**LAPORAN AKHIR**

**PEMOGRAMAM OBJEK 2**

**TUGAS BESAR**

**PEMBUATAN APLIKASI TOKO MEUBEL**

**DISUSUN OLEH :**

|  |  |
| --- | --- |
| **ROLAN FIRMANSYAH**  **ILHAM MAULANA M**  **ERWIN MAULANA** | **3411211069**  **3411211053**  **3411211047** |

****

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN INFORMATIKA**

**UNIVERSITAS JENDRAL ACHMAD YANI**

**2023**

# KATA PENGANTAR

Puji syukur ke hadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya kepada kita sehingga kita dapat berkumpul dalam suatu kesempatan yang berharga ini.

Melalui kesempatan yang baik ini, saya ingin menyampaikan kata pengantar dalam sebuah acara penting. Acara ini merupakan suatu momen yang sangat berarti bagi kita semua. Karena acara ini dapat menjadi sarana untuk saling berbagi informsi, pengetahuan, pengalaman, dan memperkuat tali silaturahmi.

Proposal ini disusun sebagai Tugas Besar mata kuliah Pemograman Objek 2 dengan menggunakan bahasa Java. Dalam proposal ini, kami memaparkan tentang pengembangan “Aplikasi Toko Meubel berbasis Java” yang bertujuan untuk memudahakan pemilik toko dalam pengelolaan data produk, pelanggan, transaksi, dan iventaris dengan mudah dan cepat.

Kami ingin mengucapakan terima kasih kepada pihak-pihak yang telah membantu kami memberikan dukungan selama mengerjakan proposal ini. Sehingga kami bisa menyelesaikan proposal ini sebagai tugas yang diberikan oleh dosen pengampu mata kuliah Ibu Rezki Yuniarti, S.Si., M.T.

Tugas ini masih jauh dari kesempurnaan, oleh karena itu kami mengharapkan segala saran dan kritik yang membangun dari semua pihak sangat kami harapkan demi perbaikan pada tugas selanjutnya. Harapan kami semoga tugas ini memberikan ilmu dan manfaat, khususnya bagi kami dan pembaca sekalian

Cimahi 1 Maret 2023

Kelompok 3

# DAFTAR ISI

[KATA PENGANTAR i](#_Toc139632976)

[DAFTAR ISI ii](#_Toc139632977)

[DAFTAR GAMBAR v](#_Toc139632978)

[BAB.I PENDAHULUAN 1](#_Toc139632979)

[I.1. Latar Belakang 1](#_Toc139632980)

[I.2. Rumusan Maslah 1](#_Toc139632981)

[I.3. Batasan Maslah 2](#_Toc139632982)

[I.4. Tujuan 2](#_Toc139632983)

[I.5. Manfaat 2](#_Toc139632984)

[BAB.II LANDASAN TEORI 3](#_Toc139632985)

[II.1. Manajement Penjualan 3](#_Toc139632986)

[II.2. Pemrograman Berorientasi objek 2 3](#_Toc139632987)

[II.3. Java 3](#_Toc139632988)

[BAB.III PERANCANGAN DAN UML 5](#_Toc139632989)

[III.1. ERD 5](#_Toc139632990)

[III.2. CLASS DIAGRAM 6](#_Toc139632991)

[BAB.IV HASIL DAN PEMBAHASAN 7](#_Toc139632992)

[IV.1. Halaman Utama 7](#_Toc139632993)

[IV.1.1. Screenshoot 7](#_Toc139632994)

[IV.1.2. Analisa 7](#_Toc139632995)

[IV.2. Halaman Login Admin 8](#_Toc139632996)

[IV.2.1. Screenshoot 8](#_Toc139632997)

[IV.2.2. Analisa 8](#_Toc139632998)

[IV.3. Halaman Daftar Menu 8](#_Toc139632999)

[IV.3.1. Screenshoot 8](#_Toc139633000)

[IV.3.2. Analisa 8](#_Toc139633001)

[IV.4. Halaman Daftar Barang 9](#_Toc139633002)

[IV.4.1. Screenshoot 9](#_Toc139633003)

[IV.4.2. Analisa 9](#_Toc139633004)

[IV.5. Halaman Daftar Petugas 9](#_Toc139633005)

[IV.5.1. Screenshoot 9](#_Toc139633006)

[IV.5.2. Analisa 10](#_Toc139633007)

[IV.6. Halaman Daftar Pelanggan 10](#_Toc139633008)

[IV.6.1. Screenshoot 10](#_Toc139633009)

[IV.6.2. Analisa 10](#_Toc139633010)

[IV.7. Halaman Daftar Distributor 10](#_Toc139633011)

[IV.7.1. Screenshoot 10](#_Toc139633012)

[IV.7.2. Analisa 11](#_Toc139633013)

[IV.8. Halaman Login Petugas 11](#_Toc139633014)

[IV.8.1. Screenshoot 11](#_Toc139633015)

[IV.8.2. Analisa 11](#_Toc139633016)

[IV.9. Halaman Menu Transaksi 12](#_Toc139633017)

[IV.9.1. Screenshoot 12](#_Toc139633018)

[IV.9.2. Analisa 12](#_Toc139633019)

[IV.10. Halaman Daftar Barang 13](#_Toc139633020)

[IV.10.1. Screenshoot 13](#_Toc139633021)

[IV.10.2. Analisa 13](#_Toc139633022)

[IV.11. Halaman Transaksi 13](#_Toc139633023)

[IV.11.1. Screenshoot 13](#_Toc139633024)

[IV.11.2. Analisa 14](#_Toc139633025)

[IV.12. Halaman Barang Masuk 14](#_Toc139633026)

[IV.12.1. Screenshoot 14](#_Toc139633027)

[IV.12.2. Analisa 14](#_Toc139633028)

[IV.13. Halaman Barang Keluar 14](#_Toc139633029)

[IV.13.1. Screenshoot 14](#_Toc139633030)

[IV.13.2. Analisa 15](#_Toc139633031)

[IV.14. Halaman Data Pelanggan 15](#_Toc139633032)

[IV.14.1. Screenshoot 15](#_Toc139633033)

[IV.14.2. Analisa 15](#_Toc139633034)

[IV.15. Halaman Data Daftar Menu 16](#_Toc139633035)

[IV.15.1. Screenshoot 16](#_Toc139633036)

[IV.15.2. Analisa 16](#_Toc139633037)

[IV.16. Halaman Stok Barang 16](#_Toc139633038)

[IV.16.1. Screenshoot 16](#_Toc139633039)

[IV.16.2. Analisa 16](#_Toc139633040)

[IV.17. Halaman Riwayat Transaksi 17](#_Toc139633041)

[IV.17.1. Screenshoot 17](#_Toc139633042)

[IV.17.2. Analisa 17](#_Toc139633043)

[IV.18. Halaman Daftar Barang Keluar 18](#_Toc139633044)

[IV.18.1. Screenshoot 18](#_Toc139633045)

[IV.18.2. Analisa 18](#_Toc139633046)

[IV.19. Halaman Daftar Barang Masuk 19](#_Toc139633047)

[IV.19.1. Screenshoot 19](#_Toc139633048)

[IV.19.2. Analisa 19](#_Toc139633049)

[BAB.V KESIMPULAN & SARAN 20](#_Toc139633050)

[V.1. KESIMPULAN 20](#_Toc139633051)

[V.2. B. SARAN 20](#_Toc139633052)

[BAB.VI DAFTAR PUSTAKA 21](#_Toc139633053)

# DAFTAR GAMBAR

[Gambar 1. 1 ERD 5](#_Toc139716388)

[Gambar 1. 2 CLASS DIAGRAM 6](#_Toc139716389)

[Gambar IV. 1 Halaman Utama 7](#_Toc139716398)

[Gambar IV. 2 Halaman Login Admin 8](#_Toc139716399)

[Gambar IV. 3 Halaman Daftar Menu 9](#_Toc139716400)

[Gambar IV. 4 Halaman Daftar Barang 9](#_Toc139716401)

[Gambar IV. 6 Halaman Daftar Petugas 10](#_Toc139716402)

[Gambar IV. 7 Halaman Daftar Pelanggan 12](#_Toc139716403)

[Gambar IV. 8 Halaman Daftar Distributor 13](#_Toc139716404)

[Gambar IV. 9 Halaman Login Petugas 14](#_Toc139716405)

[Gambar IV. 10 Halaman Menu Transaksi 15](#_Toc139716406)

[Gambar IV. 11 Daftar Barang 16](#_Toc139716407)

[Gambar IV. 12 Halaman Transaksi 17](#_Toc139716408)

[Gambar IV. 13 Halaman Barang Masuk 18](#_Toc139716409)

[Gambar IV. 14 Halaman Barang Keluar 19](#_Toc139716410)

[Gambar IV. 15 Halaman Data Pelanggan 21](#_Toc139716411)

[Gambar IV. 16 Data Daftar Menu 22](#_Toc139716412)

[Gambar IV. 17 Riwayat Trasaksi 24](#_Toc139716413)

[Gambar IV. 18 Halaman Daftar Barang Keluar 25](#_Toc139716414)

[Gambar IV. 19 Daftar Barang Masuk 26](#_Toc139716415)

# PENDAHULUAN

## Latar Belakang

Dalam era digitalisasi saat ini, toko mebel perlu memiliki aplikasi berbasis Javauntuk memudahkan pelanggan dalam berbelanja dan membantu pengelolaan stok barang. Dalam hal ini, pembuatan toko meubel berbasis JAVA adalah salah satu solusi karena JAVA merupakan bahasa pemograman yang banyak digunakan oleh para pengembang software. Dalam hal ini pengguna bahasa pemogramana Jva dan GUI dapat mempermudah pembuatan aplikasi toko meubel yang interaktif dan user-friendly.

Toko mebel ini merupakan sebuah bisnis yang berkembang di indonesia. Namun dalam hal pengelolaan sebuah toko mebel ini yang masih manual yang dimana menjadi kendala bagi pemilik sebuah toko mebel itu sendiri. Pengelolaan toko mebel secara manual menjadi kendala bagi pemilik toko, sehingga dibutuhkan aplikasi yang dapat membantu mengelola toko dengan lebih efektif dan efisien.

Kebutuhan pelanggan akan pengalaman belanja lebih efesien, dalam era digital, pengalaman belanja tidak hanya sebatas membeli produk, tetapi juga bagaimana cara konsumen memperoleh informasi, memilih produk, melakukan pembayaran, dan mendapatkan layanan purna jual. Aplikasi toko meubel datap membantu meningkakan pengalaman belanja konsumen dengan menyediakan fitur-fitur yang akan memudahkan konsumen dalam melakukan sebuah transaksi.

Aplikasi toko mebel ini akan menyediakan fitur-fitur seperti data barang, data petugas, data pelanggan, daftar barang, transaksi dan riwayat.Dengan demikian, pembuatan aplikasi toko mebel berbasis Java dapat menjadi solusi yang efektif dalam menghadapi tantangan pengelolaan toko mebel secara manual, serta memenuhi kebutuhan pelanggan akan pengalaman belanja yang efisien dan menyenangkan

## Rumusan Maslah

1. Bagaimana cara untuk login admin dan login petugas dalam aplikasi toko mebeul ini?
2. Bagaimana proses transaksi dalam aplikasi toko meubel?
3. Bagaimana cara mengimplementasikan fitur-fitur dalam aplikasi toko meubel ini?

## Batasan Maslah

1. Fokus pada pengelolaan stok barang dan pencatatan keuangan pada toko meubel.
2. Fitur pemesanan produk hanya ditampilkan pada aplikasi, dan proses pengirim dan pengambilan produk dilakukan secara manual oleh pemilik toko meubel.
3. Aplikasi tidak mencangkup fitur manajeman karyawan atau manajemen produksi karena fokus pada pengelolan stok barang dan pencatatan keuangan.
4. Aplikasi hanya dibuat dengan pemograman java dan GUI.

## Tujuan

Tujuan dibuatnya aplikasi toko ini dimana akan membantu pengeloaan toko mebel lebih efisien dan efektif. Membuat sebuah aplikasi ini akan membantu pemilik toko mebel dalam megelola data produk, pelanggan, transaksi, dan iventaris dengan mudah dan cepat.

## Manfaat

1. Menghemat waktu dan tenaga dalam pengeloaan toko mebel
2. Meningkatkan efisiensi dan efektivitas pengeloaan toko mebel berbasis java dan GUI, proses transaksi dan pengelolan stok barang dapat dilakuak dengan efesien.
3. Memudahkan pengguna melihat informasi produk seperti harga, deskripsi, dan lain-lain.
4. Memudahkan pemilik toko dalam pengelolaan stok produk meubel.

# LANDASAN TEORI

## Manajement Penjualan

Pembuatan aplikasi manajement penjualan untuk toko mebeul meliputi beberapa teori dasar seperti:

Manajemen produk (product management) adalah proses pengelolaan produk dari konsepsi hingga penarikan dari pasar. Ini melibatkan perencanaan, pengembangan, pemasaran, dan pengelolaan siklus hidup produk untuk memastikan produk yang sukses dan memenuhi kebutuhan pelanggan serta tujuan bisnis perusahaan.

Sistem basis data (database system) adalah sebuah perangkat lunak yang dirancang untuk mengelola, menyimpan, mengatur, dan mengambil data dalam suatu perusahaan atau organisasi. Sistem basis data memungkinkan pengguna untuk menyimpan, mengakses, dan memanipulasi data dengan efisien dan terstruktur.

UI (User Interface) design adalah proses merancang antarmuka yang digunakan oleh pengguna untuk berinteraksi dengan suatu produk atau sistem. Desain UI melibatkan pemilihan elemen visual seperti layout, warna, ikon, font, dan tata letak yang dirancang untuk menciptakan pengalaman pengguna yang baik dan intuitif.

## Pemrograman Berorientasi objek 2

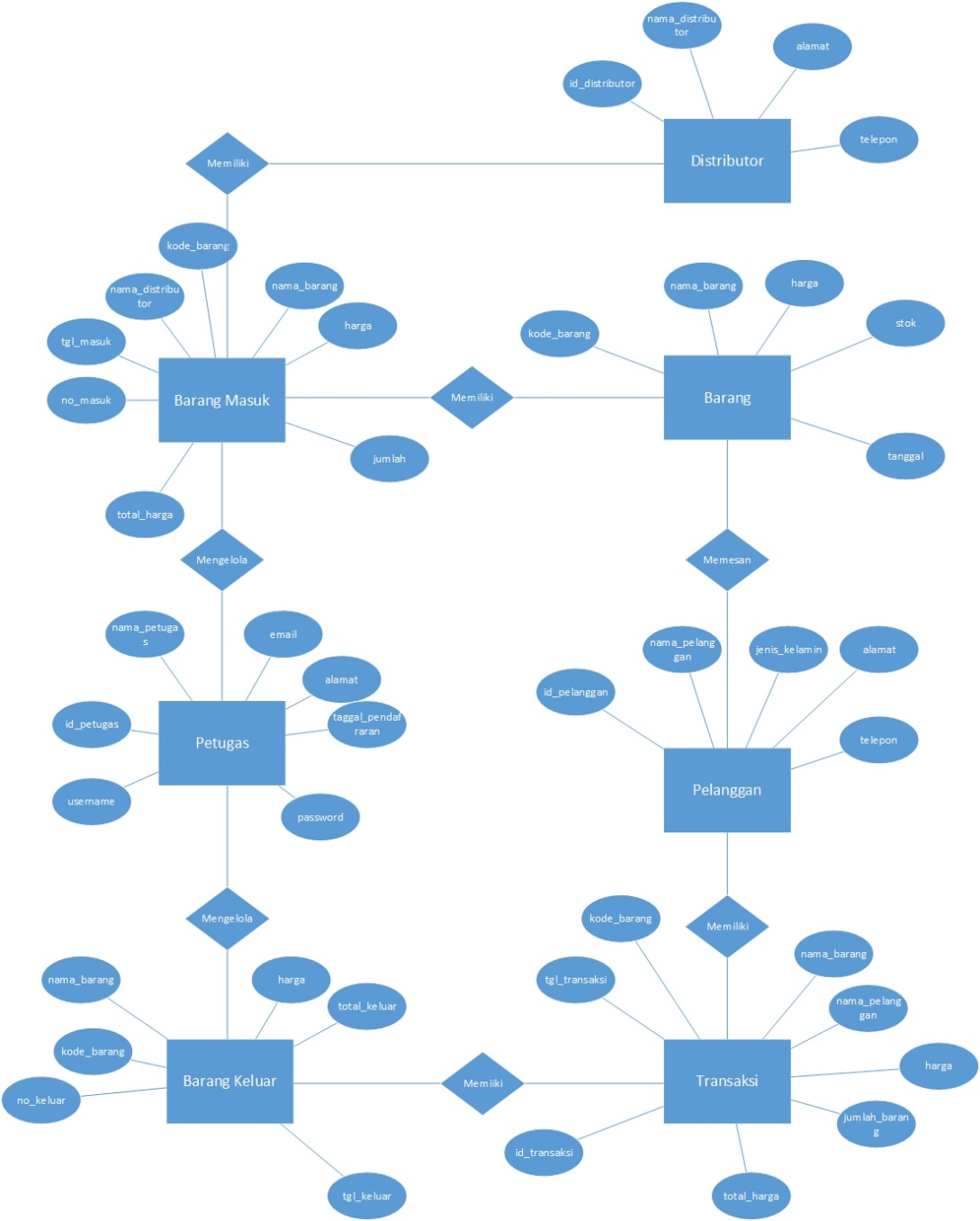
OOP 2 adalah lanjutan dari konsep Pemrograman Berorientasi Objek (OOP) yang melibatkan konsep-konsep seperti warisan, polimorfisme, enkapsulasi, antarmuka, dan abstraksi. Dalam OOP 2, Anda mempelajari cara menggunakan warisan untuk mewarisi sifat dan perilaku dari kelas yang sudah ada, memanfaatkan polimorfisme untuk menggunakan objek dengan cara yang berbeda, mengaplikasikan enkapsulasi untuk menyembunyikan detail implementasi, menggunakan antarmuka sebagai kontrak yang menentukan metode yang harus diimplementasikan, dan menggunakan abstraksi untuk menyederhanakan kompleksitas dan fokus pada aspek penting dari objek atau konsep. Dengan menerapkan konsep-konsep ini, Anda dapat mengorganisir kode dengan lebih baik, meningkatkan fleksibilitas, dan mempermudah pengembangan aplikasi yang kompleks.

## Java

Java adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Sun Microsystems (sekarang Oracle) pada tahun 1995. Dengan prinsip utama seperti portabilitas, keamanan, dan kemudahan penggunaan, Java telah menjadi salah satu bahasa pemrograman paling populer di dunia. Fitur-fitur penting dalam Java termasuk platform mandiri dengan JVM, sintaks terstruktur yang mirip dengan bahasa C/C++, pemrograman berorientasi objek yang kuat, keamanan yang terintegrasi, manajemen memori otomatis, dan pustaka kelas yang kaya. Java digunakan secara luas dalam pengembangan perangkat lunak desktop, web, permainan, aplikasi seluler, dan server.

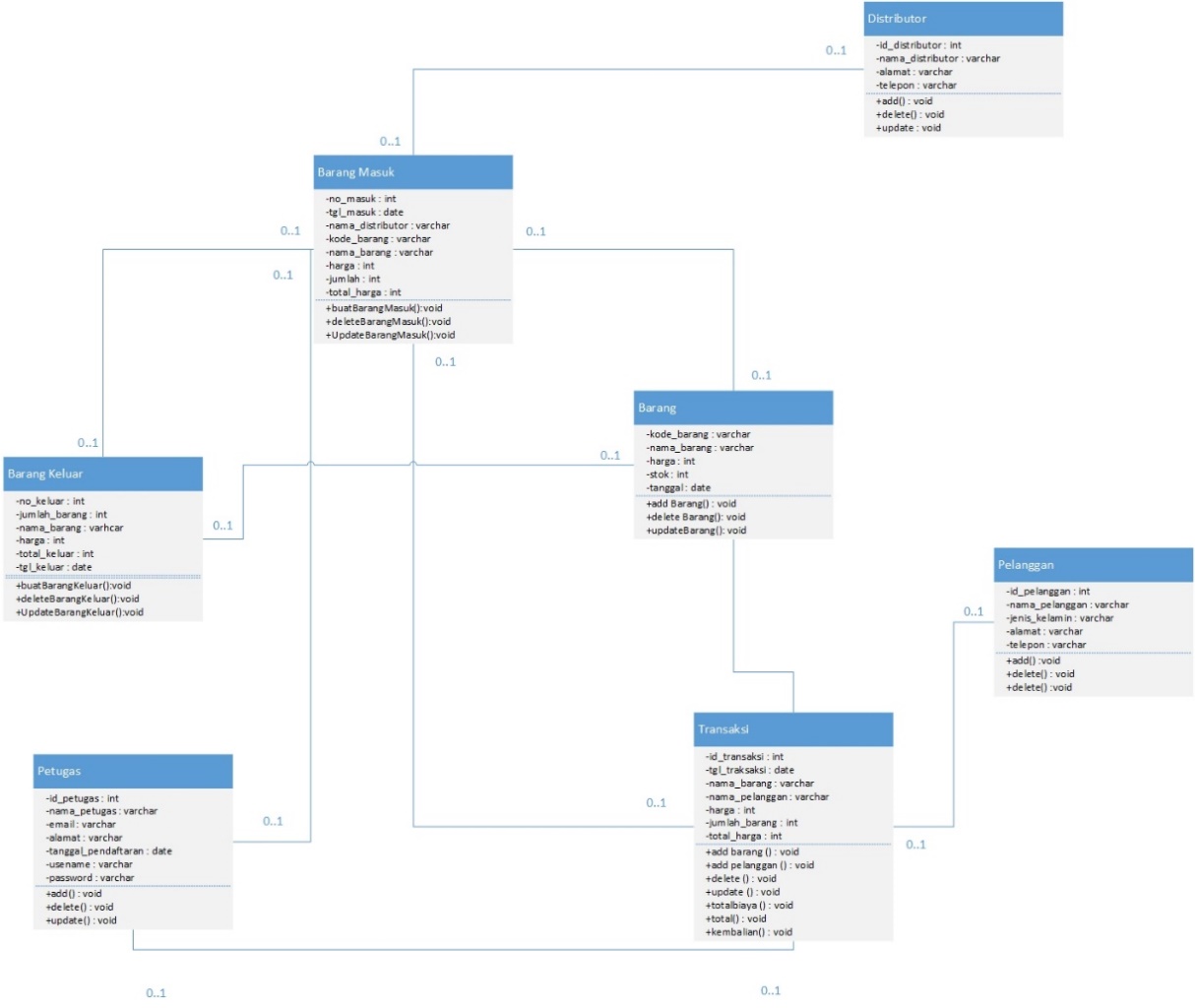
# PERANCANGAN DAN UML

## ERD



Gambar 1. 1 ERD

## CLASS DIAGRAM



Gambar 1. 2 CLASS DIAGRAM

# HASIL DAN PEMBAHASAN

## Halaman Utama

### Screenshoot



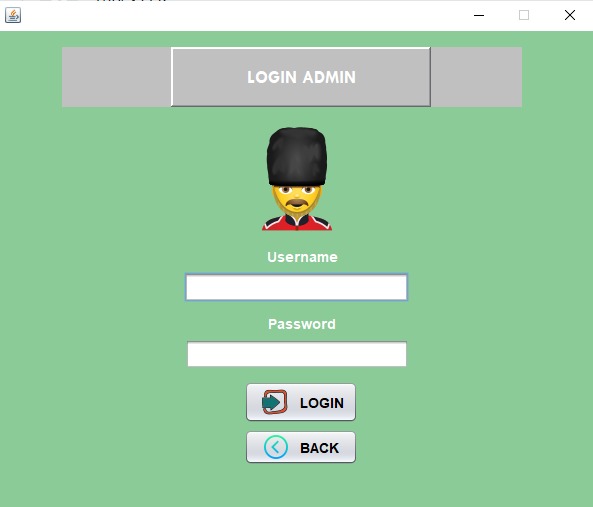
Gambar IV. 1 Halaman Utama

### Analisa

Halaman ini dimana new login\_admin().setVisible(true);: Baris kode ini menciptakan objek dari kelas login\_admin dan mengatur visibilitasnya menjadi true, yang berarti antarmuka pengguna (UI) akan ditampilkan. Ini digunakan untuk membuat jendela atau frame login\_admin muncul di layar. Dan untuk Halaman Login Petugas sama Login new login\_petugas().setVisible(true); yang dimana membuat sebuat jendela menuju ke halaman login petugas,

## Halaman Login Admin

### Screenshoot



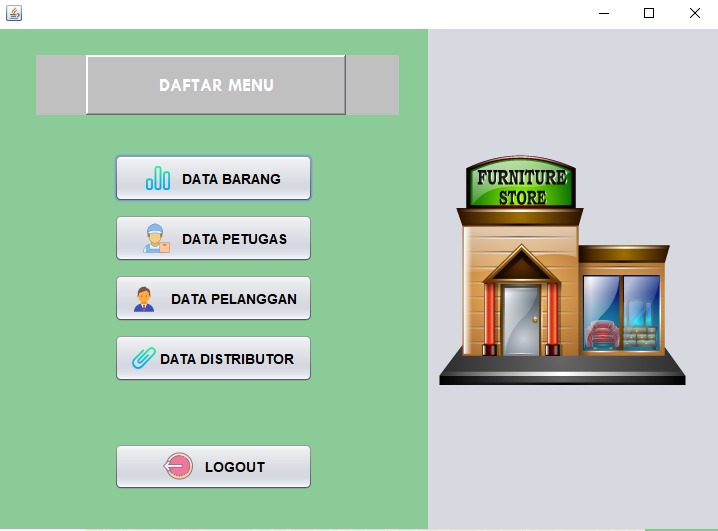
Gambar IV. 2 Halaman Login Admin

### Analisa

Metode `login()` digunakan untuk otentikasi login admin. Program terhubung ke database menggunakan objek `koneksi`. Query SQL dibuat untuk memeriksa kecocokan antara username dan password yang dimasukkan dengan data di tabel `admin`. Jika cocok, program menampilkan "Login Success" dan membuka jendela `menu\_admin`. Jika tidak cocok, program menampilkan "Username atau Password Salah" dan mengosongkan teks pada `txt\_username` dan `txt\_password`. Penanganan pengecualian dilakukan dengan blok `catch` untuk menangkap kesalahan pada eksekusi query atau saat berinteraksi dengan antarmuka pengguna. Metode ini memastikan proses login admin berjalan dengan lancar dan memberikan respons yang sesuai.

## Halaman Daftar Menu

### Screenshoot



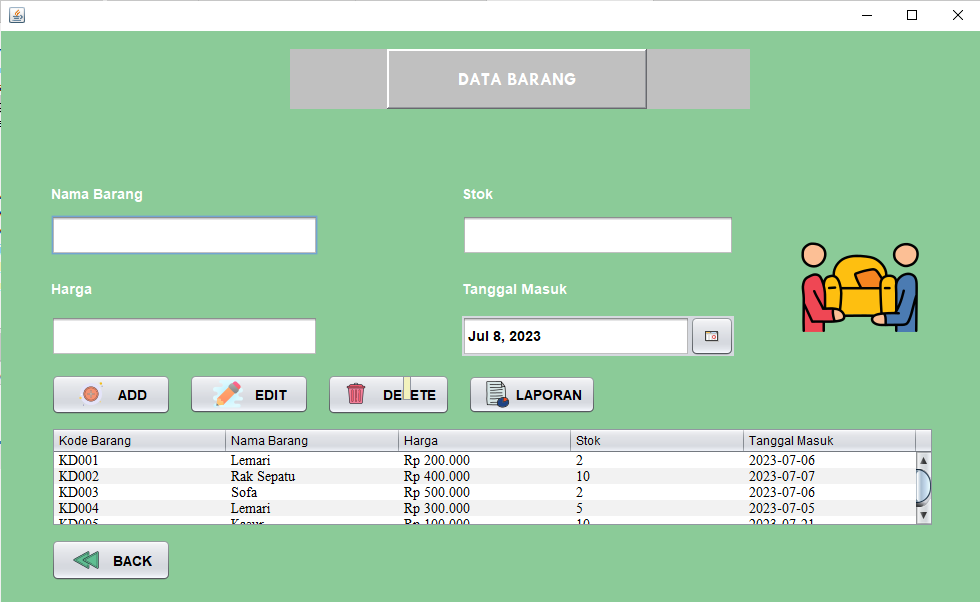
Gambar IV. 3 Halaman Daftar Menu

### Analisa

Halaman ini dimana menggunkan code seperti pada tampilan awal yang dimana menggunkan missal. new formCRUDBarang().setVisible(true); dispose(); dan seterusnya untuk yang lainya sama .

## Halaman Daftar Barang

### Screenshoot



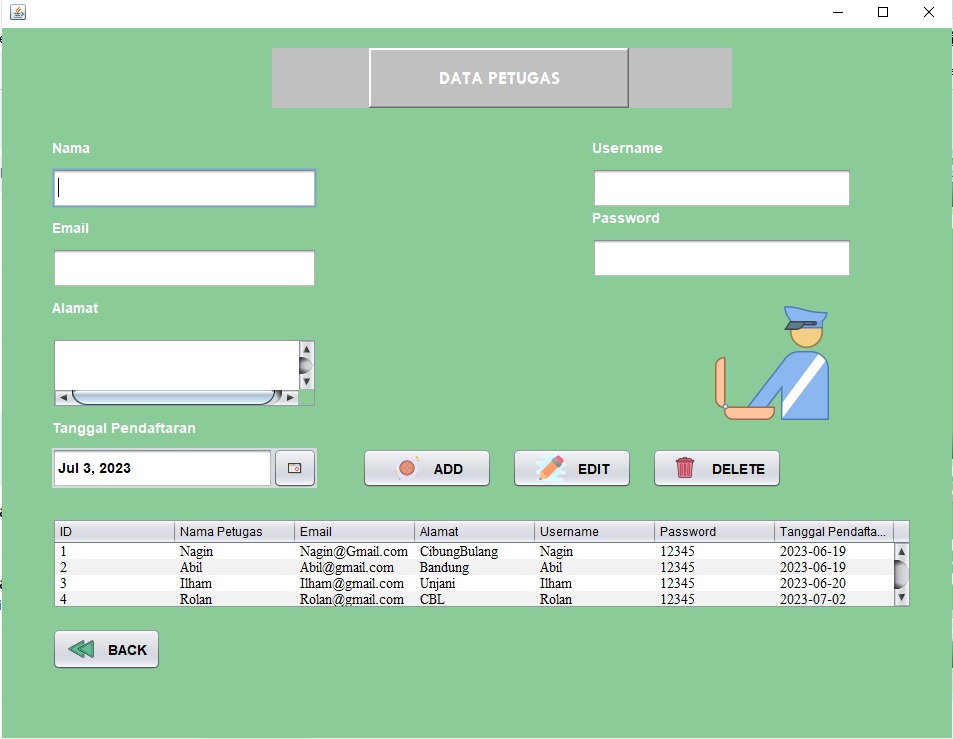
Gambar IV. 4 Halaman Daftar Barang

### Analisa

Kelas `formCRUDBarang` merupakan jendela atau frame dalam aplikasi yang digunakan untuk melakukan operasi CRUD pada data barang. Program ini terhubung ke database menggunakan objek `koneksi` dan menginisialisasi komponen antarmuka pengguna. Metode `tampilData()` digunakan untuk menampilkan data barang pada tabel dengan menjalankan query SELECT dan memasukkan hasilnya ke dalam tabel. Metode `tambahData()` digunakan untuk menambahkan data barang ke dalam tabel dengan menjalankan query INSERT. Metode `hapusData()` digunakan untuk menghapus data barang dari tabel dengan menjalankan query DELETE dan mengatur ulang kode barang pada data lain. Metode `editData()` digunakan untuk mengedit data barang pada tabel dengan menjalankan query UPDATE. Program juga memiliki metode `clear()` untuk mengosongkan komponen antarmuka pengguna setelah operasi selesai dilakukan. Keseluruhan, program ini memungkinkan pengguna untuk menampilkan, menambahkan, menghapus, dan mengedit data barang pada tabel dengan interaksi antarmuka pengguna yang disediakan.

## Halaman Daftar Petugas

### Screenshoot



Gambar IV. 6 Halaman Daftar Petugas

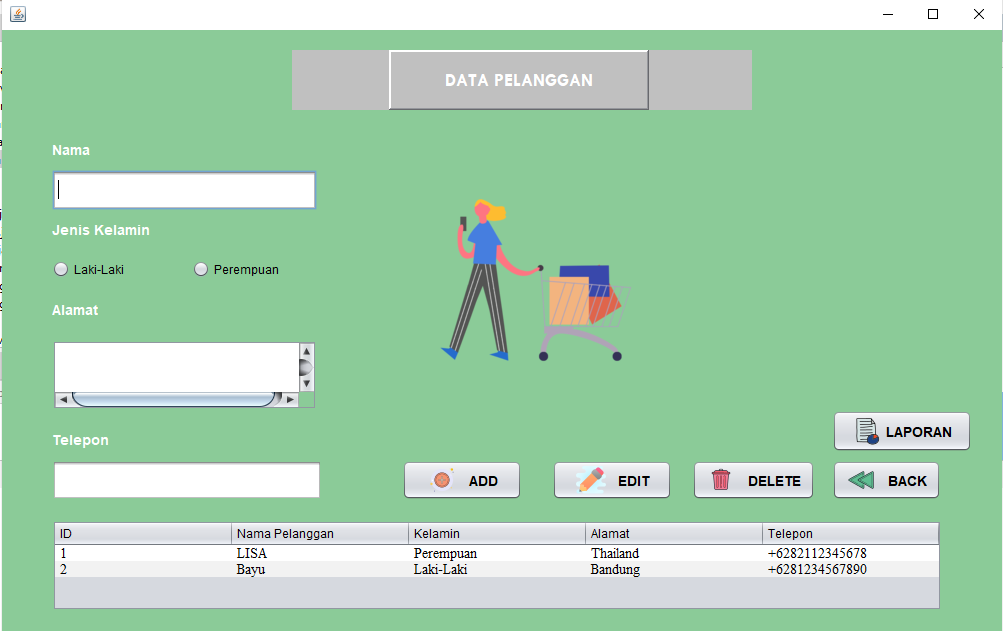
### Analisa

Program `akun\_petugas` merupakan sebuah kelas yang digunakan untuk mengelola akun petugas. Program ini memiliki beberapa metode penting, seperti `tampilData()`, `clear()`, `tambahData()`, `hapusData()`, dan `editData()`. Metode `tampilData()` digunakan untuk menampilkan data akun petugas pada tabel. Data tersebut diambil dari tabel `tb\_datapetugas` dalam database. Metode ini juga mengatur ulang model tabel untuk menampilkan data yang telah diperoleh Metode `clear()` digunakan untuk mengosongkan nilai pada komponen UI, seperti kolom input atau area teks.

Metode `tambahData()` digunakan untuk menambahkan data akun petugas baru ke dalam tabel. Data yang diinput melalui komponen UI akan disimpan dalam database melalui query INSERT. Sebelumnya, metode ini juga melakukan validasi untuk memastikan username dan password tidak sama, serta memeriksa ketersediaan username yang akan ditambahkan.Metode `hapusData()` digunakan untuk menghapus data akun petugas dari tabel. Data yang dihapus ditentukan berdasarkan ID petugas yang dipilih pada tabel. Setelah penghapusan, metode ini juga mengatur ulang ID petugas pada data lain yang memiliki ID lebih besar.Metode `editData()` digunakan untuk mengedit data akun petugas yang telah ada di tabel. Data yang diedit ditentukan berdasarkan ID petugas yang dipilih pada tabel. Metode ini melakukan update data sesuai dengan nilai yang diinput melalui komponen UI. Program `akun\_petugas` ini sangat berguna dalam mengelola data akun petugas secara efisien.

## Halaman Daftar Pelanggan

### Screenshoot



Gambar IV. 7 Halaman Daftar Pelanggan

### Analisa

Program `pelanggan` adalah sebuah kelas yang digunakan untuk mengelola data pelanggan dengan efisien. Dalam program ini, terdapat metode `tampilData()`, `clear()`, `tambahData()`, `hapusData()`, dan `editData()` yang berperan penting dalam pengelolaan data.Metode `tampilData()` menampilkan data pelanggan dari tabel `tb\_pelanggan` dalam database dan mengatur tampilan tabel pada antarmuka pengguna.Metode `clear()` menghapus nilai pada komponen antarmuka pengguna agar dapat memasukkan data baru dengan mudah.

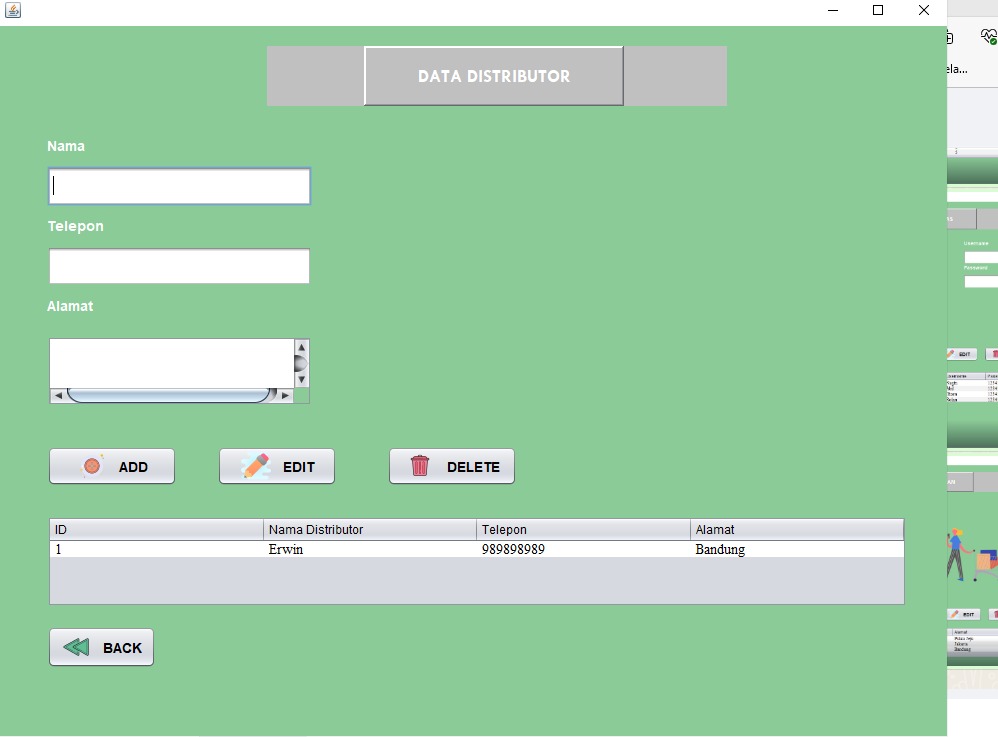
Metode tambahData() digunakan untuk menambahkan data pelanggan ke dalam tabel tb\_pelanggan. Data yang diambil meliputi kode barang, nama, jenis kelamin, alamat, dan nomor telepon. Sebelum memasukkan data, nomor telepon akan divalidasi menggunakan regex untuk memastikan format yang valid.

Metode `hapusData()` menghapus data pelanggan dari tabel berdasarkan ID pelanggan yang dipilih. Setelah penghapusan, metode ini mengatur ulang ID pelanggan pada data lain yang memiliki ID lebih besar.Metode `editData()` mengedit data pelanggan yang sudah ada dalam tabel berdasarkan ID pelanggan yang dipilih. Data diperbarui sesuai dengan input yang diberikan melalui antarmuka pengguna.

Dengan program `pelanggan`, pengelolaan data pelanggan menjadi lebih efisien dan terstruktur.

## Halaman Daftar Distributor

### Screenshoot



Gambar IV. 8 Halaman Daftar Distributor

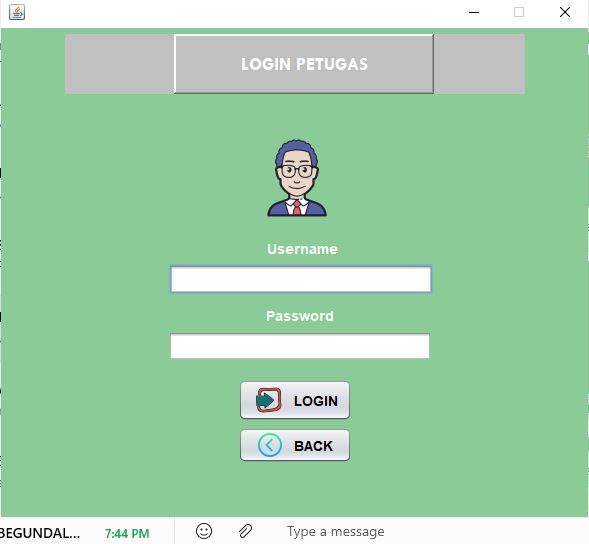
### Analisa

Program `data\_distributor` merupakan sebuah kelas yang digunakan untuk mengelola data distributor. Program ini memiliki fungsi utama yaitu menampilkan data distributor, menambah data distributor baru, menghapus data distributor, dan mengedit data distributor yang sudah ada. Dalam metode `tampilData()`, program akan mengambil data dari tabel `distributor` dalam database dan menampilkannya pada tabel dalam antarmuka pengguna. Metode `clear()` digunakan untuk mengosongkan nilai pada komponen antarmuka pengguna, sehingga memudahkan pengguna untuk memasukkan data baru. Metode `tambahData()` akan menambahkan data distributor baru ke dalam tabel. Data tersebut akan diambil dari input pengguna melalui antarmuka pengguna dan disimpan dalam database melalui query INSERT. Metode `hapusData()` digunakan untuk menghapus data distributor dari tabel berdasarkan ID distributor yang dipilih oleh pengguna. Setelah penghapusan, ID distributor lain yang memiliki nilai lebih besar akan diatur ulang. Metode `editData()` berfungsi untuk mengedit data distributor yang sudah ada di tabel. Data yang diedit ditentukan berdasarkan ID distributor yang dipilih oleh pengguna melalui antarmuka pengguna.

Program `data\_distributor` ini sangat berguna dalam mengelola data distributor dengan efisien dan terstruktur.

## Halaman Login Petugas

### Screenshoot



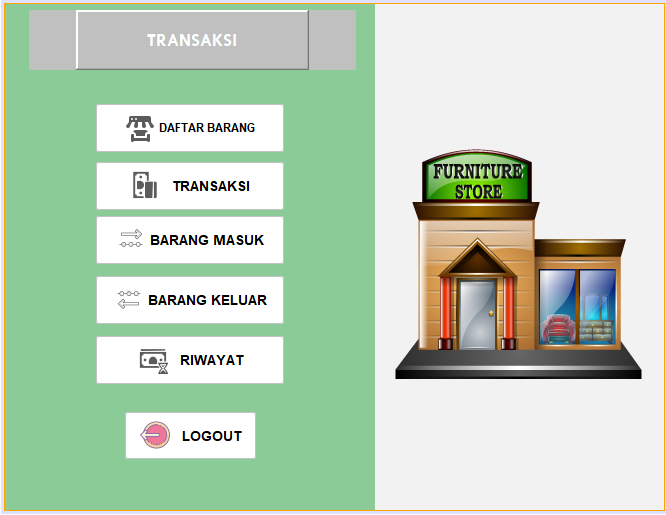
Gambar IV. 9 Halaman Login Petugas

### Analisa

Aplikasi ini merupakan metode `login()` yang digunakan untuk melakukan proses otentikasi pengguna. Aplikasi ini terhubung ke database dan memeriksa apakah kombinasi username dan password yang dimasukkan oleh pengguna cocok dengan data yang ada di tabel `tb\_datapetugas`.Pada metode `login()`, aplikasi terlebih dahulu mendapatkan koneksi ke database melalui objek `koneksi`. Kemudian, dilakukan query untuk mencocokkan username dan password yang dimasukkan oleh pengguna dengan data yang ada di tabel `tb\_datapetugas`. Jika username dan password cocok, maka aplikasi akan menampilkan pesan "Login Success" dan membuka halaman `menu\_user`, serta menutup halaman login yang sebelumnya.Namun, jika username atau password tidak cocok, maka aplikasi akan menampilkan pesan "Username atau Password Salah" dan mengosongkan kolom input username dan password.Aplikasi ini digunakan untuk mengautentikasi pengguna dan memberikan akses ke halaman `menu\_user` jika login berhasil.

## Halaman Menu Transaksi

### Screenshoot



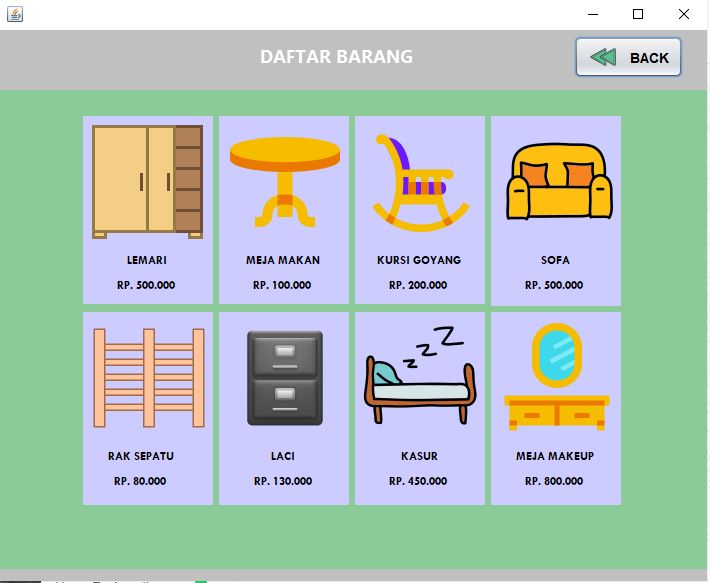
Gambar IV. 10 Halaman Menu Transaksi

### Analisa

Halaman ini dimana menggunkan code seperti pada tampilan awal yang dimana menggunkan missal. new riwayat\_transaksi().setVisible(true); dispose(); untuk yang lainya sama

## Halaman Daftar Barang

### Screenshoot



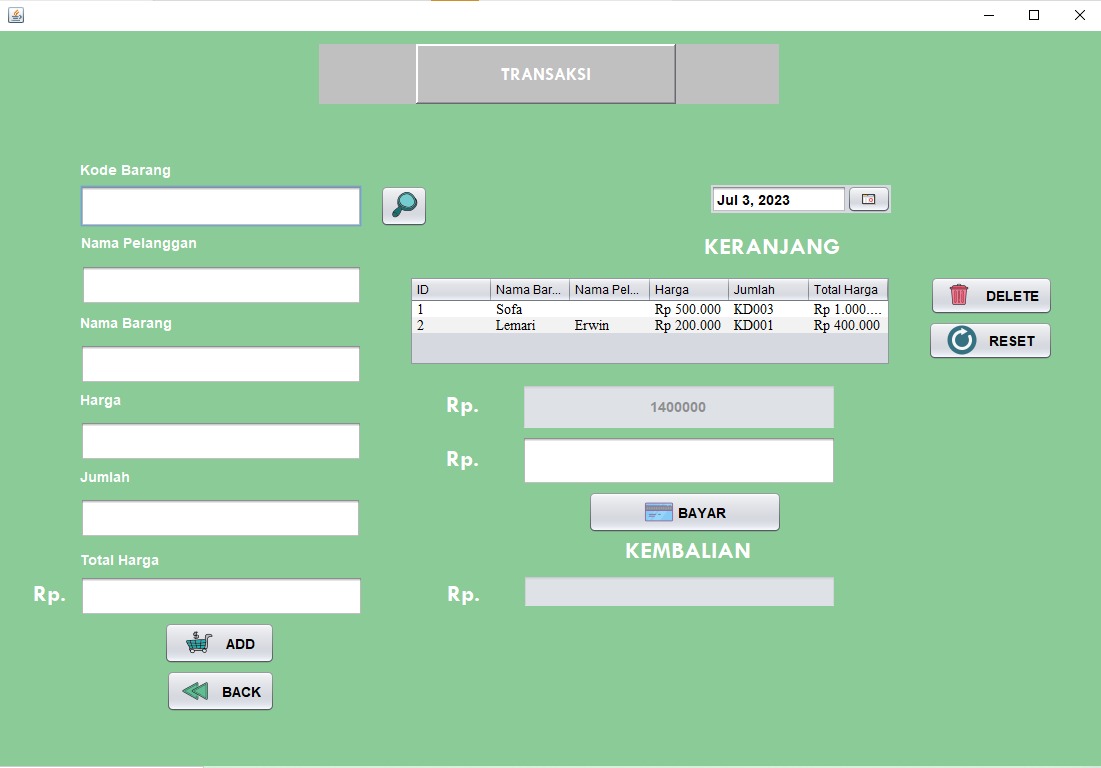
Gambar IV. 11 Daftar Barang

### Analisa

Untuk di halaman ini dimana hanya menampilkan barang apa aja yang ada di aplikasi kamai denggan menggunkan panel dan icon.

## Halaman Transaksi

### Screenshoot



Gambar IV. 12 Halaman Transaksi

### Analisa

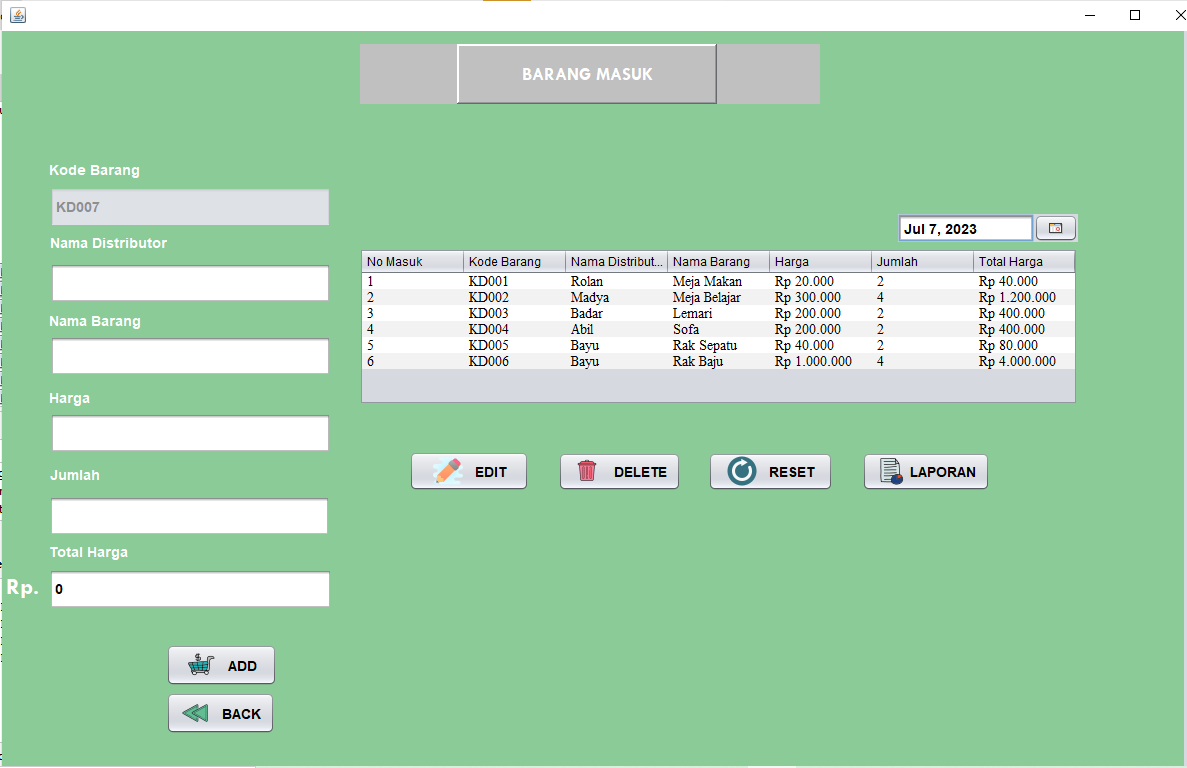
Program `Transaksi` merupakan sebuah kelas yang digunakan untuk mengelola proses transaksi penjualan barang. Program ini memiliki beberapa metode penting, seperti `tanggal()`, `tampilData()`, `clear()`, `keranjang()`, `hapusData()`, `totalBiaya()`, `total()`, `reset()`, dan `kembalian()`.Metode `tanggal()` digunakan untuk mengambil tanggal saat ini dan menampilkannya pada komponen UI yang bertipe JDateChooser.Metode `tampilData()` digunakan untuk menampilkan data transaksi pada tabel. Data tersebut diambil dari tabel `transaksi` dalam database.

Metode ini juga mengatur ulang model tabel untuk menampilkan data yang telah diperoleh.Metode `clear()` digunakan untuk mengosongkan nilai pada komponen UI, seperti kolom input atau area teks.Metode `keranjang()` digunakan untuk menambahkan data transaksi baru ke dalam tabel. Data yang diinput melalui komponen UI akan disimpan dalam database melalui query INSERT. Metode ini juga melakukan validasi dan perhitungan total harga.Metode `hapusData()` digunakan untuk menghapus data transaksi dari tabel. Data yang dihapus ditentukan berdasarkan ID transaksi yang dipilih pada tabel. Setelah penghapusan, metode ini juga mengatur ulang ID transaksi pada data lain yang memiliki ID lebih besar.Metode `totalBiaya()` digunakan untuk menghitung total biaya transaksi berdasarkan harga dan jumlah barang yang ada dalam tabel transaksi.

Metode `total()` digunakan untuk menghitung total harga barang berdasarkan harga dan jumlah yang diinput melalui komponen UI.Metode `reset()` digunakan untuk mengosongkan nilai pada kolom input uang.Metode `kembalian()` digunakan untuk menghitung kembalian dari transaksi berdasarkan total harga dan jumlah uang yang diinput melalui komponen UI.Program `Transaksi` ini digunakan untuk mengelola proses transaksi penjualan barang dengan efisien dan akurat.

## Halaman Barang Masuk

### Screenshoot



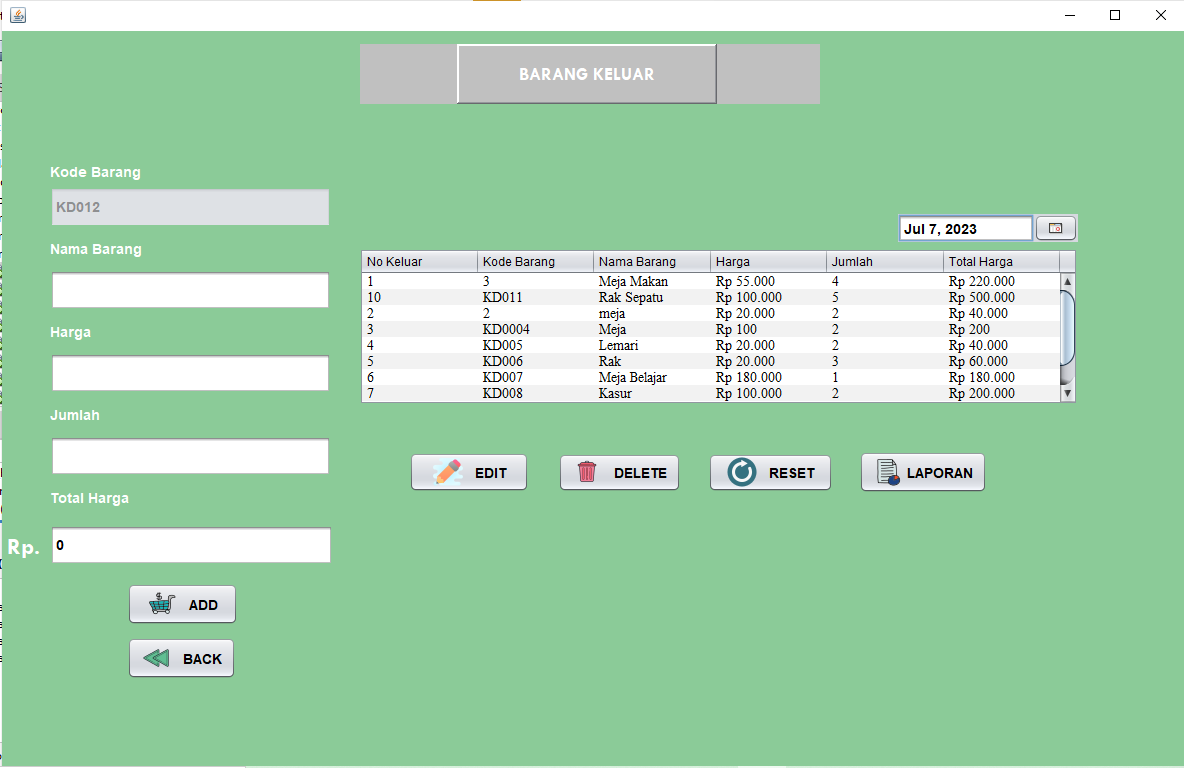
Gambar IV. 13 Halaman Barang Masuk

### Analisa

Program `Barang\_Masuk` adalah kelas yang mengelola proses penerimaan barang masuk. Metode penting dalam program ini termasuk `tanggal()`, `auto()`, `tampilData()`, `clear()`, `keranjang()`, `hapusData()`, `editData()`, `totalBiaya()`, dan `total()`. Metode `tanggal()` digunakan untuk menampilkan tanggal saat ini pada komponen JDateChooser. Metode `auto()` menghasilkan kode barang secara otomatis berdasarkan kode terakhir yang ada dalam database. Metode `tampilData()` menampilkan data penerimaan barang pada tabel dengan mengambil data dari tabel `barang\_masuk` dan mengatur ulang model tabel. Metode `clear()` mengosongkan nilai pada komponen UI. Metode `keranjang()` menambahkan data penerimaan barang baru ke dalam tabel dan menyimpannya dalam database melalui query INSERT. Metode ini juga melakukan validasi dan menghitung total harga. Metode `hapusData()` menghapus data penerimaan barang dari tabel berdasarkan nomor masuk yang dipilih dan mengatur ulang nomor masuk pada data lain. Metode `editData()` mengedit data penerimaan barang berdasarkan nomor masuk yang dipilih dan nilai yang diinput melalui komponen UI. Metode `totalBiaya()` menghitung total biaya penerimaan barang berdasarkan harga dan jumlah barang dalam tabel. Metode `total()` menghitung total harga barang berdasarkan nilai yang diinput melalui komponen UI. Program `Barang\_Masuk` memfasilitasi proses penerimaan barang masuk dengan efisiensi dan keakuratan.

## Halaman Barang Keluar

### Screenshoot



Gambar IV. 14 Halaman Barang Keluar

### Analisa

`Barang\_Keluar` adalah kelas yang digunakan untuk mengelola transaksi barang keluar. Berikut adalah penjelasan singkat mengenai setiap metode yang ada dalam kelas:`Barang\_Keluar()`: Konstruktor kelas `Barang\_Keluar` yang digunakan untuk menginisialisasi komponen-komponen GUI, menghubungkan ke database, mengatur model tabel, dan menampilkan data.`tanggal()`: Metode untuk mengatur tanggal pada komponen `txt\_tanggal` dengan menggunakan tanggal saat ini.auto()`: Metode untuk menghasilkan kode barang otomatis dengan mengambil kode barang terakhir dari database dan menambahkannya.

`tampilData()`: Metode untuk menampilkan data barang keluar dari database ke dalam tabel.`clear()`: Metode untuk menghapus teks pada komponen input.`keranjang()`: Metode untuk memasukkan data barang keluar ke dalam keranjang. Data diambil dari komponen input, termasuk tanggal. Metode ini juga menghitung total harga.`hapusData()`: Metode untuk menghapus data barang keluar dari keranjang berdasarkan baris yang dipilih pada tabel.`editData()`: Metode untuk mengedit data barang keluar pada keranjang. Data diambil dari komponen input, termasuk tanggal.`totalBiaya()`: Metode untuk menghitung total biaya berdasarkan harga dan jumlah barang yang diinput. Hasilnya ditampilkan pada komponen `txt\_totalharga`. `total()`: Metode untuk menghitung total harga barang berdasarkan harga dan jumlah yang diinput. Hasilnya ditampilkan pada komponen `txt\_totalharga`.

## Halaman Data Pelanggan

### Screenshoot



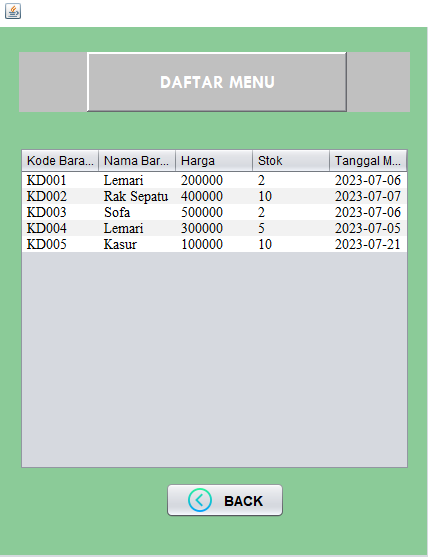
Gambar IV. 15 Halaman Data Pelanggan

### Analisa

tampilData() adalah metode yang digunakan untuk menampilkan data pelanggan dari tabel tb\_pelanggan ke dalam table\_pelanggan. Berikut adalah penjelasan singkat untuk metode ini:Pertama, metode ini menghapus semua baris yang ada dalam table\_pelanggan dengan menggunakan perulangan dan removeRow().Kemudian, metode ini menjalankan query untuk mengambil data dari tabel tb\_pelanggan.Setiap baris hasil query diproses dalam perulangan while. Data dari setiap baris diambil menggunakan getString() dan disimpan dalam variabel-variabel sementara.Data yang diambil dari setiap baris disimpan dalam array data.Setelah perulangan selesai, array data ditambahkan sebagai baris baru dalam table\_pelanggan menggunakan addRow().Terakhir, table\_pelanggan diatur agar menampilkan nilai-nilai yang telah ditambahkan dengan menggunakan setModel(table).

## Halaman Data Daftar Menu

### Screenshoot



Gambar IV. 16 Data Daftar Menu

### Analisa

Untuk harga disni kenpa masih belum decimal karena, form ini untuk form transaksi dimana nanti yang di panggil dari daftar menu ke daftar transaksi kode barang. Nama barang, dan harga . ketika sya mencoba mengubah harga menjadi decimal dan di panggil di form transaksi terjadi eror.

tampilData() adalah metode yang digunakan untuk menampilkan data barang dari tabel tb\_databarang ke dalam table\_barang1. Berikut adalah penjelasan singkat untuk metode ini:Pertama, metode ini menghapus semua baris yang ada dalam table\_barang1 dengan menggunakan perulangan dan removeRow().Kemudian, metode ini menjalankan query untuk mengambil data dari tabel tb\_databarang.Setiap baris hasil query diproses dalam perulangan while. Data dari setiap baris diambil menggunakan getString() dan disimpan dalam variabel-variabel sementara.Data yang diambil dari setiap baris disimpan dalam array data.Setelah perulangan selesai, array data ditambahkan sebagai baris baru dalam table\_barang1 menggunakan addRow().Terakhir, table\_barang1 diatur agar menampilkan nilai-nilai yang telah ditambahkan dengan menggunakan setModel(table).

## Halaman Stok Barang

### Screenshoot

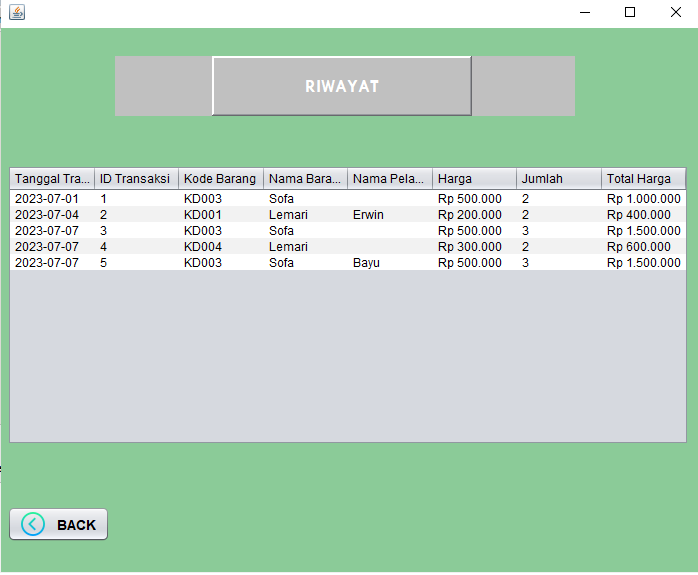


### Analisa

Sama seperti program IV.15 dimana untuk daftar menu hanya saja ini untuk laporanya dan untuk yang sebelumnya untuk form traksaksi . untuk di sini untuk harga sudah menggunakan decimal.

## Halaman Riwayat Transaksi

### Screenshoot



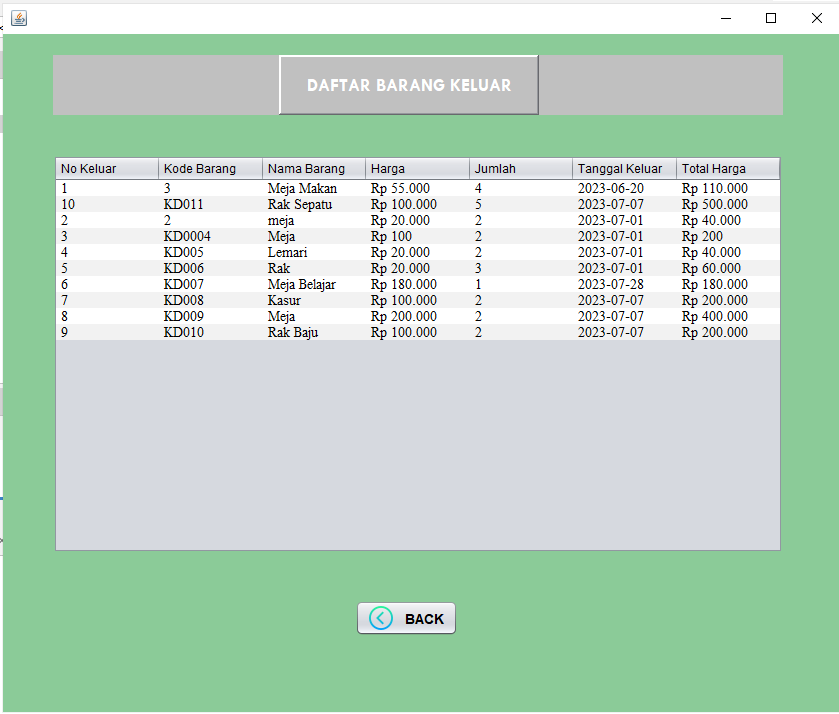
Gambar IV. 17 Riwayat Trasaksi

### Analisa

riwayat\_transaksi adalah kelas yang digunakan untuk menampilkan riwayat transaksi dalam sebuah tabel. Berikut adalah penjelasan singkat untuk kelas dan metode yang ada di dalamnya:Konstruktor riwayat\_transaksi() adalah metode yang dipanggil saat objek dari kelas riwayat\_transaksi dibuat. Metode ini melakukan inisialisasi komponen GUI dan memanggil metode koneksi.getKoneksi() untuk mengatur koneksi ke database. Selain itu, metode ini juga mengatur model tabel (table) dan menambahkan kolom-kolom yang diperlukan.Metode tampilData() digunakan untuk menampilkan data riwayat transaksi ke dalam tabel. Pertama, metode ini menghapus semua baris yang ada dalam tb\_riwayat dengan menggunakan perulangan dan removeRow(). Kemudian, metode ini menjalankan query untuk mengambil data dari tabel transaksi. Setiap baris hasil query diproses dalam perulangan while. Data dari setiap baris diambil menggunakan getString() dan disimpan dalam variabel-variabel sementara. Data yang diambil dari setiap baris disimpan dalam array data. Selanjutnya, array data ditambahkan sebagai baris baru dalam table menggunakan addRow(). Terakhir, tb\_riwayat diatur agar menampilkan nilai-nilai yang telah ditambahkan dengan menggunakan setModel(table).

## Halaman Daftar Barang Keluar

### Screenshoot



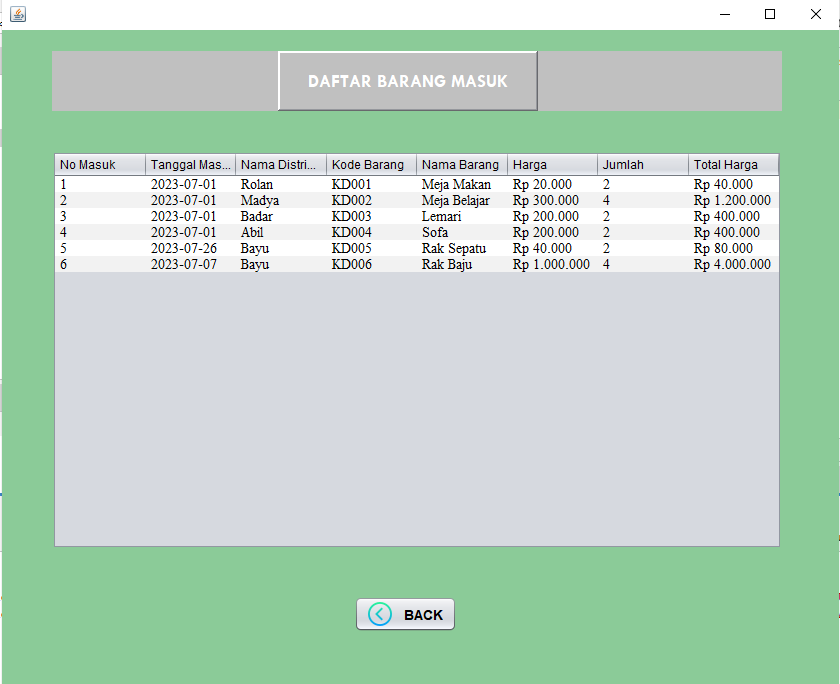
Gambar IV. 18 Halaman Daftar Barang Keluar

### Analisa

Laporan\_barang\_keluar adalah kelas yang digunakan untuk menampilkan laporan barang keluar dalam sebuah tabel. Berikut adalah penjelasan singkat untuk kelas dan metode yang ada di dalamnya:Konstruktor Laporan\_barang\_keluar() adalah metode yang dipanggil saat objek dari kelas Laporan\_barang\_keluar dibuat. Metode ini melakukan inisialisasi komponen GUI dan memanggil metode koneksi.getKoneksi() untuk mengatur koneksi ke database. Selain itu, metode ini juga mengatur model tabel (table) dan menambahkan kolom-kolom yang diperlukan. Metode tampilData() digunakan untuk menampilkan data laporan barang keluar ke dalam tabel. Pertama, metode ini menghapus semua baris yang ada dalam table\_barang dengan menggunakan perulangan dan removeRow(). Kemudian, metode ini menjalankan query untuk mengambil data dari tabel barang\_keluar. Setiap baris hasil query diproses dalam perulangan while. Data dari setiap baris diambil menggunakan getString() dan disimpan dalam variabel-variabel sementara. Data yang diambil dari setiap baris disimpan dalam array data. Selanjutnya, array data ditambahkan sebagai baris baru dalam table menggunakan addRow(). Terakhir, table\_barang diatur agar menampilkan nilai-nilai yang telah ditambahkan dengan menggunakan setModel(table).

## Halaman Daftar Barang Masuk

### Screenshoot



Gambar IV. 19 Daftar Barang Masuk

### Analisa

Metode `tampilData()` dalam kelas `Laporan\_barang\_masuk` digunakan untuk menampilkan data dalam tabel `table\_barang` dengan lebih singkat dan efisien. Pertama, metode ini menghapus semua baris dalam tabel dengan mengatur jumlah baris tabel menjadi 0. Selanjutnya, sebuah query SQL dibuat untuk mengambil semua data dari tabel `barang\_masuk`. Kemudian, menggunakan blok `try-with-resources`, koneksi ke database dibuka, statement dibuat, dan query dieksekusi untuk mendapatkan hasil data dalam objek ResultSet. Selama perulangan, data sementara diambil dari setiap baris hasil query dan dimasukkan ke dalam array `data`. Array tersebut kemudian ditambahkan sebagai baris baru ke dalam tabel menggunakan metode `table.addRow()`. Akhirnya, tabel diatur ulang untuk menampilkan nilai-nilai yang baru. Dengan cara ini, metode `tampilData()` memperbarui dan menampilkan data dengan cara yang lebih singkat dan efisien.

# KESIMPULAN & SARAN

## KESIMPULAN

Dengan pembuatan aplikasi toko meubel berbasis JAVA dan GUI ini yang fokus pada fitur pembelian produk meubel, diharapkan pengelolaan toko mebel akan menjadi lebih efisien dan efektif. Aplikasi ini juga dilengkapi dengan database yang dimana mencangkup CRUD. Agar aplikasi bisa create produk, delete produk dan seterusnya sesuai CRUD. Aplikasi ini akan memudahkan pemilik toko mebel dalam mengelola data produk, pelanggan, transaksi, Sehingga toko mebel dapat beroperasi dengan lebih baik dan memberikan pengalaman belanja yang lebih baik bagi pelanggan.

## B. SARAN

Meskipun kita menginginkan kesempurnaan dalam penyusunan makalah ini, akan tetapi pada kenyataannya masih banyak kekurangan yang perlu kita perbaiki. Hal ini dikarenakan masih minimnya pengetahuan dari kami.

Oleh karena itu kritik dan saran yang membangun dari para pembaca sangat diharapkan sebagai bahan evaluasi untuk ke depannya. Sehingga kita bisa terus menghasilkan penelitian dan karya tulis yang bermanfaat bagi banyak orang.

# DAFTAR PUSTAKA

**[1]** Waterfall, Metode, et al. *Rancang Bangun Pemprograman Penjualan Furniture Indoor Dengan*.

‌**[2]** Wakit Indriyanto, et al. *Sistem Informasi Penjualan Mebeul CV Mandiri Furniture Berbasis Java*. Vol. 1, no. 02, 15 Apr. 2020, jim.unindra.ac.id/index.php/jrami/article/view/241, [https://doi.org/10.30998/jrami.v1i02.241. Accessed 8 July 2023](https://doi.org/10.30998/jrami.v1i02.241.%20Accessed%208%20July%202023).

‌

‌