kesalahan input, baik berupa format error, validation error, atau constraint violation. Feedback yang informatif memudahkan user memahami dan memperbaiki input mereka dengan cepat.

3.3 Pembahasan Implementasi Sistem

Setelah melakukan pengkodean (coding menggunakan Java dengan IDE NetBeans, database implementation dengan MySQL, dan pengujian fungsional menyeluruh, sistem yang diimplementasikan memberikan hasil yang memuaskan dalam memenuhi objective penelitian. Hasil implementasi sistem menunjukkan peningkatan signifikan dalam berbagai aspek operasional:

1. Akurasi Data

Proses input data kini dilakukan melalui sistem, sehingga setiap perubahan stok (masuk/keluar) tercatat secara otomatis dan minim kesalahan perhitungan. Implementasi database transaction dengan properties memastikan data consistency dan mencegah data corruption bahkan dalam skenario concurrent access. Perhitungan stok secara otomatis mengeliminasi kesalahan kalkulasi manual yang sebelumnya sering terjadi, meningkatkan tingkat akurasi data menjadi mendekati 100%.

2. Efisiensi Waktu

Pegawai dapat langsung melihat jumlah stok terkini (*real-time*) melalui layar monitor, tanpa harus melakukan penghitungan fisik ke gudang setiap kali ada pertanyaan mengenai ketersediaan barang. Fitur pencarian dan filter data yang powerful memungkinkan user menemukan informasi inventory spesifik dengan waktu respons yang minimal. Peningkatan efisiensi ini membebaskan staf dari tugas-tugas administratif repetitif dan memungkinkan mereka fokus pada tugas-tugas yang lebih strategis dan value-adding seperti procurement planning dan supplier management.

3. Kemudahan Pelaporan

Sistem ini mempermudah pembuatan laporan stok barang secara berkala, yang sebelumnya memerlukan waktu lama karena harus menyalin catatan dari buku fisik. Laporan dapat di-generate secara otomatis dengan berbagai parameter filter (berdasarkan kategori produk, date range, jumlah stok, dll) untuk mendukung business decision making yang informed. Kemampuan reporting yang superior ini memberikan visibility lebih baik terhadap trend inventory dan membantu management dalam perencanaan procurement dan penjualan.

4. Audit Trail Lengkap

Setiap transaksi inventory tercatat dengan detail lengkap termasuk timestamp kapan transaksi dilakukan dan detail perubahan stok. Log transaksi ini menciptakan transparency penuh tentang siapa yang melakukan apa, kapan, dan perubahan apa yang dilakukan pada inventory. Hal ini meningkatkan accountability di antara staff dan memudahkan management untuk mengidentifikasi source dari kesalahan atau discrepancy apabila terjadi. Audit trail ini juga penting untuk compliance dan internal control purposes.

3.4 Uji Coba Sistem

Pengujian dilakukan untuk memastikan seluruh modul berfungsi dengan baik. Hasil uji coba menunjukkan bahwa:

1. Fitur *input* barang berfungsi sesuai dengan parameter yang ditentukan (Nama barang, jumlah, dan tanggal).
2. Sistem mampu menampilkan informasi stok secara akurat saat dilakukan pencarian atau pengecekan data.
3. Secara keseluruhan, aplikasi berhasil memenuhi tujuan penelitian yaitu meningkatkan akurasi pendataan stok barang pada CV. Permata Printing.

Tabel 1. Uji Coba Sistem

No	Fitur / Fungsi yang Diuji	Skenario Masukan (<i>Input</i>)	Hasil yang Diharapkan (<i>Output</i>)	Status
1	Otentikasi Hak Akses (Form <i>Login</i>)	Menginput kombinasi <i>username</i> atau <i>password</i> yang tidak terdaftar di MySQL.	Sistem menolak akses, memunculkan pesan dialog " <i>Username/Password Salah</i> ".	Sukses
2	Keamanan Enkripsi Akses (Form <i>Login</i>)	Menginput kredensial dengan kombinasi data yang benar dan terdaftar pada sistem.	Sistem meloloskan otentikasi, lalu mengarahkan antarmuka ke halaman <i>Dashboard</i> utama.	Sukses
3	Validasi Duplikasi Data (Kelola Barang)	Menginput data barang baru menggunakan kode barang yang sudah tersimpan sebelumnya.	Sistem mendeteksi duplikasi <i>Primary Key</i> , menampilkan peringatan, dan membatalkan simpan.	Sukses
4	Logika Penambahan Stok (Barang Masuk)	Menginput transaksi pasokan baru dengan data kolom yang terisi lengkap dan valid.	Sistem menyimpan riwayat ke tabel transaksi dan memperbarui penambahan stok barang.	Sukses
5	Logika Pembatasan Stok (Barang Keluar)	Menginput jumlah penarikan barang dengan kuantitas yang melebihi batas stok tersedia.	Sistem memunculkan pesan peringatan " <i>Stok Tidak Mencukupi</i> ", menolak pengurangan data.	Sukses
6	Logika Pengurangan Stok (Barang Keluar)	Menginput pengambilan barang dengan kuantitas di bawah/sama dengan nilai sisa stok.	Sistem menyimpan data transaksi dan memotong kuantitas stok di <i>database</i> secara akurat.	Sukses

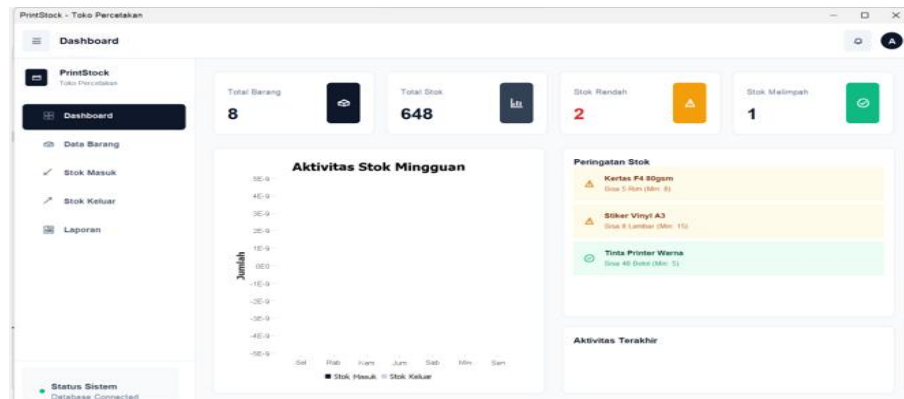
4. IMPLEMENTASI

4.1 Tampilan Antarmuka (*Interface*)

Aplikasi inventori ini dirancang dengan antarmuka berbasis desktop menggunakan Java Swing, yang memungkinkan pengguna untuk berinteraksi dengan sistem secara intuitif.

1. Menu Utama:

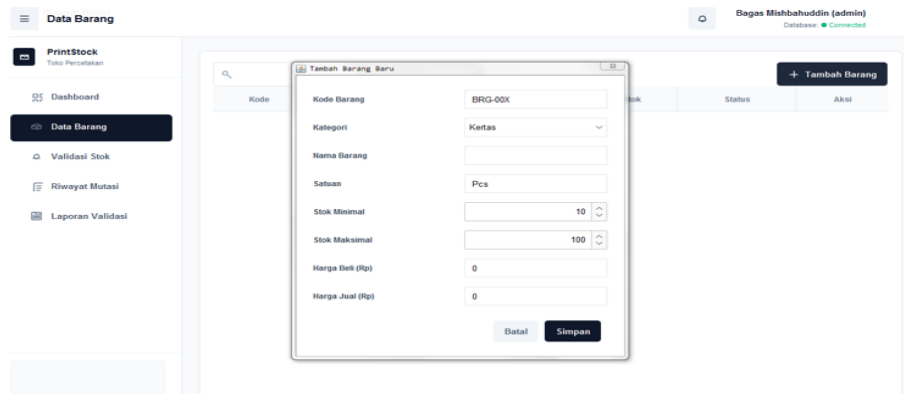
Berisi navigasi untuk mengakses fitur data barang, input stok masuk, input stok keluar, dan laporan.



Gambar 2. Halaman Utama (*Dashboard*)

2. Formulir Input Data

Memungkinkan pengguna untuk memasukkan detail barang secara akurat untuk meminimalisir kesalahan pengetikan.



The form includes the following fields:

- Kode Barang: BRG-00X
- Kategori: Kertas
- Nama Barang:
- Satuan: Pcs
- Stok Minimal: 10
- Stok Maksimal: 100
- Harga Beli (Rp): 0
- Harga Jual (Rp): 0

Buttons: Batal, Simpan

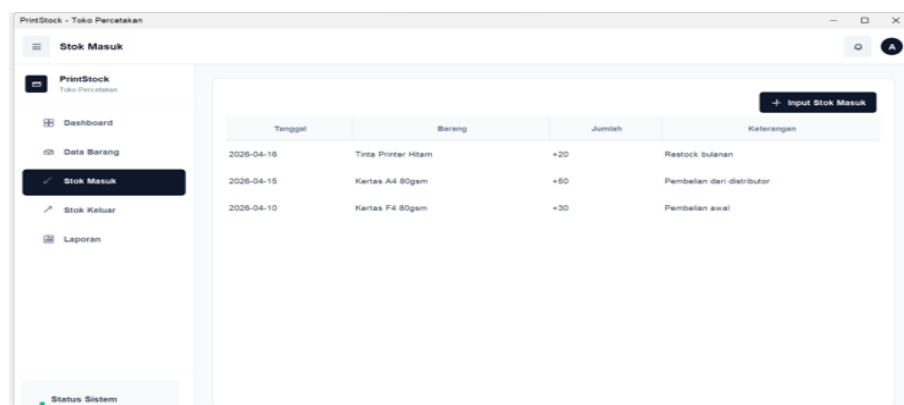
Gambar 3. Form Input Data Barang

4.2 Fitur Fungsionalitas Sistem

Berikut adalah fungsi utama yang diimplementasikan ke dalam aplikasi:

1. Manajemen Data Barang

Sistem memungkinkan penambahan, perubahan, dan penghapusan data barang secara terpusat.

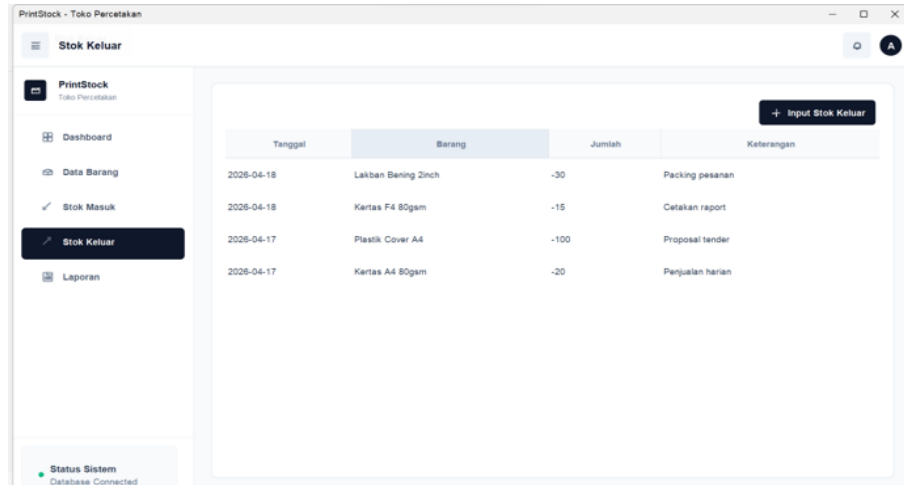


Tanggal	Barang	Jumlah	Keterangan
2025-04-16	Tinta Printer Hitam	+20	Restock bulanan
2025-04-15	Kertas A4 80gsm	+50	Pembelian dari distributor
2025-04-10	Kertas F4 80gsm	+30	Pembelian awal

Gambar 4. Halaman Stok Masuk

2. Pencatatan Transaksi

Pengguna dapat mencatat barang masuk dan barang keluar secara *real-time*, sehingga jumlah stok yang tersimpan di sistem akan otomatis terupdate.

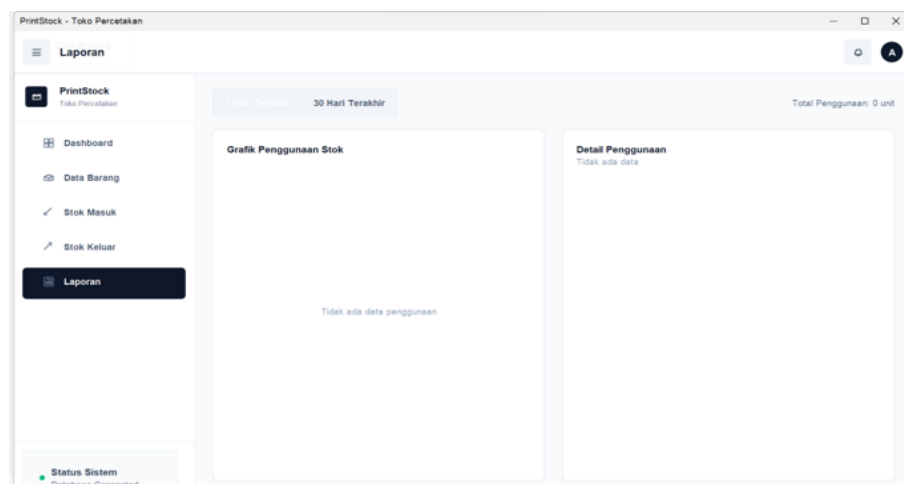


Tanggal	Barang	Jumlah	Keterangan
2026-04-18	Lakban Bening 2inch	-30	Packing pesanan
2026-04-18	Kertas F4 80gm	-15	Cetakan raport
2026-04-17	Plastik Cover A4	-100	Proposal tender
2026-04-17	Kertas A4 80gm	-20	Penjualan harian

Gambar 5. Halaman Stok Keluar

3. Sistem Pelaporan

Aplikasi mampu menghasilkan laporan stok barang yang tersusun rapi, menggantikan sistem pencatatan manual sebelumnya.



Gambar 6. Halaman Laporan

4.3 Proses Pengoperasian

Untuk menjalankan sistem ini, pengguna perlu mengikuti langkah-langkah berikut:

1. Pengguna melakukan *login* ke aplikasi untuk keamanan data.
2. Pada modul barang, pengguna mengisi informasi produk yang tersedia di gudang CV. Permata Printing.
3. Setiap kali ada aktivitas transaksi, pengguna menginput data ke modul transaksi agar data stok selalu sinkron dengan kondisi fisik

4.4 Uji Coba Fungsi (*Testing*)

Berdasarkan hasil pengujian dari Tabel 1, dapat disimpulkan bahwa semua fitur kritis sistem telah berfungsi sesuai dengan expected output dan requirement specification. Sistem mampu:

1. melakukan autentikasi user dengan aman dan mencegah unauthorized access.
2. mencegah duplikasi data melalui constraint validasi di level database dan application.
3. melakukan perhitungan stok secara otomatis dan akurat untuk setiap transaksi.
4. menerapkan business logic yang tepat untuk pembatasan stok keluar mencegah overselling.
5. maintaining data integrity di dalam database melalui transaction control dan referential integrity constraint.

Pengujian non-fungsional juga dilakukan untuk memastikan performance aplikasi dalam menangani volume data yang besar, multiple concurrent user access di masa depan, dan aspek security. Hasil pengujian performance menunjukkan bahwa aplikasi dapat menangani operasi CRUD (Create, Read, Update, Delete), yang merupakan performance yang sangat acceptable untuk aplikasi desktop berbasis desktop lokal. Pengujian security menunjukkan bahwa sistem dapat mencegah common attack vector seperti SQL injection melalui prepared statement dan input validation yang proper.

5. KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem inventori pada CV. Permata Printing, dapat disimpulkan beberapa hal sebagai berikut:

1. Efektivitas Sistem

Sistem inventori berbasis desktop yang dikembangkan dengan bahasa pemrograman Java telah berhasil menggantikan sistem manual, sehingga proses pendataan barang menjadi lebih terorganisir dan minim kesalahan manusia (*human error*).

2. Akurasi Data

Implementasi sistem ini mampu memberikan informasi ketersediaan stok barang secara *real-time* dan akurat, yang membantu perusahaan dalam meminimalisir terjadinya selisih antara data administratif dengan jumlah fisik barang di gudang.

3. Kemudahan Operasional

Model pengembangan *Waterfall* yang digunakan terbukti efektif dalam memandu proses pembuatan aplikasi yang terstruktur (Azis, 2025), menghasilkan fitur-fitur yang mudah dipahami dan digunakan oleh pegawai di lingkungan CV. Permata Printing.

5.2 Saran

Saran Untuk pengembangan sistem di masa mendatang agar menjadi lebih optimal, penulis memberikan beberapa saran:

1. Pengembangan Fitur

Kedepannya, sistem dapat dikembangkan dengan menambahkan fitur notifikasi otomatis apabila stok barang mencapai batas minimum (*reorder point*), sehingga pengadaan barang dapat dilakukan lebih tepat waktu.

2. Integrasi Jaringan

Mengingat sistem saat ini masih berbasis desktop lokal, disarankan untuk mengintegrasikan sistem ke dalam jaringan *client-server* atau *cloud* agar data dapat diakses oleh beberapa pengguna secara bersamaan di lokasi yang berbeda.

3. Keamanan Data

Untuk menjaga integritas data, disarankan untuk menambahkan sistem *backup* data otomatis secara berkala guna mengantisipasi kerusakan atau kehilangan data pada perangkat keras.



REFERENCES

- Adam, S., Novianti, D., & Apriyani, D. D. (2023). Rancangan Aplikasi Inventory dan Penjualan pada Toko Ghaizan Cosmetic. *Jurnal Riset Dan Aplikasi Mahasiswa Informatika (JRAMI)*, 4(04). <https://doi.org/10.30998/jrami.v4i04.4932>
- Azis, A. A. (2025). RANCANGAN SISTEM MONITORING SISWA BERBASIS WEB DENGAN METODE WATERFALL. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3). <https://doi.org/10.23960/jitet.v13i3.6676>
- Hidayat, F., & Ardhiansyah, M. (2022). Pengembangan Sistem Informasi Pelayanan Kesehatan Menggunakan Model Waterfall (Studi Kasus: Poliklinik PUSDIKLAT). *Scientia Sacra: Jurnal Sains*, 2(2).
- Jayema, A., Wanda Putra, F., & Muhammad, M. (2024). PERANCANGAN SISTEM INFORMASI TINGKAT LAYANAN PELANGGAN PADA MAGZ LAUNDRY. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(5). <https://doi.org/10.36040/jati.v8i5.11052>
- Manurung, R., & Rizkina, M. (2021). Analisis Efisiensi Pelaksanaan Sistem Informasi Pengelolaan Inventarisasi Aset Daerah Pada Kecamatan Purwokerto Selatan. *Prosiding Seminar Nasional Ekonomi Dan Bisnis*, 1. <https://doi.org/10.33479/sneb.v1i.161>
- Nauval Abrari, M. F., Juliani, M. R., & Arni, S. (2025). PENGEMBANGAN SISTEM APLIKASI BERBASIS JAVA NETBEANS DAN MYSQL PADA APOTEK JAYA. *Jurnal Rekayasa Sistem Informasi Dan Teknologi*, 2(4). <https://doi.org/10.70248/jrsit.v2i4.2256>
- Nurfi, S. (2020). Sistem Informasi Inventori Barang Pada CV. Putra Karya Baja Dengan Metode Waterfall. *BINA INSANI ICT JOURNAL*, 7(2). <https://doi.org/10.51211/biict.v7i2.1403>
- Prasetyo, P. B. (2025). Perancangan Sistem Informasi Inventory Sparepart Mobil Berbasis Desktop pada PT Nusantara Bekasi. *Jurnal Riset Dan Aplikasi Mahasiswa Informatika (JRAMI)*, 6(01). <https://doi.org/10.30998/jrami.v6i01.8110>
- Setiawan, H., & Imam Suharjo. (2025). Perancangan Dashboard Sistem Notifikasi Otomatis Aset Berbasis WhatsApp API Dengan Pendekatan Waterfall. *Jurnal Komputer, Informasi Dan Teknologi*, 5(1). <https://doi.org/10.53697/jkomitek.v5i1.2493>
- Sundara, A., Andriyanto, S., & Pratama, M. S. (2023). Rancang Bangun Sistem Informasi Inventaris Barang Pada SMKN 1 Parittiga. *Jurnal Inovasi Teknologi Terapan*, 1(1). <https://doi.org/10.33504/jitt.v1i1.51>