

LAPORAN PRAKTIKUM

GUI JAVA POS



Disusun Oleh:
Dzakir Tsabit Asy Syafiq (241511071)
Jurusan Teknik Komputer dan Informatika

Program Studi D-3 Teknik Informatika
Politeknik Negeri Bandung
04/11/2025

1. Deskripsi Aplikasi

Program ini merupakan aplikasi kasir sederhana berbasis **Java Swing**.

Tujuannya adalah membantu simulasi proses transaksi penjualan, mulai dari pemilihan produk, perhitungan total harga, hingga pencetakan struk belanja.

Aplikasi memiliki dua panel utama: **panel produk** dan **panel keranjang**, lengkap dengan tombol aksi seperti *Add to Cart*, *Hapus*, *Checkout*, dan *Cetak Struk*.

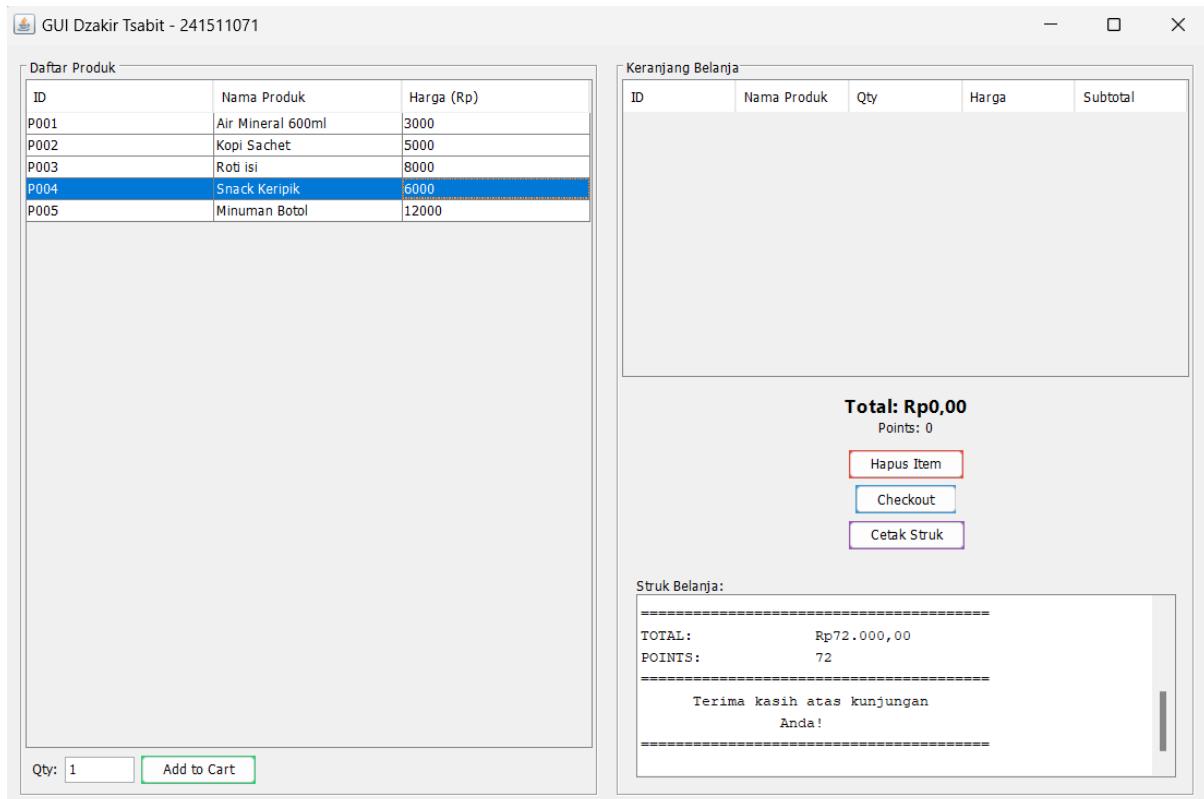
2. Penjelasan Kode Program

2.1. Struktur Utama Program

Kelas utama KasirUI mewarisi JFrame, berfungsi sebagai jendela utama aplikasi.

Pada bagian initializeFrame(), ukuran jendela, judul, layout, dan lokasi diatur menggunakan:

- `setTitle()` untuk nama jendela.
- `setSize()` untuk menentukan dimensi.
- `setLayout(new BorderLayout())` agar mudah membagi panel kiri dan kanan.



2.2. Panel Produk

Panel produk menampilkan daftar barang yang bisa dipilih oleh pengguna. Data produk disimpan dalam array dua dimensi:

```
String[][] dataProduk = {  
    { "P001", "Air Mineral 600ml", "3000" },  
    { "P002", "Kopi Sachet", "5000" },  
    { "P003", "Roti isi", "8000" },  
    { "P004", "Snack Keripik", "6000" },  
    { "P005", "Minuman Botol", "12000" }  
}
```

```

{ "P002", "Kopi Sachet", "5000" },
...
};


```

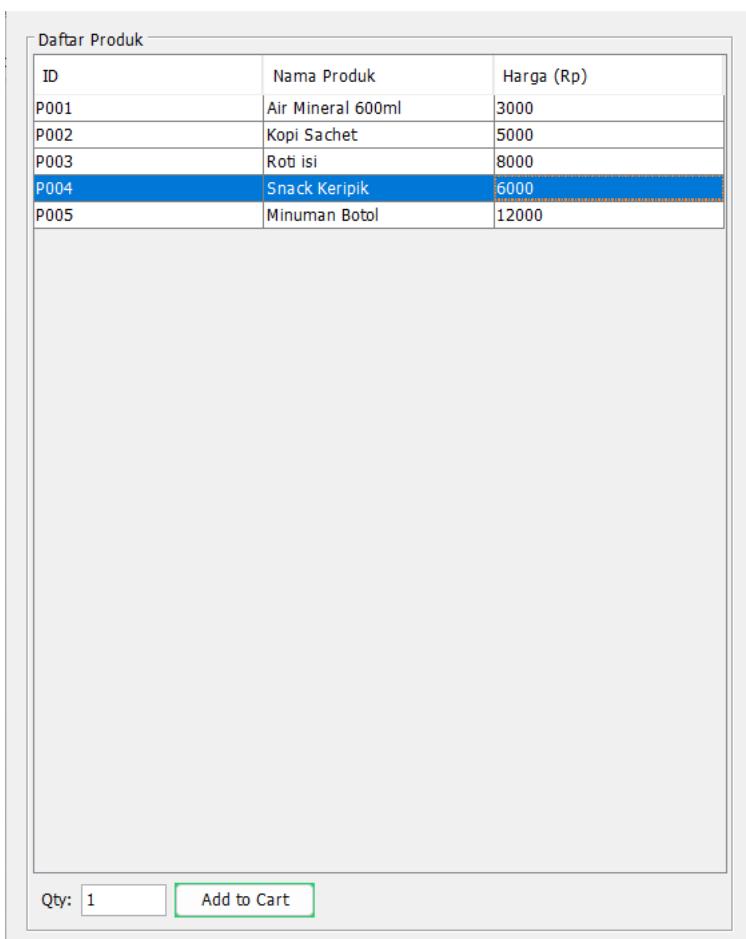
Lalu, ditampilkan pada JTable yang tidak bisa diubah manual dengan meng-override:

```

public boolean isCellEditable(int row, int column) {
    return false;
}


```

Di bawah tabel, terdapat JTextField untuk memasukkan jumlah (*Qty*) dan tombol **Add to Cart**.



2.3. Panel Keranjang

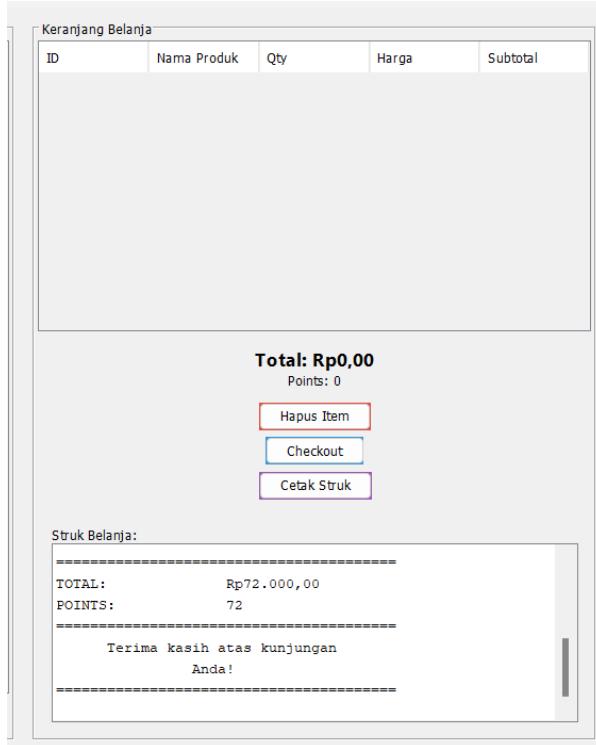
Panel kanan digunakan untuk menampilkan barang yang telah ditambahkan ke keranjang belanja.

Struktur tabelnya berisi kolom:

- ID Produk
- Nama Produk
- Jumlah (Qty)

- Harga
- Subtotal

Semua kolom bersifat *non-editable* untuk menjaga data agar hanya bisa dimanipulasi melalui tombol aksi.



2.4. Tombol Aksi dan Logika Transaksi

Bagian ini mengatur interaksi pengguna melalui tombol:

a. Add to Cart

Tombol ini mengambil produk yang dipilih dari tabel kiri dan jumlah dari qtyField. Jika produk sudah ada di keranjang, maka jumlahnya akan diupdate, bukan ditambahkan baris baru.

b. Hapus Item

Menghapus baris yang dipilih dari tabel keranjang setelah konfirmasi melalui JOptionPane.

c. Checkout

Menghitung total harga semua barang di keranjang menggunakan perulangan for:

```
for (int i = 0; i < modelKeranjang.getRowCount(); i++) {
    total += (int) modelKeranjang.getValueAt(i, 4);
}
```

Setelah itu, poin dihitung dengan rumus:

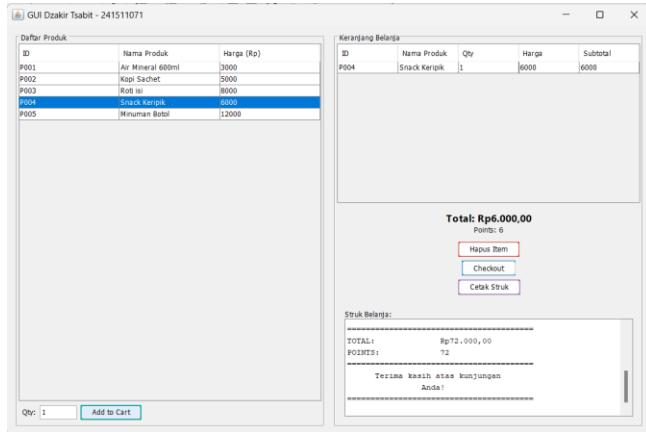
1 poin = setiap Rp 1000

Hasil total dan poin akan tampil di label **Total** dan **Points**, lalu keranjang dikosongkan.

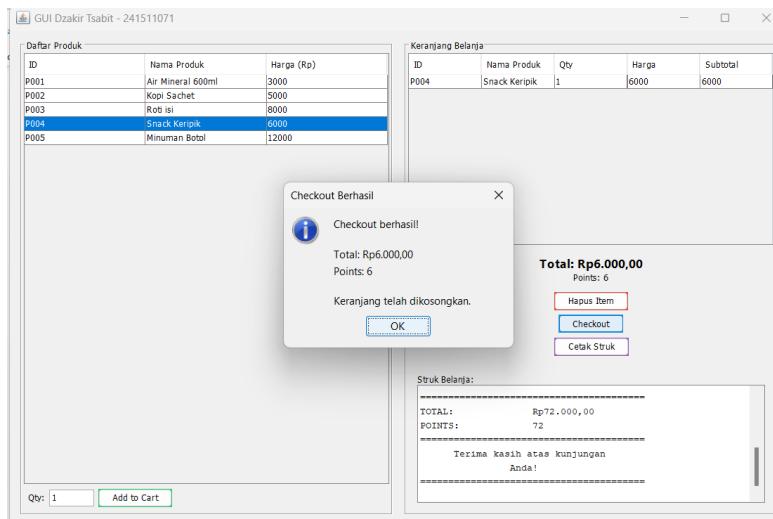
d. Cetak Struk

Menampilkan ringkasan transaksi terakhir ke dalam JTextArea menggunakan data yang tersimpan di variabel strukTerakhir.

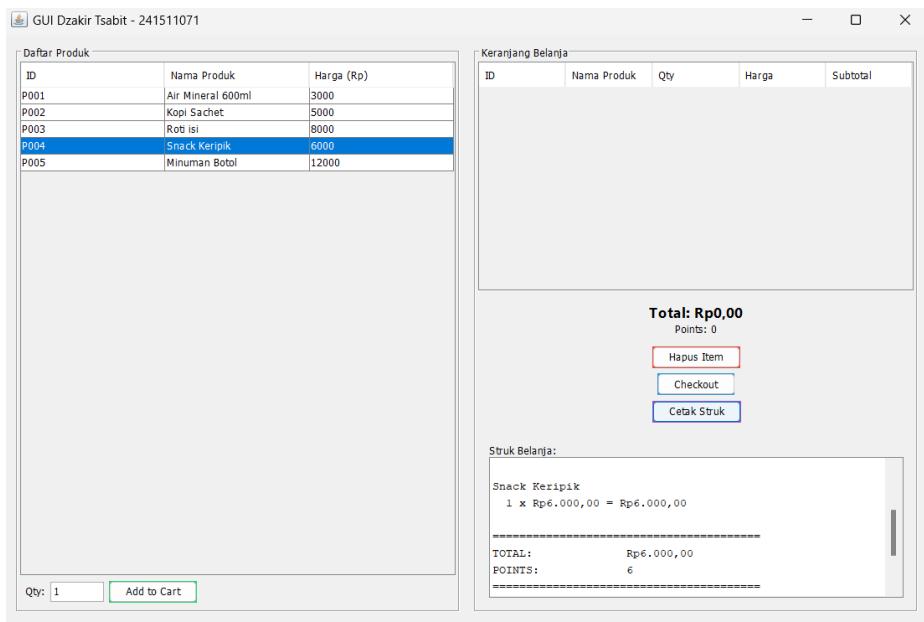
- Setelah menekan tombol **Add to Cart**, tampilkan produk muncul di tabel keranjang.



- Setelah **Checkout**, label *Total* dan *Points* berubah



- Setelah **Cetak Struk**, area teks di bawah kanan berisi struk hasil transaksi.



Text :

=====

TOKO DZAKIR TSABIT

241511071

=====

Tanggal: 05/11/2025 15:24

=====

Snack Keripik

1 x Rp6.000,00 = Rp6.000,00

=====

TOTAL: Rp6.000,00

POINTS: 6

=====

Terima kasih atas kunjungan

Anda!

=====

2.5. Area Struk

JTextArea digunakan untuk menampilkan isi struk dengan format yang rapi menggunakan StringBuilder.

Struk berisi:

- Nama toko dan identitas pembuat
- Daftar produk beserta qty dan subtotal
- Total pembayaran
- Jumlah poin yang diperoleh

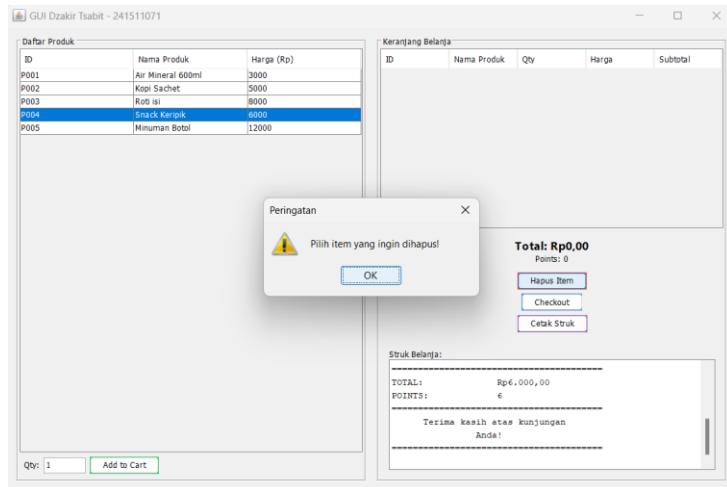
Struk dibuat otomatis saat *checkout*, dan baru ditampilkan saat tombol *Cetak Struk* ditekan

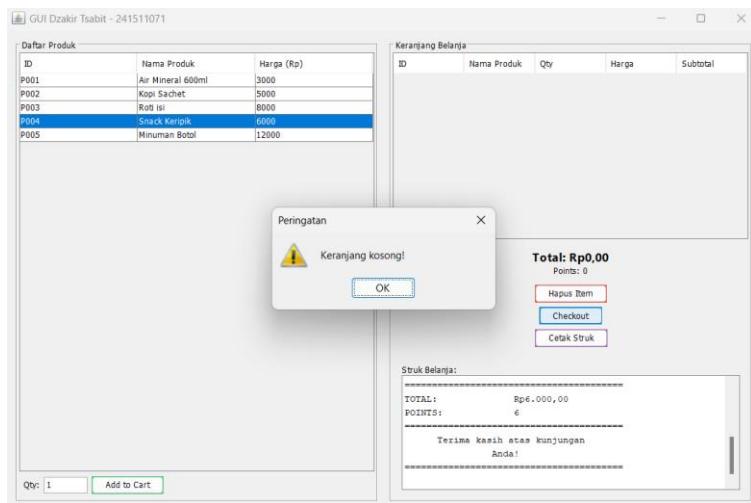
2.6. Fitur Tambahan dan Validasi

Beberapa validasi diterapkan agar program berjalan stabil:

- **Qty harus angka positif.**
- **Produk harus dipilih terlebih dahulu.**
- **Keranjang tidak boleh kosong saat checkout.**
- **Tidak ada input manual di tabel produk dan keranjang.**

Setiap kesalahan akan ditampilkan lewat JOptionPane agar pengguna mudah memahami apa yang salah.





3. Kesimpulan

Aplikasi kasir ini berhasil mengimplementasikan:

- **Event handling** menggunakan *ActionListener* untuk setiap tombol.
- **Data display** menggunakan JTable dan DefaultTableModel.
- **User feedback** melalui JOptionPane.
- **Pemformatan uang lokal** dengan NumberFormat(Locale("id","ID")).

Program ini menunjukkan bagaimana konsep **GUI berbasis event-driven** dapat diterapkan secara praktis untuk kasus transaksi sederhana.