# BAB 3

# ANALISIS DAN PERANCANGAN SISTEM

## 3.1. Analisis Sistem

Analisis sistem dapat didefinisikan sebagai tahap yang bertujuan untuk memahami sistem, mengetahui kekurangan sistem dan menentukan kebutuhan sistem penjualan dan pembelian pada toko RusticShop. Dengan menganalisis prosedur sistem yang sedang berjalan dapat di evaluasi sehingga dapat dibuat satu usulan untuk sistem yang baru dari evaluasi tersebut.

Sebagai analisis pada sistem yang sedang berjalan, akan dibahas bagaimana prosedur dan aliran dokumen yang sedang berjalan yang digambarkan dalam bentuk *flowmap*, pengkodean dan analisis sistem non fungsional yang meliputi perangkat keras dan perangkat lunak yang digunakan, serta analisis *user* yang terlibat.

### 3.1.1. Analisis Masalah

Dari hasil penelitian yang dilakukan pada toko RusticShop maka diperoleh permasalahan bahwa sistem penjualan yang berjalan saat ini menggunakan cara konvensional yakni konsumen datang langsung ke toko RusticShop untuk melakukan pembelian. Hal ini menyebabkan toko RusticShop kesulitan untuk mempromosikan produk dan menyampaikan informasi produk yang dijual secara lengkap kepada pembeli.

Selain itu toko RusticShop sering mengalami kesulitan dalam pengelola data adminstratif seperti pengolahan laporan produk dan laporan penjualan, karena data tersebut dituliskan pada lembaran-lembaran kertas yang belum dibukukan serta belum terkomputerisasi. Hal ini kerap menimbulkan kesalahan dalam penulisan, data lupa tidak dituliskan, serta mengalami kesulitan dalam pengolahan data stok produk.

### 3.1.2. Analisis Prosedur Yang Sedang Berjalan

Analisis aliran data atau prosedur sangat diperlukan untuk mengetahui proses yang berjalan di suatu perusahaan dan sebagai dasar untuk membuat atau memperbaiki serta mempermudah prosedur yang sedang berjalan diperusahaan agar menjadi lebih baik lagi. Berdasarkan hasil penelitian, terdapat beberapa prosedur yang sedang berjalan di Toko RusticShop yaitu sebagai berikut:

1. Prosedur pengadaan produk
2. Prosedur penjualan produk
3. Prosedur laporan penjualan

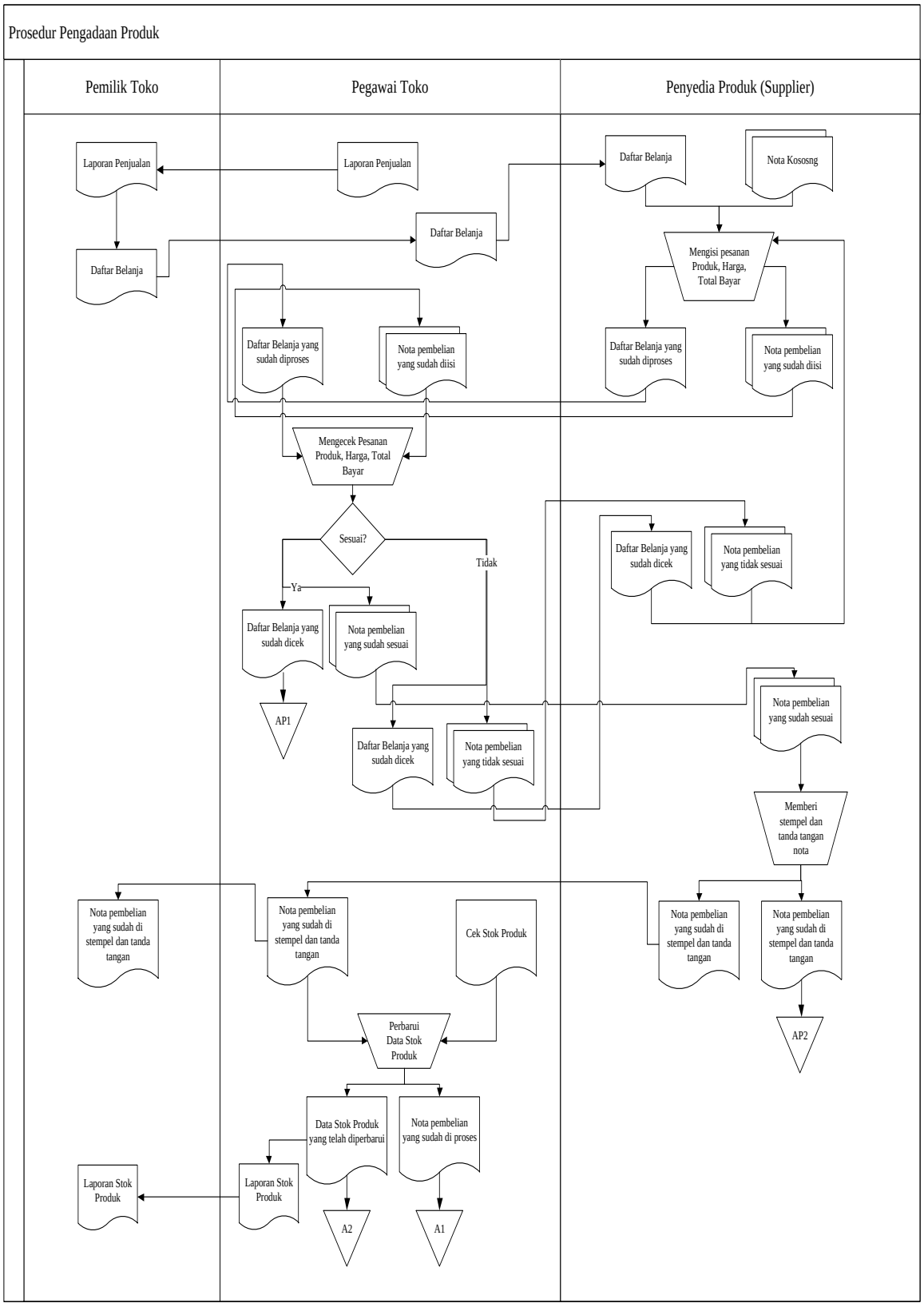
Dibawah ini adalah penjelasan prosedur-prosedur yang terlibat dalam sistem penjualan produk di Toko RusticShop.

#### 3.1.2.1. Prosedur Pengadaan Produk

Prosedur persediaan produk yang sedang berlangsung di Toko RusticShop adalah sebagai berikut:

1. Pegawai toko akan melaporkan ketersediaan produk untuk selanjutnya dilaporkan kepada pemilik.
2. Dari hasil laporan ketersediaan produk, kemudian Pemilik Toko membuat daftar belanja, apa saja produk yang akan dibeli.
3. Setelah membuat daftar belanja, pemilik toko menyerahkan lembar tersebut kepada pegawai toko untuk kemudian diserahkan kepada penyedia produk (*supplier*).
4. Penyedia produk (*supplier*) lembar pemesanan produk yang sudah diisi oleh pembeli (Pemilik Toko). Setelah itu penyedia produk (*supplier*) mengambil nota kosong, kemudian mengisi nota kosong tersebut dengan pesanan produk sesuai dengan lembar pesanan yang sudah diisi oleh pemilik toko. Kemudian, penyedia produk (*supplier*) menyediakan produk yang dipesan setelah itu, menghitung harga, dan jumlah bayar.
5. Penyedia produk (*supplier*) menyerahkan lembar pemesanan produk dan nota pembelian yang sudah diisi dengan harga dan total bayar kepada pegawai toko.
6. Pegawai toko menerima nota pembelian dan lembar pemesanan produk, kemudian melakukan pengecekan apakah sudah sesuai dengan pesanan yang diinginkan. Jika tidak sesuai, pegawai toko menyerahkan kembali ke penyedia produk (*supplier*) untuk diperbaharui agar sesuai dengan permintaan (kembali ke proses 4). Jika sesuai pemilik toko menyerahkan lembar pemesanan produk dan nota pembelian tersebut ke penyedia produk (*supplier*) untuk diproses lanjut.

7. Penyedia produk (*supplier*) menerima lembar pemesanan produk dan nota pembelian yang sudah di cek oleh pegawai toko, kemudian mencap dan menandatangani nota pembelian tersebut. Setelah itu menyimpan satu lembar nota dan lembar pemesanan produk sebagai arsip kemudian memberikan satu lembar nota ke pemilik toko sebagai bukti pembelian.
8. Pegawai Toko menerima satu lembar nota pembelian beserta produk yang dibeli.
9. Setelah melakukan pembelian produk, pegawai toko datang membawa produk yang telah dibeli dengan membawa serta nota pembelian produk.
10. Pegawai toko menyerahkan produk dan nota pembelian ke pemilik toko. Pegawai toko kemudian memperbarui data produk yang ada di dalam daftar stok produk yang diambil dari arsip stok produk sesuai nota pembelian.
11. Setelah stok produk diperbaharui, bagian produk mengarsipkan kembali daftar stok produk ke dalam arsip stok produk dan mengembalikan nota pembelian yang sudah diproses ke pemilik toko.
12. Pegawai toko mengarsipkan nota pembelian yang sudah diproses ke dalam arsip nota pembelian.




**Gambar 3.1 *Flowmap* Prosedur Pengadaan Produk**

Keterangan:

A1: Arsip Stok Produk

A2: Arsip Nota Pembelian

AP1: Arsip Daftar Belanja di Pemilik Toko

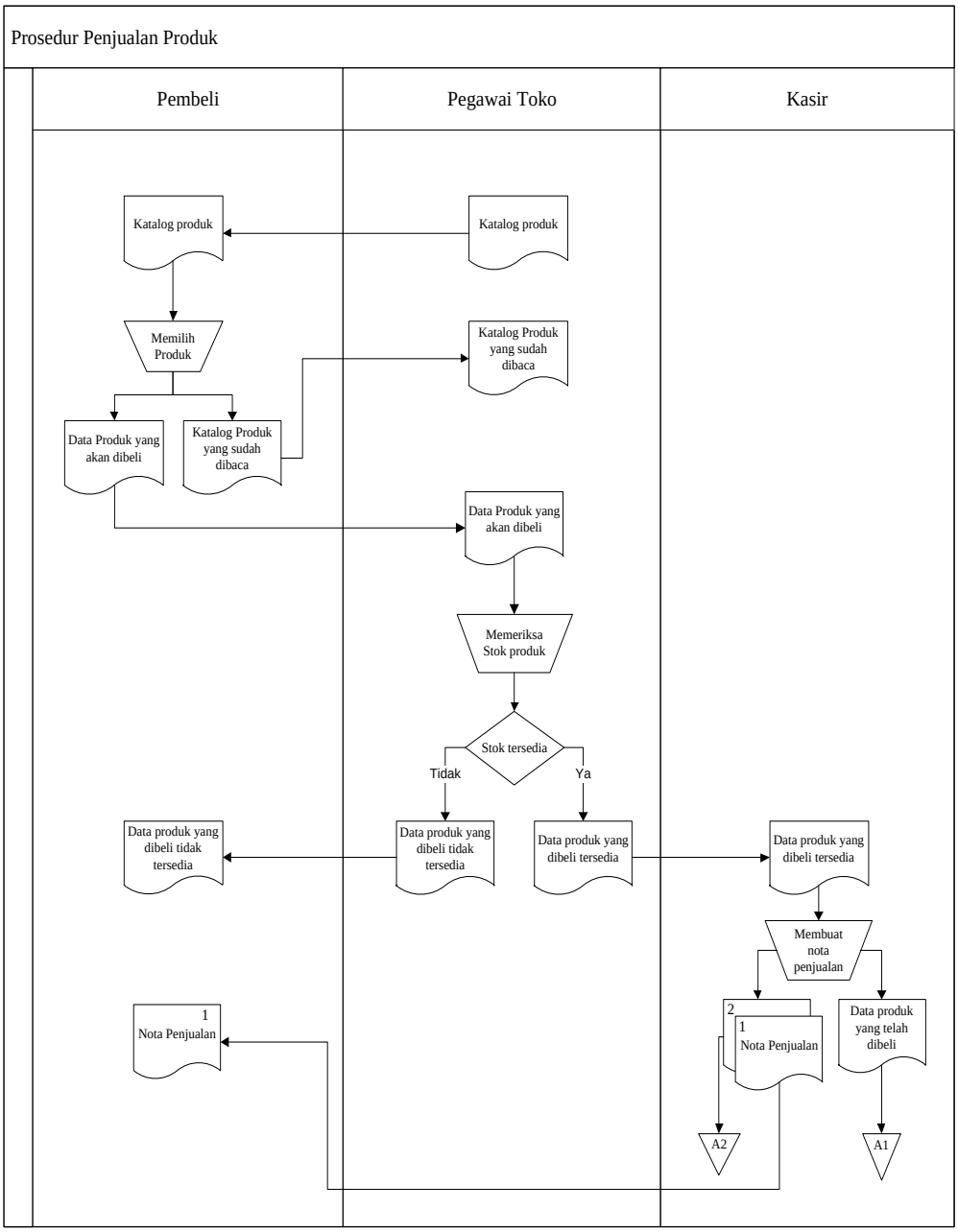
AP2: Arsip Nota Pembelian di Penyedia Produk

### 3.1.2.2. Prosedur Penjualan Produk

Proses penjualan adalah tahapan yang harus dilalui oleh pembeli dalam membeli produk dari mulai melihat produk, memilih produk, membayar sampai proses penerimaan produk oleh pembeli. Kegiatan yang dilakukan pada sistem penjualan adalah sebagai berikut:

1. Pembeli akan memilih produk yang akan dibeli dengan melihat daftar produk yang tersedia atau langsung menayakan kepada pegawai toko. Pegawai toko kemudian akan menanyakan ke bagian stok untuk mengecek ketersediaan produk yang akan di beli, jika produk ada maka bagian stok akan memberikan produk tersebut ke kasir dan mempersilahkan pembeli ke kasir untuk mengambil produk yang akan dibeli dan melakukan pembayaran.
2. Kasir membuat nota penjualan. Nota dibuat rangkap dua, satu yang berwarna putih untuk pembeli dan satu lagi yang berwarna kuning untuk toko.

3. Kasir memberikan nota penjualan tersebut kepada pembeli dan menyimpan arsip produk yang telah dibeli untuk dijadikan laporan penjualan.
4. Pembeli membayar sesuai dengan total harga yang tertera dalam nota penjualan. Kemudian akan memberikan produk yang dibeli dan nota penjualan kepada pembeli.




**Gambar 3.2 *Flowmap* Prosedur Penjualan Produk**

Keterangan:

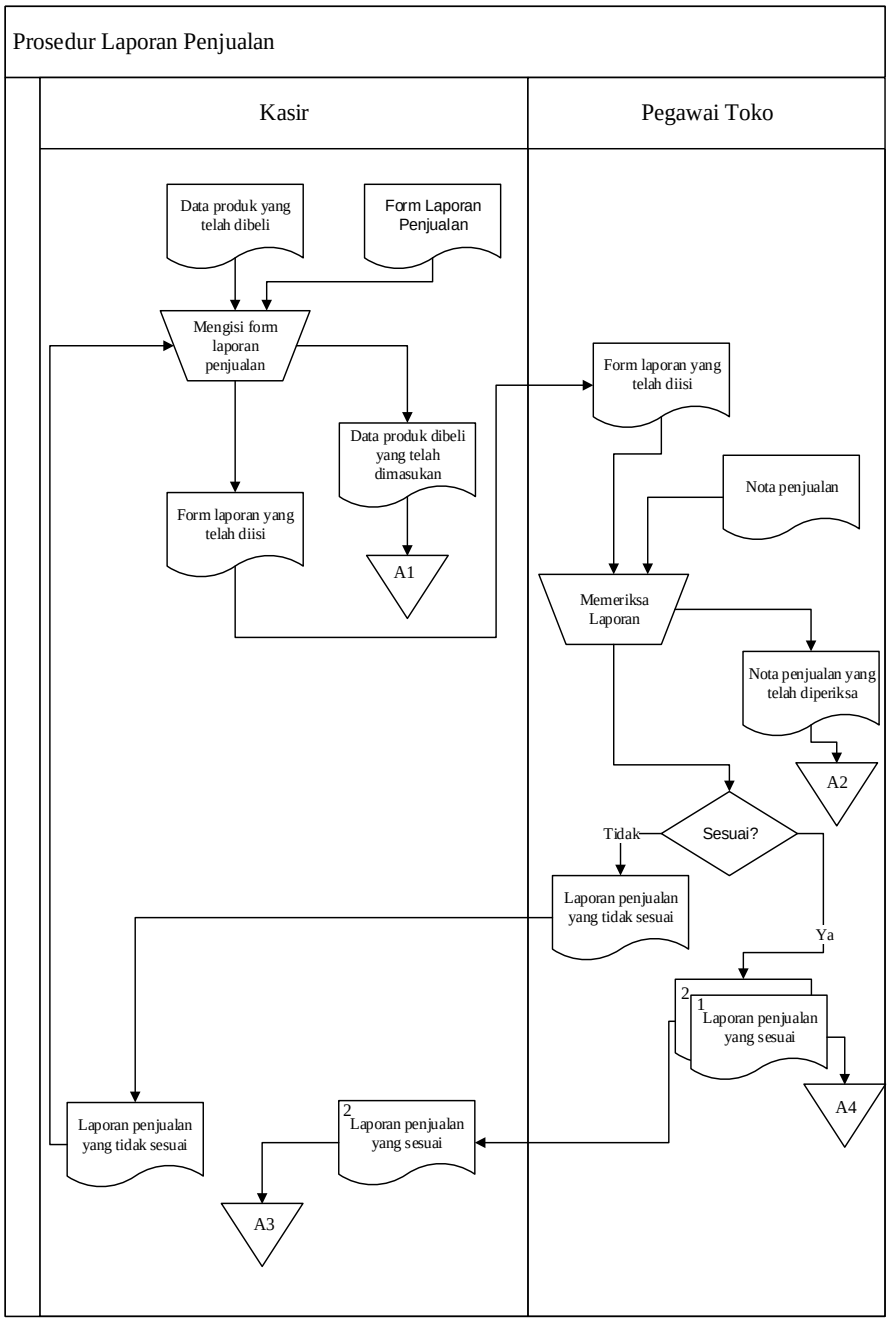
A1 : Arsip data barang yang telah dibeli untuk kasir

A2 : Arsip nota penjualan untuk konsumen

### 3.1.2.3. Prosedur Laporan Penjualan

Proses pembuatan laporan didalamnya berisi nota transaksi yang akan ditandatangani oleh pemilik. Adapun prosedur laporan penjualan ini meliputi produk apa saja yang telah dibeli, sehingga dari data pembelian produk dapat dijadikan evaluasi untuk mengetahui jenis produk apa yang paling disenangi oleh konsumen. Adapun kegiatan yang dilakukan untuk pembuatan laporan adalah sebagai berikut :

1. Kasir mengisi *form* laporan penjualan berdasarkan data produk yang telah dibeli
2. Apabila telah selesai maka kasir akan menyerahkan laporan penjualan kepada pemilik toko.
3. Pemilik toko akan memeriksa laporan apakah sudah sesuai dengan nota penjualan.
4. Jika belum sesuai maka laporan penjualan akan dikembalikan kepada kasir, kemudian kasir akan memeriksa kembali dengan berdaraskan nota penjualan produk
5. Pemilik toko menerima laporan penjualan dari kasir dan menyimpannya sebagai arsip
6. Kasir menyimpan data produk yang dibeli yang telah dimasukan, nota penjualan yang telah diperiksa dan laporan penjualan yang sesuai sebagai arsip.




**Gambar 3.3 *Flowmap* Prosedur Laporan Penjualan Produk**

Keterangan:

A1 : Arsip data barang dibeli yang telah dimasukan untuk kasir

A2 : Arsip nota penjualan yang telah diperiksa untuk kasir

A3 : Arsip laporan penjualan yang sesuai untuk kasir

A4 : Arsip laporan penjualan yang sesuai untuk pemilik toko

### 3.1.3. Evaluasi Sistem yang sedang berjalan

Evaluasi sistem yang sedang berjalan dimaksudkan guna memperoleh solusi yang lebih baik untuk perubahan sistem yang sudah ada. Evaluasi ini dilakukan setelah proses analisa sistem yang sedang berjalan. Dapat disimpulkan berdasarkan hasil analisa sistem dan wawancara yang telah dilakukan sebagai berikut:

**Tabel 3.1 Evaluasi sistem yang sedang berjalan**

| No | Permasalahan | Bagian/Pihak |
|---|---|---|
| 1. | Proses penjualan barang mengharuskan pembeli datang langsung ke toko untuk dapat membeli barang. | Pembeli |
| 2. | Keterbatasan toko dalam melakukan promosi dan penyampaian informasi produk yang lebih lengkap kepada pembeli dan Pengelolaan stok produk yang masih bersifat manual. | Petugas toko |
| 3. | Pembukuan, pembuatan laporan, rekapitulasi penjualan yang masih bersifat konvensional. | Kasir |

### 3.1.4. Aturan Bisnis

Berdasarkan hasil evaluasi sistem yang berjalan, solusi yang ditawarkan adalah membangun aplikasi *e-commerce* pada Toko RusticShop. Aturan bisnis pada aplikasi *e-commerce* pada Toko RusticShop adalah sebagai berikut:

1. Pihak Toko RusticShop hanya melayani pemesanan lingkup domestik saja.

2. Pengunjung, pihak Toko RusticShop memiliki aturan untuk pengunjung sebagai berikut:
    a. Pengunjung dapat melakukan pendaftaran sebagai member dengan melengkapi form pendaftaran yang telah disediakan.
    b. Setelah melakukan pendaftaran pengunjung wajib mengaktivasi akun sebelum dapat melakukan *login*.
    c. Pengunjung dapat melakukan reset password jika sewaktu-waktu lupa password dengan memanfaatkan fitur lupa password.
3. Member, pihak Toko RusticShop memiliki aturan untuk member sebagai berikut:
    a. Member yang dapat melakukan *login* hanyalah member yang sudah mengaktivasi akunnya.
    b. Member yang tidak *login* lebih dari 1 tahun dinon-aktifkan.
    c. Member dapat meminta pengaktifan akun kembali.
4. Produk, pihak Toko RusticShop memiliki aturan untuk produk sebagai berikut:
    a. Pihak Toko RusticShop akan menampilkan produk sering dilihat oleh pengunjung ataupun member.
    b. Pihak Toko RusticShop akan menampilkan produk terlaris berdasarkan produk yang banyak dibeli setiap bulannya oleh member. Jika stok produk sudah habis maka tidak akan dihitung.

c. Pihak Toko RusticShop akan menampilkan produk terbaru pada halaman utama berdasarkan tanggal dimasukannya produk yang terbaru setiap bulannya.
d. Pihak Toko RusticShop akan menampilkan produk diskon pada halaman utama.
e. Pihak Toko RusticShop akan memberikan diskon kepada produk yang belum ada yang terjual hingga 3 bulan semenjak produk diinputkan.
f. Besarnya diskon ditentukan oleh pemilik toko, untuk penjualan produk secara *online* diberikan diskon yang berkisar antara 15% sampai dengan 25% dari harga jual.
g. Pihak Toko RusticShop akan menampilkan rating produk dengan skala 5 bintang. Pemberian rating hanya bisa dilakukan oleh member jika sudah melakukan pembelian terhadap produk tersebut.

5. Pengelolaan Stok, pihak Toko RusticShop memiliki aturan untuk pengelolaan stok sebagai berikut:
   a. Stok akan berkurang ketika *member* memasukan produk ke dalam keranjang belanja. Apabila member sudah menyimpan pemesanan tanpa melakukan *checkout*, maka dalam waktu 2 jam pemesanan secara otomatis dibatalkan dan stok akan kembali ke jumlah stok awal.
   b. Sistem hanya akan menampilkan informasi stok produk berdasarkan ukuran yang masih tersedia.

6. Pemesanan produk, pihak Toko RusticShop memiliki aturan untuk pemesanan produk sebagai berikut:
    a. Pemesanan produk hanya bisa dilakukan oleh member, untuk pembelian minimal satu produk.
    b. Pemesanan produk tidak dibatasi tetapi disesuaikan dengan jumlah stok yang ada.
    c. Member tidak dapat melakukan pemesanan produk lagi, jika pemesanan sebelumnya belum lunas.
    d. Member wajib mengisi alamat tujuan pengiriman dan jasa pengiriman yang ingin digunakan untuk proses pengiriman barang.
7. Pembayaran, pihak Toko RusticShop memiliki aturan untuk pembayaran sebagai berikut:
    a. Member dapat melakukan pembayaran dengan cara transfer bank atau melalui *Paypal*.
    b. Member diberikan waktu maksimum 12 jam untuk melakukan konfirmasi pembayaran setelah mendapatkan nota pemesanan.
    c. Sistem akan menampilkan jumlah total pembayaran.
    d. Pembayaran harus sesuai dengan jumlah total yang harus dibayar, jika tidak sesuai maka sistem akan menampilkan pesan kesalahan. Jika pembayaran lebih dari jumlah total yang harus dibayar, kelebihan bayar menjadi milik Toko RusticShop.

e. Pihak Toko RusticShop akan merespon konfirmasi pembayaran yang dilakukan oleh member. Apabila konfirmasi diluar jam yang telah ditentukan maka akan diproses pada hari kerja berikutnya.
f. Konfirmasi pembayaran ditolak jika member tidak sesuai dengan jumlah pembayaran.

8. Pengiriman barang, pihak Toko RusticShop memiliki aturan untuk pengiriman barang sebagai berikut:
   a. Jika konfirmasi pembayaran melebihi jam 15.00 maka proses pengiriman barang akan dilakukan keesokan harinya.
   b. Proses pengiriman barang dilakukan 2 kali yaitu pada pagi hari pukul 9.00 dan pukul 16.00.
   c. Pengiriman dilakukan dengan menggunakan jasa pengiriman JNE dan Tiki.
9. Retur, pihak Toko RusticShop memiliki aturan untuk retur barang sebagai berikut:
   a. Member dapat mengajukan retur jika produk tidak sesuai pesanan atau produk rusak (cacat)
   b. Retur dapat dilakukan jika status pemesanannya sudah diterima dan belum melebihi waktu 24 jam, terhitung dari waktu penerimaan barang.
   c. Ongkos kirim retur ditanggung oleh member, sedangkan untuk ongkos kirim produk pengganti ditanggun oleh pihak Toko RusticShop.

d. Teknis penggantian produk yang diretur dapat berupa uang ataupun stok sesuai kesepakatan antara kedua belah pihak yang dilakukan diluar sistem aplikasi.

### 3.1.5. Analisis Spesifikasi Kebutuhan Perangkat Lunak

Analisis spesifikasi kebutuhan perangkat lunak untuk sistem yang akan dibangun adalah sebagai berikut

1. Proses pendaftaran, proses ini dilakukan oleh pengunjung yang hendak menjadi *member*.
   a. Verifikasi data pendaftaran, proses ini dilakukan oleh pengunjung untuk menjadi *member*, didalam proses ini pengunjung diharuskan mengisikan data seperti *email*, *password*, nama lengkap, propinsi, kota, alamat, dan kodepos.
   b. Aktivasi akun, proses ini sebagai langkah berikutnya setelah proses pengisian data pendaftaran telah selesai dilakukan, dimana sistem akan mengirimkan *link* aktivasi akun ke *email* pengunjung.
2. Proses *login*, proses ini dapat dilakukan oleh *member*, operator, admin atau pemilik toko yang terdiri dari dua proses, yaitu :
   a. Validasi data login, proses ini dilakukan oleh *member* setelah aktivasi akun telah berhasil dilakukan. *Member* akan memasukkan *email* dan *password* untuk *login* ke halaman *member*. Proses ini juga dapat dilakukan oleh operator, admin, dan pemilik toko dengan memasukkan *username* dan *password*.

b. Lupa *password*, proses ini dapat dilakukan oleh member, operator, admin, dan pemilik toko. *Member*, operator, admin, atau pemilik toko memasukkan *email* sehingga sistem akan memverifikasi *email* yang dimasukkan terdaftar atau tidak. Langkah selanjutnya adalah sistem akan mengirimkan *link* ke *email member*, operator, admin, dan pemilik toko untuk membuat *password* baru.

3. Proses pengolahan data *user*, proses ini dilakukan oleh *member*, operator, admin dan pemilik toko terdiri dari beberapa proses yaitu :
   a. Edit profil, proses ini dilakukan oleh *member*, operator, admin, dan pemilik toko untuk mengubah data akun yang telah tersimpan.
   b. Tambah operator, proses ini dilakukan oleh admin untuk menambah operator baru.
   c. Status *member* pasti, proses ini dilakukan oleh operator untuk memverivikasi kartu pasti *member*.
   d. Ubah *password*, proses ini dilakukan oleh *member*, operator, admin, dan pemilik toko untuk merubah *password* yang lama dengan *password* yang baru
4. Proses pengolahan data produk, proses ini dilakukan oleh pengunjung, *member*, operator, admin, dan pemilik toko yang terdiri dari beberapa proses, yaitu :
   a. Cari produk, proses ini dilakukan oleh pengunjung, *member*, dan admin untuk mencari produk yang diinginkan.

b. Edit produk, proses ini dilakukan oleh admin untuk merubah data produk.
c. Tambah produk, proses ini dilakukan oleh admin untuk menambah data produk yang baru.
d. Tambah *rating* produk, proses ini dilakukan oleh *member* untuk menambahkan *rating* produk yang sudah dibeli.

5. Proses pengolahan data ganbar, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
   a. Edit gambar, proses ini dilakukan oleh admin untuk merubah data gambar yang telah ada.
   b. Tambah gambar, proses ini dilakukan oleh admin untuk menambah data gambar yang baru.
   c. Hapus gambar, proses ini dilakukan oleh admin untuk menghapus data gambar yang ada.
6. Proses pengelolaan data barang, proses ini dilakukan oleh admin dan *member* yang terdiri dari beberapa proses, yaitu :
   a. Edit barang, proses ini dilakukan oleh admin untuk merubah data barang yang telah ada.
   b. Tambah barang, proses ini dilakukan oleh admin untu menambah data barang yang baru.
   c. Hapus barang, proses ini dilakukan oleh admin untuk menghapus data barang yang telah ada.

7. Proses pengolahan data paket pengiriman, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
    a. Edit paket pengiriman, proses ini dilakukan oleh admin untuk merubah data paket pengiriman
    b. Tambah paket pengiriman, proses ini dilakukan oleh admin untuk menambah data paket pengiriman yang baru
    c. Hapus paket pengiriman, proses ini dilakukan oleh admin untuk menghapus data paket pengiriman
8. Proses pengolahan data jasa pengiriman, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
    a. Edit jasa pengiriman, proses ini dilakukan oleh admin untuk merubah data jasa pengiriman.
    b. Tambah jasa pengiriman, proses ini dilakukan oleh admin untuk menambah data jasa pengiriman yang baru.
    c. Hapus jasa pengiriman, proses ini dilakukan oleh admin untuk menghapus data jasa pengiriman.
9. Proses pengolahan data ongkos kirim, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
    a. Edit ongkos kirim, proses ini dilakukan oleh admin untuk merubah data ongkos kirim yang telah tersimpan sebelumnya.
    b. Tambah ongkos kirim, proses ini dilakukan oleh admin untuk menambah data ongkos kirim yang baru.

10. Proses pengolahan data kota, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
    a. Edit kota, proses ini dilakukan oleh admin untuk merubah data kota yang telah tersimpan sebelumnya.
    b. Tambah kota, proses ini dilakukan oleh admin untuk menambah data kota yang baru.
    c. Hapus kota, proses ini dilakukan oleh admin untuk menghapus data kota.
11. Proses pengolahan data propinsi, proses ini dilakukan oleh admin yang terdiri dari beberapa proses, yaitu :
    a. Edit propinsi, proses ini dilakukan oleh admin untuk merubah data propinsi yang telah tersimpan seelumnya.
    b. Tambah propinsi, proses ini dilakukan oleh admin untuk menambah data propinsi yang baru.
    c. Hapus prospinsi, proses ini dilakukan oleh admin untuk menghapus data propinsi.
12. Proses pengolahan data laporan, proses ini dilakukan oleh pemilik toko yang terdiri dari beberapa proses yaitu :
    a. Laporan penjualan, proses ini dilakukan oleh pemilik toko ketika akan mencetak laporan penjualan.
    b. Laporan stok produk, proses ini dilakukan oleh pemilik toko ketika akan mencetak laporan stok produk.

13. Proses pengolahan data pemesanan, proses ini dilakukan oleh *member* dan operator yang terdiri dari beberapa proses, yaitu :
    a. Pengolahan data keranjang belanja, proses ini dilakukan oleh *member* untuk menyimpan data pemesanan produk
    b. Cari pemesanan, proses ini dilakukan oleh operator untuk mencari data pemesanan.
    c. Edit status pemesanan, proses ini dilakukan oleh operator untuk merubah status pemesanan *member*.
    d. Batal pemesanan, proses ini dilakukan oleh *member* untuk membatalkan pemesanan yang telah dilakukan
    e. Input nomor resi pengiriman, proses ini dilakukan oleh operator untuk untuk mengisi nomor resi pengiriman pada pemesanan *member*.
    f. History pemesanan, proses ini dilakukan oleh *member* ataupun operator untuk melihat pemesanan yang telah dilakukan.
14. Proses pengelolaan data keranjang belanja, proses ini dilakukan oleh *member* yang terdiri dari beberapa proses, yaitu :
    a. Tambah pemesanan produk, proses ini dilakukan oleh *member* untuk menambah *item* produk dikeranjang.
    b. Edit jumlah pemesanan produk, proses ini dilakukan oleh *member* untuk meruah jumlah produk yang akan dipesan
    c. Batal pemesanan, proses ini dilakukan oleh *member* untuk membatalkan produk yang akan dipesan

d. *Input* alamat pengiriman, proses ini dilakukan oleh *member* untuk memasukkan alamat tujuan pengiriman yang akan menjadi tujuan dari pengiriman produk.
e. Pilih paket pengiriman, proses ini dilakukan oleh *member* untuk memilih paket pengiriman.
f. Simpan data pemesanan, proses ini dilakukan oleh *member* untuk menyimpan data pemesanan yang telah dilakukan.

15. Proses konfirmasi pembayaran, proses ini dilakukan baik oleh *member* ataupun operator. *Member* akan melakukan konfirmasi pembayaran setelah melakukan pembayaran sesuai dengan total pembelian yang harus dibayar baik melalui setor tunai, ATM, SMS Banking, atau *Internet* Banking. Operator akan melakukan pengecekan data konfirmasi pembayaran yang dilakukan oleh *member*, apakah data tersebut *valid* atau *invalid* dan sistem akan mengirimkan pemberitahuan ke *email member*.
16. Proses pengolahan retur, proses ini dilakukan oleh *member* dan juga oleh operator yang terdiri dari beberapa proses, yaitu :
    a. Input alasan retur, proses ini dilakukan oleh *member* untuk memberikan alasan ketika retur dilakukan
    b. Input nomor resi retur, proses ini dilakukan oleh operator untuk mengisi nomor resi produk yang diretur.
    c. Simpan data retur, proses ini dilakukan oleh *member* untuk menyimpan data retur produk.

d. Edit status retur, proses ini dilakukan oleh operator untuk mengubah status retur yang diajukan oleh *member*.

17. Proses pengecekan *tracking*, proses ini dilakukan oleh *member* dan operator untuk mengetahui pengiriman produk sudah dikirim atau diterima.
18. Proses *backup* dan *restore*, proses ini dilakukan oleh admin untuk mem*backup*dan me-*restore database*.

### 3.1.6. Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

#### 3.1.6.1. Analisis Pengguna (*User*)

RusticShop saat ini memang belum memiliki salah satu sumber daya manusia khusus bidang IT. Oleh karena itu, media promosi RusticShop masih terbatas *offline*, hal itu dapat dilihat dengan tidak adanya penggunaan Facebook sebagai sarana promosi dan adanya penggunaan *email.* Keahlian khusus yang dimiliki hanyalah penggunaan *Microsoft Office*, ini dapat dilihat dari adanya

pembuatan laporan rekap menggunakan *Microsoft Excel* dan pembuatan laporan lainnya menggunakan *Microsoft Word*.

Toko RusticShop saat ini memiliki 3 penguna yaitu pemilik toko, kasir, dan pegawai toko yang dapat dilihat pada tabel 3.2 berikut:

**Tabel 3.2 Karakteristik Pengguna Saat Ini**

| Pengguna | Pengalaman | Keterampilan |
|---|---|---|
| Pemilik Toko | Pernah menggunakan aplikasi *website* dan *desktop* | Menguasai penggunaan komputer, dapat menggunakan *internet* dan memahami manajemen bisnis |
| Kasir | Pernah menggunakan aplikasi *website* dan *desktop* | Bisa mengoperasikan komputer dan menggunakan *internet* |
| Pegawai Toko | Pernah menggunakan aplikasi *website* dan *desktop* | Bisa mengoperasikan komputer dan menggunakan *internet* |

Sistem yang akan dibangun ini digunakan oleh 5 jenis pengguna utama yaitu pemilik toko, *admin*, operator, *member*, dan pengunjung. Rincian analisis pengguna dari aplikasi *e-commerce* yang dibangun adalah sebagai berikut:

1. Analisis Pengguna Pemilik Toko

   Adapun rincian analisis pengguna pemilik toko dapat dilihat pada tabel 3.3.

**Tabel 3.3 Analisis Pengguna Pemilik Toko**

| Pengguna | Pemilik |
|---|---|
| Tanggung Jawab | Mengelola *reporting*/laporan penjualan yang terjadi. |
| Hak akses | Mengelola bank, laporan penjualan, dan mengelola informasi toko. |
| Tingkat Keterampilan | Menguasai komputer, dan memiliki pengetahuan tentang *internet*. |
| Pengalaman | Tidak ada |

| | |
|---|---|
| Jenis Pelatihan | Cara menggunakan aplikasi. |

2. Analisis Pengguna *Operator*

Adapun rincian analisis pengguna *admin* dapat dilihat pada tabel 3.4.

**Tabel 3.4 Analisis Pengguna *Operator***

| Pengguna | Admin |
|---|---|
| Tanggung Jawab | Mengelola data yang terdapat di aplikasi. |
| Hak akses | Mengelola retur, mengelola pemesanan, dan pengelolaan data member. |
| Tingkat Keterampilan | Menguasai komputer, mampu menginput data, pemahaman yang cukup untuk mengelola data dalam *web* serta memiliki pengetahuan tentang internet seperti menggunakan *browser* dan *email.* |
| Pengalaman | Tidak ada |
| Jenis Pelatihan | Cara menggunakan aplikasi. |

3. Analisis Pengguna *Admin*

Adapun rincian analisis pengguna *admin* dapat dilihat pada tabel 3.5.

**Tabel 3.5 Analisis Pengguna *Admin***

| Pengguna | Admin |
|---|---|
| Tanggung Jawab | Mengelola data yang terdapat di aplikasi. |
| Hak akses | Mengelola profil operator, melakukan pengolaan kategori produk, pengelolaan produk, pengelolaan provinsi, pengelolaan kota, pengelolaan jasa penggiriman dan backup aplikasi. |
| Tingkat Keterampilan | Menguasai komputer, mampu menganalisa data, pemahaman yang cukup untuk mengelola data pada MySQL, serta memiliki pengetahuan tentang internet seperti menggunakan *browser* dan *email.* |
| Pengalaman | Tidak ada |
| Jenis Pelatihan | Cara menggunakan aplikasi *e-commerce* pada jaringan *internet* dan pengolaan data pada MySQL |

4. Analisis Pengguna Member

Adapun rincian analisis pengguna member dapat dilihat pada tabel 3.6.

**Tabel 3.6 Analisis Pengguna Member**

| Pengguna | Member |
|---|---|
| Tanggung Jawab | Melakukan pemesanan. |
| Hak Akses | Melakukan pemesanan produk, mengubah data pribadi, melihat transaksi yang sudah pernah dilakukan, mengelola keranjang belanja, dan melakukan retur. |
| Tingkat Pendidikan | Apa saja. |
| Tingkat Keterampilan | Dapat menggunakan komputer dan dapat menggunakan *internet* untuk *browsing*. |
| Pengalaman | Tidak ada |
| Jenis Pelatihan | Tidak ada |

5. Analisis Pengguna Pengunjung

**Tabel 3.7 Analisis Pengguna Pengunjung**

| Pengguna | Pengunjung |
|---|---|
| Tanggung Jawab | Tidak ada. |
| Hak Akses | Dapat melihat produk yang terdapat di website tersebut dan tidak dapat melakukan pemesanan barang karena belum terdaftar dan melakukan registrasi, dapat melakukan transaksi jika sudah melakukan registrasi. |
| Tingkat Pendidikan | Apa saja. |
| Tingkat Keterampilan | Dapat menggunakan komputer dan dapat menggunakan *internet* untuk *browsing*. |
| Pengalaman | Tidak ada |
| Jenis Pelatihan | Tidak ada |

Berdasarkan hasil seluruh analisis pengguna yang telah dipaparkan maka dapat disimpulkan bahwa solusi agar sistem pada aplikasi e-commerce pada Toko

RusticShop adalah harus diadakannya pelatihan untuk pemilik toko dan pegawai dalam cara penggunaan dan pengelolaan aplikasi.

#### 3.1.6.2. Analisis Kebutuhan Perangkat Keras

RusticShop saat ini memiliki sebuah komputer dengan spesifikasi sebagai berikut:

1. *Motherboard* dengan *Processor* Pentium IV 2.4 GHZ
2. *Hard Disk* berkapasitas 80 GB
3. RAM berkapasitas 512 MB
4. DVD *Rom*
5. *Keyboard* dan *Mouse*
6. Monitor SVGA ukuran 15” dengan resolusi maksimal 1024 x 768
7. Lan Card 10/100Mbps
8. Printer
9. Modem

Untuk menjalankan aplikasi yang akan dibuat agar berjalan dengan baik maka diperlukan 3 perangkat keras dengan syarat minimal perangkat keras yang dibutuhkan adalah:

1. *Motherboard* dengan *Processor* Pentium IV 2.4 GHz
2. *Hard Disk* berkapasitas 80 GB
3. RAM berkapasitas 1 GB
4. Lan Card 10/100Mbps
5. Monitor 15” dengan resolusi minimal 1024 x 768

6. *Keyboard* dan *Mouse*
7. Printer
8. Modem

Untuk setup aplikasi sehingga aplikasi yang akan dibuat bisa diakses secara online melalui alamat website maka dibutuhkan sebuah perangkat keras server dengan spesifikasi yang dibutuhkan adalah:

1. Intel Xeon E3 1230 Quad Core
2. 16GB DDR3 ECC RAM
3. 2x 1TB SATA-II
4. Hardware RAID 1 (512MB Cache)

Berdasarkan hasil pengamatan analisis perangkat keras diatas spesifikasi komputer pada toko RusticShop kurang memadai untuk menjalankan aplikasi, maka diperlukan adanya penambahan dan upgrade perangkat keras terutama dibagian RAM dan Harddisk, sehingga aplikasi dapat berjalan dengan baik. Sedangkan pada sisi perangkat keras server dapat disolusikan dengan membeli sebuah hosting agar aplikasi dapat diakses secara *online*.

#### 3.1.6.3. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan pada saat ini yaitu:

1. Sistem Operasi Windows XP
2. Software Pendukung Microsoft Office 2003
3. Web Browser Google Chrome

Sedangkan perangkat lunak yang dibutuhkan dalam pembangunan aplikasi ini adalah Browser untuk menjalankan aplikasi seperti Mozilla Firefox, Safari, Opera, Google Chrome, dan Internet Explorer masing-masing dari berbagai versi dan juga tools pendukung seperti Notepad++ dan FileZilla untuk menuliskan kode program dan mengupload ke server. Selain itu perangkat lunak yang dibutuhkan disisi server adalah web server, cpanel, dan phpMyAdmin agar aplikasi yang dibangun dapat berjalan dengan baik.

Berdasarkan pengamatan diatas perangkat lunak di Toko RusticShop belum cukup mendukung untuk menjalankan aplikasi yang akan dibangun. Dibutuhkan perangkat lunak pendukung untuk pembangunan aplikasi ini, sedangkan dari sisi server dapat disolusikan dengan cara membeli sebuah hosting agar memudahkan dalam pembuatan aplikasi ini.

#### 3.1.6.4. Analisis Metode Smart Recommendation System (SRS)

##### 3.1.6.4.1. Tabel Data Rating

Tabel *rating* berisi data rating yang digunakan untuk menghasilkan rekomendasi produk, yang terdiri dari *id_member, id_produk*, dan *rating*. Tabel ini berisi nilai *rating* dari setiap member untuk setiap produk yang telah dirating.

Pada Tabel 3.8 berisi data *rating* dari member a terhadap produk b yang berjumlah sebanyak 23 *instance* dengan rentang nominal *rating* antara 1-5 dengan iterasi 1, nilai rating 1 berarti produk dinilai sangat buruk, dan rating 5 berarti produk dinilai sangat baik. Dalam tabel tersebut juga berisi nilai rata-rata *rating* dari tiap member.

**Tabel 3.8 Tabel *Rating User***

| | Produk 5 | Produk 6 | Produk 7 | Produk 8 | Produk 9 | Rata-rata Rating User |
|---|---|---|---|---|---|---|
| **Member 1** | 5 | 4 | 3 | 4 | - | 4 |
| **Member 2** | 3 | 2 | 4 | 5 | 3 | 3,4 |
| **Member 3** | - | 4 | 2 | - | - | 3 |
| **Member 4** | 3 | 4 | - | - | - | 3,5 |
| **Member 5** | 4 | - | 5 | - | - | 4,5 |
| **Member 6** | 3 | - | 3 | 5 | 4 | 3,75 |
| **Member 7** | 4 | 2 | 3 | 4 | - | 3,25 |

### 3.1.6.4.2. Pembuatan Nilai Kemiripan Antar Item

Tahap ini adalah tahap untuk membuat nilai kemiripan diantara produk yang telah dirating member. Untuk membuat nilai kemiripan, digunakan persamaan *adjusted-cosine*(1), yaitu:

$$sim(i,j) = \frac{\sum_{u \epsilon U}(R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \epsilon U}(R_{u,i} - \bar{R}_u)^2}\sqrt{\sum_{u \epsilon U}(R_{u,j} - \bar{R}_u)^2}} \quad \text{..... (1)}$$

Keterangan:

$sim(i,j)$ = Nilai kemiripan antara *item* i dan *item* j

$u \epsilon U$ = Himpunan member u yang merating *item* i dan *item* j

$R_{u,i}$ = *Rating* member u pada *item* i

$R_{u,j}$ = Rating member u pada *item* j

$\bar{R}_u$ = Nilai rata-rata *rating* member u

Maka berdasarkan data rating yang ada pada tabel 3.8, contoh implementasi dari persamaan 2.1 adalah sebagai berikut:

a. Menghitung kemiripan antara produk 5 dengan produk 6.

b. Himpunan member yang merating produk 5 dan 6: u = {1,2,4,7}

c. Kemudian hitung menurut persamaan 2.1

$$sim(5,6) = \frac{(5-4)(4-4)+\ldots+(4-3{,}25)(2-3{,}25)}{\sqrt{(5-4)^2+\cdots+(4-3{,}25)^2}\sqrt{(4-4)^2+\cdots+(2-3{,}25)^2}} = \text{-}0.230033241860278$$

d. Dari perhitungan diatas diperoleh bahwa nilai kemiripan antara produk 5 dengan produk 6 adalah -0,23003.

e. Selanjutnya seluruh data kemiripan antar produk ini disajikan dalam format tabel seperti dibawah ini :

**Tabel 3.9 Nilai kemiripan antar produk**

| Produk 1 | Produk 2 | Nilai Kemiripan |
|---|---|---|
| 5 | 6 | -0.23003 |
| 5 | 7 | -0.46843 |
| 5 | 8 | -0.31022 |
| 5 | 9 | -0.06859 |
| 6 | 7 | -0.46149 |
| 6 | 8 | -0.9581 |
| 6 | 9 | 1 |
| 7 | 8 | 0.36654 |
| 7 | 9 | -0.9436 |
| 8 | 9 | -0.34195 |

Dalam sistem yang dikembangkan, nilai kemiripan ini tidak disimpan dalam *database*, tetapi dikalkulasi ketika akan melakukan rekomendasi dan hasilnya disimpan dalam suatu *array*. Hal tersebut dilakukan karena nilai kemiripan cenderung terus berubah ketika ada *rating* baru yang masuk.

### 3.1.6.4.3. Pembuatan Prediksi *Rating* Objek

Sesudah nilai kemiripan antar produk didapatkan, maka tahap selanjutnya adalah membuat prediksi *rating* terhadap suatu produk yang belum dirating oleh member. Untuk menghitung prediksi *rating*, digunakan persamaan 2.3 yaitu

persamaan *weighted sum* dengan menerapkan nearest neighbor untuk menentukan jumlah *neighbor* yang digunakan dalam proses perhitungan prediksi.

Berikut persamaan weighted sum :

$$P_{(u,j)} = \frac{\sum_{i \epsilon I_U^K(j)} (R_{u,i} * S_{i,j})}{\sum_{i \epsilon I_U^K(j)} |S_{i,j}|} \quad \text{..... (2.3)}$$

Keterangan :

$P_{(u,j)}$ = Prediksi untuk member u pada item j

$i \epsilon I_U^K(j)$ = Himpunan K item yang mirip dengan item j

$R_{u,i}$ = Rating member u pada item i

$S_{i,j}$ = Nilai kemiripan antara item i dan item j

Contoh implementasi dari persamaan 2.3 dalam menghitung nilai prediksi *rating* untuk setiap *item* yang belum dirating oleh member berdasarkan tabel data *rating* 3.8 adalah sebagai berikut:

a. Prediksi rating untuk member 6 terhadap produk 6
b. Asumsi *nearest neighbor* yang digunakan adalah 3 produk.
c. Berdasarkan tabel nilai kemiripan 3.9, himpunan 3 produk yang paling mirip dengan produk 6 (diurut berdasarkan nilai kemiripan):

   i = {9,5,7}

d. Kemudian hitung menurut persamaan 2.3

$$P_{(6,6)} = \frac{(5 * 1) + (5 * (-0.23003)) + (5 * (-0.46149))}{|1| + |-0.23003| + |-0.46149|}$$

$$= \frac{1.5424}{1.69152}$$

$$= 0.911843$$

e. Maka prediksi *rating* untuk member 6 terhadap produk 6 adalah 0,911842603.

f. Berikut tabel daftar prediksi *rating* untuk seluruh produk yang belum dirating oleh member.

**Tabel 3.10 Hasil prediksi untuk setiap produk yang belum dirating oleh member dengan menggunakan K=3**

| Member | Produk | Prediksi |
|---|---|---|
| 1 | 9 | 2,795 |
| 3 | 5 | -4 |
| 3 | 8 | -3,388 |
| 3 | 9 | -3 |
| 4 | 7 | -3,014 |
| 4 | 8 | -2,548 |
| 4 | 9 | -2,548 |
| 5 | 6 | -1,552 |
| 5 | 8 | 1,492 |
| 5 | 9 | 2,331 |
| 6 | 6 | 0,9118 |
| 7 | 9 | 0,666 |

### 3.1.6.5. Analisis SEO

SEO (*Search Engine Optimization*) adalah sebuah teknik pengoptimalisasian sebuah halaman situs yang berfungsi untuk membangun sebuah situs yang *Search Engine Friendly*. Tujuan dalam penerapan SEO dalam sebuah website yaitu untuk meningkatkan volume dan kualitas trafik dari mesin pencari menuju alamat situs web tertentu dengan memanfaatkan mekanisme kerja alami algoritma mesin pencari tersebut. Perancangan SEO yang akan diimplementasikan pada template website di ambil dari google webmaster dan rahasia *search engine*, perancangan SEO yang diambil antara lain *On Page Optimization* diataranya *kerwords*, *meta tag* dan *content*.

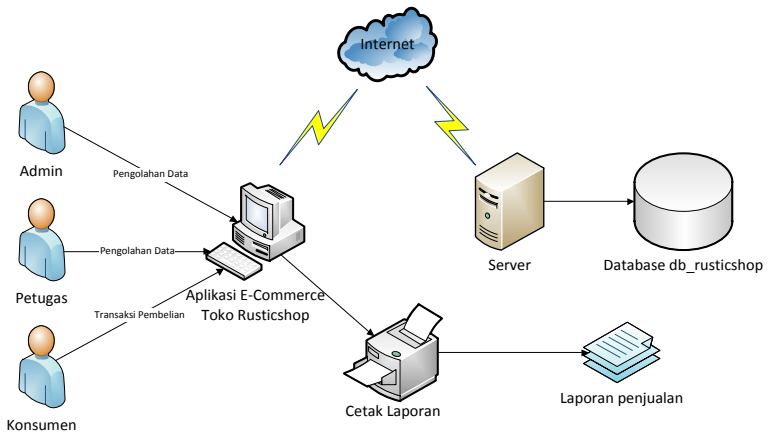
1. *Keywords* adalah satu atau dua buah kata yang menjadi topic atau tema dari suatu *website*.
2. *Meta tags* adalah sekumpulan informasi yang perlu dimasukkan pada halaman *website* diantara bagian *head* didalam *source code*.
3. *Content* merupakan raja dari semua tampilan yang ada di *website*, sehingga mesin pencari bias mengambil situs kita pada halaman pencarian.

### 3.1.7. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk mengetahui proses informasi yang mengalir melalui perangkat lunak. Analisis dilakukan dengan mendeskripsikan fungsi sistem yang dibangun, mengidentifikasi aktor, membuat *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*.

#### 3.1.7.1. Deskripsi Umum Sistem

Sistem yang dibuat merupakan aplikasi *e-commerce* yang digunakan pada toko RusticShop. Sistem yang dibangun dapat melayani member yang hendak melakukan transaksi pembelian produk secara *online*, selain itu juga membantu petugas dalam pengelolaan stok barang yang ada di Toko RusticShop. Sistem yang dibangun akan diimplementasikan pada toko RusticShop yang terhubung langsung kedalam *database* db_rusticshop.




**Gambar 3.4 Perspektif Produk**

Seperti terlihat pada gambar 3.4, petugas dan *admin* terhubung kedalam sistem. Semua data transaksi jual beli disimpan dalam *database* yang hanya dapat diakses oleh petugas dan admin yang ada di toko RusticShop. *Admin* dan dapat mengakses semua fungsionalitas yang terdapat dalam aplikasi yang dibangun dan petugas hanya dapat mengakses fungsionalitas pengolahan data master dan transaksi saja sehingga hasil yang diperoleh berupa informasi dan laporan dari hasil penjualan tiap bulan.

### 3.1.7.2. Identifikasi Aktor

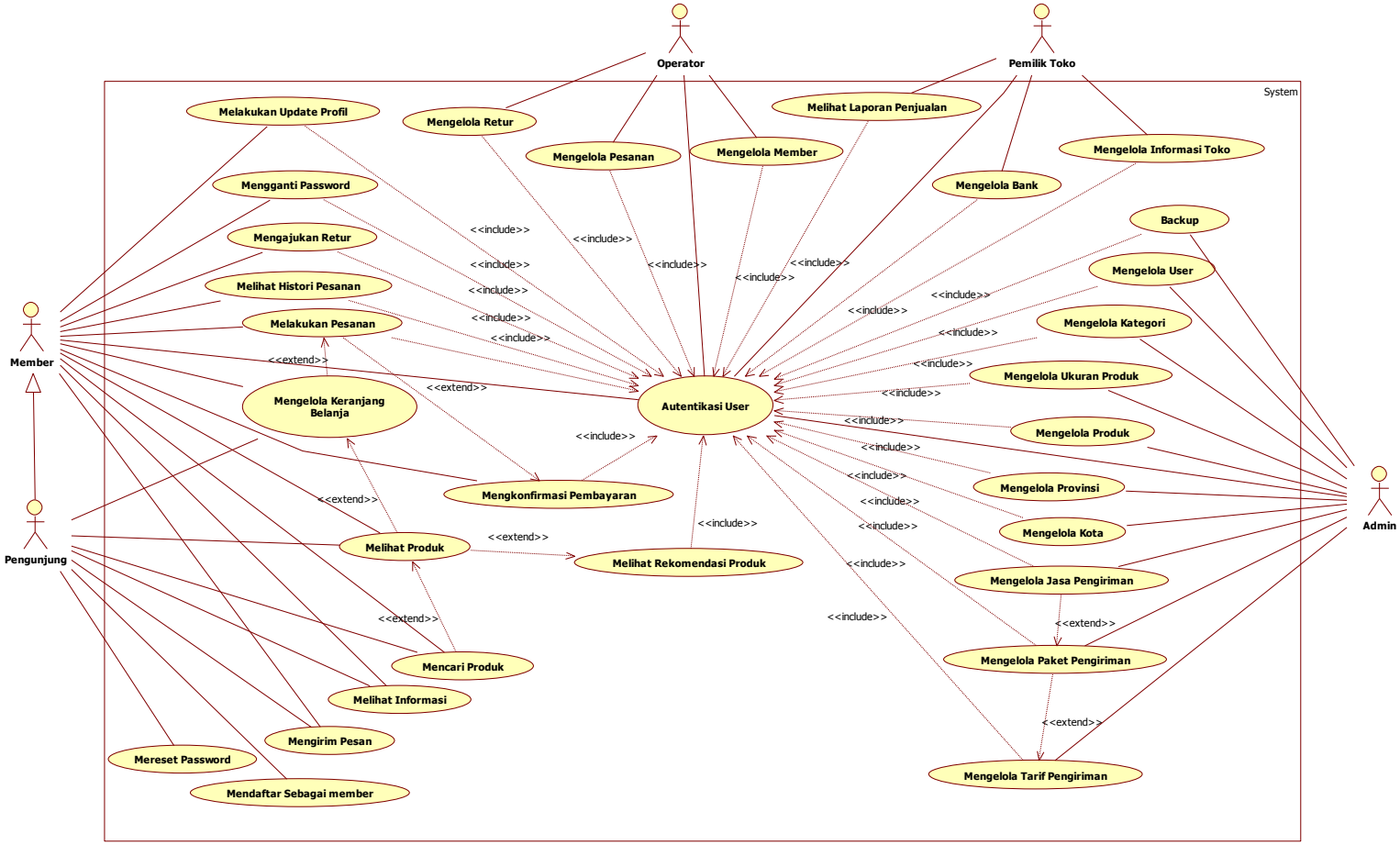
Aplikasi *e-commerce* yang dibangun akan digunakan oleh tiga orang aktor yaitu pemilik toko, *admin*, operator, *member* dan pengunjung. Deskripsi dari masing-masing aktor adalah sebagai berikut :

1. Aktor pertama adalah pemilik toko sebagai pengelola laporan penjualan dan stok barang termasuk menambah, mengubah, dan menghapus *admin*.

2. Aktor kedua adalah *admin* sebagai pengelola yang semua hak akses yang ada pada sistem termasuk untuk menambah, mengubah dan menghapus operator.
3. Aktor ketiga adalah operator yang memiliki hak akses untuk memasukan data barang, mengolah data penjualan dan membuat laporan.
4. Aktor keempat adalah *member* sebagai pengunjung yang memiliki hak untuk melakukan transaksi pembelian.
5. Aktor kelima adalah pengunjung yang memiliki hak akses hanya melihat dan melakukan pencarian produk.

#### 3.1.7.3. *Use Case* Diagram

Pemodelan *use case* adalah pemodelan sistem dari perspektif pandangan pemakai akhir *(end user).* Model *use case* adalah pandangan dari luar sistem, sementara model rancangan adalah pandangan dari dalam. Model *use case* menangkap penggunaan-penggunaan sistem, sedangkan model rancangan merepresentasikan pembangunan dari sistem. *Use case diagram* dari aplikasi *e-commerce* pada Toko RusticShop dapat dilihat pada gambar 3.5




**Gambar 3.5 *Use Case* Diagram aplikasi *e-commerce* pada Toko RusticShop**

### 3.1.7.4. Use Case Scenario

*Use case scenario* berfungsi untuk menjelaskan lebih detail mengenai modul-modul yang terdapat dalam *website e-commerce* pada toko RusticShop. *Scenario diagram* dari use case *e-commerce* pada toko RusticShop adalah sebagai berikut :

**1. Mendaftar sebagai Member**

Interaksi antara aktor dengan *use case* mendaftar sebagai member baru dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.11.

**Tabel 3.11 *Use Case Screnario* Mendaftar Sebagai Member**

| Nama *Use Case* | Mendaftar sebagai member |
|---|---|
| Aktor | Pengunjung |
| Tujuan | Melakukan pendaftaran menjadi member |
| Deskripsi | Proses untuk mendaftar menjadi member |
| Aktor | Sistem |
| 1. Masuk ke halaman utama, klik link daftar untuk mendaftar menjadi member. | |
| | 2. Menampilkan form pendaftaran member. |
| 3. Mengisi data diri pada form pendaftaran | |
| | 4. Pengecekan data pendaftaran, jika email terdaftar maka akan memunculkan pesan *error* jika email masih belum terdaftar maka akan menyimpan data pendaftaran ke dalam *database*. |

**2. Mereset *Password***

Interaksi antara aktor dengan *use case* mereset *password* dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.12.

**Tabel 3.12 *Use Case Scenario* Mereset *Password***

| Nama *Use Case* | Mereset Password |
|---|---|
| Aktor | Pengunjung |
| Tujuan | Mendapatkan *Password* Baru |
| Aktor | Sistem |
| 1. Masuk ke halaman utama, klik link lupa password | |
| | 2. Menampilkan form lupa password. |
| 3. Mengisi data berupa e-mail yang benar. | |
| | 4. Mengecek *e-mail* yang dimasukkan. Jika *e-mail* terdaftar pada sistem maka *e-mail* yang berisi konfirmasi untuk mereset password akan dikirimkan ke alamat e-mail tersebut. |
| 5. Mengecek *e-mail* dan mengklik *link* konfirmasi untuk mereset *password*. | |
| | 6. Mengecek data konfirmasi reset *password*, jika valid maka sistem akan mereset password berdasarkan alamat e-mail yang dikonfirmasi. |

### 3. Autentikasi *User*

Interaksi antara aktor dengan *use case* autentikasi *user* dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.13

**Tabel 3.13 *Use Case Scenario* Autentikasi *User***

| Nama *Use Case* | Pengolahan Data Kategori |
|---|---|
| Aktor | Member, Operator, Admin, Pemilik Toko |
| Tujuan | Melakukan login Member, Operator, Admin, Pemilik toko |
| Aktor | Sistem |
| 1. Mengisi data login | |
| | 2. Pengecekan data login dan masuk ke halaman sesuai dengan hak aksesnya. |

**4. Mengirim Pesan**

Interaksi antara aktor dengan *use case* mengirim pesan dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.14

**Tabel 3.14 *Use Case Scenario* Mengirim Pesan**

| Nama *Use Case* | Mengirim pesan |
|---|---|
| Aktor | Pengunjung, Member |
| Tujuan | Mengirimkan pesan kepada admin |
| Aktor | Sistem |
| 1. Klik menu kontak | |
| | 2. Menampilkan informasi kontak perusahaan dan *form* kontak. |
| 3. Mengisi data pada *form* kontak. | |
| | 4. Pengecekan data kontak, jika valid maka sistem akan mengirimkan e-mail kepada admin. |

**5. Melihat Informasi**

Interaksi antara aktor dengan *use case* Melihat Informasi dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.15

**Tabel 3.15 *Use Case Scenario* Melihat Informasi**

| Nama *Use Case* | Melihat Informasi |
|---|---|
| Aktor | Pengunjung, Member |
| Tujuan | Melihat informasi yang tersedia |
| Aktor | Sistem |
| 1. Klik menu profil | |
| | 2. Menampilkan informasi profil perusahaan. |
| 3. Klik menu Ukuran Produk | |
| | 4. Menampilkan informasi ukuran produk. |

**6. Mencari Produk**

Interaksi antara aktor dengan *use case* mencari produk dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.16

**Tabel 3.16 *Use case scenario* mencari produk**

| Nama *Use Case* | Mencari produk |
|---|---|
| Aktor | Pengunjung, member |
| Tujuan | Melakukan pencarian produk |
| Deskripsi | Proses untuk melakukan pencarian produk |
| Aktor | Sistem |
| 1. Mengisi nama dan kategori produk yang ingin dicari | |
| | 2. Menampilkan data hasil pencarian produk jika tersedia dan menampilkan informasi jika produk tidak tersedia. |

**7. Melihat Produk**

Interaksi antara aktor dengan *use case* melihat produk dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.17

**Tabel 3.17 *Use Case Scenario* Melihat Produk**

| Nama *Use Case* | Melihat Produk |
|---|---|
| Aktor | Pengunjung, Member |
| Tujuan | Melihat produk diskon, terlaris, terbaru dan terpopuler. |
| Deskripsi | Melakukan kegiatan melihat produk diskon, terlaris, terbaru dan terpopuler. |
| Aktor | Sistem |
| 1. Masuk ke halaman utama, mengklik link selengkapnya pada produk diskon, terlaris, terpopuler dan terbaru. | |
| | 2. Menampilkan produk-produk diskon, terlaris, terpopuler, dan terbaru jika tersedia atau jika tidak tersedia akan menampilkan pesan atau informasi “produk tidak |

| | |
|---|---|
| | tersedia". |
| 3. Mengklik nama produk yang diinginkan. | |
| | 4. Menampilkan detail produk. |

**8. Melihat Rekomendasi Produk**

Interaksi antara aktor dengan *use case* melihat rekomendasi produk dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.18.

**Tabel 3.18 *Use Case Scenario* Melihat Rekomendasi Produk**

| Nama *Use Case* | Melihat rekomendasi produk |
|---|---|
| Aktor | Member |
| Tujuan | Melihat daftar rekomendasi produk |
| Deskripsi | Melakukan kegiatan melihat rekomendasi produk |
| Aktor | Sistem |
| 1. Mengklik nama produk yang diinginkan. | |
| | 2. Menampilkan detail produk yang disertai daftar rekomendasi produk jika tersedia dan menampilkan "produk rekomendasi tidak tersedia" jika tidak tersedia. |

**9. Mengelola Keranjang Belanja**

Interaksi antara aktor dengan *use case* keranjang belanja dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.19.

**Tabel 3.19 Use Case Scenario Keranjang Belanja**

| Nama *Use Case* | Keranjang Belanja |
|---|---|
| Aktor | Pengunjung, Member |
| Tujuan | Melakukan update jumlah pesanan dan mengosongkan keranjang. |
| Aktor | Sistem |
| 1. Mengklik salah satu nama produk yang disukai. | |
| | 2. Menampilkan detail produk. |
| 3. Memilih ukuran yang diinginkan | |

| | |
|---|---|
| lalu menekan tombol tambahkan ke keranjang. | |
| | 4. Sistem akan melakukan pengecekan jika stok tersedia sistem akan menyimpan ke keranjang belanja, jika tidak tersedia maka sistem akan menampilkan pesan "Stok tidak mencukupi" |
| 5. Mengklik menu Keranjang Belanja | |
| | 6. Menampilkan halaman keranjang belanja. |
| 7. Mengubah jumlah yang dibeli, kemudian mengklik tombol update keranjang. | |
| | 8. Sistem akan melakukan pengecekan jika stok tersedia sistem akan menyimpan ke keranjang belanja, jika tidak tersedia maka sistem akan menampilkan pesan "Stok tidak mencukupi" |
| 9. Mengklik tombol kosongkan keranjang. | |
| | 10. Sistem akan menghapus produk yang ada pada keranjang user. |

**10. Melakukan Pesanan**

Interaksi antara aktor dengan *use case* melakukan pesanan dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.20.

**Tabel 3.20 *Use Case Scenario* Melakukan Pesanan**

| Nama *Use Case* | Melakukan Pesanan |
|---|---|
| Aktor | *Member* |
| Tujuan | Menampilkan daftar pemesanan dan melakukan pemesanan produk |
| Aktor | Sistem |
| 1. Mengklik menu keranjang belanja. | |
| | 2. Menampilkan data produk yang telah dimasukan ke keranjang belanja beserta jumlahnya. |
| 3. Menekan tombol selanjutnya ke proses pengisian alamat pengiriman | |

| | |
|---|---|
| | 4. Menampilkan halaman form pengisian alamat pengiriman. |
| 5. Mengisi form pengisian alamat pengiriman dengan benar dan lengkap dan menekan tombol menu selanjutnya | |
| | 6. Menampilkan detail halaman pesanan dan pengiriman. |
| 7. Menekan tombol selesai belanja untuk menyelesaikan pemesanan. | |
| | 8. Mengirimkan e-mail notifikasi kemudian menampilkan halaman detail pesanan. |
| 9. Mengklik menu pembelian saya | |
| | 10. Menampilkan daftar pembelian berjalan dan daftar *history* pembelian. |
| 11. Menekan tombol detail pada salah satu pemesanan yang ada di daftar pembelian berjalan. | |
| | 12. Menampilkan detail pemesanan berdasarkan pemesanan yang dipilih. |
| 13. Mengklik menu pembelian saya | |
| | 14. Menampilkan daftar pembelian berjalan dan daftar *history* pembelian. |
| 15. Menekan tombol batal pesan pada salah satu pemesanan yang ada di daftar pembelian berjalan. | |
| | 16. Mengubah status pemesanan menjadi batal. |

**11. Mengkonfirmasi Pembayaran**

Interaksi antara aktor dengan *use case* mengkonfirmasi pembayaran dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.21

**Tabel 3.21 *Use Case Scenario* Mengkonfirmasi Pembayaran**

| | |
|---|---|
| Nama *Use Case* | Mengkonfirmasi Pembayaran |
| Aktor | *Member* |
| Tujuan | Melakukan konfirmasi pembayaran |

| Aktor | Sistem |
|---|---|
| 1. Memilih menu pembelian saya. | |
| | 2. Mengecek pembelian berjalan jika tersedia maka sistem akan menampilkan halaman pembelian saya, jika tidak tersedia maka sistem akan memunculkan pesan tidak ada pembelian berjalan. |
| 3. Menekan tombol konfirmasi pembayaran pada pemesanan yang sedang berjalan. | |
| | 4. Menampilkan form halaman konfirmasi pembayaran beserta total pembelian yang harus dibayarkan. |
| 5. Mengisi data konfirmasi pembayaran dengan lengkap dan benar | |
| | 6. Mengecek data konfirmasi pembayaran jika data yang dimasukan benar maka sistem akan menyimpan ke database, jika data salah maka akan menampilkan pesan error. |

**12. Melihat *History* Pesanan**

Interaksi antara aktor dengan *use case* melihat *history* pesanan dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.22

**Tabel 3.22 *Use Case Scenario* Melihat *History* Pemesanan**

| Nama *Use Case* | Lihat *History* Pemesanan |
|---|---|
| Aktor | *Member* |
| Tujuan | Melihat dan mengecek produk yang dipesan |
| Aktor | Sistem |
| 1. Mengklik menu pembelian saya | |
| | 2. Menampilkan data *history* pemesanan jika tersedia atau menampilkan informasi "data pemesanan tidak tersedia". |
| 3. Mengklik menu pembelian saya | |
| | 4. Menampilkan daftar pembelian berjalan dan daftar *history* |

| | |
|---|---|
| | pembelian. |
| 5. Menekan tombol detail pada salah satu pemesanan yang ada di daftar *history* pembelian. | |
| | 6. Menampilkan detail pemesanan berdasarkan pemesanan yang dipilih. |

**13. Mengajukan Retur**

Interaksi antara aktor dengan *use case* mengajukan retur dalam *use case scenario* yang dapat dilihat pada tabel 3.23

**Tabel 3.23 *Use Case Scenario* Mengajukan Retur**

| Nama *Use Case* | Mengajukan retur |
|---|---|
| Aktor | *Member* |
| Tujuan | Melakukan Pengajuan Retur Produk dan mengirim produk yang diretur |
| Aktor | Sistem |
| 1. Mengklik menu retur | |
| | 2. Menampilkan halaman yang berisi daftar pembelian yang dilakukan. |
| 3. Memilih pesanan mana yang akan diretur produknya, dengan mengklik detail pada salah satu pesanan | |
| | 4. Menampilkan halaman detail pesanan yang berisi list barang yang dibeli. |
| 5. Memilih produk mana yang akan diretur beserta jumlah produk yang akan diretur. Kemudian menekan tombol Ajukan Retur untuk melanjutkan. | |
| | 6. Pengajuan retur akan disimpan ke dalam database dan mengirimkan email notifikasi pengajuan retur sudah disimpan dan sedang direview oleh operator. |
| 7. Mengklik menu retur | |
| | 8. Menampilkan halaman yang berisi daftar pembelian yang dilakukan. |

| | |
|---|---|
| 9. Mengklik link detail pada salah satu pesanan yang ada di daftar retur berjalan. | |
| | 10. Menampilkan halaman detail retur yang berisi status retur dan daftar produk yang diretur. Jika pengajuan retur disetujui maka akan muncul form pengisian no.resi. |
| 11. Mencetak label kemudian mengirimkan produk yang diretur, kemudian mengisi nomor resi pengiriman barang yang diretur. | |
| | 12. Sistem akan menyimpan nomor resi pada pesanan yang diretur. Dan mengirimkan email konfirmasi bahwa produk pengganti akan dikirim setelah produk retur sudah diterima. |
| 13. Mengklik menu retur | |
| | 14. Menampilkan halaman yang berisi daftar pembelian yang dilakukan. |
| 15. Mengklik link batal retur pada salah satu ppesanan yang ada di daftar retur berjalan | |
| | 16. Menghapus data pengajuan retur. |

**14. Melakukan *Update* Profil**

Interaksi antara aktor dengan *use case* melakukan *update* profil dalam *use case scenario* yang dapat dilihat pada tabel 3.24

**Tabel 3.24 *Use Case Scenario* Melakukan *Update* Profil**

| Nama *Use Case* | Melakukan Update Profil |
|---|---|
| Aktor | *Member* |
| Tujuan | Melakukan Update Profil |
| Aktor | Sistem |
| 1. Mengklik menu pengaturan akun | |
| | 2. Menampilkan halaman yang berisi form update profil. |
| 3. Mengisi data akun dengan lengkap dan benar | |
| | 4. Mengecek data akun jika data yang dimasukan benar maka sistem akan |

| | |
|---|---|
| | menyimpan perubahan ke database, jika data salah maka akan menampilkan pesan error. |

**15. Mengganti *Password***

Interaksi antara aktor dengan *use case* ganti *password* dalam *use case scenario* yang dapat dilihat pada tabel 3.25

**Tabel 3.25 *Use Case Scenario* Mengganti *Password***

| Nama *Use Case* | Mengganti *Password* |
|---|---|
| Aktor | *Member* |
| Tujuan | Mengganti *Password* Member |
| Aktor | Sistem |
| 1. Mengklik menu ganti *password* | |
| | 2. Menampilkan halaman yang berisi form ganti *password*. |
| 3. Mengisi data *password* lama dan *password* baru. | |
| | 4. Mengecek data *password* lama, jika password lama benar maka password lama akan diganti dengan password baru. |

**16. Mengelola Retur**

Interaksi antara aktor dengan *use case* mengelola retur dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.26

**Tabel 3.26 *Use Case Scenario* Mengelola Retur**

| Nama *Use Case* | Mengelola pemesanan |
|---|---|
| Aktor | *Operator* |
| Tujuan | Melakukan pengelolaan pemesanan |
| Aktor | Sistem |
| 1. Memilih menu retur pada halaman utama | |
| | 2. Menampilkan halaman retur yang berisi data retur |
| 3. Mengecek data retur, status barang | |

| | |
|---|---|
| yang diretur dan memberikan aksi tentang barang yang diretur. | |
| | 4. Menyimpan data status terbaru kedalam database. |

**17. Mengelola Member**

Interaksi antara aktor dengan *use case* mengelola member dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.27

**Tabel 3.27 *Use Case Scenario* Mengelola Member**

| Nama *Use Case* | Mengelola member |
|---|---|
| Aktor | Operator |
| Tujuan | Melakukan pengelolaan data member |
| Aktor | Sistem |
| 1. Memilih menu member pada halaman utama | |
| | 2. Menampilkan data member |
| 3. Melakukan pengecekan data dan aksi berupa penghapusan data member. | |

**18. Mengelola Pesanan**

Interaksi antara aktor dengan *use case* mengelola pesanan dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.28.

**Tabel 3.28 *Use Case Scenario* Mengelola Pemesanan**

| Nama *Use Case* | Mengelola pesanan |
|---|---|
| Aktor | Operator |
| Tujuan | Melakukan pengelolaan pesanan |
| Aktor | Sistem |
| 1. Memilih menu pesanan | |
| | 2. Menampilkan halaman yang berisi semua daftar pesanan produk. Jika tidak tersedia maka akan memunculkan pesan “Pesanan tidak tersedia” |
| 3. Mengklik link detail pada salah | |

| | |
|---|---|
| satu pesanan yang tersedia | |
| | 4. Menampilkan Halaman Detail Pesanan. |
| 5. Memilih menu pesanan | |
| | 6. Sistem akan menampilkan halaman yang berisi semua daftar pesanan. Jika tidak tersedia maka akan memunculkan pesan “Pesanan tidak tersedia” |
| 7. Mengklik link Pesanan Baru | |
| | 8. Sistem akan menampilkan daftar pesanan yang baru masuk dan belum melakukan pembayaran. Jika tidak tersedia maka akan memunculkan pesan “Pesanan tidak tersedia” |
| 9. Memilih menu pesanan | |
| | 10. Sistem akan menampilkan halaman semua daftar pesanan. Jika tidak tersedia maka akan memunculkan pesan “Pesanan tidak tersedia” |
| 11. Mengklik link Verifikasi Pembayaran | |
| | 12. Sistem akan menampilkan daftar pesanan yang sudah melakukan konfirmasi pembayaran Jika tidak tersedia maka akan memunculkan pesan “Pesanan tidak tersedia” |
| 13. Mengklik link detail pada salah satu pesanan | |
| | 14. Sistem akan menampilkan halaman detail pesanan beserta data konfirmasi pembayaran. |
| 15. Mengecek data pesanan dan konfirmasi pembayaran apakah valid atau tidak | |
| 16. Menekan tombol valid | |
| | 17. Sistem akan mengirimkan email konfirmasi ke member bahwa pembayaran telah lunas dan mengupdate status pembayaran. |
| 18. Menekan tombol tidak valid | |
| | 19. Sistem akan mengirimkan email konfirmasi kepada member bahwa konfirmasi pembayaran yang dilakukan tidak valid. Dan |

| | |
|---|---|
| | menghapus data konfirmasi pembayaran. |
| 20. Memilih menu pesanan | |
| | 21. Sistem akan menampilkan halaman semua daftar pesanan. Jika tidak tersedia maka akan memunculkan pesan "Pesanan tidak tersedia" |
| 22. Mengklik link Kirim Barang | |
| | 23. Sistem akan menampilkan daftar pesanan yang sudah diverifikasi pembayarannya. |
| 24. Mengklik link detail pada salah satu pesanan yang tersedia. | |
| | 25. Menampilkan detail pesanan beserta data alamat tujuan pengiriman barang. |
| 26. Mengklik cetak label | |
| | 27. Melakukan proses cetak label. |
| 28. Mengisi nomor resi pengiriman barang | |
| | 29. Menyimpan nomor resi pengiriman barang berdasarkan pesanan dan mengirimkan email notifikasi kepada member bahwa barang sudah dikirim. |

**19. Melihat Laporan Penjualan**

Interaksi antara aktor dengan *use case* cetak laporan penjualan dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.29

**Tabel 3.29 *Use Case Scenario* Laporan Penjualan**

| Nama *Use Case* | Laporan penjualan |
|---|---|
| Aktor | Pemilik Toko |
| Tujuan | Melihat laporan penjualan |
| Aktor | Sistem |
| 1. Memilih menu laporan pada halaman utama | |
| | 2. Menampilkan data laporan penjualan. |
| 3. Mengklik cetak laporan penjualan | |
| | 4. Sistem akan menampilkan data laporan penjualan dalam format pdf. |

**20. Mengelola Informasi Toko**

Interaksi antara aktor dengan *use case* mengelola informasi toko dalam *use case scenario* yang dapat dilihat pada tabel 3.30

**Tabel 3.30 *Use Case Scenario* Mengelola Informasi Toko**

| Nama *Use Case* | Mengelola Informasi Toko |
|---|---|
| Aktor | Pemilik Toko |
| Tujuan | Melakukan Update Profil |
| Aktor | Sistem |
| 1. Mengklik menu konfigurasi | |
| | 2. Menampilkan halaman yang berisi form informasi toko. |
| 3. Mengisi data informasi toko dengan lengkap dan benar | |
| | 4. Mengecek data informasi toko jika data yang dimasukan benar maka sistem akan menyimpan perubahan ke database, jika data salah maka akan menampilkan pesan error. |

**21. Mengelola Bank**

Interaksi antara aktor dengan *use case* mengelola bank dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.31.

**Tabel 3.31 *Use Case Scenario* Mengelola Bank**

| Nama *Use Case* | Mengelola Bank |
|---|---|
| Aktor | Pemilik Toko |
| Tujuan | Melakukan pengelolaan bank |
| Aktor | Sistem |
| 1. Memilih menu bank pada halaman utama. | |
| | 2. Menampilkan halaman daftar bank. |
| 3. Mengisi data pada form tambah bank | |
| | 4. Mengecek validitas data. Jika data tidak valid maka data harus diisi ulang dengan benar dan jika data valid maka |

| | |
|---|---|
| | sistem akan menyimpan ke *database* |
| 5. Memilih menu ubah pada bank yang ingin di ubah | |
| | 6. Menampilkan form ubah bank |
| 7. Mengisi data yang ingin diubah pada form bank | |
| | 8. Mengecek data masukkan. Jika tidak valid maka data harus diisi kembali dengan benar dan jika valid maka data akan di simpan ke dalam *database*. |
| 9. Memilih bank yang ingin dihapus | |
| | 10. Menampilkan konfirmasi hapus bank |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data bank dari *database* sesuai data yang dipilih. |

**22. *Backup***

Interaksi antara aktor dengan *use case* backup dalam *use case scenario* yang dapat dilihat pada tabel 3.32

**Tabel 3.32 *Use Case Diagram* Backup**

| Nama *Use Case* | Backup |
|---|---|
| Aktor | Admin |
| Tujuan | Backup |
| Aktor | Sistem |
| 1. Mengklik menu backup. | |
| | 2. Menampilkan halaman backup. |
| 3. Mengisi tombol backup. | |
| | 4. Sistem akan melakukan backup database dan source code aplikasi dalam bentuk file zip. |

**23. Mengelola *User***

Interaksi antara aktor dengan *use case* mengelola *user* dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.33

**Tabel 3.33 *Use Case Scenario* Mengelola *User***

| Nama *Use Case* | Mengelola *user* |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan data *user* |
| Aktor | Sistem |
| 1. Memilih menu *user* pada halaman utama | |
| | 2. Menampilkan data *user* |
| 3. Mengisi data form *user* | |
| | 4. Mengecek data masukan. Jika tidak valid maka data harus diisi kembali dan jika valid maka data akan disimpan ke dalam *database*. |
| 5. Memilih menu ubah *user* yang diiginkan | |
| | 6. Menampilkan form ubah *user* |
| 7. Mengisi form ubah *user* yang diinginkan diubah | |
| | 8. Memeriksa data masukan. Jika tidak valid maka data harus diisi kembali dan jika valid maka data akan disimpan ke dalam *database*. |
| 9. Memilih menu hapus pada data yang dipilih | |
| | 10. Menampilkan konfirmasi hapus *user* |
| 11. Menyetujui konfirmasi hapus *user* dengan menekan tombol ok | |
| | 12. Menghapus data dari *database*. |

**24. Mengelola Kategori**

Interaksi antara aktor dengan *use case* mengelola kategori dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.34

**Tabel 3.34 *Use Case Scenario* Mengelola Kategori**

| Nama *Use Case* | Mengelola kategori |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan kategori |
| Aktor | Sistem |
| 1. Memilih menu kategori pada | |

| | |
|---|---|
| halaman utama | |
| | 2. Menampilkan halaman kategori |
| 3. Mengisi data form tambah kategori | |
| | 4. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam database. |
| 5. Memilih menu ubah pada kategori yang ingin diubah | |
| | 6. Menampilkan form ubah kategori berisi data kategori yang diubah |
| 7. Mengisi data kategori yang ingin diubah | |
| | 8. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam database. |
| 9. Memilih menu hapus pada data kategori yang ingin dihapus s | |
| | 10. Menampilkan konfirmasi hapus kategori |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data kategori dari database. |

**25. Mengelola Ukuran Produk**

Interaksi antara aktor dengan *use case* mengelola ukuran produk dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.35

**Tabel 3.35 *Use Case Scenario* Mengelola Ukuran Produk**

| Nama *Use Case* | Mengelola ukuran produk |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan ukuran produk |
| Aktor | Sistem |
| 1. Memilih menu ukuran produk pada halaman utama | |
| | 2. Menampilkan halaman ukuran |

| | |
|---|---|
| | produk |
| 3. Mengisi data form tambah ukuran produk | |
| | 4. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 5. Memilih menu ubah pada ukuran produk yang ingin diubah | |
| | 6. Menampilkan form ubah ukuran produk berisi data ukuran produk yang diubah |
| 7. Mengisi data ukuran produk yang ingin diubah | |
| | 8. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 9. Memilih menu hapus pada data ukuran produk yang ingin dihapus | |
| | 10. Menampilkan konfirmasi hapus ukuran produk |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data ukuran produk dari *database*. |

**26. Mengelola Produk**

Interaksi antara aktor dengan *use case* mengelola produk dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.36

**Tabel 3.36 *Use Case Scenario* Mengelola Produk**

| Nama *Use Case* | Mengelola Produk |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan produk |
| Aktor | Sistem |
| 1. Memilih menu produk pada halaman utama. | |
| | 2. Menampilkan halaman produk yang |

| | |
|---|---|
| | berisi data produk |
| 3. Mengisi data pada form produk | |
| | 4. Mengecek validitas data. Jika data tidak valid maka data harus diisi ulang dengan benar dan jika data valid maka sistem akan menyimpan ke *database* |
| 5. Memilih menu ubah produk yang ingin di ubah | |
| | 6. Menampilkan form ubah produk |
| 7. Mengisi data yang ingin diubah pada form produk | |
| | 8. Mengecek data masukkan.jika tidak valid maka data harus diisi kembali dengan benar dan jika valid maka data akan di simpan ke dalam *database*. |
| 9. Memilih produk yang ingin dihapus | |
| | 10. Menampilkan konfirmasi hapus produk |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data produk yang dipilih dari database. |

**27. Mengelola Provinsi**

Interaksi antara aktor dengan use case mengelola provinsi dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.37.

**Tabel 3.37 *Use Case Scenario* Mengelola Provinsi**

| Nama *Use Case* | Mengelola Provinsi |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan provinsi |
| Aktor | Sistem |
| 1. Memilih menu provinsi pada halaman utama. | |
| | 2. Menampilkan halaman provinsi yang berisi data provinsi |
| 3. Mengisi data pada form provinsi | |
| | 4. Mengecek validitas data. Jika data |

| | |
|---|---|
| | tidak valid maka data harus diisi ulang dengan benar dan jika data valid maka sistem akan menyimpan ke database |
| 5. Memilih menu ubah provinsi yang ingin di ubah | |
| | 6. Menampilkan form ubah provinsi |
| 7. Mengisi data yang ingin diubah pada form provinsi | |
| | 8. Mengecek data masukkan.jika tidak valid maka data harus diisi kembali dengan benar dan jika valid maka data akan di simpan ke dalam database. |
| 9. Memilih provinsi yang ingin dihapus | |
| | 10. Menampilkan konfirmasi hapus provinsi |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data provinsi yang dipilih dari database. |

**28. Mengelola Kota**

Interaksi antara aktor dengan *use case* mengelola kota dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.38

**Tabel 3.38 *Use Case Scenario* Mengelola Kota**

| Nama *Use Case* | Mengelola kota |
|---|---|
| Aktor | *Operator* |
| Tujuan | Melakukan pengelolaan kota |
| Aktor | Sistem |
| 1. Memilih menu kota pada halaman utama | |
| | 2. Menampilkan halaman kota yang berisi data kota |
| 3. Memilih provinsi dan mengisi data nama kota pada form kota | |
| | 4. Mengecek data masukan. Jika tidak valid maka data harus diisi kembali dan jika valid maka data akan |

| | |
|---|---|
| | disimpan ke dalam *database*. |
| 5. Memilih kota yang ingin diubah | |
| | 6. Manampilkan form ubah yang berisi data kota |
| 7. Mengubah data kota | |
| | 8. Mengecek data masukan. Jika daa tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam database. |
| 9. Memilih menu hapus pada data kota yang ingin dihapus | |
| | 10. Menampilkan konfirmasi hapus kota |
| 11. Menyetujui hapus kota dengan menekan tombol ok | |
| | 12. Menghapus data kota dari database. |

**29. Mengelola Jasa Pengiriman**

Interaksi antara aktor dengan *use case* mengelola jasa pengiriman dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.39

**Tabel 3.39 *Use Case Scenario* Mengelola Jasa Pengiriman**

| Nama *Use Case* | Mengelola jasa pengiriman |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan jasa pengiriman |
| Aktor | Sistem |
| 1. Memilih menu jasa pengiriman pada halaman utama | |
| | 2. Menampilkan halaman jasa pengiriman |
| 3. Mengisi data form jasa pengiriman | |
| | 4. Mengecek data masukan. Jika data tidak valid maka form harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 5. Memilih menu ubah pada jasa pengiriman yang diinginkan | |
| | 6. Menampilkan form ubah jasa pengiriman yang berisi data |

| | |
|---|---|
| 7. Mengisi data pada form ubah jasa pengiriman yang ingin diubah | |
| | 8. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam database. |
| 9. Memilih menu hapus pada data jasa pengiriman yang akan dihapus | |
| | 10. Menampilkan konfirmasi hapus jasa pengiriman |
| 11. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 12. Menghapus data dari database |

**30. Mengelola Paket Pengiriman**

Interaksi antara aktor dengan *use case* mengelola paket pengiriman dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.40

**Tabel 3.40 *Use Case Scenario* Mengelola Paket Pengiriman**

| Nama *Use Case* | Mengelola paket pengiriman |
|---|---|
| Aktor | *Admin* |
| Tujuan | Melakukan pengelolaan paket pengiriman |
| Aktor | Sistem |
| 1. Memilih menu jasa pengiriman pada halaman utama | |
| | 2. Menampilkan halaman jasa pengiriman |
| 3. Mengklik tombol paket pengiriman pada salah satu jasa pengiriman yang tersedia. | |
| | 4. Menampilkan halaman paket pengiriman berdasarkan jasa pengiriman yang dipilih. |
| 5. Mengisi data form paket pengiriman | |
| | 6. Mengecek data masukan. Jika data tidak valid maka form harus diisi kembali dan jika data valid maka data akan disimpan ke dalam database. |

| | |
|---|---|
| 7. Memilih menu ubah pada paket pengiriman yang diinginkan | |
| | 8. Menampilkan form ubah paket pengiriman yang berisi data |
| 9. Mengisi data pada form ubah paket pengiriman yang ingin diubah | |
| | 10. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 11. Memilih menu hapus pada data paket pengiriman yang akan dihapus | |
| | 12. Menampilkan konfirmasi hapus paket pengiriman |
| 13. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 14. Menghapus data dari *database* |

**31. Mengelola Ongkos Kirim**

Interaksi antara aktor dengan *use case* mengelola ongkos kirim dijelaskan dalam *use case scenario* yang dapat dilihat pada tabel 3.41

**Tabel 3.41 *Use Case Scenario* Mengelola Ongkos Kirim**

| Nama *Use Case* | Mengelola ongkos kirim |
|---|---|
| Aktor | *Operator* |
| Tujuan | Melakukan pengelolaan ongkos kirim |
| Aktor | Sistem |
| 1. Memilih menu jasa pengiriman pada halaman utama | |
| | 2. Menampilkan halaman jasa pengiriman |
| 3. Memilih paket pengiriman pada salah satu jasa pengiriman | |
| | 4. Menampilkan halaman paket pengiriman berdasarkan jasa pengiriman yang dipilih |
| 5. Memilih ongkos kirim pada salah satu paket pengiriman | |

| | |
|---|---|
| | 6. Menampilkan halaman ongkos kirim berdasarkan paket pengiriman yang dipilih. |
| 7. Mengisi data form ongkos kirim | |
| | 8. Mengecek data masukan. Jika data tidak valid maka form harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 9. Memilih menu ubah pada ongkos kirim yang diinginkan | |
| | 10. Menampilkan form ubah ongkos kirim yang berisi data |
| 11. Mengisi data pada form ubah ongkos kirim yang ingin diubah | |
| | 12. Mengecek data masukan. Jika data tidak valid maka data harus diisi kembali dan jika data valid maka data akan disimpan ke dalam *database*. |
| 13. Memilih menu hapus pada data ongkos kirim yang akan dihapus | |
| | 14. Menampilkan konfirmasi hapus ongkos kirim |
| 15. Menyetujui konfirmasi dengan menekan tombol ok | |
| | 16. Menghapus data dari database |

### 3.1.7.5. *Activity* Diagram

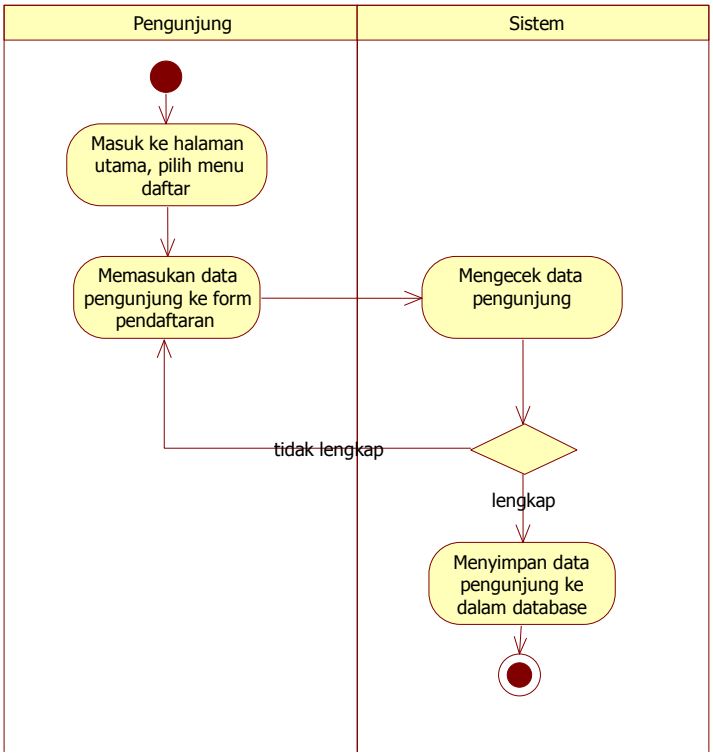
*Activity diagram* (diagram aktivitas) adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*).

1. ***Activity* Diagram Mendaftar sebagai member**

   Deskripsi :

   Pengunjung masuk ke halaman utama dan memilih menu daftar, sistem akan menampilkan halaman pendaftaran. Pengunjung mengisi form

pendaftaran dengan lengkap dan benar lalu sistem akan menyimpan ke database.

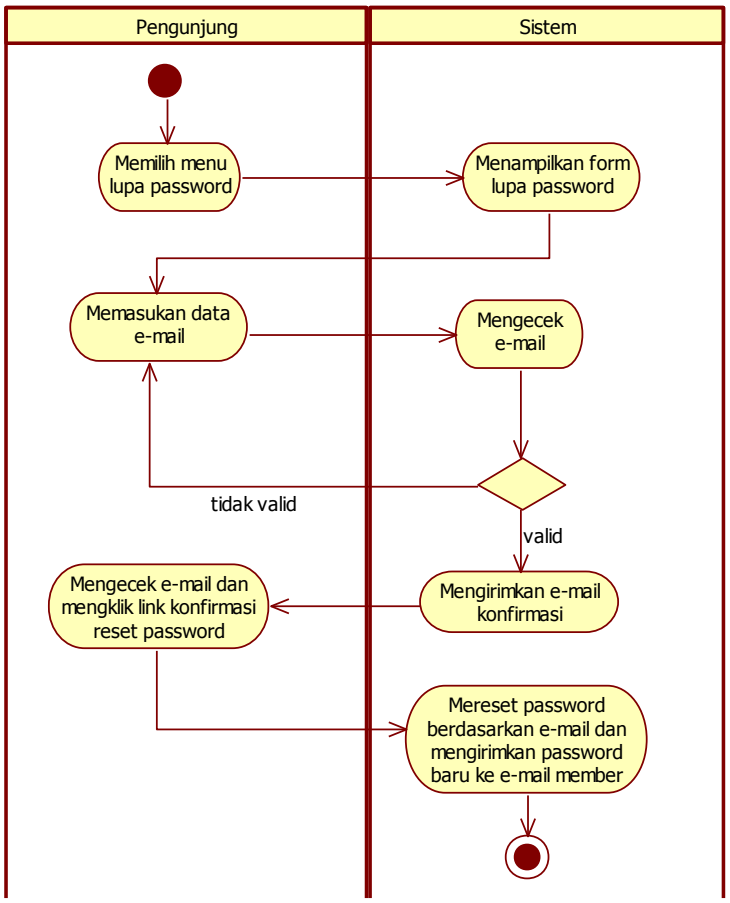



**Gambar 3.6 *Activity* Diagram Mendaftar sebagai member**

**2. *Activity* Diagram Mereset *Password***

Deskripsi :

Member memilih fasilitas lupa password dan sistem akan menampilkan form lupa password. Member memasukkan data berupa e-mail untuk diproses oleh sistem sehingga password baru akan dikirimkan ke e-mail tersebut.

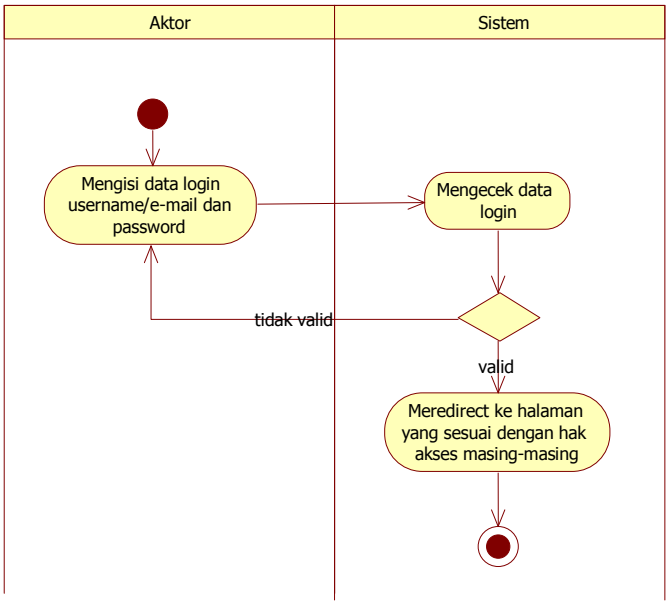



**Gambar 3.7 *Activity* Diagram Mereset *Password***

### 3. *Activity* Diagram Autentikasi *User*

Deskripsi:

Pengunjung, member, operator, *admin*, maupun pemilik toko mengisi data login berupa *username*/*e-mail* dan *password* pada halaman login. Sistem akan mengecek apakah user tersedia pada *database*, jika tersedia dan *password* yang dimasukkan benar maka akan masuk ke halaman *member area*/*panel admin*.

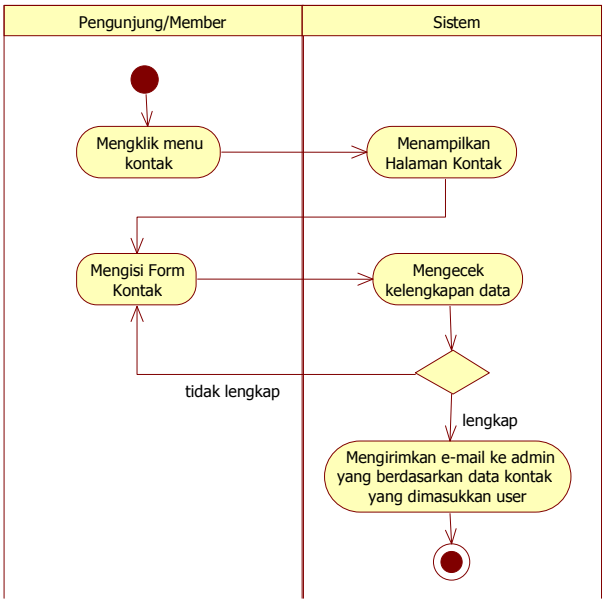



**Gambar 3.8 *Activity* Diagram Autentikasi *User***

**4. *Activity* Diagram Mengirim Pesan**

Deskripsi:

Pengunjung/member mengirimkan pesan kepada admin website dengan melengkapi form kontak.

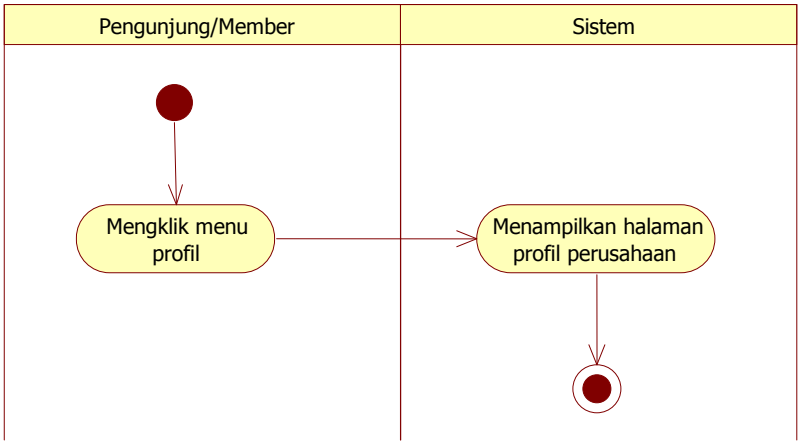



**Gambar 3.9 *Activity* Diagram Mengirim Pesan**

### 5. *Activity* Diagram Melihat Informasi

Deskripsi:

Pengunjung atau member mengklik menu profil kemudian sistem akan menampilkan halaman profil perusahaan.

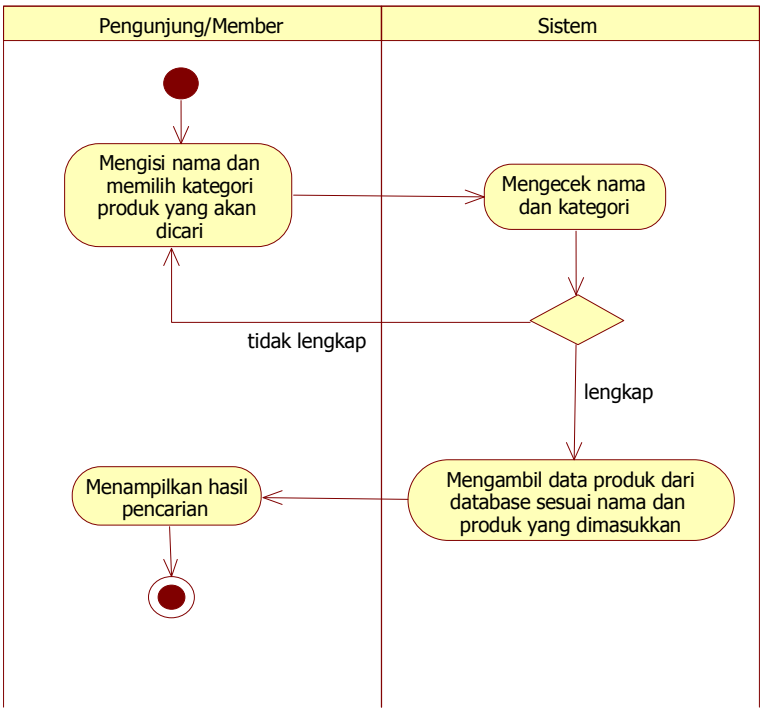



**Gambar 3.10 *Activity* Diagram Melihat Informasi**

**6. *Activity* Diagram Mencari Produk**

Deskripsi :

Pengunjung atau member mengisi data pencarian produk yang diinginkan pada form pencarian. Sistem akan mengecek data yang tersedia pada *database* jika tersedia maka produk akan di tampilkan jika tidak ada maka sistem akan menampilkan informasi bahwa produk tidak tersedia.



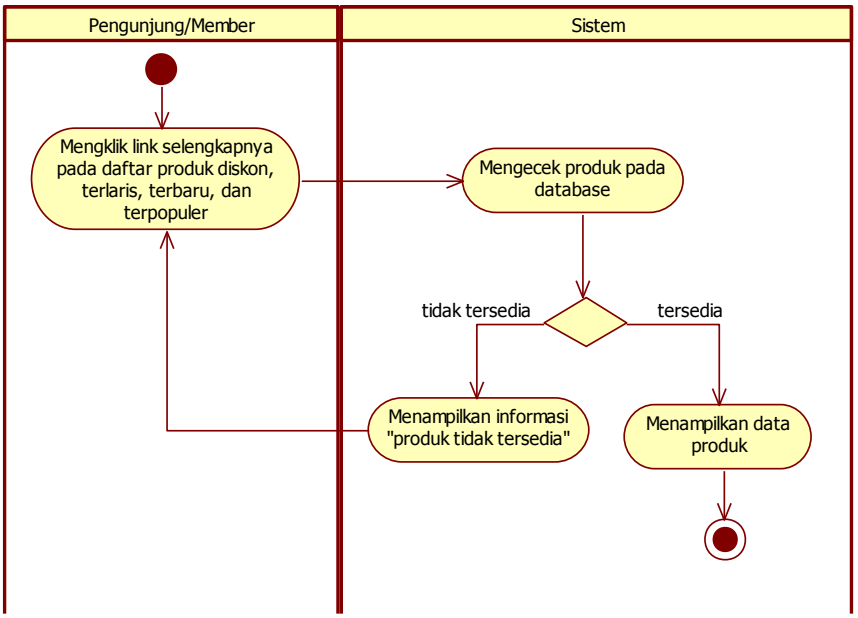

**Gambar 3.11 *Activity* Diagram Mencari Produk**

**7. *Activity* Diagram Melihat Produk**

Deskripsi :

Pengunjung dan member masuk ke halaman utama dan mengklik link selengkapnya pada daftar produk terlaris, terbaru terpopuler atau diskon. Sistem akan menampilkan halaman yang diinginkan pengunjung

atau member jika produk tersedia dan jika produk tidak tersedia maka sistem akan menampilkan informasi bahwa produk tidak tersedia.

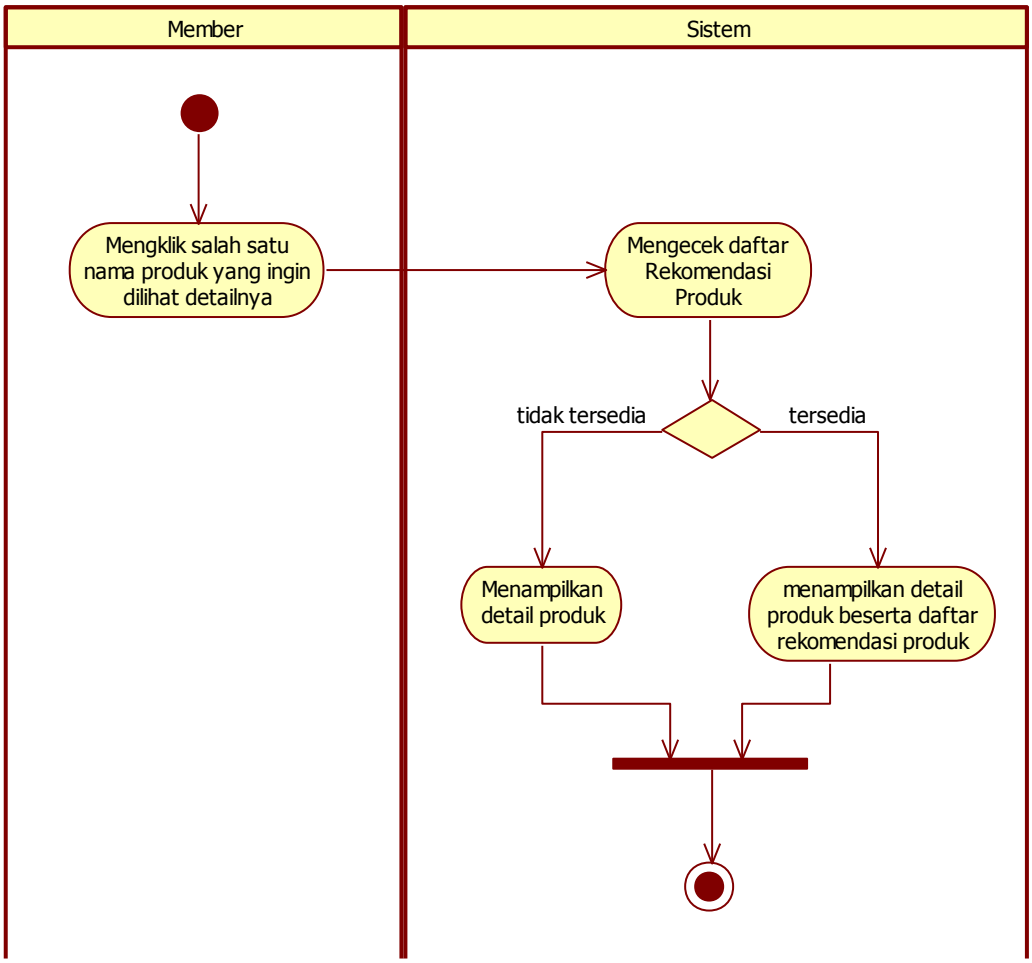



**Gambar 3.12 *Activity* Diagram Melihat Produk**

**8. *Activity* Diagram Melihat Rekomendasi Produk**

Deskripsi:

Member melihat detail produk kemudian sistem akan memberikan daftar rekomendasi jika tersedia.

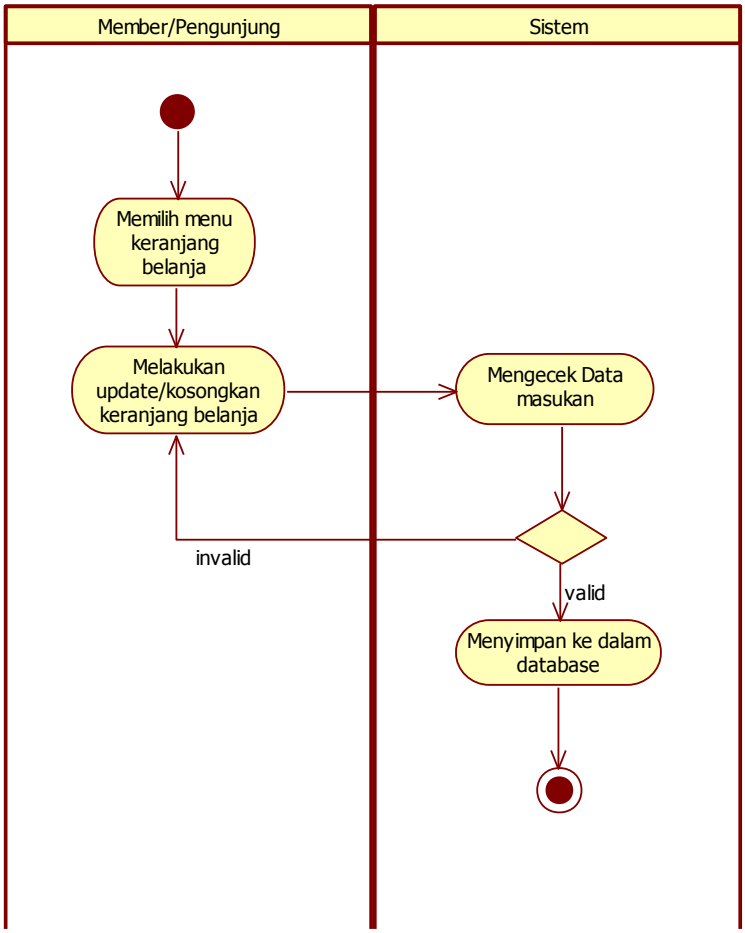



**Gambar 3.13 *Activity* Diagram Melihat Rekomendasi Produk**

**9. *Activity* Diagram Mengelola Keranjang Belanja**

Deskripsi:

Pengunjung atau Member masuk ke halaman ke halaman keranjang belanja. Pengunjung atau Member dapat melakukan update atau kosongkan keranjang belanja.

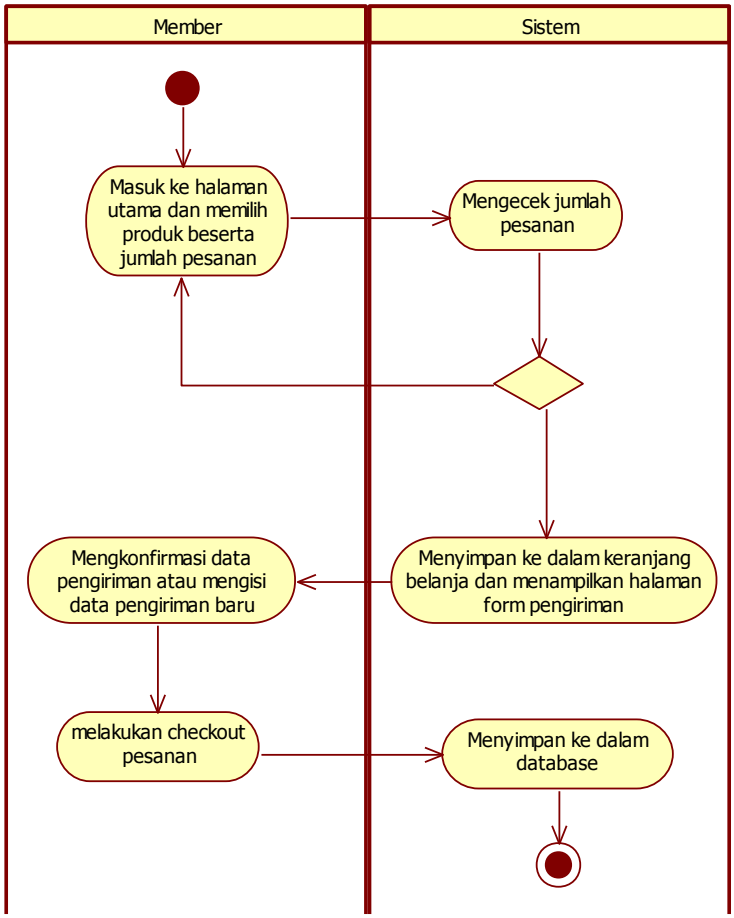



**Gambar 3.14 *Activity* Diagram Keranjang Belanja**

**10. *Activity* Diagram Melakukan Pesanan**

Deskripsi :

Member melakukan pemesanan dengan memasukkan jumlah produk yang dipesan, selanjutnya memasukkan alamat pengiriman pemesanan dan mengakhiri pemesanan dengan menekan menu checkout.

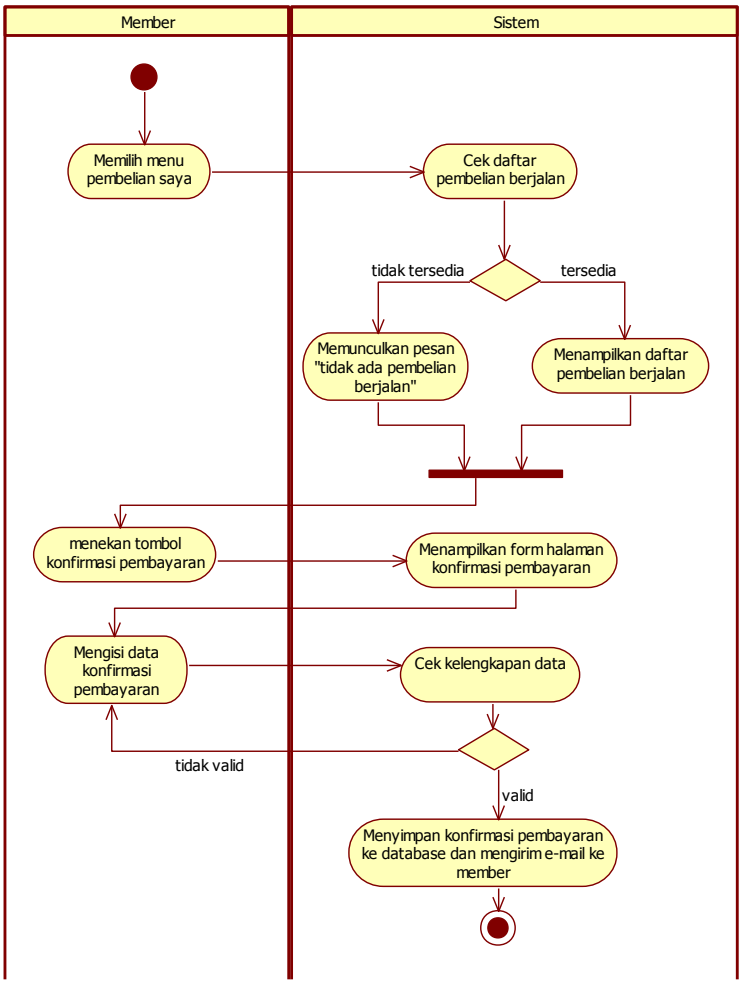



**Gambar 3.15 *Activity* Diagram Melakukan Pesanan**

### 11. *Activity* Diagram Mengkonfirmasi Pembayaran

Deskripsi :

Member memilih menu konfirmasi pembayaran, lalu sistem akan menampilkan form konfirmasi pembayaran. Member mengisi data konfirmasi dengan lengkap dan benar, lalu sistem akan menyimpan data tersebut ke dalam database.

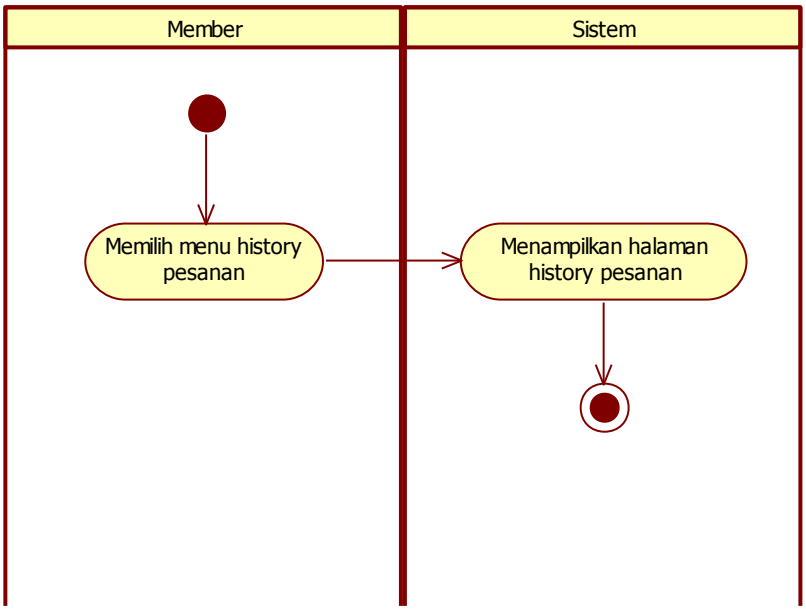



**Gambar 3.16 *Activity* Diagram Konfirmasi Pembayaran**

### 12. *Activity* Diagram Melihat *History* Pesanan

Deskripsi :

Member dapat melihat history pemesanannya di menu history untuk melihat status pemesanannya.

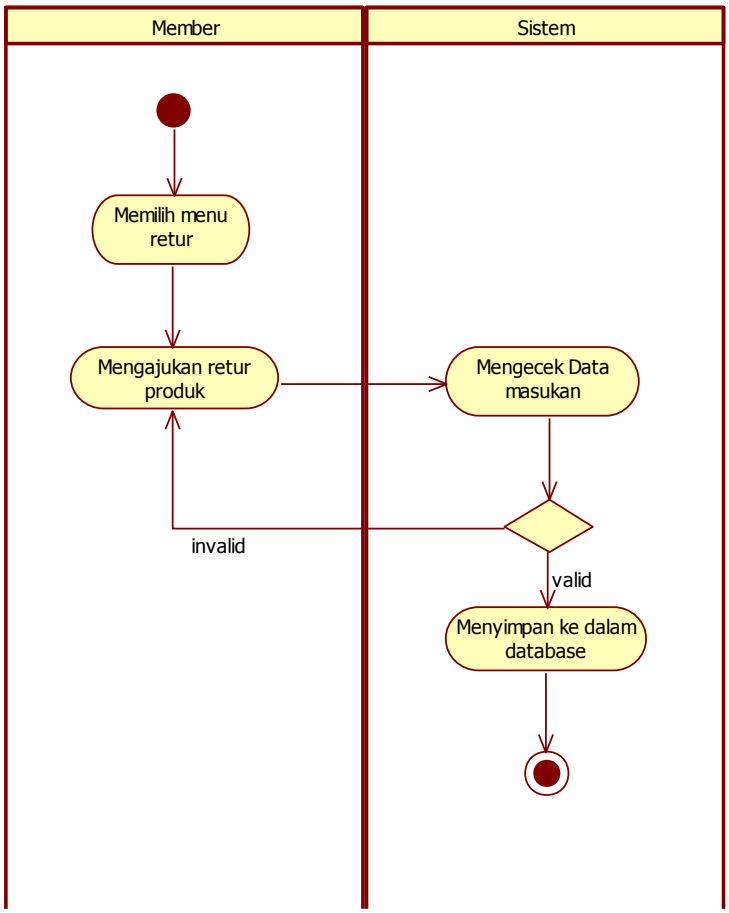



**Gambar 3.17 Activity Diagram Melihat History Pesanan**

**13. *Activity* Diagram Mengajukan Retur**

Deskripsi:

Member memilih menu retur, kemudian member dapat melakukan pengajuan retur produk dari barang yang sudah dibeli.

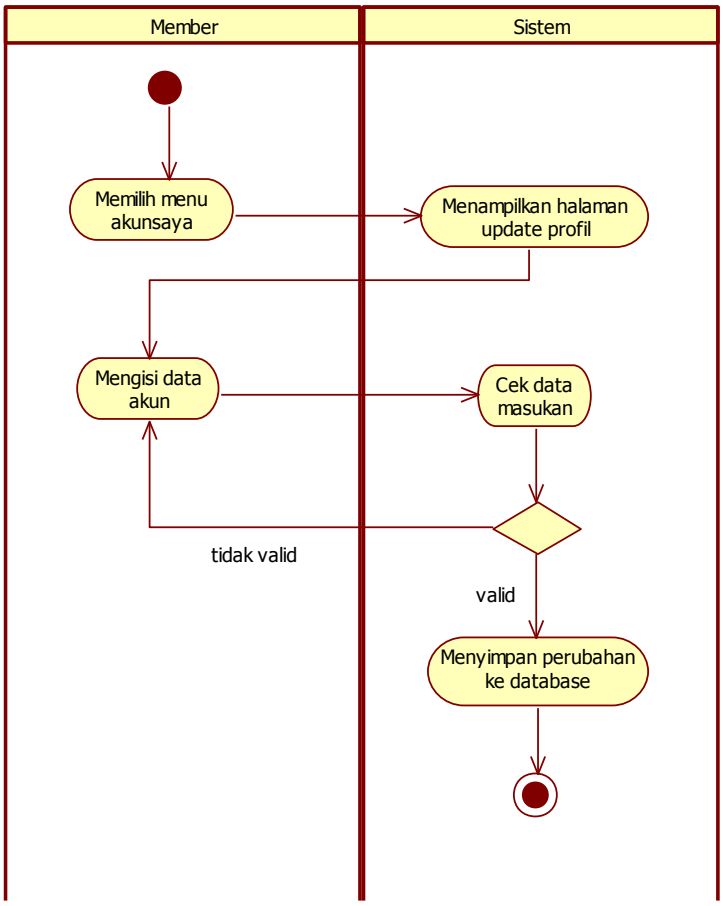



**Gambar 3.18 *Activity* Diagram Mengajukan Retur**

### 14. *Activity* Diagram Melakukan *Update* Profil

Deskripsi :

Member memilih menu info member untuk mengolah akunnya. Sistem akan menampilkan berupa form yang berisi data member dan member dapat langsung mengubah data tersebut.

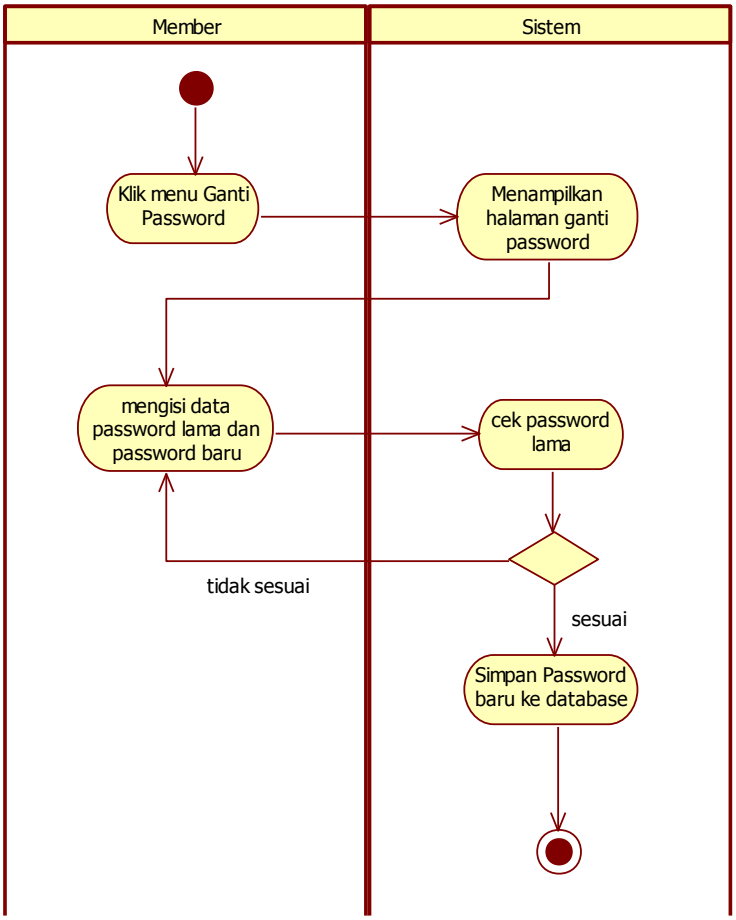



**Gambar 3.19 Activity Diagram Update Profil**

**15. *Activity* Diagram Mengganti *Password***

Deskripsi:

Member memilih menu ganti password, sistem akan menampilkan form ganti password yang terdiri dari password lama, password baru, dan ulangi password baru.

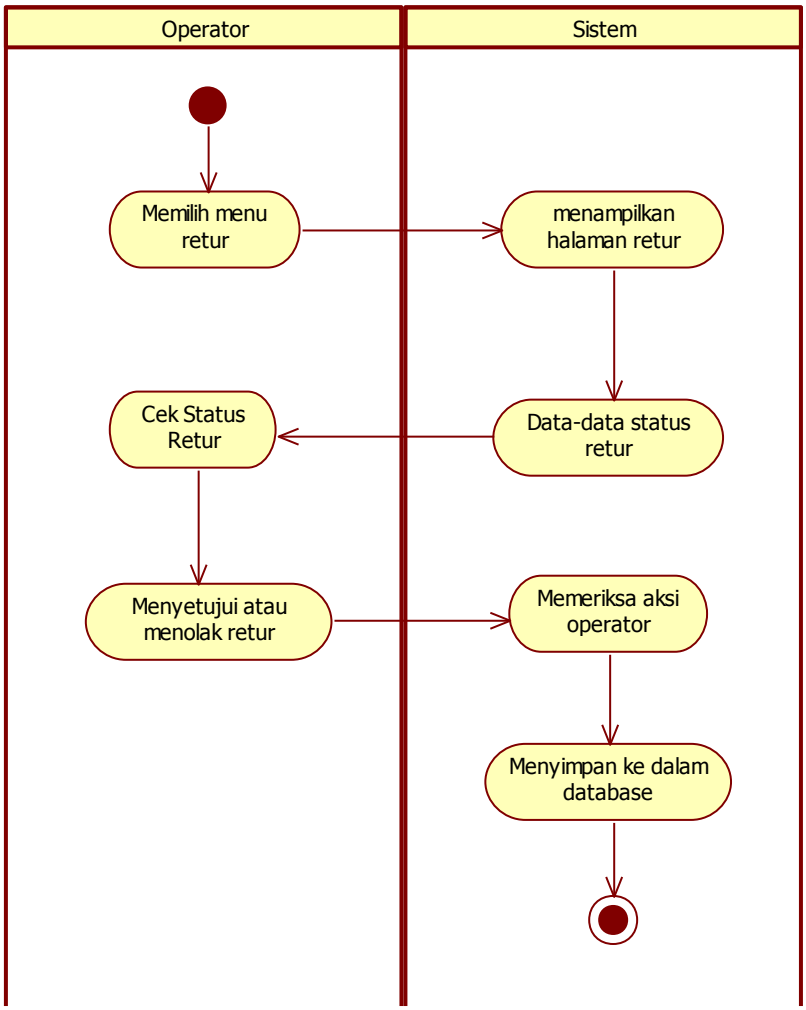



**Gambar 3.20 *Activity* Diagram Ganti *Password***

### 16. *Activity* Diagram Mengelola Retur

Deskripsi :

Operator masuk ke halaman retur, kemudian sistem akan menampilkan halaman retur. Operator dapat melakukan aksi menyetujui atau menolak pengajuan retur.

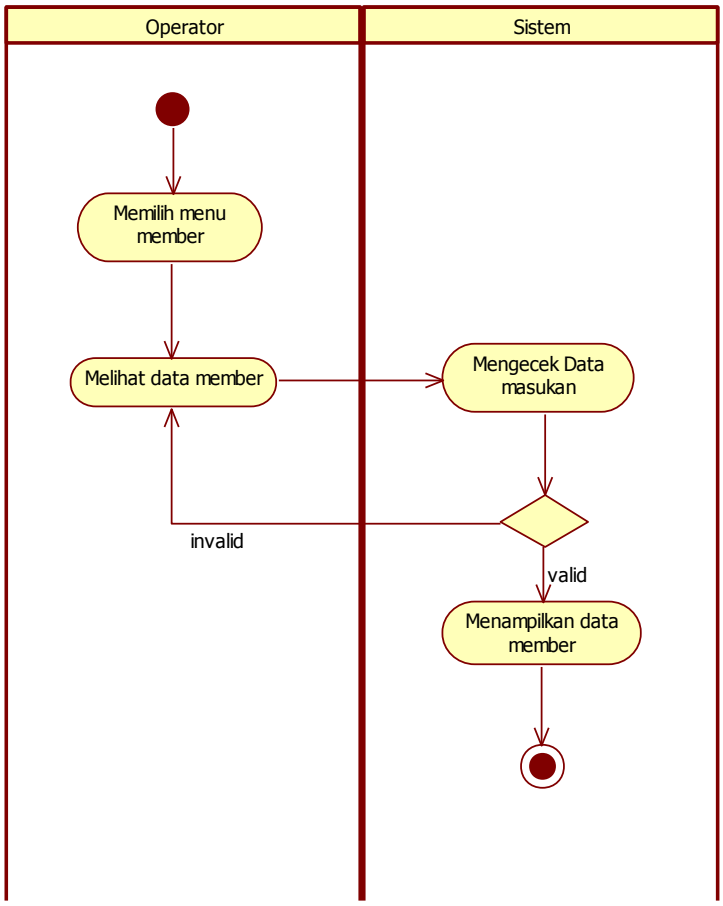



**Gambar 3.21 *Activity* Diagram Mengelola Retur**

### 17. *Activity* Diagram Mengelola Member

Deskripsi :

Operator memilih menu member. operator dapat melihat data member.

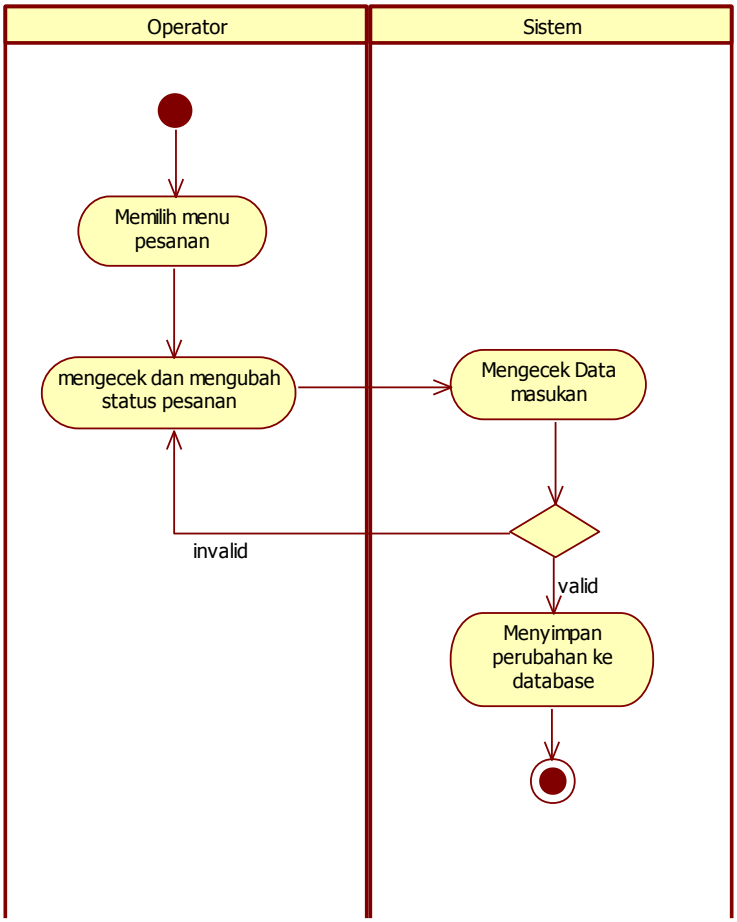



**Gambar 3.22 Activity Diagram Mengelola Data Member**

### 18. *Activity* Diagram Mengelola Pesanan

Deskripsi :

Operator memilih menu pesanan. operator dapat melakukan pengelolaan pemesanan yang dilakukan member dengan mengupdate status pesanan.

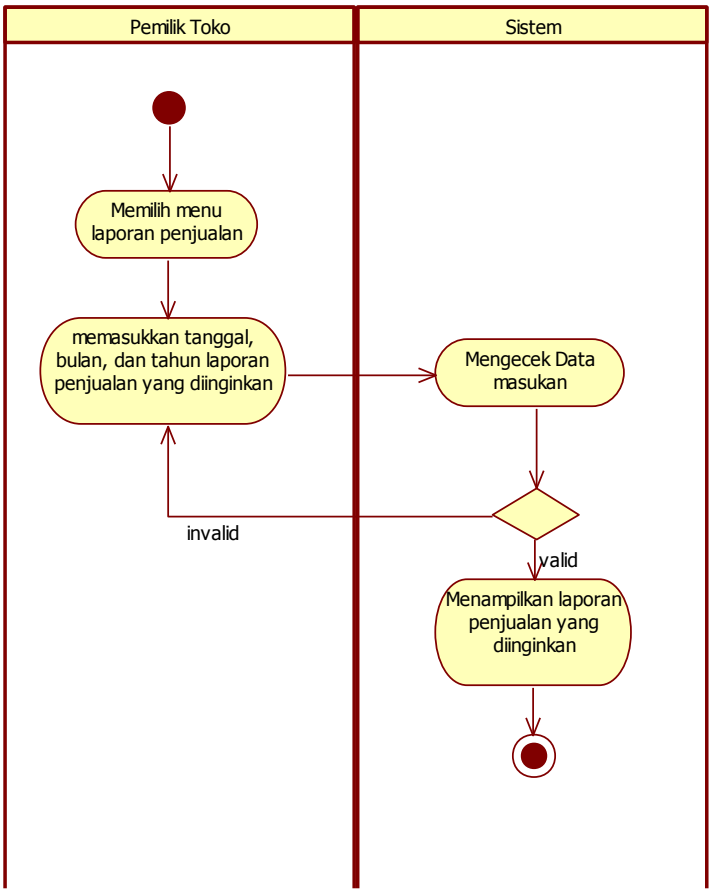



**Gambar 3.23 Activity Diagram Mengelola Pesanan**

**19. *Activity* Diagram Melihat Laporan Penjualan**

Deskripsi :

Pemilik toko memilih menu laporan untuk masuk ke halaman laporan penjualan. Pemilik toko dapat mencetak laporan sesuai dengan periode yang diinginkan. Sistem akan menyediakan fungsi pembuatan laporan yang diinginkan sesuai dengan tanggal, bulan dan tahun yang tersedia pada sistem.

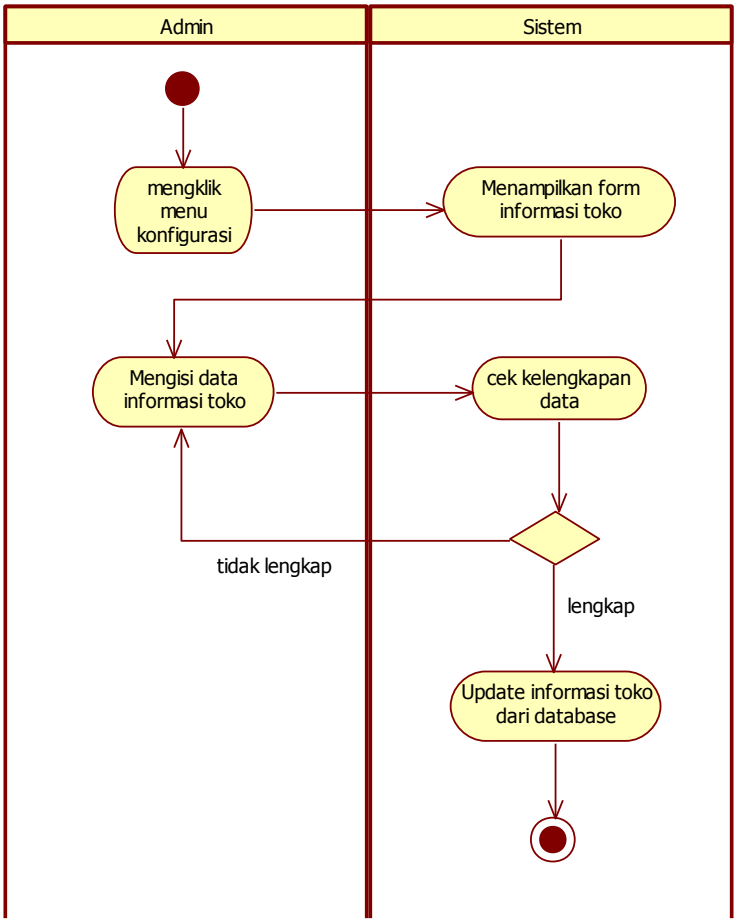



**Gambar 3.24 *Activity* Diagram Laporan Penjualan**

### 20. *Activity* Diagram Mengelola Informasi Toko

Deskripsi:

*Admin* memilih menu konfigurasi, sistem kemudian akan menampilkan form update informasi toko. *Admin* mengubah data informasi toko, sistem kemudian akan mengecek kelengkapan data, jika lengkap maka perubahan informasi toko akan disimpan ke *database*.

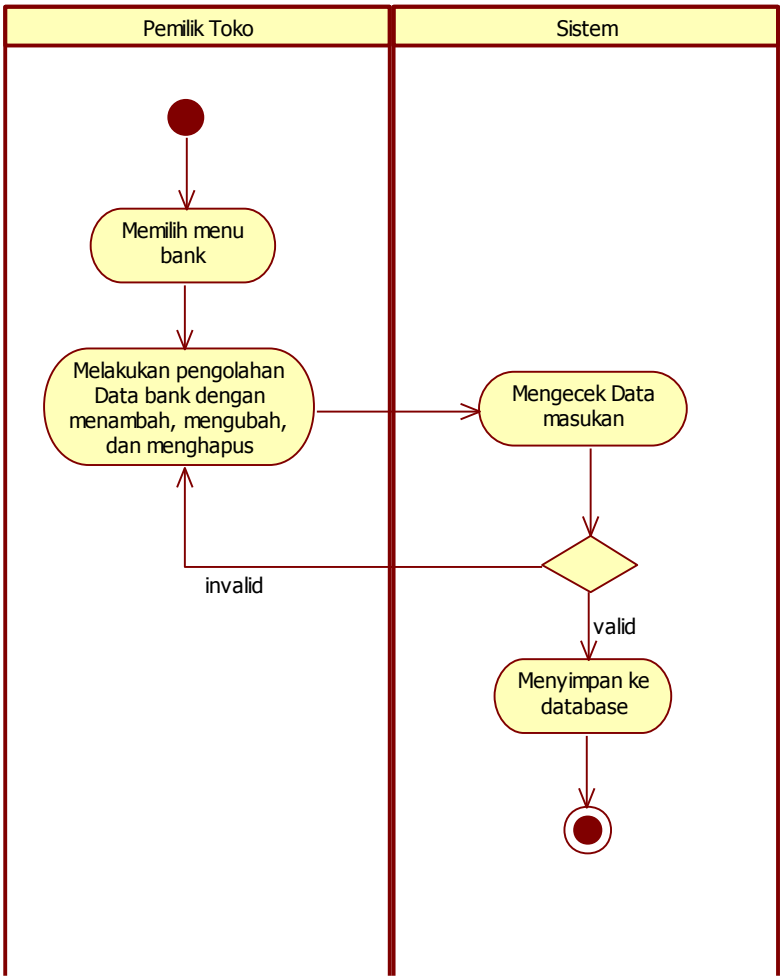



**Gambar 3.25 *Activity* Diagram Mengelola Informasi Toko**

**21. *Activity* Diagram Mengelola Bank**

Deskripsi :

Pemilik toko masuk ke halaman *bank*. Pemilik toko dapat melakukan penambahan data *bank*, pengubahan data yang diinginkan, dan menghapus data bank.

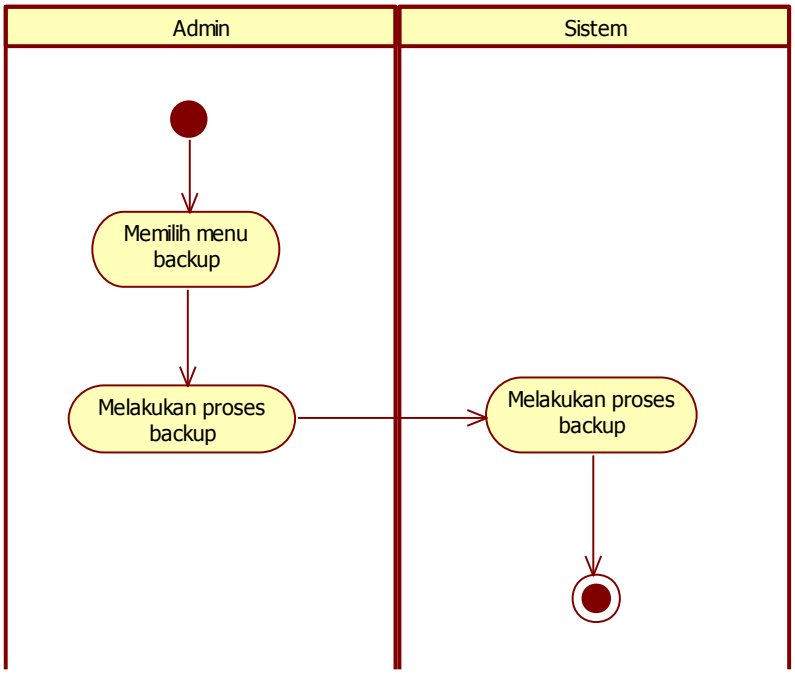



**Gambar 3.26 *Activity* Diagram Mengelola *Bank***

**22. *Activity* Diagram Backup**

Deskripsi :

*Admin* masuk ke halaman *backup*. *Admin* dapat melakukan proses *backup source code*, *database*, dan gambar produk.

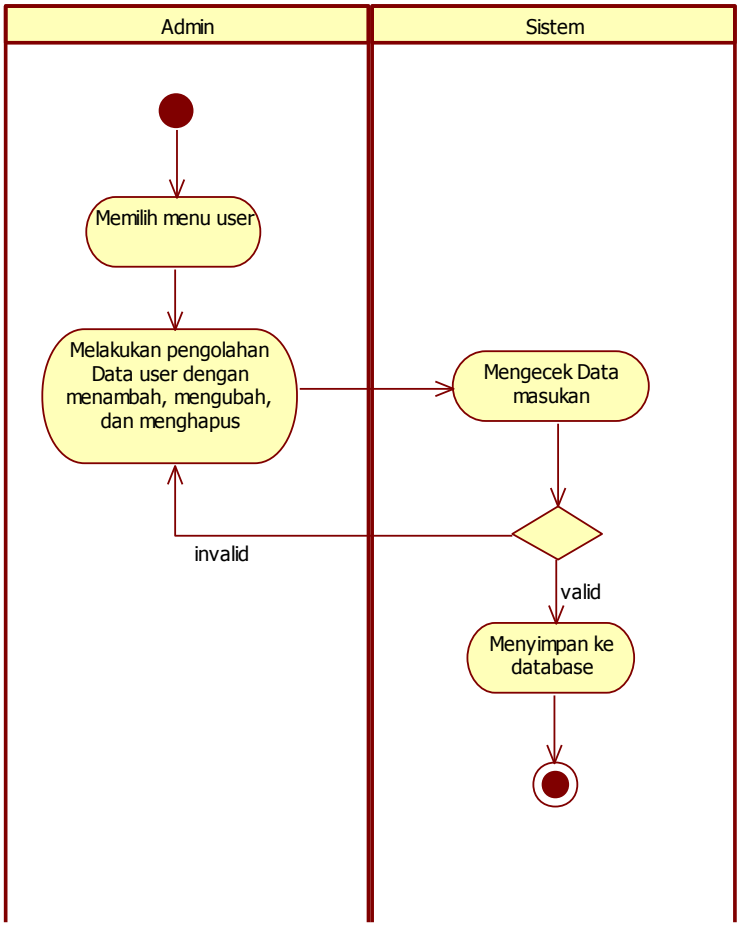



**Gambar 3.27 *Activity* Diagram *Backup***

**23. *Activity* Diagram Mengelola User**

Deskripsi :

*Admin* masuk ke halaman *user*. *Admin* dapat melakukan penambahan data *user*, pengubahan data yang diinginkan, dan menghapus data.

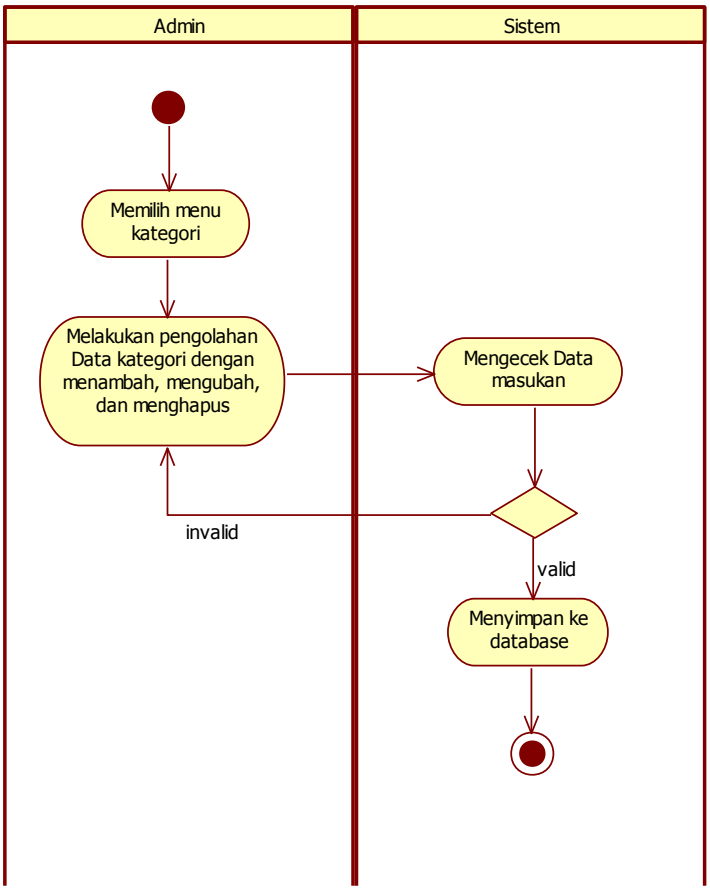



**Gambar 3.28 *Activity* Diagram Mengelola User**

### 24. *Activity* Diagram Mengelola Kategori

Deskripsi :

*Admin* masuk ke halaman kategori. *Admin* dapat melakukan penambahan data kategori, pengubahan data yang diinginkan, dan menghapus data.

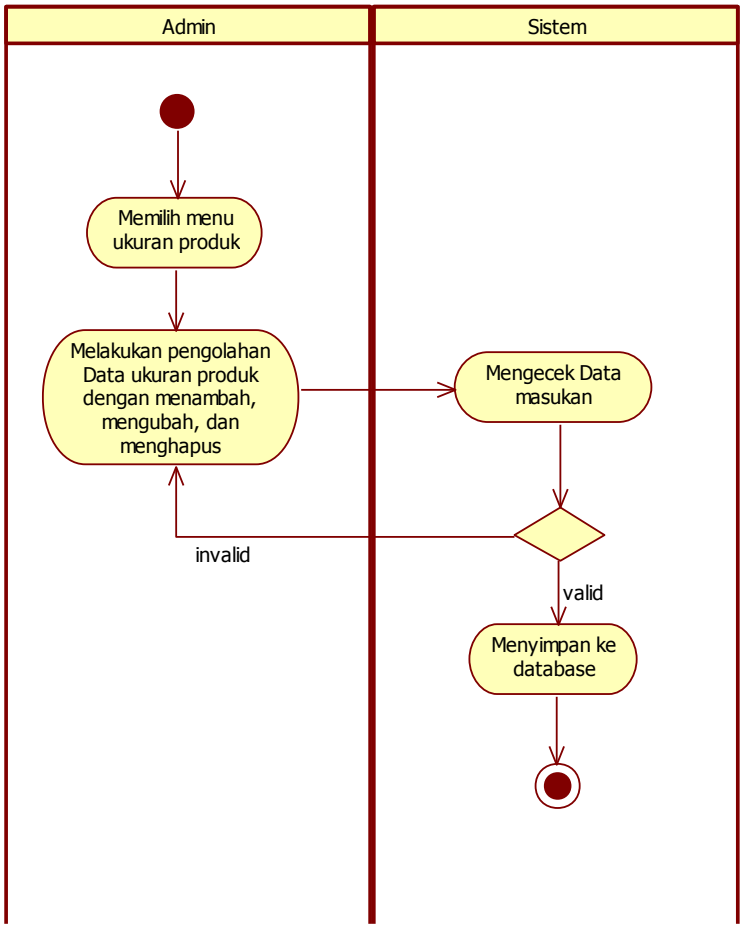



**Gambar 3.29 Activity Diagram Mengelola Kategori**

**25. *Activity* Diagram Mengelola Ukuran Produk**

Deskripsi :

*Admin* masuk ke halaman ukuran. *Admin* dapat melakukan penambahan data ukuran produk, pengubahan data yang diinginkan, dan menghapus data.

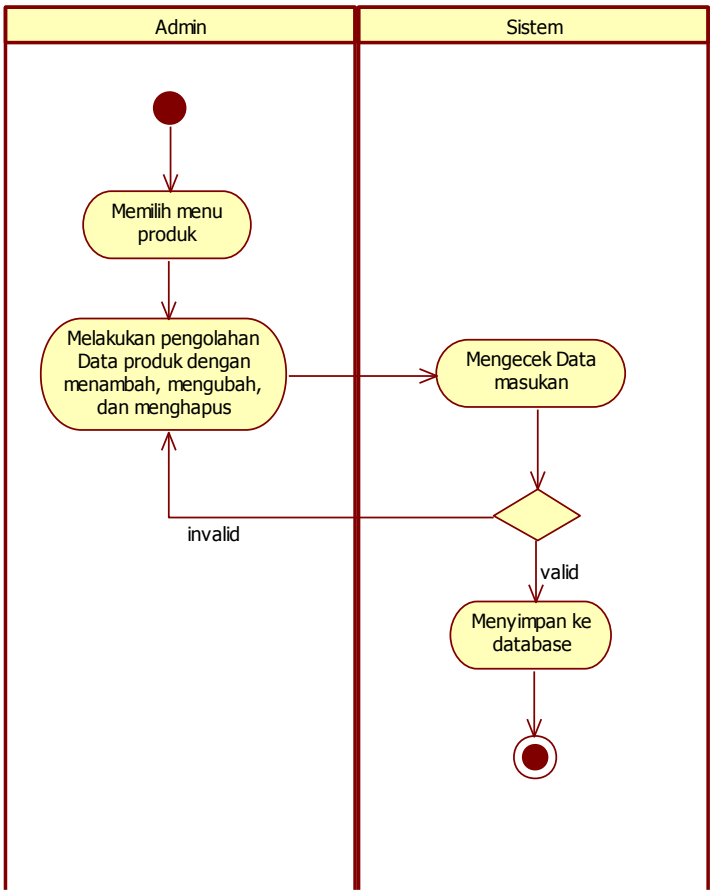



**Gambar 3.30 *Activity* Diagram Mengelola Ukuran Produk**

**26. *Activity* Diagram Mengelola Produk**

Deskripsi :

*Admin* masuk ke halaman produk, *admin* dapat menbamah, mengubah dan menghapus produk. Sistem akan membaca perintah dari petugas dan memprosesnya jika perintah tersebut benar dan sesuai maka sistem akan menyimpan perintah tersebut.

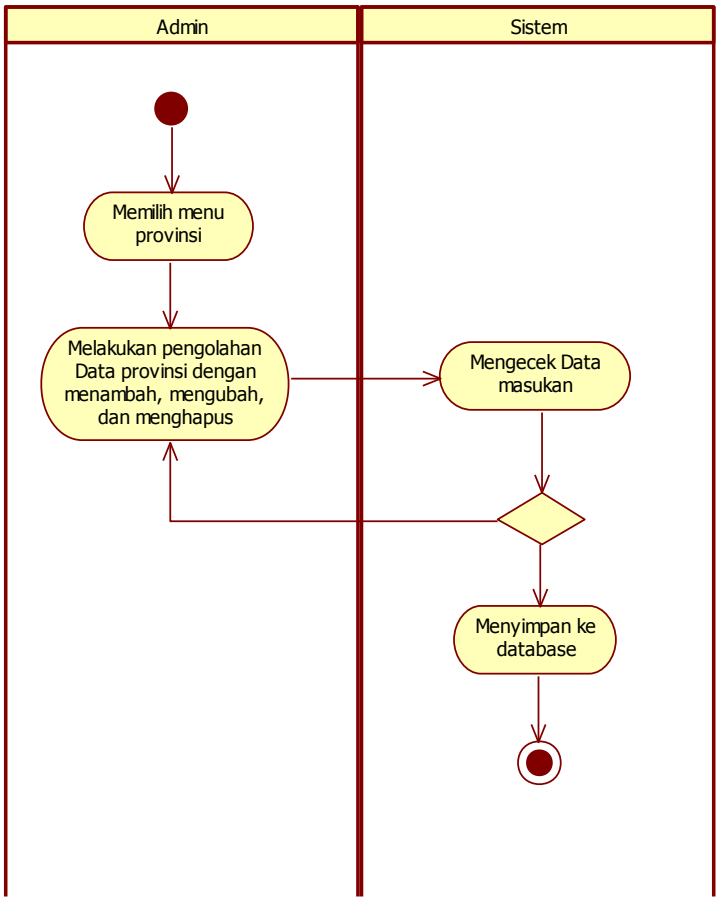



**Gambar 3.31 Activity Diagram Mengelola Produk**

**27. *Activity* Diagram Mengelola Provinsi**

Deskripsi :

*Admin* masuk ke halaman provinsi. *Admin* dapat melakukan penambahan data provinsi, pengubahan data yang diinginkan, dan menghapus data.

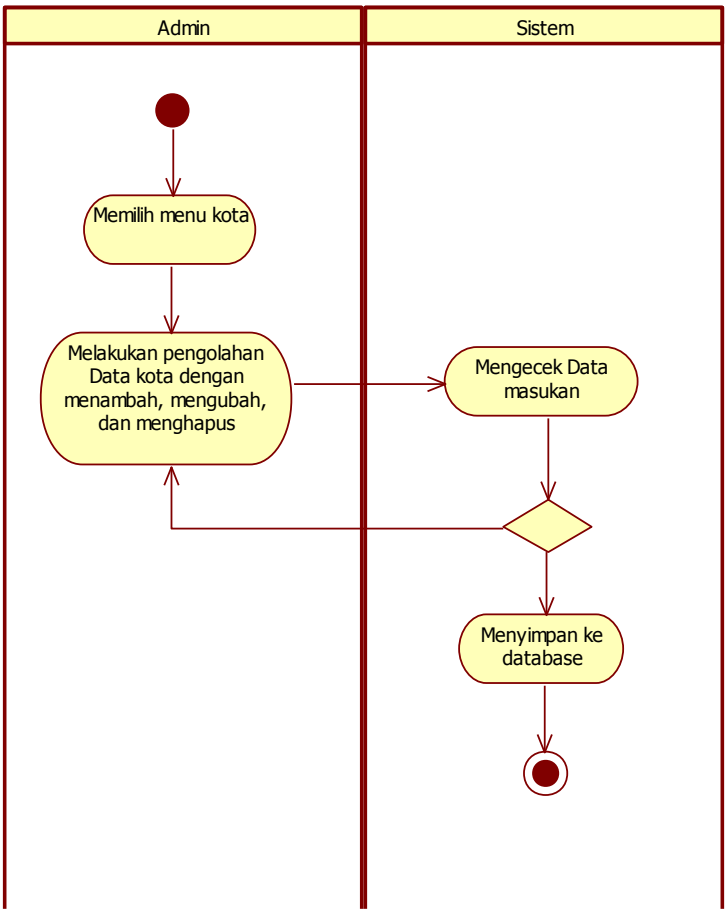



**Gambar 3.32 *Activity* Diagram mengelola provinsi**

### 28. *Activity* Diagram Mengelola Kota

Deskripsi :

*Admin* masuk ke halaman kota dan sistem akan menampilkan daftar kota. *Admin* dapat menambah, mengubah, dan menghapus data kota.

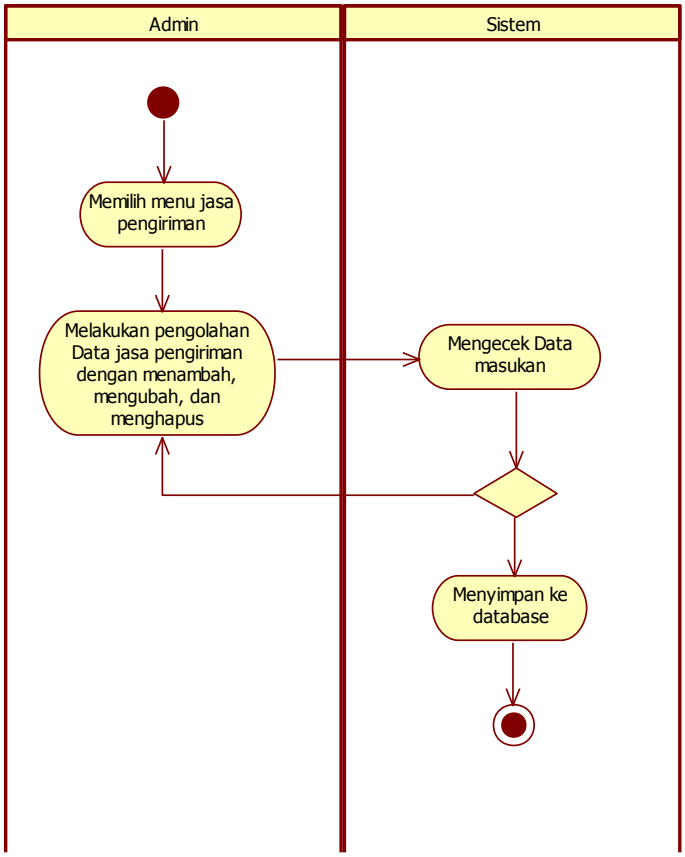



**Gambar 3.33 *Activity* Diagram Mengelola Kota**

### 29. *Activity* Diagram Mengelola Jasa Pengiriman

Deskripsi:

*Admin* masuk ke menu jasa pengiriman dan sistem akan menampilkan data jasa pengiriman. *Admin* dapat menambah, mengubah dan menghapus data jasa pengiriman.

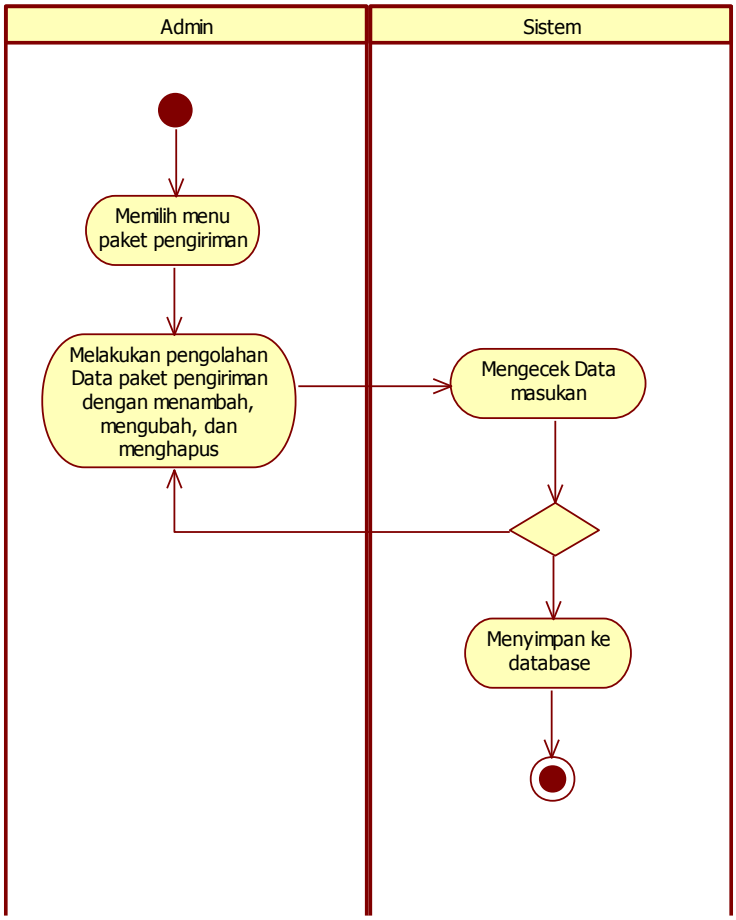



**Gambar 3.34 *Activity* Diagram Jasa Pengiriman**

### 30. *Activity* Diagram Mengelola Paket Pengiriman

Deskripsi:

*Admin* masuk ke paket pengiriman dan sistem akan menampilkan data paket pengiriman. *Admin* dapat menambah, mengubah dan menghapus data paket pengiriman.

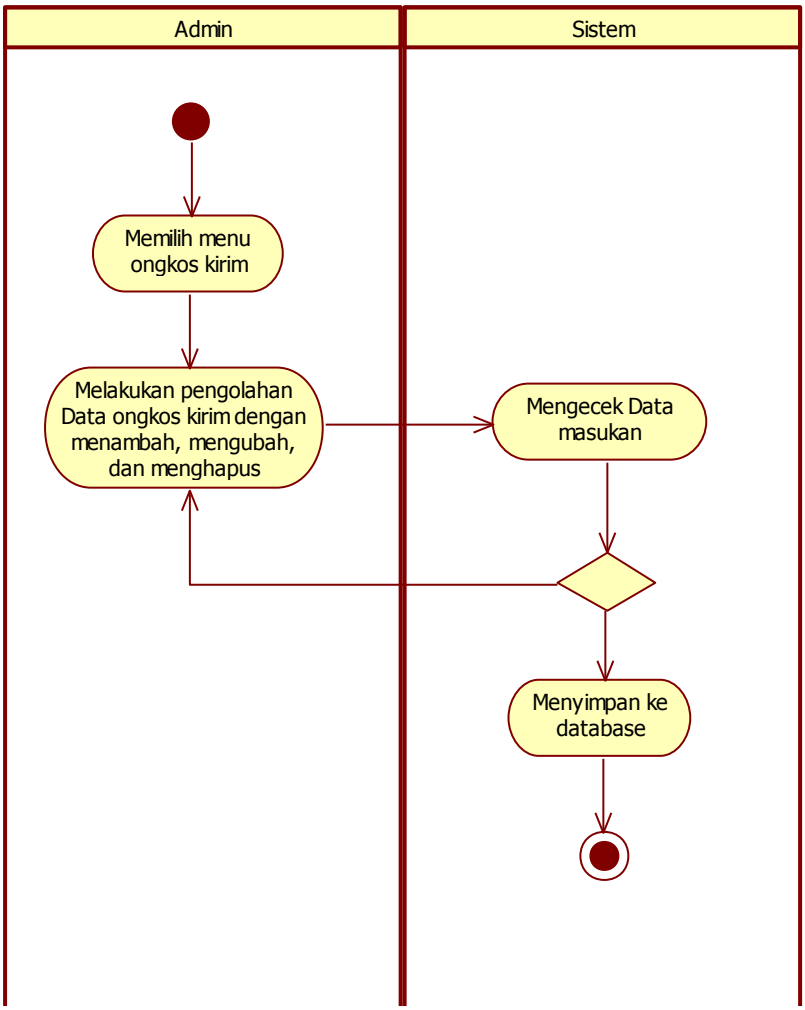



**Gambar 3.35 *Activity* Diagram Paket Pengiriman**

### 31. *Activity* Diagram Mengelola Ongkos Kirim

Deskripsi :

*Admin* masuk ke menu ongkos kirm dan sistem akan menampilkan data ongkos kirim. *Admin* dapat menambah, mengubah dan menghapus data ongkos kirim.

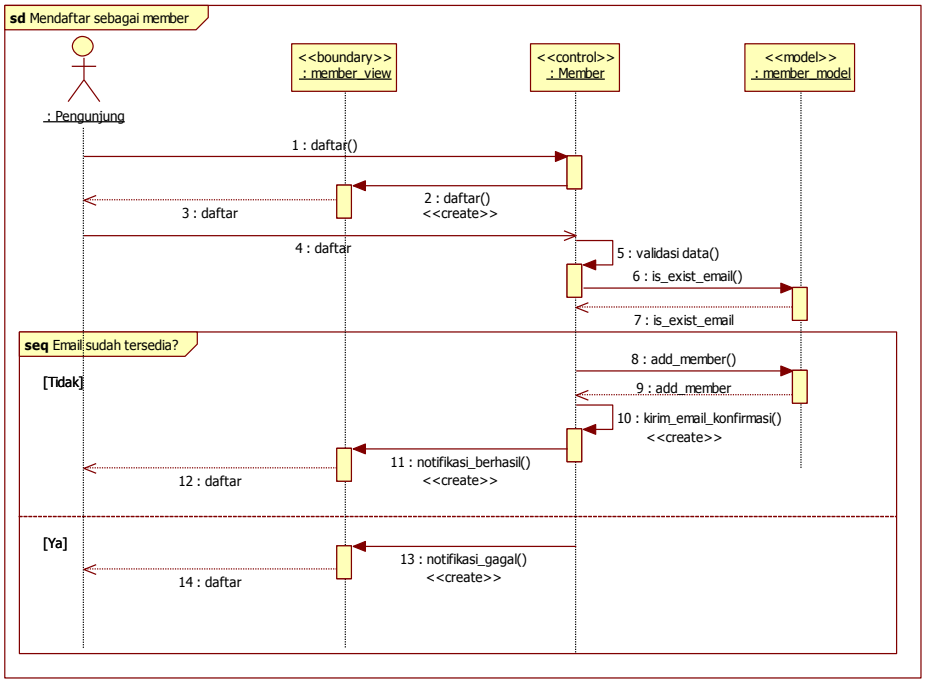



**Gambar 3.36 *Activity* Diagram Mengelola ongkos kirim**

### 3.1.7.6. *Sequence* Diagram

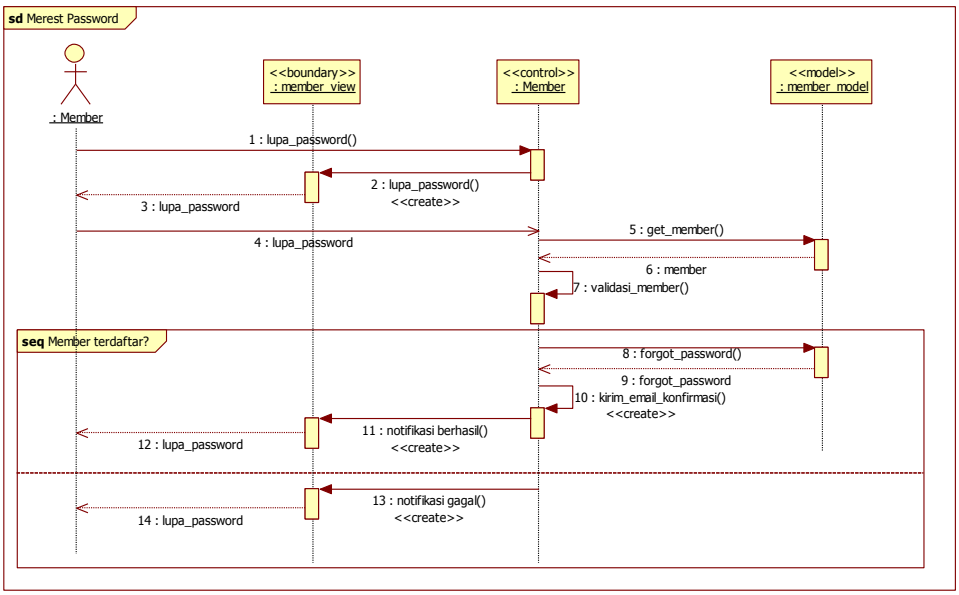
*Sequence* diagram (diagram sekuensial) adalah diagram yang digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Berikut ini adalah sequence diagram dari *use case* diagram aplikasi e-commerce pada toko RusticShop:

## 1. Mendaftar sebagai member



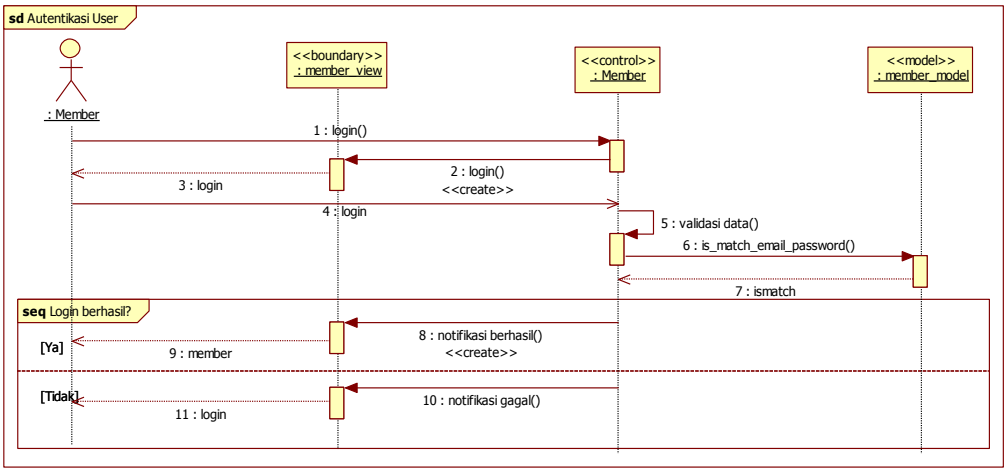

**Gambar 3.37 *Sequence* Diagram Mendaftar sebagai member**

## 2. Mereset *Password*



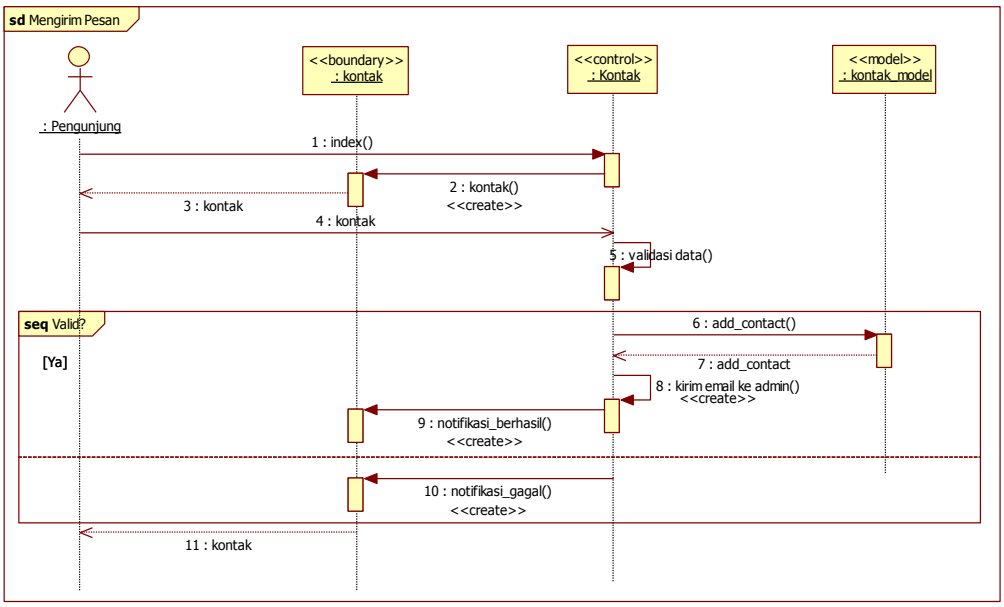

**Gambar 3.38 *Sequence* Diagram Mereset *Password***

### 3. Autentikasi *User*



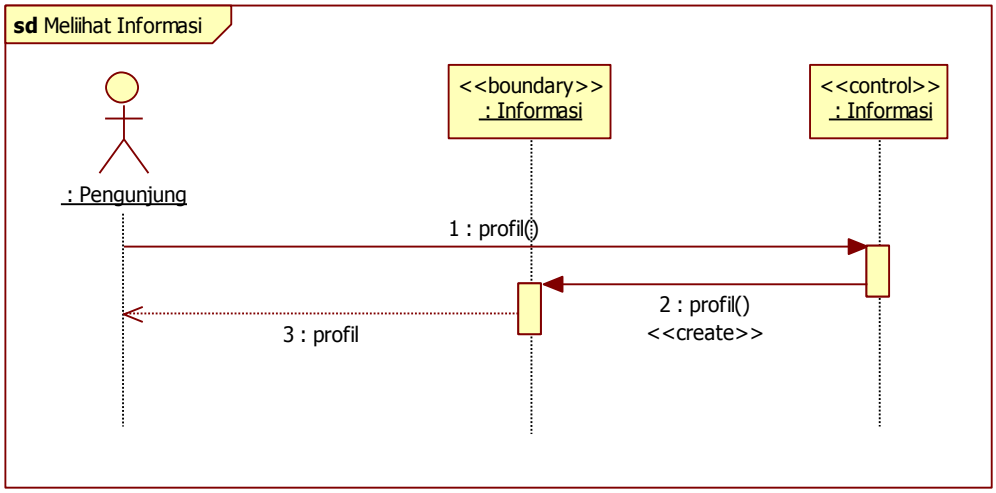

**Gambar 3.39 *Sequence* Diagram Autentikasi *User***

### 4. Mengirim Pesan



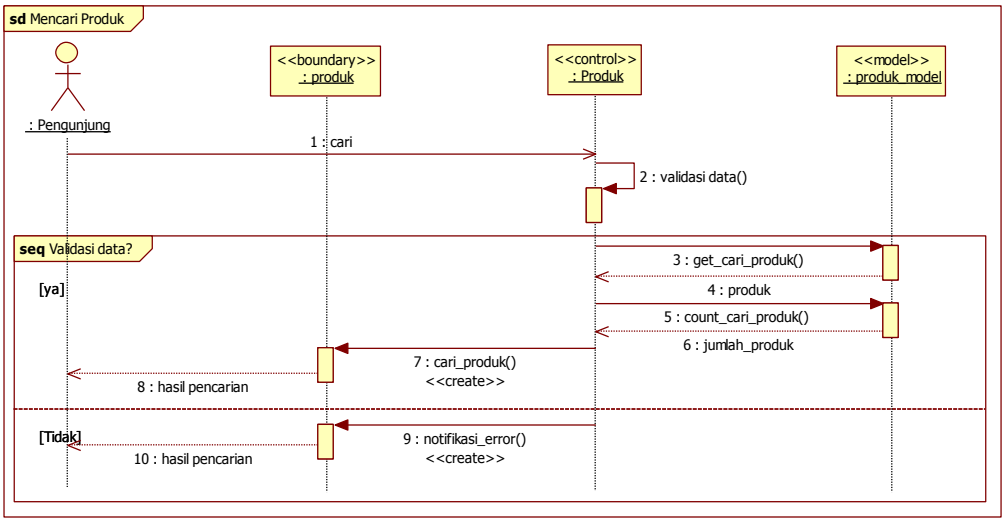

**Gambar 3.40 *Sequence* Diagram Mengirim Pesan**

## 5. Melihat Informasi



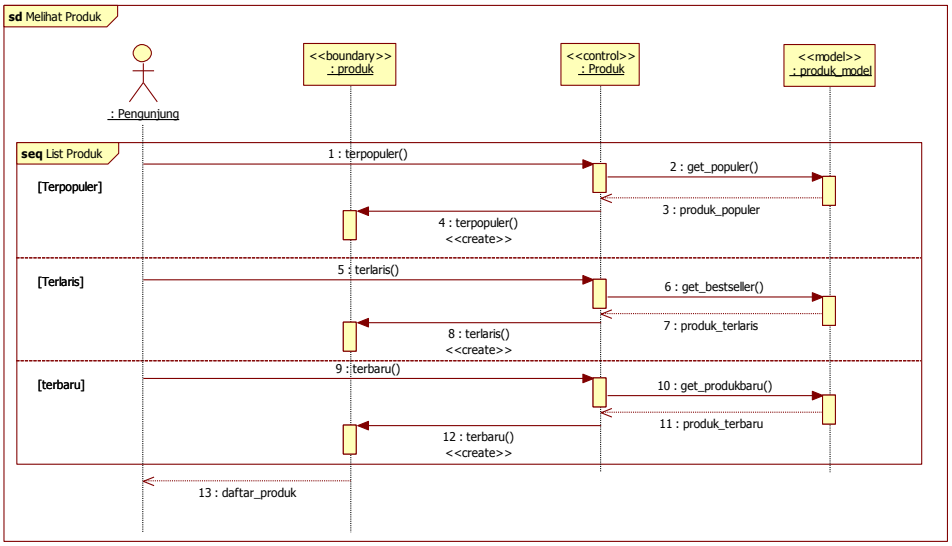

**Gambar 3.41 *Sequence* Diagram Melihat Informasi**

## 6. Mencari Produk



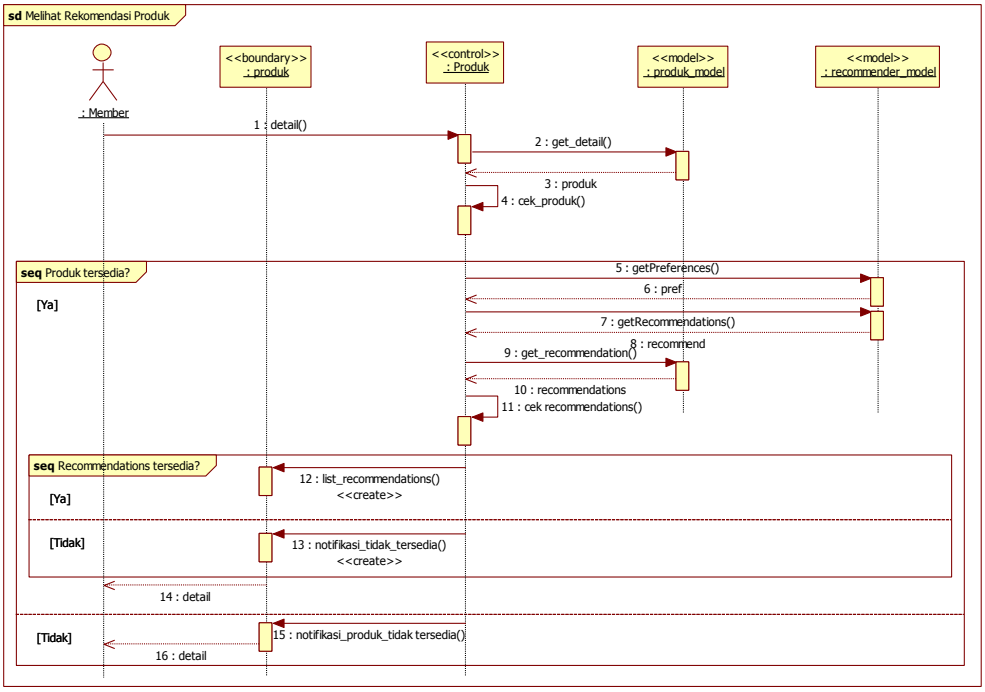

**Gambar 3.42 *Sequence* Diagram Mencari Produk**

## 7. Melihat Produk



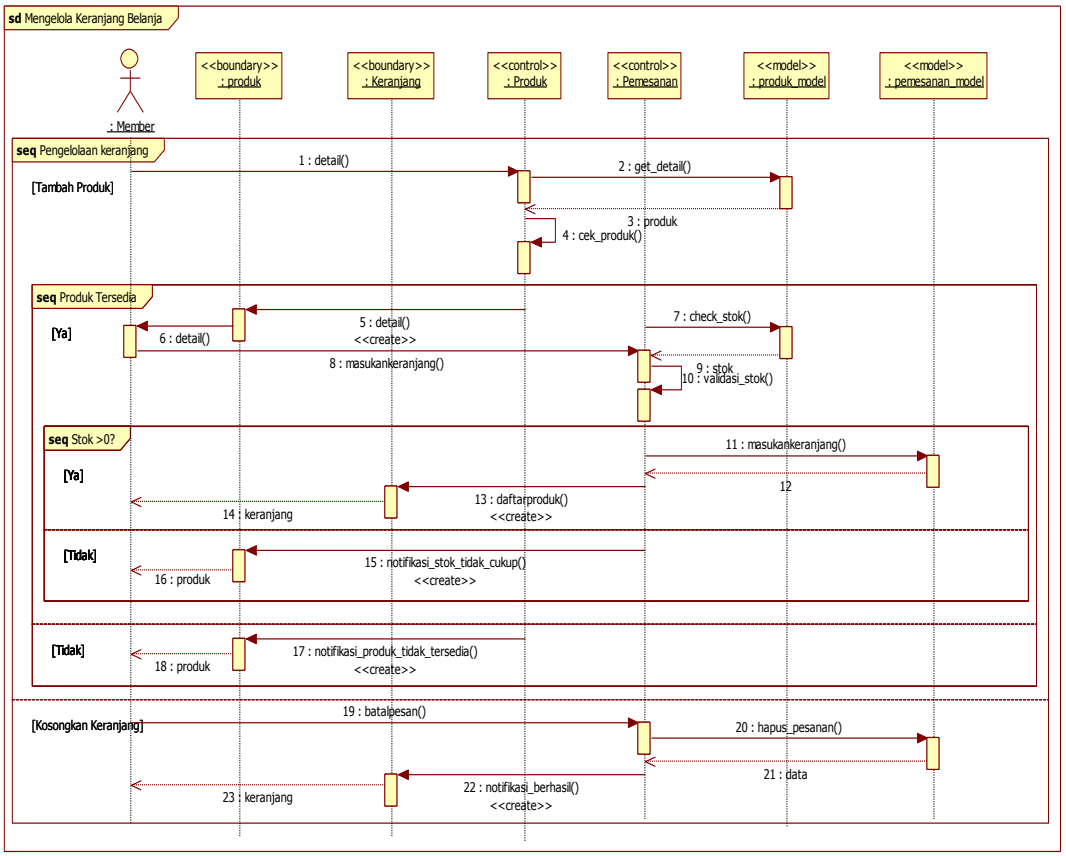

**Gambar 3.43 *Sequence* Diagram Lihat Produk**

## 8. Melihat Rekomendasi Produk



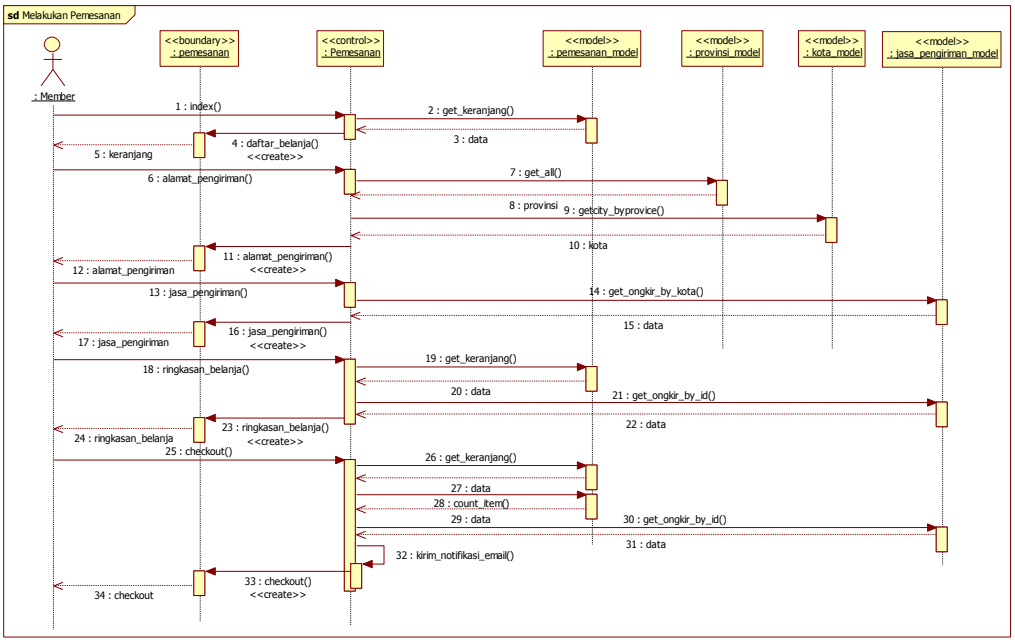

**Gambar 3.44 *Sequence* Diagram Melihat Rekomendasi Produk**

## 9. Mengelola Keranjang Belanja



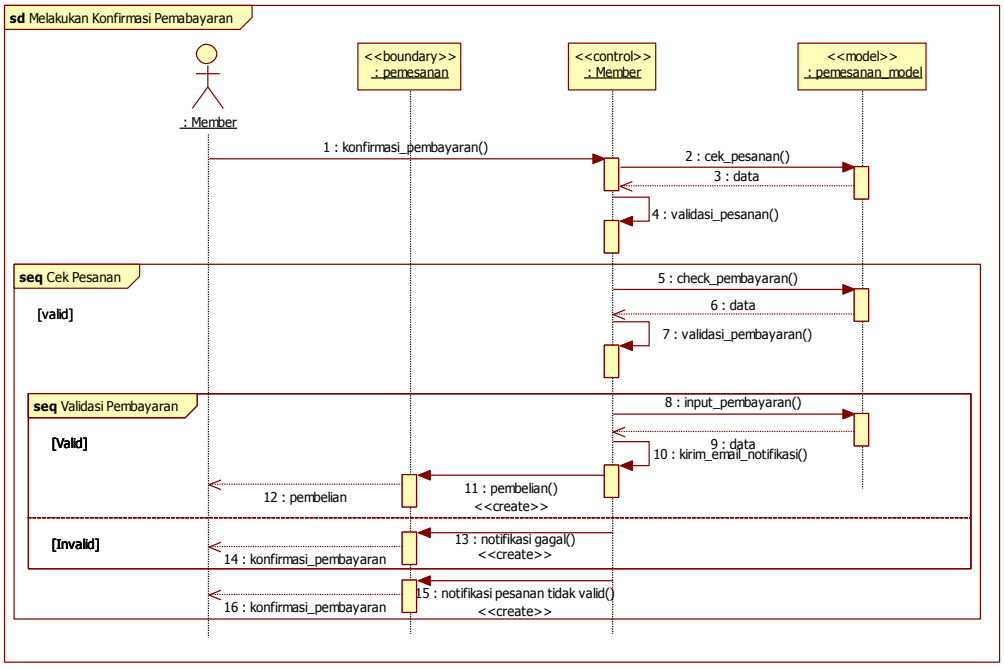

**Gambar 3.45 *Sequence* Diagram Mengelola Keranjang Belanja**

## 10. Melakukan Pesanan



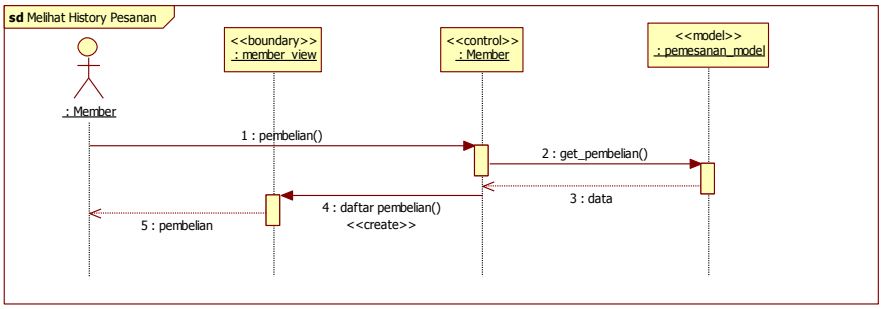

**Gambar 3.46 *Sequence* Diagram Melakukan Pesanan**

## 11. Mengkonfirmasi Pembayaran



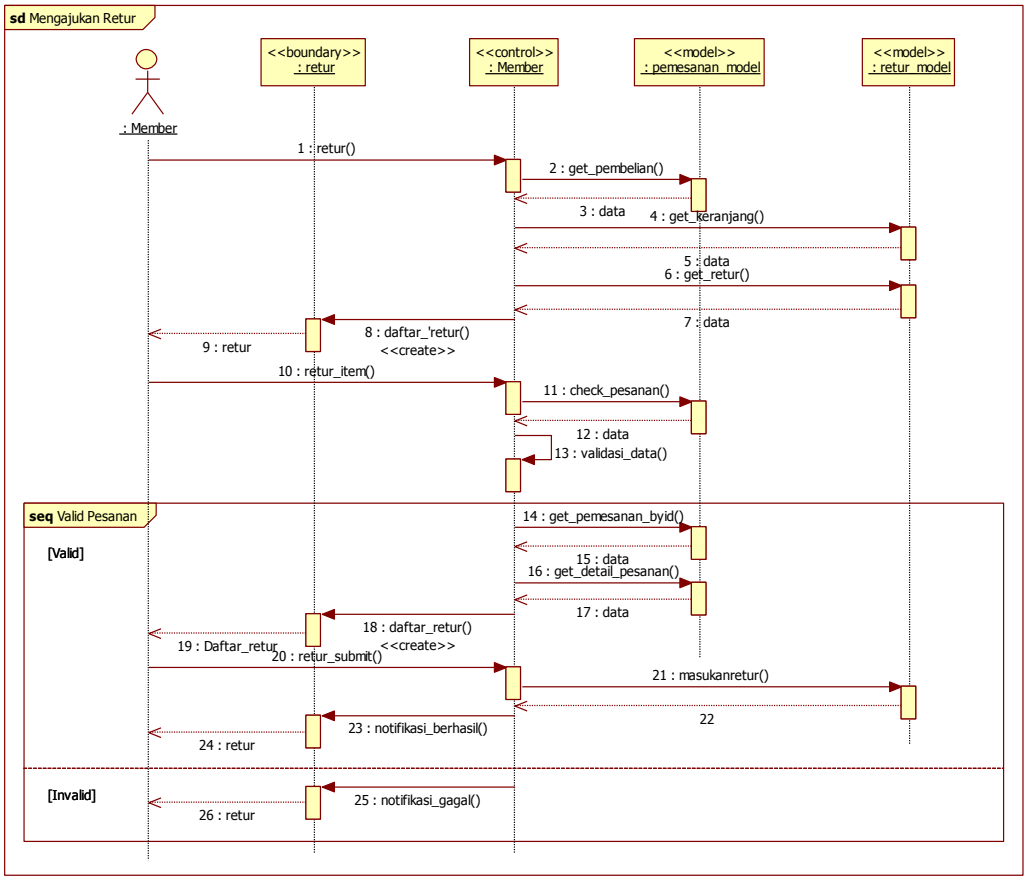

**Gambar 3.47 *Sequence* Diagram Mengkonfirmasi Pembayaran**

## 12. Melihat *History* Pesanan



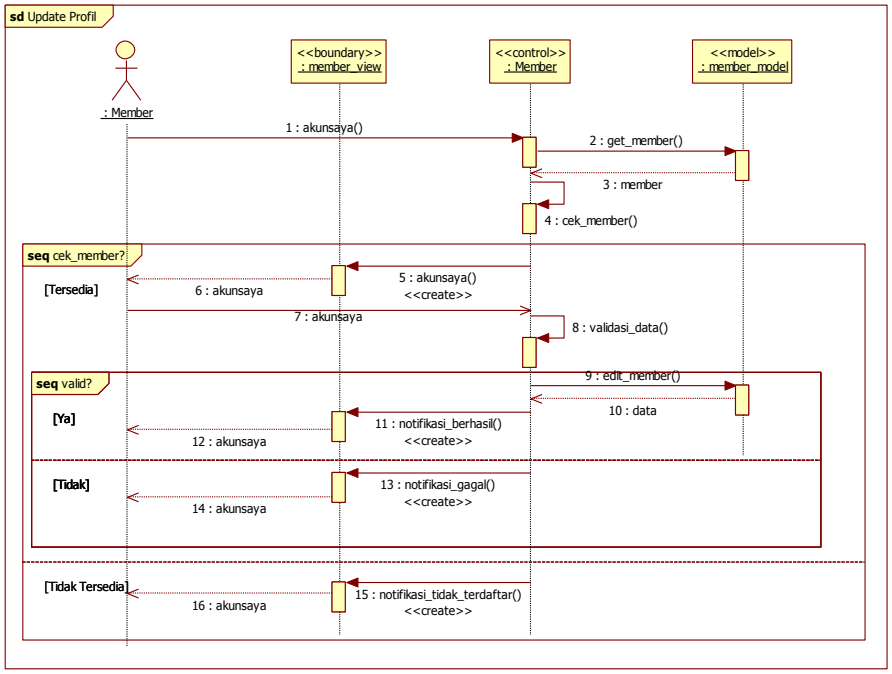

**Gambar 3.48 *Sequence* Diagram Melihat History Pesanan**

## 13. Mengajukan Retur



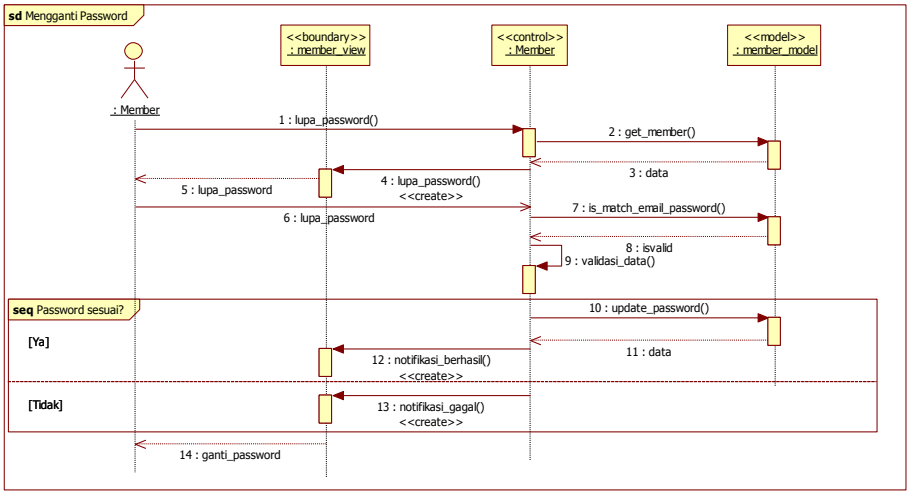

**Gambar 3.49 *Sequence* Diagram Mengajukan Retur**

## 14. Update Profil



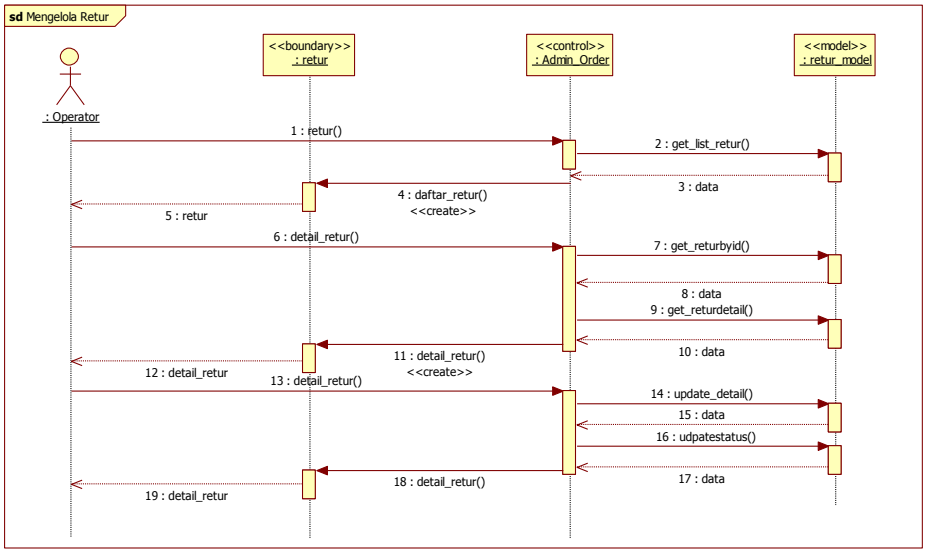

**Gambar 3.50 *Sequence* Diagram Update Profil**

## 15. Mengganti *Password*



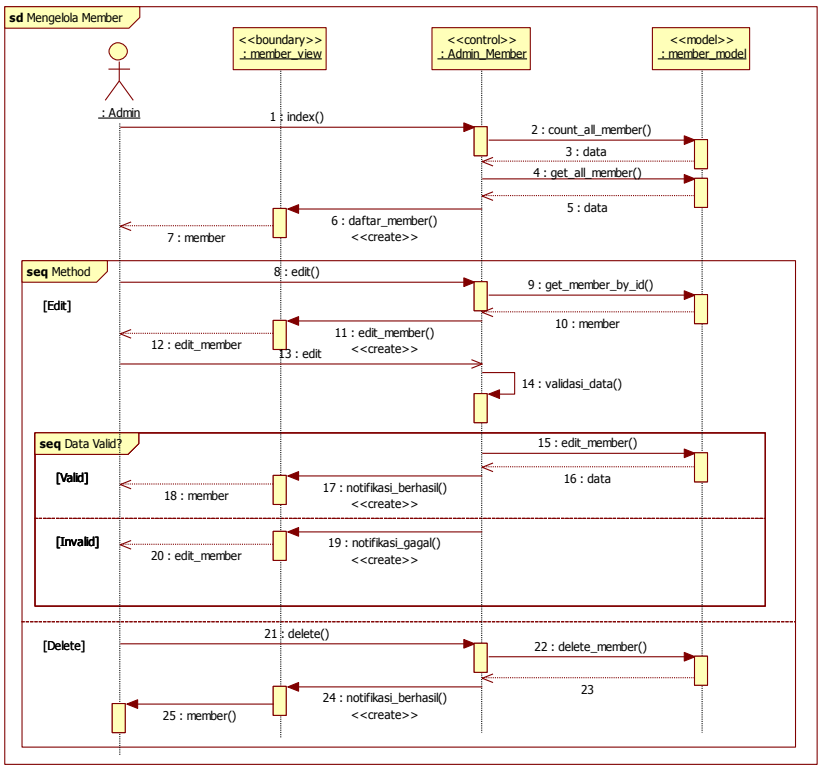

**Gambar 3.51 *Sequence* Diagram Ganti *Password***

## 16. Mengelola Retur



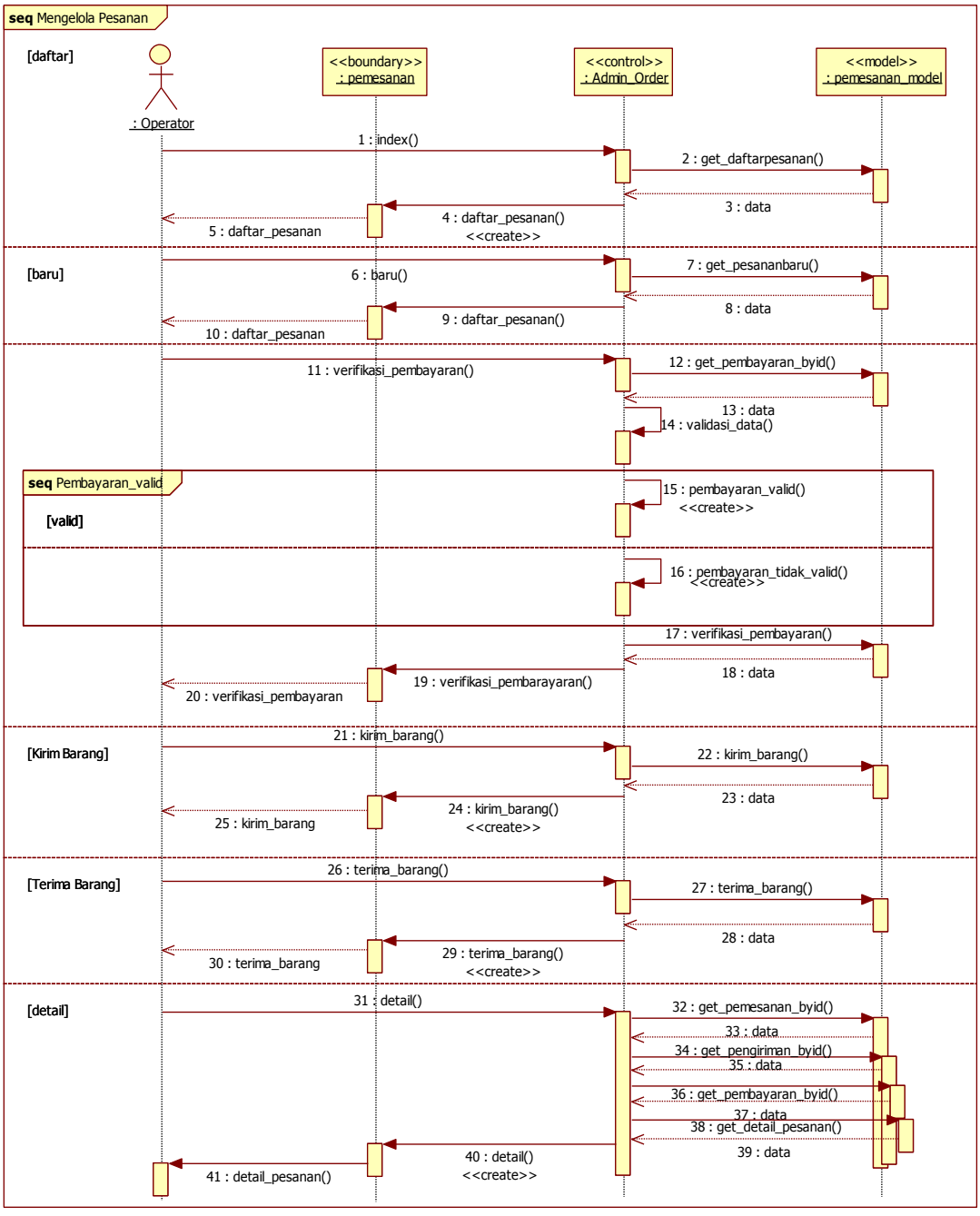

**Gambar 3.52 *Sequence* Diagram Mengelola Retur**

## 17. Mengelola Member



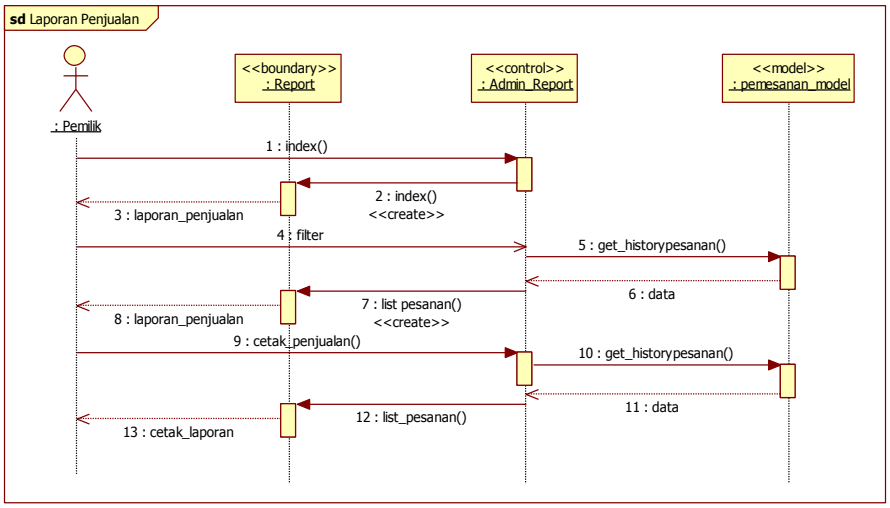

**Gambar 3.53 *Sequence* Diagram Mengelola Member**

## 18. Mengelola Pesanan



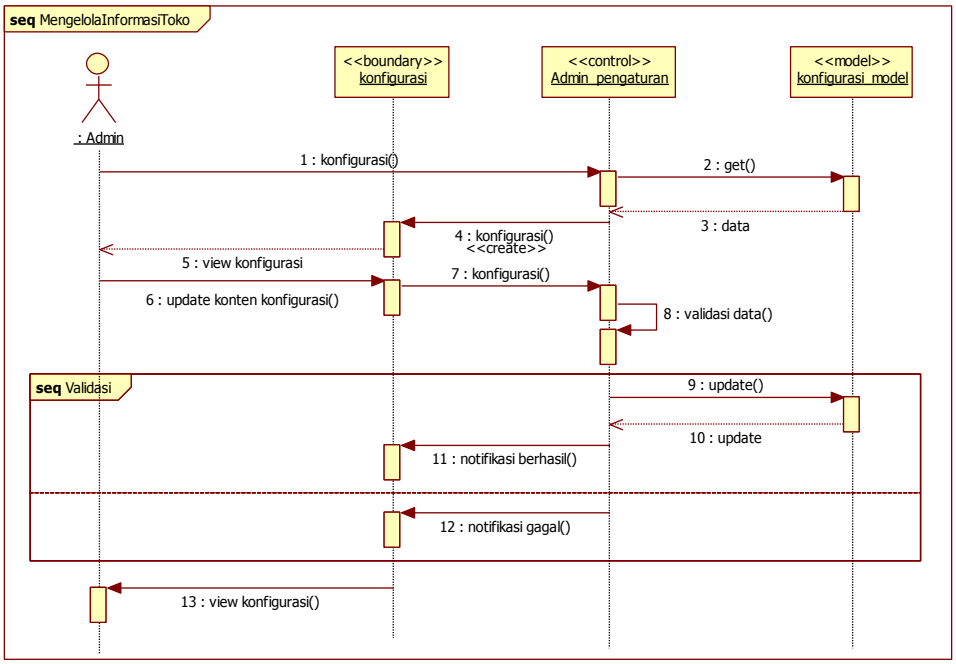

**Gambar 3.54 *Sequence* Diagram Mengelola Pesanan**

## 19. Laporan Penjualan



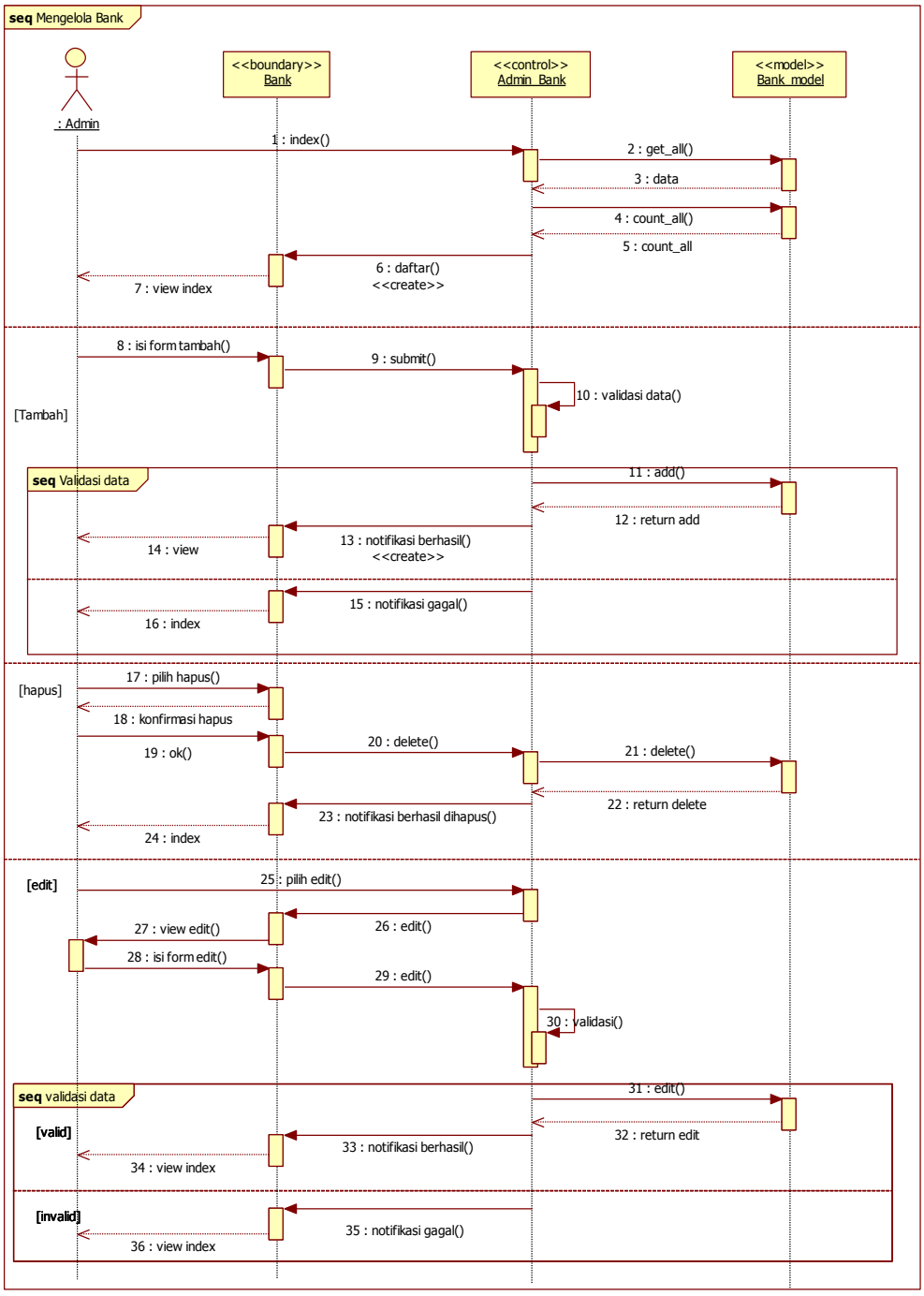

**Gambar 3.55 *Sequence* Diagram Laporan Penjualan**

## 20. Mengelola Informasi Toko



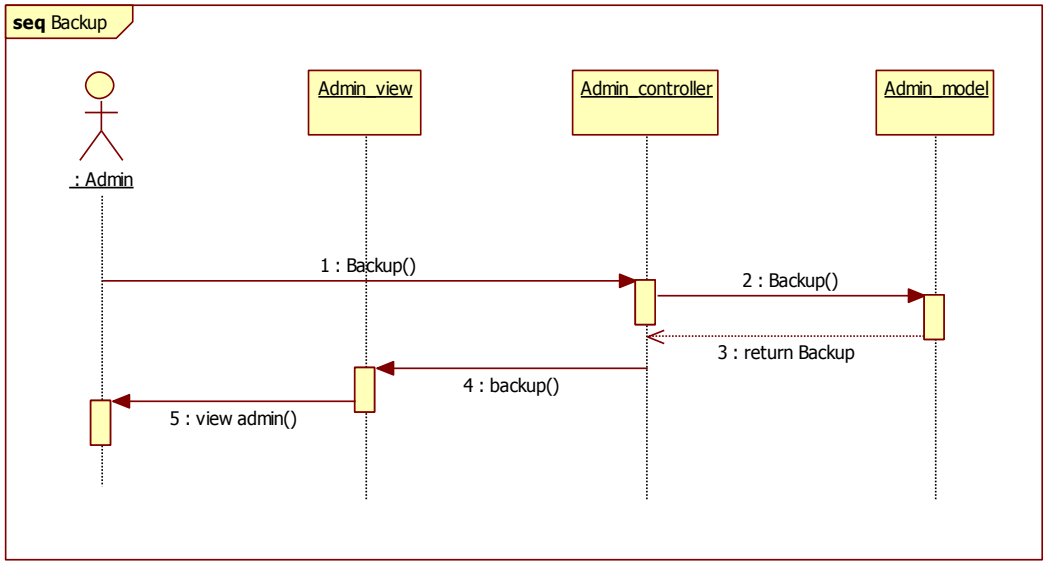

**Gambar 3.56 *Sequence* Diagram Mengelola Informasi Toko**

## 21. Mengelola Bank



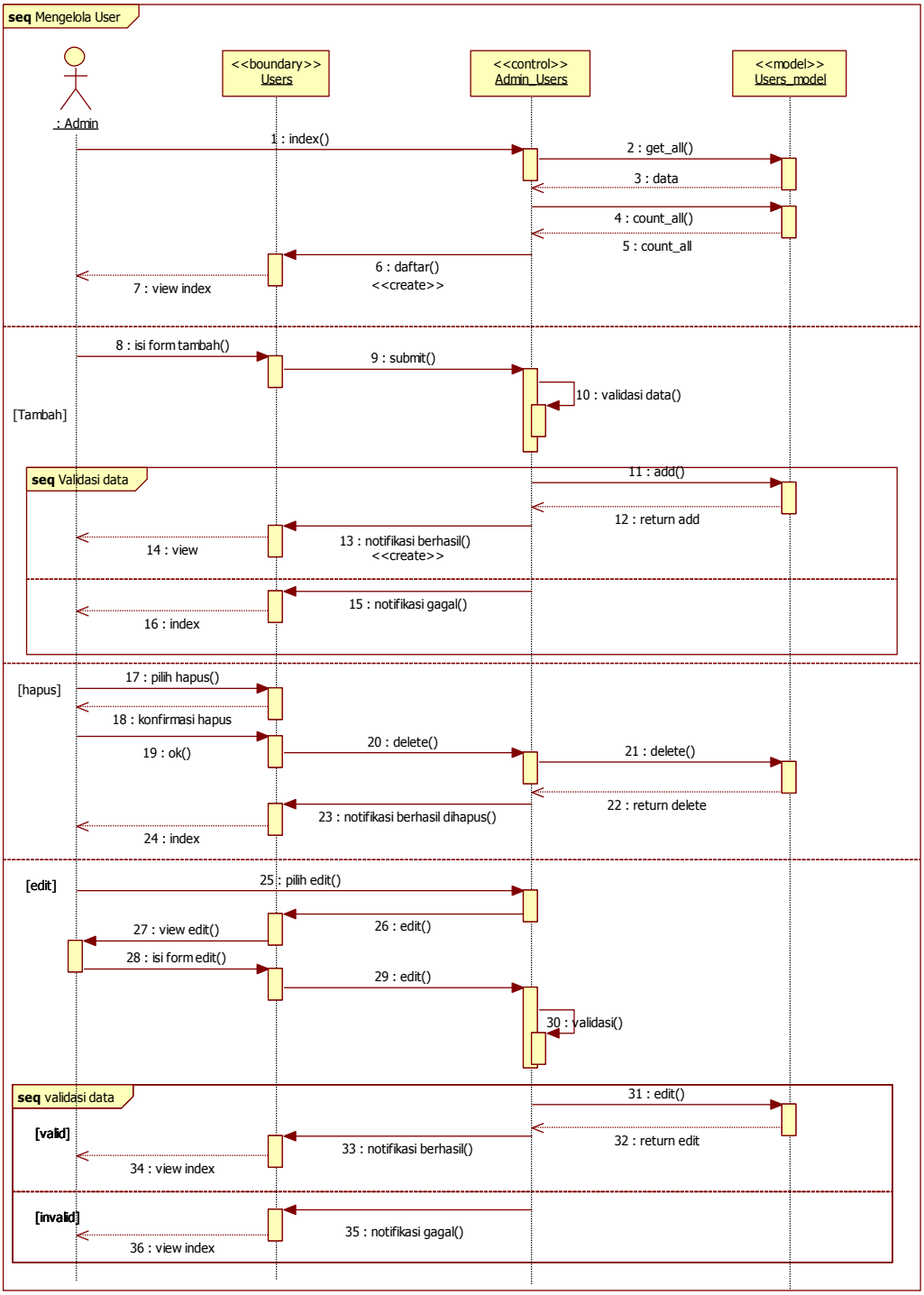

**Gambar 3.57 *Sequence* Diagram Mengelola *Bank***

## 22. *Backup*



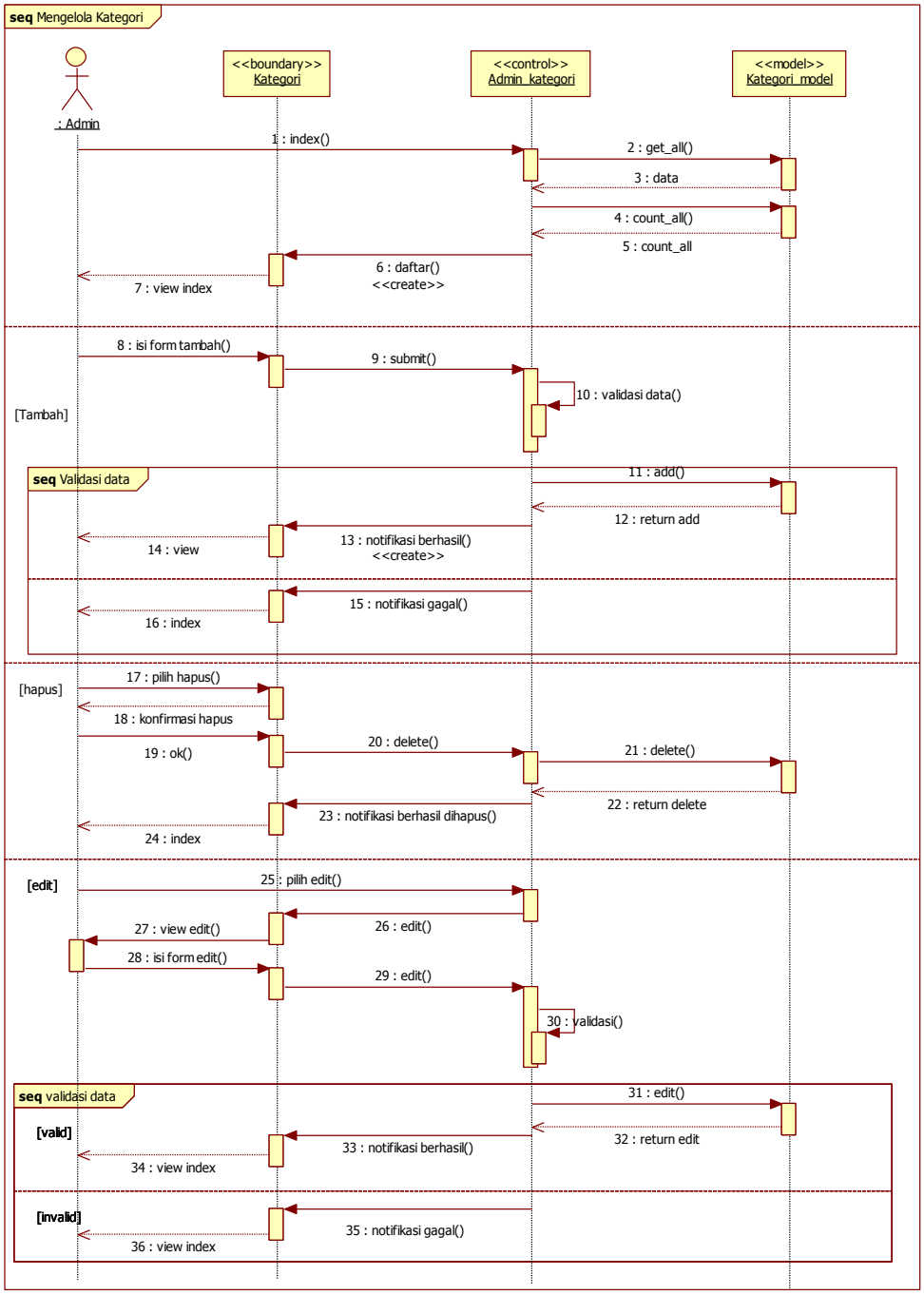

**Gambar 3.58 *Sequence* Diagram Backup**

## 23. Mengelola *User*



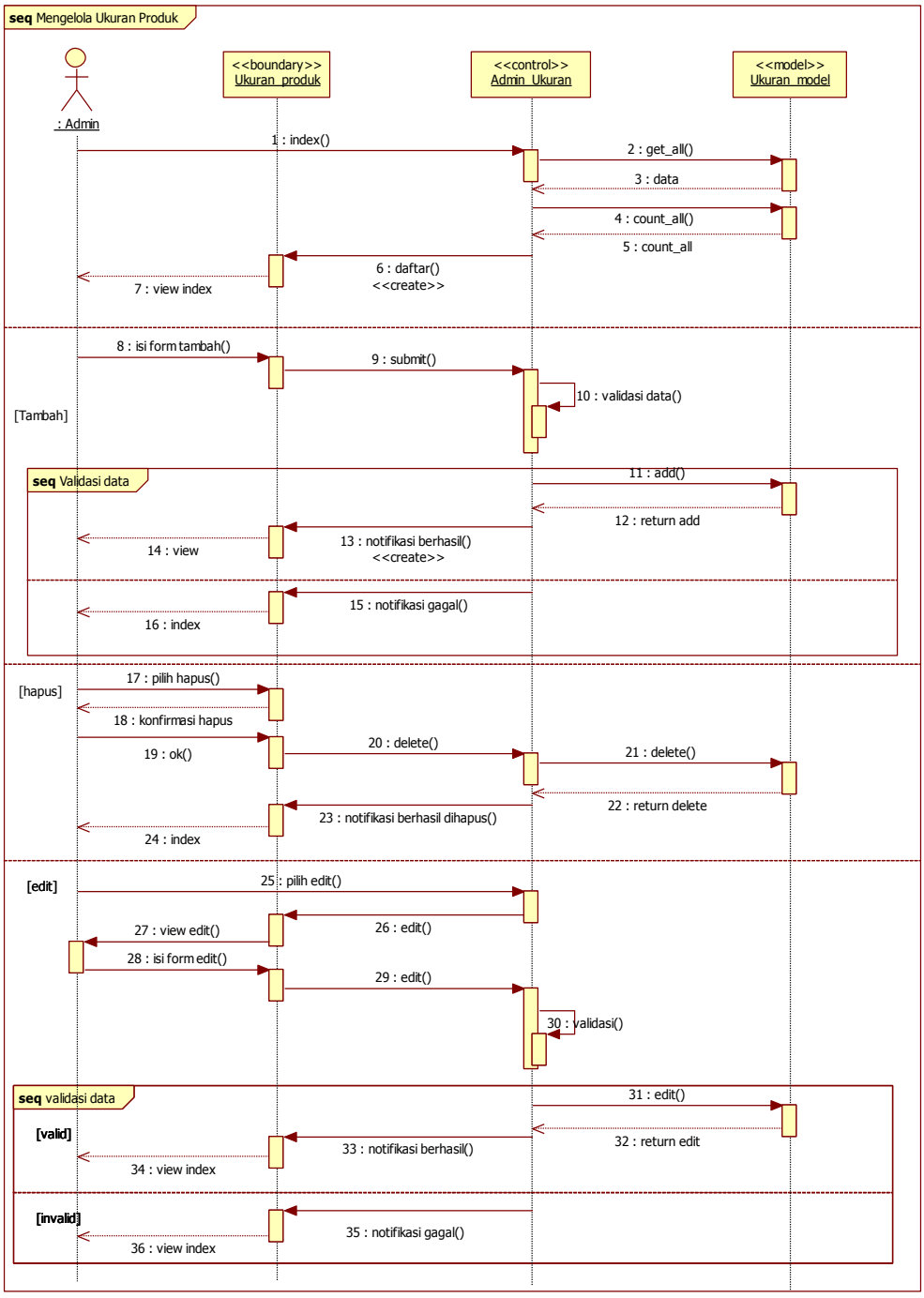

**Gambar 3.59 *Sequence* Diagram Mengelola *User***

## 24. Mengelola Kategori



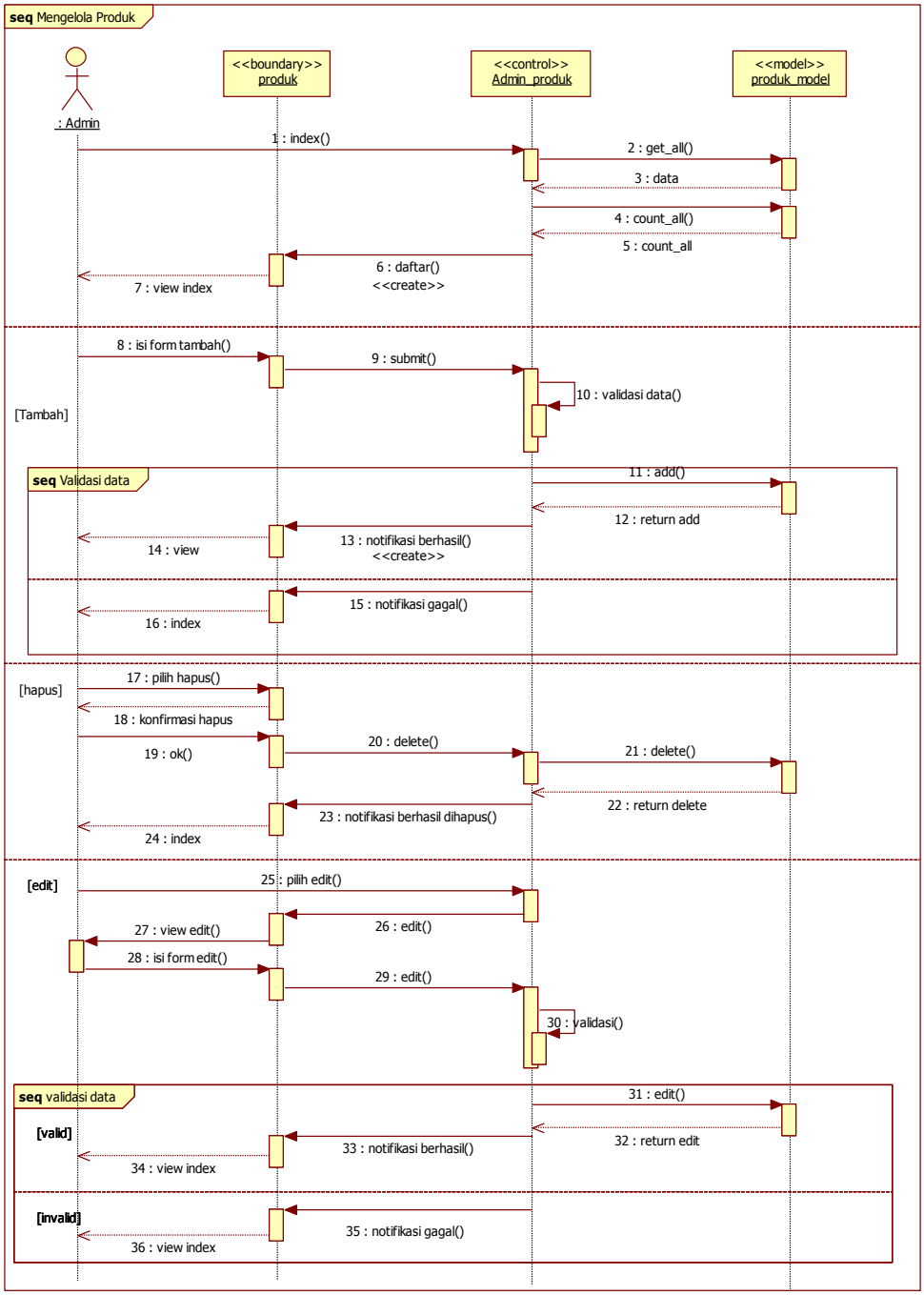

**Gambar 3.60 *Sequence* Diagram Mengelola Kategori**

## 25. Mengelola Ukuran Produk



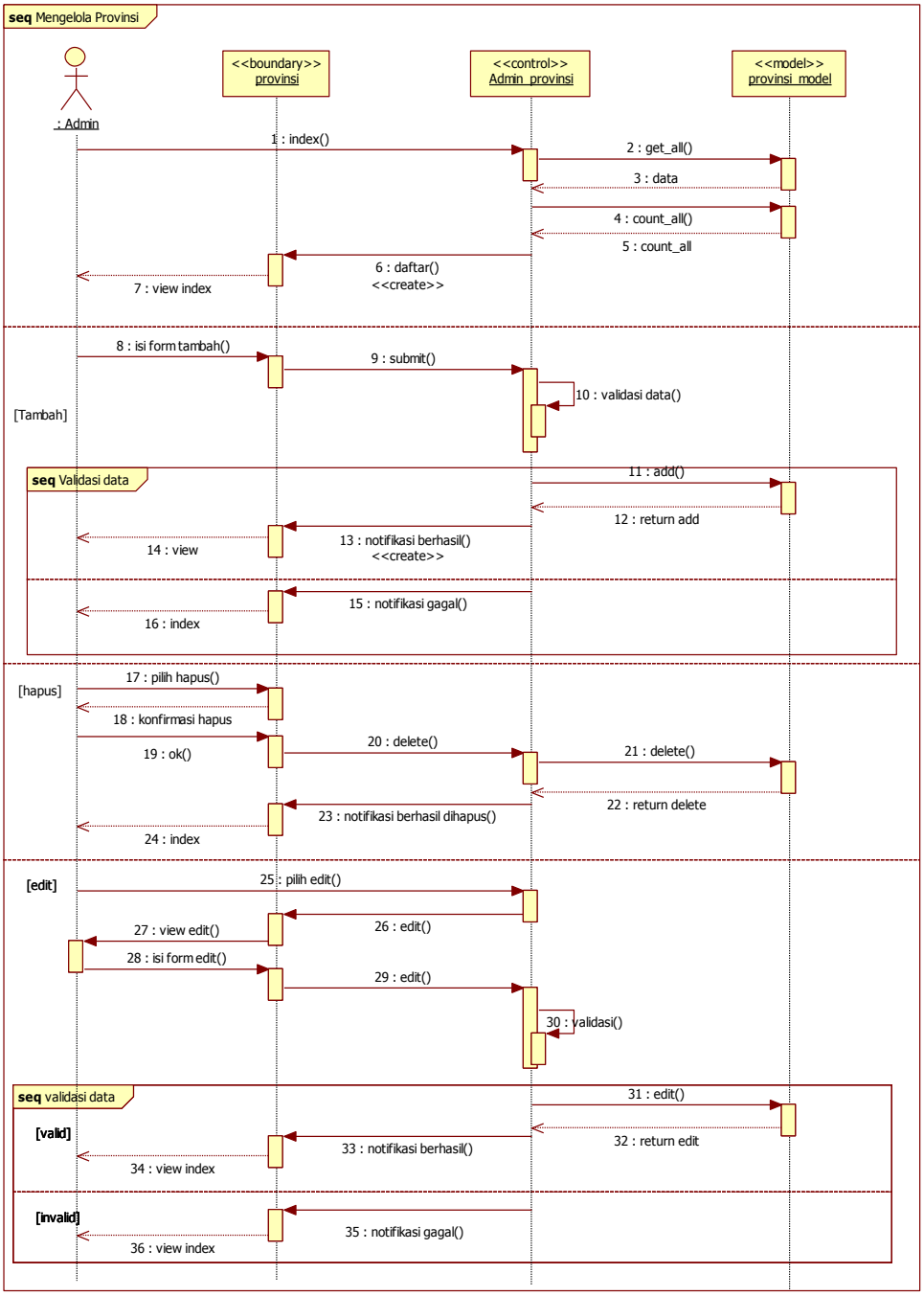

**Gambar 3.61 *Sequence* Diagram Mengelola Ukuran Produk**

## 26. Mengelola Produk



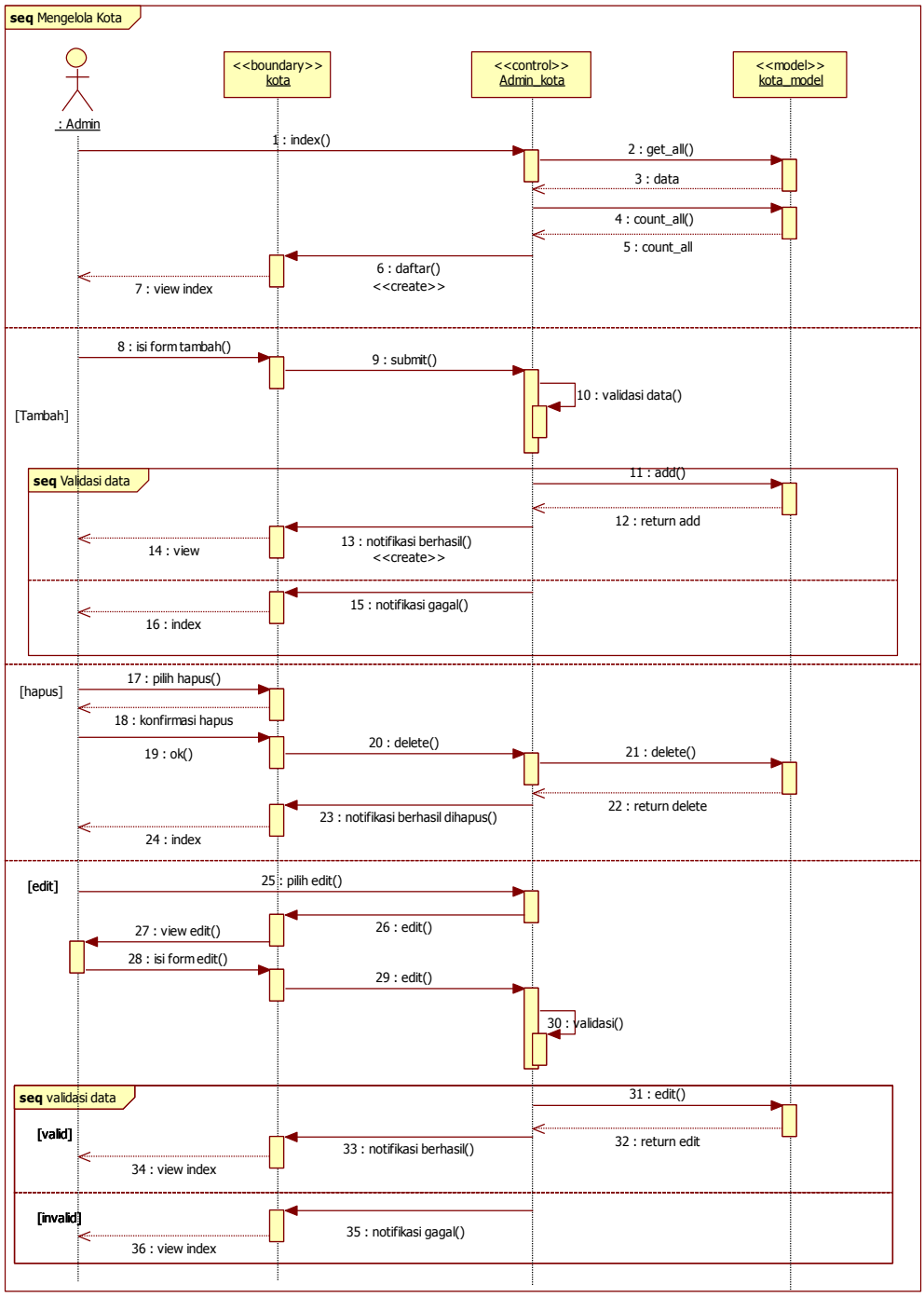

**Gambar 3.62 *Sequence* Diagram Mengelola Produk**

## 27. Mengelola Provinsi



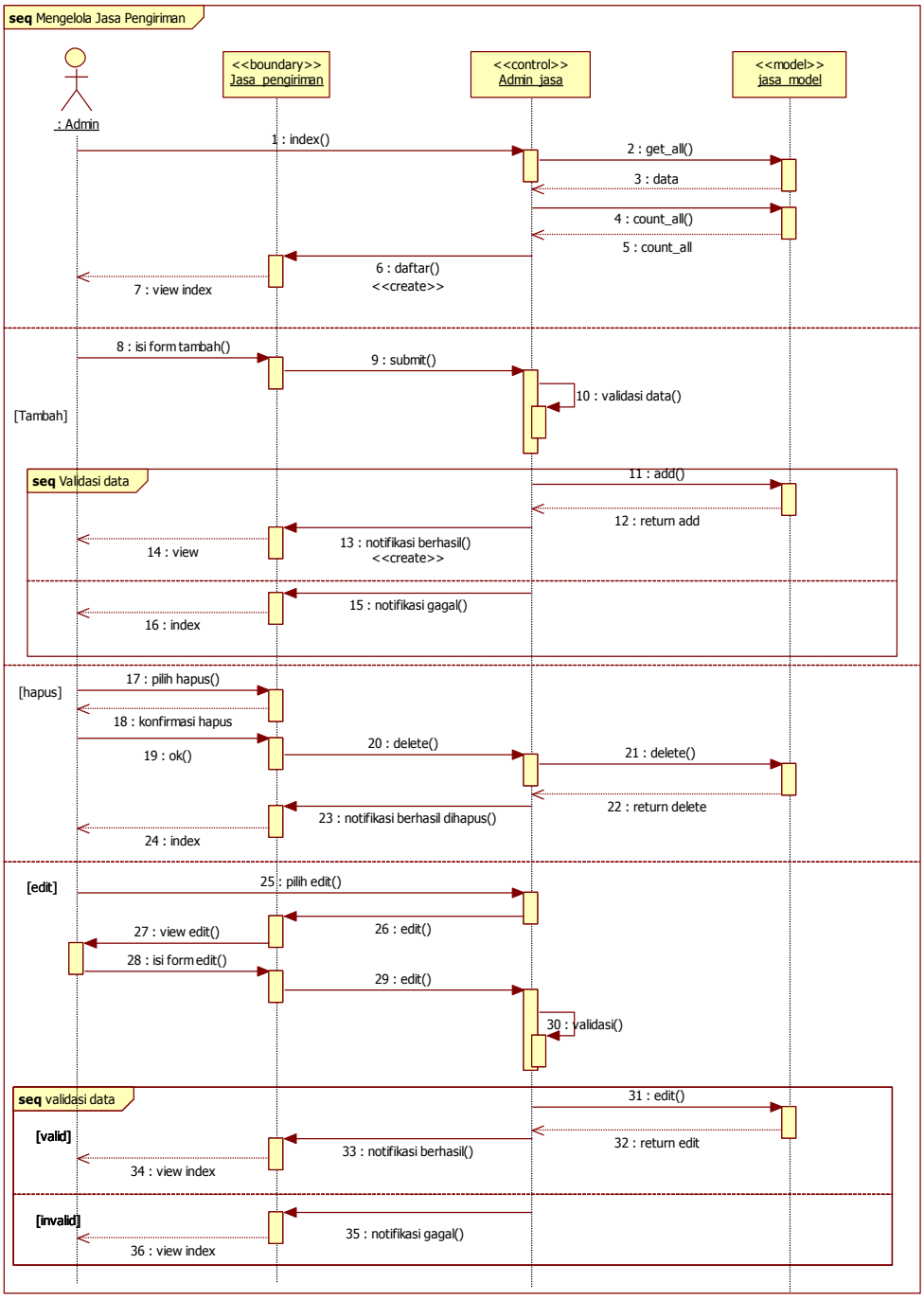

**Gambar 3.63 *Sequence* Diagram Mengelola Provinsi**

## 28. Mengelola Kota



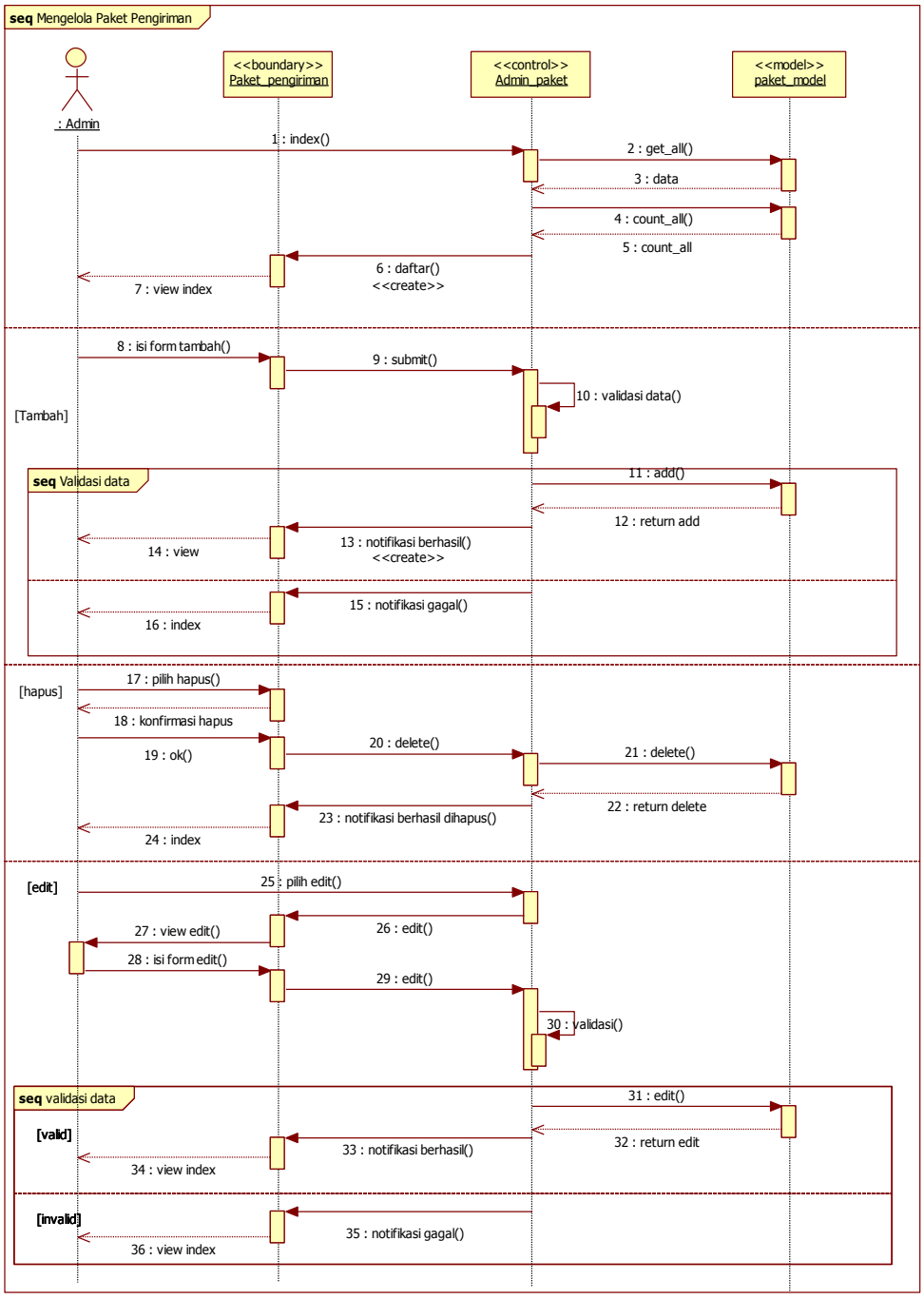

**Gambar 3.64 *Sequence* Diagram Mengelola Kota**

## 29. Mengelola Jasa Pengiriman



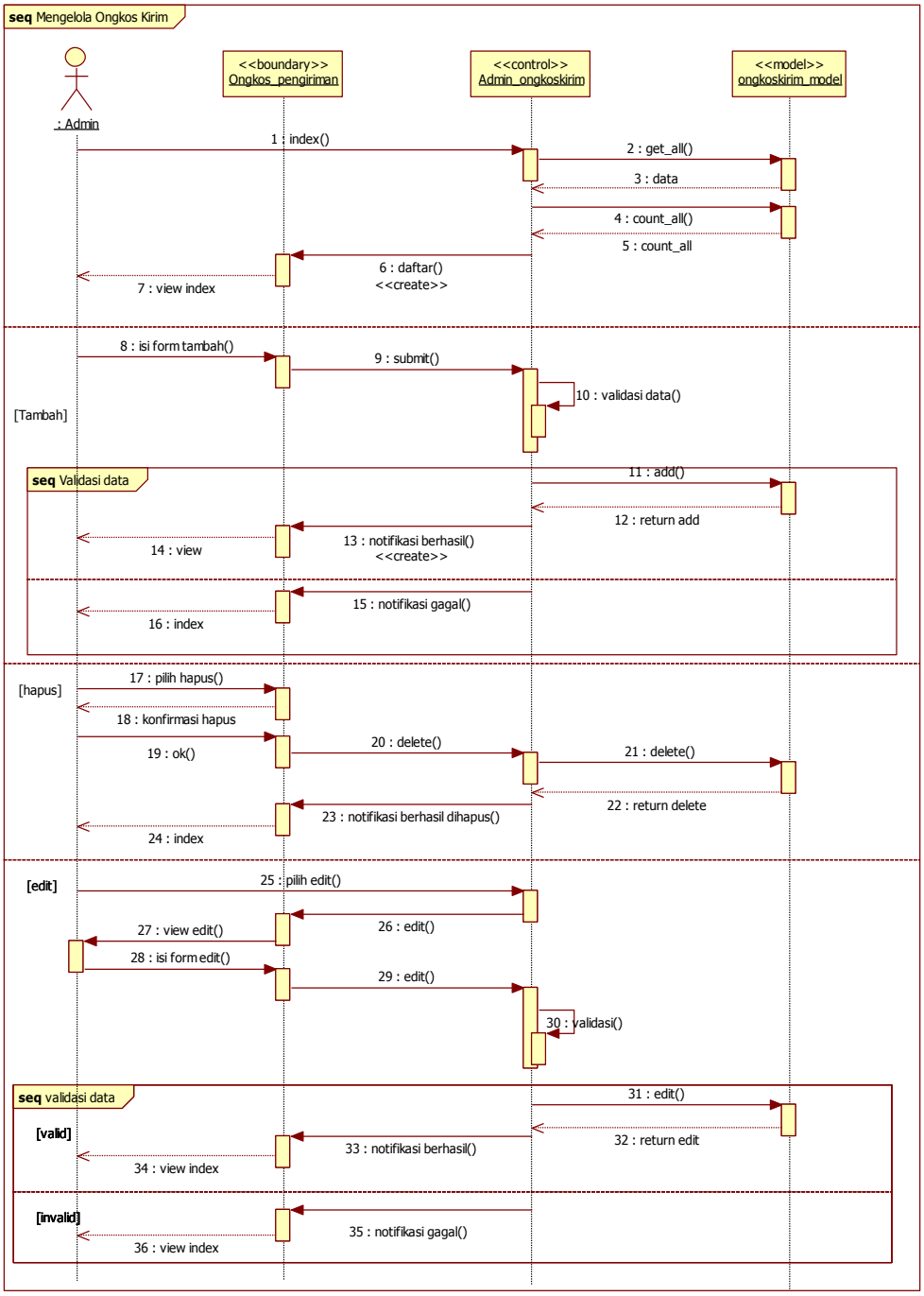

**Gambar 3.65 *Sequence* Diagram Mengelola Jasa Pengiriman**

## 30. Mengelola Paket Pengiriman




**Gambar 3.66 *Sequence* Diagram Mengelola Paket Pengiriman**

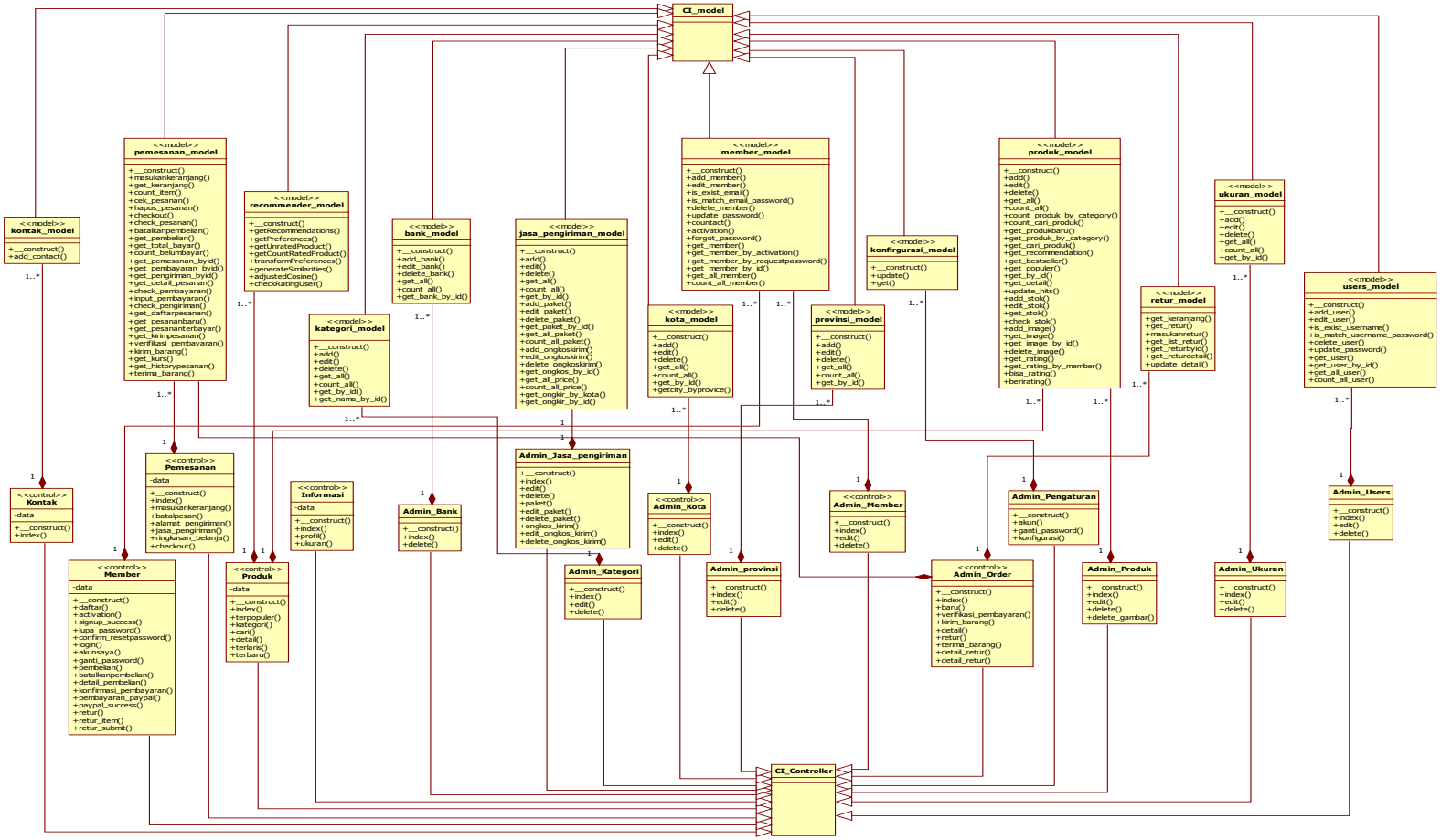
## 31. Mengelola Ongkos Kirim




**Gambar 3.67 *Sequence* Diagram Mengelola Ongkos Kirim**

### 3.1.7.7. *Class* Diagram

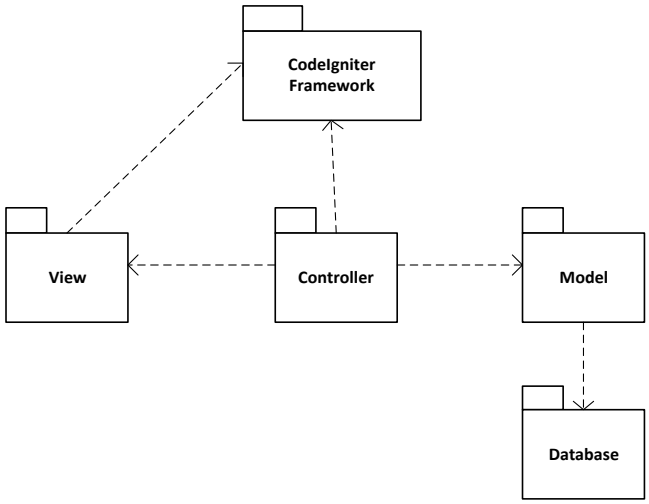
*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (fungsi). *Class* diagram dari *e-commerce* pada toko RusticShop dapat dilihat pada gambar 3.68




**Gambar 3.68 *Class* Diagram aplikasi e-commerce pada Toko RusticShop**

### 3.1.7.8. *Package Diagram*

*Package diagram* adalah sebuah bentuk pengelompokkan yang memungkinkan untuk mengambil setiap bentuk di UML dan mengelompokkan elemen-elemennya dalam tingkatan unit yang lebih tinggi. *Package* diagram untuk pengelompokkan *class* aplikasi *e-commerce* pada toko RusticShop dapat dilihat pada gambar 3.47

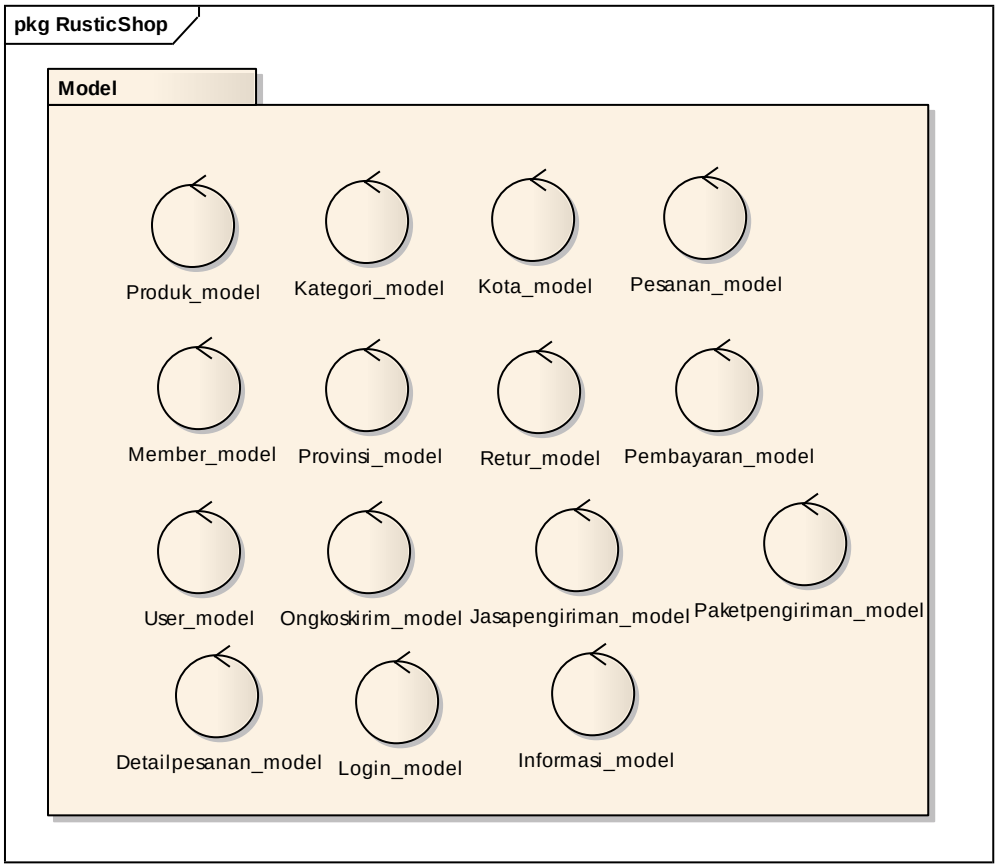



**Gambar 3.69 *Package* Diagram aplikasi *e-commerce* pada Toko RusticShop**

Deskripsi untuk *package diagram* pada gambar 3.69 adalah sebagai berikut :

1. *Package Model*

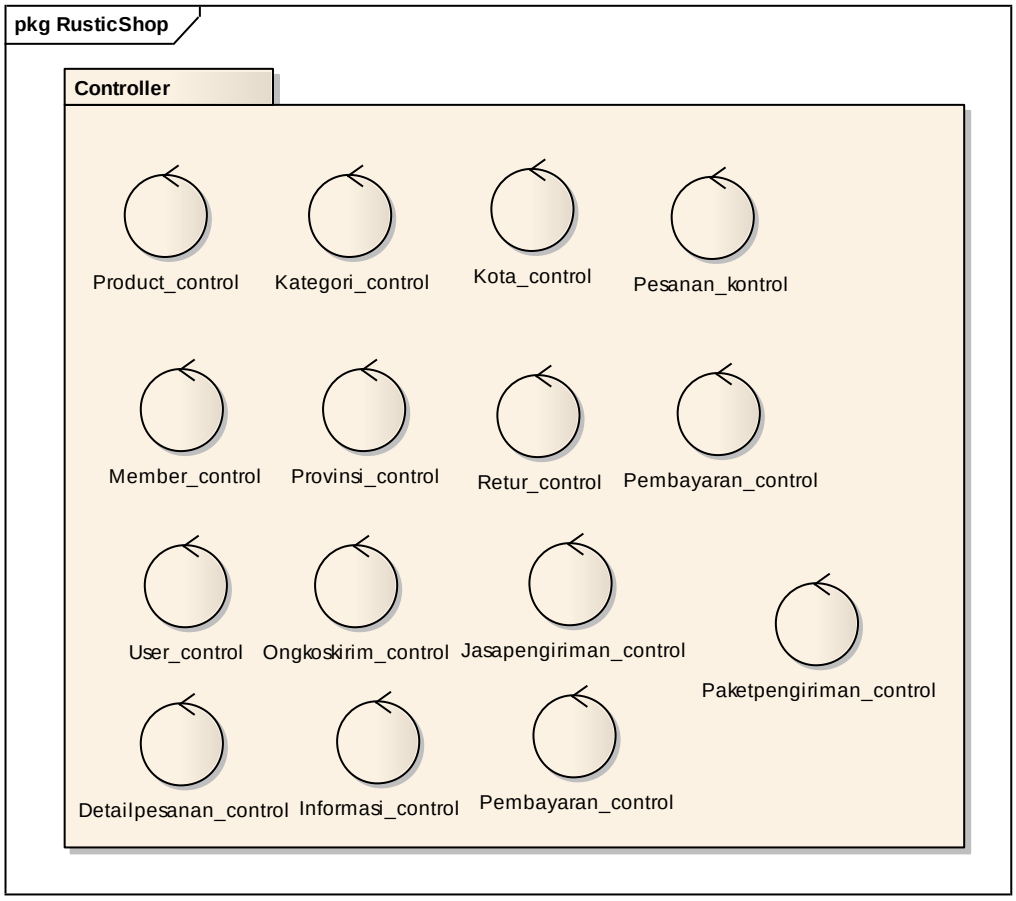
   *Package Model* berisi *class-class* model yang mengakses langsung pada *database* untuk dapat mengolah data. Isi dari *package* Model dapat dilihat pada gambar 3.70




**Gambar 3.70 Komponen P*ackage* Model**

2. *Package Controllers*

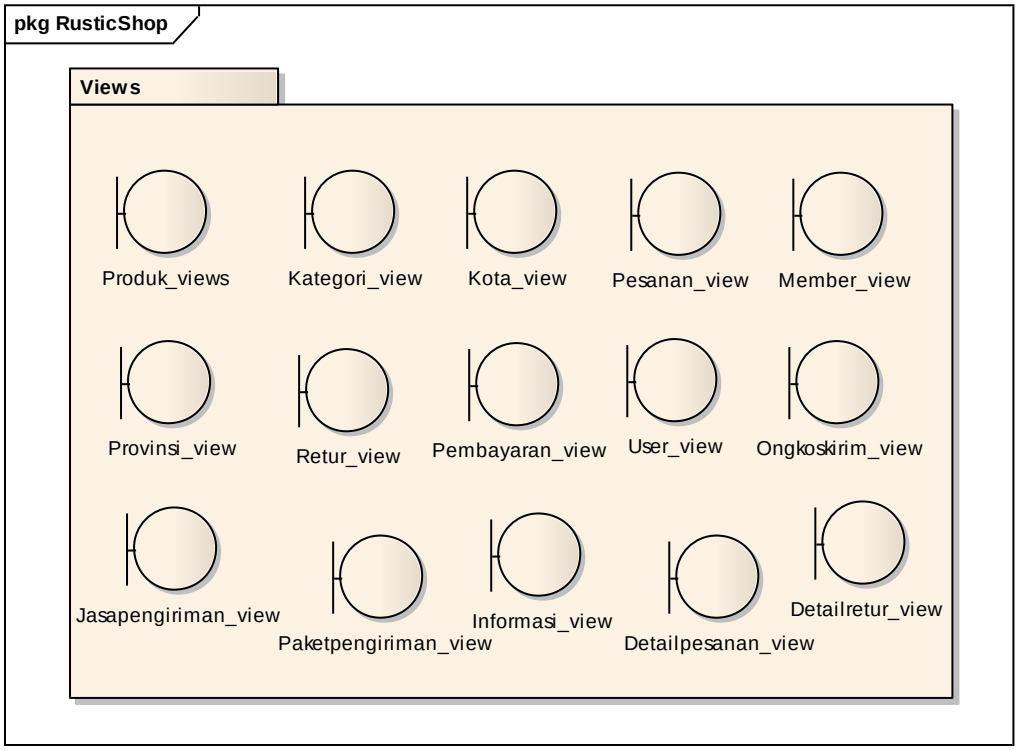
*Package Controllers* berisi *class-class control* yang merupakan pengatur dalam setiap proses yang ada. Isi dari *package Controllers* dapat dilihat pada gambar 3.71




**Gambar 3.71 Komponen *Package Controller***

3. *Package View*

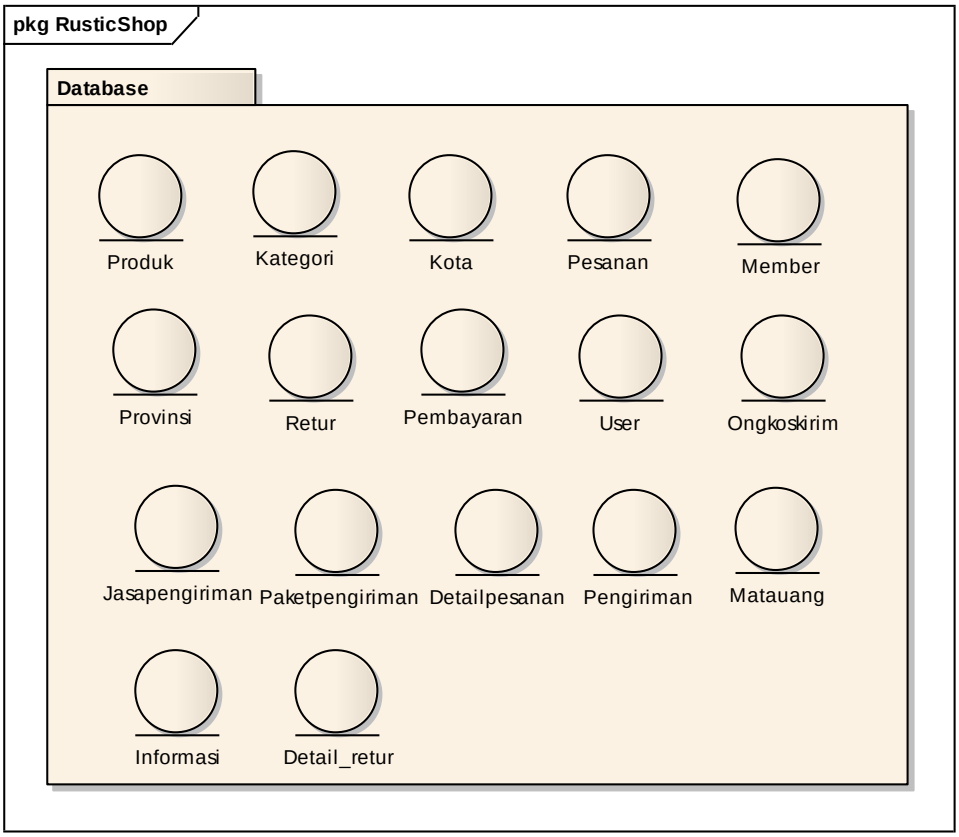
*Package View* berisi *class-class view* yang digunakan untuk menampilkan interface, dapat berupa tabel atau *form*. Isi dari *Package View* dapat dilihat pada gambar 3.72




**Gambar 3.72 Komponen *Package Views***

4. *Package Database*

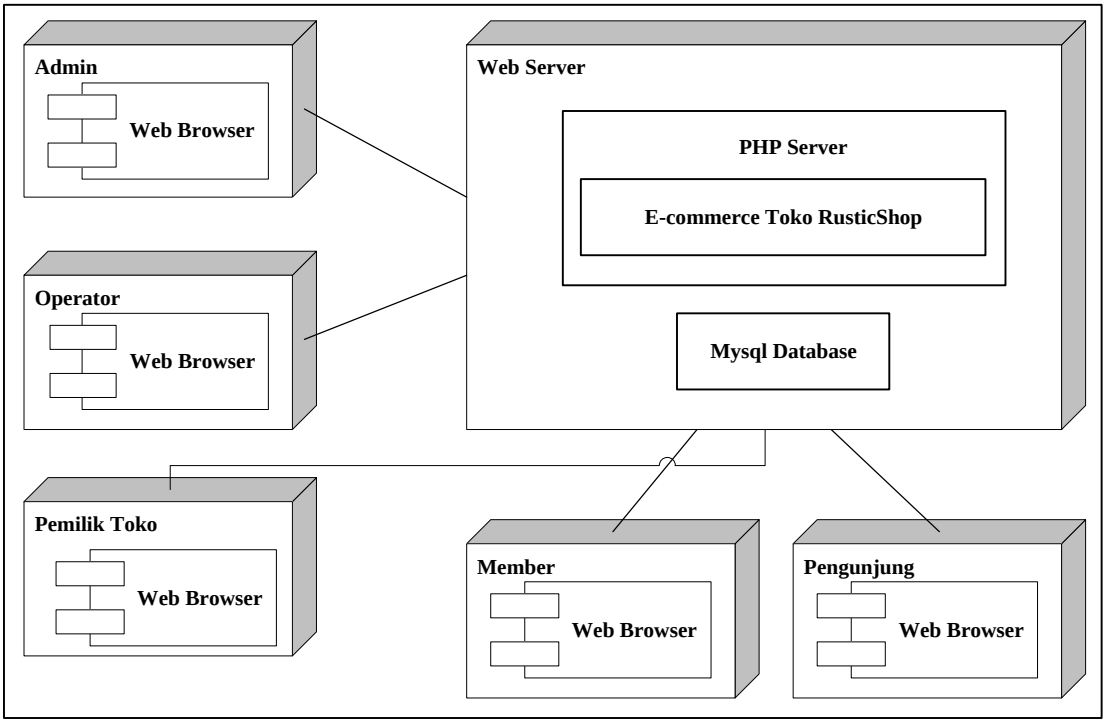
Package *database* berisi *class-class view* yang digunakan untuk menyimpan data. Isi dari *package database* dapat dilihat pada gambar 3.73




**Gambar 3.73 Komponen *Package Database***

### 3.1.7.9. Deployment Diagram

Menggambarkan sumber fisik dalam sistem, termasuk *node*, komponen dan koneksi (model implementasi sistem yang statistik). Dalam hal ini meliputi topologi *hardware* yang dipakai sistem. Deployment dari aplikasi *e-commerce* pada toko RusticShop dapat dilihat pada gambar 3.74




**Gambar 3.74 *Deployment* Diagram aplikasi *E-commerce* toko RusticShop**

## 3.2. Perancangan Sistem

Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan sistem yang dimaksudkan untuk menggambarkan perbedaan antara sistem yang sedang berjalan dengan sistem yang diusulkan.

### 3.2.1. Perancangan Data

Perancangan data atau lebih dikenal dengan perancangan basis data yaitu menciptakan atau merancang data yang terhubung dan disimpan secara bersamasama. Untuk menggambarkannya digunakanlah skema relasi dan struktur *file*. Dari dua hasil tersebut, implementasi basis data akan bisa dikerjakan.

#### 3.2.1.1. Diagram Relasi

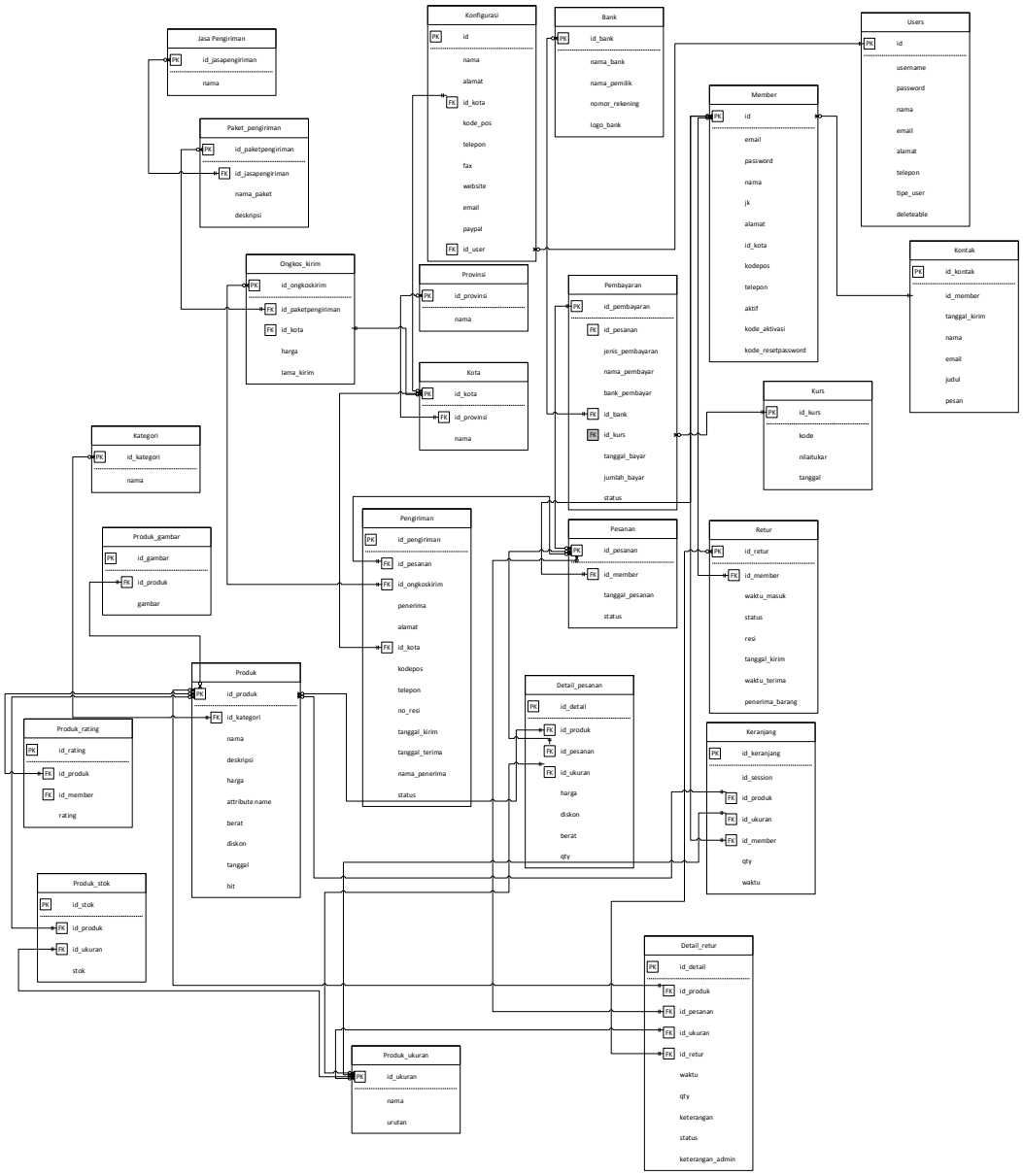
Skema relasi merupakan penggambaran hubungan diantara dua tabel atau lebih pada sistem *database*. Tabel yang ada pada aplikasi yang akan dibangun adalah sebagai berikut :

1. Bank {id_bank, nama_bank, nama pemilik, nomor_rekening, logo_bank}
2. Detail_pesanan {id_detail, id_produk, id_pesanan, id_ukuran, harga, berat, diskon, qty}
3. Detail_retur {id_detail, id_produk, id_pesanan, id_retur, id_ukuran, waktu, qty, keterangan, status, keterangan_admin}
4. Jasa Pengiriman { Id_jasapengiriman, nama }
5. Kategori { Id_kategori, nama }
6. Keranjang {id_keranjang, id_session, id_produk, id_ukuran, id_member, qty, waktu}
7. Konfigurasi {id, nama, alamat, id_kota, kode_pos, telepon, fax, website, email, paypal}
8. Kontak {id_kontak, id_member, tanggal_kirim, nama, email, judul, pesan}
9. Kota { Id_kota, id_provinsi, nama }

10. Kurs {id_kurs, kode, nilaitukar, tanggal}
11. Member {id, email, password, nama, jk, alamat, id_kota, kodepos, telepon, aktif, kode_aktivasi,kode_resetpassword}
12. Ongkos_kirim {id_ongkoskirim, id_paketpengiriman, id_kota, harga, lamakirim}
13. Paket_pengiriman {id_paketpengiriman, id_jasapengiriman, nama_paket, deskripsi}
14. Pembayaran {id_pembayaran, id_pesanan, jenis_pembayaran, nama_pembayar, bank_pembayar, id_bank, jumlah_bayar, status }
15. Pengiriman {id_pengiriman, id_pesanan, id_ongkoskirim, penerima, alamat, id_kota, kodepos, telepon, no_resi, tanggal_kirim, tanggal_terima, nama_penerima, status}
16. Pesanan {id_pesanan, id_member, tanggal_pesanan, status}
17. Produk {id_produk, id_kategori, nama, deskripsi, harga, berat, diskon, tanggal,hit}
18. Produk_gambar {id_gambar, id_produk,gambar}
19. Produk_rating {id_rating, id_produk, id_member, rating}
20. Produk_stok {id_stok, id_produk, id_ukuran, stok}
21. Produk_ukuran {id_ukuran, nama, urutan}
22. Provinsi {id_provinsi, nama}
23. Retur {id_retur, id_member, waktu_masuk, status, resi, tanggal_kirim, waktu_terima, penerima_barang}

24. Users {id, username, password, nama, email, alamat, telepon, tipe_user, deleteable}

Skema relasi *E-commerce* pada toko RusticShop dapat dilihat pada gambar 3.75.




**Gambar 3.75 Skema Relasi *e-commerce* pada Toko RusticShop**

### 3.2.1.2. Struktur Tabel

Struktur tabel adalah spesifikasi *file* dari tiap tabel yang digunakan dalam sistem yang akan dibangun. Struktur tabel dari sistem yang dibangun adalah sebagai berikut:

1. **Tabel bank**

Tabel ini berfungsi untuk menyimpan data bank. Struktur tabel bank adalah sebagai berikut :

**Tabel 3.42 Tabel Bank**

| Column | Type | Null | Extra | Comments |
|---|---|---|---|---|
| id_bank | int(11) | No | AUTO_INCREMENT | Primary Key |
| nama_bank | varchar(100) | No | | |
| nama_pemilik | varchar(200) | No | | |
| nomor_rekening | varchar(50) | No | | |
| logo_bank | varchar(100) | No | | |

2. **Tabel detail_pesanan**

Tabel detail_pesanan digunakan untuk menyimpan detail pesanan. Struktur tabel detail_pesanan adalah sebagai berikut:

**Tabel 3.43 Tabel detail_pesanan**

| Column | Type | Null | Extra | Comments |
|---|---|---|---|---|
| id_detail | int(11) | No | AUTO_INCREMENT | Primary Key |
| id_produk | int(11) | No | | |
| id_pesanan | int(11) | No | | |
| id_ukuran | smallint(5) | No | | |
| harga | float | No | | |

| | | | | |
|---|---|---|---|---|
| berat | int(7) | No | | |
| diskon | float | No | | |
| qty | int(11) | No | | |

### 3. Tabel Detail_retur

Tabel detail_retur digunakan untuk menyimpan detail retur. Struktur tabel detail_retur adalah sebagai berikut:

**Tabel 3.44 Tabel Detail_retur**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_detail | int(11) | No | | Auto_increment, primary_key |
| id_pesanan | int(11) | No | | |
| id_produk | int(11) | No | | |
| id_retur | int(11) | No | | |
| id_ukuran | smallint(5) | No | | |
| waktu | datetime | No | | |
| qty | smallint(3) | No | | |
| keterangan | text | Yes | *NULL* | |
| status | enum('Waiting', 'Diterima', 'Ditolak') | No | Waiting | |
| keterangan_admin | text | Yes | *NULL* | |

### 4. Tabel Jasa Pengiriman

Tabel jasa_pengiriman digunakan untuk menyimpan data jasa pengiriman. Struktur tabel jasa_pengiriman adalah sebagai berikut:

**Tabel 3.45 Tabel Jasa_pengiriman**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_jasapengiriman | int(11) | No | | Auto_increment, primary_key |
| nama | varchar(50) | No | | |

**5. Tabel Kategori**

Tabel kategori digunakan untuk menyimpan data kategori. Struktur tabel kategori adalah sebagai berikut:

**Tabel 3.46 Tabel Kategori**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_kategori | int(11) | No | | Auto_increment, primary_key |
| nama | varchar(50) | No | | |

**6. Tabel Keranjang**

Tabel keranjang digunakan untuk menyimpan data pesanan member. Struktur tabel keranjang adalah sebagai berikut:

**Tabel 3.47 Tabel Keranjang**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_keranjang | int(11) | No | | Auto_increment, primary_key |
| id_session | varchar(100) | No | | |
| id_produk | int(11) | No | | |
| id_ukuran | smallint(6) | No | | |

| | | | | |
|---|---|---|---|---|
| id_member | int(11) | No | | |
| qty | int(11) | No | | |
| waktu | timestamp | No | CURRENT_TIMESTAMP | |

**7. Tabel Konfigurasi**

Tabel konfigurasi digunakan untuk menyimpan konfigurasi toko. Struktur tabel konfigurasi adalah sebagai berikut:

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id | int(11) | No | | Auto_increment, primary_key |
| nama | varchar(200) | No | | |
| alamat | text | No | | |
| id_kota | int(11) | No | | |
| kodepos | varchar(6) | No | | |
| telepon | varchar(100) | No | | |
| fax | varchar(100) | No | | |
| website | varchar(200) | No | | |
| email | varchar(200) | No | | |
| paypal | varchar(200) | No | | |

**8. Tabel Kontak**

Tabel kontak digunakan untuk menyimpan data kontak. Struktur tabel kontak adalah sebagai berikut:

**Tabel 3.48 Tabel Kontak**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_kontak | int(11) | No | | Auto_increment, primary_key |
| id_member | int(11) | No | | |
| tanggal_kirim | datetime | No | | |
| nama | varchar(200) | No | | |
| email | varchar(250) | No | | |
| judul | varchar(50) | No | | |
| pesan | text | No | | |

**9. Tabel Kota**

Tabel kota digunakan untuk menyimpan data kota. Struktur tabel kota adalah sebagai berikut:

**Tabel 3.49 Tabel Kota**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_kota | int(11) | No | | Auto_increment, primary_key |
| id_provinsi | smallint(5) | No | | |
| nama | varchar(50) | No | | |

**10. Tabel Kurs**

Tabel kurs digunakan untuk menyimpan data nila tukar. Struktur tabel kurs adalah sebagai berikut:

**Tabel 3.50 Tabel Kurs**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_kurs | int(11) | No | | Auto_increment, primary_key |
| kode | char(4) | No | | |
| nilaitukar | float | No | | |
| tanggal | date | No | | |

**11. Tabel Member**

Tabel member digunakan untuk menyimpan data member. Struktur tabel member adalah sebagai berikut:

**Tabel 3.51 Tabel Member**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id | int(11) | No | | Auto_increment, primary_key |
| email | varchar(200) | No | | |
| password | varchar(100) | No | | |
| nama | varchar(200) | No | | |
| jk | enum('Laki-laki', 'Perempuan') | No | Laki-laki | |
| alamat | varchar(200) | No | | |
| id_kota | smallint(5) | No | | |
| kodepos | varchar(6) | No | | |
| telepon | varchar(25) | No | | |
| aktif | tinyint(1) | No | | |
| kode_aktivasi | varchar(20) | No | | |
| kode_resetpassword | varchar(20) | No | | |

**12. Tabel Ongkos_kirim**

Tabel ongkos_kirim digunakan untuk menyimpan data ongkos kirim. Struktur tabel ongkos_kirim adalah sebagai berikut:

**Tabel 3.52 Tabel Ongkos_kirim**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_ongkoskirim | int(11) | No | | Auto_increment, primary_key |
| id_paketpengiriman | int(11) | No | | |
| id_kota | int(11) | No | | |
| harga | float | No | | |
| lamakirim | tinyint(3) | No | | |

**13. Tabel Paket_pengiriman**

Tabel paket_pengiriman digunakan untuk menyimpan data paket pengiriman. Struktur tabel paket_pengiriman adalah sebagai berikut:

**Tabel 3.53 Tabel Paket_pengiriman**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_paketpengiriman | int(11) | No | | Auto_increment, primary_key |
| id_jasapengiriman | int(11) | No | | |
| nama_paket | varchar(50) | No | | |
| deskripsi | text | No | | |

**14. Tabel Pembayaran**

Tabel pembayaran digunakan untuk menyimpan data pembayaran. Struktur tabel pembayaran adalah sebagai berikut:

**Tabel 3.54 Tabel Pembayaran**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_pembayaran | int(11) | No | | Auto_increment, primary_key |
| id_pesanan | int(11) | No | | |
| id_kurs | int(11) | No | | |
| jenis_pembayaran | varchar(20) | No | | |
| nama_pembayar | varchar(50) | No | | |
| bank_pembayar | varchar(50) | No | | |
| id_bank | int(11) | No | | |
| jumlah_bayar | float | No | | |
| id_transaksi | varchar(50) | Yes | *NULL* | |
| status | tinyint(1) | No | | |

**15. Tabel Pengiriman**

Tabel pengiriman digunakan untuk menyimpan data pengiriman. Struktur tabel pengiriman adalah sebagai berikut:

**Tabel 3.55 Tabel Pengiriman**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_pengiriman | int(11) | No | | Auto_increment, primary_key |
| id_pesanan | int(11) | No | | |
| id_ongkoskirim | int(11) | No | | |

| | | | | |
|---|---|---|---|---|
| penerima | varchar(100) | No | | |
| alamat | text | No | | |
| id_kota | int(11) | No | | |
| kodepos | int(6) | No | | |
| telepon | varchar(32) | No | | |
| no_resi | varchar(50) | No | | |
| tanggal_kirim | date | No | | |
| tanggal_terima | varchar(50) | No | | |
| nama_penerima | varchar(100) | No | | |
| status | tinyint(1) | No | | |

**16. Tabel Pesanan**

Tabel pesanan digunakan untuk menyimpan data pesanan. Struktur tabel pesanan adalah sebagai berikut:

**Tabel 3.56 Tabel Pesanan**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_pesanan | int(11) | No | | Auto_increment, primary_key |
| id_member | int(11) | No | | |
| tanggal_pesanan | timestamp | No | CURRENT_TIMESTAMP | |
| status | tinyint(4) | No | | |

**17. Tabel Produk**

Tabel produk digunakan untuk menyimpan data produk. Struktur tabel produk adalah sebagai berikut:

**Tabel 3.57 Tabel Produk**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_produk | int(11) | No | | Auto_increment, primary_key |
| id_kategori | int(11) | No | | |
| nama | varchar(50) | No | | |
| deskripsi | text | No | | |
| harga | float | No | 0 | |
| berat | int(7) | No | 0 | |
| diskon | float | No | 0 | |
| tanggal | timestamp | No | CURRENT_TIMESTAMP | |
| hit | int(11) | No | 0 | |

**18. Tabel Produk_gambar**

Tabel produk_gambar digunakan untuk menyimpan data gambar dari masing-masing produk. Struktur tabel produk_gambar adalah sebagai berikut:

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_gambar | int(11) | No | | Auto_increment, primary_key |
| id_produk | int(11) | No | | |
| gambar | varchar(100) | No | | |

**19. Tabel Produk_rating**

Tabel produk_rating digunakan untuk menyimpan data rating untuk masing-masing produk oleh member. Struktur tabel produk_rating adalah sebagai berikut:

**Tabel 3.58 Produk_rating**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_rating | int(11) | No | | Auto_increment, primary_key |
| id_produk | int(11) | No | | |
| id_member | int(11) | No | | |
| rating | tinyint(1) | No | | |

**20. Tabel Produk_stok**

Tabel produk_stok digunakan untuk menyimpan data stok produk. Struktur tabel produk_stok adalah sebagai berikut:

**Tabel 3.59 Tabel Produk_stok**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_stok | int(11) | No | | Auto_increment, primary_key |
| id_produk | int(11) | No | | |
| id_ukuran | smallint(5) | No | | |
| stok | smallint(5) | No | | |

**21. Tabel Produk_ukuran**

Tabel produk_ukuran digunakan untuk menyimpan data ukuran produk. Struktur tabel produk_ukuran adalah sebagai berikut:

**Tabel 3.60 Tabel Produk_ukuran**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_ukuran | smallint(5) | No | | Auto_increment, primary_key |

| | | | | |
|---|---|---|---|---|
| nama | varchar(50) | No | | |
| urutan | tinyint(1) | No | 0 | |

**22. Tabel Provinsi**

Tabel provinsi digunakan untuk menyimpan data provinsi. Struktur tabel provinsi adalah sebagai berikut:

**Tabel 3.61 Tabel Provinsi**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_provinsi | smallint(5) | No | | Auto_increment, primary_key |
| nama | varchar(50) | No | | |

**23. Tabel Retur**

Tabel Retur digunakan untuk menyimpan data retur barang. Struktur tabel retur adalah sebagai berikut:

**Tabel 3.62 Tabel Retur**

| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id_retur | int(11) | No | | Auto_increment, primary_key |
| id_member | int(11) | No | | |
| waktu_masuk | datetime | No | | |
| status | enum('Keranjang', 'Waiting', 'Sudah Direspon', 'Retur telah Dikirim', 'Retur telah Diterima', 'Retur Ditolak') | No | Waiting | |
| resi | varchar(20) | Yes | *NULL* | |
| tanggal_kirim | date | Yes | *NULL* | |

| | | | | |
|---|---|---|---|---|
| waktu_terima | datetime | Yes | *NULL* | |
| penerima_barang | varchar(100) | Yes | *NULL* | |

**24. Tabel Users**

Tabel users digunakan untuk menyimpan data pengguna aplikasi. Struktur tabel users adalah sebagai berikut:

**Tabel 3.63 Tabel Users**

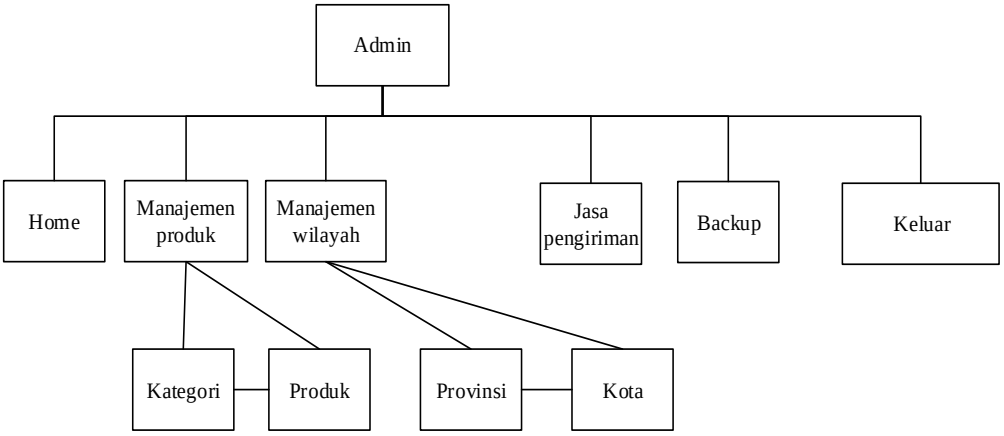
| Column | Type | Null | Default | Comments |
|---|---|---|---|---|
| id | int(11) | No | | |
| username | varchar(100) | No | | |
| password | varchar(100) | No | | |
| nama | varchar(200) | No | | |
| email | varchar(250) | No | | |
| alamat | text | No | | |
| telepon | varchar(50) | No | | |
| tipe_user | tinyint(1) | No | | '1' = admin, '2' = operator, '3' = pemilik |
| deleteable | tinyint(1) | No | | key ini juga bisa digunakan untuk hak akses super user dan user biasa |

### 3.2.2. Perancangan Arsitektur Menu

Menu dirancang untuk menggambarkan segala macam bentuk kegiatan yang bisa dilakukan dalam program aplikasi, seperti *input* data, cari data dan membuat laporan berdasarkan data *input*. Struktur menu yang dirancang sesuai dengan level pengguna sistem. Terdapat 5 pengguna aplikasi *e-commerce* pada

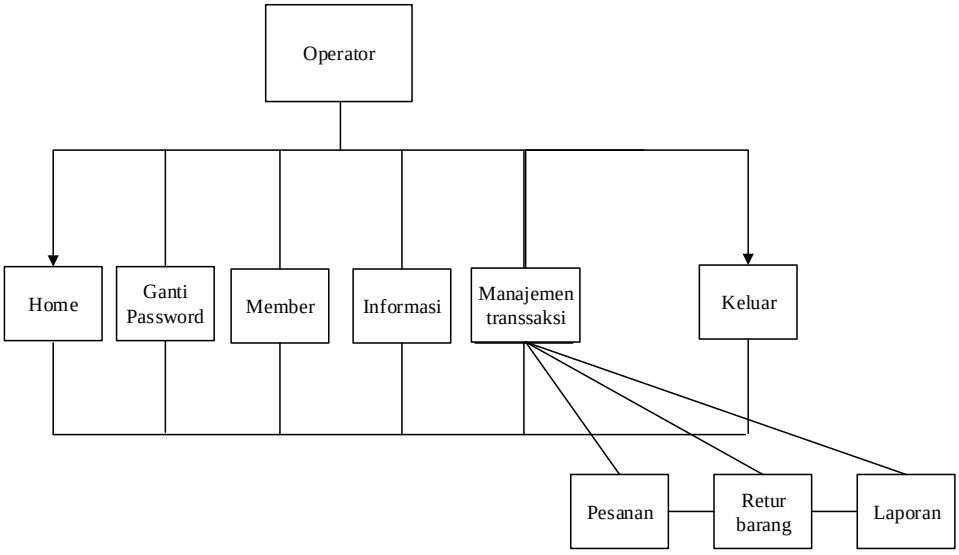
toko RusticShop yaitu *admin*, operator, pemilik toko , member dan pengunjung. Adapun struktur menu dari setiap pengguna adalah sebagai berikut:

1. **Struktur Menu *Admin***



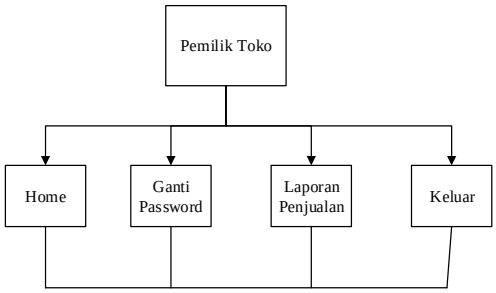

**Gambar 3.76 Struktur Menu *Admin***

2. **Struktur Menu Operator**



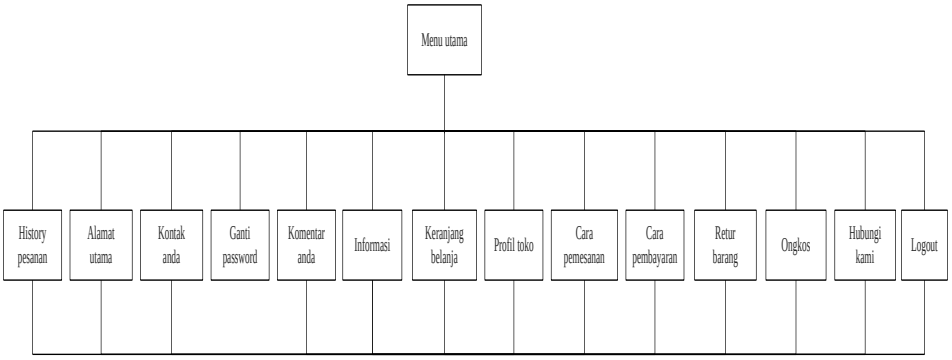

**Gambar 3.77 Struktur Menu *Operator***

### 3. Struktur Menu Pemilik Toko



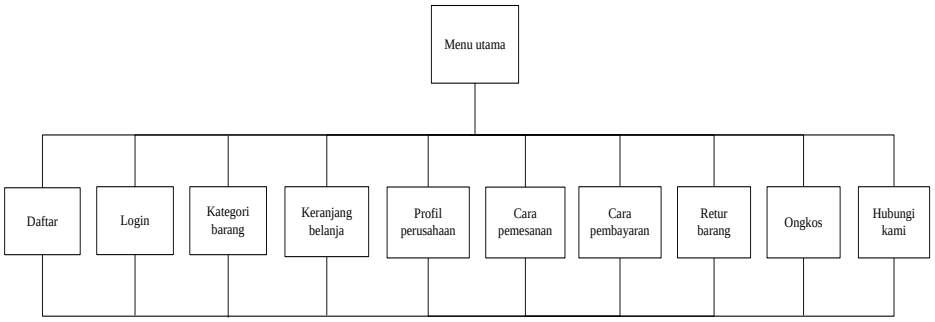

**Gambar 3.78 Struktur Pemilik Toko**

### 4. Struktur Menu Member




**Gambar 3.79 Struktur Menu *Member***

### 5. Struktur Menu Pengunjung



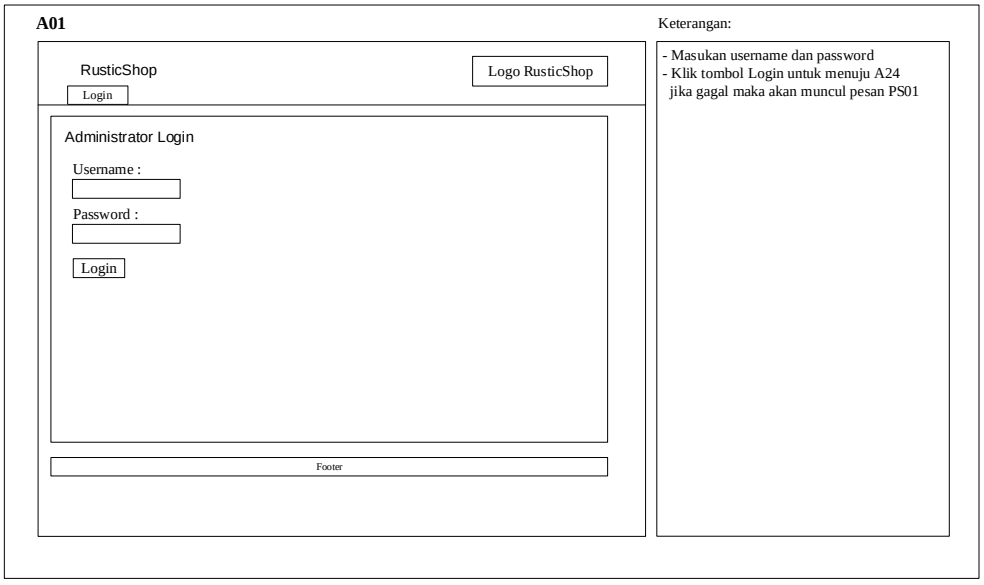

**Gambar 3.80 Struktur Menu Pengunjung**

### 3.2.3. Perancangan Antarmuka

*Interface* atau antar muka merupakan tampilan dari suau program aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara program dengan *user*. Sistem yang akan dibangun diharapkan menyediakan *interface* yang mudah dipahami dan digunakan oleh *user*.

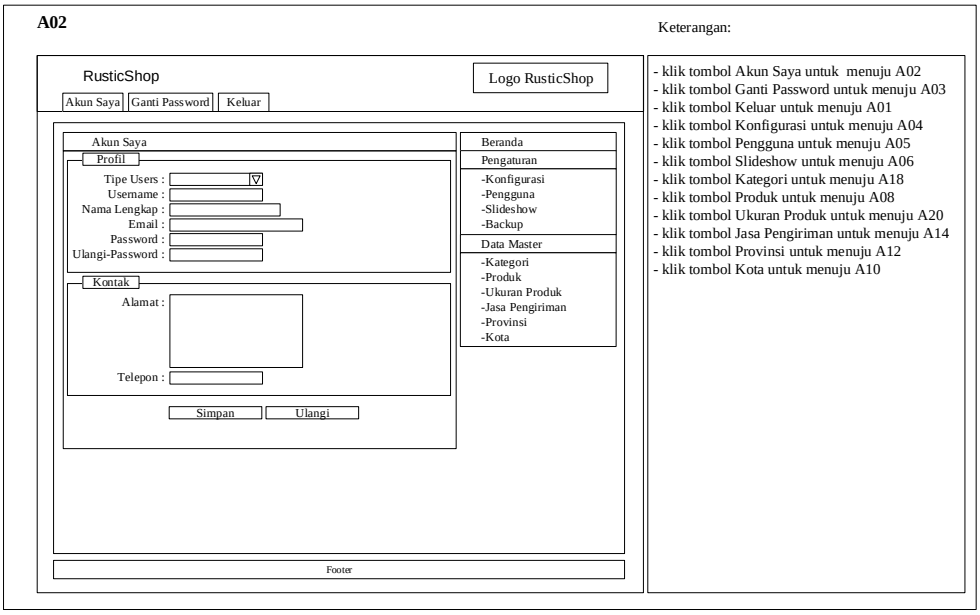
#### 3.2.3.1. Perancangan Antarmuka *Admin*

Perancangan tampilan admin - login (A01)



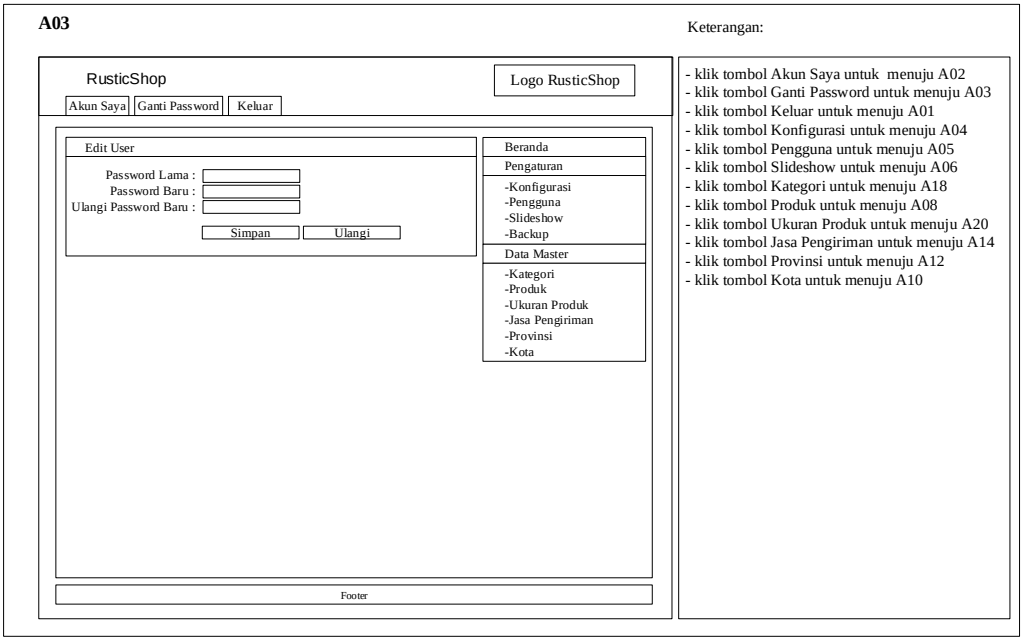

**Gambar 3.81 Perancangan tampilan admin – login (A01)**

Perancangan tampilan admin - Users (A02)



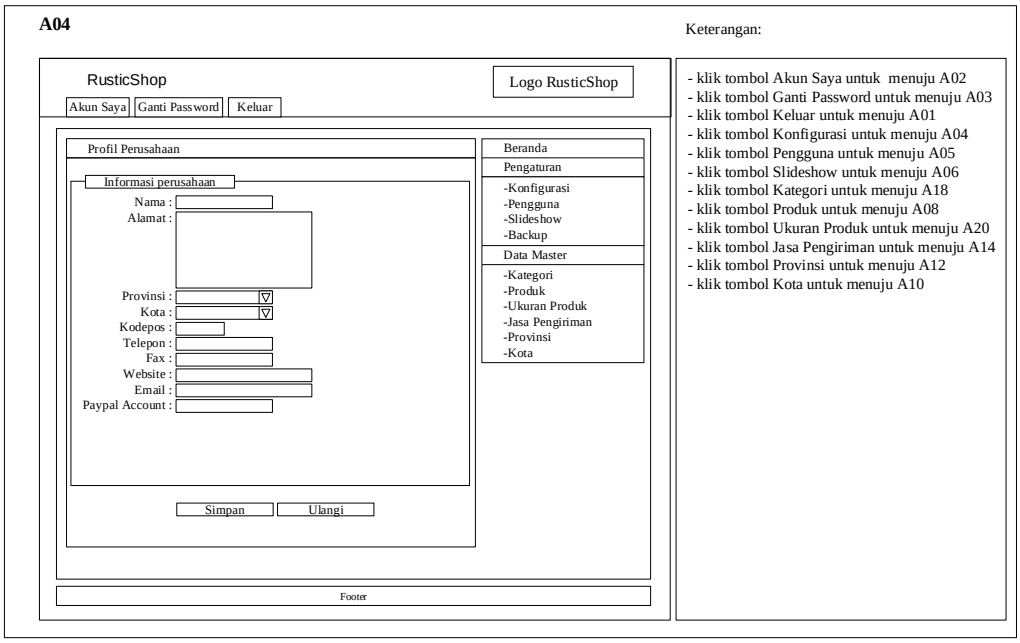

**Gambar 3.82 Perancangan tampilan admin – users (A02)**

Perancangan tampilan admin – Ganti Password (A03)



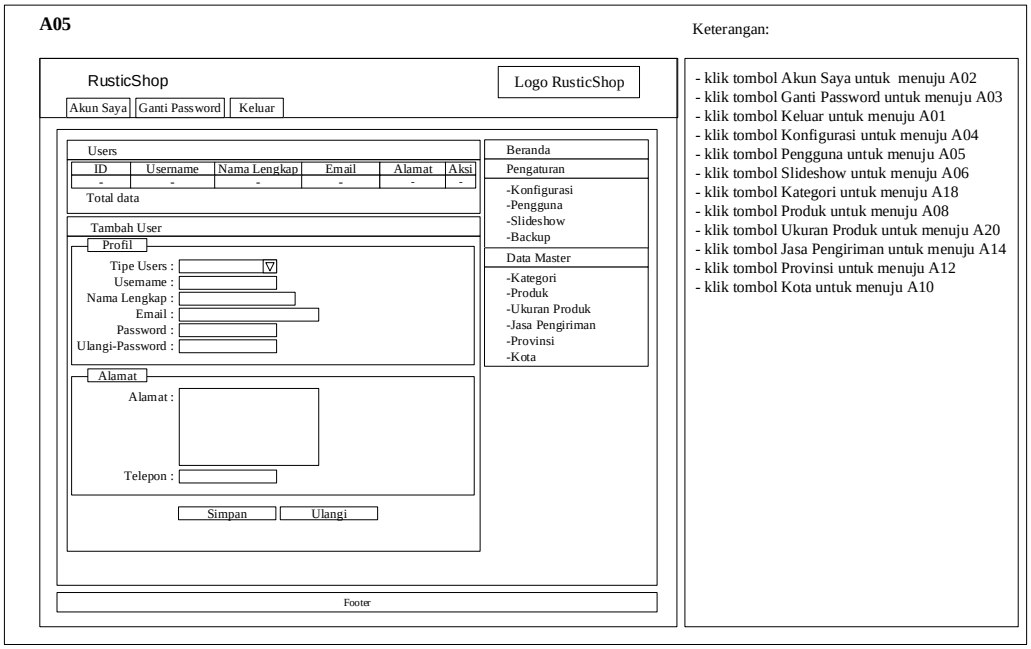

**Gambar 3.83 Perancangan tampilan admin – ganti password (A03)**

Perancangan tampilan admin - Konfigurasi (A04)



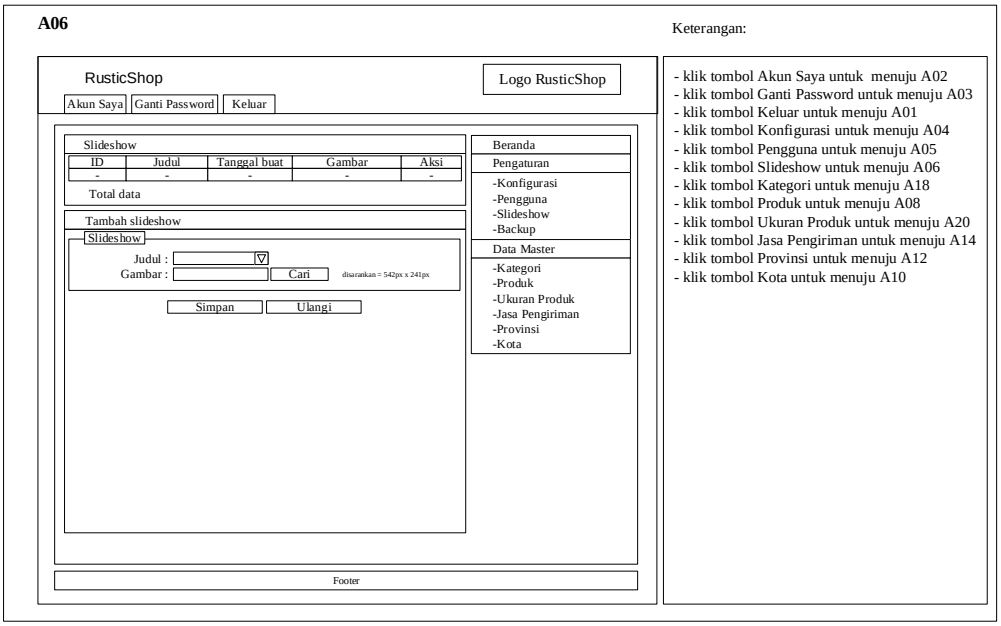

**Gambar 3.84 Perancangan tampilan admin – konfigurasi (A04)**

Perancangan tampilan admin – Pengguna (A05)



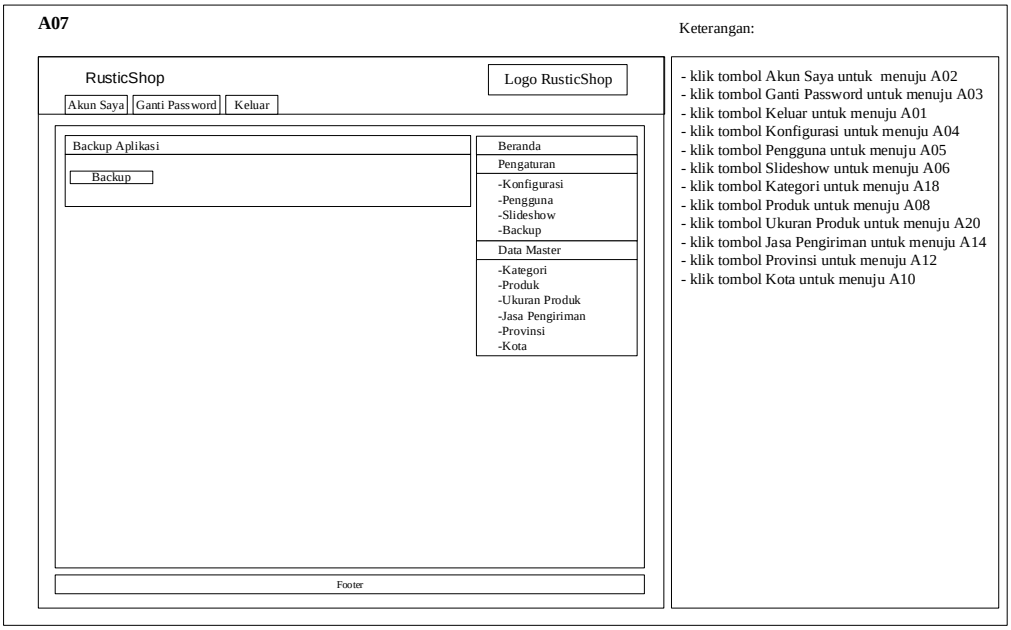

**Gambar 3.85 Perancangan tampilan admin – pengguna(A05)**

Perancangan tampilan admin - Slideshow (A06)



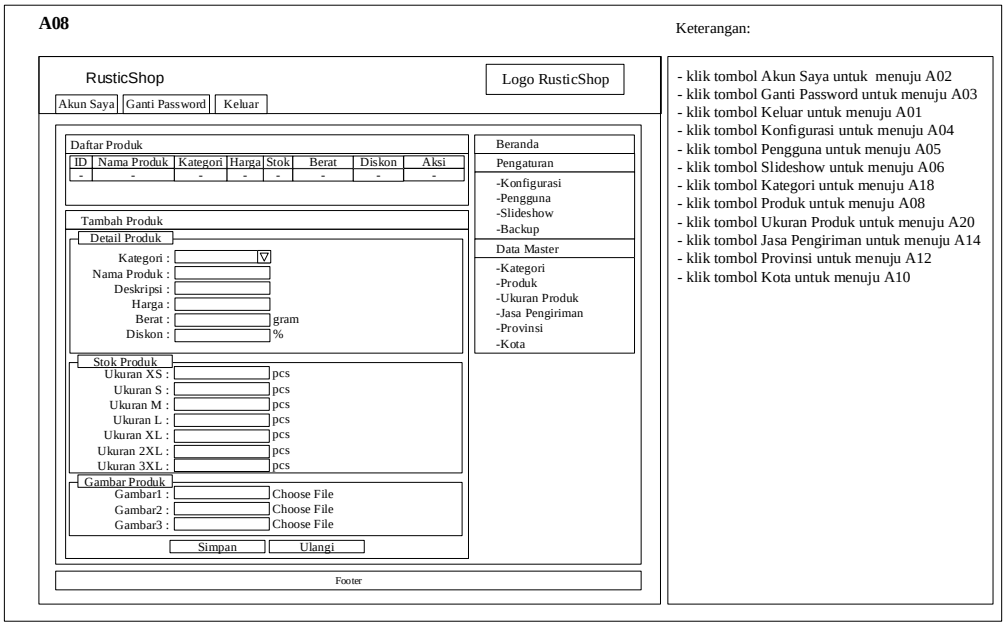

**Gambar 3.86 Perancangan tampilan admin – slideshow (A06)**

Perancangan tampilan admin – backup (A07)



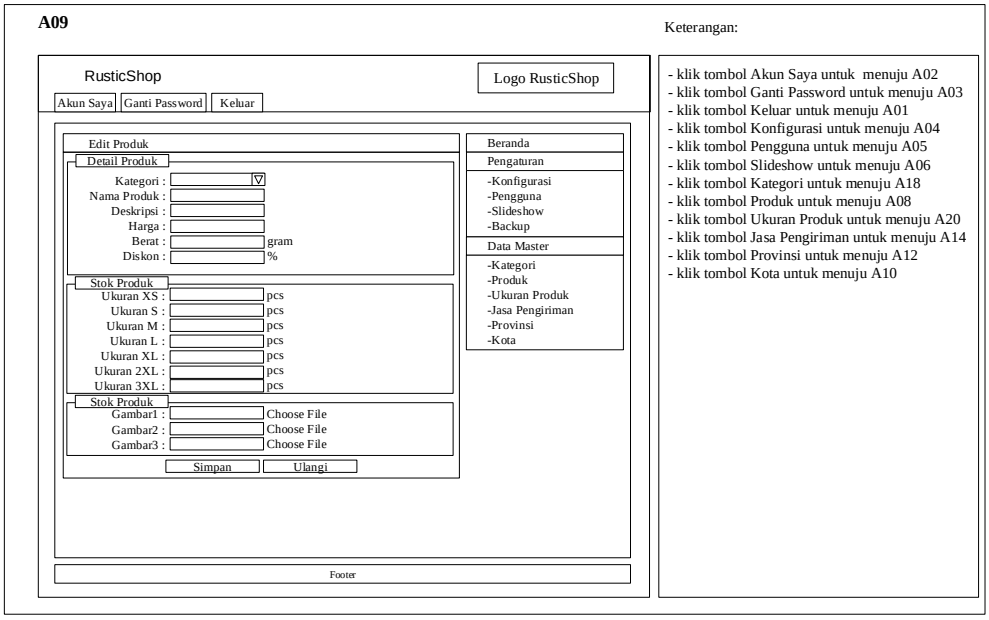

**Gambar 3.87 Perancangan tampilan admin – backup(A07)**

Perancangan tampilan admin - produk (A08)



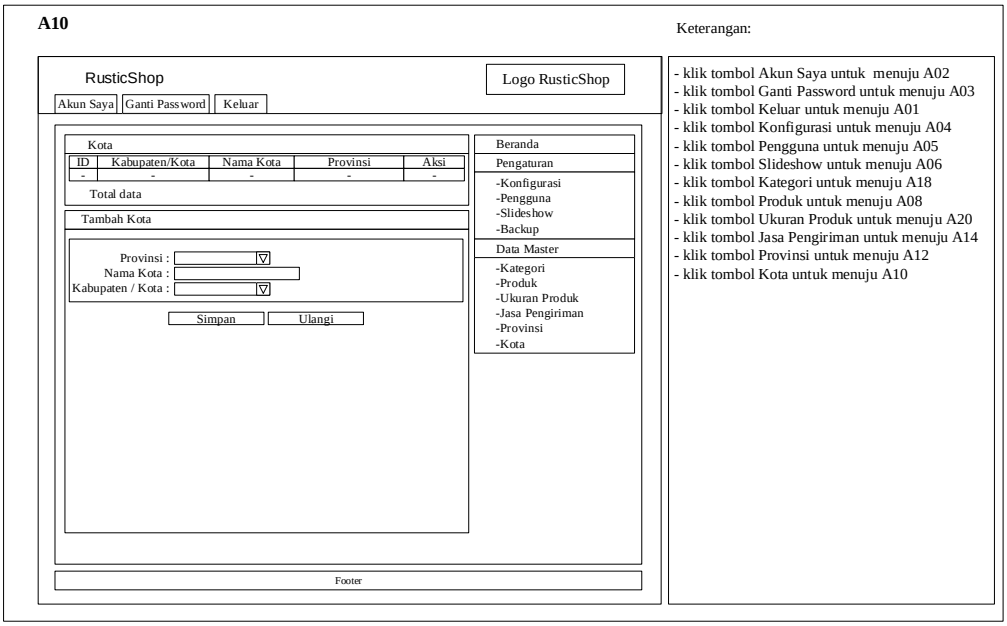

**Gambar 3.88 Perancangan tampilan admin – produk (A08)**

Perancangan tampilan admin – editproduk (A09)



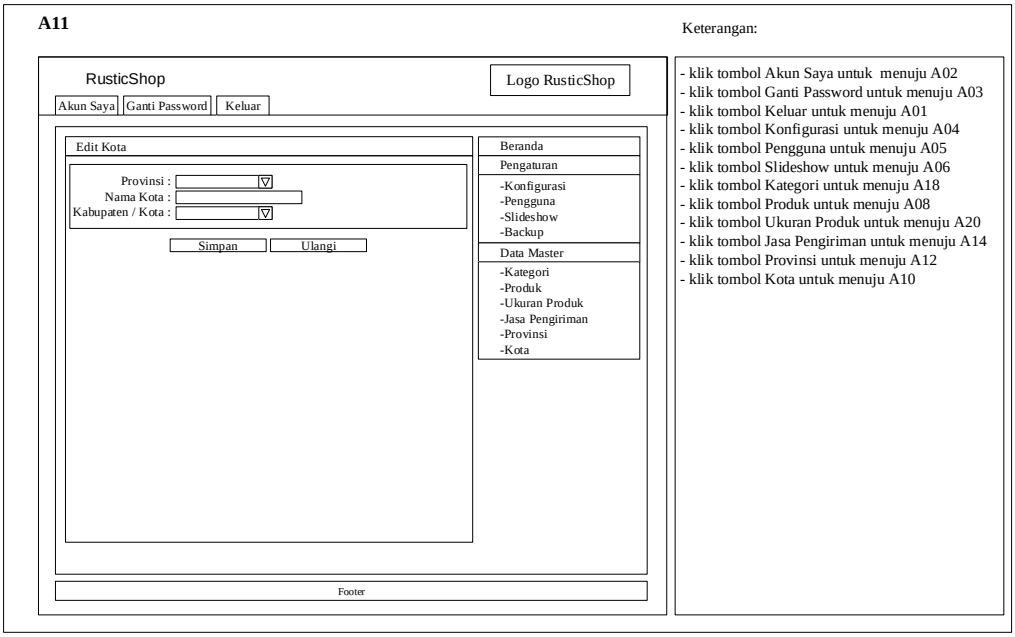

**Gambar 3.89 Perancangan tampilan admin – editproduk(A09)**

Perancangan tampilan admin - kota (A10)



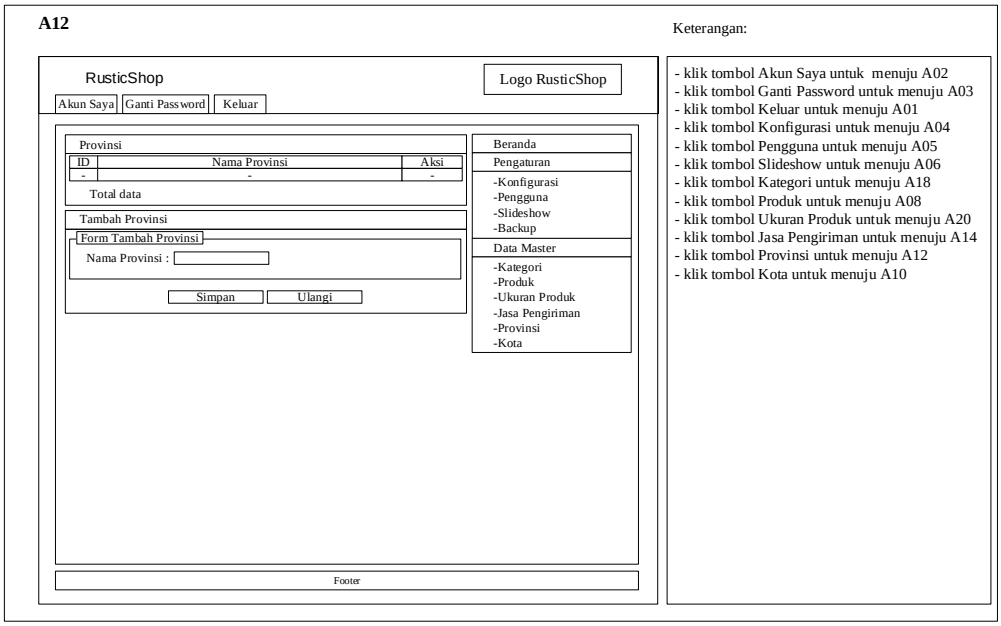

**Gambar 3.90 Perancangan tampilan admin – kota (A10)**

Perancangan tampilan admin – editkota(A11)



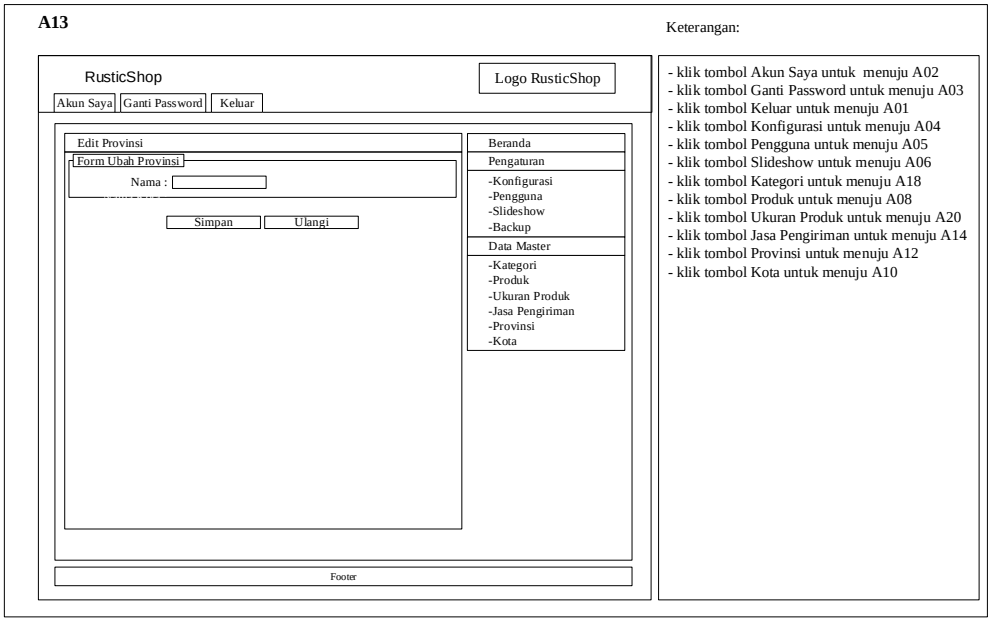

**Gambar 3.91 Perancangan tampilan admin – editkota(A11)**

Perancangan tampilan admin - provinsi (A12)



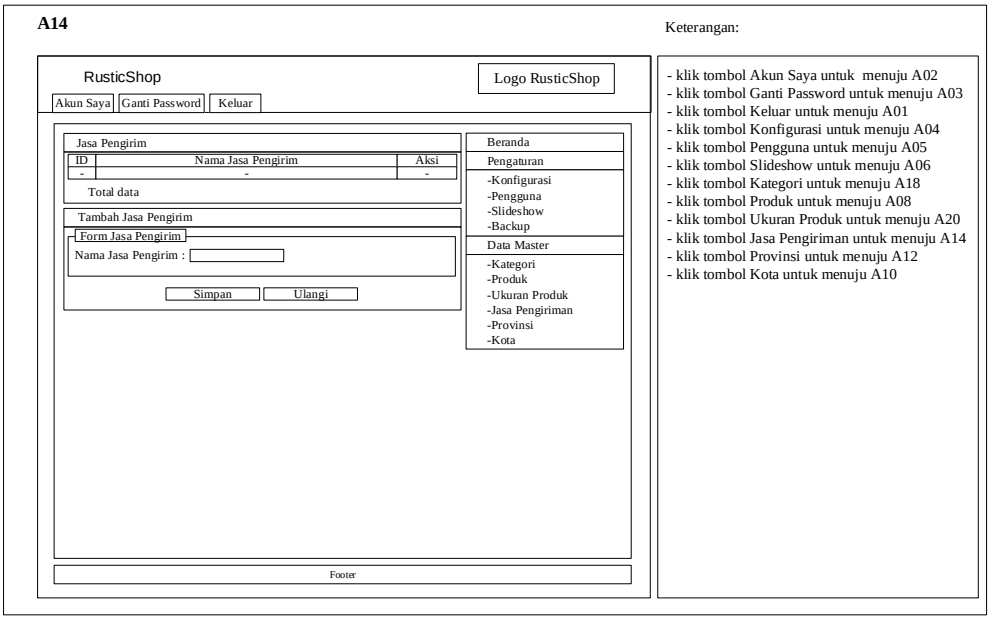

**Gambar 3.92 Perancangan tampilan admin – provinsi (A12)**

Perancangan tampilan admin – editprovinsi(A13)



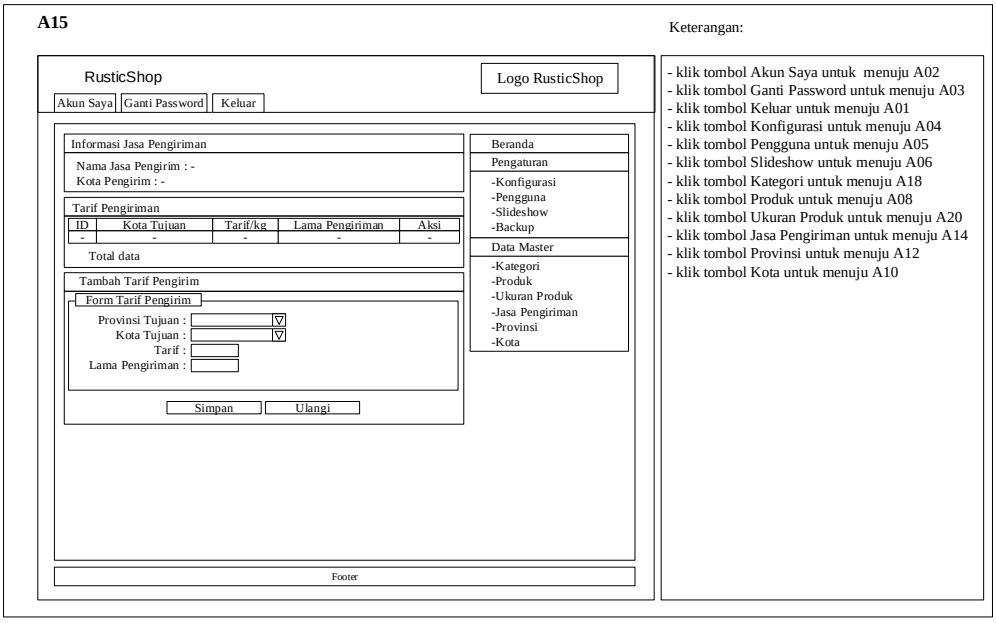

**Gambar 3.93 Perancangan tampilan admin – editprovinsi(A13)**

Perancangan tampilan admin - jasakirim (A14)



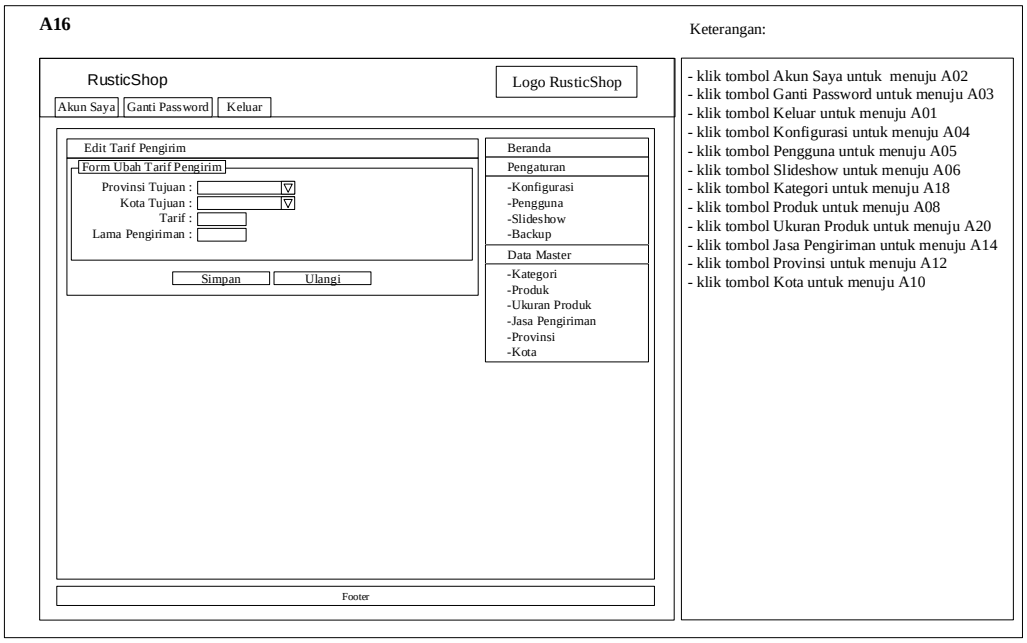

**Gambar 3.94 Perancangan tampilan admin – jasakirim (A14)**

Perancangan tampilan admin – tarifkirim(A15)



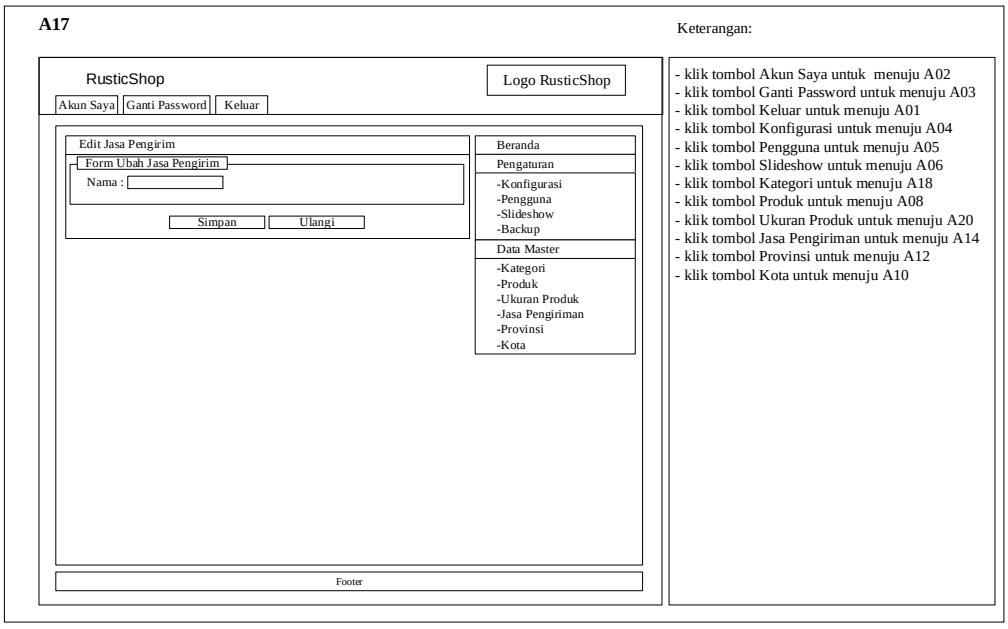

**Gambar 3.95 Perancangan tampilan admin – tarifkirim(A15)**

Perancangan tampilan admin - edittarif (A16)



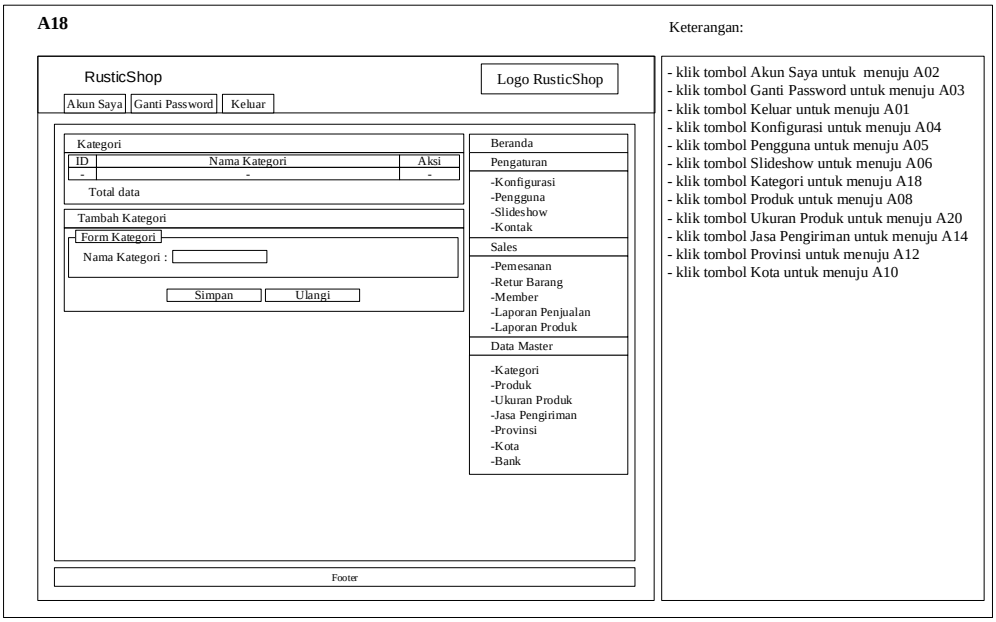

**Gambar 3.96 Perancangan tampilan admin – edittarif (A16)**

Perancangan tampilan admin – editjasa(A17)



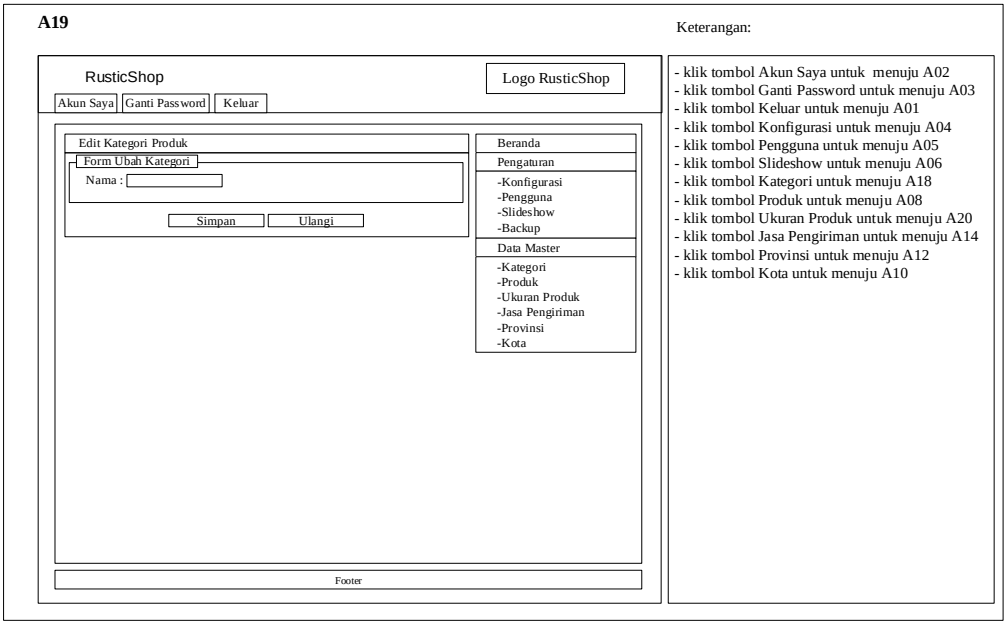

**Gambar 3.97 Perancangan tampilan admin – editjasa(A17)**

Perancangan tampilan admin - kategori(A18)



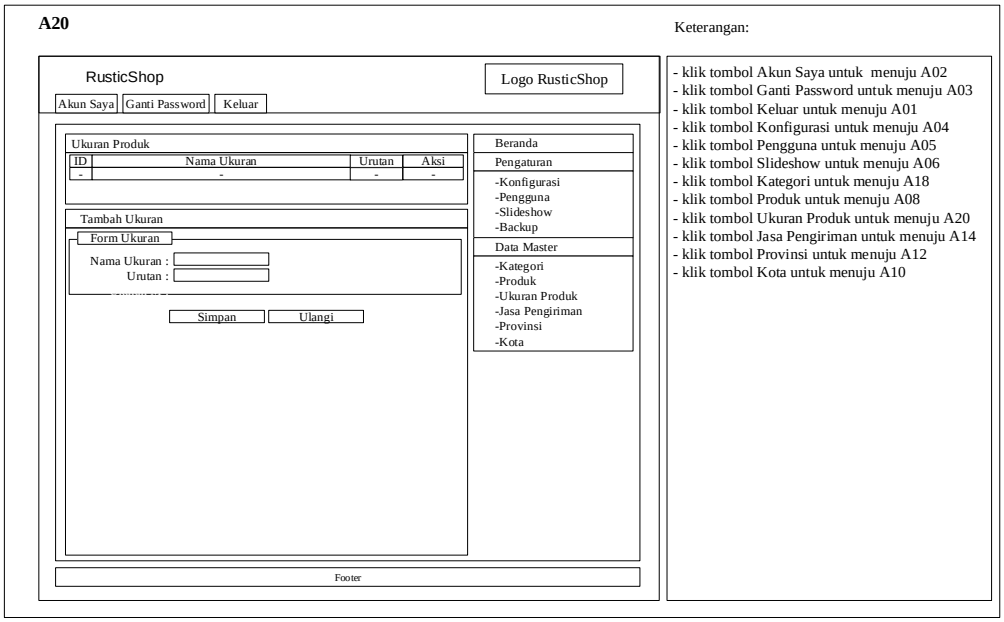

**Gambar 3.98 Perancangan tampilan admin – kategori(A18)**

Perancangan tampilan admin – editkategori(A19)



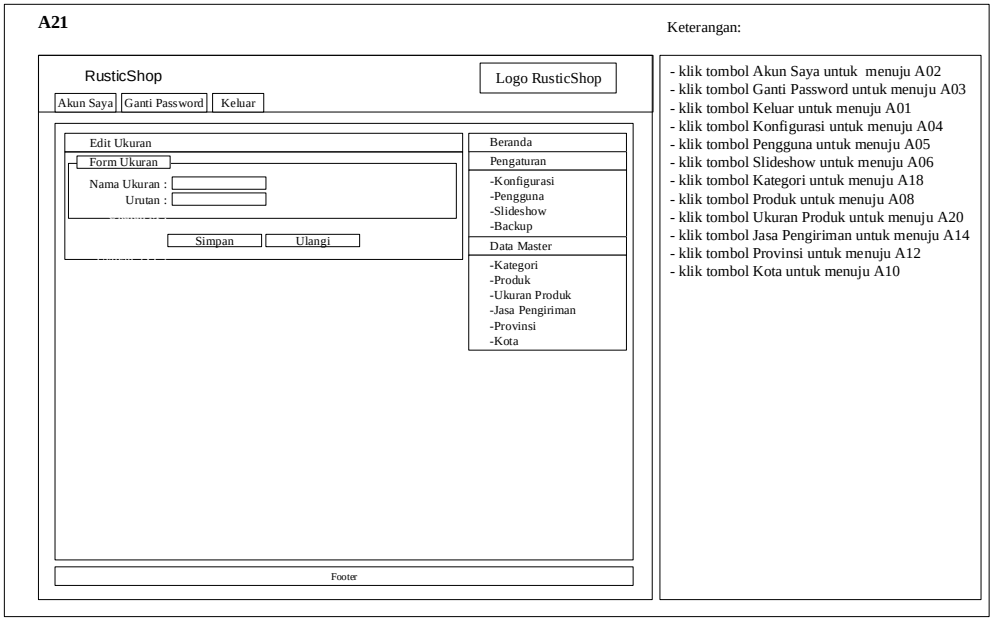

**Gambar 3.99 Perancangan tampilan admin – editkategori(A19)**

Perancangan tampilan admin - ukuran(A20)



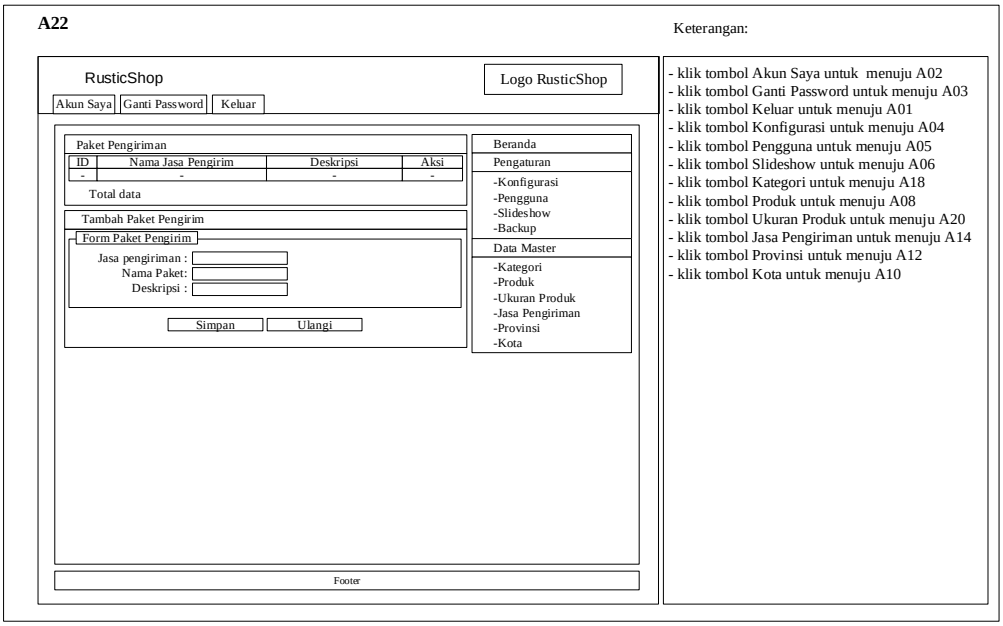

**Gambar 3.100 Perancangan tampilan admin – ukuran(A20)**

Perancangan tampilan admin – editukuran(A21)



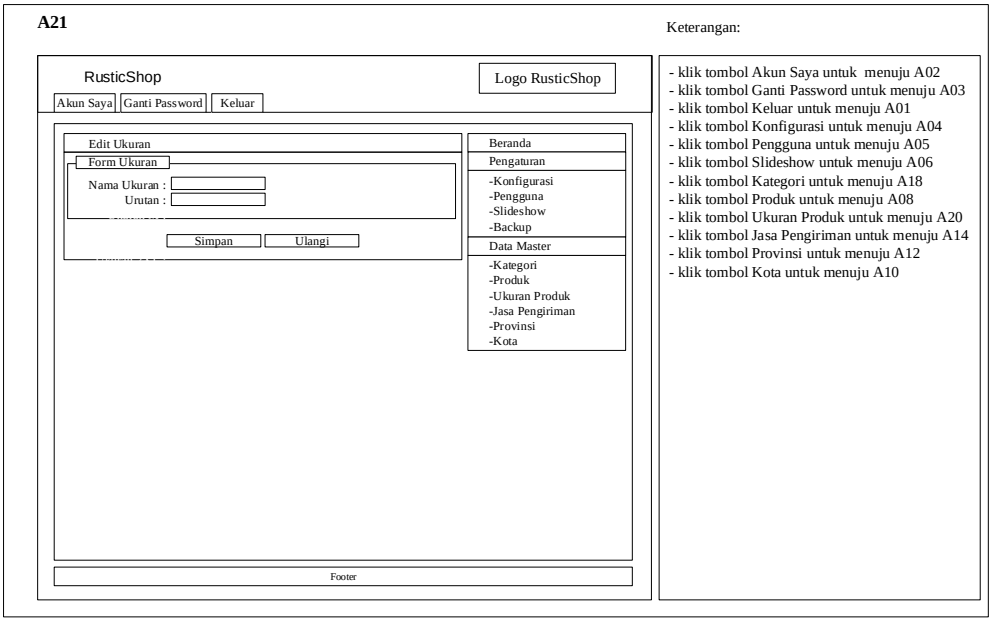

**Gambar 3.101 Perancangan tampilan admin – editkategori(A21)**

Perancangan tampilan admin - paketkirim (A22)



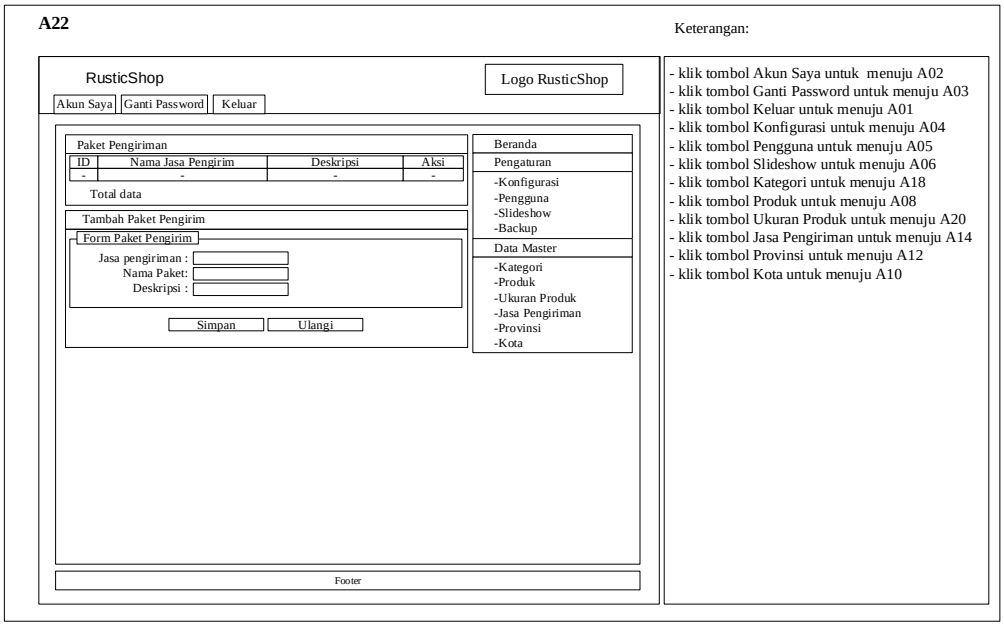

**Gambar 3.102 Perancangan tampilan admin – ukuran(A22)**

Perancangan tampilan admin – editpaketkirim(A23)




**Gambar 3.103 Perancangan tampilan admin – editpaketkirim(A23)**

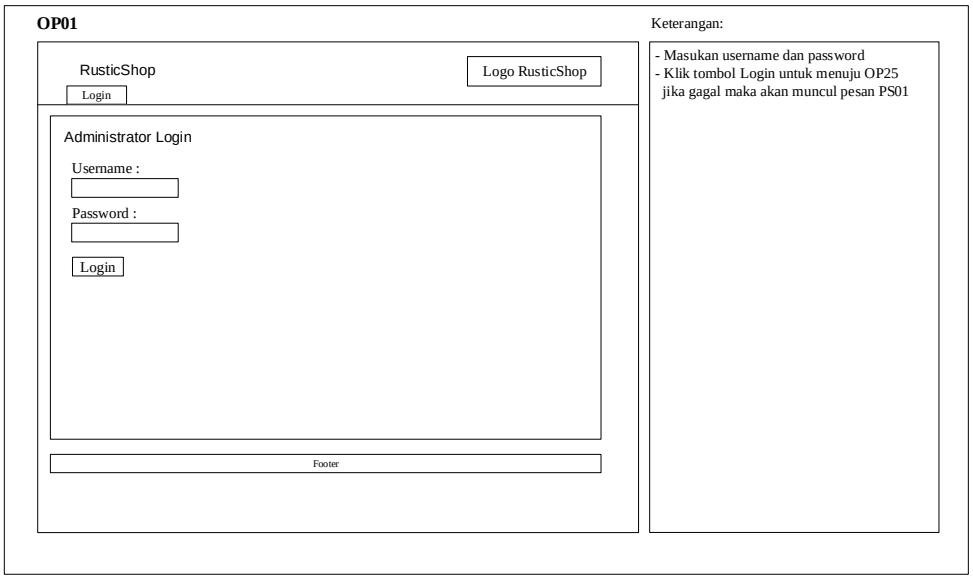
Perancangan tampilan admin - beranda (A24)




**Gambar 3.104 Perancangan tampilan admin – beranda(A24)**

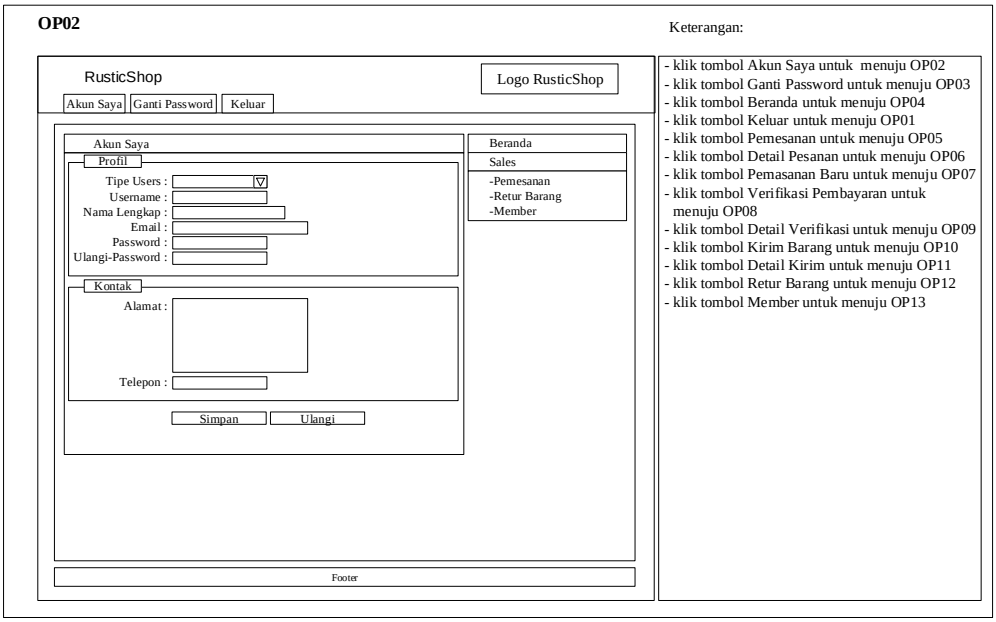
### 3.2.3.2. Perancangan Antarmuka Operator

Perancangan tampilan operator – login(OP01)



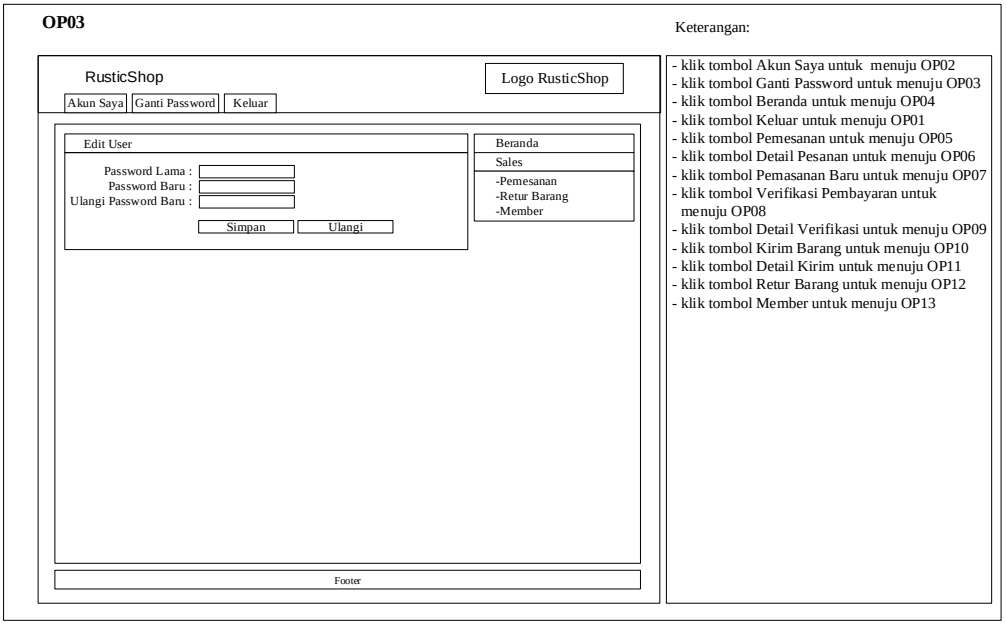

**Gambar 3.105 Perancangan tampilan operator – login(OP01)**

Perancangan tampilan operator – user (OP02)



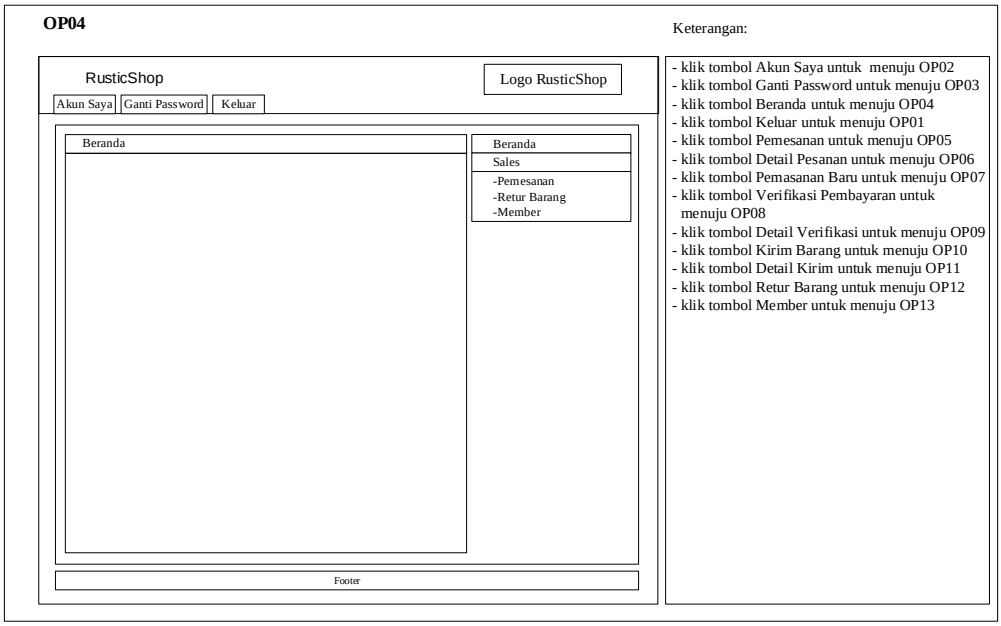

**Gambar 3.106 Perancangan tampilan operator – users(OP02)**

Perancangan tampilan operator – gantipassword(OP02)



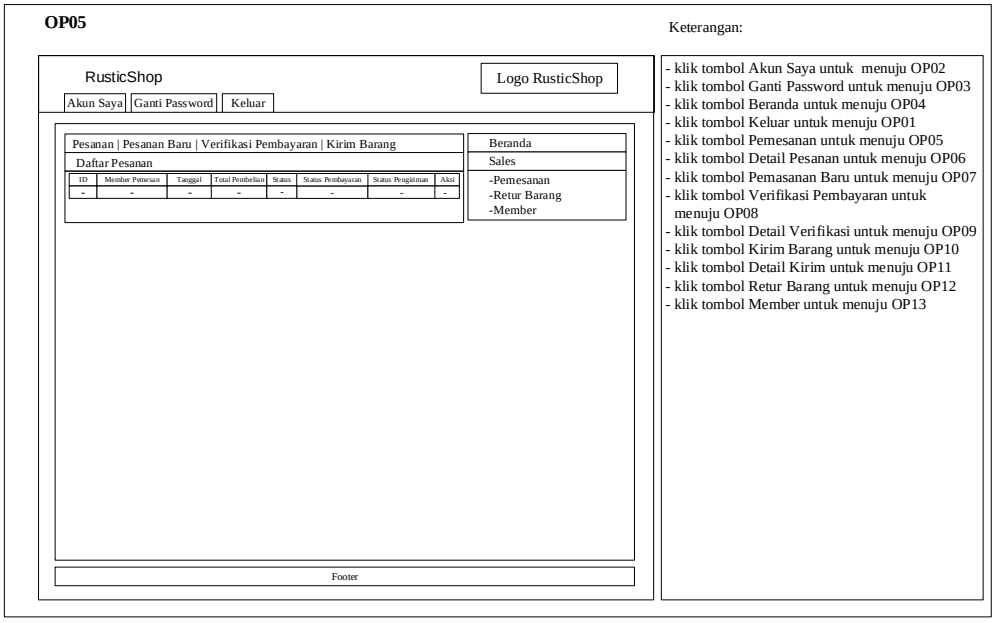

**Gambar 3.107 Perancangan tampilan operator – ganti password(OP03)**

Perancangan tampilan operator – beranda (OP02)



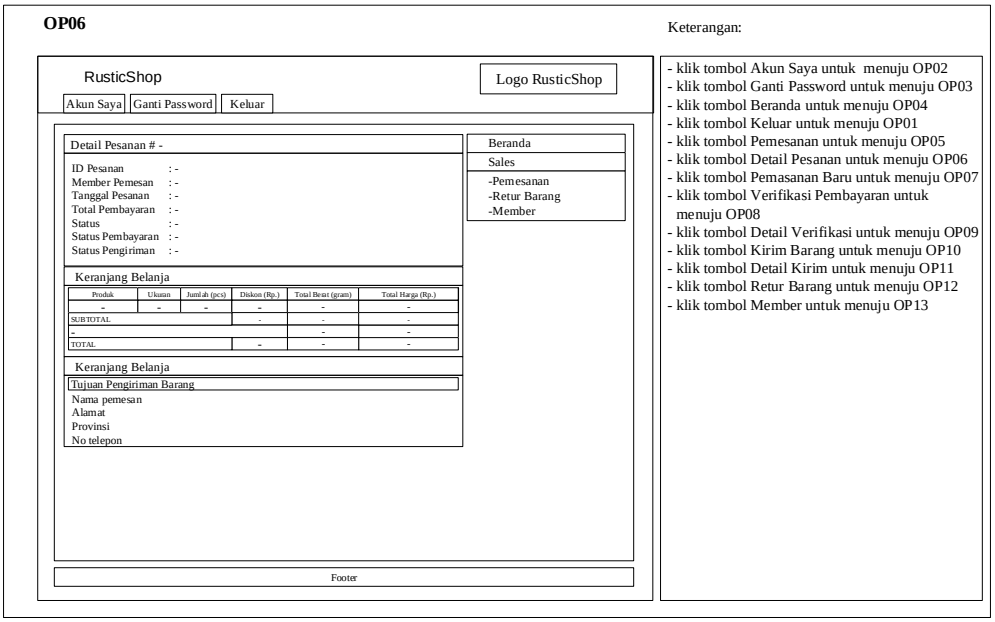

**Gambar 3.108 Perancangan tampilan operator – beranda(OP04)**

Perancangan tampilan operator – pemesanan(OP05)



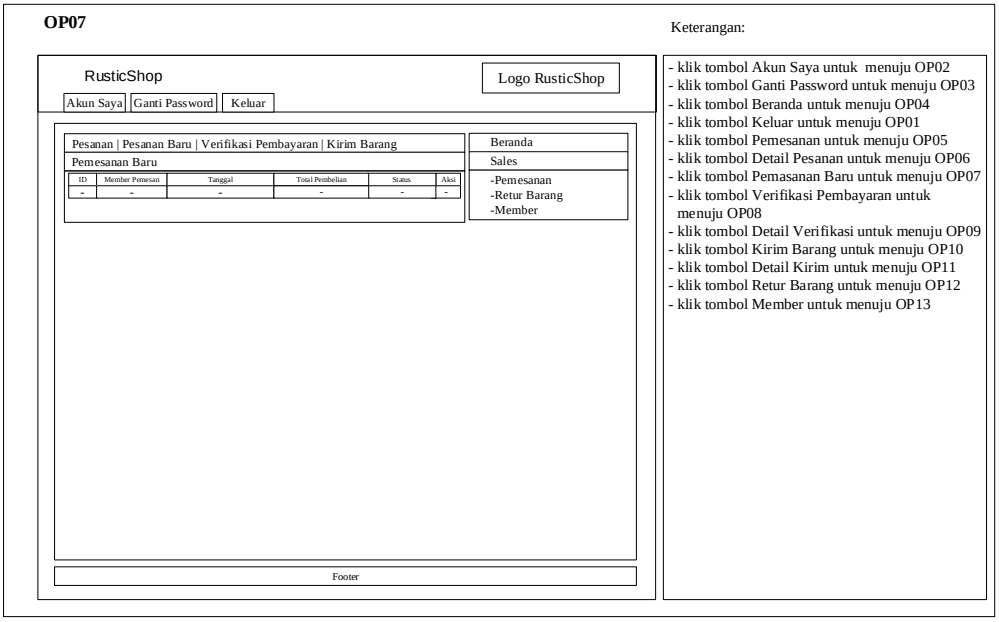

**Gambar 3.109 Perancangan tampilan operator – pemesanan(OP05)**

Perancangan tampilan operator – detailpesanan (OP06)



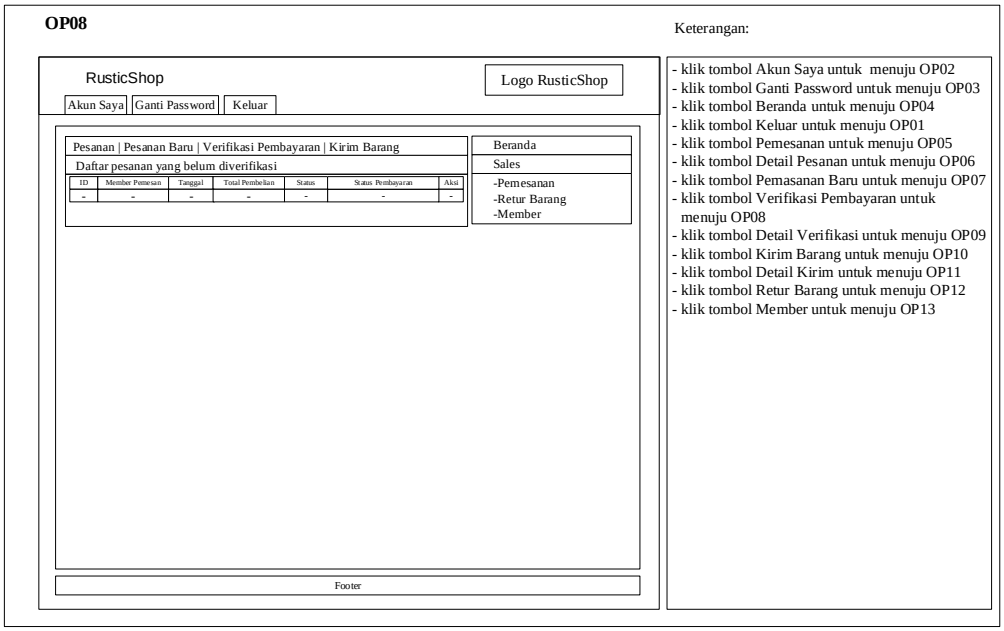

**Gambar 3.110 Perancangan tampilan operator – detailpesanan(OP06)**

Perancangan tampilan operator – pemesanan baru(OP07)



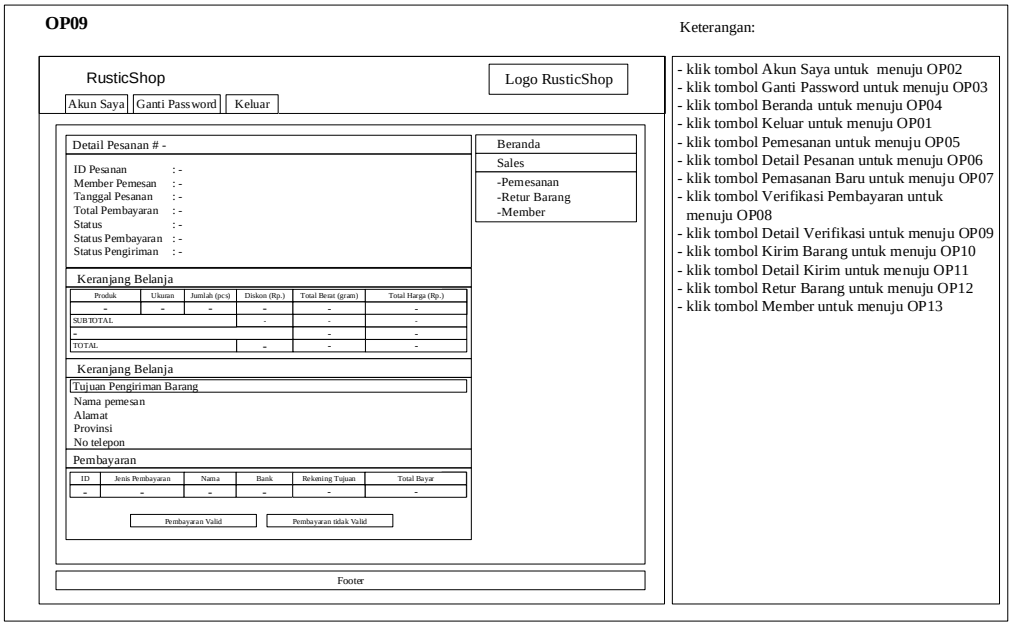

**Gambar 3.111 Perancangan tampilan operator – pemesanan baru(OP07)**

Perancangan tampilan operator – verifikasi pembayaran (OP08)



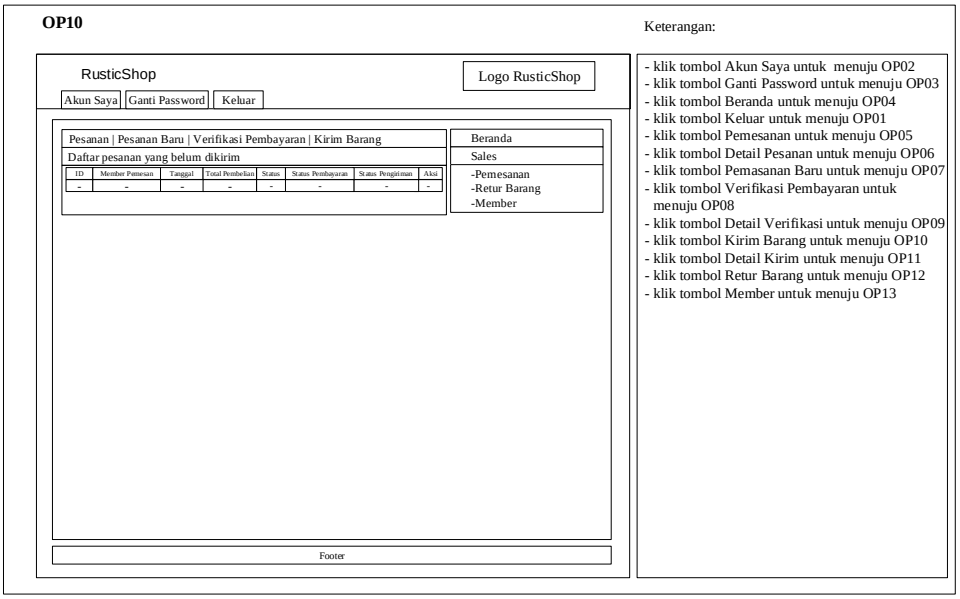

**Gambar 3.112 Perancangan tampilan operator – verifikasi pembayaran(OP08)**

Perancangan tampilan operator – detail verifikasi pembayaran(OP09)



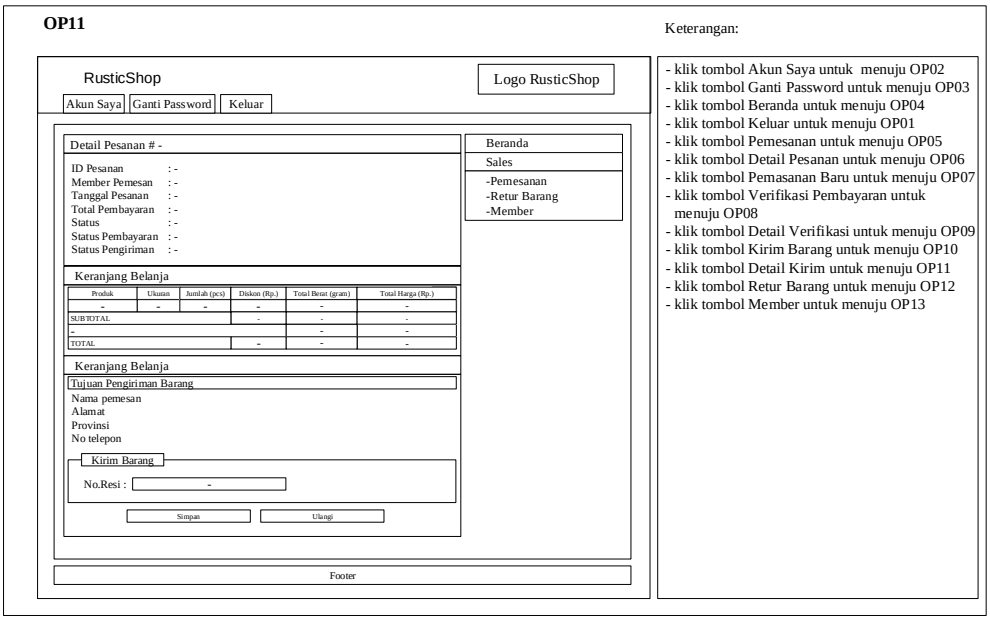

**Gambar 3.113 Perancangan tampilan operator – detail verifikasi pembayaran(OP09)**

Perancangan tampilan operator – kirim barang (OP10)



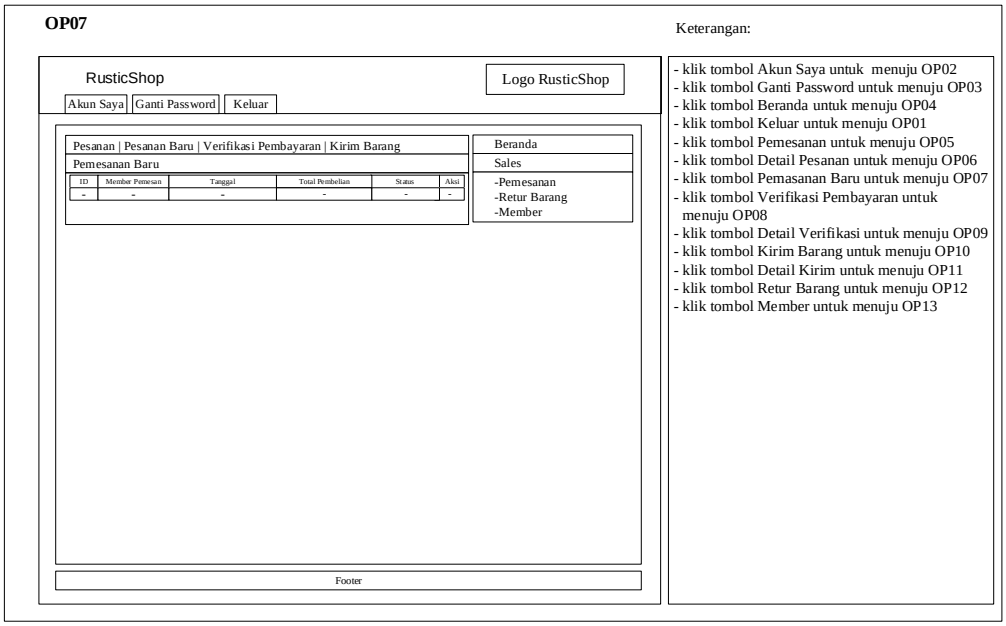

**Gambar 3.114 Perancangan tampilan operator – kirim barang(OP10)**

Perancangan tampilan operator – detail kirim barang(OP11)



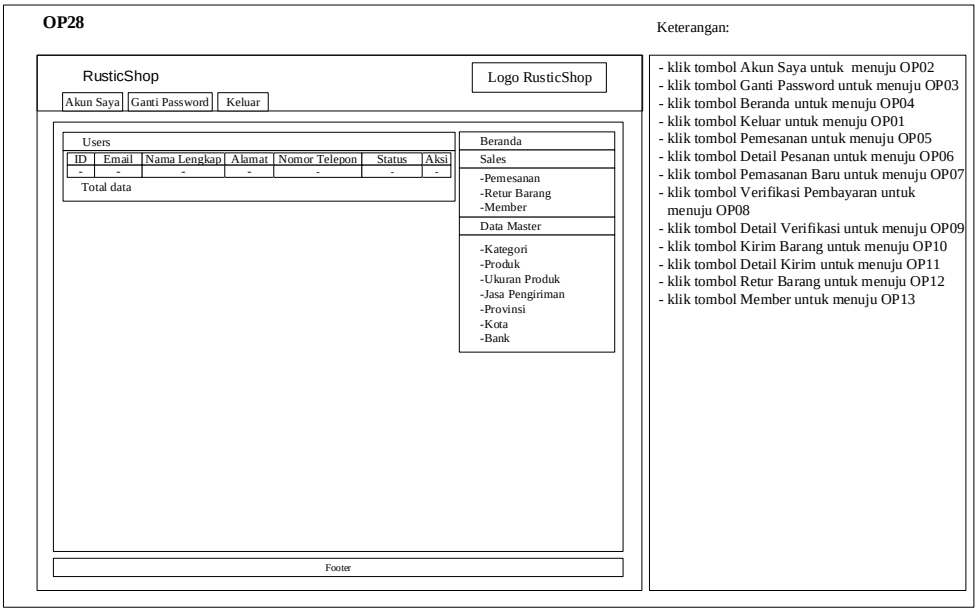

**Gambar 3.115 Perancangan tampilan operator – detail kirim barang(OP11)**

Perancangan tampilan operator – retur (OP12)




**Gambar 3.116 Perancangan tampilan operator – retur(OP12)**

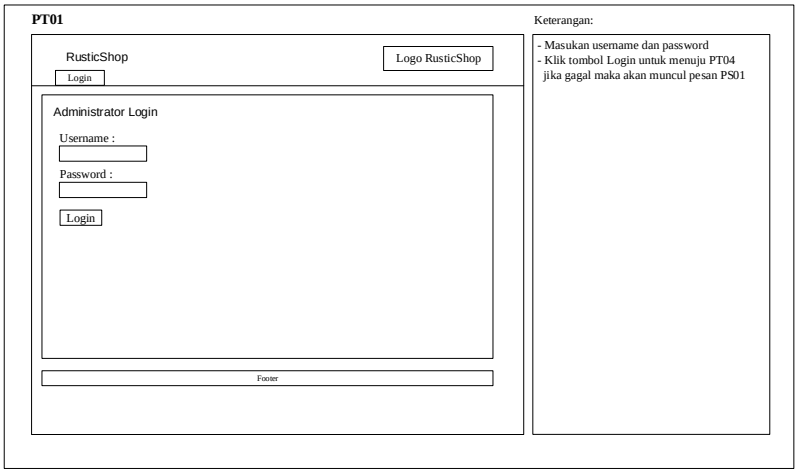
Perancangan tampilan operator – member (OP13)




**Gambar 3.117 Perancangan tampilan operator – member(OP13)**

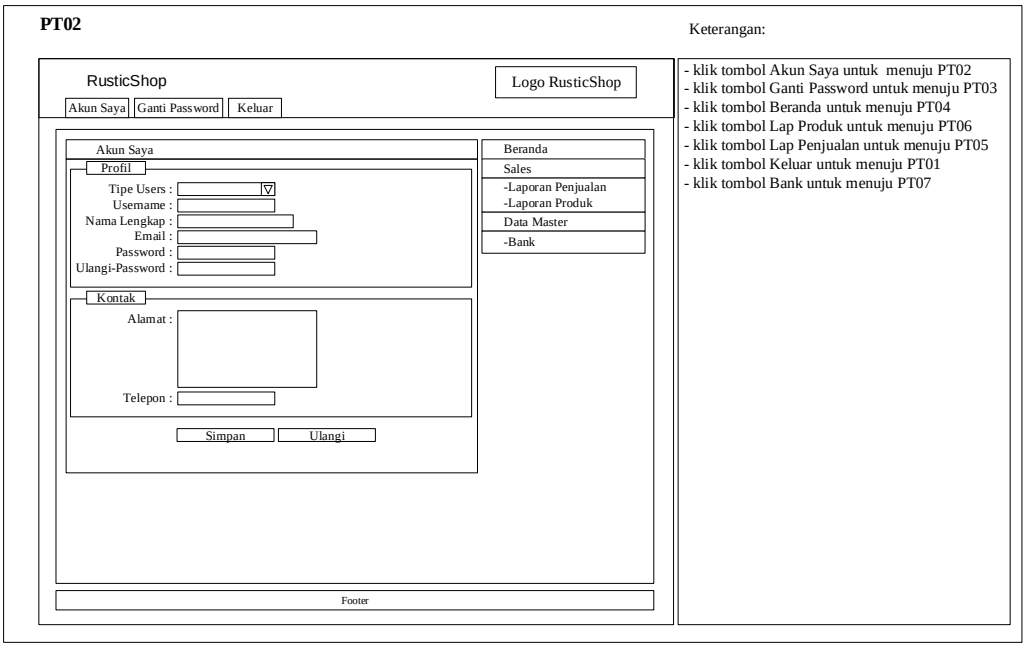
### 3.2.3.3. Perancangan Antarmuka Pemilik Toko

Perancangan tampilan pemilik toko – login(PT01)



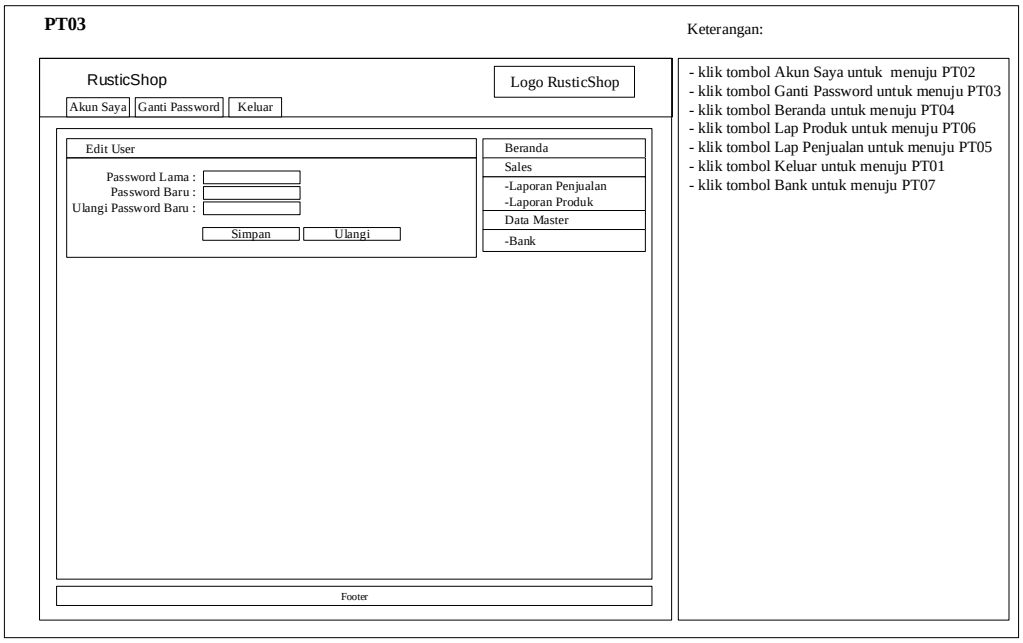

**Gambar 3.118 Perancangan tampilan pemilik toko – login(PT01)**

Perancangan tampilan pemilik toko – users (PT02)



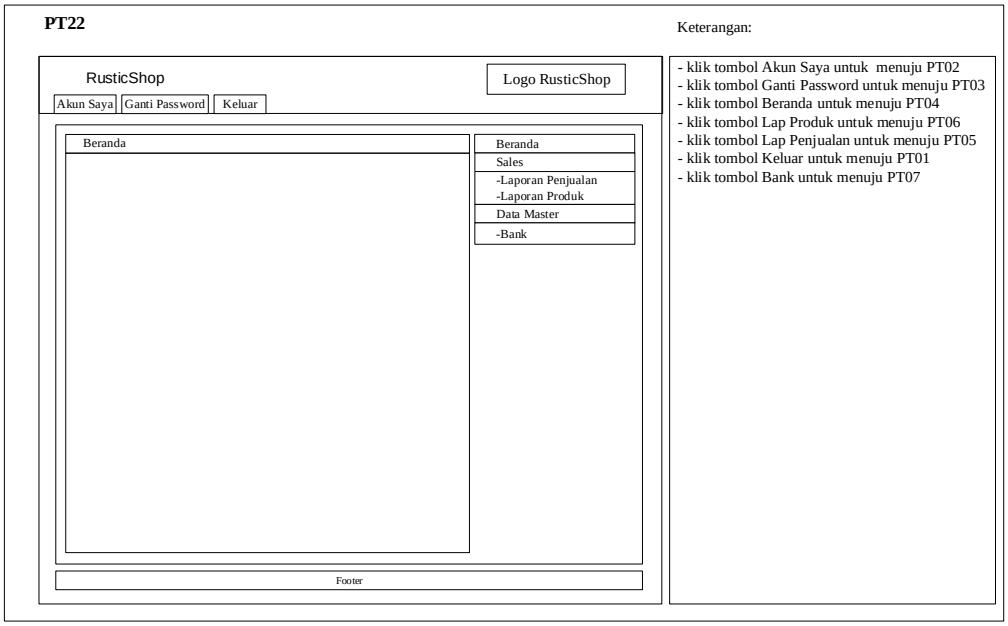

**Gambar 3.119 Perancangan tampilan pemilik toko – users(OP12)**

Perancangan tampilan pemilik toko – ganti password (PT03)



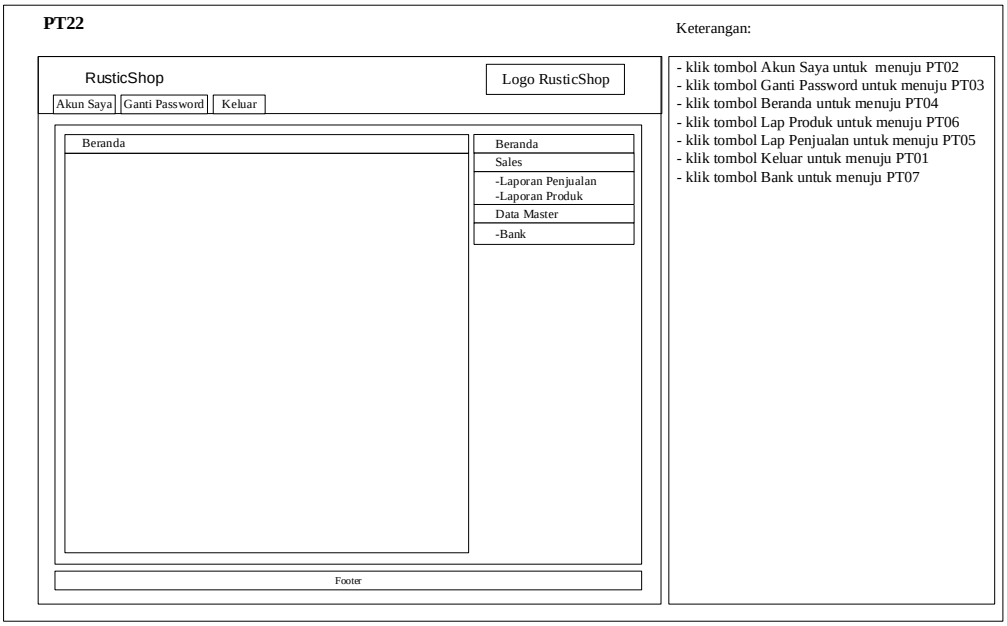

**Gambar 3.120 Perancangan tampilan pemilik toko– gantipassword(PT03)**

Perancangan tampilan pemilik toko – beranda (PT04)



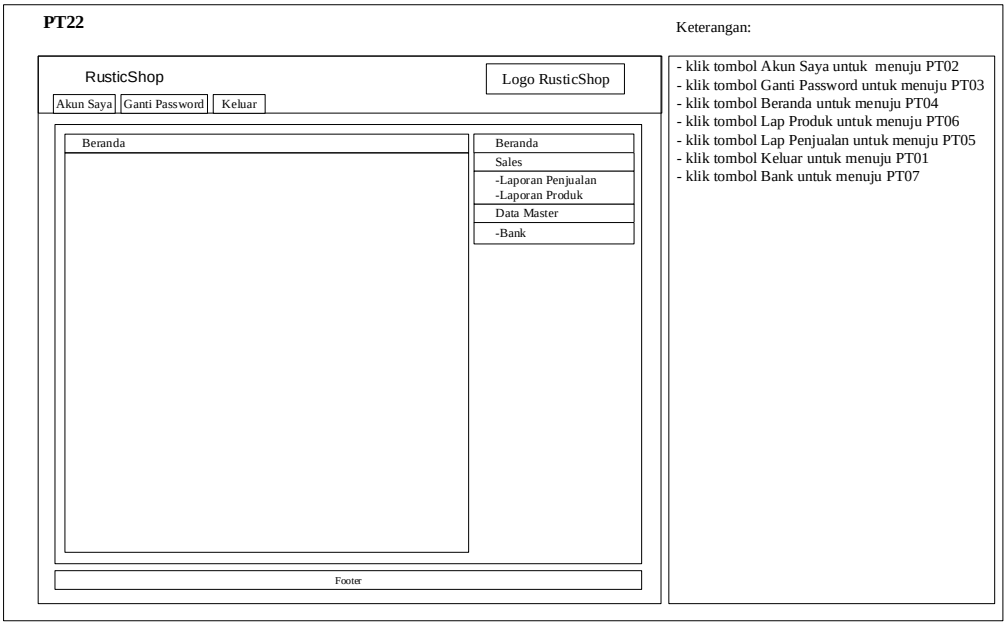

**Gambar 3.121 Perancangan tampilan pemilik toko – beranda(PT04)**

Perancangan tampilan pemilik toko – laporan penjualan(PT05)



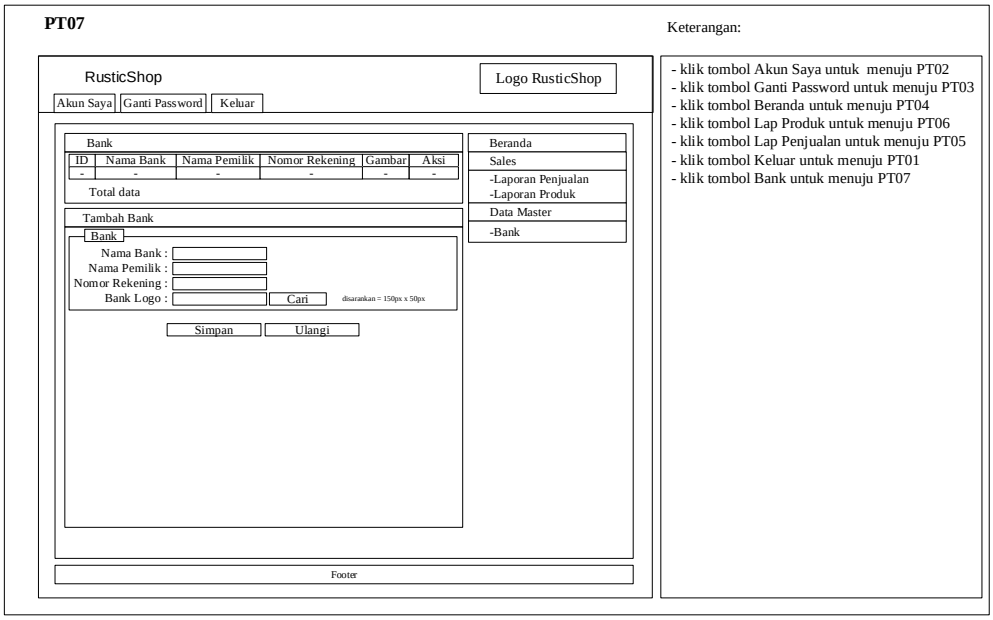

**Gambar 3.122 Perancangan tampilan pemilik toko– laporan penjualan(PT05)**

Perancangan tampilan pemilik toko – laporan produk (PT06)




**Gambar 3.123 Perancangan tampilan pemilik toko – laporan produk(PT06)**

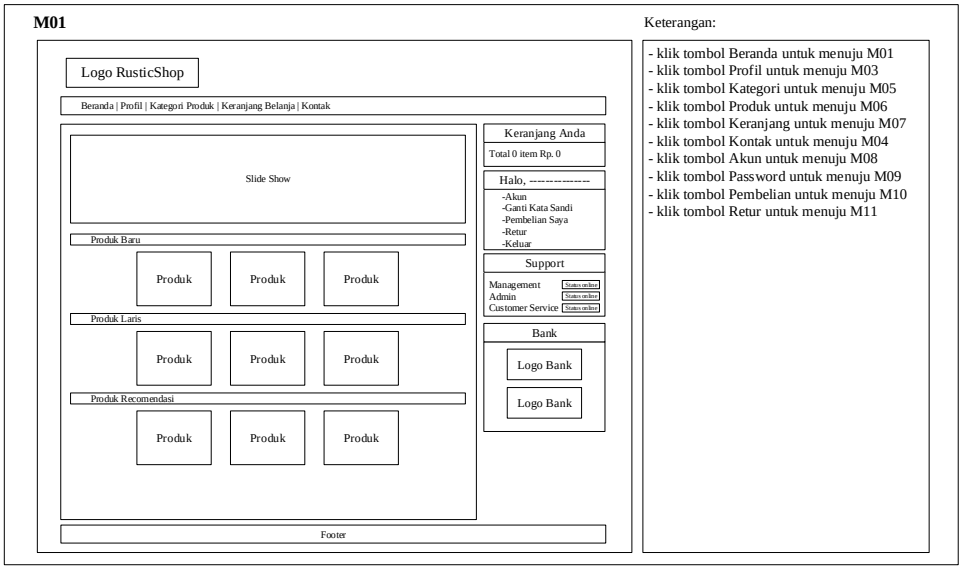
Perancangan tampilan pemilik toko – bank(PT07)




**Gambar 3.124 Perancangan tampilan pemilik toko– bank(PT07)**

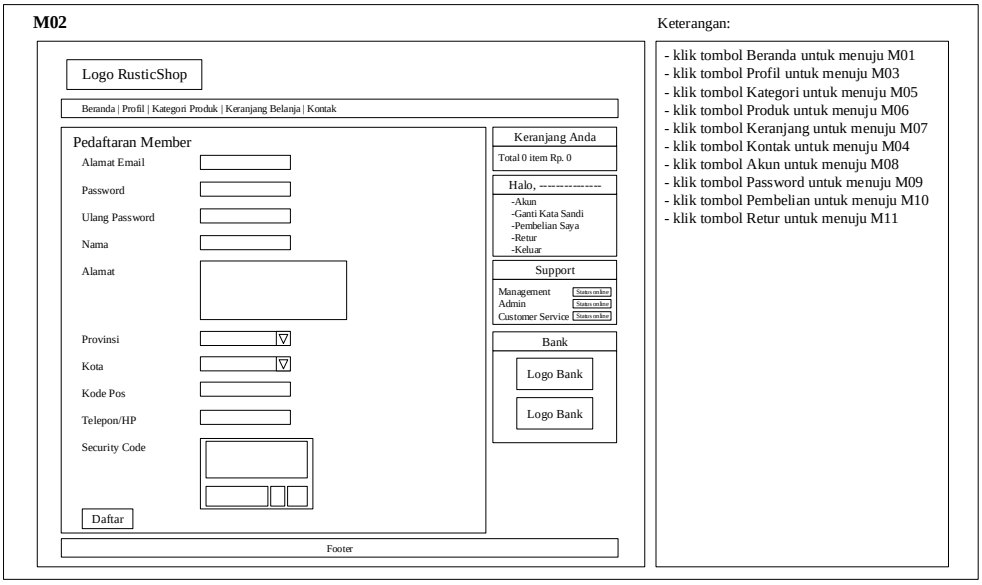
### 3.2.3.4. Perancangan Antarmuka Member

Perancangan tampilan member – beranda (M01)



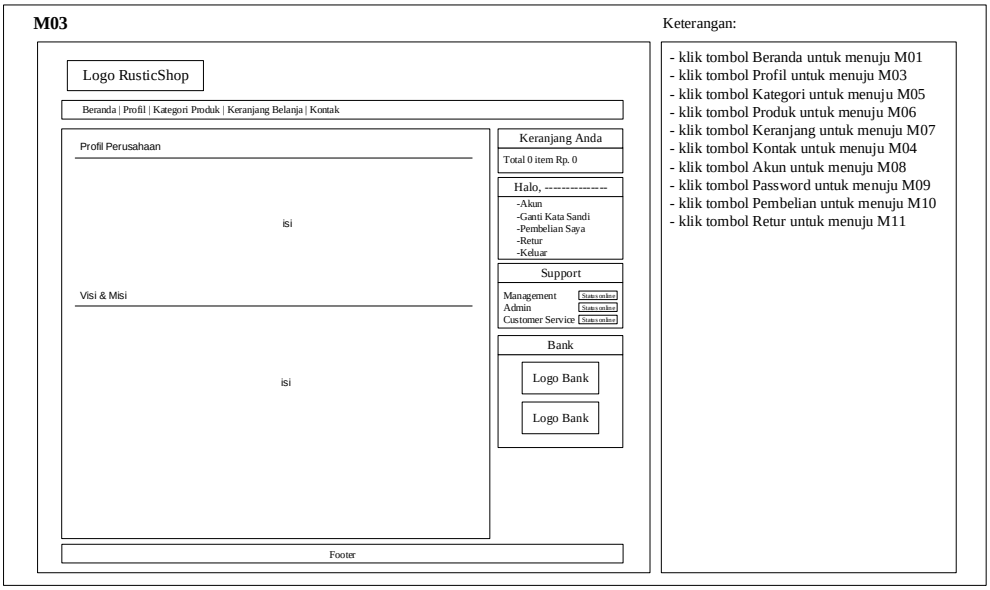

**Gambar 3.125 Perancangan tampilan member – beranda(M01)**

Perancangan tampilan member – daftar(M02)



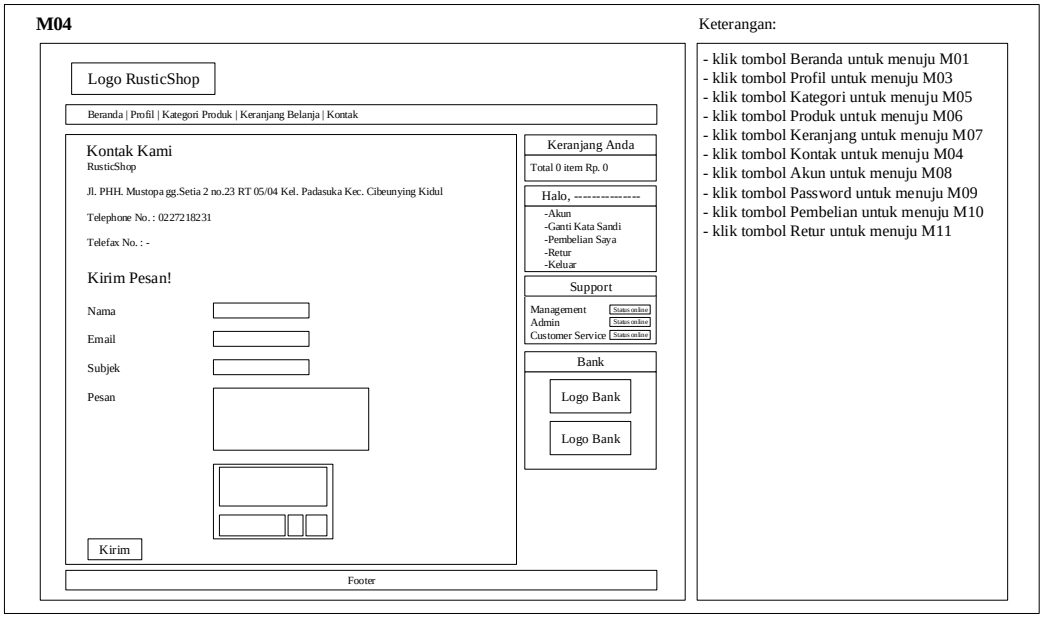

**Gambar 3.126 Perancangan tampilan member– daftar(M02)**

Perancangan tampilan member – profil perusahaan (M03)



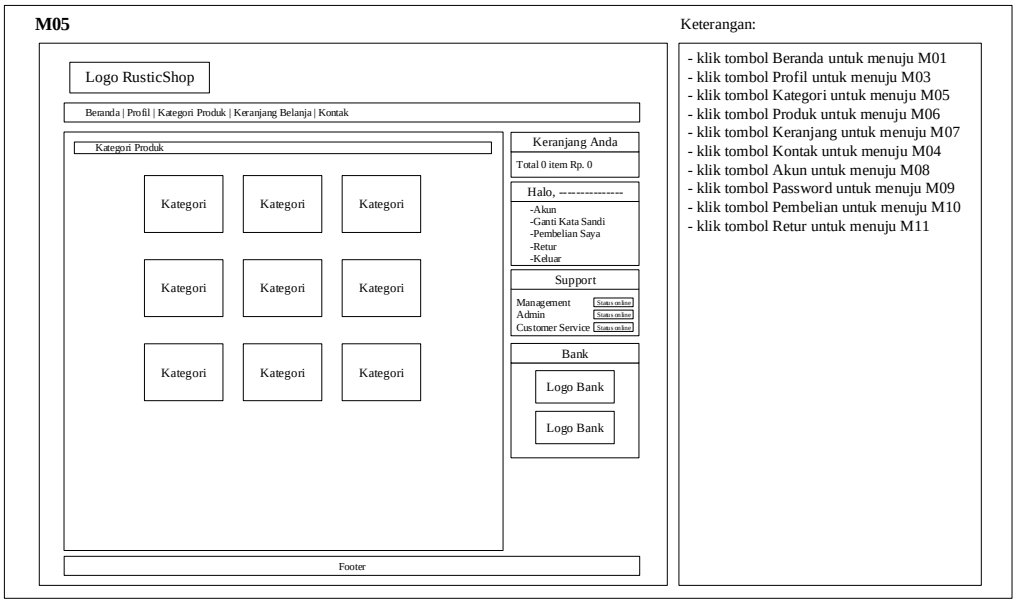

**Gambar 3.127 Perancangan tampilan member – profilperusahaan(M03)**

Perancangan tampilan member – kontak(M04)



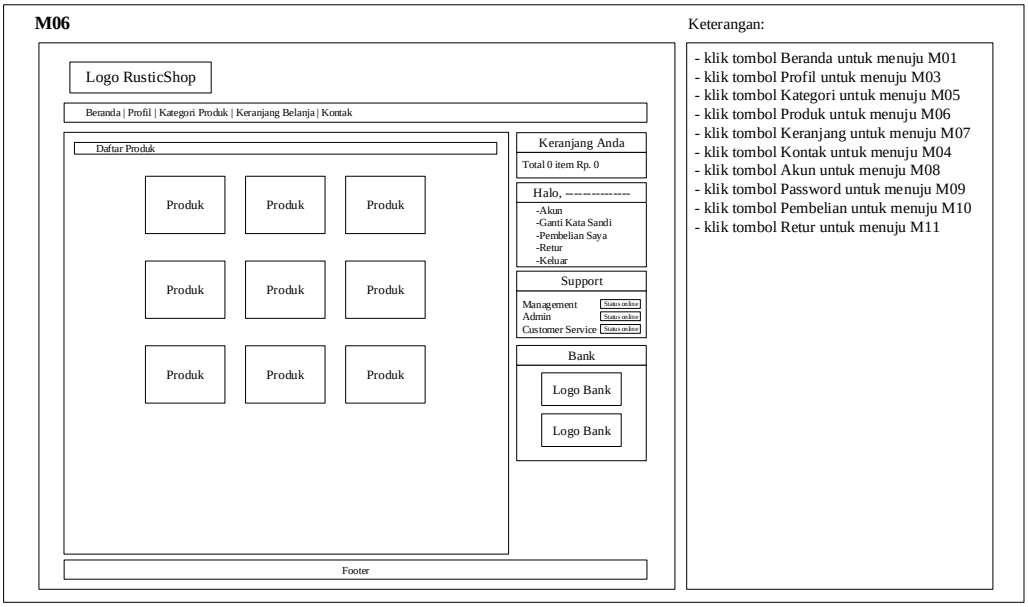

**Gambar 3.128 Perancangan tampilan member– kontak(M04)**

Perancangan tampilan member – kategori (M05)



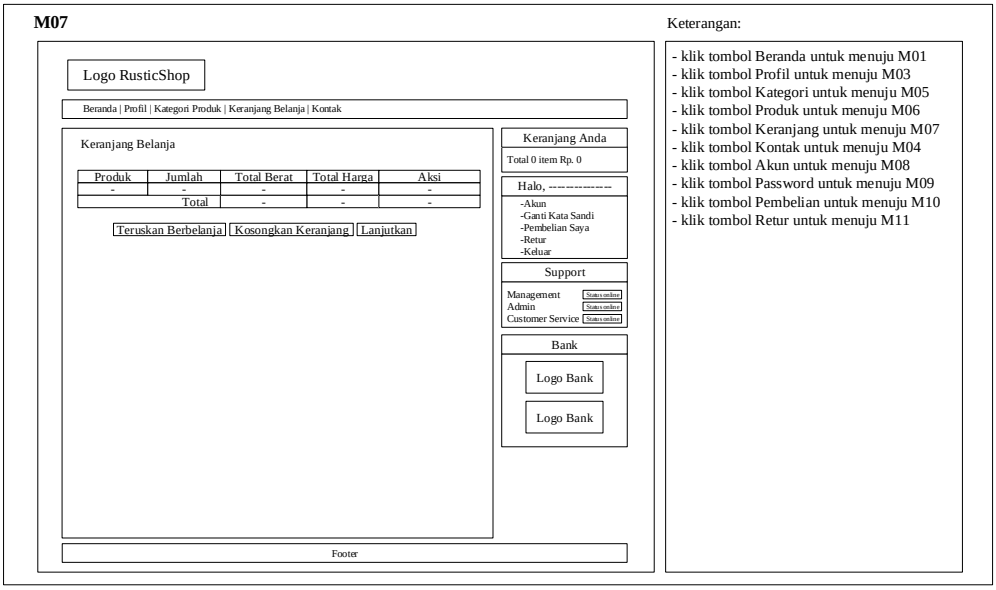

**Gambar 3.129 Perancangan tampilan member – kategori(M05)**

Perancangan tampilan member – produk(M06)



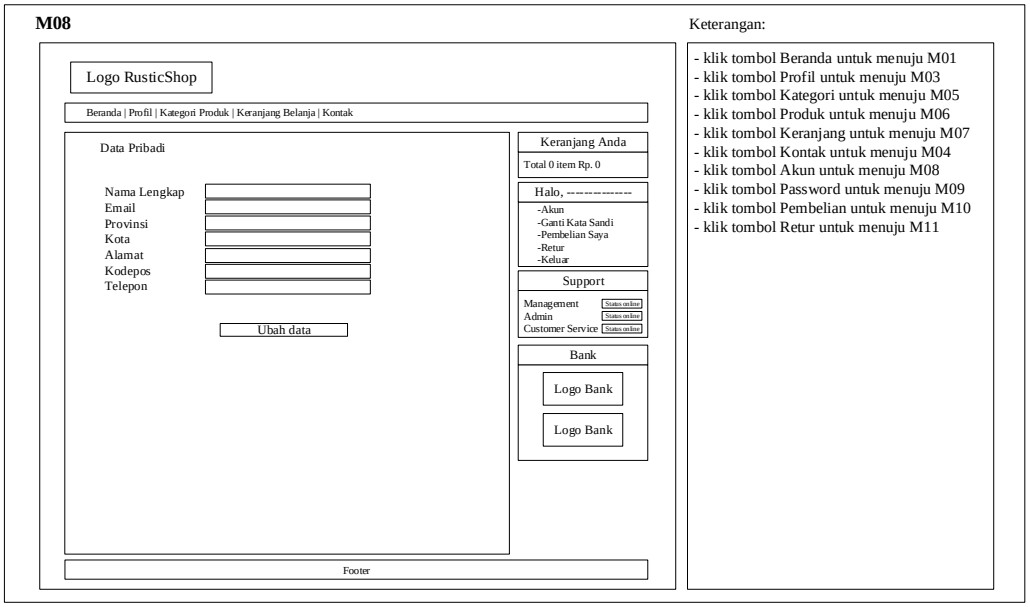

**Gambar 3.130 Perancangan tampilan member– produk(M06)**

Perancangan tampilan member – keranjang(M07)



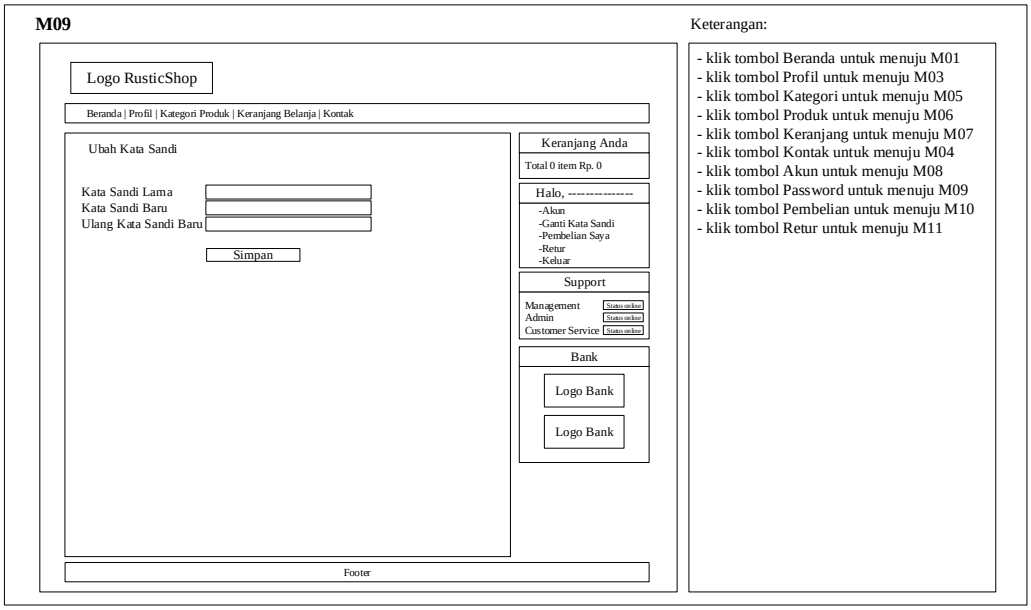

**Gambar 3.131 Perancangan tampilan member – keranjang(M07)**

Perancangan tampilan member – datapribadi(M08)



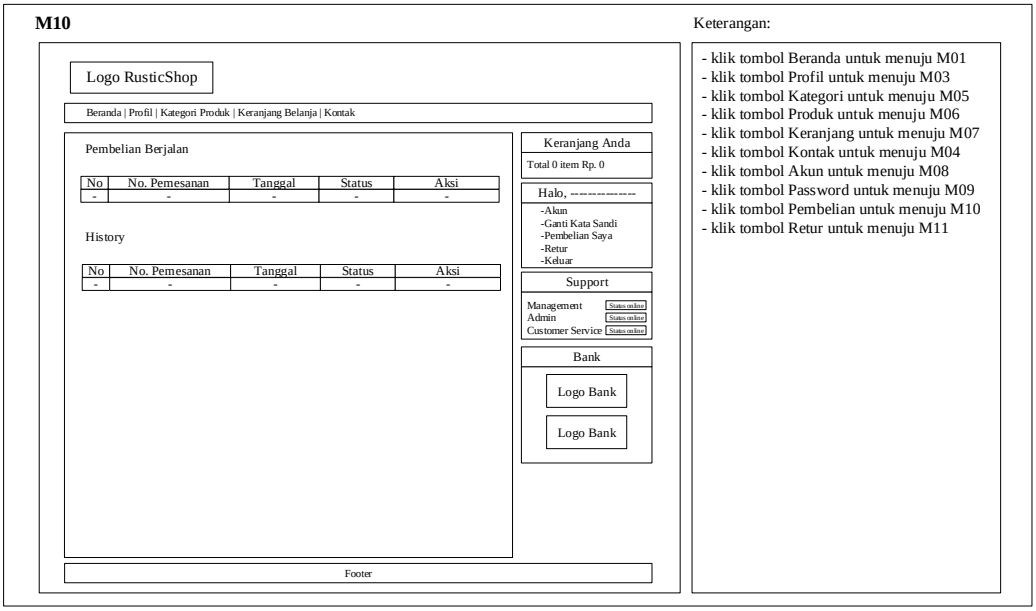

**Gambar 3.132 Perancangan tampilan member– datapribadi(M08)**

Perancangan tampilan member – gantipassword(M09)



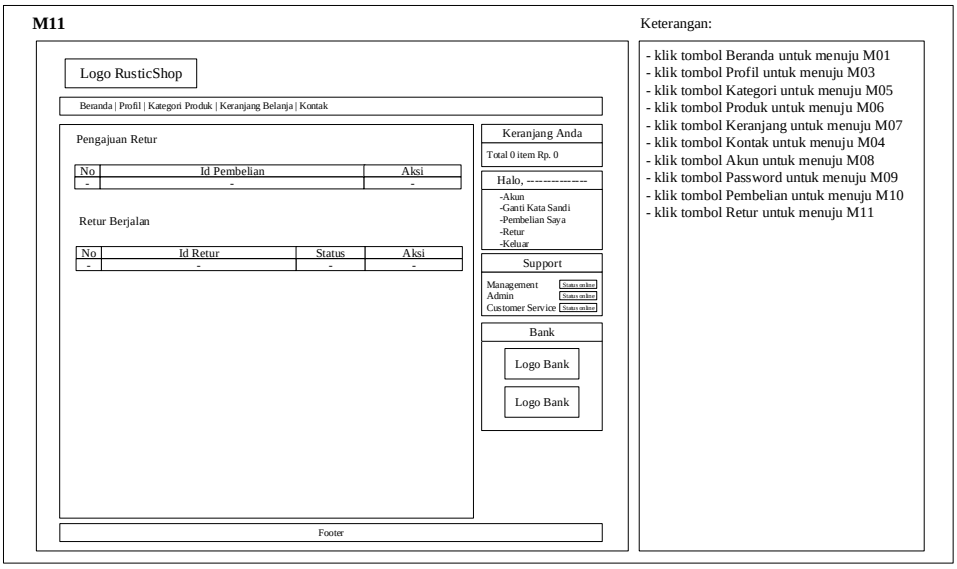

**Gambar 3.133 Perancangan tampilan member – ganti password(M09)**

Perancangan tampilan member – pembelian saya(M10)



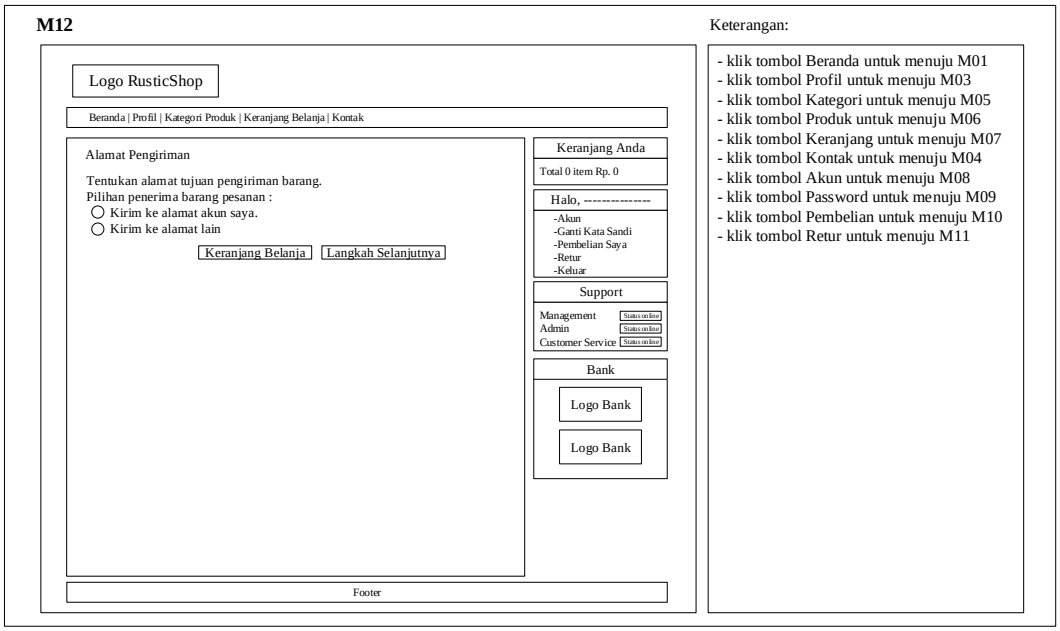

**Gambar 3.134 Perancangan tampilan member– pembelian saya(M10)**

Perancangan tampilan member – return(M11)



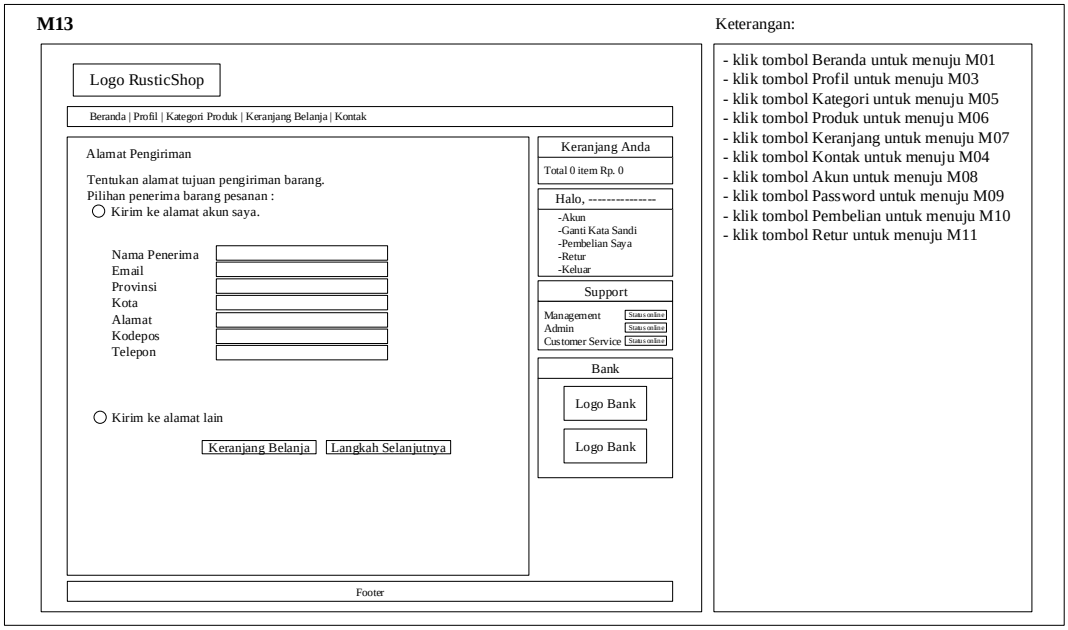

**Gambar 3.135 Perancangan tampilan member – retur(M11)**

Perancangan tampilan member – alamat kirim(M12)



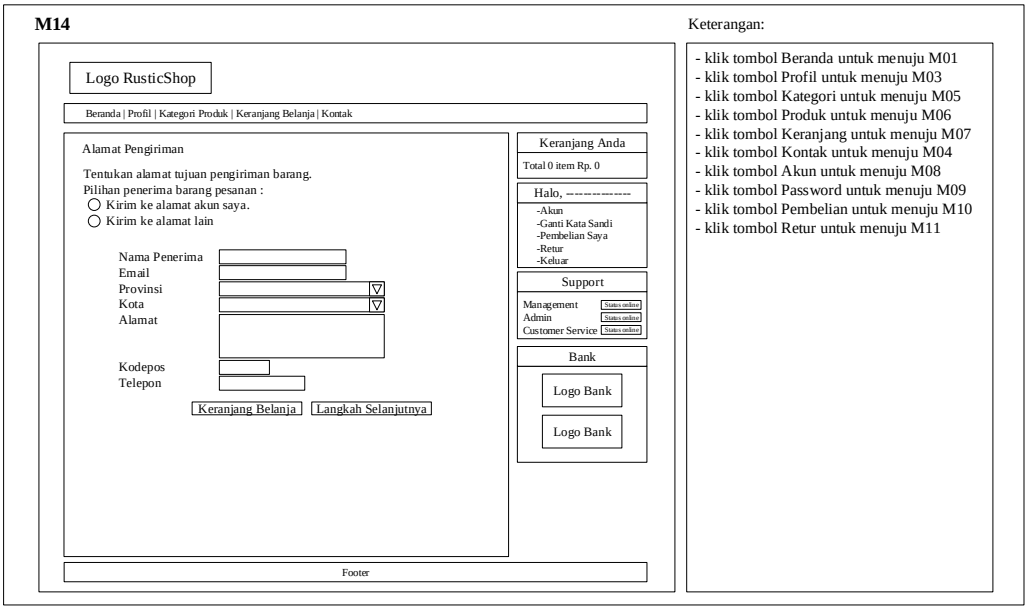

**Gambar 3.136 Perancangan tampilan member– alamatkirim(M12)**

Perancangan tampilan member – alamat akun(M13)



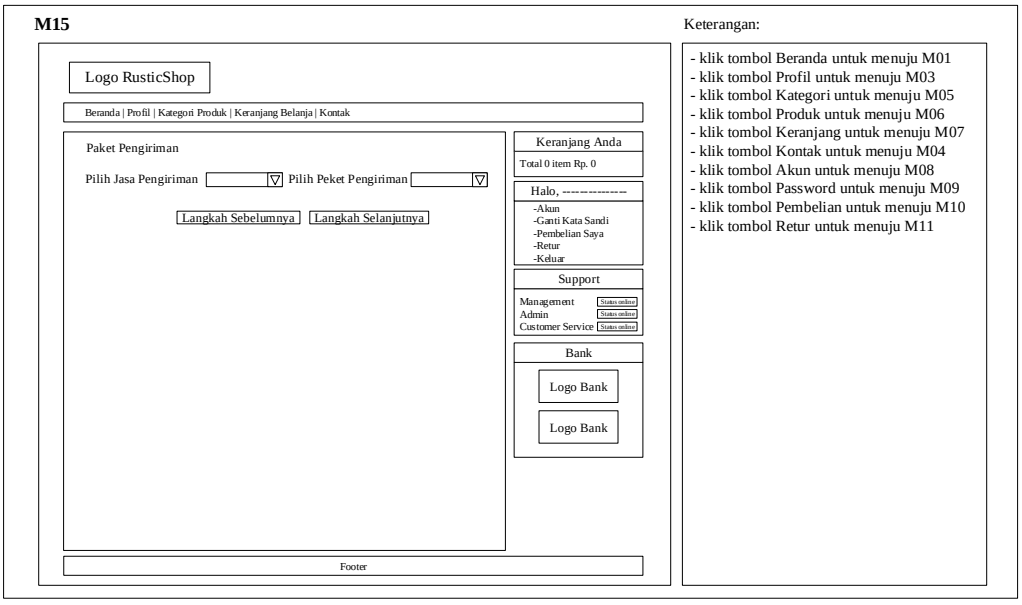

**Gambar 3.137 Perancangan tampilan member – alamat akun(M13)**

Perancangan tampilan member – alamat lain(M14)



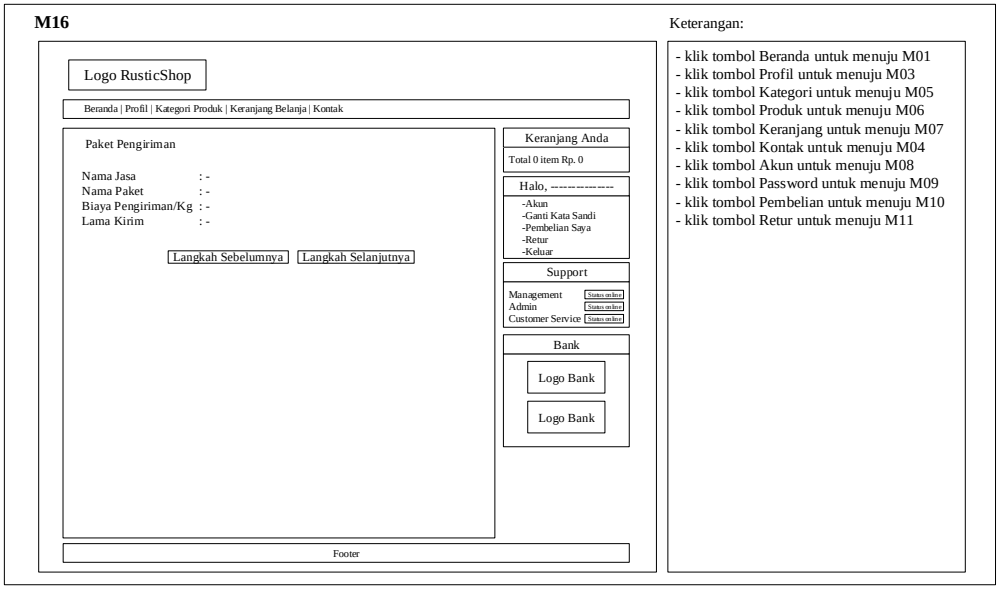

**Gambar 3.138 Perancangan tampilan member– alamatlain(M14)**

Perancangan tampilan member – paket kirim(M15)



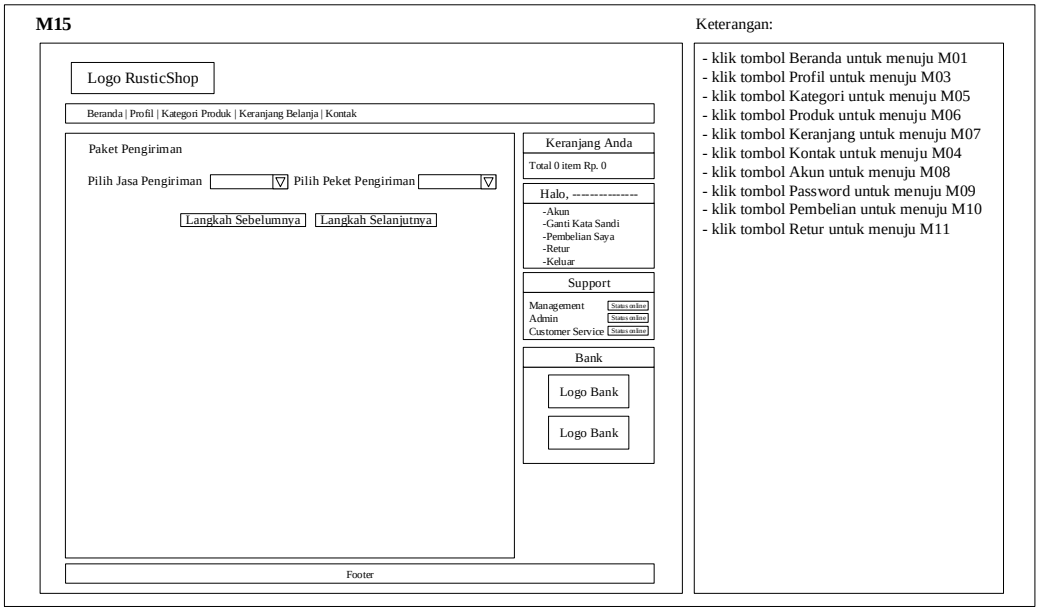

**Gambar 3.139 Perancangan tampilan member – paket kirim(M15)**

Perancangan tampilan member – detail paket(M16)



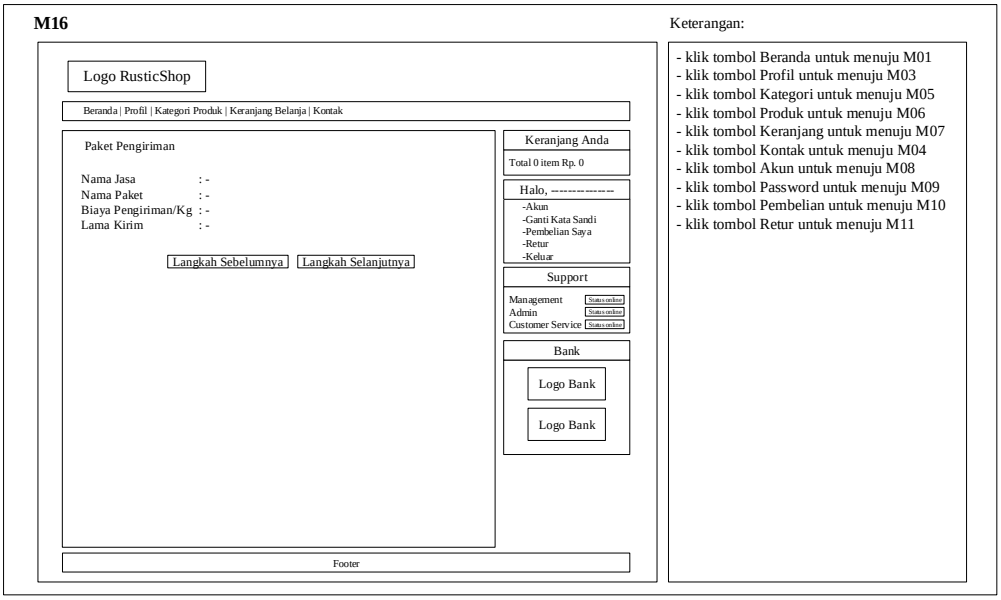

**Gambar 3.140 Perancangan tampilan member– detail paket(M16)**

Perancangan tampilan member – paket kirim(M15)



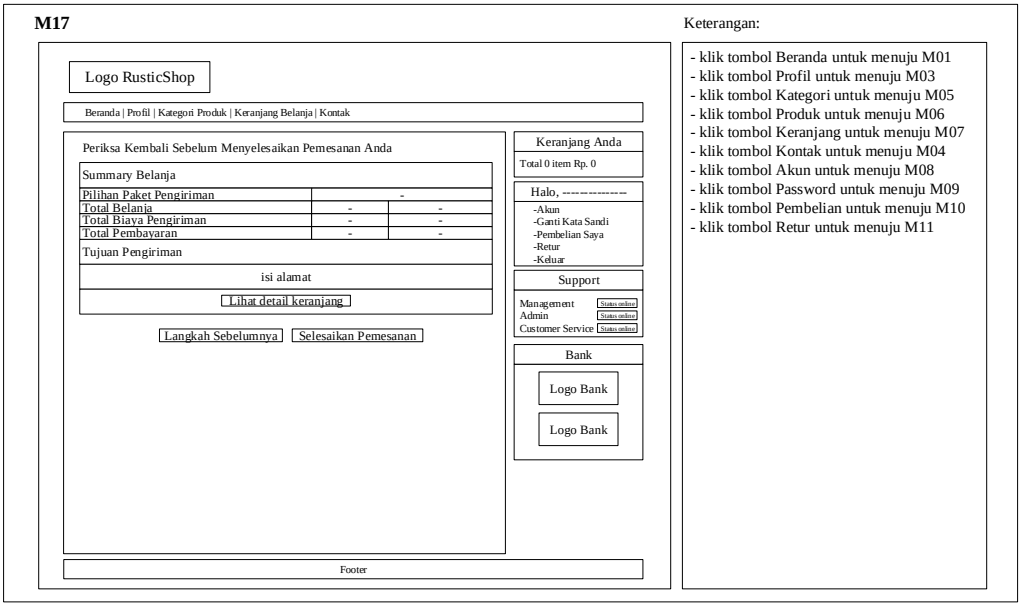

**Gambar 3.141 Perancangan tampilan member – paket kirim(M15)**

Perancangan tampilan member – detail paket(M16)



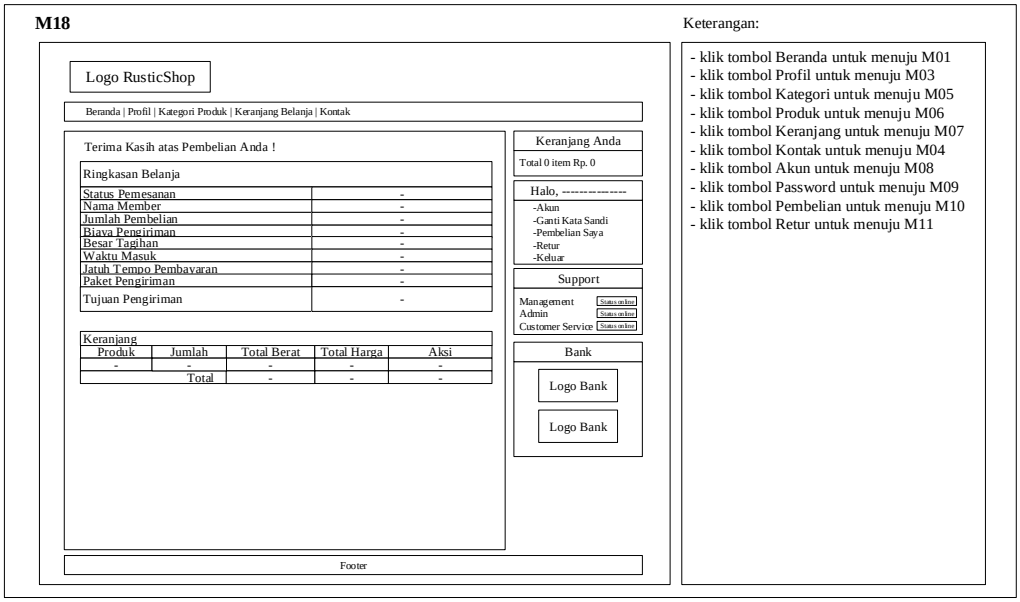

**Gambar 3.142 Perancangan tampilan member– detail paket(M16)**

Perancangan tampilan member – summary belanja(M17)




**Gambar 3.143 Perancangan tampilan member – summary belanja(M17)**

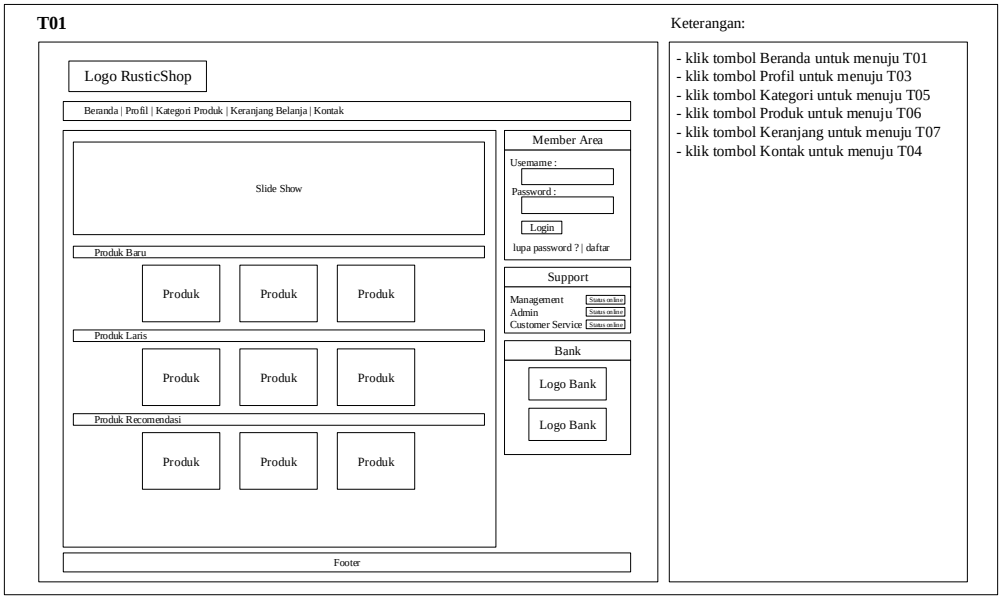
Perancangan tampilan member – selesai belanja(M18)




**Gambar 3.144 Perancangan tampilan member– selesai belanja(M18)**

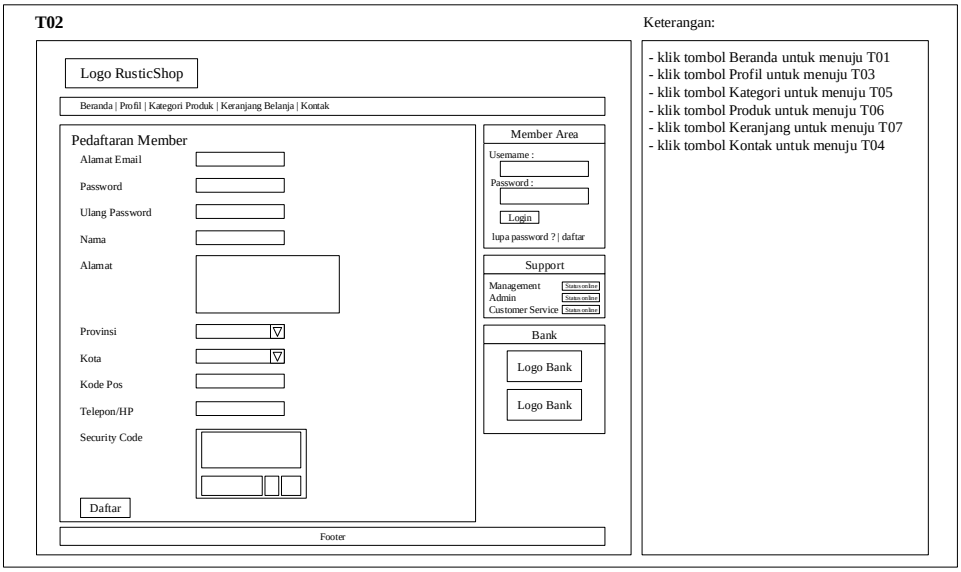
### 3.2.3.5. Perancangan Antarmuka Pengunjung

Perancangan tampilan pengunjung – beranda(T01)



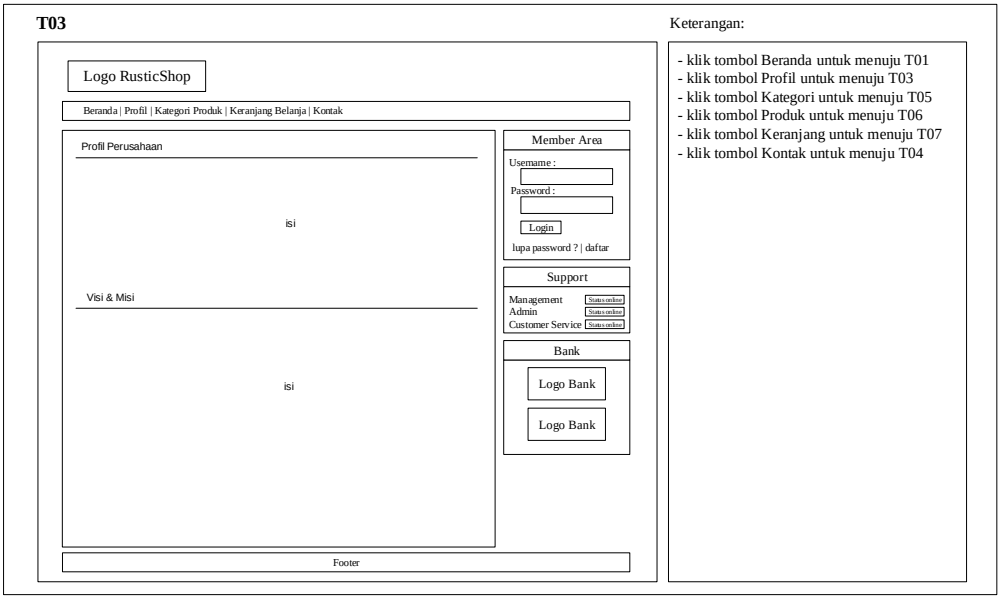

**Gambar 3.145 Perancangan tampilan pengunjung – summary belanja(T01)**

Perancangan tampilan pengunjung – daftar(T02)



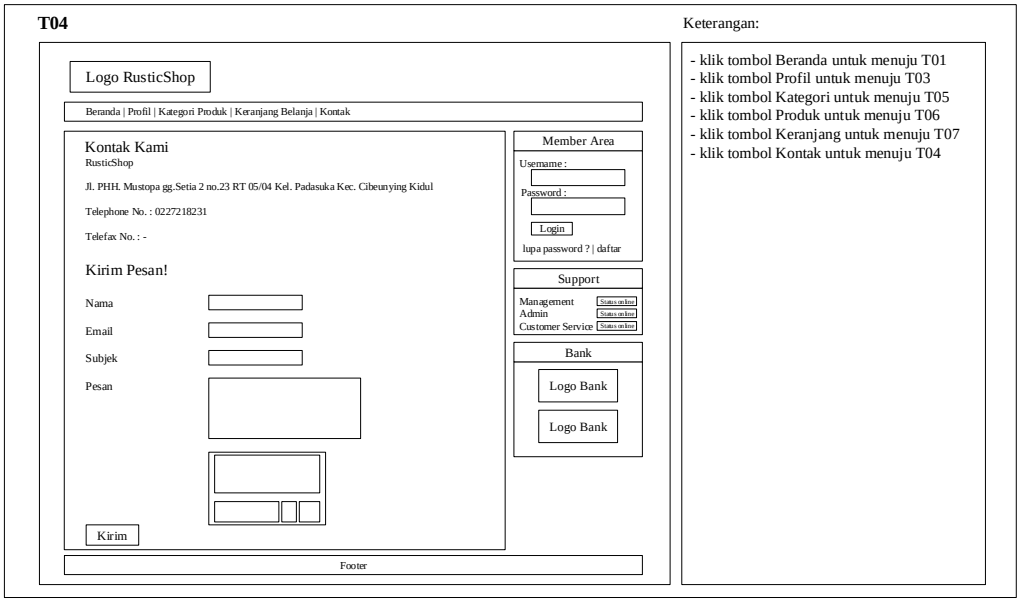

**Gambar 3.146 Perancangan tampilan pengunjung – selesai belanja(T02)**

Perancangan tampilan pengunjung – profil perusahaan(T03)



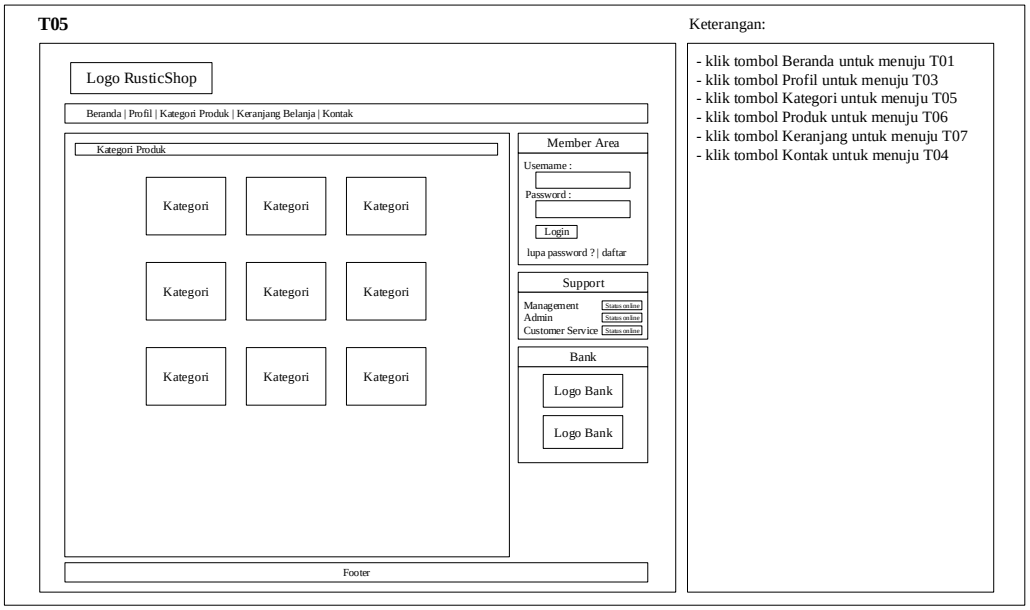

**Gambar 3.147 Perancangan tampilan pengunjung – profil perusahaan(T03)**

Perancangan tampilan pengunjung – kontak(T04)



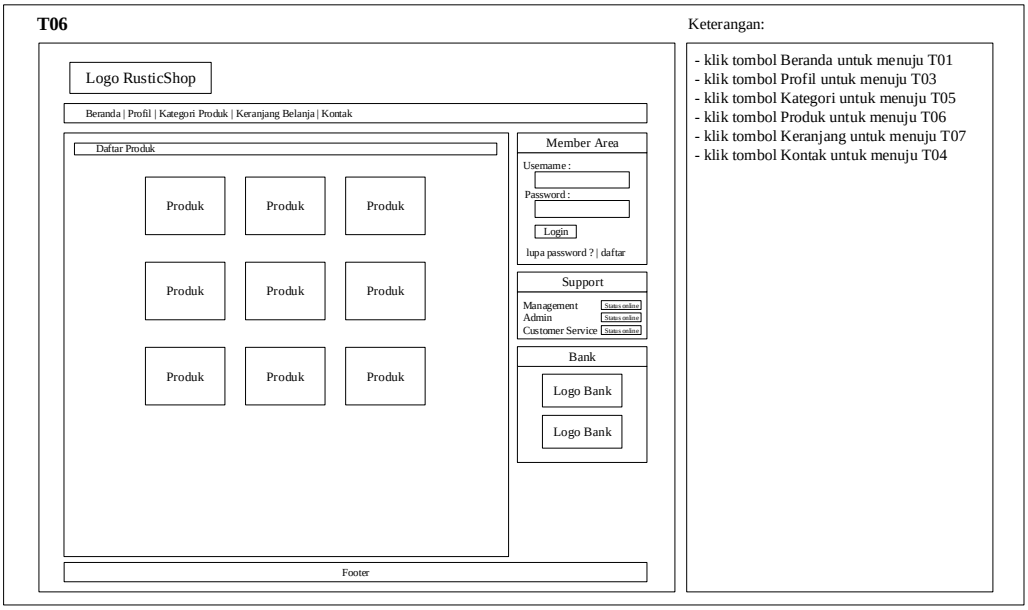

**Gambar 3.148 Perancangan tampilan pengunjung – kontak(T04)**

Perancangan tampilan pengunjung – kategori(T05)



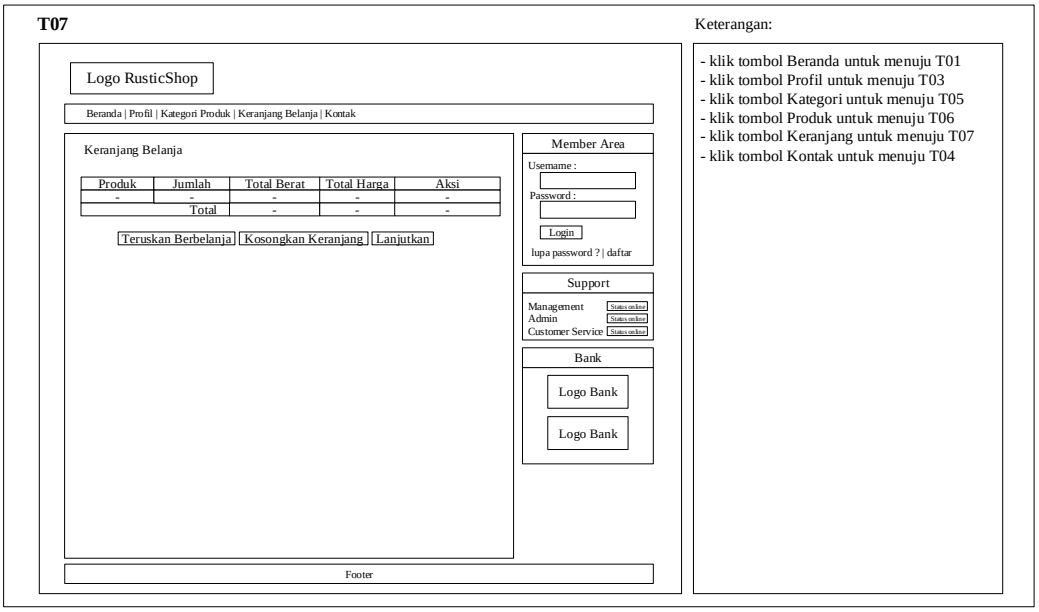

**Gambar 3.149 Perancangan tampilan pengunjung – kategori(T05)**

Perancangan tampilan pengunjung – produk(T06)



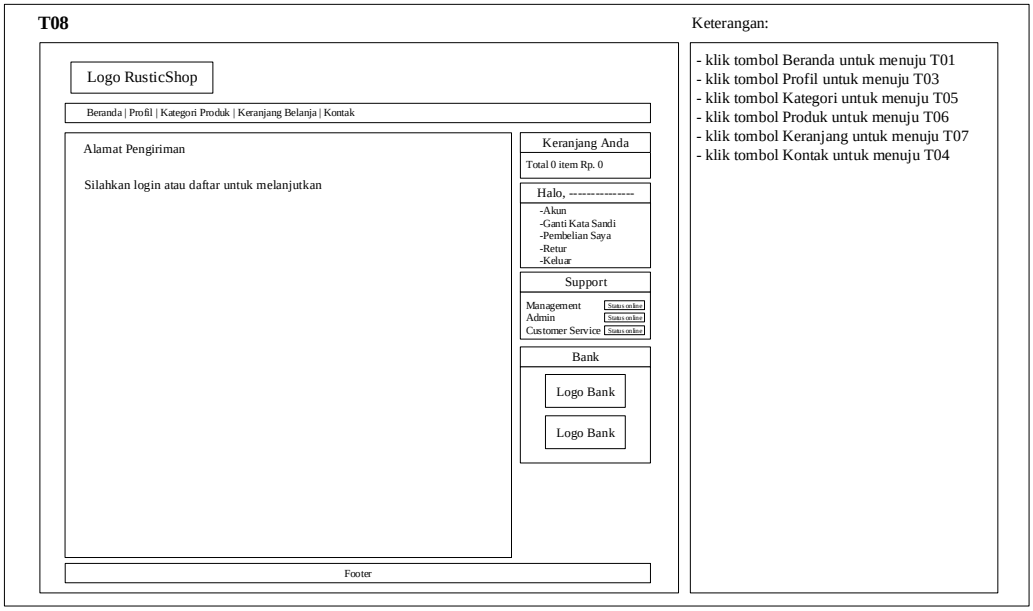

**Gambar 3.150 Perancangan tampilan pengunjung – produk(T06)**

Perancangan tampilan pengunjung – keranjang(T07)




**Gambar 3.151 Perancangan tampilan pengunjung – keranjang(T07)**

Perancangan tampilan pengunjung – alamat kirim(T08)




**Gambar 3.152 Perancangan tampilan pengunjung – alamat kirim(T08)**

### 3.2.4. Perancangan Tampilan Pesan

Perancangan pesan yang terdapat pada aplikasi *E-commerce* di Toko RusticShop dapat dilihat pada tabel 3.64

**Tabel 3.64 Tabel Perancangan Pesan**

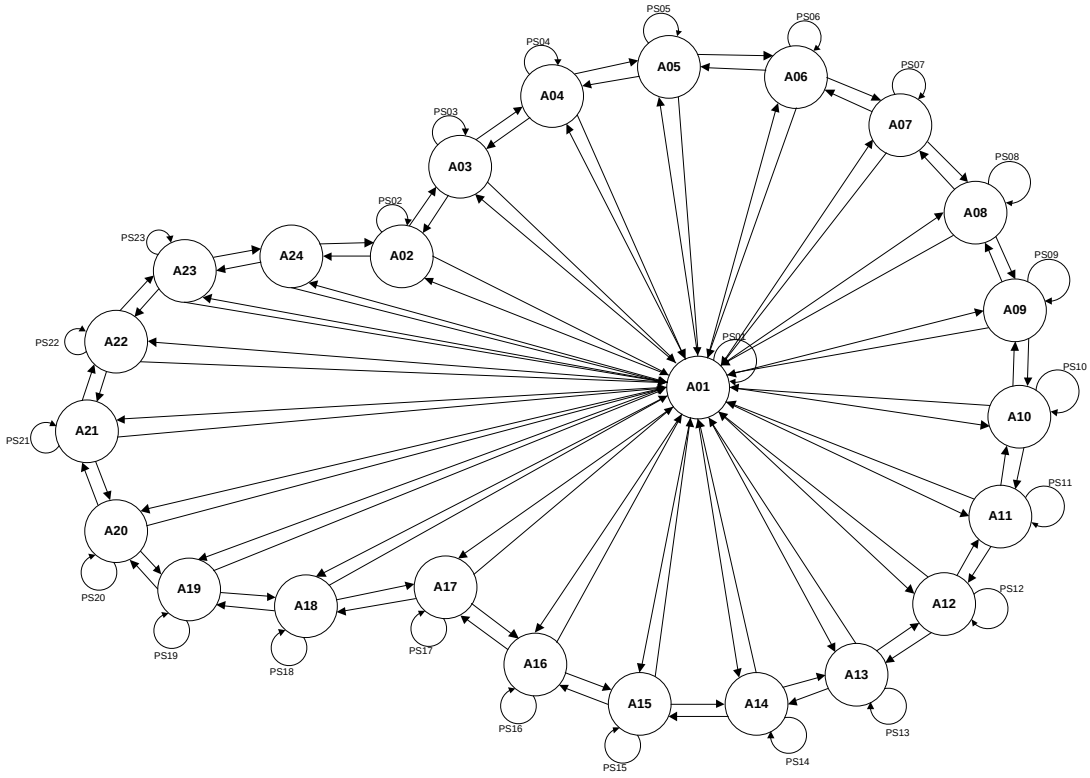
| No Pesan | Isi Pesan | Jenis Pesan | Terdapat Pada halaman |
|---|---|---|---|
| PS01 | Email Dibutuhkan | Peringatan | Pendaftaran,login Member, Akun Member |
| PS02 | Password Dibutuhkan | Peringatan | Pendaftaran,Login Member,Login Administrator,Login Operator, Ganti Password |
| PS03 | Password Ulang dibutuhkan | Peringatan | Ganti Password, Pendaftaran |
| PS04 | Nama Depan Dibutuhkan | Peringatan | Pendaftaran |
| PS05 | Alamat dibutuhkan | Peringatan | Pendaftaran, ubah profil |

| | | | |
|---|---|---|---|
| PS06 | Email harus berisi email yang benar | Peringatan | Tambah Administrator |
| PS07 | Nama belakang dibutuhkan | Peringatan | Pendaftaran |
| PS08 | Kode pos dibutuhkan | Peringatan | Pendaftaran |
| PS09 | Password tidak cocok dengan password ulang | Peringatan | Pendaftaran, ganti Password |
| PS10 | Nama pemilik rekening dibutuhkan | Peringatan | Konfirmasi pembayaran |
| PS11 | Nomor pemilik rekening dibutuhkan | Peringatan | Konfirmasi pembayaran |
| PS12 | Nama kategori dibutuhkan | Peringatan | Ubah kategori, tambah kategori |
| PS13 | Nama produk dibutuhkan | Peringatan | Tambah produk, ubah produk |
| PS14 | Harga dibutuhkan | Peringatan | Tambah produk, ubah produk |
| PS15 | Berat dibutuhkan | Peringatan | Tambah produk, ubah produk |
| PS16 | Stok dibutuhkan | Peringatan | Tambah produk, ubah produk |
| PS17 | Kode pengiriman dibutuhkan | Peringatan | Update pesanan |
| PS18 | Provinsi dibutuhkan | Peringatan | Pendaftaran, tambah provinsi, ubah povinsi |
| PS19 | Kota dibutuhkan | Peringatan | Pendaftaran , tambah kota, ubah kota |
| PS20 | Ongkos kirim dibutuhkan | Peringatan | Tambah ongkos kirim |
| PS21 | Anda telah berhasil login | Pemberitahuan | Home administrator, Home operator |
| PS22 | Harga hanya berisi angka | Peringatan | Tambah produk, ubah produk |
| PS23 | Maaf, Username tidak dikenali | Peringatan | Login Administrator, Login Operator |
| PS24 | Username atau Password salah! | Peringatan | Login Administrator, Login Operator |
| PS25 | Data telah berhasil disimpan | Pemberitahuan | Tambah kategori |
| PS26 | Username Dibutuhkan | Peringatan | Login Administrator, Login Operator |
| PS27 | Judul dibutuhkan | Peringatan | Tambah informasi |
| PS28 | Data berhasil dihapus | Pemberitahuan | Admin |
| PS29 | Deskripsi produk | Peringatan | Tambah produk, |

| | | | |
|---|---|---|---|
| | dibutuhkan | | ubah produk |
| PS30 | Data sudah ada, tidak dapat disimpan | Peringatan | Tambah kategori, ubah kategori |
| PS31 | Data telah berhasil diubah | Pemberitahuan | Ubah kategori |
| PS32 | Maaf, kode x sisa barang hanya x | Peringatan | Keranjang |
| PS33 | Password Baru dibutuhkan | Peringatan | Ganti Password |

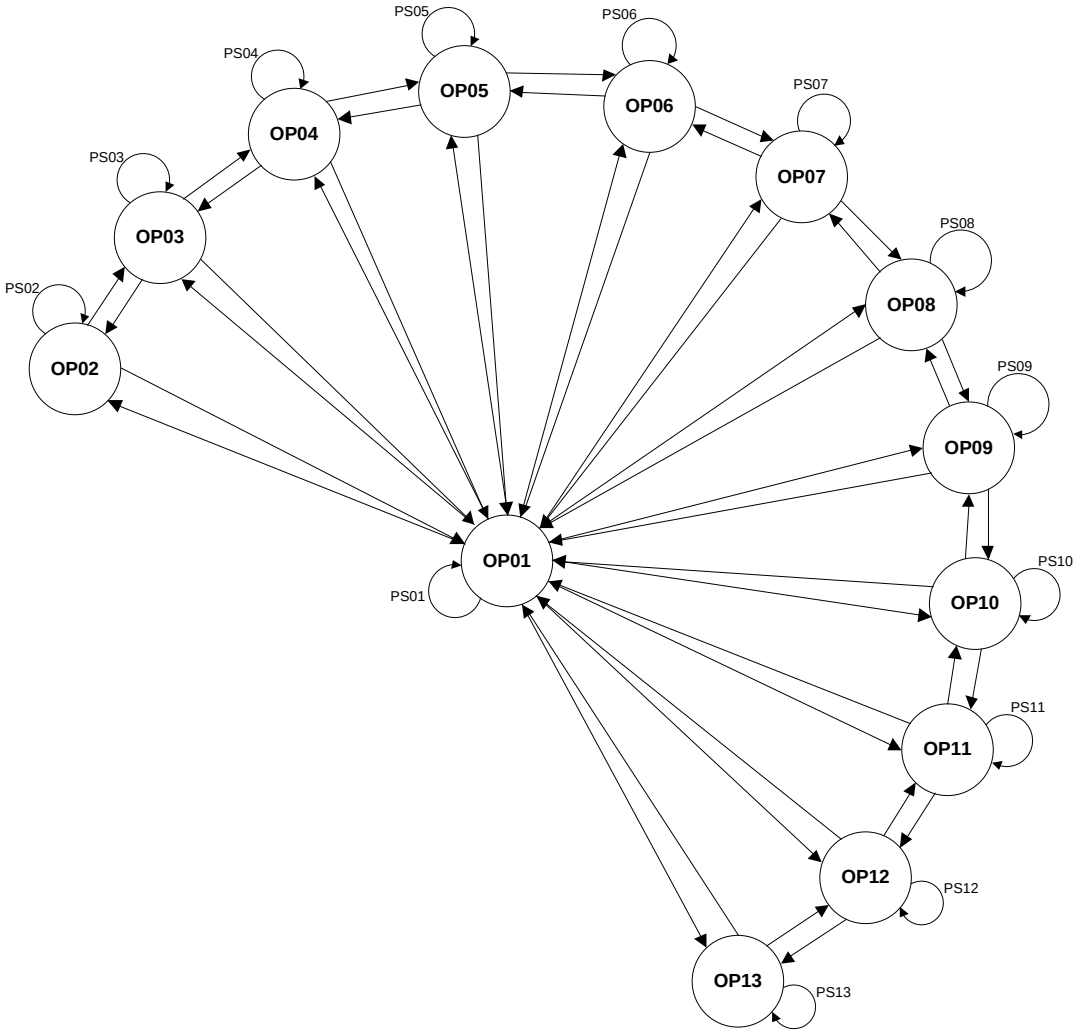
### 3.2.5. Jaringan Semantik

#### 1. Jaringan Semantik *Admin*



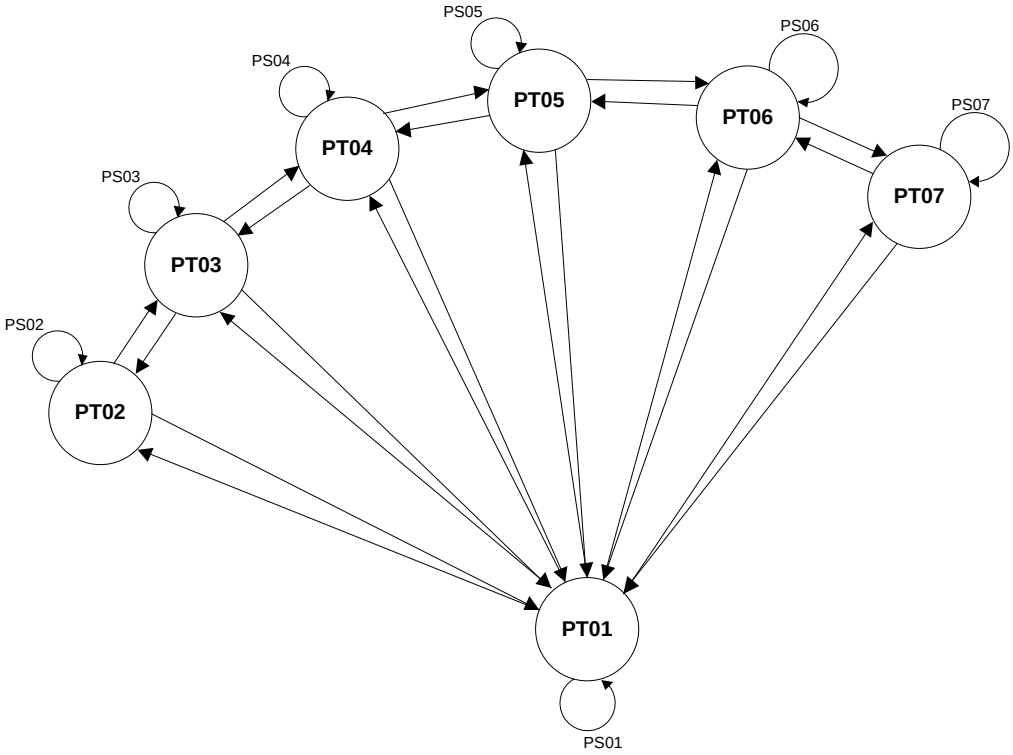

**Gambar 3.153 Jaringan Semantik *Admin***

## 2. Jaringan Semantik Operator



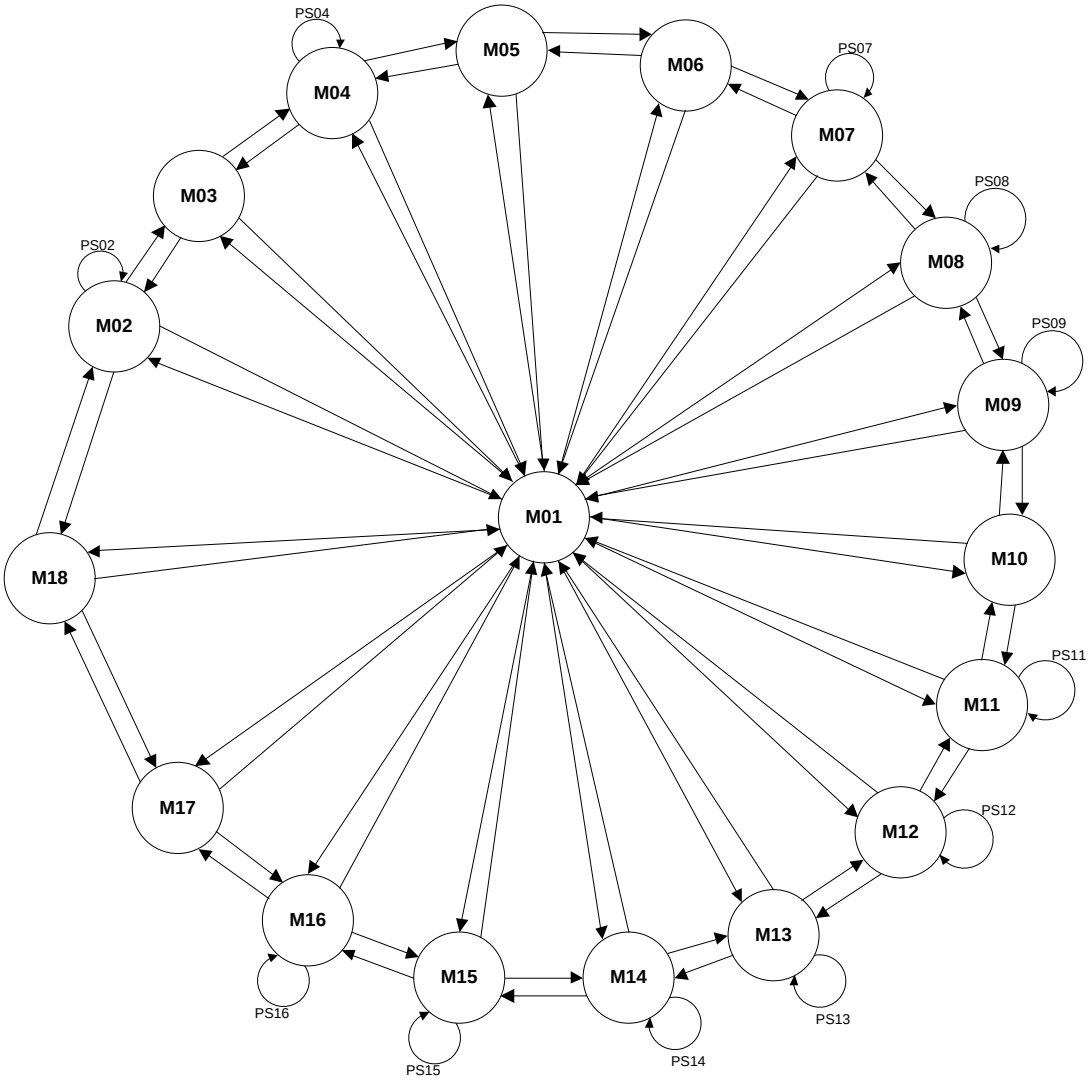

**Gambar 3.154 Jaringan Semantik Operator**

### 3. Jaringan Semantik Pemilik Toko



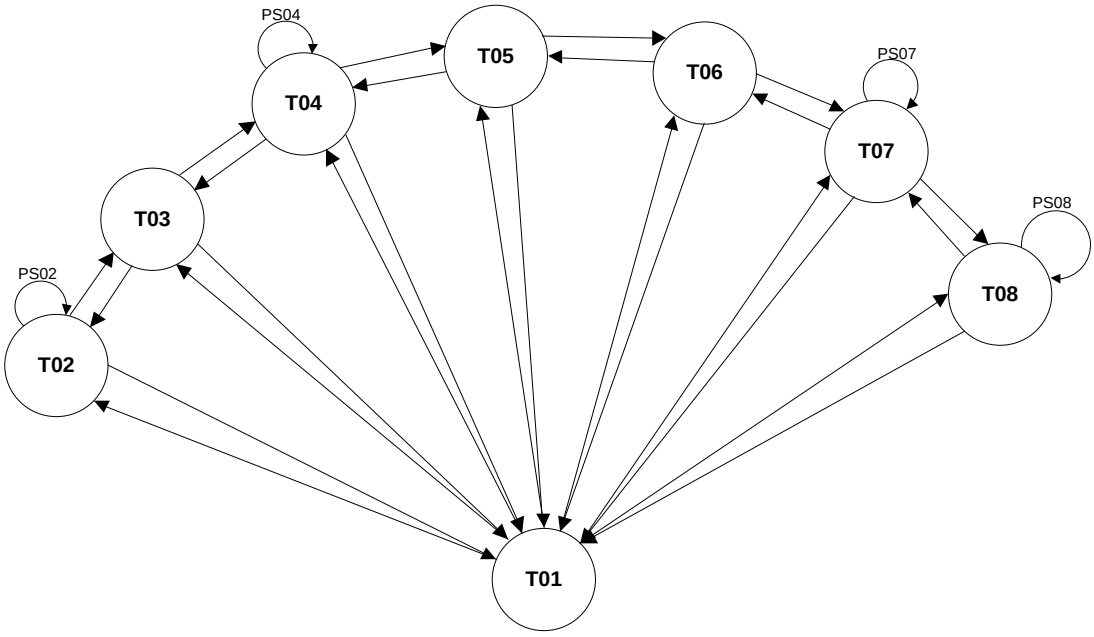

**Gambar 3.155 Jaringan Semantik Pemilik Toko**

## 4. Jaringan Semantik Member




**Gambar 3.156 Jaringan Semantik Member**

**5. Jaringan Semantik Pengunjung**



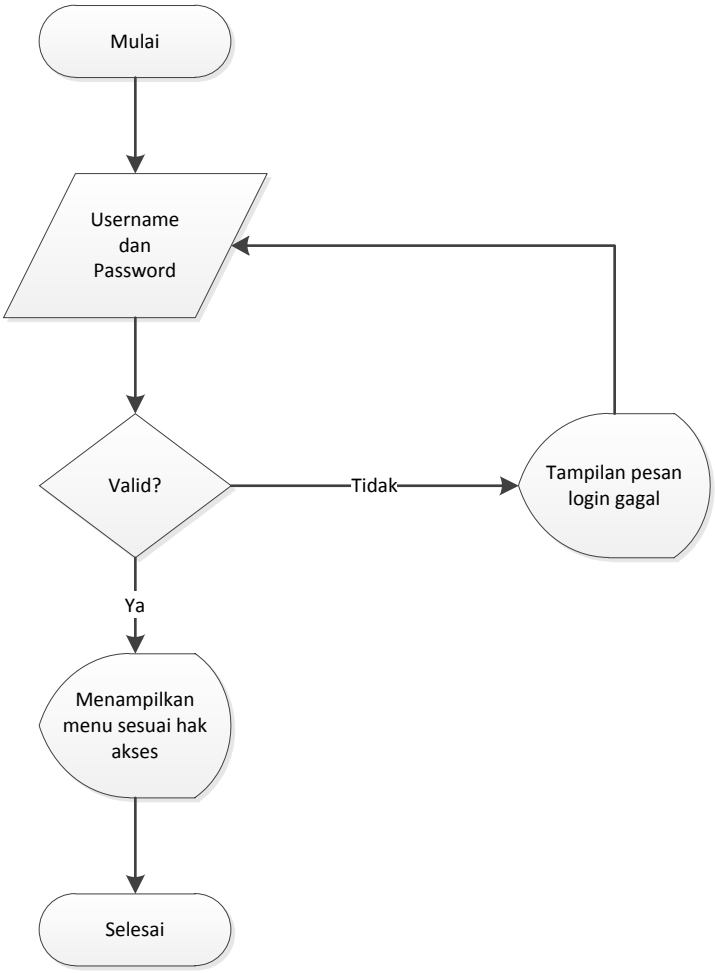

**Gambar 3.157 Jaringan Semantik Pengunjung**

### 3.2.6. Perancangan Prosedural

Sebagai langkah terakhir dalam perancangan yaitu perancangan procedural yang akan di implementasikan ke dalam sistem. Prosedur ini akan digunakan sebagai algoritma dasar dalam mengkodekan prosedur yang ada. Adapun perancangan prosedural untuk pembangunan sistem *e-commerce* pada Toko RusticShop yang akan dibangun adalah sebagai berikut:

### 1. *Flowchart* Login

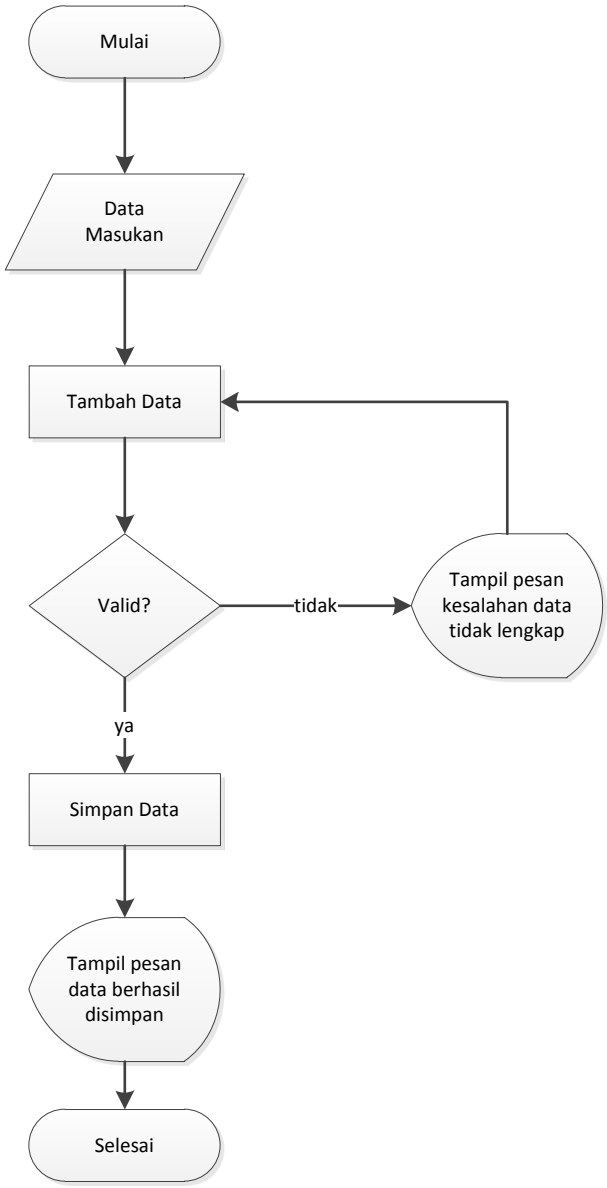
Diagram alir pada prosedur *login* adalah sebagai berikut:




**Gambar 3.158 *Flowchart Login***

## 2. *Flowchart* Tambah Data

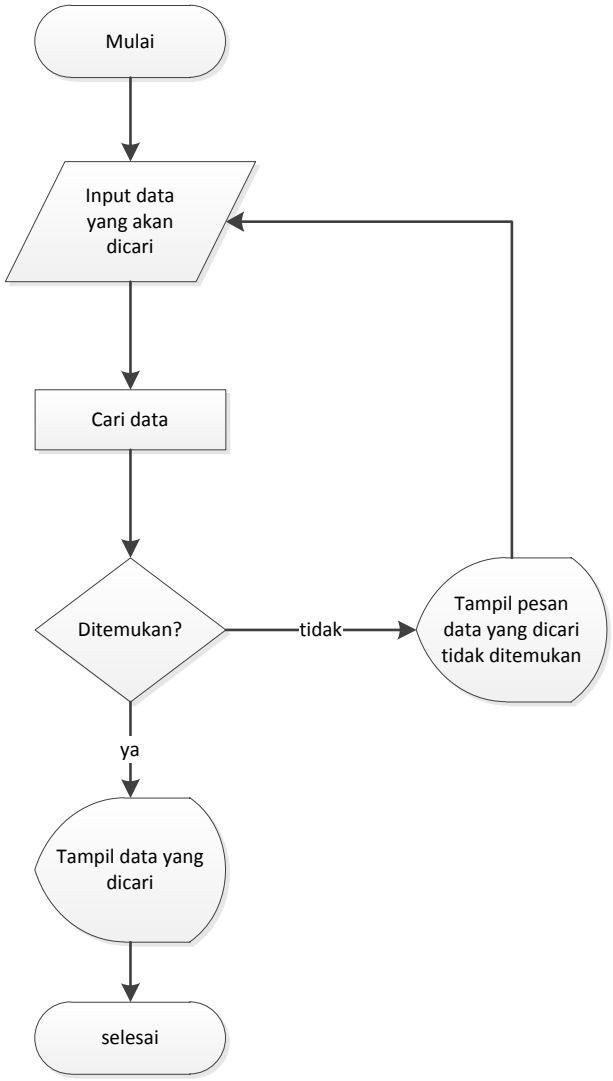
Diagram alir pada prosedur tambah data adalah sebagai berikut:




**Gambar 3.159 *Flowchart* Tambah Data**

### 3. *Flowchart* Cari Data

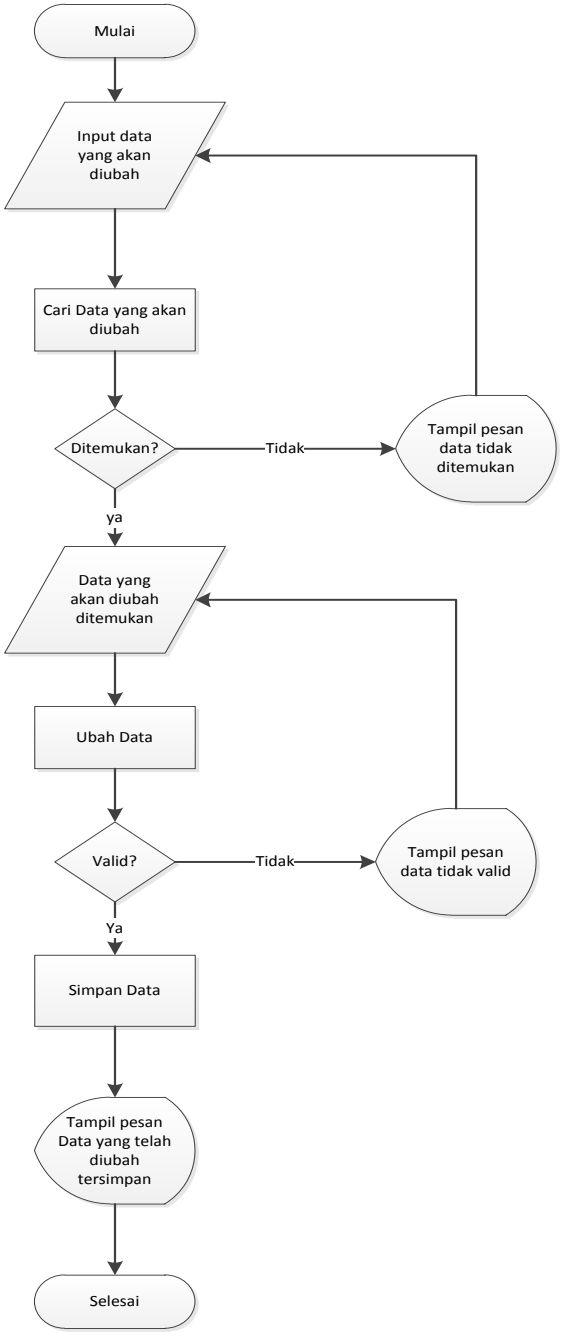
Diagram alir pada prosedur cari data adalah sebagai berikut:




**Gambar 3.160 *Flowchart* Cari Data**

### 4. *Flowchart* Ubah Data

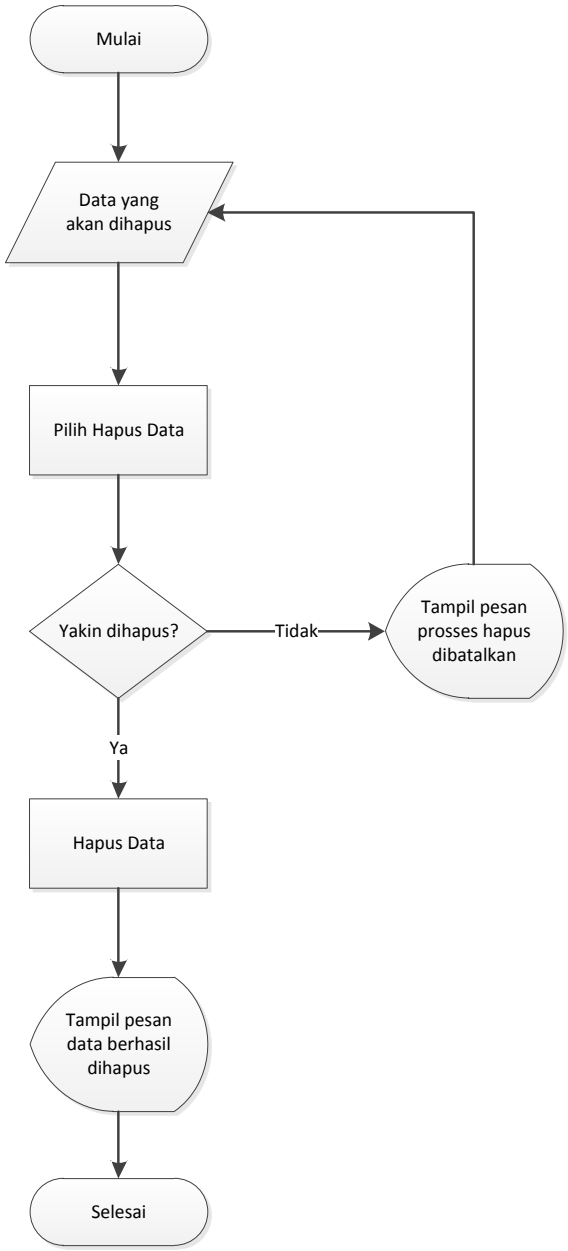
Diagram alir pada prosedur ubah data adalah sebagai berikut:




**Gambar 3.161 *Flowchart* Ubah Data**

### 5. *Flowchart* Hapus Data

Diagram alir pada prosedur hapus data adalah sebagai berikut:




**Gambar 3.162 *Flowchart* Hapus Data**