

IMPLEMENTASI ALGORITMA K-MEANS UNTUK OPTIMALISASI STOK BARANG PADA TOKO ONLINE GENTA SARANA TEKNIK

Muhammad Rizky Ramadhan¹, Fanisya Alva Mustika², Ari Irawan³

Program Studi Teknik Informatika, Fakultas Teknik dan Ilmu Komputer

Universitas Indraprasta PGRI

Jalan Raya Tengah No 80, Kelurahan Gedong, Pasar Rebo, Jakarta Timur

aomineikie@gmail.com¹, funny.alva@gmail.com², masariiaja@gmail.com³

Abstrak

Pengelolaan stok pada Toko Online Genta Sarana Teknik masih bergantung pada pencatatan sederhana dan pengalaman, sehingga sering terjadi stok habis mendadak (stock-out) atau penumpukan barang (overstock). Penelitian ini bertujuan mengoptimalkan pengendalian persediaan melalui pengelompokan SKU berdasarkan tingkat pergerakan stok menggunakan algoritma K-Means. Data penelitian berasal dari data master SKU dan histori mutasi stok (stok masuk dan stok keluar) yang dikelola pada aplikasi desktop berbasis Java. Setiap SKU direpresentasikan oleh tiga fitur numerik, yaitu rata-rata stok masuk, rata-rata stok keluar, dan sisa stok, kemudian dinormalisasi dan diproses dengan $K=3$ untuk menghasilkan klaster prioritas tinggi, sedang, dan rendah. Hasil akhir menunjukkan 198 SKU aktif berhasil terkategorisasi menjadi Cluster 0 (88 SKU), Cluster 1 (71 SKU), dan Cluster 2 (39 SKU) dengan centroid akhir masing-masing (0,102609; 0,284091; 0,139312), (0,143482; 0,605634; 0,048502), dan (0,473412; 0,700855; 0,049925). Aplikasi menyediakan fitur pengelolaan stok, link alternatif pembelian Shopee/Tokopedia, proses clustering, serta laporan stok menipis dan hasil K-Means. Penerapan K-Means membantu penentuan prioritas restok lebih objektif dan terstruktur sehingga efisiensi pengelolaan stok dapat ditingkatkan.

Kata Kunci: *K-Means, Clustering, Persediaan, Stok Barang, Java*

Abstract

Inventory management at the Genta Sarana Teknik Online Store still relies on simple record-keeping and experience, resulting in frequent instances of sudden stockouts or inventory buildup. This study aims to optimize inventory control by clustering SKUs based on inventory turnover rates using the K-Means algorithm. The research data was derived from SKU master data and inventory transaction history managed in a Java-based desktop application. Each SKU was represented by three numerical features—average stock inflow, average stock outflow, and remaining stock—which were then normalized and processed with $K=3$ to generate high-, medium-, and low-priority clusters. The final results show that 198 active SKUs were successfully clustered into Cluster 0 (88 SKUs), Cluster 1 (71 SKUs), and Cluster 2 (39 SKUs) with final centroids of 0.102609, 0.284091, and 0.139312; 0.143482, 0.605634, and 0.048502; and 0.473412, 0.700855, and 0.049925. The app provides inventory management features, alternative purchase links for Shopee/Tokopedia, and the clustering process, as well as low-stock alerts and K-Means results. The application of K-Means helps determine restocking priorities in a more objective and structured manner, thereby improving inventory management efficiency.

Keywords: *K-Means, Clustering, Inventory, Stock Control, Java*

PENDAHULUAN

Persediaan barang merupakan aset penting bagi usaha yang secara langsung mempengaruhi tingkat layanan pelanggan dan arus kas perusahaan (Samsudin et al., 2024). Pada Toko Online Genta Sarana Teknik, penentuan jumlah stok masih mengandalkan pencatatan sederhana serta pengalaman sebelumnya, sehingga belum mampu menggambarkan pola pergerakan barang secara akurat. Kondisi tersebut menyebabkan dua risiko utama, yaitu kekosongan stok saat permintaan tinggi (*stock-out*) dan penumpukan barang yang memperbesar modal tertahan (*overstock*) (Rajsya, 2024). Ketidakseimbangan antara pasokan dan permintaan akibat pengelolaan stok yang kurang tepat dapat berdampak signifikan terhadap profitabilitas dan kepuasan pelanggan (Komtekinfo, 2023). Untuk mengurangi risiko tersebut, diperlukan pengolahan data historis mutasi stok agar prioritas pengadaan ulang (restok) dapat ditentukan secara objektif.

Salah satu pendekatan yang relevan adalah *data mining* dengan teknik *clustering*, yang mengelompokkan data berdasarkan kemiripan karakteristik (Jauhari, 2022). Metode ini telah

terbukti efektif untuk mendukung pengambilan keputusan persediaan pada berbagai jenis usaha. Algoritma *K-Means* merupakan salah satu algoritma *clustering* yang banyak digunakan karena sifatnya yang sederhana, efisien secara komputasi, dan mudah diimplementasikan pada data numerik (Indriani et al., 2020; Ariansyah et al., 2021). Beberapa penelitian sebelumnya menunjukkan bahwa *K-Means* mampu mengklasterisasi data stok barang sehingga kebijakan pengadaan dapat disesuaikan dengan karakteristik masing-masing kelompok barang (Asrul, 2019; Setiawan & Sulastri, 2021).

Rumusan masalah pada penelitian ini adalah bagaimana menerapkan algoritma *K-Means* pada aplikasi desktop berbasis Java untuk mengelompokkan barang berdasarkan tingkat pergerakan stok, serta bagaimana memanfaatkan hasil kluster sebagai dasar kebijakan prioritas restok pada toko. Tujuan penelitian adalah membangun aplikasi yang mampu mencatat stok masuk/keluar, mengelola data master SKU dan link pembelian *marketplace*, menjalankan proses *K-Means*, serta menghasilkan laporan analisis stok. Manfaat penelitian diharapkan membantu pengelola toko dalam pengambilan keputusan persediaan secara lebih terukur.

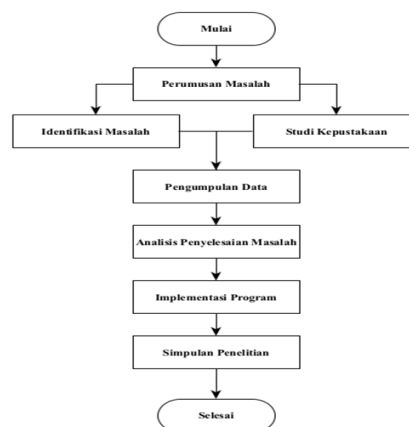
Pengelompokan prioritas stok juga diperlukan karena toko menyediakan beberapa kanal pembelian. Ketika salah satu *marketplace* mengalami keterbatasan stok atau perubahan harga, admin membutuhkan tautan alternatif untuk tetap dapat melakukan pengadaan. Oleh karena itu, sistem juga menyediakan fitur link pembelian alternatif beserta prioritas penayangan.

PENELITIAN RELEVAN

Penelitian terkait pengendalian persediaan dengan *K-Means* telah banyak dilakukan. Indriani et. (2022) menerapkan *K-Means* untuk menentukan persediaan stok barang dan menunjukkan bahwa pengelompokan membantu penentuan kebijakan pengadaan. Pada konteks pengendalian persediaan aki, Asrul (2019) memanfaatkan *K-Means* untuk klasterisasi kebutuhan sehingga keputusan pemesanan lebih tepat. Setiawan & Sulastri (2021) membandingkan *K-Means* dan *K-Medoids* untuk optimalisasi stok dan menemukan bahwa kedua metode menghasilkan tiga kluster dengan karakteristik berbeda. Samsudin et al. (2024) menekankan bahwa analisis *K-Means* membantu identifikasi pola permintaan untuk strategi persediaan yang lebih efektif. Berbeda dari penelitian-penelitian tersebut, studi ini mengintegrasikan *K-Means* langsung ke dalam aplikasi desktop Java yang juga menyediakan fitur pencatatan mutasi stok, pengelolaan link alternatif pembelian Shopee/Tokopedia, serta laporan stok menipis dan hasil kluster sebagai dukungan operasional harian. Secara umum, kontribusi utama penelitian ini adalah penggabungan proses analitik (*K-Means*) dan proses operasional (pencatatan stok) pada satu aplikasi. Dengan demikian, hasil *clustering* dapat langsung digunakan sebagai rekomendasi tindak lanjut pada aktivitas restok harian.

METODE PENELITIAN

Penelitian ini dirancang dengan menerapkan metode *K-Means* yang dilakukan secara bertahap untuk memastikan prosesnya berjalan sesuai tujuan yang telah ditetapkan. Tahapan pelaksanaan penelitian ditampilkan dalam Gambar 1.



Gambar 1. Tahapan Penelitian

Tahapan penelitian meliputi pengumpulan data, analisis kebutuhan, studi kepustakaan, perancangan sistem, pengujian sistem, dan penyusunan laporan. Objek penelitian adalah data barang (SKU) yang dikelola pada sistem. Sumber data berasal dari data master SKU (kode, nama barang, satuan, stok minimum) dan data mutasi stok (stok masuk dan stok keluar). Pada proses *clustering*, setiap SKU direpresentasikan sebagai vektor fitur numerik yang mencerminkan pergerakan stok, yaitu rata-rata stok masuk, rata-rata stok keluar, dan sisa stok. Data yang tidak aktif dikeluarkan dari perhitungan sehingga hanya SKU aktif yang diproses.

Metode pengembangan aplikasi mengikuti pendekatan *waterfall* yang mencakup analisis, perancangan, implementasi, dan pengujian. Implementasi dilakukan menggunakan bahasa pemrograman Java pada NetBeans 8.2 dengan basis data relasional. Pengujian fungsional dilakukan dengan metode *black box* untuk memastikan setiap fitur utama berjalan sesuai kebutuhan pengguna.

Kebutuhan Sistem

Kebutuhan Fungsional. Sistem menyediakan autentikasi pengguna, pengelolaan data master SKU, pengelolaan stok minimum, pencatatan mutasi stok masuk dan stok keluar, pengelolaan link pembelian alternatif (Shopee/Tokopedia) beserta prioritas, proses *clustering K-Means* per periode, tampilan hasil *cluster*, dan pembuatan laporan (stok saat ini, stok menipis/habis, rekap mutasi, dan hasil *clustering*).

Kebutuhan Nonfungsional. Aplikasi berjalan pada desktop dengan antarmuka sederhana, memanfaatkan basis data relasional, mendukung validasi input untuk mencegah stok minus, serta menampilkan informasi yang mudah dipahami admin toko.

Algoritma K-Means

Proses *K-Means* dilakukan dengan langkah: (1) menentukan jumlah kluster K, (2) memilih centroid awal secara acak dari data, (3) menghitung jarak setiap data ke centroid menggunakan Euclidean Distance, (4) mengelompokkan data berdasarkan jarak minimum, (5) memperbarui centroid sebagai rata-rata anggota kluster, dan (6) mengulangi proses hingga centroid stabil atau tidak ada perubahan keanggotaan kluster.

Pseudocode

```
Input: data SKU ter-normalisasi X, jumlah cluster K
Output: label cluster untuk tiap SKU dan centroid akhir C
1. Pilih K centroid awal C secara acak dari X
2. Ulangi
3. Untuk setiap data xi di X:
4. Hitung jarak d(xi,Cj) untuk j=1..K (Euclidean)
5. Tentukan label(xi) = argmin_j d(xi,Cj)
6. Untuk setiap cluster j:
7. Perbarui centroid Cj = rata-rata semua xi dengan label(xi)=j
8. Sampai centroid stabil / tidak ada perubahan label
```

Tabel 1. Ringkasan Proses Iterasi K-Means

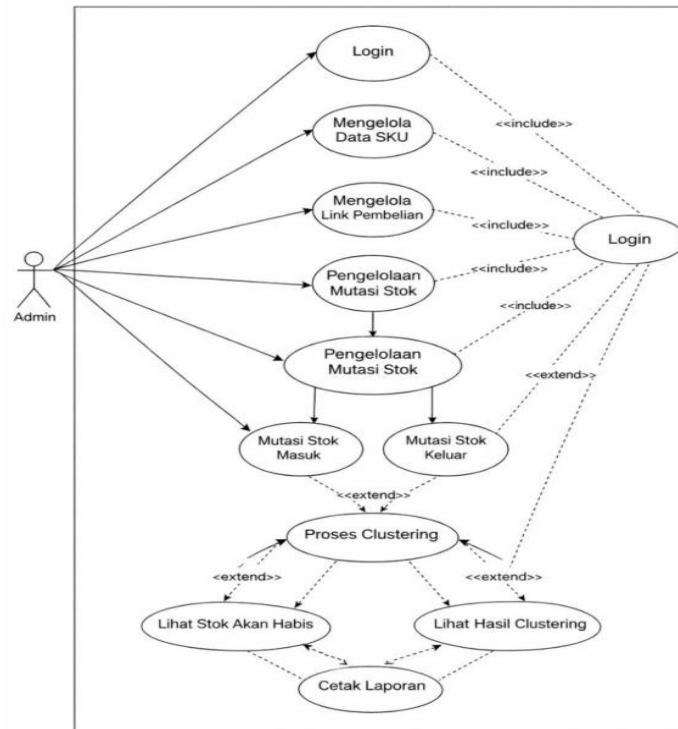
Tahap	Aktivitas	Keluaran
1	Inisialisasi K dan centroid awal	Centroid awal untuk tiap kluster
2	Hitung jarak Euclidean setiap SKU ke semua centroid	Matriks jarak
3	Assign kluster berdasarkan jarak minimum	Keanggotaan kluster sementara
4	Hitung centroid baru (rata-rata fitur anggota)	Centroid baru
5	Cek konvergensi (centroid/anggota berubah atau tidak)	Status berhenti/lanjut

HASIL DAN PEMBAHASAN

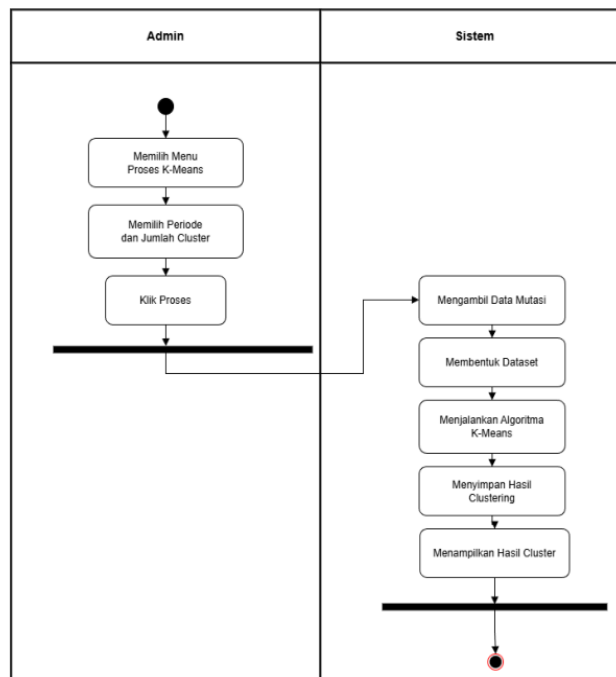
Pada tahap implementasi, aplikasi desktop menyediakan modul login, pengelolaan data SKU, mutasi stok masuk/keluar, pengelolaan link pembelian *marketplace*, proses *K-Means*, tampilan

hasil klaster, serta pencetakan laporan. Perancangan sistem didokumentasikan menggunakan UML untuk menggambarkan fungsi dan alur proses utama.

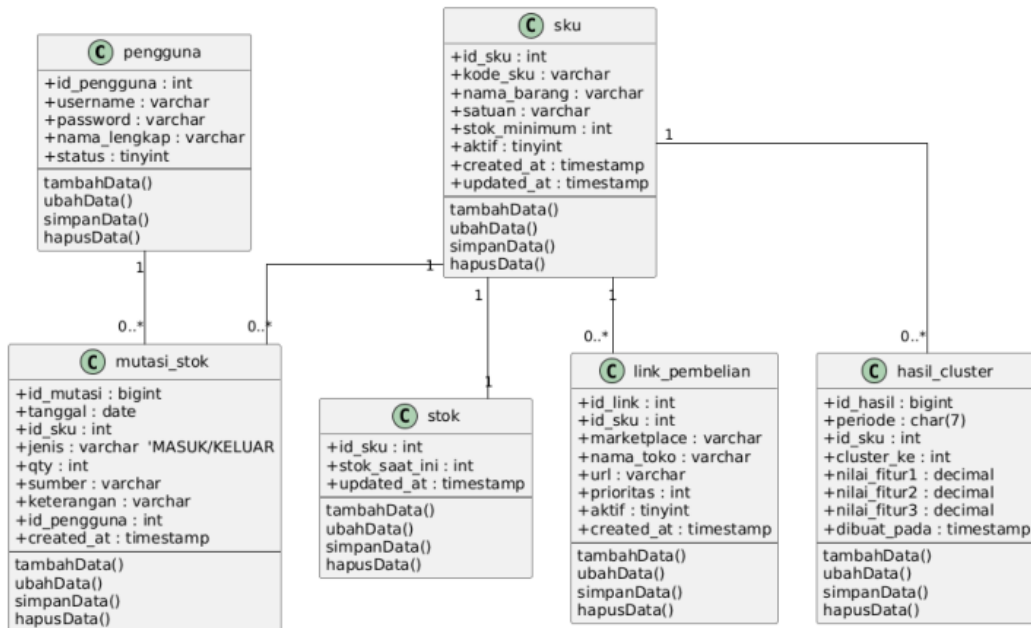
Dokumentasi perancangan menggunakan UML ditunjukkan pada Gambar 2–4. Diagram ini memastikan bahwa kebutuhan pengguna terpetakan ke fungsi aplikasi dan struktur data yang dibangun.



Gambar 2. Use Case Diagram



Gambar 3. Activity Diagram Proses K-Means



Gambar 4. Class Diagram

Desain Basis Data

Basis data menyimpan data master dan transaksi untuk mendukung proses perhitungan fitur, *clustering*, dan pelaporan. Ringkasan entitas utama ditunjukkan pada Tabel 2.

Tabel 2. Ringkasan Entitas Basis Data

Entitas	Atribut Utama
Pengguna	id_pengguna, username, password, nama_lengkap, status
Sku	id_sku, kode_sku, nama_barang, satuan, stok_minimum, aktif
Stok	id_sku, stok_saat_ini, updated_at
mutasi_stok	id_mutasi, tanggal, id_sku, jenis (MASUK/KELUAR), qty, keterangan, id_pengguna
link_pembelian	id_link, id_sku, marketplace, nama_toko, nama_produk, url, prioritas, aktif
hasil_cluster	id_hasil, periode, id_sku, cluster_ke, nilai_fitur1, nilai_fitur2, nilai_fitur3

Pembentukan Dataset dan Normalisasi

Dataset dibentuk dari histori mutasi stok untuk setiap SKU. Setelah atribut dihitung, nilai fitur dinormalisasi agar berada pada rentang 0–1 sehingga perbedaan skala antar fitur tidak mendominasi perhitungan jarak. Penelitian menetapkan $K=3$ untuk menghasilkan kelompok prioritas restok tinggi, sedang, dan rendah.

Tabel 3 menyajikan contoh 10 baris data hasil normalisasi (subset) untuk menggambarkan bentuk dataset.

Tabel 3. Contoh Data Normalisasi (Subset)

Kode SKU	Fitur 1	Fitur 2	Fitur 3
SB0001	0,011084	0,000000	0,204545
SB0002	0,000000	0,000000	0,000000
SB0003	0,012315	0,166667	0,080128
SB0004	0,005542	0,166667	0,085470
SB0005	0,009852	0,000000	0,152597
SB0006	0,014778	0,333333	0,092703
SB0007	0,006773	0,166667	0,064103

SB0008	0,002463	0,000000	0,041667
SB0009	1,000000	0,833333	0,009680
SB0010	0,018473	0,333333	0,137821

Hasil Clustering

Berikut hasil centroid akhir dan distribusi anggota *cluster* yang menjadi dasar pemetaan prioritas restok.

Tabel 4. Centroid Akhir Tiap Cluster

Cluster	Centroid Fitur 1	Centroid Fitur 2	Centroid Fitur 3
0	0,102609	0,284091	0,139312
1	0,143482	0,605634	0,048502
2	0,473412	0,700855	0,049925

Berdasarkan centroid akhir pada Tabel 4, interpretasi karakteristik klaster adalah sebagai berikut. *Cluster 2* memiliki nilai centroid fitur 1 dan fitur 2 paling tinggi sehingga merepresentasikan barang dengan intensitas mutasi tinggi (*fast moving*) dan dipetakan sebagai prioritas restok tinggi. *Cluster 1* menunjukkan pergerakan sedang dan menjadi prioritas restok terjadwal. *Cluster 0* memiliki pergerakan relatif rendah sehingga prioritas restoknya lebih longgar untuk mencegah penumpukan barang.

Tabel 5. Distribusi Jumlah Anggota Cluster

Cluster	Jumlah SKU	Kategori Prioritas
0	88	Rendah (tidak prioritas restok)
1	71	Sedang (restok terjadwal)
2	39	Tinggi (prioritas restok)

Total terklastrer adalah 198 SKU aktif; terdapat 2 SKU nonaktif sehingga tidak diproses pada pengelompokan.

Contoh Perhitungan Jarak

Misalkan salah satu SKU memiliki vektor fitur $x = (1,000000; 0,833333; 0,009680)$. Jarak ke centroid *Cluster 0* dihitung dengan: $d_0 = \sqrt{((x_1 - c_{01})^2 + (x_2 - c_{02})^2 + (x_3 - c_{03})^2)}$. Proses serupa dilakukan untuk semua *cluster*, kemudian SKU ditempatkan pada *cluster* dengan jarak minimum. Secara komputasi, proses ini dieksekusi untuk seluruh SKU pada setiap iterasi pembaruan centroid.

Implementasi *clustering* per periode memungkinkan admin melakukan evaluasi secara berkala (misalnya bulanan) sehingga perubahan pola penjualan dapat terdeteksi. Hasil *cluster* disimpan pada tabel hasil_cluster sehingga historinya dapat dibandingkan antar periode.

Dalam operasional, prioritas restok ditentukan dengan aturan: (a) jika barang berada pada *Cluster 2* dan stok saat ini $\leq$ stok minimum maka masuk daftar prioritas tinggi; (b) jika barang berada pada *Cluster 1* dan stok saat ini mendekati stok minimum maka masuk prioritas sedang; (c) *Cluster 0* dipantau namun tidak direkomendasikan untuk restok besar agar menghindari *overstock*.

Keterbatasan penelitian ini adalah pemilihan $K=3$ masih berdasarkan kebutuhan bisnis dan interpretasi prioritas, sehingga pada kondisi data berbeda bisa dilakukan evaluasi menggunakan metode *elbow* atau *Davies–Bouldin Index*. Selain itu, fitur yang digunakan masih terbatas pada ringkasan mutasi; pengembangan berikutnya dapat menambahkan fitur frekuensi transaksi, tren musiman, atau integrasi data penjualan *marketplace*.

Pengujian Sistem

Pengujian dilakukan dengan metode *black box* berdasarkan skenario penggunaan admin. Ringkasan pengujian ditunjukkan pada Tabel 6.

Tabel 6. Ringkasan Pengujian Fungsional (*Black Box*)

No	Fitur	Skenario Uji	Hasil
1	Login	Input username/password valid	Berhasil masuk ke dashboard
2	Login	Input tidak valid	Muncul pesan kesalahan
3	Data SKU	Tambah/ubah/hapus SKU	Data tersimpan dan daftar terbaru
4	Mutasi Stok	Input stok masuk/keluar	Stok terhitung otomatis sesuai mutasi
5	Link Pembelian	Tambah link Shopee/Tokopedia dan prioritas	Link tersimpan dan tampil sesuai urutan prioritas
6	Proses K-Means	Menjalankan proses dengan periode dan K=3	Hasil kluster tersimpan dan dapat ditampilkan

Berdasarkan hasil pengujian, fungsi utama sistem berjalan sesuai kebutuhan. Keberadaan modul laporan memungkinkan admin mencetak atau menyimpan laporan stok saat ini, stok menipis/habis, mutasi stok, dan hasil *K-Means* sebagai bahan evaluasi pengelolaan persediaan.

Tampilan Layar

No	Kode SKU	Nama Barang	Satuan	Stok Minimum	Stok Saat Ini	Aksi
1	SR0001	Kontaktor 9A Coil 220VAC Schneider	buah	1	5	Aksi
2	SR0002	MCB 2P 16A Schneider	kg	3	44	Aksi
3	SR0003	LED 1W42 12C Generic	cm	20	17	Aksi
4	SR0004	OLED 0.96 inch I2C Generic	lembar	20	20	Aksi
5	SR0005	Obeng Set 6 pcs Knibrow	buah	15	285	Aksi
6	SR0006	Tang Kombinasi 7 inch Kemaster	pcs	2	111	Aksi
7	SR0007	Panel LED Downlight 12W Osram	pcs	20	4	Aksi
8	SR0008	Kabel LAN Cat5 UTP 305m Canare	box	5	18	Aksi
9	SR0009	Tang Kombinasi 7 inch Inggo	pcs	2	16	Aksi
10	SR0010	Kabel LAN Cat5 UTP 305m Supreme	box	0	5	Aksi
11	SR0011	Tang Kombinasi 7 inch Dewatt	pcs	2	55	Aksi
12	SR0012	Tang Kombinasi 7 inch Stanley	pcs	1	14	Aksi
13	SR0013	Isopropil Alcohol (IPA) 1 liter Lochite	liter	0	6	Aksi
14	SR0014	Baterai 9V 100 Vivan	pcs	1	6	Aksi
15	SR0015	LED 1W42 12C Raspberry Pi	pcs	5	57	Aksi
16	SR0016	Duct Cable 6W40 2m Hager	batang	0	0	Aksi
17	SR0017	Power Supply Switching 24V 10A Omron	unit	2	0	Aksi
18	SR0018	LED Strip 5050 RGB 5m Generic	meter	15	13	Aksi
19	SR0019	Palu 16 oz Dewatt	pcs	0	108	Aksi
20	SR0020	Flux Solder 50 ml Nitto	ml	3	1	Aksi
21	SR0021	Kunci Ingress 12 inch Tekiro	pcs	10	68	Aksi
22	SR0022	Kunci Ingress 12 inch Kemaster	pcs	10	49	Aksi
23	SR0023	OLED 0.96 inch I2C Raspberry Pi	pcs	0	2	Aksi
24	SR0024	Kapalator Keramik 10K 100V5% Yageo	pak	2	1	Aksi
25	SR0025	Lampu LED Bulb 10W E27 Osram	pcs	3	78	Aksi
26	SR0026	7-Segment 0.96 inch 100V Raspberry Pi	pak	20	1	Aksi
27	SR0027	Meteran 5 meter Kemaster	pcs	0	40	Aksi
28	SR0028	Tang Kombinasi 7 inch Bosch	pcs	20	83	Aksi
29	SR0029	MCB 1P 16A ABB	pcs	2	30	Aksi

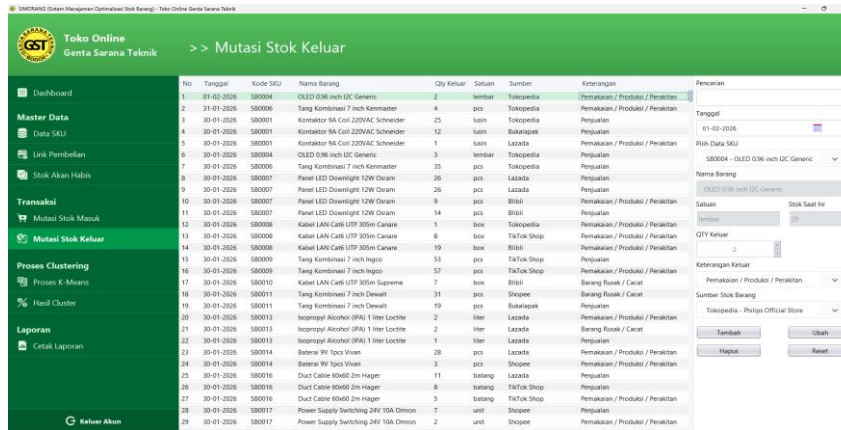
Gambar 5. Tampilan Layar Data SKU

Pada halaman ini ditampilkan ringkasan informasi penting, seperti jumlah total SKU, jumlah barang dengan kondisi stok menipis atau habis, serta ringkasan aktivitas mutasi stok. *Dashboard* dirancang untuk memberikan gambaran umum kondisi stok secara cepat sehingga pengguna dapat segera mengambil keputusan terkait pengelolaan persediaan barang.

No	Tanggal	Kode SKU	Nama Barang	Qty Masuk	Satuan	Sumber	Nama Toko	Keterangan
1	31-01-2026	SR0001	Kontaktor 9A Coil 220VAC Schneider	5	buah	Tokopedia	JM Official Store	Re-stok barang
2	31-01-2026	SR0004	OLED 0.96 inch I2C Generic	10	lembar	Tokopedia	Philly Official Store	Diunggah
3	30-01-2026	SR0001	Kontaktor 9A Coil 220VAC Schneider	13	buah	Tokopedia	JM Official Store	Penggantian Barang Rusak
4	30-01-2026	SR0003	LED 1W42 12C Generic	11	cm	Bibit	Stanley Official Store	Barang datang Berhapp
5	30-01-2026	SR0004	OLED 0.96 inch I2C Generic	4	lembar	Lazada	Tekiro Official Store	Re-stok barang
6	30-01-2026	SR0005	Obeng Set 6 pcs Knibrow	11	buah	Bibit	Maista Official Store	Re-stok barang
7	30-01-2026	SR0005	Obeng Set 6 pcs Knibrow	22	buah	Shopee	Stanley Official Store	Keluar dan Pengalangan
8	30-01-2026	SR0005	Obeng Set 6 pcs Knibrow	36	buah	Shopee	Stanley Official Store	Penggantian Barang Rusak
9	30-01-2026	SR0007	Panel LED Downlight 12W Osram	46	pcs	Bibit	Schneider Electric Off...	Pembelian Awal Stok
10	30-01-2026	SR0008	Kabel LAN Cat5 UTP 305m Canare	11	box	Bibit	Gross Perakasan Nuan...	Pembelian Awal Stok
11	30-01-2026	SR0008	Kabel LAN Cat5 UTP 305m Canare	15	box	Tokopedia	Gross Perakasan Nuan...	Barang datang Berhapp
12	30-01-2026	SR0008	Kabel LAN Cat5 UTP 305m Canare	7	box	Bibit	Maista Official Store	Re-stok barang
13	30-01-2026	SR0008	Kabel LAN Cat5 UTP 305m Canare	5	box	TiTaTa Shop	Perkakas Pro Store	Barang datang Berhapp
14	30-01-2026	SR0008	Kabel LAN Cat5 UTP 305m Canare	6	box	Bibit	Maista Official Store	Pembelian Awal Stok
15	30-01-2026	SR0009	Tang Kombinasi 7 inch Inggo	36	pcs	TiTaTa Shop	LED & Display Store	Penggantian Barang Rusak
16	30-01-2026	SR0009	Tang Kombinasi 7 inch Inggo	43	pcs	Lazada	Schneider Electric Off...	Pembelian Awal Stok
17	30-01-2026	SR0010	Kabel LAN Cat5 UTP 305m Supreme	4	box	Bibit	Parasonic Official Store	Pembelian Awal Stok
18	30-01-2026	SR0011	Tang Kombinasi 7 inch Dewatt	23	pcs	Shopee	Knibrow Official Store	Penggantian Barang Rusak
19	30-01-2026	SR0011	Isopropil Alcohol (IPA) 1 liter Lochite	2	liter	Tokopedia	Komponen & Sensor	Pembelian Awal Stok
20	30-01-2026	SR0013	Isopropil Alcohol (IPA) 1 liter Lochite	6	liter	Lazada	Energizer Official Store	Barang datang Berhapp
21	30-01-2026	SR0016	Duct Cable 6W40 2m Hager	7	batang	Lazada	Energizer Official Store	Penyesuaian Stok (opname)
22	30-01-2026	SR0016	Duct Cable 6W40 2m Hager	4	batang	Lazada	Energizer Official Store	Pembelian Awal Stok
23	30-01-2026	SR0018	LED Strip 5050 RGB 5m Generic	14	meter	Tokopedia	Knibrow Official Store	Pembelian Awal Stok
24	30-01-2026	SR0018	LED Strip 5050 RGB 5m Generic	8	meter	TiTaTa Shop	Maista Official Store	Pembelian Awal Stok
25	30-01-2026	SR0020	Flux Solder 50 ml Nitto	3	ml	Tokopedia	Toko Panel & MCB	Pembelian Awal Stok
26	30-01-2026	SR0020	Flux Solder 50 ml Nitto	15	ml	Tokopedia	Toko Panel & MCB	Penggantian Barang Rusak
27	30-01-2026	SR0021	Kapalator Keramik 10K 100V5% Yageo	1	pak	Bibit	Parasonic Official Store	Penggantian Barang Rusak
28	30-01-2026	SR0027	Meteran 5 meter Kemaster	37	pcs	Tokopedia	Bawes Official Store	Pembelian Awal Stok
29	30-01-2026	SR0028	Tang Kombinasi 7 inch Bosch	2	pcs	TiTaTa Shop	Bosch Professional Off...	Barang datang Berhapp

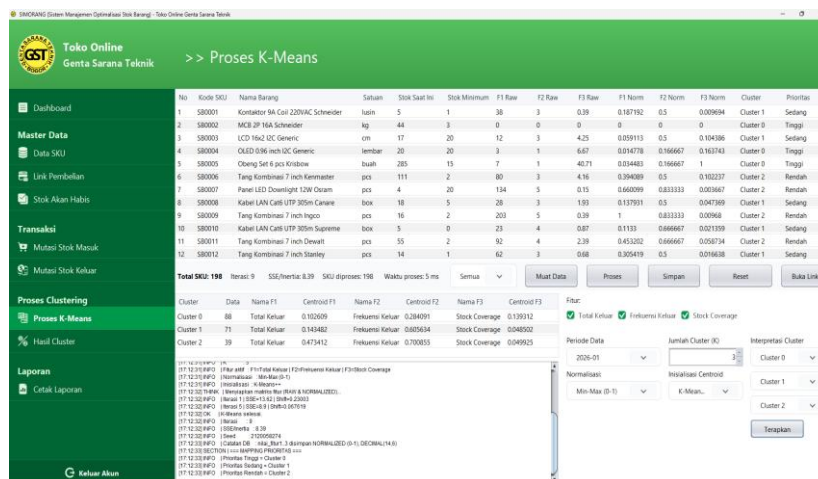
Gambar 6. Tampilan Layar Mutasi Stok Masuk

Pada layar ini, pengguna mengisi data mutasi berupa tanggal, SKU, jumlah barang, dan sumber pembelian. Setelah data disimpan, sistem akan secara otomatis menambahkan jumlah stok barang sesuai dengan data mutasi yang dimasukkan.



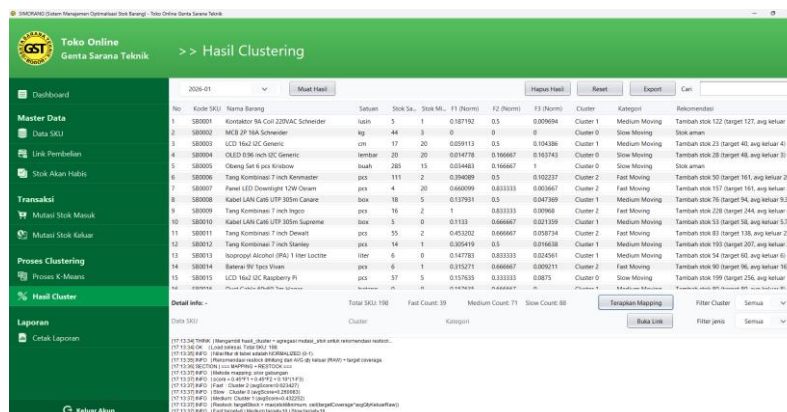
Gambar 7. Tampilan Layar Mutasi Stok Keluar

Pada layar ini, pengguna memasukkan data mutasi yang diperlukan, kemudian sistem akan melakukan validasi ketersediaan stok. Apabila stok mencukupi, data mutasi akan disimpan dan stok barang akan berkurang secara otomatis.



Gambar 8. Tampilan Layar Proses K-Means

Pada layar ini, pengguna menentukan periode data dan jumlah *cluster* yang akan digunakan dalam proses perhitungan. Setelah tombol proses ditekan, sistem akan membentuk dataset dari data mutasi stok, melakukan perhitungan algoritma *K-Means*, dan menyimpan hasil pengelompokan ke dalam database. Proses ini bertujuan untuk mengelompokkan barang menjadi kategori tertentu, seperti barang dengan pergerakan cepat dan lambat.



Gambar 9. Tampilan Layar Hasil Clustering

Pada layar ini, pengguna dapat memilih periode hasil clustering yang ingin ditampilkan, Sistem kemudian menampilkan daftar barang beserta informasi cluster dan nilai fitur yang digunakan, Informasi ini dapat digunakan sebagai dasar pengambilan keputusan dalam menentukan prioritas pengadaan ulang stok barang.



Period	Cluster	Kode	Nama Barang	Satuan	Stock Update	Stock Minimum	Total	Prioritas	Nilai rata
2026-01	0	580002	MCB 2P 15A Schneider	kg	44	3	0.000000	0.000000	0.000000
2026-01	0	580004	OLED 0.96 inch GC Generic	lembar	20	20	0.014778	0.166667	0.163743
2026-01	0	580005	Chong Set 6 pos Kriblow	buah	285	15	0.034483	0.166667	1.000000
2026-01	0	580015	LCD 16x2 GC Raspberry Pi	pcs	67	5	0.157635	0.333333	0.087500
2026-01	0	580021	Kunci Inggris 12 inch Telexo	pcs	68	10	0.261084	0.333333	0.063025
2026-01	0	580022	Kunci Inggris 12 inch Kermaster	pcs	49	10	0.029957	0.333333	0.401169
2026-01	0	580024	Kapastor Karamis 104 100pcs	pak	1	2	0.029113	0.333333	0.004064
2026-01	0	580026	7-Segment 0.56 inch 10pcs	pak	1	20	0.088670	0.333333	0.002729
2026-01	0	580033	Module Step Down LM2596	pcs	2	0	0.300403	0.333333	0.001811
2026-01	0	580035	Chong Set 6 pos Meista	set	29	15	0.024631	0.333333	0.284912
2026-01	0	580036	Power Supply Switching 24V 10A	unit	10	3	0.196374	0.333333	0.022329
2026-01	0	580040	Lampu LED Bulb 12W E27	pcs	273	0	0.116227	0.333333	0.058772
2026-01	0	580042	Kabel LAN Cat5 UTP 305m	box	43	5	0.083744	0.333333	0.124251
2026-01	0	580044	Baterai AA Alkaline 4pcs Vixon	pak	7	0	0.123153	0.333333	0.013754
2026-01	0	580045	Kabel NYA 1.5mm 100m Belden	dl	52	10	0.009452	0.166667	0.638596
2026-01	0	580049	Kunci L Set 1.5-10mm Kriblow	set	1	15	0.066966	0.333333	0.003510
2026-01	0	580050	MCB 1P 6A Schneider	pcs	34	20	0.064039	0.333333	0.128476
2026-01	0	580062	Powerbank 10000mAh Panasonic	unit	5	15	0.113300	0.333333	0.010579
2026-01	0	580067	Flux Solder 50 ml Generic	ml	16	0	0.029457	0.333333	0.150603
2026-01	0	580069	Kabel LAN Cat5 UTP 305m	box	2	15	0.054187	0.333333	0.008931
2026-01	0	580080	Cable Tie 20mm 100pcs Mbu	pak	16	0	0.103448	0.333333	0.027427
2026-01	0	580081	Bow Panel 30x40x20mm ABS	unit	26	2	0.073362	0.333333	0.085147
2026-01	0	580096	7-Segment 0.56 inch 10pcs	pak	30	20	0.034483	0.166667	0.156263
2026-01	0	580098	Duct Cable 60x60 2m Schneider	belang	34	0	0.088670	0.333333	0.082788
2026-01	0	580070	Kondensor 10K Cat 220VAC	pcs	176	10	0.088670	0.166667	0.240196

Gambar 10. Tampilan Layar Cetak Laporan Hasil Proses K-Means

Laporan hasil *K-Means* ini digunakan sebagai dasar analisis untuk menentukan prioritas pengadaan ulang stok barang, terutama untuk mengidentifikasi barang dengan tingkat pergerakan stok yang tinggi maupun rendah.

SIMPULAN

Algoritma *K-Means clustering* berhasil diterapkan pada aplikasi desktop berbasis Java untuk mengelompokkan barang berdasarkan tingkat pergerakan stok pada Toko Online Genta Sarana Teknik. Aplikasi yang dibangun mampu membantu pencatatan stok masuk dan stok keluar serta menyajikan laporan analisis stok yang dapat digunakan sebagai dasar pengambilan keputusan pengelolaan persediaan. Dengan sistem ini, risiko *stock-out* dan *overstock* dapat diminimalkan sehingga efisiensi pengelolaan stok dapat ditingkatkan.

DAFTAR PUSTAKA

- Ariansyah, P. E., Santi, I. H., & Kirom, S. (2021). Penerapan algoritma K-Means untuk mengetahui akurasi persediaan barang menggunakan evaluasi Davies–Bouldin Index. *Jurnal Universitas Islam Balitar*.
- Arniati, L. (2020). Penerapan metode K-Means untuk menentukan persediaan stok pada PT. Aneka Usaha. *Jurnal Nusa Mandiri*.
- Asrul. (2019). Klasterisasi pengendalian persediaan aki menggunakan metode K-Means. *JOINTECS*, 4(1), 5–12.
- Dhona, D. R., Yulianingsih, & Saputra, S. (2025). Implementasi sistem klasifikasi inventaris menggunakan metode clustering dengan algoritma K-Means untuk pengelolaan stok barang di D'Cafe Indramayu. *Faktor Exacta*, 18(2), 155–160.
- Indriani, D., Irawan, B., & Bahtiar, A. (2020). Penerapan algoritma K-Means clustering untuk menentukan persediaan stok barang. *JATI*.
- Jauhari. (2022). Penerapan metode K-Means clustering dalam penentuan persediaan stok barang di Tono Tani Depok. *Jurnal Rekayasa Komputasi Terapan (JRKT)*, 4(4). <https://doi.org/10.30998/jrkt.v4i04.11914>.
- Komtekinfo. (2023). Analisis pengelompokkan stok barang dengan memanfaatkan algoritma K-Means. *Komtekinfo*.
- Kurniawan, T. A., & Sari, R. P. (2022). Implementasi metode waterfall pada pengembangan aplikasi manajemen stok berbasis desktop. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 4(2), 112–121.
- Lestari, F., & Handayani, T. (2021). Analisis perbandingan metode clustering K-Means dan DBSCAN untuk segmentasi data penjualan. *Jurnal Teknologi dan Sistem Informasi*, 7(1), 45–54.
- Nugroho, A. S., & Fitriansyah, R. (2023). Penerapan normalisasi Min-Max pada algoritma K-Means untuk meningkatkan kualitas clustering data produk. *Jurnal Sistem Informasi dan Teknologi*, 5(3), 78–87.

- Prasetyo, E. (2020). *Data mining: Konsep dan aplikasi menggunakan MATLAB*. Yogyakarta: Andi Offset.
- Rajsya, A., Purnawansyah, & Manga', A. R. (2024). Rancang bangun penerapan metode elbow pada K-Means untuk clustering data persediaan barang. *LINIER*, 1(4), 395–403.
- Samsudin, R., Martanto, & Hayati, U. (2024). Optimalisasi stok barang melalui algoritma K-Means clustering analisis untuk manajemen persediaan dalam konteks bisnis modern. *JATI*.
- Setiawan, B. A., & Sulastri. (2021). Perbandingan clustering optimalisasi stok barang menggunakan algoritma K-Means dan K-Medoids. *SENdi U*.
- Widodo, P., & Rahayu, S. (2022). Sistem informasi pengendalian persediaan barang menggunakan metode Economic Order Quantity (EOQ) berbasis Java. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 6(4), 1823–1832.