# BAB 3

# ANALISIS DAN PERANCANGAN SISTEM

## 3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan, yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Pada tahap ini yang perlu dilakukan adalah analisis terhadap sistem yang sedang berjalan dan bagaimana aliran dokumen yang sedang berjalan yang akan digambarkan dalam bentuk *flowmap*, analisis kebutuhan fungsional, analisis kebutuhan non-fungsional, dan analisis pengguna.

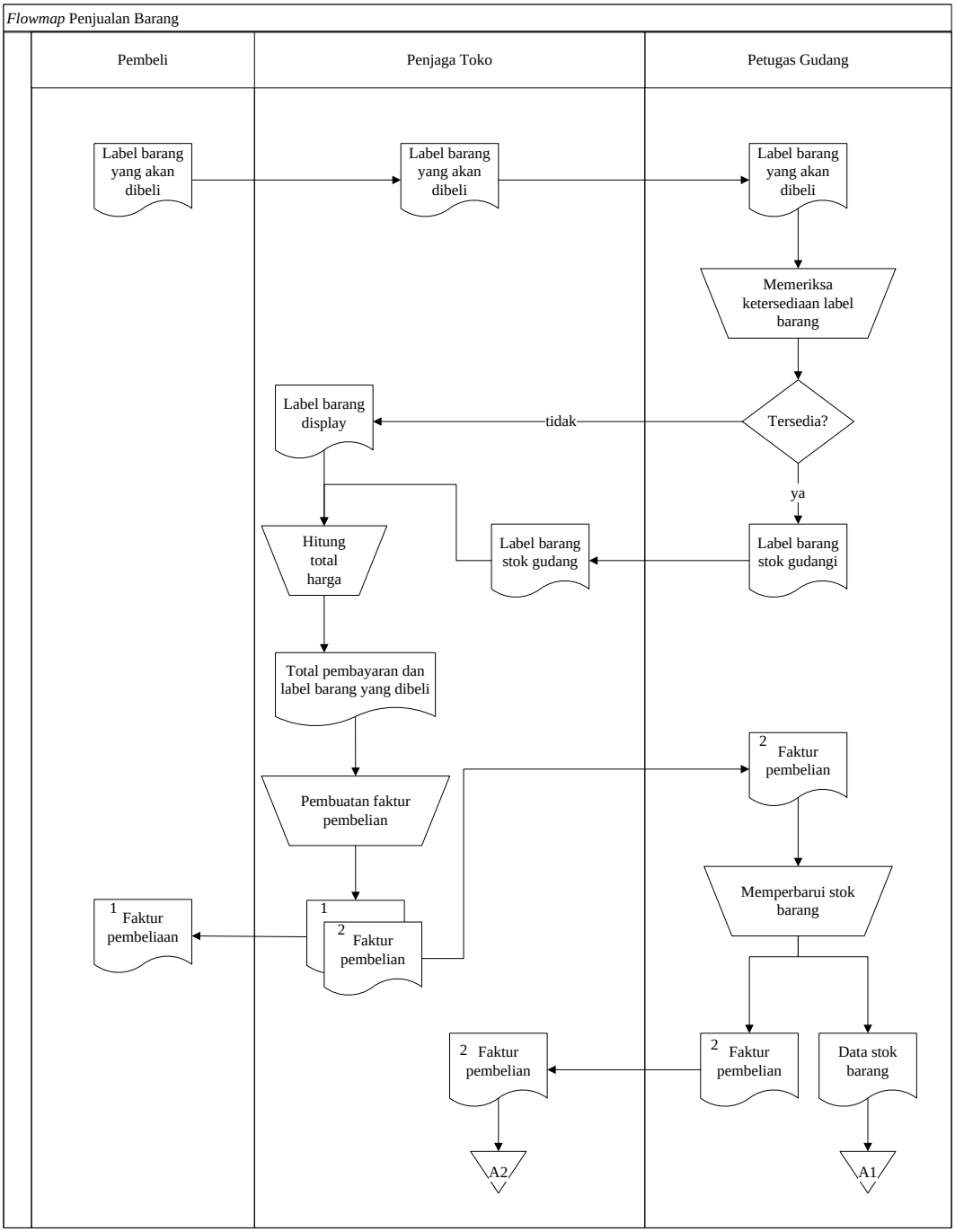
### 3.1.1 Analisis Sistem yang Sedang Berjalan

Berdasarkan hasil *observasi* yang dilakukan di My Joy Ride *Shop*, terdapat beberapa prosedur yang dilakukan diantaranya adalah prosedur penjualan barang, prosedur pengembalian barang (retur), prosedur pembuatan laporan.

#### 3.1.1.1 Prosedur Penjualan Barang

Prosedur penjualan langsung atau pembelian yang dilakukan dengan mendatangi My Joy Ride *Shop* yang sedang berlangsung saat ini (Gambar 3.1) adalah sebagai berikut:

1. Pembeli memilih barang dari barang yang dipajang lalu diberikan kepada penjaga toko.
2. Penjaga toko akan mengkonfirmasi kepada petugas gudang apakah barang tersebut memiliki stok atau tidak, jika stok barang tersebut kosong maka barang yang dijual adalah barang *display*.
3. Penjaga toko akan menghitung total belanja dari pembeli dan membuatkan faktur pembelian dalam rangkap dua.
4. Barang dan faktur pertama pembelian kemudian akan diserahkan ke pembeli.
5. Pembeli melakukan pembayaran sesuai dengan total harga barang yang dibeli.
6. Penjaga toko akan memberikan faktur kedua pembelian kepada petugas gudang untuk memperbarui daftar barang.
7. Setelah petugas gudang memperbarui daftar barang, maka faktur tersebut dikembalikan kepada penjaga toko.
8. Penjaga toko akan mengarsipkan faktur pembelian kedalam pembukuan.




**Gambar 3.1 *Flowmap* Penjualan Barang.**

Keterangan :

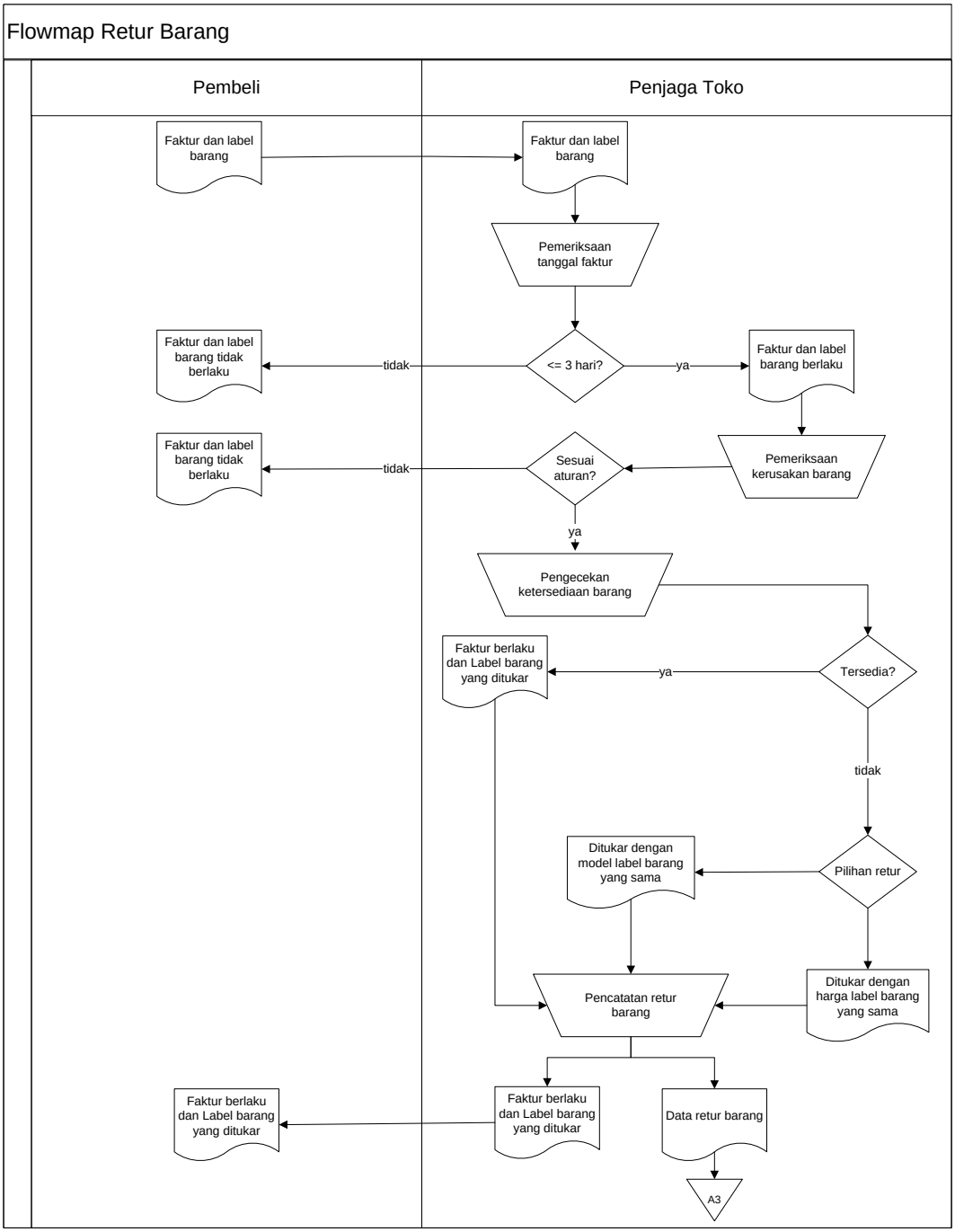
A1 : Arsip data stok barang oleh petugas gudang.

A2 : Arsip faktur penjualan oleh penjaga toko.

### 3.1.1.2 Prosedur Pengembalian Barang (*Retur*)

Prosedur pengembalian barang yang dilakukan dengan mengembalikan langsung barang ke My Joy Ride *Shop* dengan syarat barang yang dibeli tidak lebih dari tiga hari masa pembelian, prosedur pengembalian barang yang sedang saat ini (Gambar 3.2) adalah sebagai berikut:

1. Pembeli mengembalikan barang beserta faktur pembelian.
2. Penjaga toko akan memeriksa tanggal pembelian. Jika tanggal pembelian lebih dari tiga hari, barang tidak dapat ditukar. Jika tanggal pembelian tidak lebih dari tiga hari, maka barang dapat ditukar.
3. Penjaga toko memeriksa barang yang diretur, apabila terjadi kerusakan maka retur dapat dilakukan. Apabila tidak terjadi kerusakan barang retur tidak dapat dilakukan.
4. Penjaga toko memeriksa ketersediaan barang. Jika barang tersedia, maka barang ditukar. Jika barang tidak tersedia maka pembeli dapat memilih pilihan retur barang yang ditetapkan oleh My Joy Ride *Shop*.
5. Penjaga toko mencatat penukaran barang dalam bentuk pembukuan penukaran barang.
6. Pembeli menerima barang hasil penukaran.




**Gambar 3.2 *Flowmap* Pengembalian Barang.**

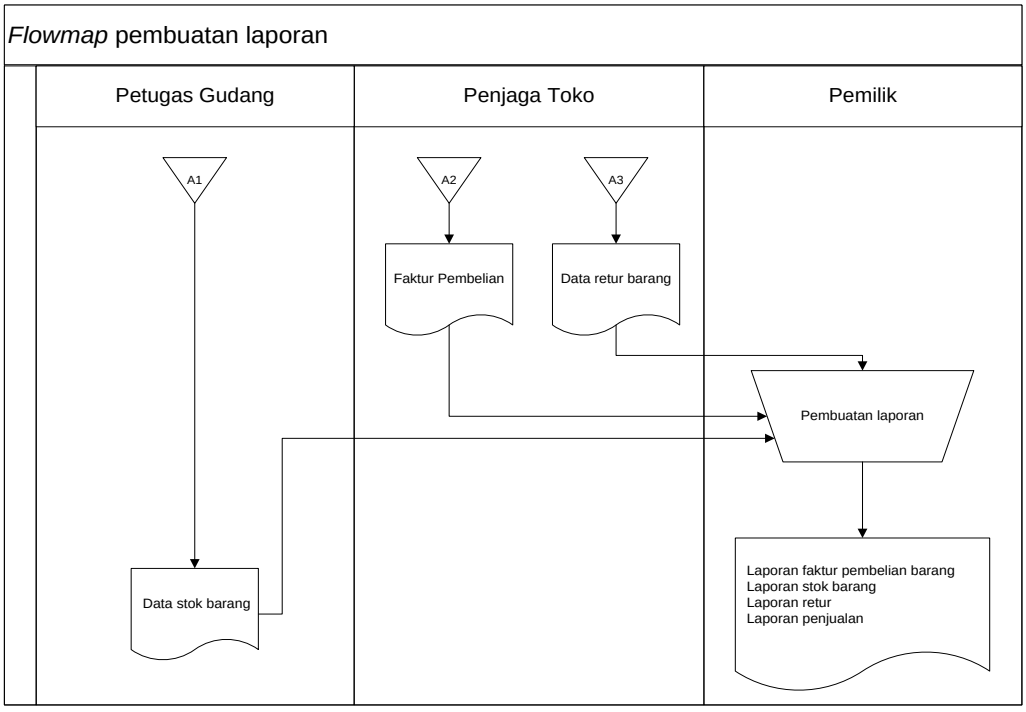
Keterangan :

A3 : Arsip data pengembalian barang oleh penjaga toko.

### 3.1.1.3 Prosedur Pembuatan Laporan

Prosedur pembuatan laporan yang dilakukan dengan mengumpulkan arsip dan data barang yang disimpan pemilik My Joy Ride *Shop* yang sedang berlangsung saat ini (Gambar 3.3) adalah sebagai berikut:

1. Petugas gudang memberikan daftar pembaruan barang kepada pemilik.
2. Penjaga toko memberikan pembukuan penjualan, faktur pembelian, dan pembukuan retur kepada pemilik
3. Pemilik membuat laporan dari arsip-arsip yang telah dikumpulkan oleh penjaga toko dan petugas gudang.




**Gambar 3.3 *Flowmap* Pembuatan Laporan.**

Keterangan :

A1 : Arsip data sok barang oleh petugas gudang

A2 : Arsip faktur penjualan oleh penjaga toko.

A3 : Arsip data pengembalian barang penjaga toko.

#### 3.1.1.4 Aturan Bisnis

Aturan bisnis di *website e-commerce* My Joy Ride *Shop* adalah sebagai berikut:

1. Stok akan berkurang ketika proses selesai dipesan (*chekout*).
2. Stok barang dibedakan antara penjualan *online* dan *offline*.
3. Barang yang dijual adalah barang *ready stock.*
4. Harga barang belum termasuk ongkos kirim.
5. Barang akan dikirim setelah adanya transaksi pembayaran dan mendapat konfirmasi pembayaran *valid*.
6. Pemberian diskon tergantung kebijakan pemilik toko, pemberian diskon meliputi event-event tertentu seperti diskon akhir tahun dan tahun baru, barang ditentukan sendiri oleh pemilik toko untuk diberi diskon .
7. Barang yang diberi diskon biasanya adalah barang yang selama enam bulan hanya berkurang lima dari stok yang ada dengan diskon sebesar 20%.

8. Pesanan yang sudah dibayar tidak dapat dibatalkan.
9. Batas konfirmasi retur tidak lebih dari tiga hari, terhitung saat barang diterima, setelah melakukan konfirmasi retur barang langsung dikirim ke pihak My Joy Ride *Shop*.
10. Biaya pengiriman retur ditanggung pembeli.
11. Pengembalian barang berlaku bila barang dalam keadaan cacat atau barang yang dikirim tidak sesuai.
12. Retur tidak berlaku apabila kerusakan dikarenakan pembeli.
13. Retur hanya bisa dilakukan dengan barang yang sama, jika stok habis maka uang kembali.
14. Batas waktu pembayaran adalah 1x24 jam dari waktu pemesanan.
15. Jika melewati batas waktu pembayaran, pesanan akan dibatalkan, dan pembeli akan mendapat peringatan, jika hal tersebut terjadi hingga tiga kali, maka *member* akan di non-aktifkan.
16. Jika Pemesanan dan konfirmasi pembayaran dilakukan setelah jam 3 sore, maka pengiriman dilakukan besoknya.
17. Pengiriman pesanan dapat dilakukan sesuai data profil atau dapat dikirim ke alamat lain, pengiriman dilakukan sesuai jam kerja My Joy Ride *Shop*.

### 3.1.2 Analisis Masalah

Berdasarkan hasil penelitian yang telah dilakukan dengan cara *observasi* dan *interview* dengan pihak My Joy Ride *Shop* bahwa toko tersebut masih memiliki kendala seperti :

1. Sistem penjualan yang masih dilakukan secara konvensional.
2. Promosi barang yang masih kurang efektif.
3. Kurangnya media untuk pemasaran.

Berdasarkan permasalahan yang telah diuraikan maka perlu dibangunnya sistem yang terkomputerisasi untuk memaksimalkan dalam proses penjualannya.

### 3.1.3 Analisis Sistem Rekomendasi *Collaborative Filtering*

Menu rekomendasi merupakan menu yang akan menampilkan barang yang akan direkomendasikan kepada pelanggan tersebut. Rekomendasi tersebut berasal dari perhitungan algoritma pembangkit rekomendasi. Barang yang memiliki nilai kegunaan tertinggilah yang kemudian dijadikan rekomendasi. Pelanggan akan mendapatkan rekomendasi buku apabila pelanggan tersebut telah melakukan rating satu atau lebih item yang disediakan oleh sistem rekomendasi. Jadi jika pelanggan tersebut belum melakukan rating terhadap item di sistem, maka pelanggan tersebut akan mendapatkan rekomendasi rata-rata dari jumlah keseluruhan rating pelanggan.

Pemberian nilai rekomendasi terdiri beberapa langkah, yaitu:

1. Pengecekan pelanggan, jika diketahui pelanggan A login ke sistem, maka sistem akan mengecek siapa saja pelanggan dengan riwayat pembelian atau peratingan yang sama dengan pelanggan A.
2. Jika sudah diketahui pelanggan-pelanggan tersebut maka sistem akan menghitung jumlah pelanggan yang sama dengan pelanggan A.
3. Setelah data sudah didapat maka pembangkitan rekomendasi akan dihitung dengan persamaan:

$$R = \frac{\frac{\sum B \times rating \ \times 100}{5 \times \mathrm{I}}}{C} \tag{3.1}$$

Keterangan:

B = Nilai pembelian barang (1 jika membeli dan 0 jika tidak membeli)

Rating = Jumlah rating yang diberikan oleh pelanggan pada item yang disuka.

100 = Nilai maksimum 100, nilai rekomendasi tertinggi adalah 100.

5 = Nilai maksimum rating yang diberikan oleh pelanggan.

1 = Nilai maksimum pembelian yang dilakukan oleh pelanggan, jika pelanggan membeli barang lebih dari satu barang yang sama, dalam persamaan ini tetap akan dihitung membeli satu barang.

C = jumlah pengguna yang mempunyai pola pembelian yang sama dengan pelanggan yang melakukan login.

Setelah pembangkit rekomendasi dihitung maka sistem akan merekomendasikan beberapa item.

Contoh : diasumsikan pembeli bernama Damon.

**Tabel 3.1 Tabel Skenario Pembangkitan Rekomendasi.**

| Kode Barang | ME007 (Damon) | ME008 (Albarn) | ME009 (Sandara) | ME010 (Noel) | ME011 (Zoey) |
|---|---|---|---|---|---|
| TS0001 | $1^4$ | 0 | $1^1$ | $1^4$ | 0 |
| TS0002 | $1^5$ | $1^3$ | 0 | $1^3$ | $1^2$ |
| TS0003 | $1^4$ | $1^5$ | $1^3$ | 0 | 0 |
| TS0004 | $1^2$ | $1^4$ | $1^2$ | $1^2$ | $1^3$ |
| TS0005 | 0 | 0 | 1 | 0 | 0 |

Keterangan:

TS0001-TS0005 = kode barang yang ada pada toko.

ME007-ME011 = kode pelanggan yang sudah terdaftar.

Angka 1 dan 0 = 0 jika pelanggan tidak membeli barang, 1 jika pelanggan membeli barang.

Angka superskrip (1-5)= rating pelanggan yang diberikan oleh pelanggan.

Skala rating = 1 (tidak suka), 2 (sedikit suka), 3 (lumayan suka), 4 (suka), dan 5 (sangat suka).

Jika pembeli sudah membeli barang tetapi belum melakukan peratingan maka barang tersebut tidak dihitung nilai rekomendasinya. Dari data yang didapat pada Tabel 3.1, maka pembangkitan rekomendasi akan dihitung.

Rating Barang TS0001 =(((1*4*100)/5*1) + 0 + ((1*1*100)/5*1) +
((1*4*100)/5*1) + 0) / 4= 45

Rating Barang TS0002 =(((1*5*100)/5*1) + ((1*3*100)/5*1) + 0 +
((1*3*100)/5*1) + ((1*2*100)/5*1)) / 4= 65

Rating Barang TS0003 =(((1*4*100)/5*1) + ((1*5*100)/5*1) + ((1*3*100)/5*1)
+ 0 + 0 / 4= 60

Rating Barang TS0004 =(((1*2*100)/5*1) + ((1*4*100)/5*1) + ((1*2*100)/5*1)
+ ((1*5*100)/5*1) + ((1*3*100)/5*1)) / 4= 80

Jadi rekomendasi barang untuk pembeli yang bernama Damon terdapat pada Tabel 3.2.

**Tabel 3.2 Tabel Skenario Hasil Pembangkitan Rekomendasi.**

| No | Kode Barang | Nilai Rekomendasi |
|---|---|---|
| 1 | TS0004 | 80 |
| 2 | TS0002 | 65 |
| 3 | TS0003 | 60 |
| 4 | TS0001 | 45 |

### 3.1.4 Analisis Fitur

Fitur yang terdapat pada aplikasi ini terbagi menjadi beberapa fitur yaitu:

A. *Administrative Tools* (*Backend Office*)

1. Perancangan *website* dibuat berdasarkan permintaan dan kebutuhan My Joy Ride *Shop*. Sistem yang dibangun dapat menangani pengelolaan data barang dan layanan yang disediakan My Joy Ride *Shop* serta menampilkannya berdasarkan kategori.

2. Sistem menyediakan layanan pengaturan barang yaitu:

   a. Data Barang.

   b. Data Kategori Barang Baju dan Jaket.

   c. Detail Barang (Gambar, Deskripsi Barang, Harga, Ukuran, Warna).

   d. Manajemen Harga (layanan yang dilakukan jika ada perubahan harga).

   e. Rating Barang.

   f. Produksi Promosi.

   g. Pengelolaan Diskon.

   h. Pengelolaan Kurs.

   i. Pengelolaan Stok.

3. Sistem menyediakan manajemen pemesanan yaitu:

a. Pemesan harus melakukan registrasi terlebih dahulu.

b. Permintaan pemesanan akan dikonfirmasi oleh pemilik toko atau pengelola.

c. Pengiriman dilakukan oleh jasa pengiriman kurir.

4. Manajemen pembayaran yang ditangani yaitu:

a. Pembayaran *Offline*: Transfer via Bank BCA.

b. Pembayaran *Online*: Paypal

c. Pembayaran CoD (*Cash on Delivery*) jika pemesan masih berada di area kota Bandung.

5. Manajemen Pengiriman yaitu:

a. Pada pengiriman barang dilakukan melalui jasa pengiriman kurir JNE.

b. Sistem mengolah kota/kabupaten berikut harganya tiap kota/kabupaten.

6. Pembuatan Laporan

Dalam pembuatan laporan dapat disusun dalam jangka perbulan. Terdapat beberapa laporan yang dibuat yaitu Laporan Penjualan, Laporan Pemesanan, Laporan Barang.

7. Sebagai media promosi sistem yang dibangun akan berintegrasi dengan situs jejaring sosial seperti facebook dan twitter.

8. Konfirmasi Pembayaran

   Konfirmasi pembayaran dapat dilakukan dengan cara memberikan konfirmasi kepada pembeli melalui *email*.

9. *BackUp* dan *Restore*

   Sistem akan menggunakan fasilitas *backup* dan *restore* yang telah disediakan oleh *webhost* yang digunakan.

10. Pengelolaan Retur

    Fasilitas pengelolaan Retur barang dilakukan apabila barang tidak sesuai dengan keinginan pembeli maka barang akan ditarik kembali oleh toko dan diganti sesuai dengan barang yang diinginkan dalam jangka waktu 3 hari setelah barang diterima.

11. Sistem yang dibangun memperhatikan keamanan pada saat terjadi pertukaran data dan transaksi dengan memanfaatkan layanan SSL yang terdapat pada *webhost*. Dengan membuat koneksi yang ter-enkripsi antara server dengan pengunjungnya. Selain itu, sistem akan melakukan *valid*asi terhadap masukan pengguna untuk menghindari kesalahan masukan data dan menggunakan *username* dan *password* saat login, *password* tersebut sudah ter-enkripsi.

B. Fitur Belanja (*Frontend*)

1. Pencarian Barang (berdasarkan nama, fitur, harga, dan kategori).

2. Menampilkan barang diskon, barang terbaru, barang terlaku, barang yang sering dilihat.
3. Fasilitas pendaftaran *member* dan *login member*.
4. *History* pemesanan untuk mempermudah pembeli dalam melakukan transaksi pemesanan.
5. Pengiriman informasi mengenai konfirmasi pemesanan melalui *email* ke pembeli setiap ada perubahan status pesanan.
6. *Zoom* gambar barang.
7. Mendukung SEO (*Search Engine Optimization*) agar situs terdaftar di *search engine* pada halaman-halaman depan.
8. Terdapat pengisian *testimony* (komentar).
9. Pembeli dapat melakukan retur dengan ketentuan yang berlaku.
10. *Smart Recommendation Syatem*

    Dimana sistem akan merekomendasikan barang lainnya agar pembeli dapat memilih barang lain yang direkomendasikan oleh sistem. Dengan menggunakan metode *content based recomendation*.

C. Keamanan

1. Menggunakan *IP-Dedicated* untuk keamanan data pembeli.

2. Menggunakan SSL (*Secure Socket Layer*) untuk menjamin keamanan data dan keamanan yang mendukung protokol https. Dengan membuat koneksi yang ter-enkripsi antara server atau situs dengan pengunjung.

3. Menggunakan *username* dan *password* oleh *member* yang sudah terdaftar untuk *autentifikasi* pengguna dan *password* yang udah ter-enkripsi.

D. Komunikasi

Fasilitas yang dapat dipakai sebagai komunikasi antara pemesan dan pemilik toko, diantaranya:

1. Menggunakan *email* yang telah di daftarkan pemesan saat pendaftaran.
2. Menggunakan Yahoo messenger untuk informasi *real time* secara *online*.
3. Memanfaatkan situs jejaring sosial untuk promosi dengan menggunakan facebook dan twitter.
4. Menggunakan telepon untuk berbicara langsung atau dengan mengirimkan SMS (*Short Message Service*).

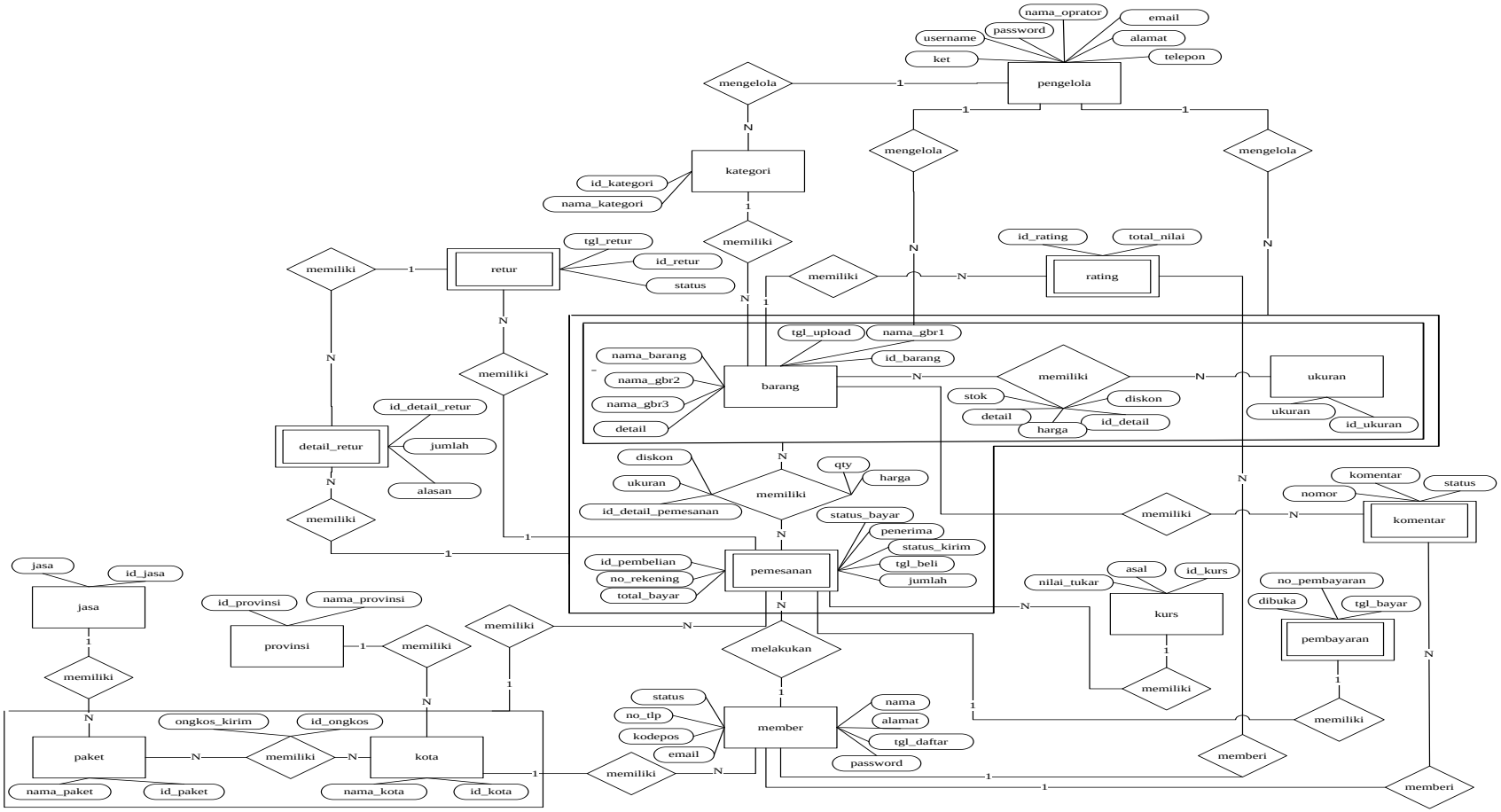
### 3.1.5 Analisis Basis Data

Basis data adalah representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Pada tahap ini digunakan untuk menganalisis data apa saja yang berelasi dan terlibat dalam pembuatan sistem.

### 3.1.5.1 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram (ERD)* merupakan suatu alat pemodelan yang digunakan untuk menggambarkan hubungan antar entitas yang terjadi, berisi komponen-komponen entitas dan relationship yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari sebagian dunia nyata.

Hubungan antar entitas yang terjadi dalam sistem *e-commerce* yang akan dirancang dapat dilihat pada Gambar 3.4 :




**Gambar 3.4** ***Entity Relationship Diagram***

### 3.1.6 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional meliputi elemen-elemen apa saja yang dibutuhkan, spesifikasi masukan yang diperlukan sistem, keluaran yang dihasilkan sistem dan proses yang dibutuhkan sampai sistem tersebut diimplementasi. Tahap ini terbagi menjadi tiga bagian yaitu analisis perangkat keras, analisis perangkat lunak, dan analisis pengguna.

#### 3.1.6.1 Analisis Perangkat Keras

Perangkat keras atau hardware merupakan peralatan fisik dari komputer yang dapat kita lihat dan rasakan. Perangkat keras ini merupakan perangkat yang sangat penting karena sebuah aplikasi yang dibangun tidak akan berjalan tanpa adanya perangkat keras.

Perangkat keras yang ada di My Joy Ride *Shop* adalah sebuah laptop dengan spesifikasi seperti yang terdapat pada Tabel 3.3.

**Tabel 3.3 Analisis Perangkat Keras di MyJoy Ride *Shop*.**

| No | Perangkat keras | Spesifikasi |
|---|---|---|
| 1 | Prosessor | Dualcore 2.0 Hz |
| 2 | VGA | 512 mb |
| 3 | Memori | 2 Gb |
| 4 | Harddisk | 160 Gb |
| 5 | Monitor | 14” |
| 6 | Koneksi Internet | Ya |

Sedangkan untuk sistem yang akan dibangun harus memiliki spesifikasi minimum perangkat keras seperti terdapat pada Tabel 3.4.

**Tabel 3.4 Tabel Perangkat Keras Minimum.**

| No | Perangkat keras | Spesifikasi |
|---|---|---|
| 1 | Prosessor | Dualcore 1.6 Hz |
| 2 | VGA | 256 Mb |
| 3 | Memori | 1 Gb |
| 4 | Harddisk | 100 Gb |
| 5 | Monitor | 14” |
| 6 | Koneksi Internet | Ya |

### 3.1.6.2 Analisis perangkat lunak

Perangkat lunak adalah istilah umum untuk data yang diformat dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang dibaca dan ditulis komputer. Perangkat lunak yang digunakan di My Joy Ride *Shop* untuk mengolah data adalah dengan menggunakan perangkat lunak *Microsoft Excel* 2007.

Sedangkan untuk membangun aplikasi, dibutuhkan perangkat lunak dengan spesifikasi seperti terdapat pada Tabel 3.5.

**Tabel 3.5 Tabel Perangkat Lunak.**

| No | Perangkat lunak | Spesifikasi |
|---|---|---|
| 1 | Sistem Operasi | Windows XP |
| 2 | Database | MySQL |
| 3 | Bahasa Pemograman | PHP |
| 4 | Tool Pembangun Aplikasi | Macromedia Dreamweaver 8 |
| 5 | Browser | Firefox, google chrome |

Untuk mengakses aplikasi yang akan dibangun, dibutuhkan perangkat lunak dengan spesifikasi minimal seperti terdapat pada Tabel 3.6.

**Tabel 3.6 Tabel Perangkat Lunak Minimum.**

| No | Perangkat lunak | Spesifikasi |
|---|---|---|
| 1 | Sistem Operasi | Windows XP |
| 2 | Browser | Firefox, google chrome |

### 3.1.6.3 Analisis *User*

My Joy Ride *Shop* saat ini memiliki 3 jenis *user* yaitu pemilik, petugas toko, dan penjaga gudang yang dapat dilihat di Tabel 3.7.

**Tabel 3.7 Analisis *User* di My Joy Ride *Shop*.**

| Pengguna | Pengalaman | Tingkat Pendidikan | Tingkat Keterampilan |
|---|---|---|---|
| Pemilik | Pernah menggunakan aplikasi berbasis *desktop* dan aplikasi berbasis *web* | Sarjana (S1) | Mampu menggunakan perangkat lunak *office* dan menggunakan *internet* |
| Penjaga Toko | Pernah menggunakan aplikasi berbasis *desktop* | SMA | Mampu menggunakan perangkat lunak *office* dan menggunakan *internet* |
| Petugas Gudang | Pernah menggunakan aplikasi berbasis *desktop* | SMA | Mampu menggunakan perangkat lunak *office* dan menggunakan *internet* |

*Website e-commerce* ini akan dipakai oleh 3 jenis yaitu pengelola, pengunjung *member* dan pengunjung *non* member yang dapat dilihat di Tabel 3.8.

**Tabel 3.8 Analisis *User* yang dibutuhkan.**

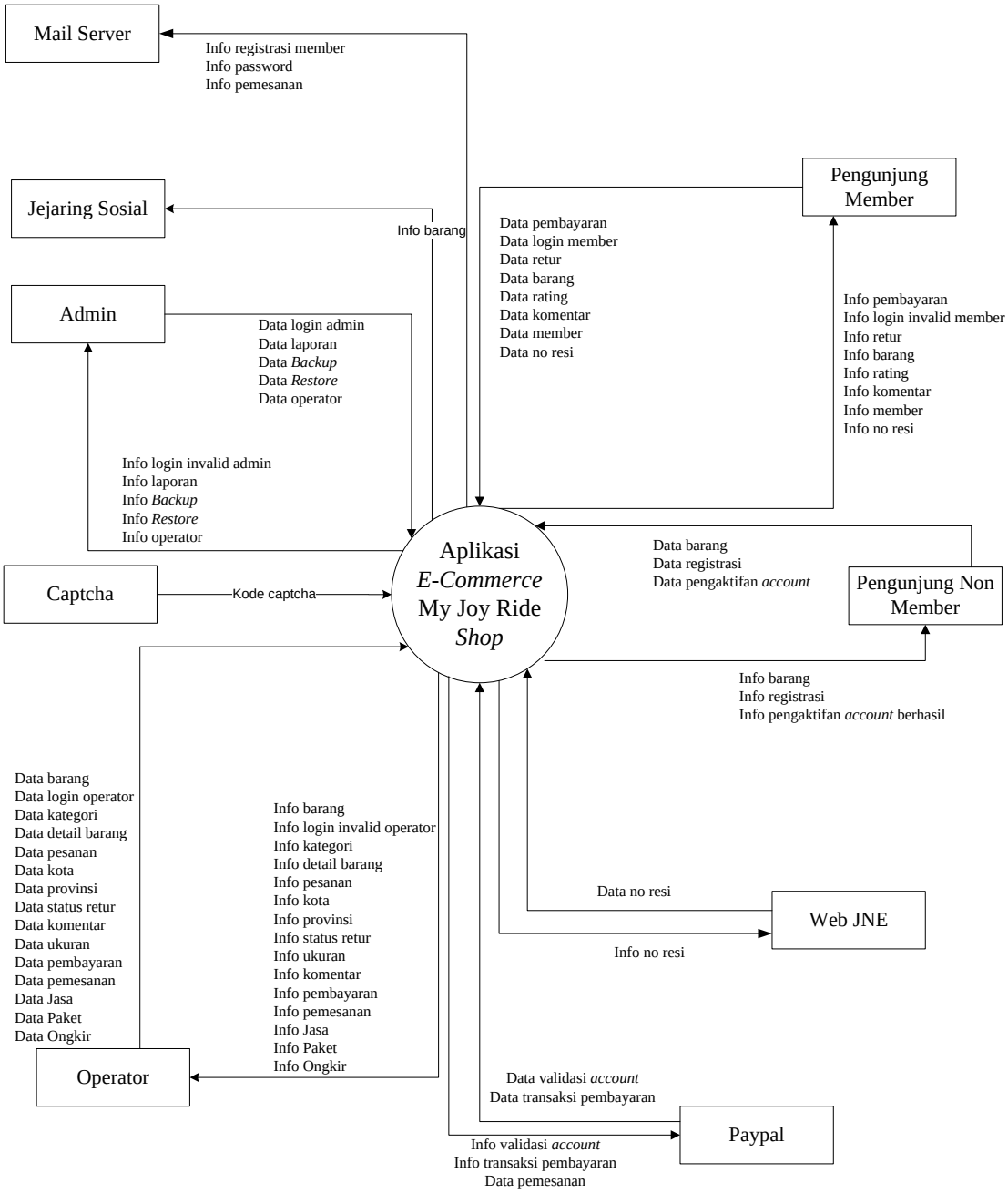
| Pengguna | Tanggung Jawab | Hak Akses | Tingkat Keterampilan | Pengalaman | Jenis Pelatihan |
|---|---|---|---|---|---|
| Admin (Pemilik) | Mengolah data yang terdapat dalam aplikasi | Melakukan pengolahan data pengelola, pengolahan *backup* dan *restore*, dan dapat melihat laporan | Menguasai penggunaan komputer, mengerti cara penggunaan aplikasi berbasis *web* dan menggunakan *internet* | Menggunakan komputer dan dapat menggunakan *internet* | Cara menggunakan aplikasi *website e-commerce* |
| Operator (Penjaga Toko) | Mengolah data yang terdapat dalam aplikasi | Melakukan pengolahan data *member*, data master seperti data barang, data ongkos kirim, data kota, dan data transaksi penjualan | Menguasai penggunaan komputer, mengerti cara penggunaan aplikasi berbasis *web* dan menggunakan *internet* | Menggunakan komputer dan dapat menggunakan *internet* | Cara menggunakan aplikasi *website e-commerce* |
| Pengunjung *member* | Melakukan pemesanan | Melakukan pemesanan , mengubah data pribadi, melihat *history* transaksi pembelian | Menggunakan komputer dan dapat menggunakan *internet* untuk *browsing* | - | - |
| Pengunjung *non member* | - | Hanya dapat melihat katalog barang | Menggunakan komputer dan dapat menggunakan *internet* dan *browsing* | - | - |

### 3.1.7 Analisis Kebutuhan Fungsional

Pemodelan analisis yang digunakan dalam pembangunan sistem ini berdasarkan analisis terstruktur menggunakan alat bantu diagram konteks dan *data flow digram* (DFD), spesifikasi proses dan kamus data.

#### 3.1.7.1 Diagram Konteks

Diagram konteks adalah arus data yang berfungsi untuk menggambarkan keterkaitan aliran-aliran data antara sistem dengan bagian-bagian luar. Adapun diagram konteks untuk sistem yang akan dibangun dapat dilihat pada Gambar 3.5.



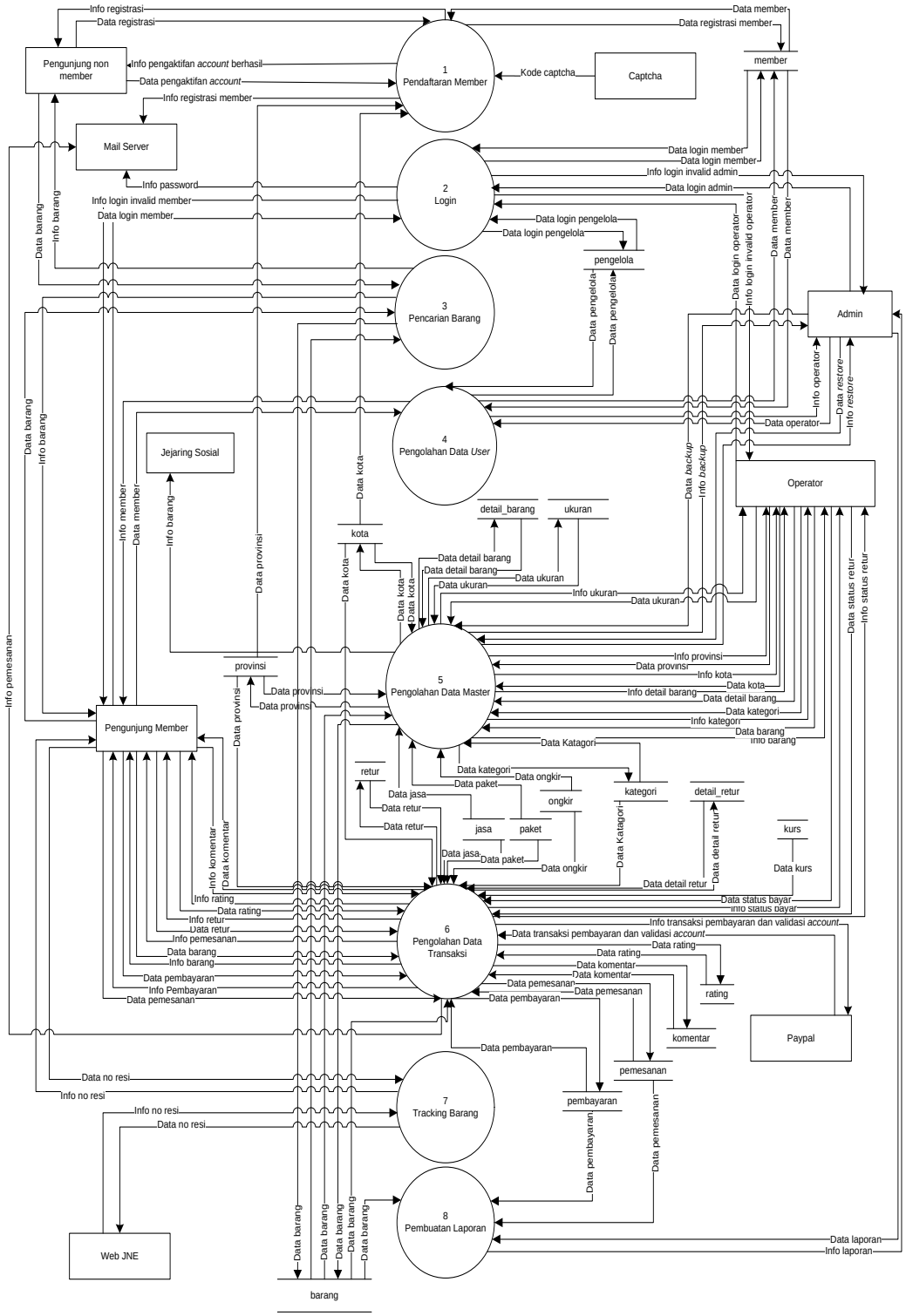

**Gambar 3.5 Diagram Konteks.**

### 3.1.7.2 *Data Flow Diagram* (DFD)

DFD (*Data Flow Diagram*) adalah sebuah teknik grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari *input* menjadi *output*. DFD juga dikenali sebagai grafik aliran data atau *bubbe chart*. DFD dapat digunakan untuk menyajikan sebuah sistem atau perangkat lunak pada setiap tingkat abstraksi. DFD dapat dipartisi ke dalam tingkat-tingkat yang merepresentasikan aliran informasi yang bertambah dan fungsi ideal. DFD memberikan suatu mekanisme bagi pemodelan fungsional dan pemodelan aliran informasi. [4]

#### 3.1.7.2.1 DFD Level 1

Berikut ini adalah DFD level 1 dari sistem yang akan dibangun.

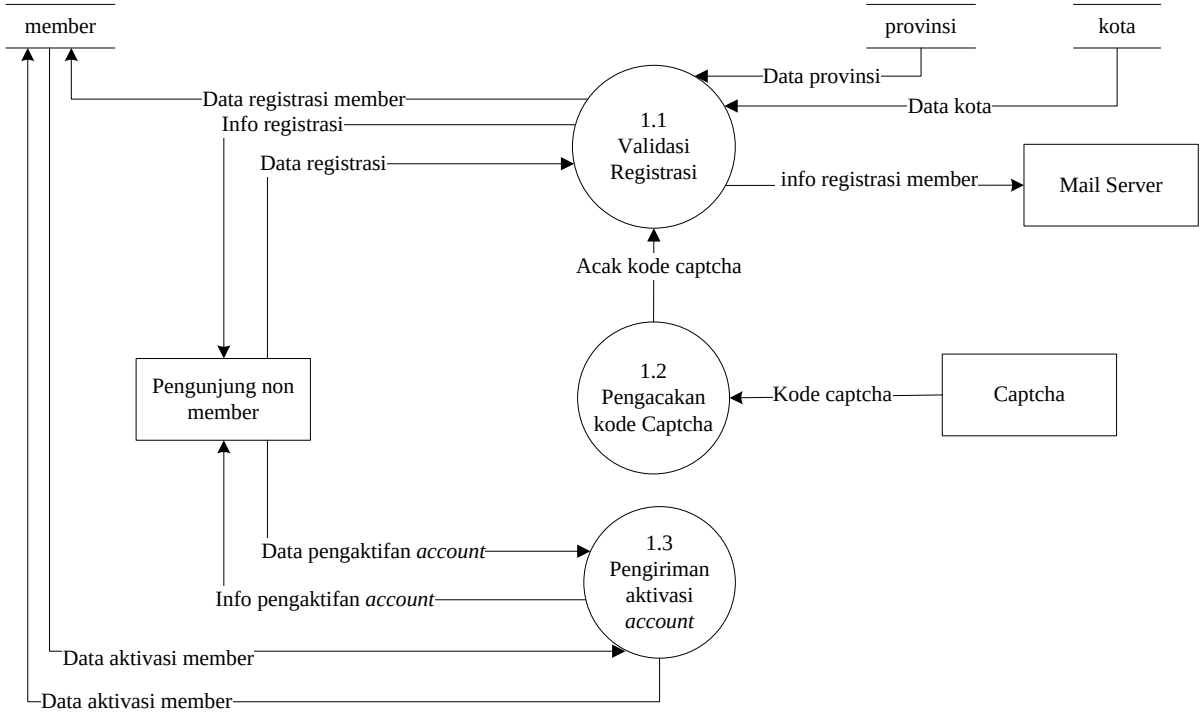



**Gambar 3.6 DFD Level 1.**

### 3.1.7.2.2 DFD Level 2

**1. DFD Level 2 Proses 1**

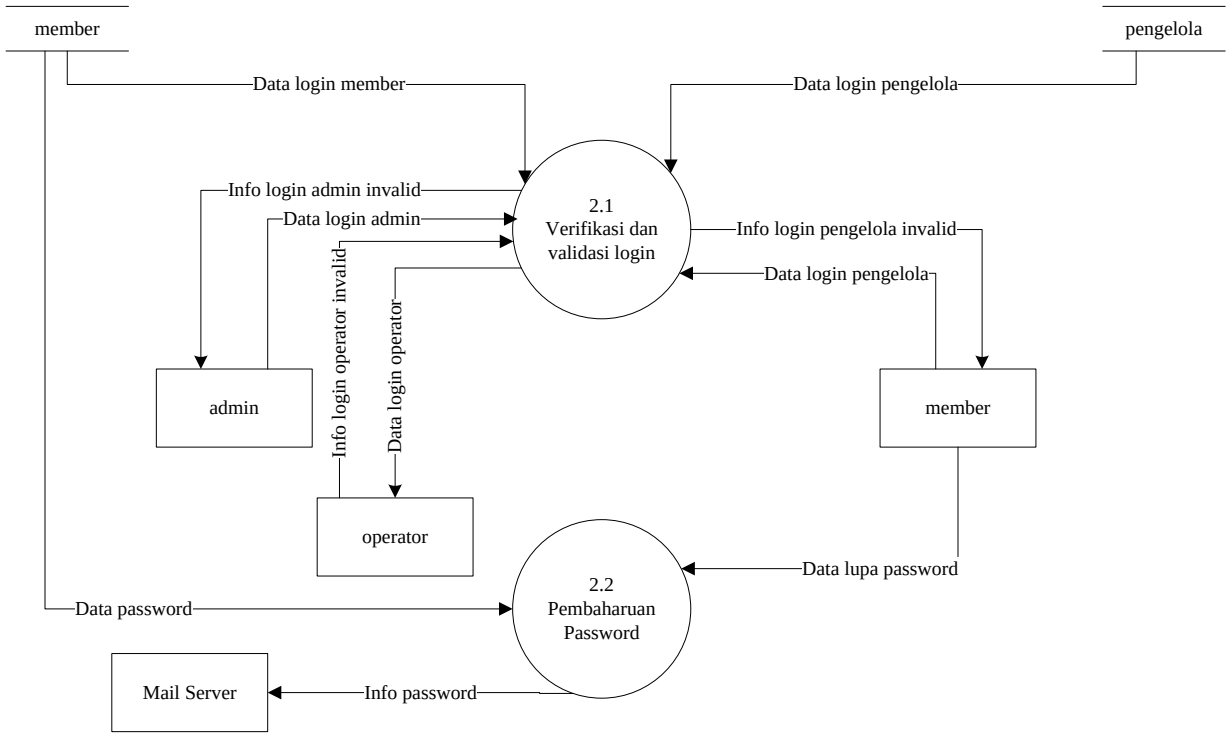
Diagram aliran data pada level 2 proses 1 daftar *member* dapat dilihat pada Gambar 3.7.




**Gambar 3.7 DFD Level 2 Proses 1.**

## 2. DFD Level 2 Proses 2

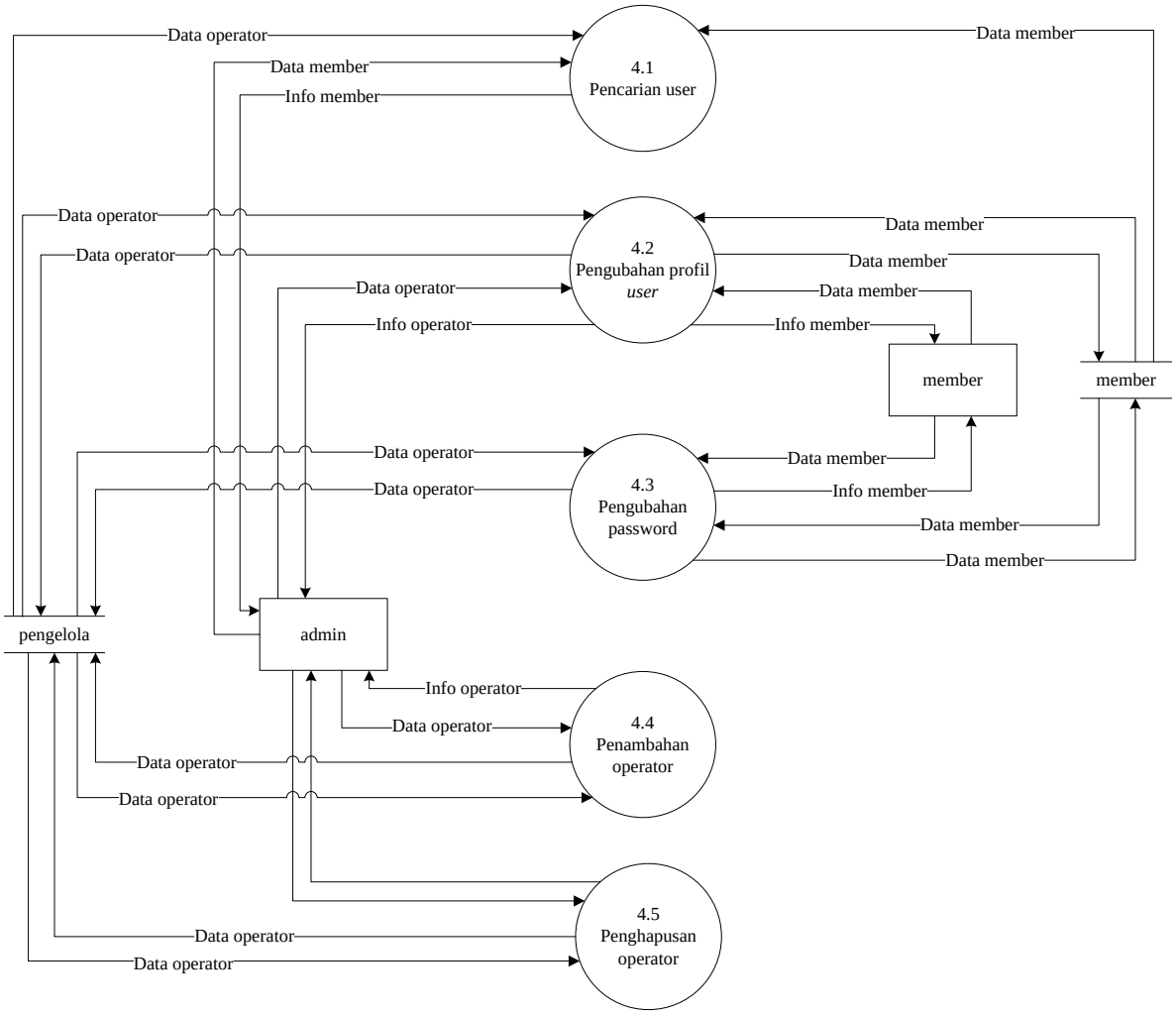
Diagram aliran data pada level 2 proses 2  login dapat dilihat pada Gambar 3.8.




**Gambar 3.8  DFD Level 2 Proses 2.**

### 3. DFD Level 2 Proses 4

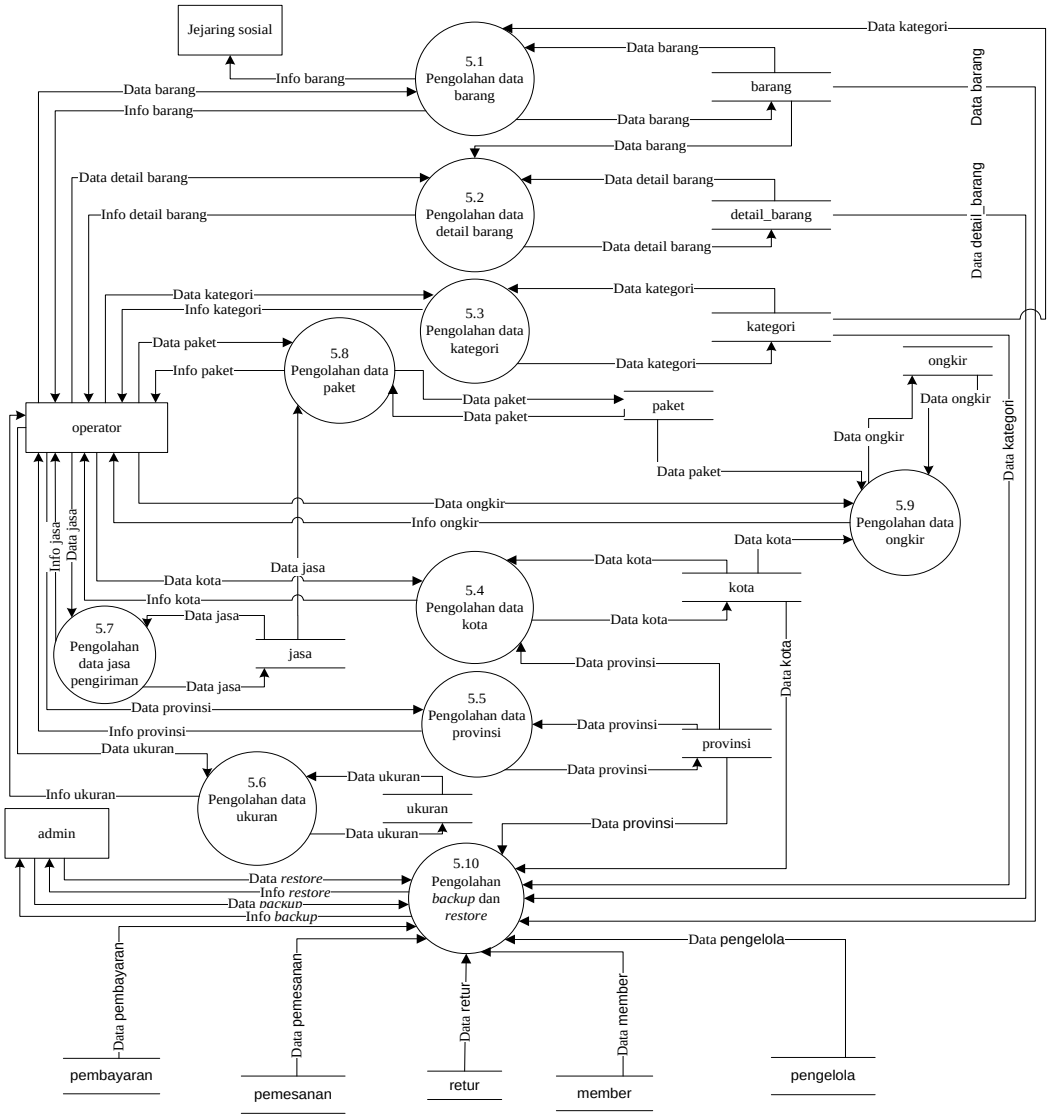
Diagram aliran data pada level 2 proses 4 pengolahan data *user* dapat dilihat pada Gambar 3.9.




**Gambar 3.9 DFD Level 2 Proses 4.**

## 4. DFD Level 2 Proses 5

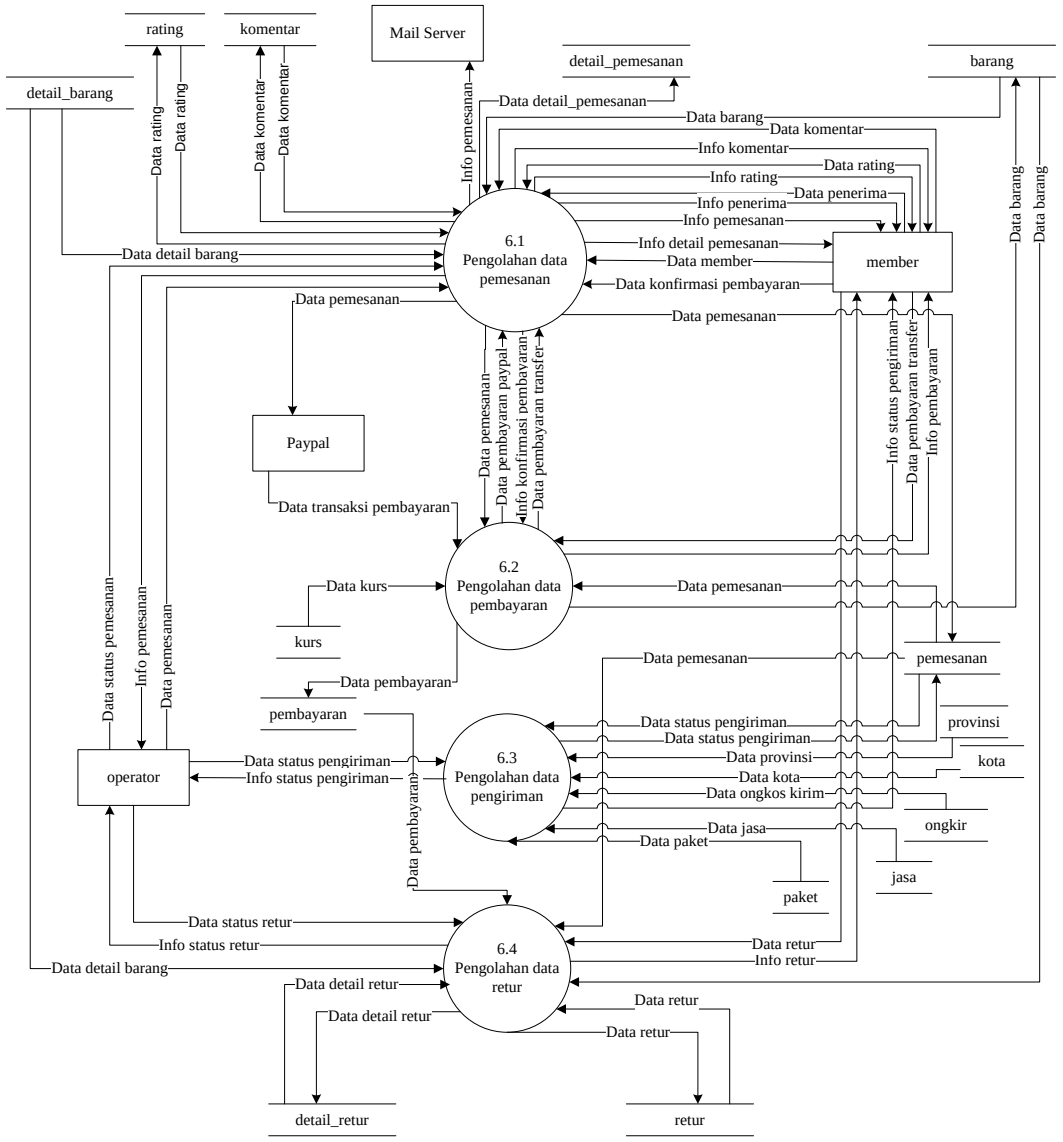
Diagram aliran data pada level 2 proses 5 pengolahan data master dapat dilihat pada Gambar 3.10.




**Gambar 3.10 DFD Level 2 Proses 5.**

## 5. DFD Level 2 Proses 6

Diagram aliran data pada level 2 proses 6 pengolahan data transaksi dapat dilihat pada Gambar 3.11.

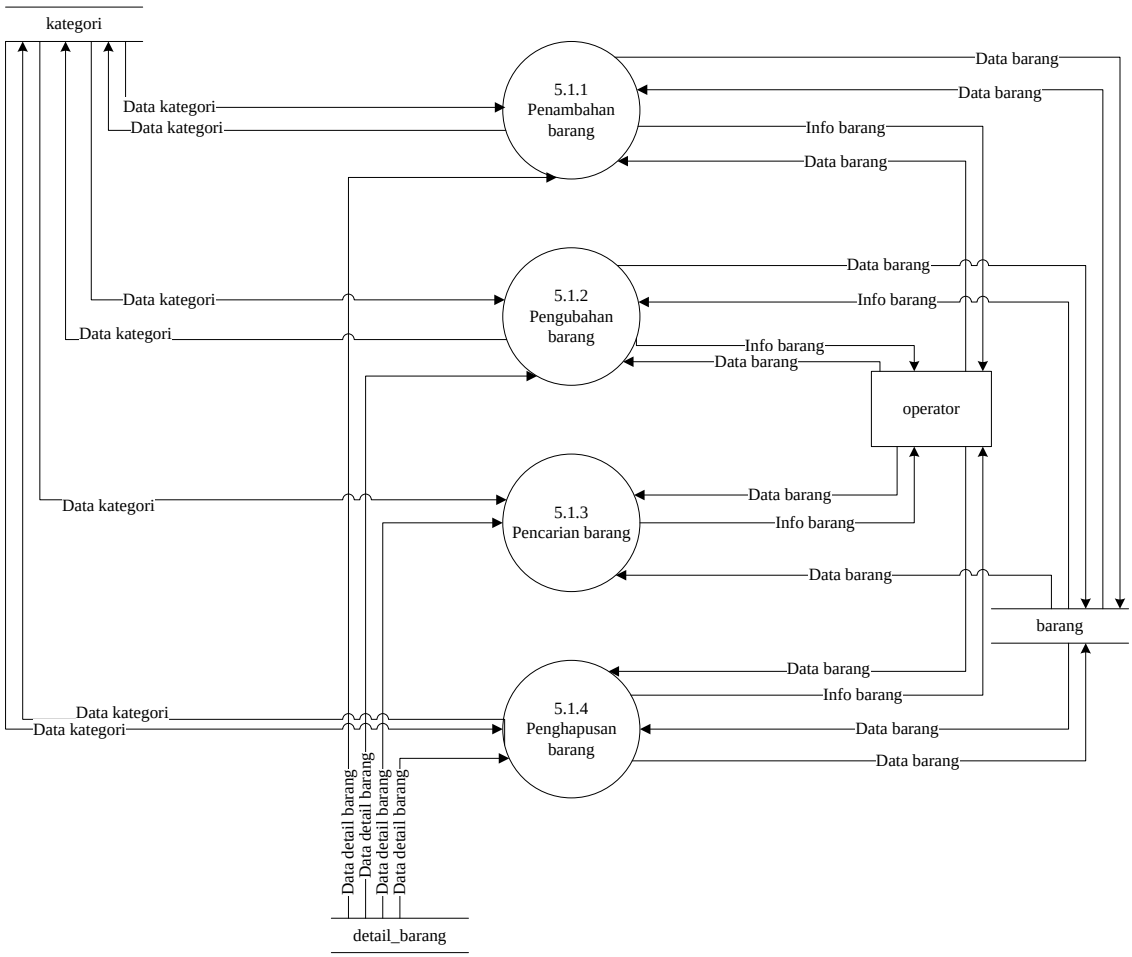



**Gambar 3.11 DFD Level 2 Proses 6.**

### 3.1.7.2.3 DFD level 3

**1. DFD Level 3 Proses 5.1**

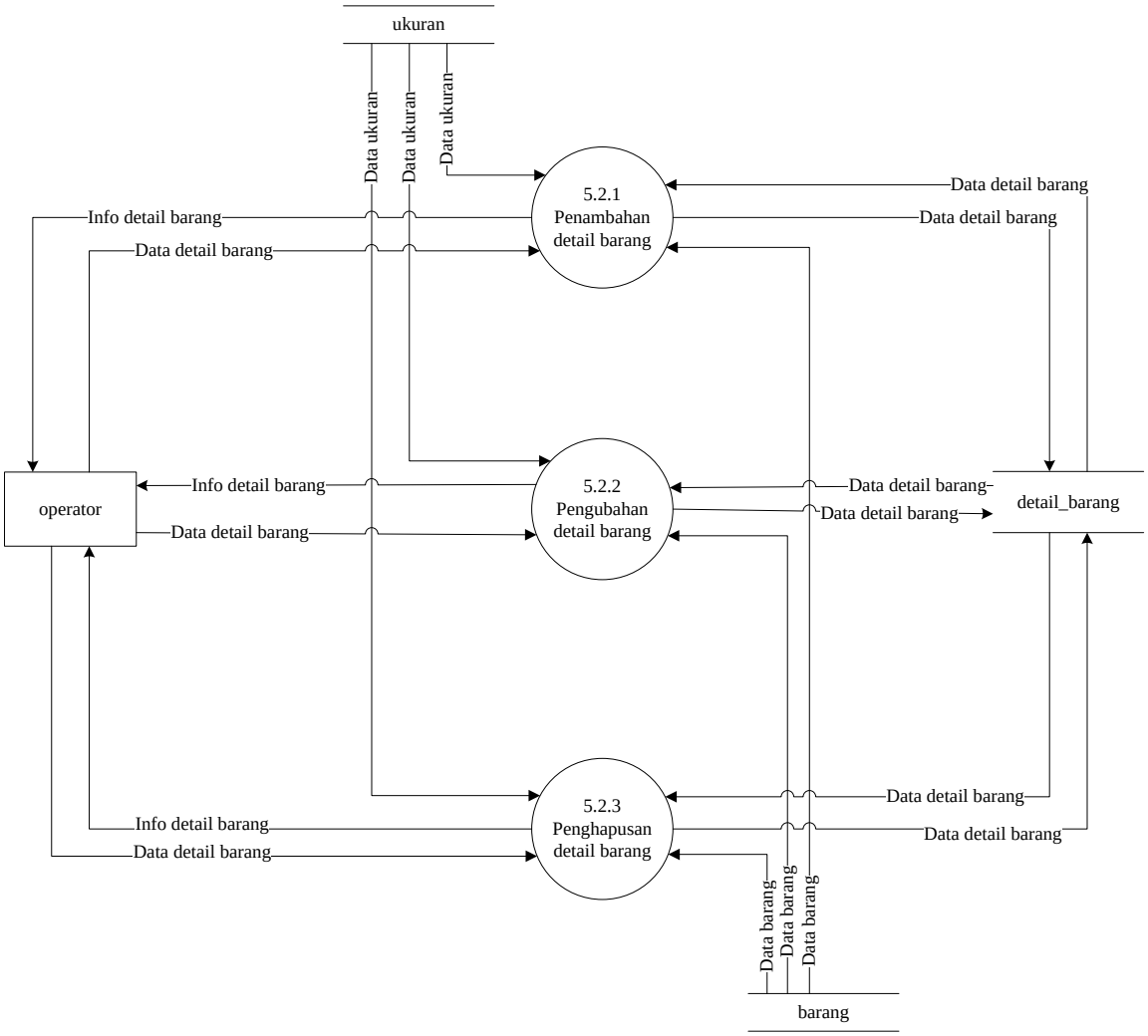
Diagram aliran data pada level 3 proses 5.1 pengolahan data barang dapat dilihat pada Gambar 3.12.




**Gambar 3.12 DFD Level 3 Proses 5.1.**

## 2. DFD Level 3 Proses 5.2

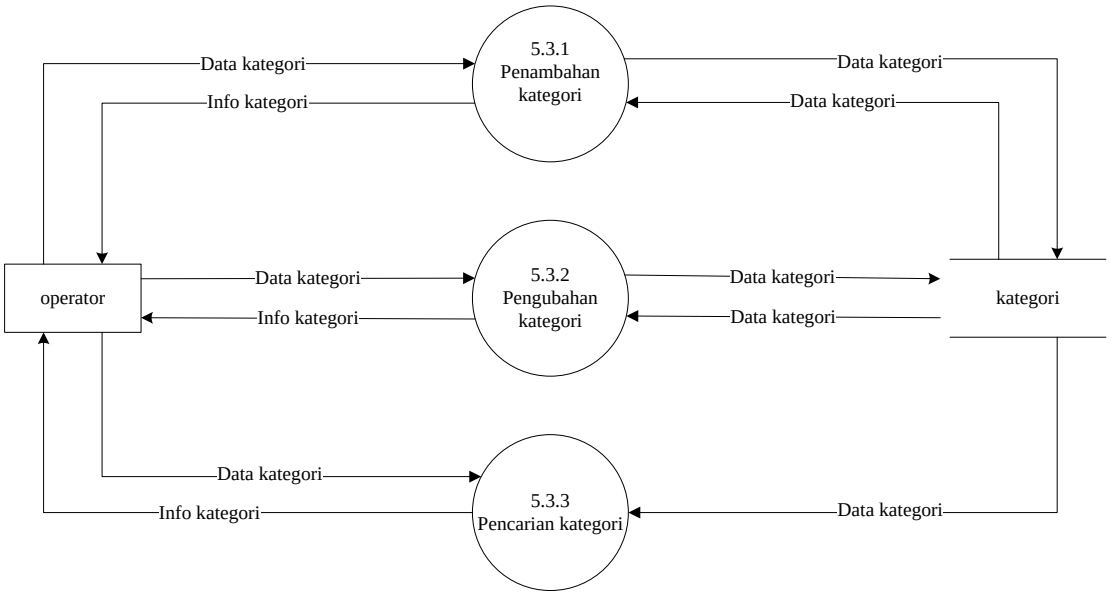
Diagram aliran data pada level 3 proses 5.2 pengolahan data detail barang dapat dilihat pada Gambar 3.13.




**Gambar 3.13 DFD Level 3 Proses 5.2.**

### 3. DFD Level 3 Proses 5.3

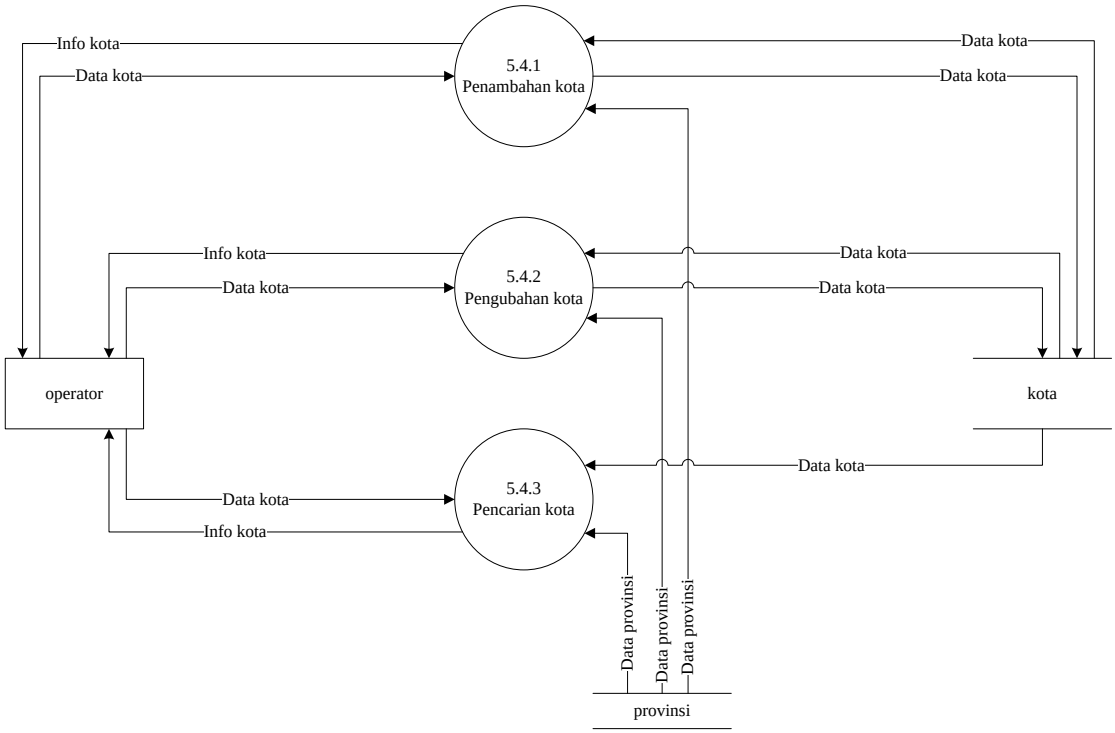
Diagram aliran data pada level 3 proses 5.3 pengolahan data kategori dapat dilihat pada Gambar 3.14.




**Gambar 3.14 DFD Level 3 Proses 5.3.**

## 4. DFD Level 3 Proses 5.4

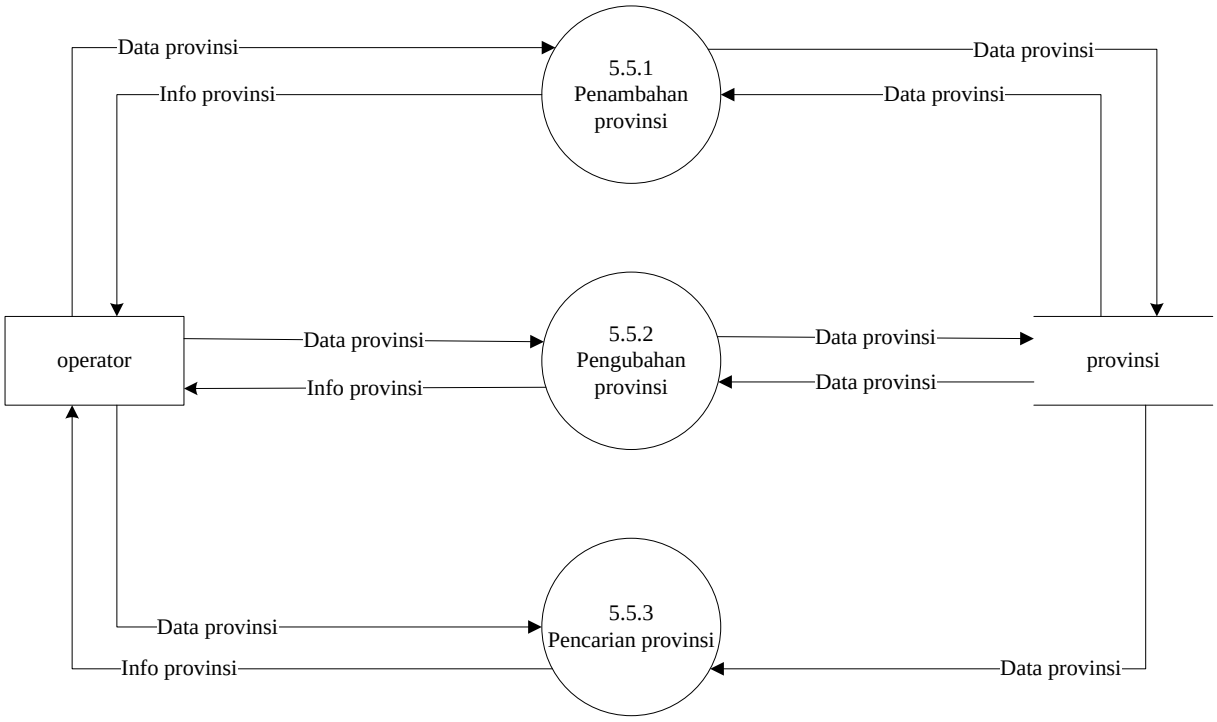
Diagram aliran data pada level 3 proses 5.4 pengolahan data kota dapat dilihat pada Gambar 3.15.




**Gambar 3.15 DFD Level 3 Proses 5.4.**

## 5. DFD Level 3 Proses 5.5

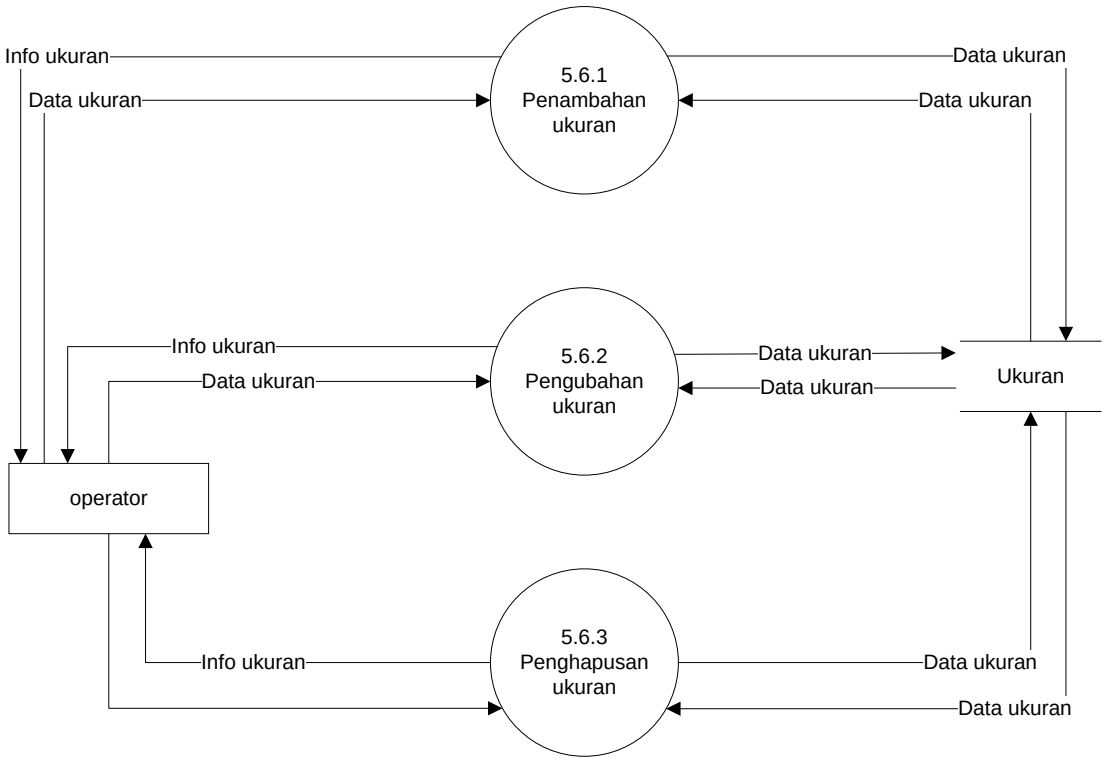
Diagram aliran data pada level 3 proses 5.5 pengolahan data provinsi dapat dilihat pada Gambar 3.16.




**Gambar 3.16 DFD Level 3 Proses 5.5.**

## 6. DFD Level 3 Proses 5.6

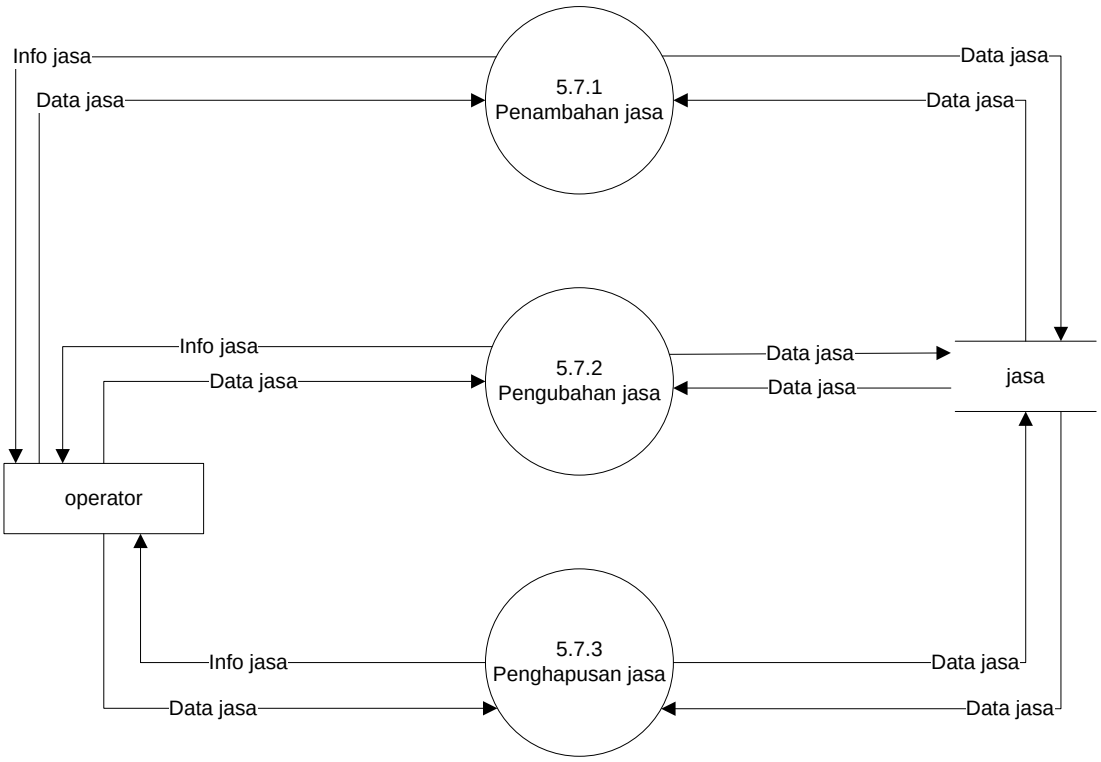
Diagram aliran data pada level 3 proses 5.6 pengolahan data ukuran dapat dilihat pada Gambar 3.17.




**Gambar 3.17 DFD Level 3 Proses 5.6.**

## 7. DFD Level 3 Proses 5.7

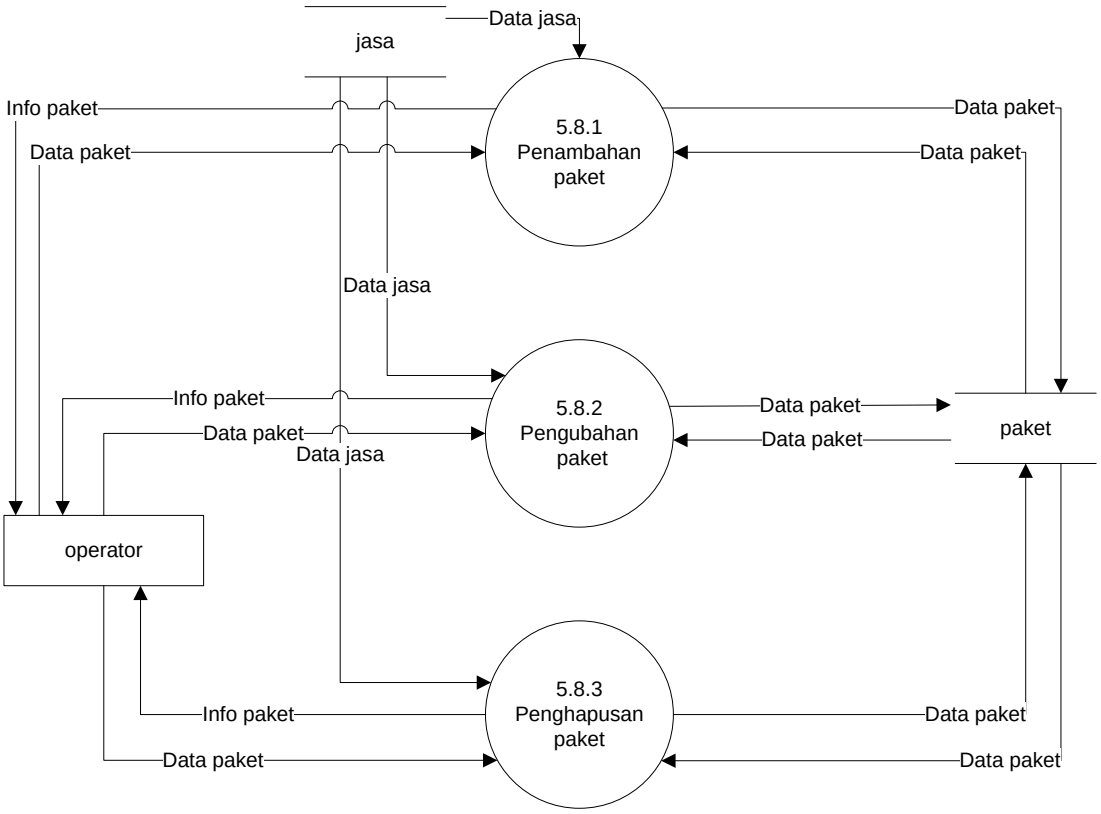
Diagram aliran data pada level 3 proses 5.7 pengolahan data jasa dapat dilihat pada Gambar 3.18.




**Gambar 3.18 DFD Level 3 Proses 5.7.**

## 8. DFD Level 3 Proses 5.8

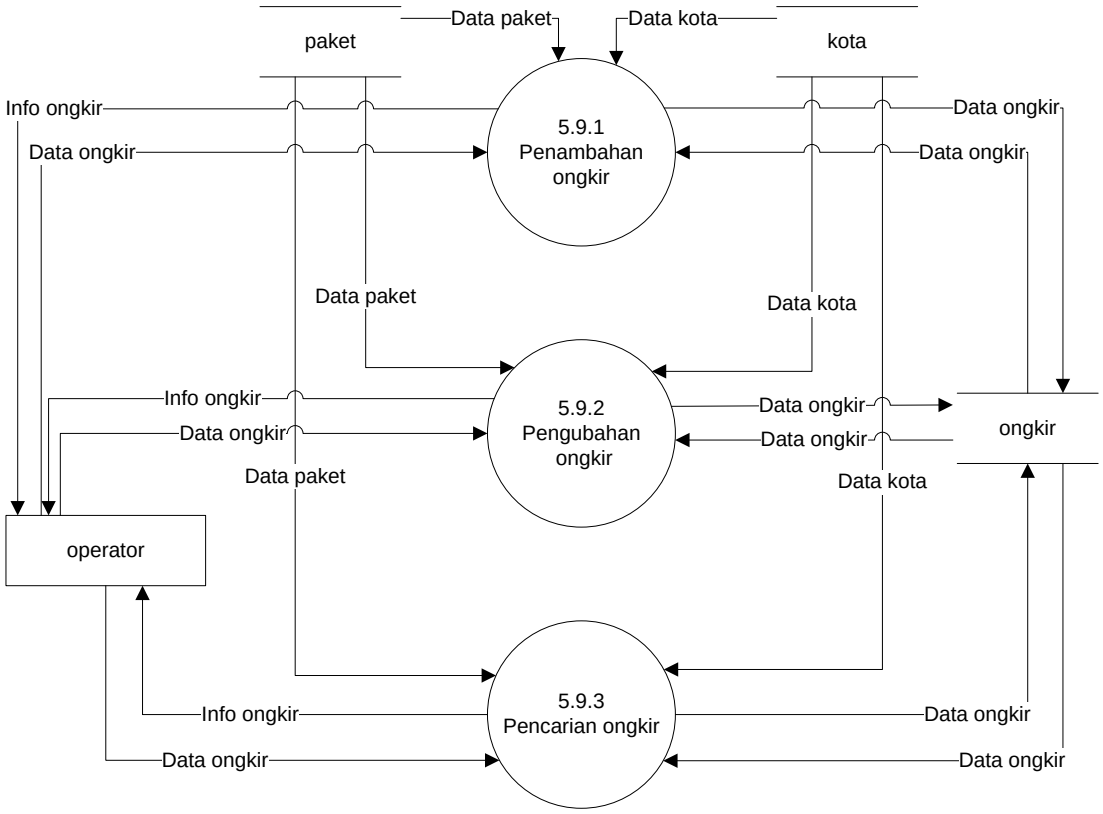
Diagram aliran data pada level 3 proses 5.8 pengolahan data paket dapat dilihat pada Gambar 3.19.




**Gambar 3.19 DFD Level 3 Proses 5.8.**

## 9. DFD Level 3 Proses 5.9

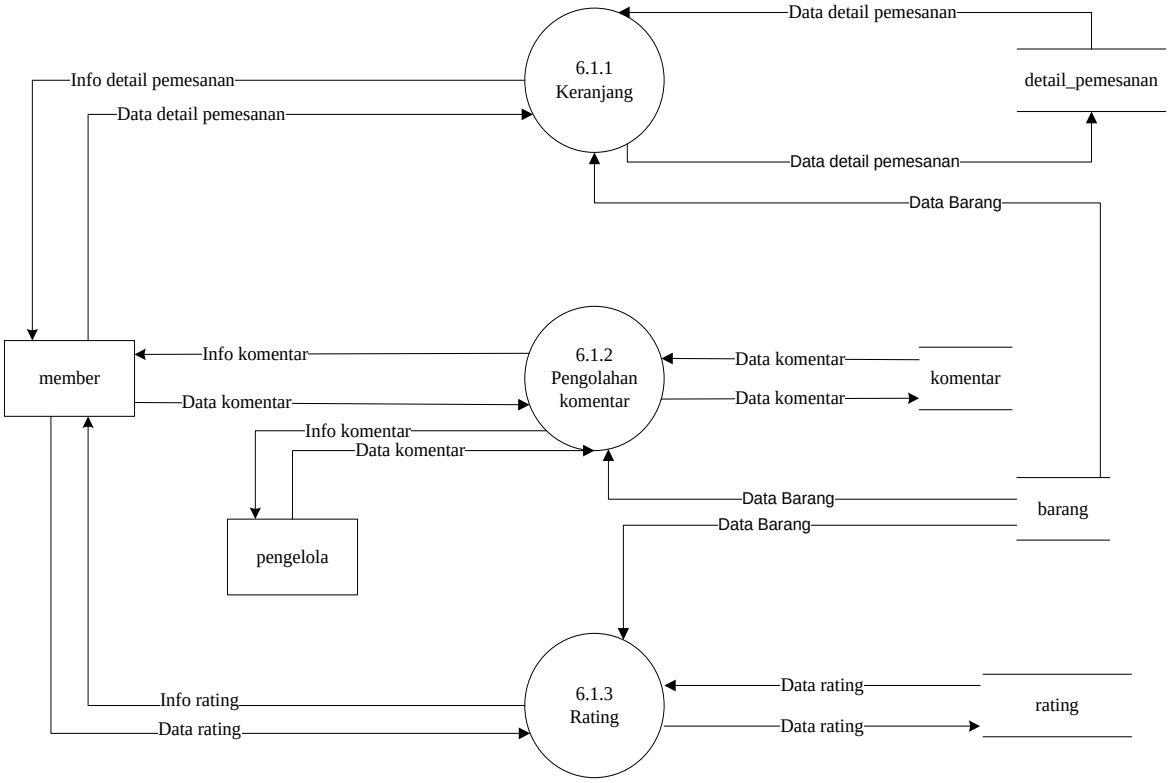
Diagram aliran data pada level 3 proses 5.9 pengolahan data ukuran dapat dilihat pada Gambar 3.20.




**Gambar 3.20 DFD Level 3 Proses 5.9.**

**10. DFD Level 3 Proses 6.1**

Diagram aliran data pada level 3 proses 6.1 dapat dilihat pada Gambar 3.18.




**Gambar 3.21 DFD Level 3 Proses 6.1.**

### 3.1.8 Spesifikasi Proses

Spesifikasi proses merupakan deskripsi dari setiap elemen proses yang terdapat dalam program, yang meliputi nama proses, input, output, dan keterangan dari proses. Spesifikasi proses dari gambaran DFD di atas akan dijelaskan pada Tabel 3.9 di bawah ini.

**Tabel 3.9 Spesifikasi Proses.**

| No | Proses | Keterangan |
|---|---|---|
| 1 | No. Proses | 1 |
| | Nama Proses | Pendaftaran *member* |
| | Asal | Pengunjung non *member* |
| | Masukan | Data registrasi, Data pengaktifan *account*, data *member*, data kota, data provinsi |
| | Keluaran | Info registrasi, Info pengaktifan *account* berhasil, data registrasi *member* |
| | Tujuan | Pengunjung non *member* dan *Mail Server* |
| | Logika Proses | 1. Pengunjung non member memasukan data registrasi<br>2. Sistem melakukan pengecekan apakah data yang dimasukan kosong atau tidak<br>3. Jika data kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong !”<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan apakah email valid atau tidak valid<br>5. Jika email tidak valid, maka akan menampilkan pesan “email sudah terdaftar, gunakan email yang lain”<br>6. Jika email valid, maka sistem akan melakukan pengecekan apakah password dan re-password sama<br>7. Jika tidak sama, maka sistem akan menampilkan pesan “password dan re- |

| | | |
|---|---|---|
| | | password tidak sama”<br>8. Jika sama, maka sistem akan melakukan pengecekan apakah kode captcha benar<br>9. Jika tidak maka, sistem akan menampilkan pesan “Kode captcha salah !”<br>10. Jika ya, maka sistem akan menampilkan pesan “Konfirmasi pengaktifan akun sudah dikirim ke email anda” |
| | | |
| 2 | No. Proses | 1.1 |
| | Nama Proses | *Valid*asi Registrasi |
| | Asal | Pengunjung non *member* |
| | Masukan | Data registrasi, Data provinsi, Data kota |
| | Keluaran | Info registrasi, Data registrasi *member,* Info registrasi *member* |
| | Tujuan | Pengunjung non *member* dan *Mail Server* |
| | Logika Proses | 1. Sistem melakukan pengecekan apakah data yang di masukan kosong atau tidak<br>2. Jika data kosong, maka akan muncul pesan “Data tidak boleh kosong !”<br>3. Jika data tidak kosong, maka sistem akan melakukan pengecekan apakah *email valid* atau tidak *valid*<br>4. Jika email tidak *valid*, maka akan muncul pesan “email sudah terdaftar, gunakan email yang lain !”<br>5. Jika *email valid*, maka sistem akan melakukan pengecekan apakah password dan re-password sama<br>6. Jika tidak sama, maka sistem akan menampilkan pesan “password dan re-password tidak sama !”<br>7. Jika sama, maka sistem akan melakukan pengecekan apakah kode captcha benar<br>8. Jika tidak, maka sistem akan menampilkan pesan “Kode captcha salah !”<br>9. Jika ya, maka sistem akan menampilkan pesan “Konfirmasi pengaktifan akun sudah dikirim ke *email* anda” |
| | | |

| | | |
|---|---|---|
| | No. Proses | 1.2 |
| | Nama Proses | Pengacakan Kode *Captcha* |
| | Asal | Captcha |
| | Masukan | Kode *Captcha* |
| | Keluaran | Acak kode *Captcha* |
| | Tujuan | Sistem |
| | Logika Proses | 1. Sistem akan melakukan pengecekan apakah kode captcha benar<br>2. Jika tidak, maka sistem akan menampilkan pesan "Kode captcha salah!"<br>3. Jika ya, maka sistem akan menampilkan pesan "Konfirmasi pengaktifan akun sudah dikirim ke *email* anda" |
| | | |
| 3 | No. Proses | 1.3 |
| | Nama Proses | Pengiriman aktivasi *account* |
| | Asal | Pengunjung non *member* |
| | Masukan | Data pengaktifan *account* dan Data aktivasi *member* |
| | Keluaran | Info pengaktifan *account*, Data aktivasi *member* |
| | Tujuan | Pengunjung |
| | Logika Proses | 1. Sistem akan memberikan data peangaktifan *member* ke *Mail Server*<br>2. Dari *Mail Server* pengunjung akan mengklik link pengaktifan<br>3. Sistem akan mengaktifkan data *member* |
| | | |
| 4 | No. Proses | 2 |
| | Nama Proses | *Login* |
| | Asal | *Member*, Admin, dan Operator |
| | Masukan | Data *login member*, Data *login* admin, Data |

| | | |
|---|---|---|
| | | *login* operator, Data login pengelola |
| | Keluaran | info login admin *invalid*, info login operator *invalid,* info login member *invalid,* Data *login member* dan Data *login* pengelola |
| | Tujuan | *Member*, Admin, dan Operator |
| | Logika Proses | 1. Member, admin atau operator memasukan data *login* kedalam sistem<br>2. Sistem akan melakukan pengecekan apakah data *login valid* atau tidak *valid*<br>3. Jika data *login* tidak *valid*, maka sistem akan menampilkan pesan "password salah atau email belum terdaftar !"<br>4. Jika data *login valid*, maka menuju halaman utama |
| | | |
| 5 | No. Proses | 2.1 |
| | Nama Proses | Verifikasi dan *valid*asi *login* |
| | Asal | Member, admin dan operator |
| | Masukan | Data login admin, data login operator, data login member |
| | Keluaran | Info *login* member *invalid*. Info *login* admin *invalid,* data *login* operator *invalid* |
| | Tujuan | Admin, operator dan *member* |
| | Logika Proses | 1. Sistem akan melakukan pengecekan apakah data *login valid* atau tidak *valid*<br>2. Jika data *login* tidak *valid*, maka sistem akan menampilkan pesan "password salah atau email belum terdaftar !"<br>3. Jika data *login valid*, maka menuju halaman utama |
| | | |
| 6 | No. Proses | 2.2 |
| | Nama Proses | Pengubahan *password* |
| | Asal | *Member* |
| | Masukan | Data lupa *password,* data *password* |

| | | |
|---|---|---|
| | Keluaran | Info *password* |
| | Tujuan | *Mail Server* |
| | Logika Proses | 1. *Member* meminta data *password*<br>2. Sistem akan memberikan data *password* ke *Mail Server* |
| | | |
| 7 | No. Proses | 3 |
| | Nama Proses | Pencarian barang |
| | Asal | *Member,* pengunjung non *member* |
| | Masukan | Data barang |
| | Keluaran | Info barang dan data barang |
| | Tujuan | *Member*, pengunjung non *member* |
| | Logika Proses | 1. *Member* atau Pengunjung non *member* memasukan data barang yang dicari<br>2. Sistem akan melakukan pengecekan data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan “Data tidak ditemukan !”<br>4. Jika data ditemukan, maka sistem akan menampilkan data barang yang dicari |
| | | |
| 8 | No. Proses | 4 |
| | Nama Proses | Pengolahan data *user* |
| | Asal | *Member,* admin |
| | Masukan | Data *member*, Data operator |
| | Keluaran | Info *member,* info operator, data operator dan data *member* |
| | Tujuan | *Member* , admin |
| | Logika Proses | 1. *Member* memilih menu ubah *profile*<br>2. Sistem akan menampilkan form ubah *profile*<br>3. Admin memilih menu cari *member*<br>4. Sistem akan menampilkan data *member* |

| | | |
|---|---|---|
| | | yang dicari<br>5. Admin memilih menu ubah operator<br>6. Sistem akan menampilkan form ubah operator<br>7. Admin memilih menu tambah data operator untuk menambah data operator<br>8. Sistem akan menampilkan form pengisian data operator<br>9. Admin memilih menu hapus data operator untuk menghapus data operator<br>10. Sistem akan menampilkan informasi penghapusan data operator |
| | | |
| 9 | No. Proses | 4.1 |
| | Nama Proses | Pencarian *user* |
| | Asal | Admin |
| | Masukan | Data *member* dan data *member* |
| | Keluaran | Info *member* |
| | Tujuan | Admin |
| | Logika Proses | 1. Admin memasukan data *member* yang dicari<br>2. Jika data tidak ditemukan, maka sistem akan menampilkan pesan "Data tidak ditemukan !"<br>3. Jika data ditemukan, maka sistem akan menampilkan data *member* yang dicari |
| | | |
| 10 | No. Proses | 4.2 |
| | Nama Proses | Pengubahan profil *user* |
| | Asal | *Member*, admin |
| | Masukan | Data operator, data *member* |
| | Keluaran | info operator, info *member*, data *member* dan data operator |
| | Tujuan | *Member*, admin |
| | Logika Proses | 1. *Member* memasukan data ubah profil |

| | | |
|---|---|---|
| | | 2. Sistem menampilkan pesan “Ubah data profil berhasil”<br>3. Admin memilih data operator yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data *valid*, maka sistem akan menampilkan pesan “data sudah terdaftar !”<br>6. Jika *valid*, maka sistem akan menampilkan pesan “Data berhasil diperbaharui” |
| | | |
| 11 | No. Proses | 4.3 |
| | Nama Proses | Pengubahan *password* |
| | Asal | *Member* |
| | Masukan | Data operator, data *member* |
| | Keluaran | Info operator, info *member*, Data operator, data *member* |
| | Tujuan | *Member* |
| | Logika Proses | 1. *Member* memasukan data *password* lama dan data *password* baru kedalam sistem<br>2. Sistem akan melakukan pengecekan data *valid* atau tidak *valid*<br>3. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Password lama dan Re-password lama tidak sama !” , “Password baru dan Re-password baru tidak sama !”, “Password lama salah !”<br>4. Jika data *valid*, maka sistem akan menampilkan pesan “Ganti Password berhasil” |
| | | |
| 12 | No. Proses | 4.4 |
| | Nama Proses | Penambahan operator |
| | Asal | Admin |
| | Masukan | Data operator |
| | Keluaran | Info operator, data operator |

| | | |
|---|---|---|
| | Tujuan | Admin |
| | Logika Proses | 1. Admin memasukan data operator baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong !”<br>4. Jika data tidak kosong, maka sistem skan memerikasa data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar !”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 13 | No. Proses | 4.5 |
| | Nama Proses | Penghapusan operator |
| | Asal | Admin |
| | Masukan | Data operator, data operator |
| | Keluaran | Info operator, data operator |
| | Tujuan | Admin |
| | Logika Proses | 1. Admin memilih data operator yang akan dihapus<br>2. Sistem akan memberikan pesan peringatan “Apakah anda yakin data dihapus?”<br>3. Jika tidak, maka data gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak<br>5. Jika ya, maka sistem akan menampilkan pesan “Data tidak dapat dihapus !”<br>6. Jika tidak, maka sistem akan menampilkan pesan “Data berhasil dihapus” |
| | | |
| 14 | No. Proses | 5 |
| | Nama Proses | Pengolahan data master |
| | Asal | Operator, Admin, jejaring sosial, *database* kategori, *database* ongkir, *database* kota, |

| | | |
|---|---|---|
| | | *database* provinsi, *database* barang, *database* detail_barang, *database* ukuran, *database* ongkir, *database* paket, *database* jasa |
| | Masukan | data provinsi, data kota, data barang, data detail barang, data kategori, data ongkos kirim, data *backup*, data *restore*, data ukuran, data ongkir, data paket, data jasa |
| | Keluaran | info provinsi, info kota, info barang, info detail barang, info kategori, data provinsi, data kota, data barang, data detail barang, data kategori, data ongkos kirim, info *backup*, data *restore* |
| | Tujuan | Operator, Admin, jejaring sosial, *database* kategori, *database* ongkir, *database* kota, *database* provinsi, *database* barang |
| | Logika Proses | 1. Operator memilih menu pengolahan data barang untuk mengolah data barang<br>2. Sistem akan menampilkan form data barang<br>3. Operator memilih menu pengolahan detail barang untuk mengelola data detail barang<br>4. Sistem akan menampilkan form detail barang<br>5. Operator memilih menu pengelohan data kategori untuk mengelola data kategori<br>6. Sistem akan menampilkan form pengolahan data kategori<br>7. Operator memilih menu pengelohan data kota untuk mengelola data kota<br>8. Sistem akan menampilkan form pengolahan data kota<br>9. Operator memilih menu pengelohan data provinsi untuk mengelola data provinsi<br>10. Sistem akan menampilkan form pengolahan data provinsi<br>11. Operator memilih menu pengelohan data ukuran untuk mengelola data ukuran<br>12. Sistem akan menampilkan form pengolahan data ukuran<br>13. Operator memilih menu pengelohan data jasa untuk mengelola data jasa<br>14. Sistem akan menampilkan form pengolahan data jasa<br>15. Operator memilih menu pengelohan data paket untuk mengelola data paket |

| | | |
|---|---|---|
| | | 16. Sistem akan menampilkan form pengolahan data paket<br>17. Operator memilih menu pengelohan data ongkir untuk mengelola data ongkir<br>18. Sistem akan menampilkan form pengolahan data ongkir<br>19. Admin memilih menu *backup* dan *restore*<br>20. Sistem akan menampilkan form *backup* dan *restore* |
| | | |
| 15 | No. Proses | 5.1 |
| | Nama Proses | Pengolahan data barang |
| | Asal | Operator, jejaring sosial |
| | Masukan | Data barang, info barang, data kategori |
| | Keluaran | Info barang, data barang |
| | Tujuan | Operator, jejaring sosial |
| | Logika Proses | 1. Operator memilih menu tambah data barang<br>2. Sistem akan menampilkan form pengisian data barang<br>3. Operator memilih menu ubah data barang<br>4. Sistem akan menampilkan form pengisian ubah data barang<br>5. Operator memilih menu cari data barang<br>6. Sistem akan menampilkan data barang yang dicari<br>7. Operator memilih hapus data barang<br>8. Sistem akan menampilkan informasi penghapusan data barang |
| | | |
| 16 | No. Proses | 5.1.1 |
| | Nama Proses | Penambahan barang |
| | Asal | Operator |
| | Masukan | Data barang, data kategori, data detail barang, data barang |
| | Keluaran | Info barang, data barang |

| | | |
|---|---|---|
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data barang baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong !”<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “data sudah terdaftar !”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 17 | No. Proses | 5.1.2 |
| | Nama Proses | Pengubahan barang |
| | Asal | Operator |
| | Masukan | Data barang, data kategori, data detail barang, data barang |
| | Keluaran | Info barang, data barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data barang yang diubah<br>2. Sistem akan menampilkan data barang yang akan diubah<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar !”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil diperbaharui” |
| | | |
| 18 | No. Proses | 5.1.3 |
| | Nama Proses | Pencarian barang |

| | | |
|---|---|---|
| | Asal | Operator |
| | Masukan | Data barang, data kategori, data detail barang dan data barang |
| | Keluaran | Info barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data barang yang dicari<br>2. Sistem akan melakukan pengecekan apakah data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan “Data tidak ditemukan !”<br>4. Jika data ditemukan maka sistem akan menampilkan data barang yang dicari |
| | | |
| 19 | No. Proses | 5.1.4 |
| | Nama Proses | Penghapusan barang |
| | Asal | Operator |
| | Masukan | Data barang, data kategori, data detail barang, data barang |
| | Keluaran | Info barang, data barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data barang yang akan dihapus<br>2. Sistem akan memberikan pesan peringatan “Apakah anda yakin data ingin dihapus”<br>3. Jika tidak, maka data gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak<br>5. Jika ya, maka sistem akan menampilkan pesan “data tidak dapat dihapus !”<br>6. Jika tidak, maka sistem akan menampilkan pesan “Data berhasil dihapus” |
| 20 | No. Proses | 5.2 |
| | Nama Proses | Pengolahan data detail barang |

| | | |
|---|---|---|
| | Asal | Operator |
| | Masukan | Data detail barang, data barang, data ukuran |
| | Keluaran | Info detail barang, data detail barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih menu tambah data detail barang untuk menambah data detail barang<br>2. Sistem akan menampilkan form pengisian data detail barang<br>3. Operator memilih menu ubah data detail barang untuk mengubah data detail barang<br>4. Sistem akan menampilkan form ubah data detail barang<br>5. Operator memilih menu hapus data detail barang untuk menghapus data detail barang<br>6. Sistem akan menampilkan informasi penghapusan data detail barang |
| | | |
| 21 | No. Proses | 5.2.1 |
| | Nama Proses | Penambahan detail barang |
| | Asal | Operator |
| | Masukan | Data detail barang, data ukuran, data barang |
| | Keluaran | Info detail barang, data detail barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data detail barang baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong !”<br>4. Jika data tidak kosong, maka sistem akan memeriksa data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar !”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |

| | | |
|---|---|---|
| | | |
| 22 | No. Proses | 5.2.2 |
| | Nama Proses | Pengubahan detail barang |
| | Asal | Operator |
| | Masukan | Data detail barang, data ukuran, data barang |
| | Keluaran | Info detail barang, data detail barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data detail barang yang akan diubah<br>2. Sistem akan menampilkan form ubah data detail barang<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar!"<br>6. Jika *valid*, maka sistem akan menampilkan pesan "Data berhasil diperbaharui" |
| | | |
| 23 | No. Proses | 5.2.3 |
| | Nama Proses | Penghapusan detail barang |
| | Asal | Operator |
| | Masukan | Data detail barang, data ukuran, data barang |
| | Keluaran | Info detail barang, data detail barang |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data detail barang yang akan dihapus<br>2. Sistem akan memberikan pesan peringatan "Apakah anda yakin data ingin dihapus?"<br>3. Jika tidak, maka data detail barang gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak |

| | | |
|---|---|---|
| | | 5. Jika ya, maka sistem akan menampilkan pesan "Data tidak dapat dihapus !"<br>6. Jika tidak, maka sistem akan menampilkan pesan "Data berhasil dihapus" |
| | | |
| 24 | No. Proses | 5.3 |
| | Nama Proses | Pengolahan data kategori |
| | Asal | operator |
| | Masukan | Data kategori |
| | Keluaran | Info kategori, data kategori |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih menu tambah data kategori untuk menambah data kategori<br>2. Sistem akan menampilkan form pengisian data kategori<br>3. Operator memilih menu ubah data kategori untuk mengubah data kategori<br>4. Sistem akan menampilkan form pengisian ubah data kategori<br>5. Operator memilih menu cari data kategori<br>6. Sistem akan menampilkan data kategori yang dicari |
| | | |
| 25 | No. Proses | 5.3.1 |
| | Nama Proses | Penambahan kategori |
| | Asal | Operator |
| | Masukan | Data kategori |
| | Keluaran | Info kategori, data kategori |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data kategori baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan "Data tidak boleh kosong !" |

| | | |
|---|---|---|
| | | 4. Jika data tidak kosong, maka sistem akan memeriksa data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar !"<br>6. Jika data *valid*, maka sistem akan menampilkan pesan "Data berhasil disimpan" |
| | | |
| 26 | No. Proses | 5.3.2 |
| | Nama Proses | Pengubahan kategori |
| | Asal | Operator |
| | Masukan | Data kategori |
| | Keluaran | Info kategori, data kategori |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data kategori yang akan diubah<br>2. Sistem akan menampilkan form ubah detail barang<br>3. Operator memasukan data kategori yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar !"<br>6. Jika data *valid*, maka sistem akan menampilkan pesan "data berhasil diubah" |
| | | |
| 27 | No. Proses | 5.2.3 |
| | Nama Proses | Pencarian kategori |
| | Asal | Operator |
| | Masukan | Data kategori |
| | Keluaran | Info kategori |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data kategori yang dicari |

| | | |
|---|---|---|
| | | 2. Sistem akan melakukan pengecekan data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan “Data tidak ditemukan!”<br>4. Jika data ditemukan, maka sistem akan menampilkan data kategori yang dicari |
| | | |
| 28 | No. Proses | 5.4 |
| | Nama Proses | Pengolahan data kota |
| | Asal | Operator |
| | Masukan | Data kota, data provinsi |
| | Keluaran | Info kota, data kota |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih menu tambah data kota untuk menambah data kota<br>2. Sistem akan menampilkan form pengisian data kota<br>3. Operator memilih menu ubah data kota untuk mengubah data kota<br>4. Sistem akan menampilkan form pengisian ubah data kota<br>5. Operator memilih menu cari data kota<br>6. Sistem akan menampilkan data kota yang dicari |
| | | |
| 29 | No. Proses | 5.4.1 |
| | Nama Proses | Penambahan kota |
| | Asal | Operator |
| | Masukan | Data kota, data provinsi |
| | Keluaran | Info kota, data kota |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data kota baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak |

| | | |
|---|---|---|
| | | 3. Jika kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong !”<br>4. Jika data tidak kosong, maka sistem akan memeriksa data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 30 | No. Proses | 5.4.2 |
| | Nama Proses | Pengubahan kota |
| | Asal | Operator |
| | Masukan | Data kota, data provinsi |
| | Keluaran | Info kota, data kota |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data kota yang akan diubah<br>2. Sistem akan menampilkan form ubah data kota<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “data berhasil diperbaharui” |
| | | |
| 31 | No. Proses | 5.4.3 |
| | Nama Proses | Pencarian kota |
| | Asal | Operator |
| | Masukan | Data kota, data provinsi |
| | Keluaran | Info kota |

| | | |
|---|---|---|
| | Tujuan | operator |
| | Logika Proses | 1. Operator memasukan data kota yang dicari<br>2. Sistem akan melakukan pengecekan data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan “Data tidak ditemukan!”<br>4. Jika data ditemukan, maka sistem akan menampilkan data kota yang dicari |
| | | |
| 32 | No. Proses | 5.5 |
| | Nama Proses | Pengolahan data provinsi |
| | Asal | Operator |
| | Masukan | Data provinsi |
| | Keluaran | Info provinsi, data provinsi |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih menu tambah data provinsi untuk menambah data provinsi<br>2. Sistem akan menampilkan form pengisian data provinsi<br>3. Operator memilih menu ubah data provinsi untuk mengubah data provinsi<br>4. Sistem akan menampilkan form pengisian ubah data provinsi<br>5. Operator memilih menu cari data provinsi<br>6. Sistem akan menampilkan data provinsi yang dicari |
| | | |
| 33 | No. Proses | 5.5.1 |
| | Nama Proses | Penambahan provinsi |
| | Asal | Operator |
| | Masukan | Data provinsi |
| | Keluaran | Info provinsi, data provinsi |
| | Tujuan | Operator |

| | | |
|---|---|---|
| | Logika Proses | 1. Operator memasukan data provinsi baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong!”<br>4. Jika data tidak kosong, maka sistem akan memeriksa data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka data akan disimpan dan sistem akan menampilkan pesan “Pesan berhasil disimpan” |
| | | |
| 34 | No. Proses | 5.5.2 |
| | Nama Proses | Pengubahan provinsi |
| | Asal | Operator |
| | Masukan | Data provinsi |
| | Keluaran | Info provinsi, data provinsi |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data provinsi yang akan diubah<br>2. Sistem akan menampilkan form ubah data provinsi<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil diperbaharui” |
| | | |
| 35 | No. Proses | 5.5.3 |
| | Nama Proses | Pencarian provinsi |
| | Asal | Operator |
| | Masukan | Data provinsi |

| | | |
|---|---|---|
| | Keluaran | Info provinsi |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data provinsi yang dicari<br>2. Sistem akan melakukan pengecekan data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan “Data tidak ditemukan”<br>4. Jika data ditemukan, maka sistem akan menampilkan data provinsi yang dicari |
| | | |
| 36 | No. Proses | 5.6 |
| | Nama Proses | Pengolahan data ukuran |
| | Asal | Operator |
| | Masukan | Data ukuran |
| | Keluaran | Info ukuran, data ukuran |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih menu tambah ukuran untuk menambah data ukuran<br>2. Sistem akan menampilkan form pengisian data ukuran<br>3. Operator memilih menu ubah data ukuran untuk mengubah data ukuran<br>4. Sistem akan menampilkan form ubah data ukuran<br>5. Operator memilih menu hapus data ukuran untuk menghapus data ukuran<br>6. Sistem akan menampilkan informasi penghapusan data ukuran |
| | | |
| 37 | No. Proses | 5.6.1 |
| | Nama Proses | Penambahan ukuran |
| | Asal | Operator |
| | Masukan | Data ukuran |

| | | |
|---|---|---|
| | Keluaran | Info ukuran, data ukuran |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data ukuran baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika data kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong!”<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 38 | No. Proses | 5.6.2 |
| | Nama Proses | Pengubahan ukuran |
| | Asal | Operator |
| | Masukan | Data ukuran |
| | Keluaran | Info ukuran, data ukuran |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data ukuran yang akan diubah<br>2. Sistem akan menampilkan form ubah data ukuran<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil diperbaharui” |
| | | |
| 39 | No. Proses | 5.6.3 |

| | | |
|---|---|---|
| | Nama Proses | Penghapusan ukuran |
| | Asal | Operator |
| | Masukan | Data ukuran |
| | Keluaran | Info ukuran |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data ukuran yang akan dihapus<br>2. Sistem akan memberikan pesan peringatan "Apakah anda yakin data ingin dihapus"<br>3. Jika tidak, maka data gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak<br>5. Jika ya, maka sistem akan menampilkan pesan "Data tidak dapat dihapus"<br>6. Jika tidak, maka sistem akan menampilkan pesan "Data berhasil dihapus" |
| | | |
| 40 | No. Proses | 5.7 |
| | Nama Proses | Pengolahan data jasa |
| | Asal | Operator |
| | Masukan | Data jasa |
| | Keluaran | Info jasa, data jasa |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih menu tambah data jasa untuk menambah data jasa<br>2. Sistem akan menampilkan form pengisian data jasa<br>3. Operator memilih menu ubah data jasa untuk mengubah data jasa<br>4. Sistem akan menampilkan form ubah data jasa<br>5. Operator memilih menu hapus data jasa untuk menghapus data jasa<br>6. Sistem akan menampilkan informasi penghapusan data jasa |

| | | |
|---|---|---|
| | | |
| 41 | No. Proses | 5.7.1 |
| | Nama Proses | Penambahan jasa |
| | Asal | Operator |
| | Masukan | Data jasa |
| | Keluaran | Info jasa, data jasa |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data jasa baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika kosong, maka sistem akan menampilkan pesan "Data tidak boleh kosong!"<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan data *valid* atau tidak<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar!"<br>6. Jika data *valid*, maka sistem akan menampilkan pesan "Data berhasil disimpan" |
| | | |
| 42 | No. Proses | 5.7.2 |
| | Nama Proses | Pengubahan jasa |
| | Asal | Operator |
| | Masukan | Data jasa |
| | Keluaran | Info jasa, data jasa |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data jasa yang akan diubah<br>2. Sistem akan menampilkan form ubah data jasa<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan |

| | | |
|---|---|---|
| | | menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil diubah” |
| | | |
| 43 | No. Proses | 5.7.3 |
| | Nama Proses | Penghapusan jasa |
| | Asal | Operator |
| | Masukan | Data jasa |
| | Keluaran | Info jasa, data jasa |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data jasa yang akan dihapus<br>2. Sistem akan memberikan pesan peringatan “Apakah anda yakin data ingin dihapus”<br>3. Jika tidak, maka data gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak<br>5. Jika ya, maka sistem akan menampilkan pesan “Data tidak dapat dihapus!”<br>6. Jika tidak, maka sistem akan menampilkan pesan “Data berhasil dihapus” |
| | | |
| 44 | No. Proses | 5.8 |
| | Nama Proses | Pengolahan data paket |
| | Asal | Operator |
| | Masukan | Data paket, data jasa |
| | Keluaran | Info paket, data paket |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih menu tambah data paket untuk menambah data paket<br>2. Sistem akan menampilkan form pengisian data paket<br>3. Operator memilih menu ubah data paket untuk mengubah data paket |

| | | |
|---|---|---|
| | | 4. Sistem akan menampilkan form ubah data paket<br>5. Operator memilih menu hapus data paket untuk menghapus data paket<br>6. Sistem akan menampilkan informasi penghapusan data paket |
| | | |
| 45 | No. Proses | 5.8.1 |
| | Nama Proses | Penambahan paket |
| | Asal | Operator |
| | Masukan | Data paket, data jasa |
| | Keluaran | Info paket, data paket |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memasukan data paket baru<br>2. Sistem akan melakukan pengecekan data kosong atau tidak<br>3. Jika kosong maka sistem akan menampilkan pesan “Data tidak boleh kosong!”<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka data paket akan disimpan dan akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 46 | No. Proses | 5.8.2 |
| | Nama Proses | Pengubahan paket |
| | Asal | Operator |
| | Masukan | Data paket, data jasa |
| | Keluaran | Info paket, data paket |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih data paket yang akan diubah |

| | | |
|---|---|---|
| | | 2. Sistem akan menampilkan form ubah data jasa<br>3. Operator memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar"<br>6. Jika data *valid*, maka pengubahan data akan disimpan dan sistem akan menampilkan pesan "data berhasil diubah" |
| | | |
| 47 | No. Proses | 5.8.3 |
| | Nama Proses | Penghapusan paket |
| | Asal | Operator |
| | Masukan | Data paket, data jasa |
| | Keluaran | Info paket, data paket |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memilih data paket yang akan dihapus<br>2. Sistem akan memberi peringatan "Apakah anda yakin data ingin dihapus?"<br>3. Jika tidak, maka data gagal dihapus<br>4. Jika ya, maka sistem akan melakukan pengecekan apakah data sedang dipakai atau tidak<br>5. Jika ya, maka sistem akan menampilkan pesan "Data tidak dapat dihapus!"<br>6. Jika tidak, maka sistem akan menampilkan pesan "Data berhasil dihapus" |
| | | |
| 48 | No. Proses | 5.9 |
| | Nama Proses | Pengolahan data ongkir |
| | Asal | Operator |
| | Masukan | Data ongkir, data paket, data kota |
| | Keluaran | Info ongkir, data ongkir |

| | | |
|---|---|---|
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih menu tambah data ongkir untuk menambah data ongkir<br>2. Sistem akan menampilkan form pengisian data ongkir<br>3. Operator memilih menu ubah data ongkir untuk mengubah data ongkir<br>4. Sistem akan menampilkan form ubah data ongkir<br>5. Operator memilih menu cari data ongkir<br>6. Sistem akan menampilkan data ongkir yang dicari |
| | | |
| 49 | No. Proses | 5.9.1 |
| | Nama Proses | Penambahan ongkir |
| | Asal | Operator |
| | Masukan | Data ongkir, data paket, data kota |
| | Keluaran | Info ongkir, data ongkir |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memasukan data ongkir baru<br>2. Sistem akan melakukan pengecekan apakah data kosong atau tidak<br>3. Jika data kosong, maka sistem akan menampilkan pesan “Data tidak boleh kosong!”<br>4. Jika data tidak kosong, maka sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan “Data sudah terdaftar!”<br>6. Jika data *valid*, maka sistem akan menampilkan pesan “Data berhasil disimpan” |
| | | |
| 50 | No. Proses | 5.9.2 |
| | Nama Proses | Pengubahan ongkir |
| | Asal | Operator |

| | | |
|---|---|---|
| | Masukan | Data ongkir, data paket, data kota |
| | Keluaran | Info ongkir, data ongkir |
| | Tujuan | operator |
| | Logika Proses | 1. Operator memilih data ongkir yang akan diubah<br>2. Sistem akan menampilkan form ubah data ongkir<br>3. Operatot akan memasukan data yang akan diubah<br>4. Sistem akan melakukan pengecekan apakah data *valid* atau tidak *valid*<br>5. Jika data tidak *valid*, maka sistem akan menampilkan pesan "Data sudah terdaftar!"<br>6. Jika data *valid*, maka sistem akan menampilkan pesan "data berhasil diperbaharui" |
| | | |
| 51 | No. Proses | 5.9.3 |
| | Nama Proses | Pencarian ongkir |
| | Asal | Operator |
| | Masukan | Data ongkir, data paket, data kota |
| | Keluaran | Info ongkir, data ongkir |
| | Tujuan | Operator |
| | Logika Proses | 1. Operator memasukan data ongkir yang dicari<br>2. Sistem akan melakukan pengecekan apakah data ada atau tidak<br>3. Jika data tidak ditemukan, maka sistem akan menampilkan pesan "Data tidak ditemukan!"<br>4. Jika data ditemukan maka sistem akan menampilkan data ongkir yang dicari |
| | | |
| 52 | No. Proses | 5.10 |
| | Nama Proses | Pengolahan *backup* dan *restore* |

| | | |
|---|---|---|
| | Asal | Admin |
| | Masukan | Data *backup*, *restore* data barang, data detail barang, data kategori, data kota, data provinsi, data pengelola, data *member*, data retur, data pembayaran, data pemesanan |
| | Keluaran | Info *backup* dan *restore* |
| | Tujuan | Admin |
| | Logika Proses | 1. Admin memilih tabel yang akan di *backup* atau *restore*<br>2. Sistem akan memproses masukan data tabel yang dipilih<br>3. Sistem mengeluarkan hasil *backup* atau *restore* |
| | | |
| 53 | No. Proses | 6 |
| | Nama Proses | Pengolahan data transaksi |
| | Asal | *Member*, operator, retur |
| | Masukan | Data rating, data komentar, data retur, data barang, data pembayaran, data pemesanan, data status bayar, info status pembayaran, data barang, data retur, data detail retur |
| | Keluaran | Info rating, info komentar, info retur, info barang, info pembayaran, info pemesanan, info status bayar, data status pembayaran, data rating, data komentar, data retur, data barang, data pembayaran, data pemesanan, data retur, data detail retur |
| | Tujuan | Operator, *member*, Paypal, *Mail Server* |
| | Logika Proses | 1. *Member* memilih menu produk untuk melakukan transaksi<br>2. Sistem akan menampilkan halaman produk<br>3. *Member* menu *history pembelian* untuk mengetahui informasi pemesanan<br>4. Sistem akan menampilkan pemesanan<br>5. Operator memilih menu pengolahan data pemesanan untuk mengelola data pemesanan<br>6. Sistem akan menampilkan form data |

| | | |
|---|---|---|
| | | pemesanan<br>7. Operator memilih menu pengolahan data pembayaran untuk mengelola data pembayaran<br>8. Sistem akan menampilkan form data pembayaran<br>9. Operator memilih menu pengolahan data pengiriman untuk mengelola data pengiriman<br>10. Sistem akan menampilkan form data pengiriman<br>11. Operator memilih menu pengolahan data retur untuk mengelola data retur |
| | | |
| 54 | No. Proses | 6.1 |
| | Nama Proses | Pengolahan data pemesanan |
| | Asal | Operator, *member* |
| | Masukan | Data pemesanan, data status pemesanan, data konfirmasi pembayaran, data member, data penerima, data detail barang, data barang |
| | Keluaran | Info pemesanan, data detail pemesanan, info penerima, data pemesanan, info detail pemesanan, info penerima, data detail pemesanan, data pemesanan |
| | Tujuan | Operator, *Mail Server*, *member* |
| | Logika Proses | 1. *Member* memilih produk untuk melakukan pemesanan<br>2. Sistem akan menampilkan halaman keranjang belanja<br>3. *Member* memilih menu *history pembelian* untuk mengetahui pemesanan<br>4. Sistem akan menampilkan pemesanan<br>5. Operator memilih menu pemesanan untuk mengolah data pemesanan<br>6. Sistem akan menampilkan form data pemesanan |
| | | |
| 55 | No. Proses | 6.1.1 |

| | | |
|---|---|---|
| | Nama Proses | Keranjang |
| | Asal | *Member* |
| | Masukan | Data detail pemesanan |
| | Keluaran | Info detail pemesanan, Data detail pemesanan |
| | Tujuan | *Member* |
| | Logika Proses | 1. *Member* memasukan jumlah pesanan<br>2. Sistem akan melakukan pengecekan stok tersedia atau tidak<br>3. Jika tidak, maka sistem akan menampilkan pesan “Jumlah barang yang diminta melebihi jumlah stok yang kami miliki. Jumlah stok yang kami miliki bisa dilihat pada kolom “Jumlah” ”<br>4. Jika ya, maka sistem akan melakukan pengecekan *member* sudah login atau belum<br>5. Jika belum, maka sistem akan menampilkan pesan “Anda belum login, silahkan login terlebih dahulu”<br>6. Jika sudah, maka sistem akan menampilkan halaman konfirmasi pemesanan |
| | | |
| 56 | No. Proses | 6.1.2 |
| | Nama Proses | Pengolahan komentar |
| | Asal | Operator, *member* |
| | Masukan | Data komentar |
| | Keluaran | Info komentar, data komentar |
| | Tujuan | Operator, *member* |
| | Logika Proses | 1. *Member* memasukan data komentar kedalam sistem<br>2. Operator akan memeriksa isi komentar.<br>3. Jika layak tampil, maka operator akan menampilkan komentar tersebut<br>4. Jika tidak layak, maka pengelola tidak akan menampilkan komentar tersebut |
| | | |

| | | |
|---|---|---|
| 57 | No. Proses | 6.1.3 |
| | Nama Proses | Rating |
| | Asal | *Member* |
| | Masukan | Data rating |
| | Keluaran | Info rating, data rating |
| | Tujuan | *Member* |
| | Logika Proses | *Member* memberikan rating terhadap suatu barang dengan skala 1-5 |
| | | |
| 58 | No. Proses | 6.2 |
| | Nama Proses | Pengolahan data pembayaran |
| | Asal | *Member*, Paypal |
| | Masukan | Data pembayaran paypal, data pembayaran transfer, data kurs, data pemesanan |
| | Keluaran | Info pembayaran, data pembayaran, data barang |
| | Tujuan | *Member* |
| | Logika Proses | 1. *Member* memilih menu bayar untuk melakukan pembayaran<br>2. Sistem akan menampilkan menu pembayaran *online* dan pembayaran *offline*<br>3. *Member* memilih menu pembayaran *offline* untuk melakukan konfirmasi pembayaran via bank<br>4. *Member* memilih pembayaran *online* untuk melakukan pembayaran via Paypal<br>5. Operator memilih menu pembayaran untuk mem*valid*asi konfirmasi pembayaran<br>6. Sistem akan menampilkan data pembayaran |
| | | |
| 59 | No. Proses | 6.3 |
| | Nama Proses | Pengolahan data pengiriman |
| | Asal | Operator |

| | | |
|---|---|---|
| | Masukan | Data status pengiriman, data status pengiriman, data provinsi, data kota, data ongkos kirim, data paket, data jasa |
| | Keluaran | Info status pengiriman, data status pengiriman |
| | Tujuan | Operator, *member* |
| | Logika Proses | 1. Operator memilih menu pemesanan untuk memeriksa status pembayaran<br>2. Jika belum bayar, maka menu bayar tidak akan tampil<br>3. Jika sudah bayar, maka sistem akan menampilkan menu kirim barang<br>4. Operator memasukan no resi<br>5. Sistem akan melakukan pengecekan apakah no resi *valid* atau tidak *valid*<br>6. Jika no resi tidak *valid*, maka sistem akan menampilkan pesan "No Resi sudah digunakan!"<br>7. Jika *valid,* maka sistem akan menampilkan menu pemesanan<br>8. *Member* memilih menu *history* pembelian untuk mengetahui status pengiriman |
| | | |
| 60 | No. Proses | 6.4 |
| | Nama Proses | Pengolahan data retur |
| | Asal | Operator, *member* |
| | Masukan | Data retur, data status retur, data pembayaran, data pemesanan, data detail barang, data barang, data retur, data detail retur |
| | Keluaran | Info status retur, info retur, data retur, data status retur, |
| | Tujuan | Operator dan *member* |
| | Logika Proses | 1. *Member* memilih pemesanan yang akan diretur<br>2. Sistem akan melakukan pengecekan apakah pemesanan sudah lebih dari 3 hari atau tidak<br>3. Jika sudah, maka sistem tidak akan menampilkan menu retur<br>4. Jika tidak, maka sistem akan menampilkan |

| | | |
|---|---|---|
| | | menu retur<br>5. *Member* memasukan jumlah barang yang akan diretur<br>6. Sistem akan melakukan pengecekan apakah stok sesuai atau tidak<br>7. Jika tidak, maka sistem akan menampilkan pesan "Jumlah barang lebih dari jumlah barang yang dipesan"<br>8. Jika ya, maka sistem akan menampilkan pesan "Terima kasih anda sudah melakukan proses retur, konfirmasi retur akan dikirim ke email anda"<br>9. Operator memilih menu retur untuk melihat data retur yang sudah dimasukan oleh *member*<br>10. Sistem akan menampilkan data retur<br>11. Operator memilih proses retur untuk memproses retur yang dilakukan oleh *member* |
| | | |
| 61 | No. Proses | 7 |
| | Nama Proses | *Tracking* barang |
| | Asal | *Web* JNE dan *member* |
| | Masukan | Data no resi, info no resi |
| | Keluaran | Info no resi, data no resi |
| | Tujuan | *Web* JNE dan *member* |
| | Logika Proses | 1. *Member* memilih no resi<br>2. Sistem akan mengirimkan no resi ke *Web* JNE<br>3. *Web* JNE akan memproses no resi dan mengirim hasilnya ke sistem<br>4. Sistem menampilkan data *tracking* dari *Web* JNE |
| | | |
| 62 | No. Proses | 8 |
| | Nama Proses | Pembuatan laporan |
| | Asal | Admin |

| | | |
|---|---|---|
| | Masukan | Data laporan, data pembayaran, data barang, data pemesanan |
| | Keluaran | Info laporan |
| | Tujuan | Admin |
| | Logika Proses | 1. Admin merekap data laporan<br>2. Sistem akan menampilkan laporan berdasarkan data yang diminta oleh admin |

### 3.1.9 Kamus Data

Kamus data merupakan sebuah daftar yang tersusun dari elemen data yang berhubungan dengan sistem. Kamus data dibuat berdasarkan arus data yang ada *diagram flow data*. Berikut ini adalah kamus data yang di ambil dari *data flow diagram* aplikasi *E-Commerce* yang diusulkan dapat dilihat pada Tabel 3.10.

**Tabel 3.8 Kamus Data.**

| Nama Aliran Data | Data member |
|---|---|
| Digunakan pada | Pengunjung non *member*<br>Proses 1, Proses 1.1 |
| Deskripsi | Berisi data *member* yang digunakan untuk registrasi dan akan disimpan didalam *database*. |
| Struktur data | {email}+{*password}*+{nama}+{alamat}+{no_telp}+<br>{kodepos}+{id_kota}+{status}+{tgl_daftar}+{aktivasi}+<br>{*username*} |
| email<br>*password*<br>nama<br>alamat<br>no_telp | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z]<br>[A-Z\|a-z\|0-9]<br>[0-9] |

| | |
|---|---|
| kodepos<br>id_kota<br>status<br>tgl_daftar<br>aktivasi<br>*user*name | [0-9]<br>[0-9]<br>[A-Z\|a-z]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data *login member* |
| Digunakan pada | Member<br>Proses 2, Proses 2.1, Proses 2.2 |
| Deskripsi | Berisi data *login member*. |
| Struktur data | {email}+{password} |
| email<br>*password* | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data *login* Admin |
| Digunakan pada | Admin<br>Proses 2, Proses 2.1, Proses 2.2 |
| Deskripsi | Berisi data *login* Pengelola |
| Struktur data | {*username*}+{password} |
| *username*<br>*password* | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data *login* Operator |
| Digunakan pada | Operator<br>Proses 2, Proses 2.1, Proses 2.2 |
| Deskripsi | Berisi data *login* Pengelola |
| Struktur data | {*username*}+{password} |
| *username*<br>*password* | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data operator |

| | |
|---|---|
| Digunakan pada | Admin<br>Proses 4, Proses 4.1, Proses 4.3 |
| Deskripsi | Berisi data operator yang akan disimpan didalam *database*. |
| Struktur data | {*username*}+{nama_ pengelola}+{*email*}+<br>{*password*}+{telepon}+{alamat}+{ket} |
| *user*name<br>nama_pengelola<br>email<br>*password*<br>telepon<br>alamat<br>ket | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data barang |
| Digunakan pada | Operator<br>Proses 5, Proses 5.1, Proses 5.1.1, Proses 5.1.2 |
| Deskripsi | Berisi data barang. |
| Struktur data | {id_barang}+{nama}+{detail}+{nama_gbr1}+{nama_gbr2}+<br>{nama_gbr3}+{lihat}+{id_kategori}+{tgl_upload}+<br>{*username*}+{dihapus} |
| id_barang<br>nama<br>detail<br>nama_gbr1<br>nama_gbr2<br>nama_gbr3<br>lihat<br>id_kategori<br>tgl_upload | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[0-1]<br>[A-Z\|a-z\|0-9]<br>[0-9] |

| | |
|---|---|
| *username*<br>dihapus | [A-Z\|a-z\|0-9]<br>[0-1] |
| Nama Aliran Data | Data ketegori |
| Digunakan pada | Operator<br>Proses 5, Proses 5.3, Proses 5.3.1, Proses 5.3.2 |
| Deskripsi | Berisi data kategori. |
| Struktur data | {id_kategori}+{nama_kategori} |
| id_kategori<br>nama_kategori | [A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data kota |
| Digunakan pada | Operator<br>Proses 5, Proses 5.4, Proses 5.4.1, Proses 5.4.2 |
| Deskripsi | Berisi data kota. |
| Struktur data | {id_kota}+{nama_kota}+{ id_provinsi }+{ket } |
| id_kota<br>nama_kota<br>id_provinsi<br>ket | [0-9]<br>[A-Z\|a-z\|0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data provinsi |
| Digunakan pada | Operator<br>Proses 5, Proses 5.5, Proses 5.5.1, Proses 5.5.2 |
| Deskripsi | Berisi data provinsi. |
| Struktur data | {id_provinsi}+{nama_prov} |
| id_provinsi<br>nama_prov | [0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data ukuran |
| Digunakan pada | Operator |

| | |
|---|---|
| | Proses 5, Proses 5.6, Proses 5.6.1, Proses 5.6.2 |
| Deskripsi | Berisi data ukuran. |
| Struktur data | {id_ ukuran }+{ ukuran } |
| id_ ukuran<br>ukuran | [0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data detail pemesanan |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.1, Proses 6.2 |
| Deskripsi | Berisi data detail pemesanan. |
| Struktur data | {id_detail_pemesanan}+{id_detail}+{ukuran}+<br>{harga}+{qty}+{diskon}+{id_pemesanan} |
| id_detail_pemesanan<br>id_detail<br>ukuran<br>harga<br>qty<br>diskon<br>id_pemesanan | [0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[0-9]<br>[0-9]<br>[0-9] |
| Nama Aliran Data | Data pemesanan |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.2, Proses 6.3 |
| Deskripsi | Berisi data pemesanan. |
| Struktur data | {id_pemesanan}+{tgl_beli}+{total_bayar}+{status_bayar}+<br>{status_kirim}+{tgl_kirim}+{penerima}+{alamat_kirim}+<br>{kode_pos}+{id_ongkos}+{paket}+{penerima_barang}+<br>{tgl_terima}+{id_kurs}+{no_resi}+{*email*}+{belum_dibuka}+<br>{lihat_kirim}+{lihat_retur}+<br>{*username*} |

| | |
|---|---|
| id_pemesanan | [0-9] |
| tgl_beli | [0-9] |
| total_bayar | [0-9] |
| status_bayar | [A-Z\|a-z] |
| status_kirim | [A-Z\|a-z] |
| tgl_kirim | [0-9] |
| penerima | [A-Z\|a-z\|0-9] |
| alamat_kirim | [A-Z\|a-z\|0-9] |
| kode_pos | [A-Z\|a-z\|0-9] |
| id_ongkos | [0-9] |
| paket | [A-Z\|a-z\|0-9] |
| penerima_barang | [A-Z\|a-z\|0-9] |
| tgl_terima | [0-9] |
| id_kurs | [0-9] |
| no_resi | [A-Z\|a-z\|0-9] |
| *email* | [A-Z\|a-z\|0-9] |
| belum_dibuka | [0-1] |
| lihat_kirim | [0-1] |
| lihat_retur | [0-1] |
| *username* | [A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data retur |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.5 |
| Deskripsi | Berisi data rating. |
| Struktur data | {id_retur}+{id_pemesanan}+{tgl_retur}+{tgl_kirim}+<br>{status}+{no_resi}+{alasan_*invalid*} |
| id_retur | [0-9] |
| id_pemesanan | [A-Z\|a-z\|0-9] |

| | |
|---|---|
| tgl_retur<br>tgl_kirim<br>status<br>no_resi<br>alasan_*invalid* | [0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data komentar |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.1 |
| Deskripsi | Berisi data komentar. |
| Struktur data | {nomor}+{komentar}+{status}+{*email*}+{id_barang}+<br>{*username*} |
| nomor<br>komentar<br>status<br>*email*<br>id_barang<br>*username* | [0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data rating |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.1 |
| Deskripsi | Berisi data rating. |
| Struktur data | {id_rating}+{id_barang}+{rating}+{email} |
| id_rating<br>id_barang<br>rating<br>email | [0-9]<br>[A-Z\|a-z\|0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9] |
| Nama Aliran Data | Data detail barang |
| Digunakan pada | Operator |

| | |
|---|---|
| | Proses 5, Proses 5.2, Proses 5.2.1, Proses 5.2.2, Proses 5.2.3 |
| Deskripsi | Berisi data detail barang. |
| Struktur data | {id_detail}+{harga}+{stok}+{berat}+{diskon}+<br>{id_barang}+{id_ukuran}+{dihapus} |
| id_detail<br>harga<br>stok<br>berat<br>diskon<br>id_barang<br>id_ukuran<br>dihapus | [A-Z\|a-z\|0-9]<br>[0-9]<br>[0-9]<br>[0-9]<br>[0-9]<br>[A-Z\|a-z\|0-9]<br>[A-Z\|a-z\|0-9]<br>[0-1] |
| Nama Aliran Data | Data pembayaran |
| Digunakan pada | *Member*<br>Proses 6, Proses 6.2, Proses 6.4 |
| Deskripsi | Berisi data pembayaran. |
| Struktur data | {no_pembayaran}+{no_rekening}+{tgl_bayar}+<br>{id_pemesanan}+{dibuka} |
| no_pembayaran<br>no_rekening<br>tgl_bayar<br>id_pemesanan<br>dibuka | [0-9]<br>[0-9]<br>[0-9]<br>[0-9]<br>[0-1] |

## 3.2 Perancangan Sistem

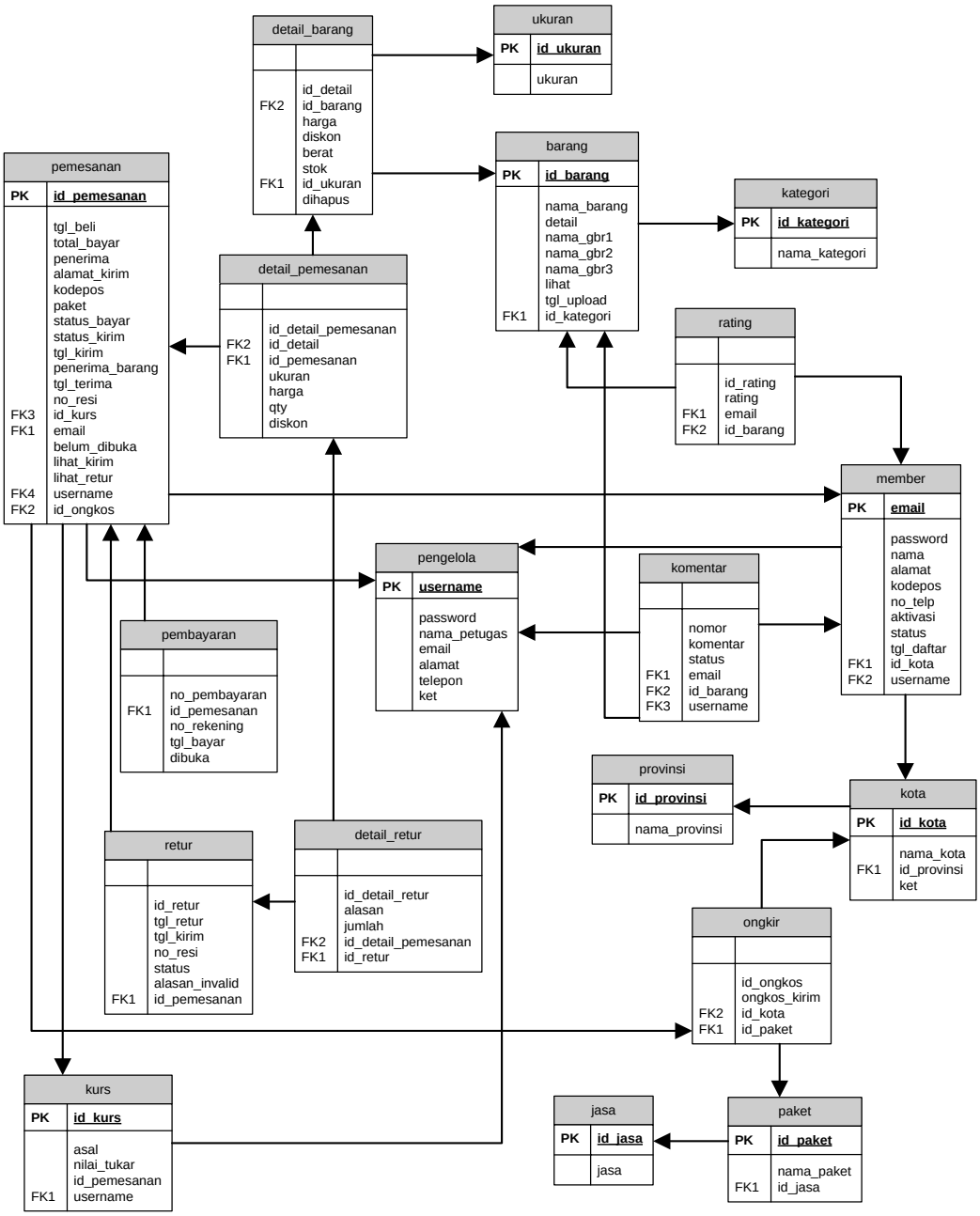
Perancangan sistem merupakan penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh. Tahapan ini meliputi mengonfigurasi komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

### 3.2.1 Perancangan Basis data

Perancangan basis data adalah menciptakan atau merancang kumpulan datayang terhubung dan tersimpan secara bersama-sama.

#### 3.2.1.1 Diagram Relasi

Diagram relasi menggambarkan hubungan antar data, arti data dan batasannya dijelaskan dengan baris dan kolom pada Gambar 3.19




**Gambar 3.22 Diagram Relasi.**

### 3.2.1.2 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam sistem ini adalah sebagai berikut :

**1. Tabel Pengelola**

Tabel ini adalah perancangan untuk tabel pengelola yang akan digunakan pada pembangunan sistem. Struktur tabel pengelola dijelaskan pada Tabel 3.11.

**Tabel 3.11 Tabel Pengelola.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | *username* | varchar | 30 | Primary Key, Not Null |
| 2 | *password* | varchar | 50 | Not Null |
| 3 | nama_petugas | varchar | 30 | Not Null |
| 4 | email | varchar | 30 | Not Null |
| 5 | alamat | varchar | 160 | Not Null |
| 6 | telepon | int | 15 | Not Null |
| 7 | ket | varchar | 10 | Not Null |

**2. Tabel *Member***

Tabel ini adalah perancangan untuk tabel member yang akan digunakan pada pengembangan sistem. Struktur tabel *member* dijelaskan pada Tabel 3.12.

**Tabel 3.12 Tabel *Member*.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | *email* | varchar | 30 | Primary Key, Not Null |
| 2 | *password* | varchar | 50 | Not Null |

| | | | | |
|---|---|---|---|---|
| 3 | nama | varchar | 30 | Not Null |
| 4 | alamat | varchar | 160 | Not Null |
| 5 | kodepos | varchar | 5 | Not Null |
| 6 | no_telp | int | 15 | Not Null |
| 7 | id_kota | int | | Foreign Key, Referance kota(id_kota) |
| 8 | tgl_daftar | datetime | | Not Null |
| 9 | status | varchar | 15 | Not Null |
| 10 | aktivasi | varchar | 100 | Not Null |
| 11 | *username* | varchar | 30 | Foreign Key, Referance pengelola(*username*) |

**3. Tabel Kategori**

Tabel ini berfungsi untuk mengelompokan setiap barang ke dalam kategori. Struktur tabel kategori dijelaskan pada Tabel 3.13.

**Tabel 3.13 Tabel Kategori.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_kategori | int | 10 | Primary Key, Not Null |
| 2 | nama_kategori | varchar | 20 | Not Null |

**4. Tabel Barang**

Tabel ini berfungsi untuk menyimpan data barang. Struktur tabel barang dijelaskan pada Tabel 3.14.

**Tabel 3.14 Tabel Barang.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_barang | int | 10 | Primary Key, Not Null |
| 2 | nama | varchar | 30 | Not Null |
| 3 | detail | varchar | 160 | Not Null |
| 4 | nama_gbr1 | varchar | 30 | Not Null |
| 5 | nama_gbr2 | varchar | 30 | Not Null |
| 6 | nama_gbr3 | varchar | 30 | Not Null |
| 7 | lihat | int | | Not Null, Default 0 |
| 8 | tgl_upload | datetime | | Not Null |
| 9 | id_kategori | varchar | 10 | Foreign Key, Referance kategori(id_kategori) |
| 10 | *username* | varchar | 30 | Foreign Key, Referance pengelola(*username*) |
| 11 | dihapus | enum | (‘0’,’1’) | Not Null, Default 1 |

**5. Tabel Provinsi**

Tabel ini berfungsi untuk menyimpan data provinsi. Struktur tabel provinsi dijelaskan pada Tabel 3.15.

**Tabel 3.15 Tabel Provinsi.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_provinsi | int | | Primary Key, Not Null, Auto Increment |
| 2 | nama_prov | varchar | 30 | Not Null |

**6. Tabel Kota**

Tabel ini berfungsi untuk menyimpan data kota. Struktur tabel kota dijelaskan pada Tabel 3.16.

**Tabel 3.16 Tabel Kota.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_kota | int | | Primary Key, Not Null, Auto Increment |
| 2 | nama_kota | varchar | 30 | Not Null |
| 3 | id_provinsi | int | | Foreign Key, Referance provinsi(id_provinsi) |
| 4 | ket | varchar | 10 | Not Null |

**7. Tabel Kurs**

Tabel ini berfungsi untuk menyimpan data nilai tukar rupiah. Struktur tabel rating dijelaskan pada Tabel 3.17.

**Tabel 3.17 Tabel Kurs.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_kurs | int | | Primary Key, Not Null, Auto Increment |
| 2 | asal | varchar | 10 | Not Null |
| 3 | nilai_tukar | int | 15 | Not Null |
| 4 | id_pemesanan | int | | Foreign Key, Referance pemesanan(id_pemesanan) |
| 5 | *username* | varchar | 30 | Foreign Key, Referance pengelola(*username*) |

**8. Tabel Rating**

Tabel ini berfungsi untuk menyimpan data rating. Struktur tabel rating dijelaskan pada Tabel 3.18.

**Tabel 3.18 Tabel Rating.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_rating | int | | Primary Key, Not Null, Auto Increment |
| 2 | rating | int | 15 | Not Null |
| 3 | *email* | varchar | 30 | Foreign Key, Referance tb_member(*email*) |
| 4 | id_barang | varchar | 10 | Foreign Key, Referance barang(id_barang) |

**9. Tabel Komentar**

Tabel ini berfungsi untuk menyimpan data komentar. Struktur tabel komentar dijelaskan pada Tabel 3.19.

**Tabel 3.19 Tabel Komentar.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | nomor | int | | Primary Key, Not Null, Auto Increment |
| 2 | komentar | varchar | 160 | Not Null |
| 4 | status | varchar | 30 | Not Null |
| 3 | email | varchar | 30 | Foreign Key, Referance member(email) |
| 4 | id_barang | varchar | 10 | Foreign Key, Referance barang(id_barang) |
| 5 | *username* | varchar | 30 | Foreign Key, Referance pengelola(*username*) |

**10. Tabel Ukuran**

Tabel ini berfungsi untuk menyimpan data gambar. Struktur tabel gambar dijelaskan pada Tabel 3.20.

**Tabel 3.20 Tabel Ukuran.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_ukuran | int | | Primary Key, Not Null, Auto Increment |
| 2 | ukuran | varchar | 10 | Not Null |

**11. Tabel Detail Pemesanan**

Tabel ini berfungsi untuk menyimpan data detail pemesanan. Struktur tabel detail pemesanan dijelaskan pada Tabel 3.21.

**Tabel 3.21 Tabel Detail Pemesanan.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_detail_pemesanan | int | | Primary Key, Not Null, Auto Increment |
| 2 | id_detail | varchar | 10 | Foreign Key, Referance detail_barang(id_detail) |
| 3 | ukuran | varchar | 10 | Not Null, Default N/A |
| 4 | harga | int | 10 | Not Null |
| 5 | qty | Int | 5 | Not Null |
| 6 | diskon | float | | Not Null |
| 7 | id_pemesanan | int | 11 | Foreign Key, Referance pemesanan(id_pemesanan) |

**12. Tabel Pembayaran**

Tabel ini berfungsi untuk menyimpan data pembayaran. Struktur tabel pembayaran dijelaskan pada Tabel 3.22.

**Tabel 3.22 Tabel Pembayaran.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | no_pembayaran | int | | Primary Key, Not Null, Auto Increment |
| 2 | no_rekening | int | 20 | Not Null |
| 3 | tgl_bayar | Datetime | | Not Null |
| 4 | id_pemesanan | int | | Foreign Key, Referance pemesanan(id_pemesanan) |
| 5 | dibuka | tinyint | | Not Null, Default 1 |

**13. Tabel Retur**

Tabel ini berfungsi untuk menyimpan data retur. Struktur tabel retur dijelaskan pada Tabel 3.23.

**Tabel 3.23 Tabel Retur.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_retur | int | | Primary Key, Not Null, Auto Increment |
| 2 | id_pemesanan | int | | Foreign Key, Referance pemesanan(id_ pemesanan) |
| 3 | tgl_retur | datetime | | Not Null |
| 4 | tgl_kirim | datetime | | Null |
| 5 | status | varchar | 20 | Not Null, Default Retur |
| 6 | no_resi | varchar | 15 | Not Null, Default 0 |

| | | | | |
|---|---|---|---|---|
| 7 | alasan_*invalid* | varchar | 160 | Not Null, Default Tidak Ada |

**14. Tabel Detail Retur**

Tabel ini berfungsi untuk menyimpan data retur. Struktur tabel retur dijelaskan pada Tabel 3.24.

**Tabel 3.24 Tabel Detail Retur.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_detail_retur | int | | Primary Key, Not Null, Auto Increment |
| 2 | id_retur | int | | Foreign Key, Referance retur(id_retur) |
| 3 | jumlah | int | 15 | Not Null |
| 4 | alasan | varchar | 160 | Null |
| 5 | id_detail_pemesanan | int | | Foreign Key, Referance detail_pemesanan(id_detail_pemesanan) |

**15. Tabel Pemesanan**

Tabel ini berfungsi untuk menyimpan data pemesanan. Struktur tabel pemesanan dijelaskan pada Tabel 3.25.

**Tabel 3.25 Tabel pemesanan.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_pemesanan | int | | Primary Key, Not Null, Auto Increment |
| 2 | tgl_beli | datetime | | Not Null |
| 3 | total_bayar | int | 10 | Not Null |
| 4 | status_bayar | varchar | 20 | Not Null, Default Belum Dibayar |

| | | | | |
|---|---|---|---|---|
| 5 | status_kirim | varchar | 20 | Not Null, Default Belum Dikirim |
| 6 | tgl_kirim | datetime | | Null |
| 7 | penerima | varchar | 20 | Not Null |
| 8 | alamat_kirim | varchar | 160 | Not Null |
| 9 | kodepos | varchar | 10 | Not Null |
| 10 | id_ongkos | int | | Foreign Key, Referance ongkir(id_ongkos) |
| 11 | paket | varchar | 6 | Not Null |
| 12 | penerima_barang | varchar | 30 | Null |
| 13 | tgl_terima | datetime | | Null |
| 14 | id_kurs | int | 11 | Foregn Key, Referance kurs(id_kurs) |
| 15 | no_resi | varchar | 25 | Null |
| 16 | *email* | varchar | 30 | Foregn Key, Referance *member*(*email*) |
| 17 | belum_dibuka | tinyint | | Not Null, Default 1 |
| 18 | lihat_kirim | tinyint | | Not Null, Default 1 |
| 19 | lihat_retur | tinyint | | Not Null, Default 1 |
| 20 | *username* | varchar | 30 | Foregn Key, Referance pengelola(*username*) |

**16. Tabel Detail Barang**

Tabel ini berfungsi untuk menyimpan data detail barang. Struktur tabel detail barang dijelaskan pada Tabel 3.26.

**Tabel 3.26 Tabel Detail Barang.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_detail | varchar | | Primary Key, Not Null, Auto Incremen |
| 2 | harga | int | 10 | Not Null |
| 3 | stok | int | 5 | Not Null |
| 4 | berat | float | | Not Null, Default 0 |
| 5 | diskon | float | | Not Null, Default 0 |
| 6 | id_barang | varchar | 10 | Foreign Key, Referance barang(id_barang) |
| 7 | id_ukuran | varchar | 10 | Foreign Key, Referance ukuran(id_ ukuran) |
| 8 | dihapus | enum | (‘0’,’1’) | Not Null, Default 1 |

**17. Tabel Paket**

Tabel ini berfungsi untuk menyimpan data paket. Struktur tabel paket dijelaskan pada Tabel 3.27.

**Tabel 3.27 Tabel Paket.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_paket | int | | Primary Key, Not Null, Auto Increment |
| 2 | nama_paket | varchar | 10 | Not Null |
| 3 | id_jasa | varchar | 10 | Foreign Key, Referance jasa(id_ jasa) |

**18. Tabel Ongkir**

Tabel ini berfungsi untuk menyimpan data ongkos kirim. Struktur tabel ongkos kirim dijelaskan pada Tabel 3.28.

**Tabel 3.28 Tabel Ongkir.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_ongkos | int | | Primary Key, Not Null, Auto Increment |
| 2 | ongkos_kirim | int | 10 | Not Null |
| 3 | id_paket | int | | Foreign Key, Referance paket(id_ paket) |
| 4 | id_kota | int | | Foreign Key, Referance kota(id_ kota) |

**19. Tabel Jasa**

Tabel ini berfungsi untuk menyimpan data ongkos kirim. Struktur tabel ongkos kirim dijelaskan pada Tabel 3.29.

**Tabel 3.29 Tabel Jasa.**

| No | Field | Type | Size | Keterangan |
|---|---|---|---|---|
| 1 | id_jasa | int | | Primary Key, Not Null, Auto Increment |
| 2 | jasa | varchar | 10 | Not Null |

### 3.2.2 Perancangan Antarmuka

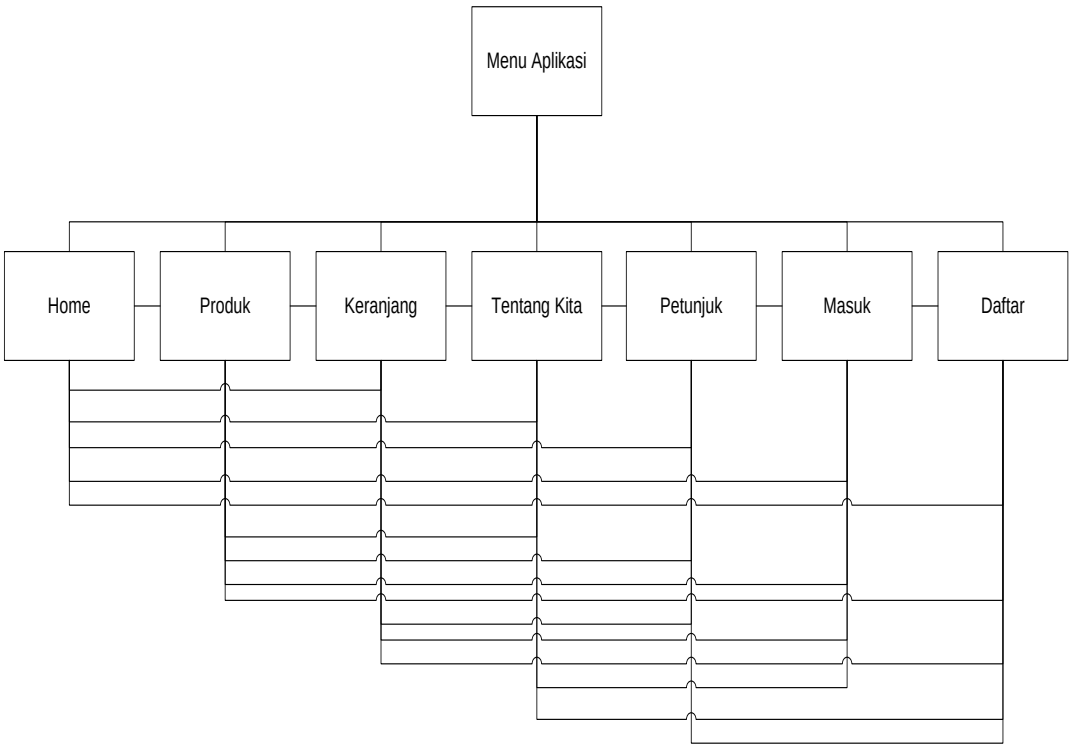
*Interface* atau antar muka merupakan tampilan dari suau program aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara program dengan *user*. Sistem yang akan dibangun diharapkan menyediakan *interface* yang mudah dipahami dan digunakan oleh *user*.

Perancangan interface untuk aplikasi *E-Commerce* di My Joy Ride *Shop* adalah sebagai berikut:

### 3.2.2.1 Perancangan Struktur Menu

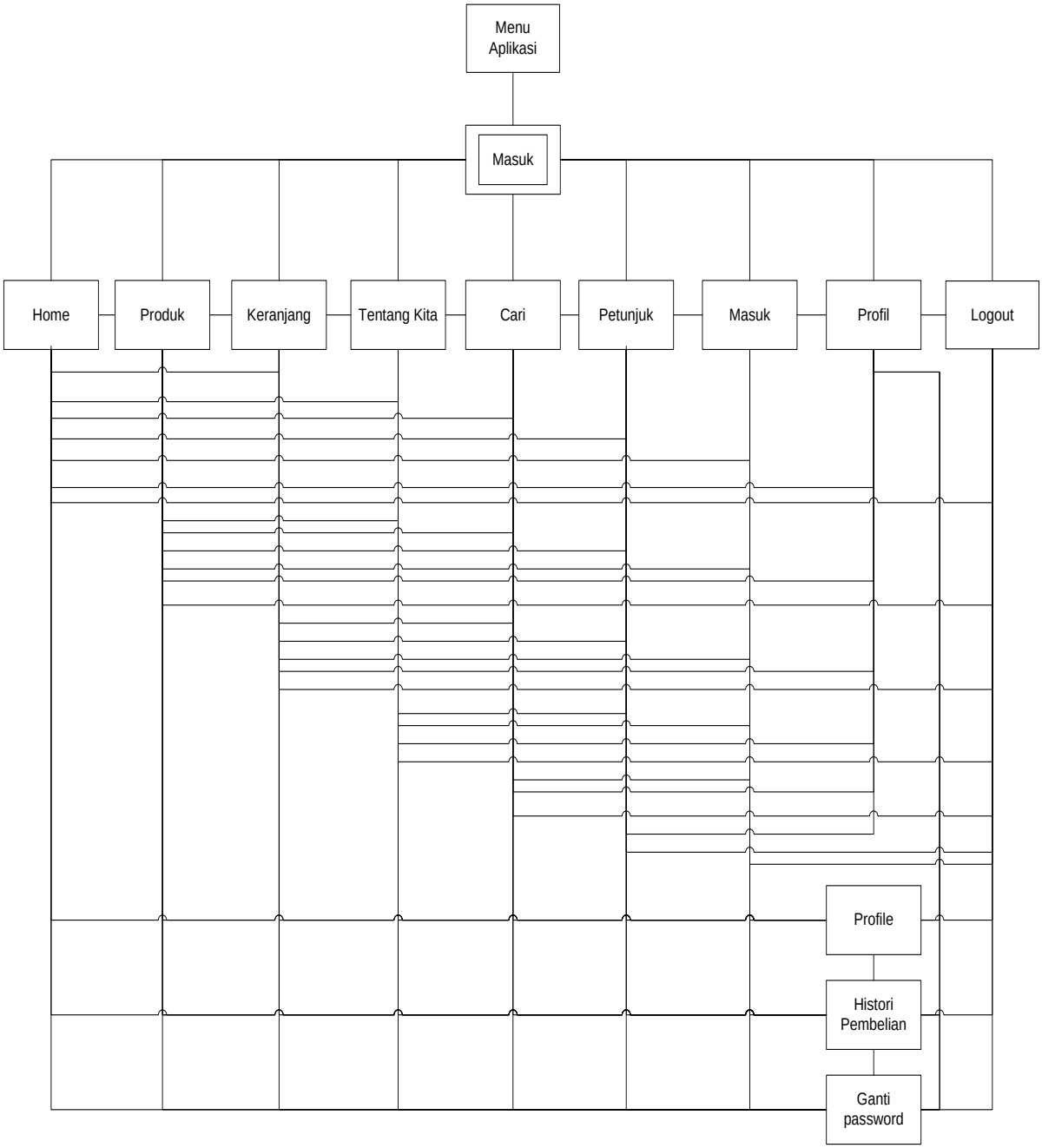
Perancangan Struktur menu menggambarkan keterkaitan setiap menu yang bisa di akses oleh *user* yang digambarkan sebagai berikut:

1. Struktur Menu Pengunjung Non *Member*



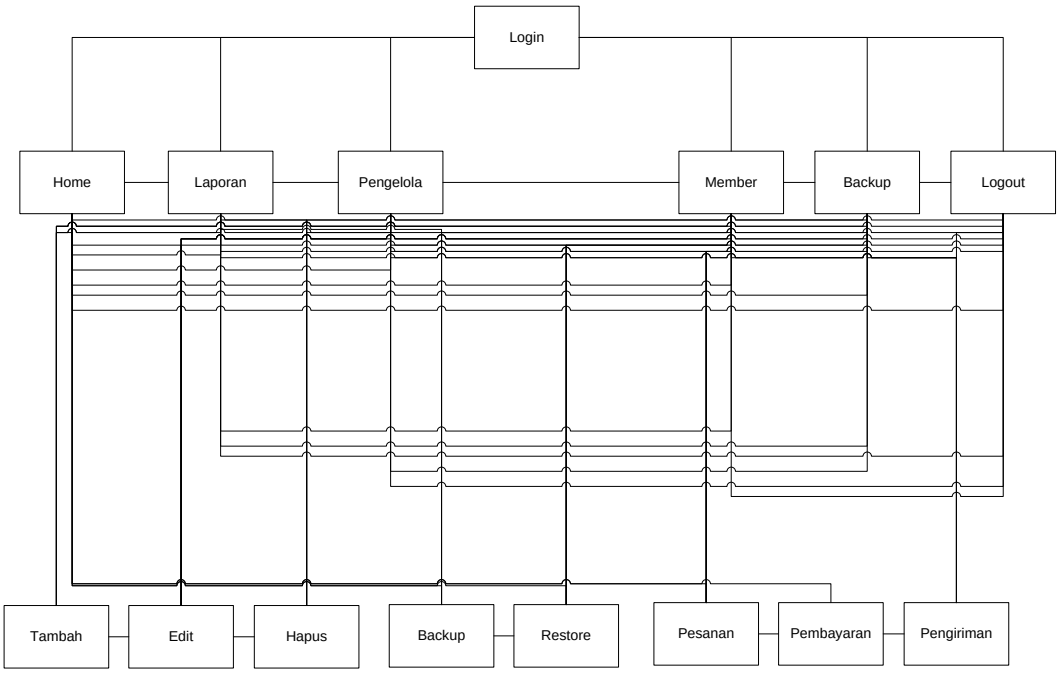

**Gambar 3.23 Struktur Menu Pengunjung Non *Member*.**

2. Struktur Menu Pengunjung *Member*



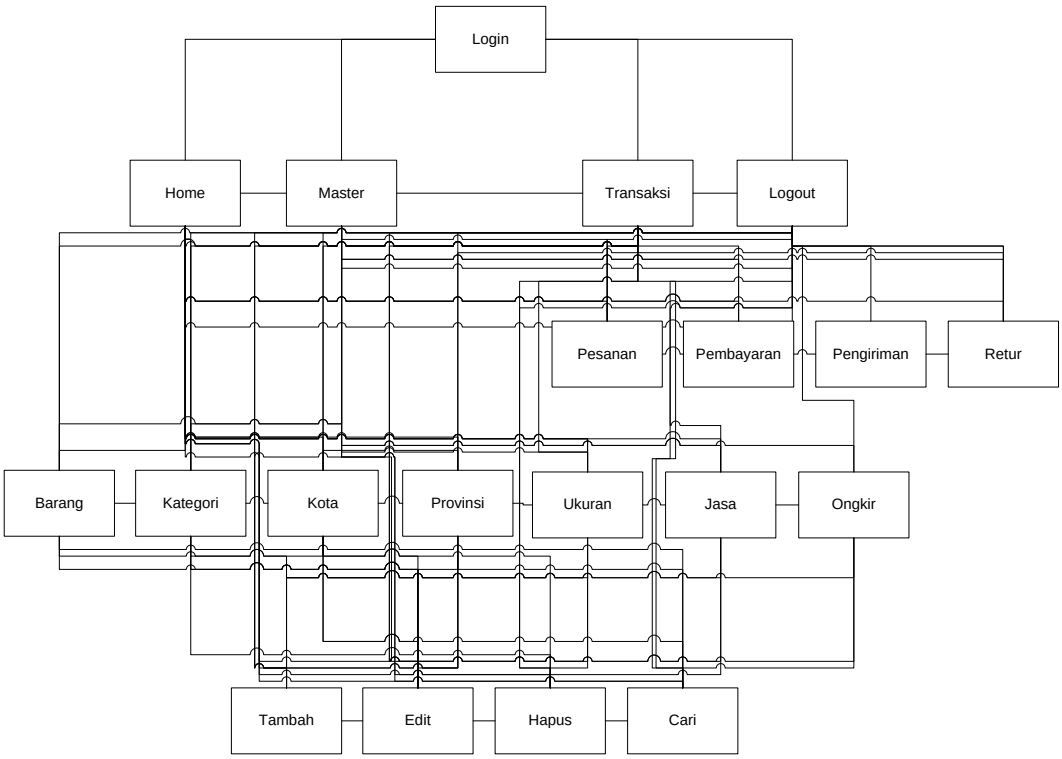

**Gambar 3.24 Struktur Menu Pengunjung *Member*.**

3. Struktur Menu Admin




**Gambar 3.25 Struktur Menu Admin.**

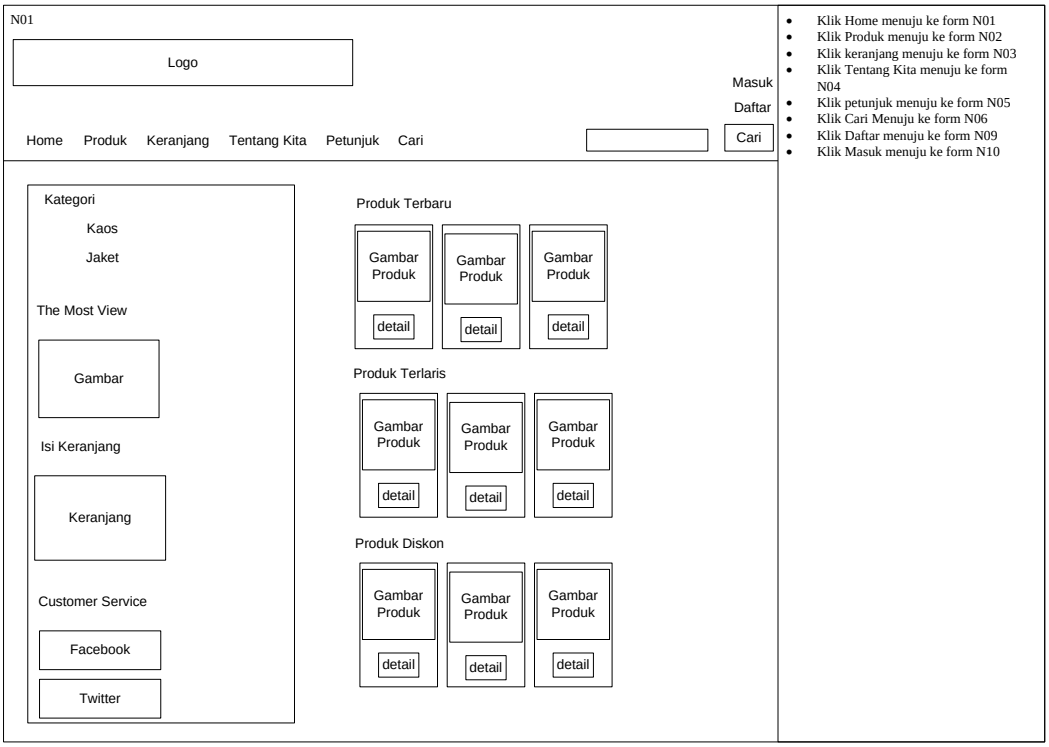
4. Struktur Menu Operator




**Gambar 3.26 Struktur Menu Operator.**

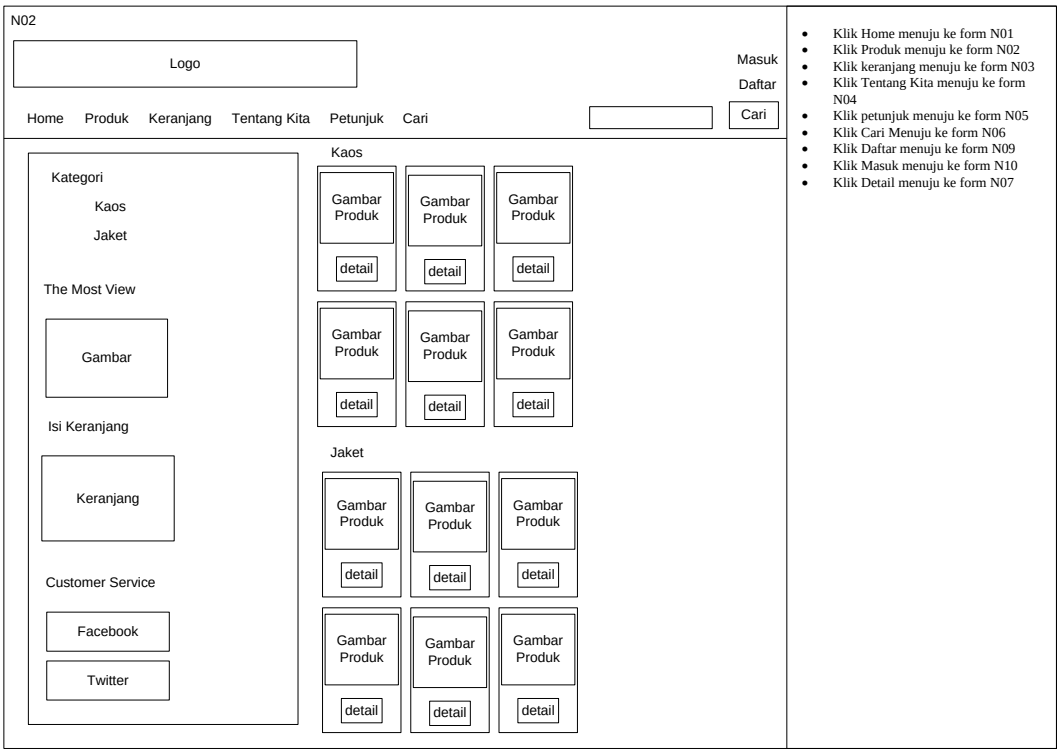
### 3.2.2.2 Perancangan Antarmuka Pengunjung Non *Member*

1. Form *Home*



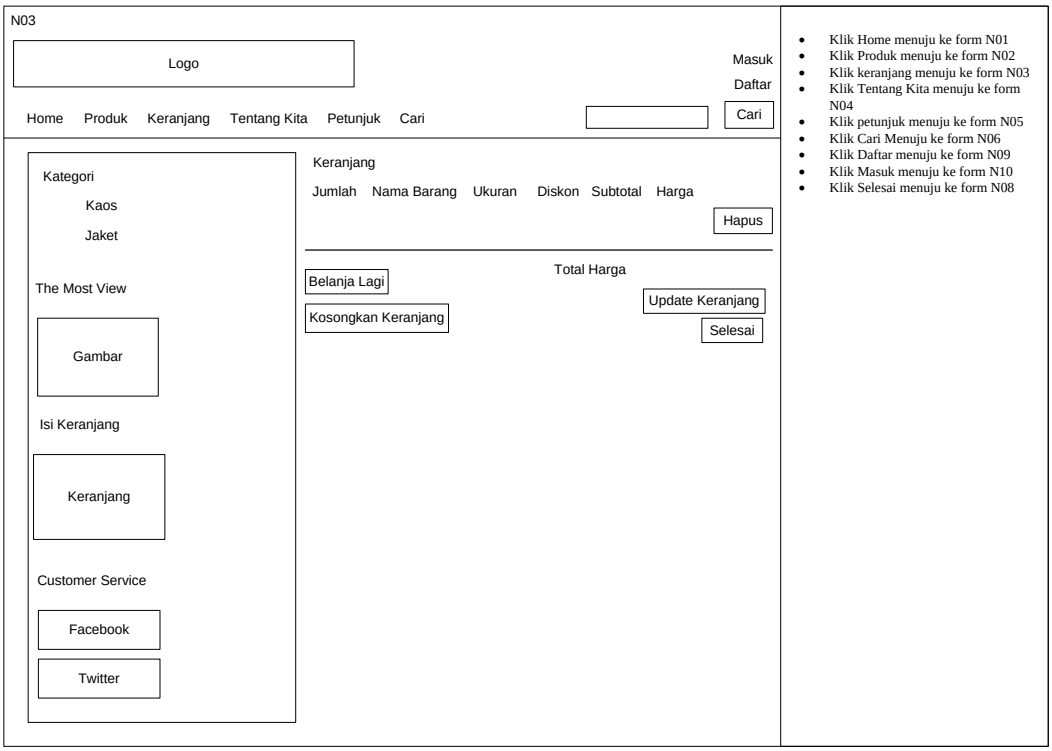

**Gambar 3.27 Form *Home*.**

2. Form Produk



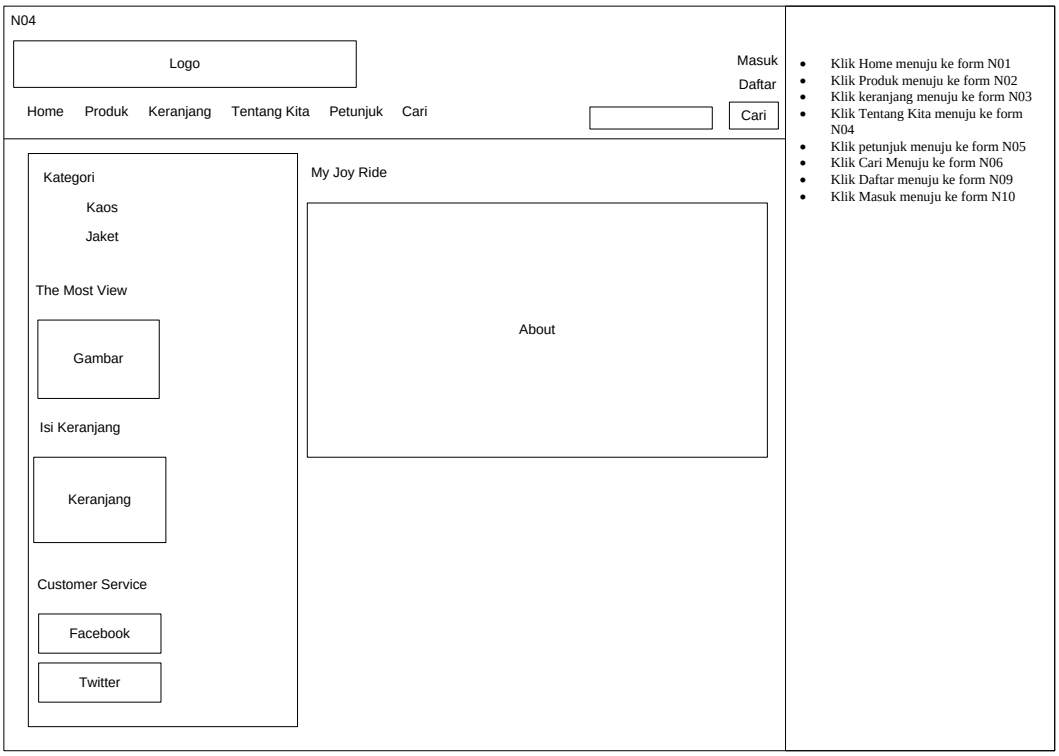

**Gambar 3.28 Form Produk.**

3. Form Keranjang Belanja



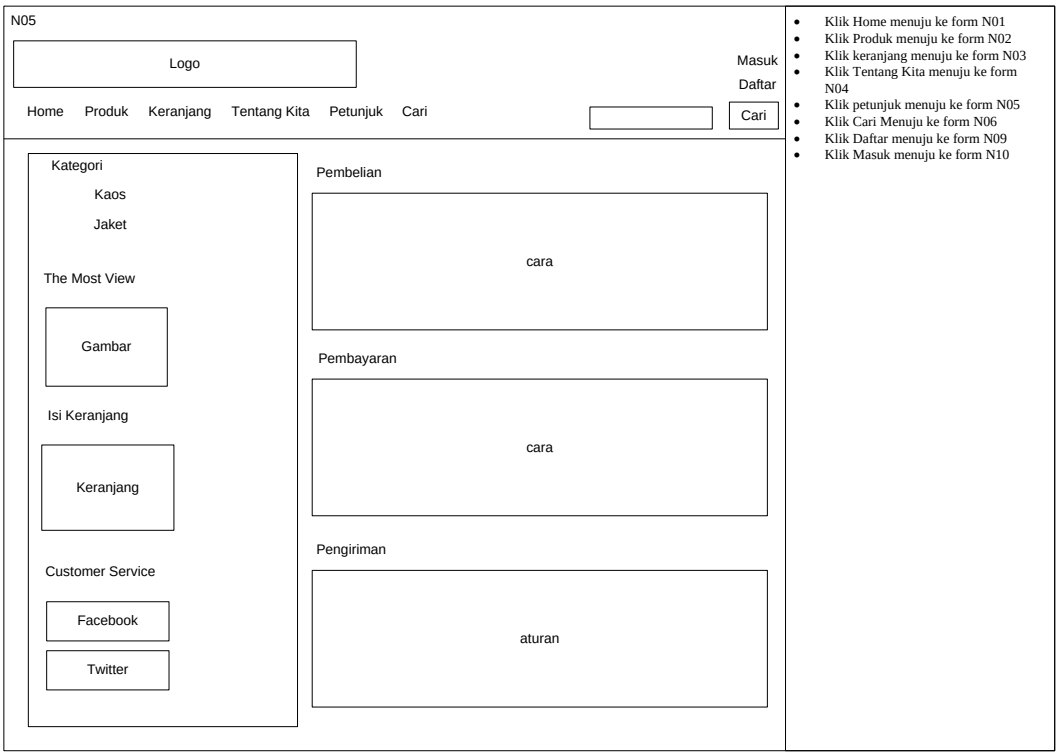

**Gambar 3.29 Form Keranjang Belanja.**

4. Form Tentang Kita



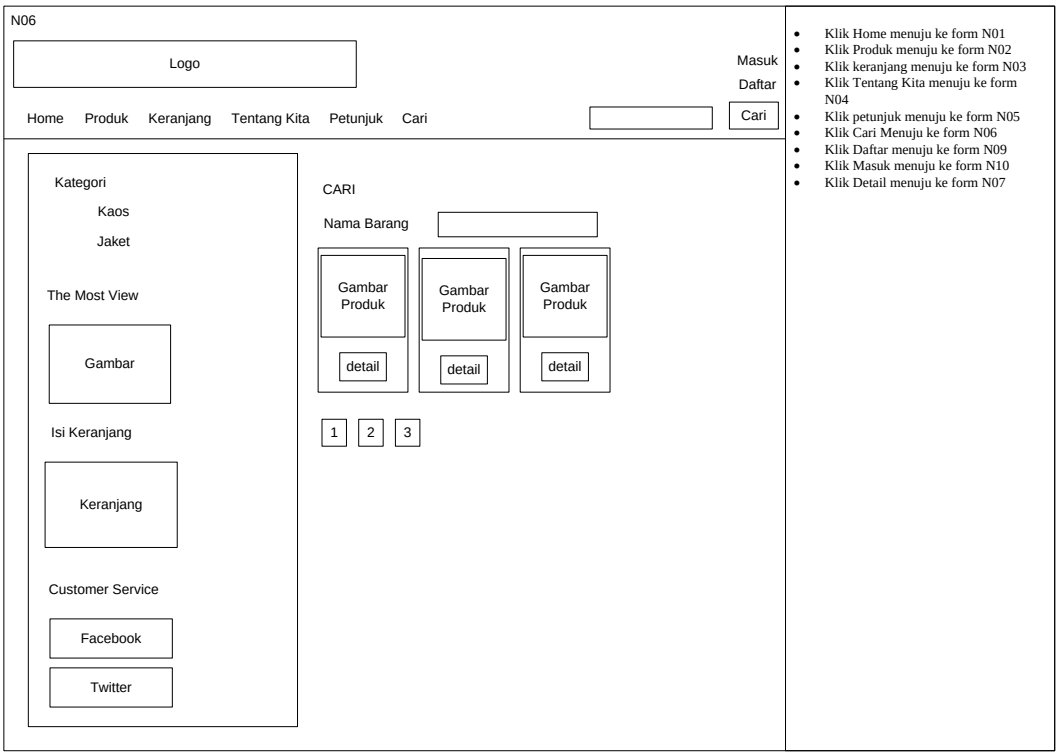

**Gambar 3.30 Form Tentang Kita.**

5. Form Petunjuk



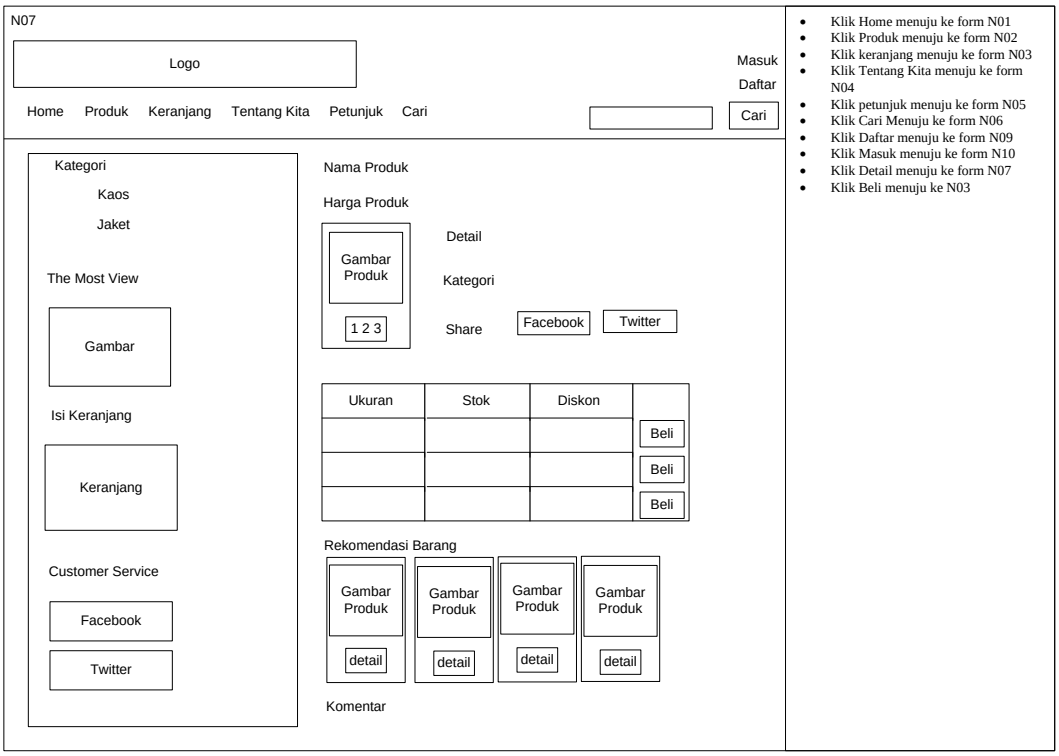

**Gambar 3.31 Form Petunjuk.**

6. Form Cari



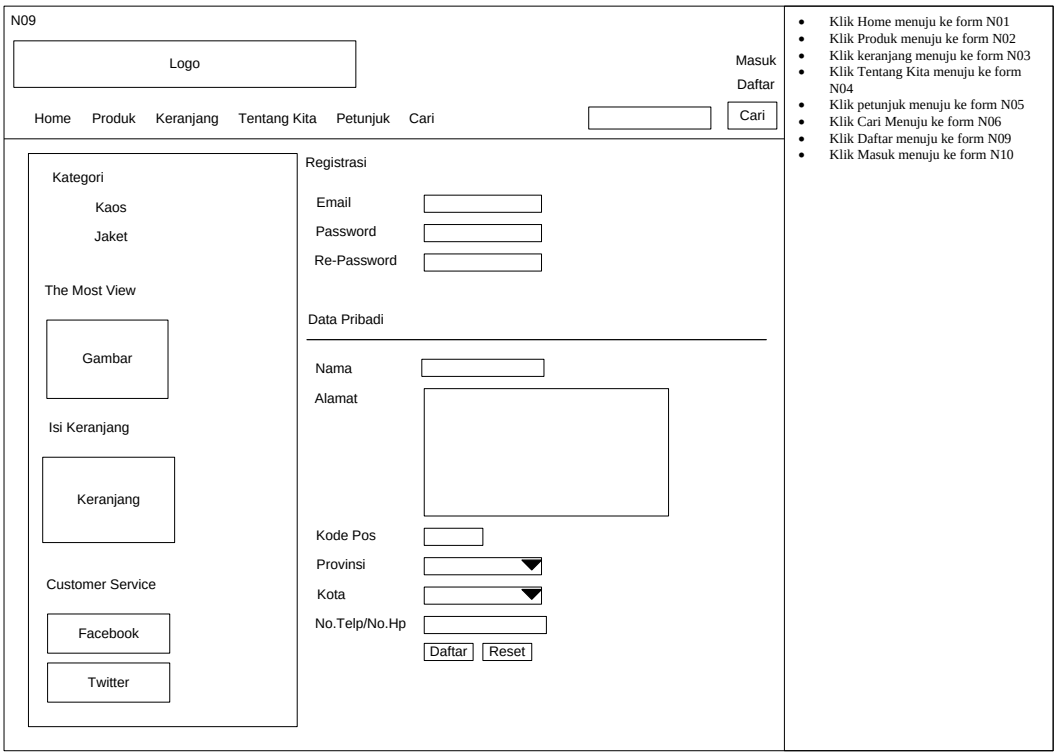

**Gambar 3.32 Form Cari.**

7. Form Detail Produk




**Gambar 3.33 Form Detail Produk.**

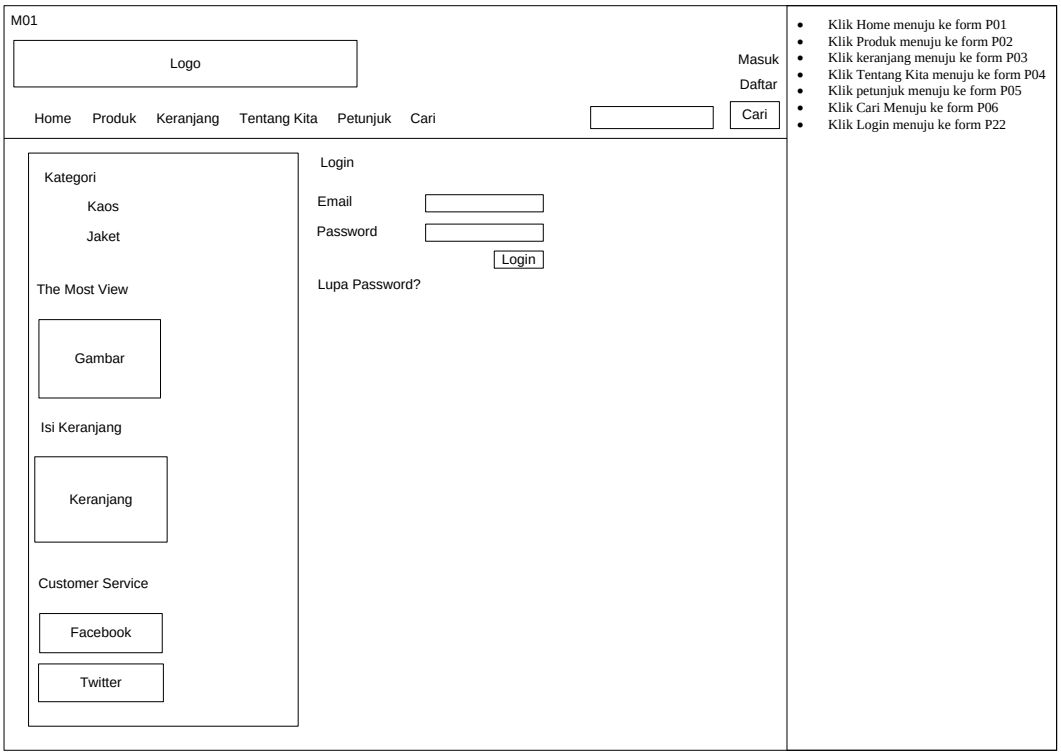
8. Form Daftar




**Gambar 3.34 Form Daftar.**

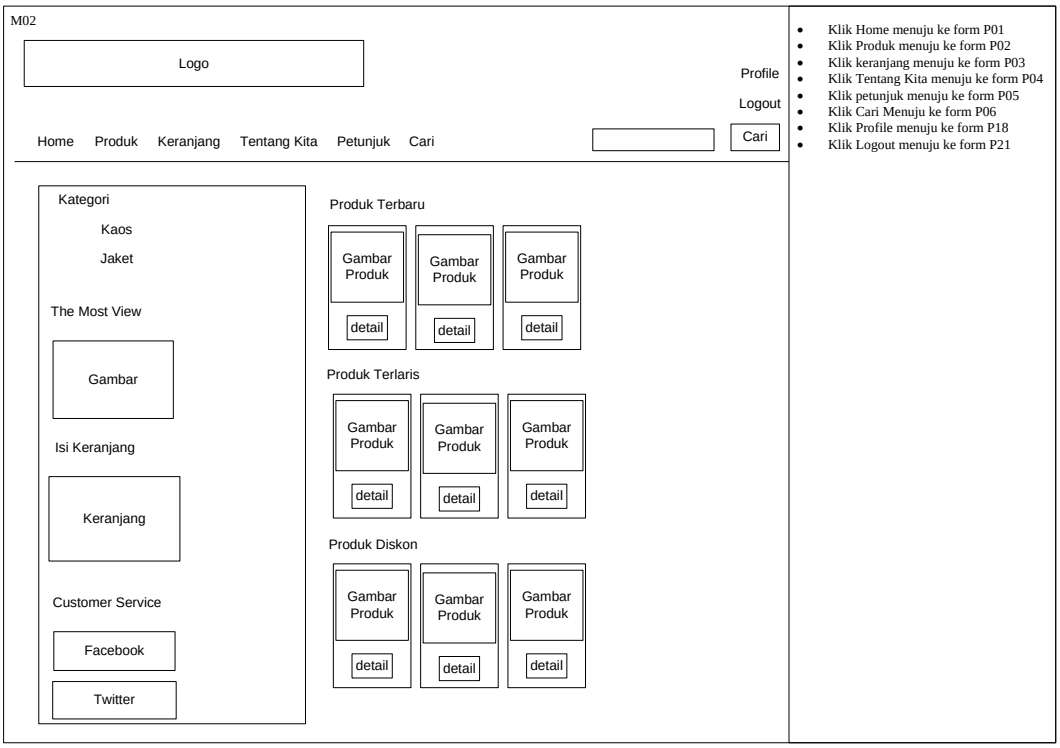
### 3.2.2.3 Perancangan Antarmuka Pengunjung *Member*

1. Form *Login*



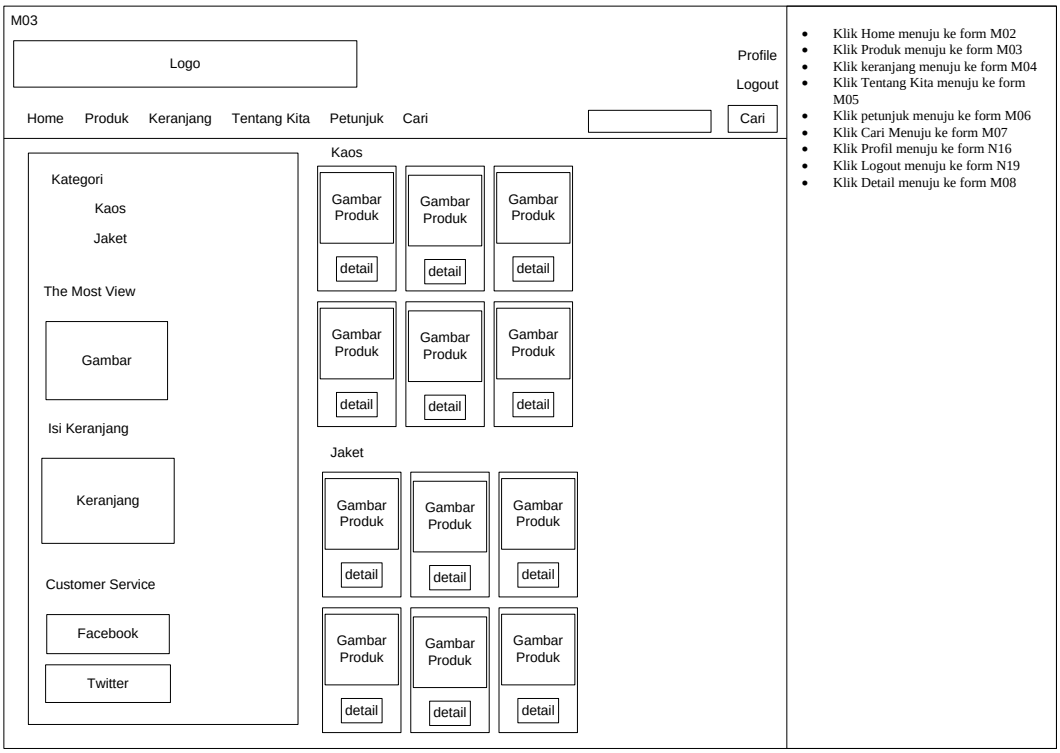

**Gambar 3.35 Form *Login*.**

2. Form *Home*



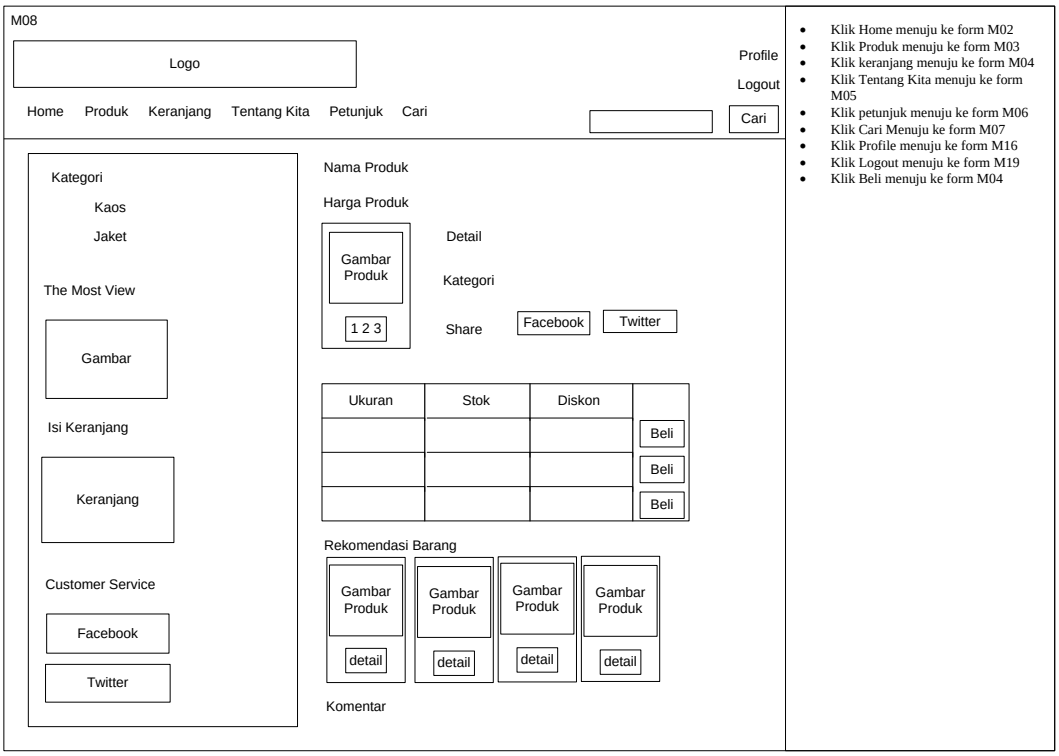

**Gambar 3.36 Form *Home*.**

3. Form Produk



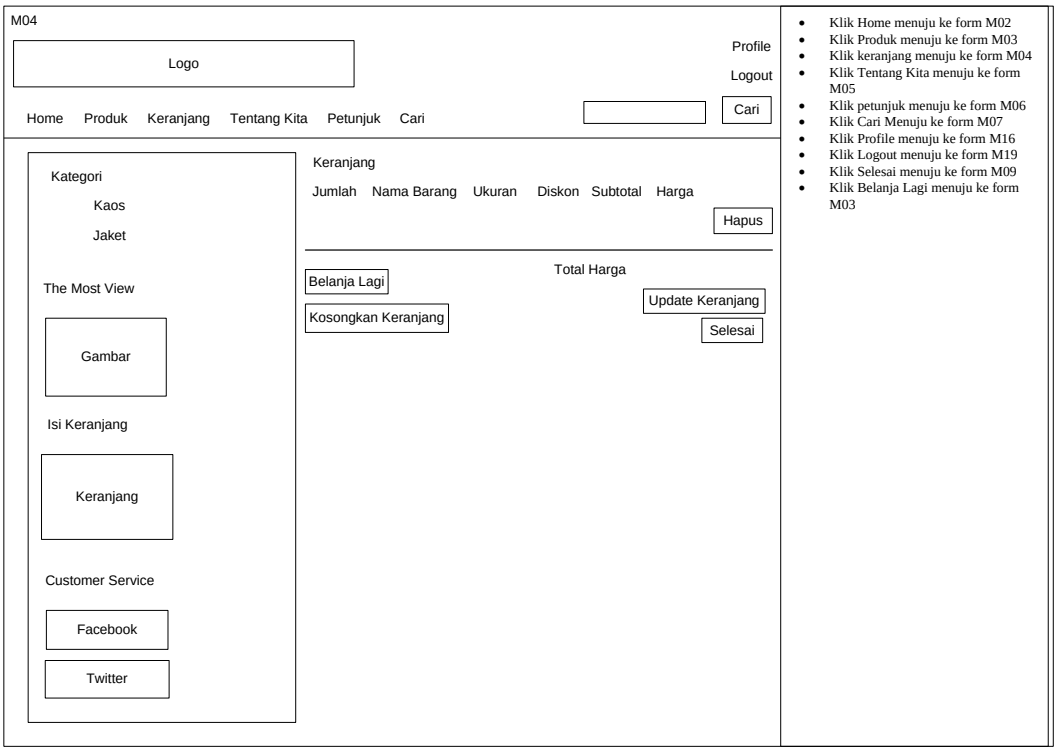

**Gambar 3.37 Form Produk.**

4. Form Detail Produk



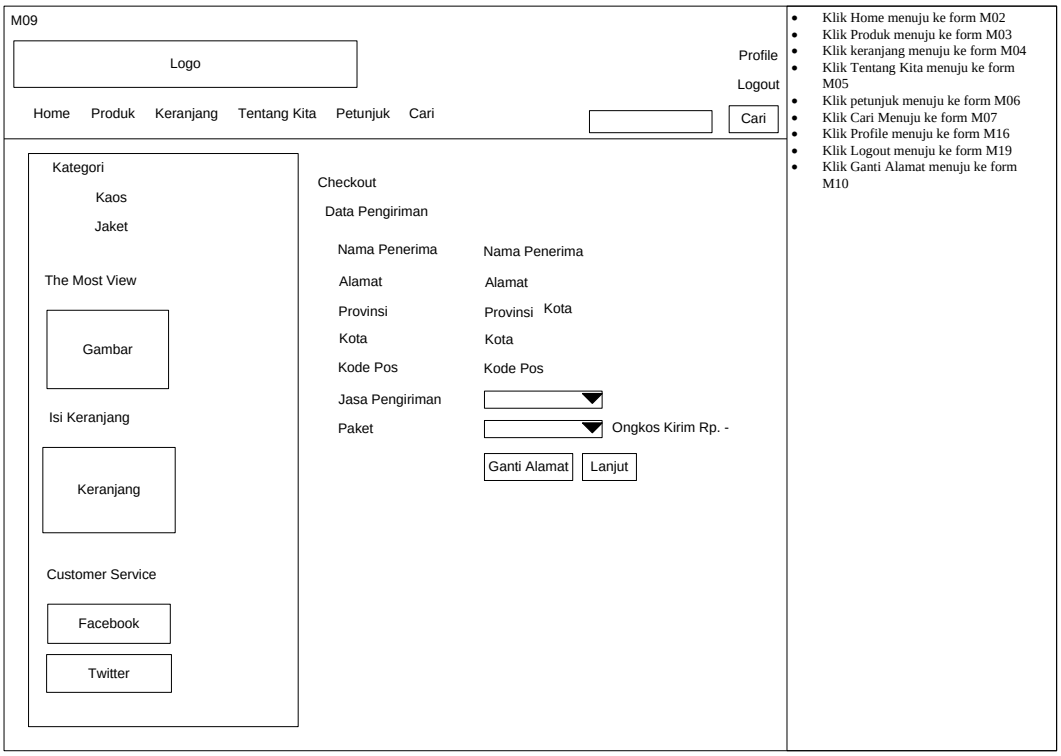

**Gambar 3.38 Form Detail Produk.**

5. Form Keranjang Belanja



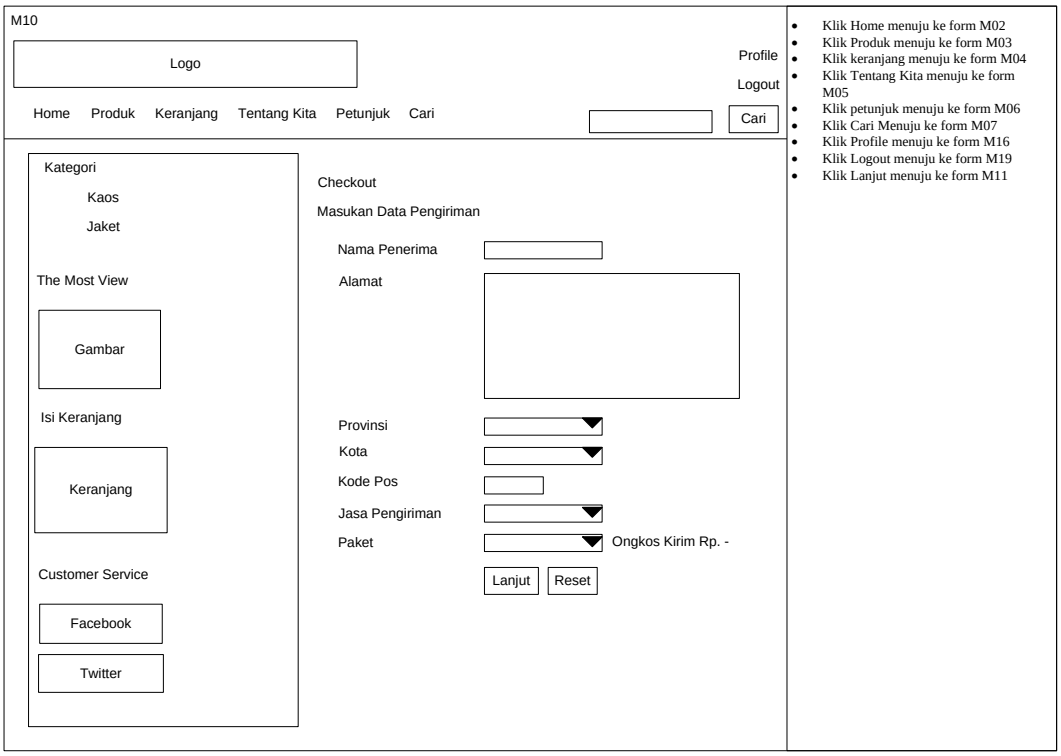

**Gambar 3.39 Form Keranjang Belanja.**

6. Form *Chekout*



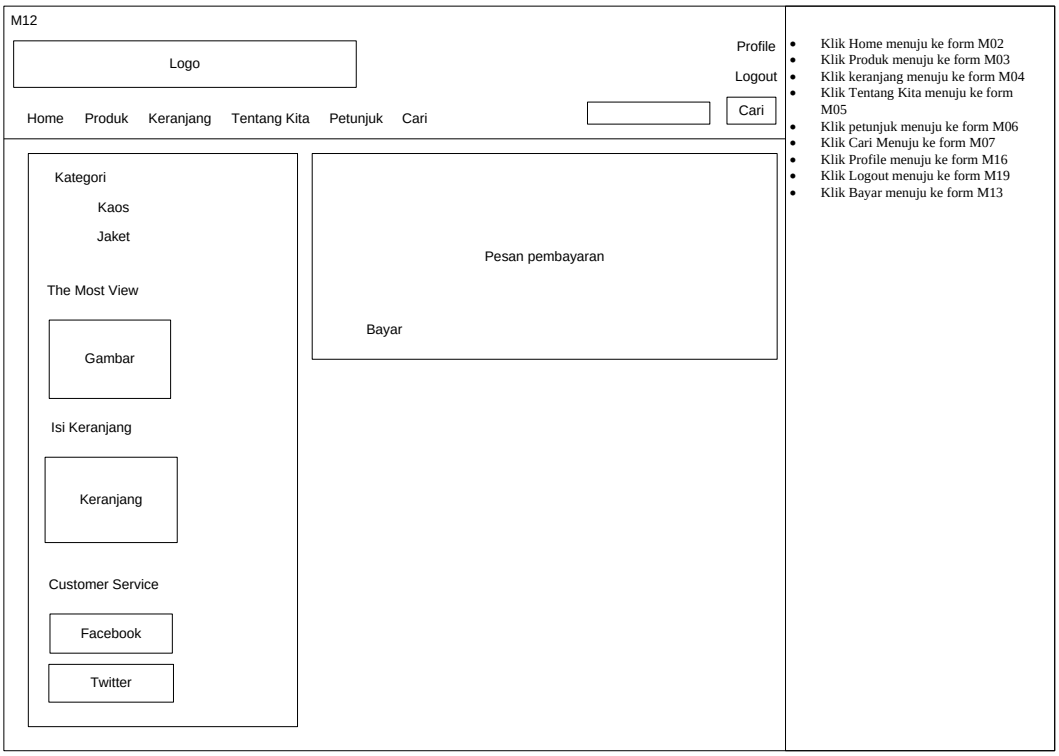

**Gambar 3.40 Form *Chekout*.**

7. Form Ganti Alamat



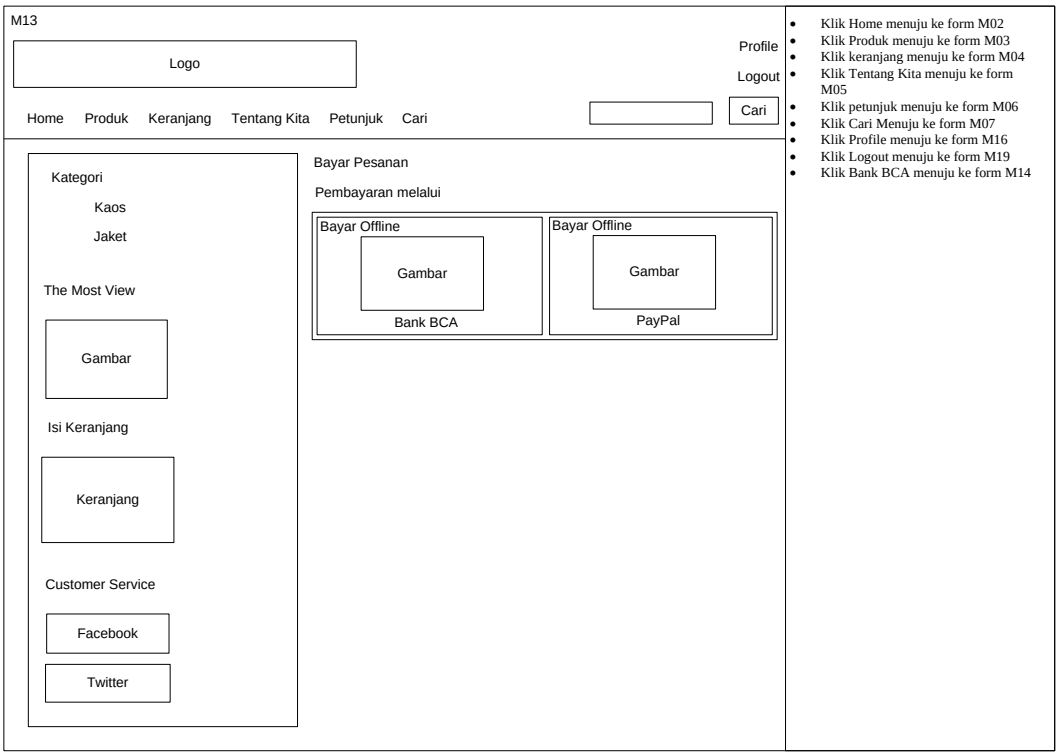

**Gambar 3.41 Form Daftar.**

8. Form Pesan Pembayaran



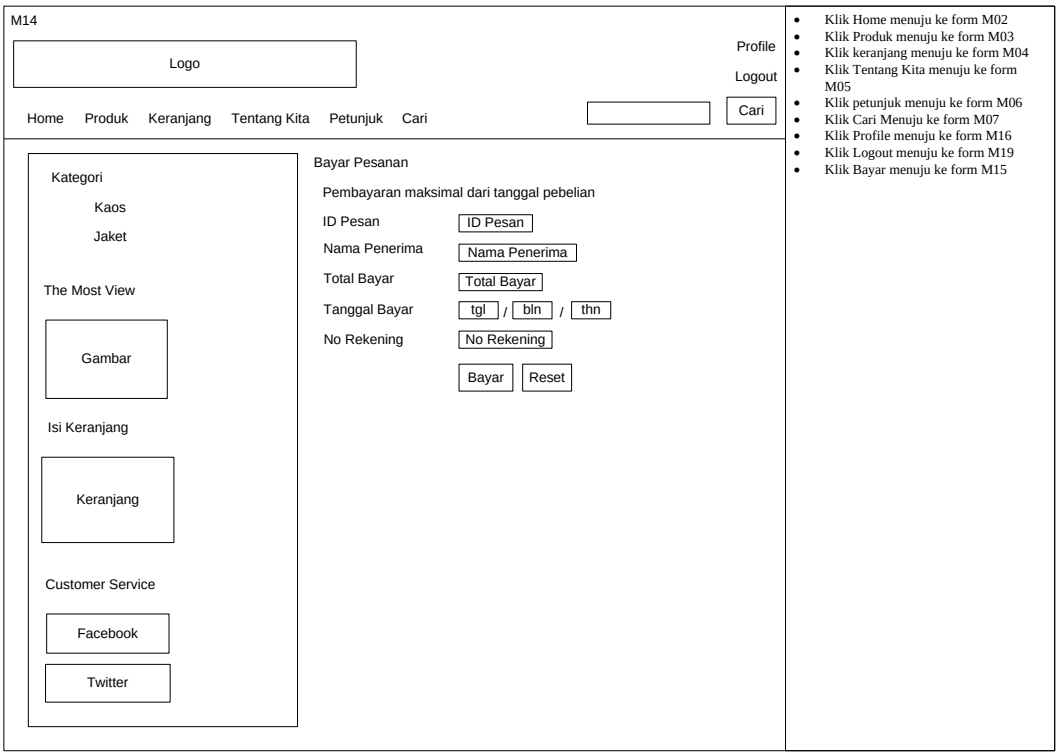

**Gambar 3.42 Form Pesan Pembayaran.**

9. Form Pembayaran



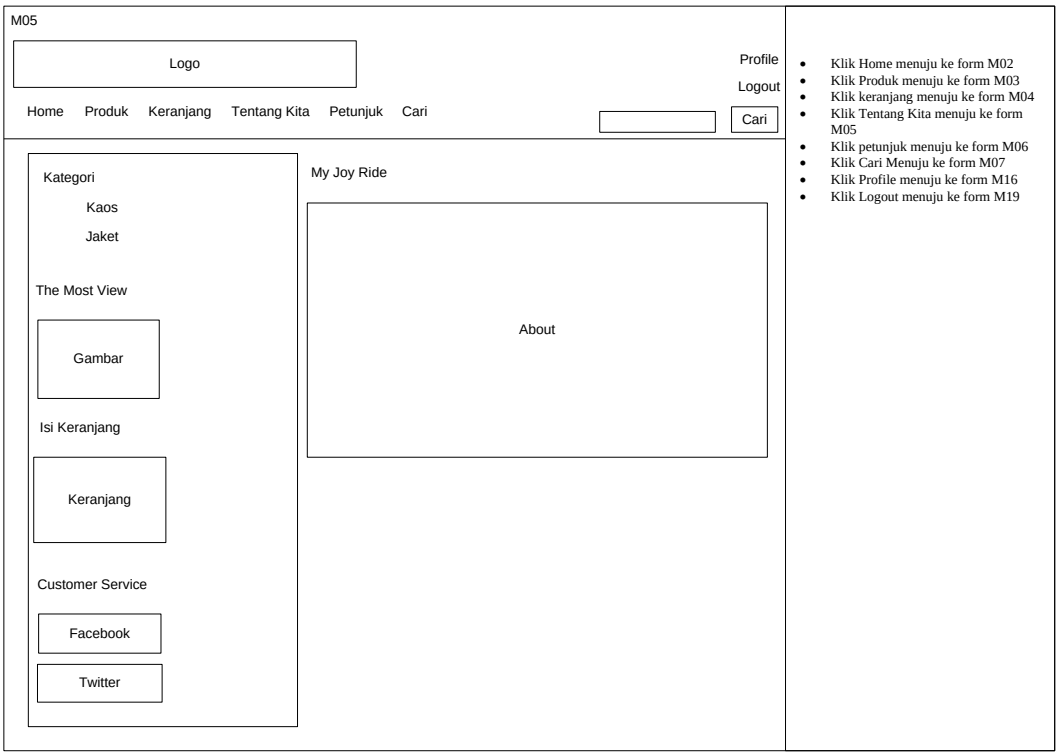

**Gambar 3.43 Form Pembayaran.**

10. Form Pembayaran BANK



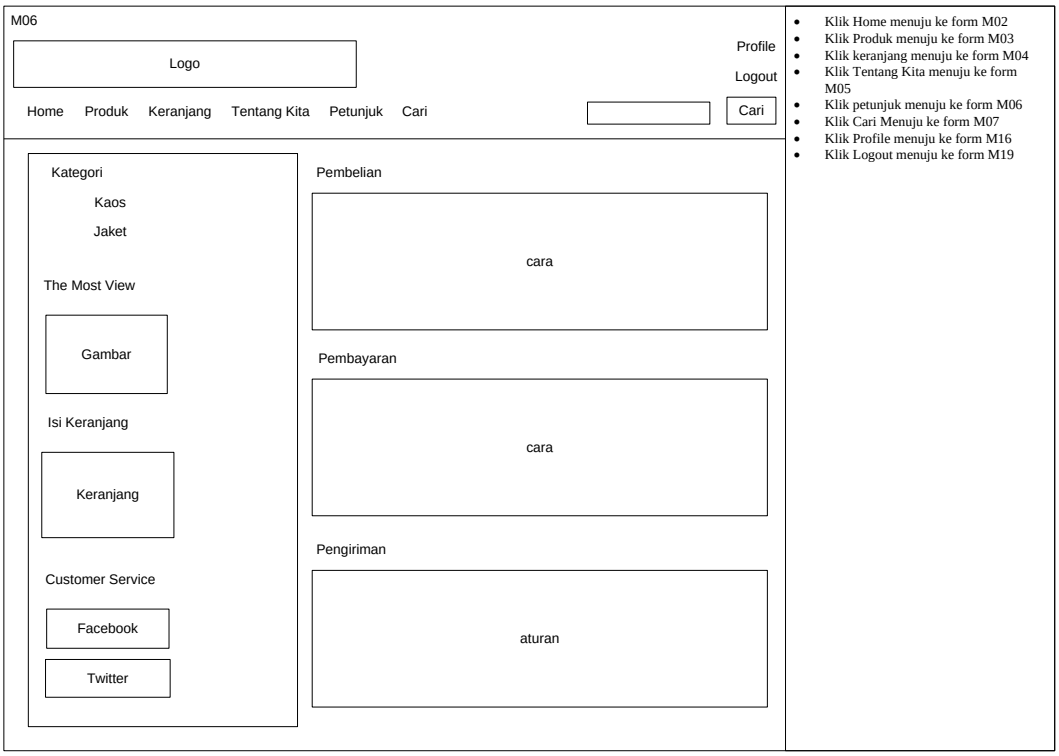

**Gambar 3.44 Form Pembayaran BANK.**

11. Form Tentang Kita



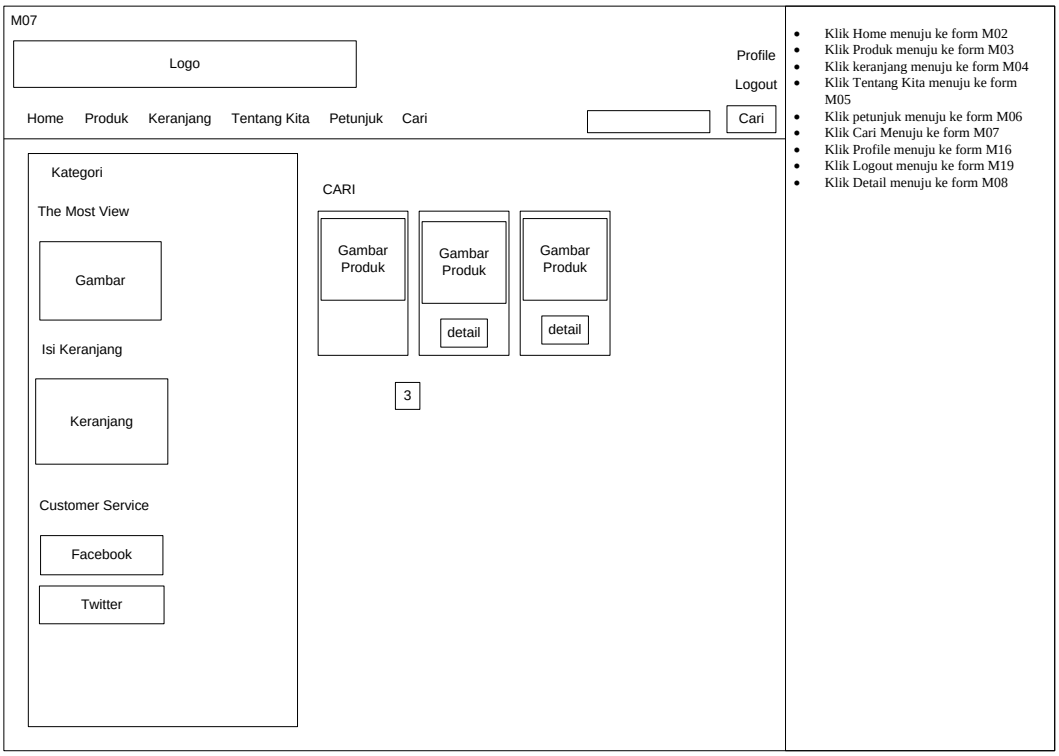

**Gambar 3.45 Form Tentang Kita.**

12. Form Petunjuk



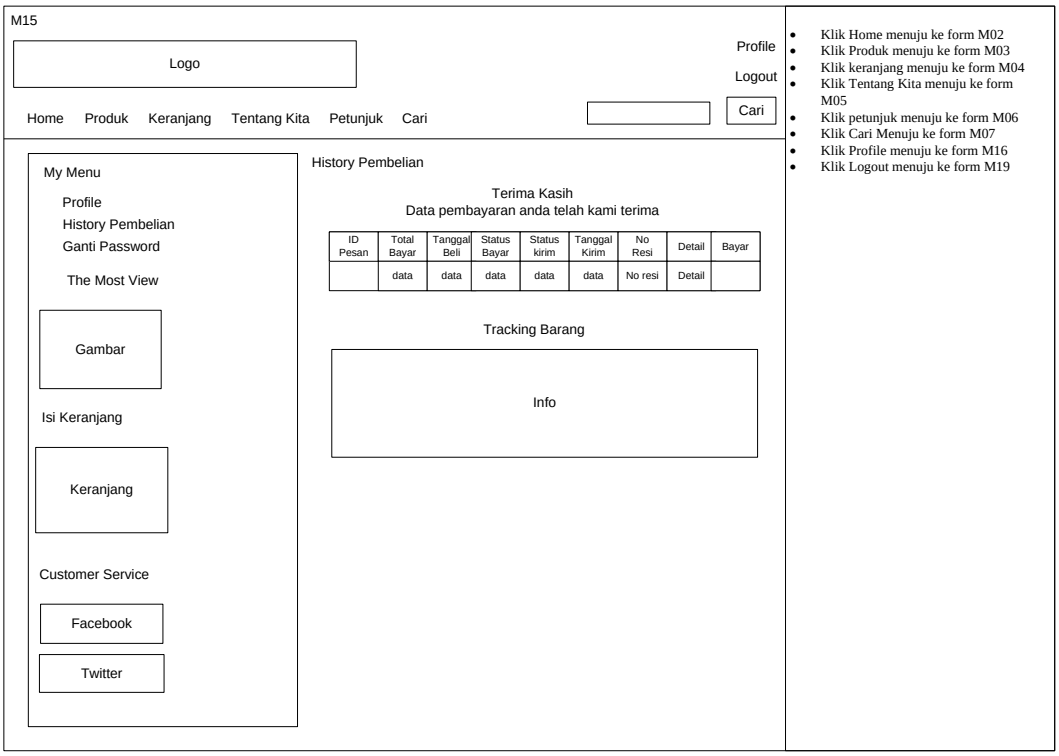

**Gambar 3.46 Form Petunjuk.**

13. Form Cari



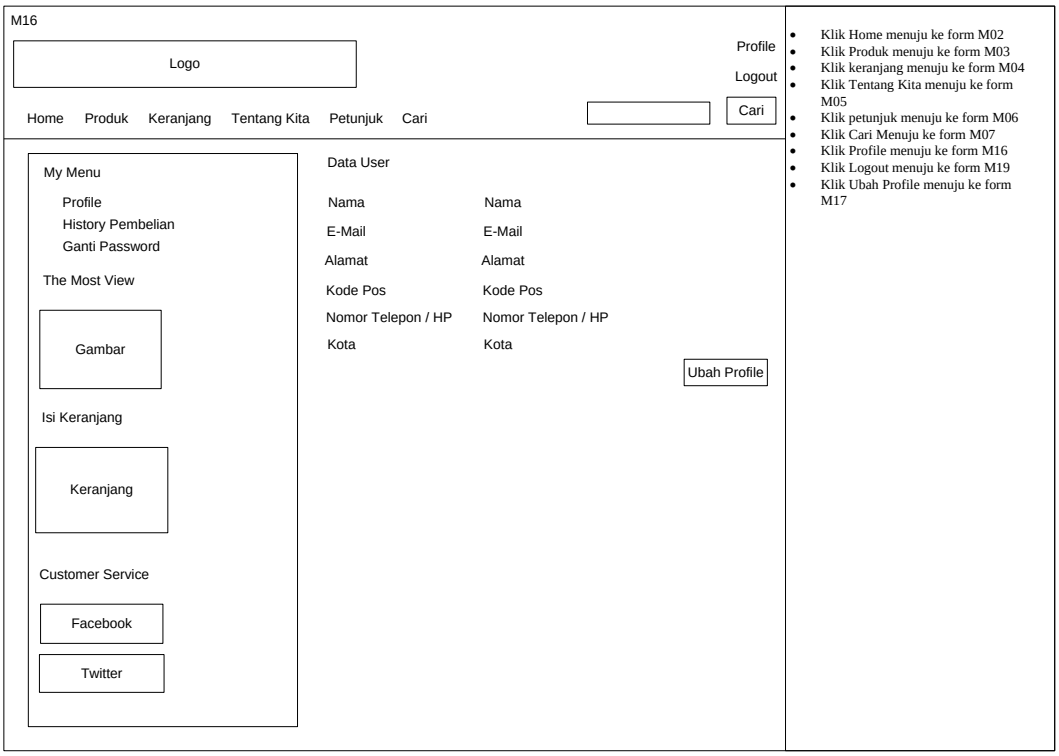

**Gambar 3.47 Form Cari.**

14. Form *History* Pemesanan



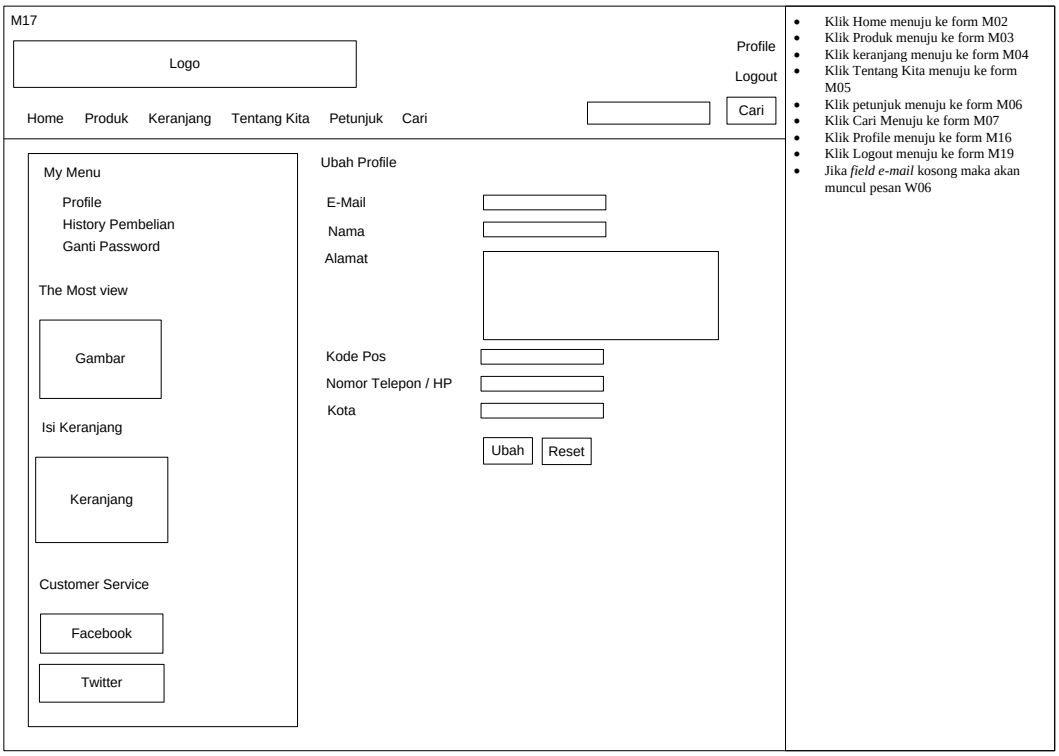

**Gambar 3.48 Form *History* Pemesanan.**

15. Form *Profile*



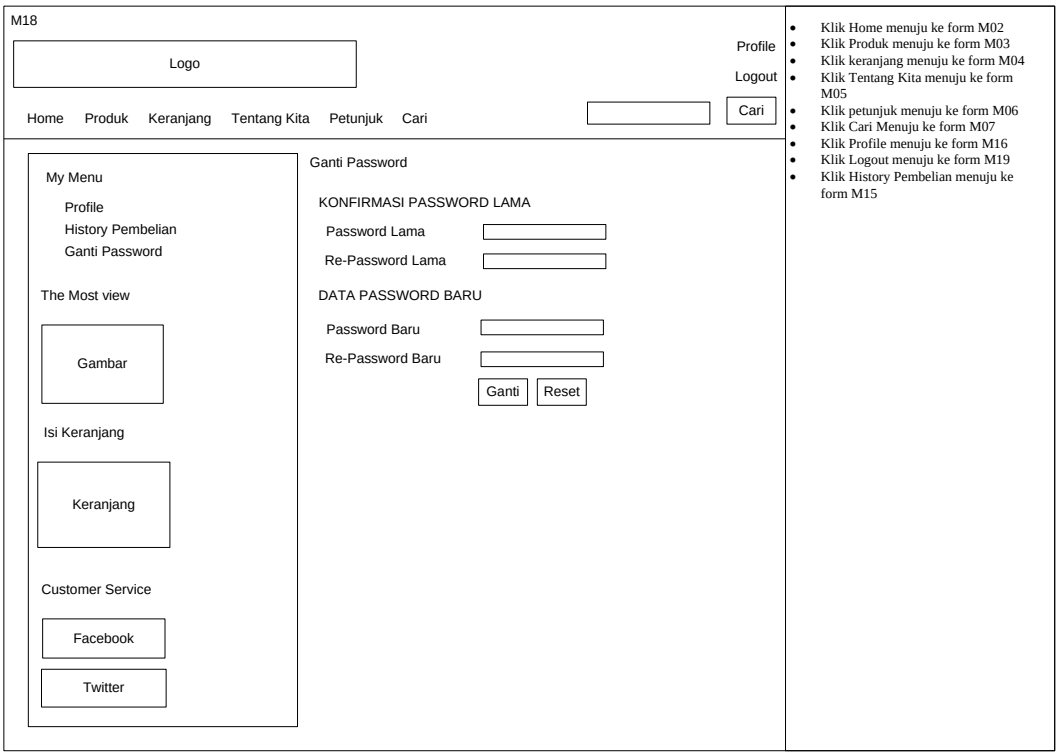

**Gambar 3.49 Form *Profile*.**

16. Form Ubah *Profile*



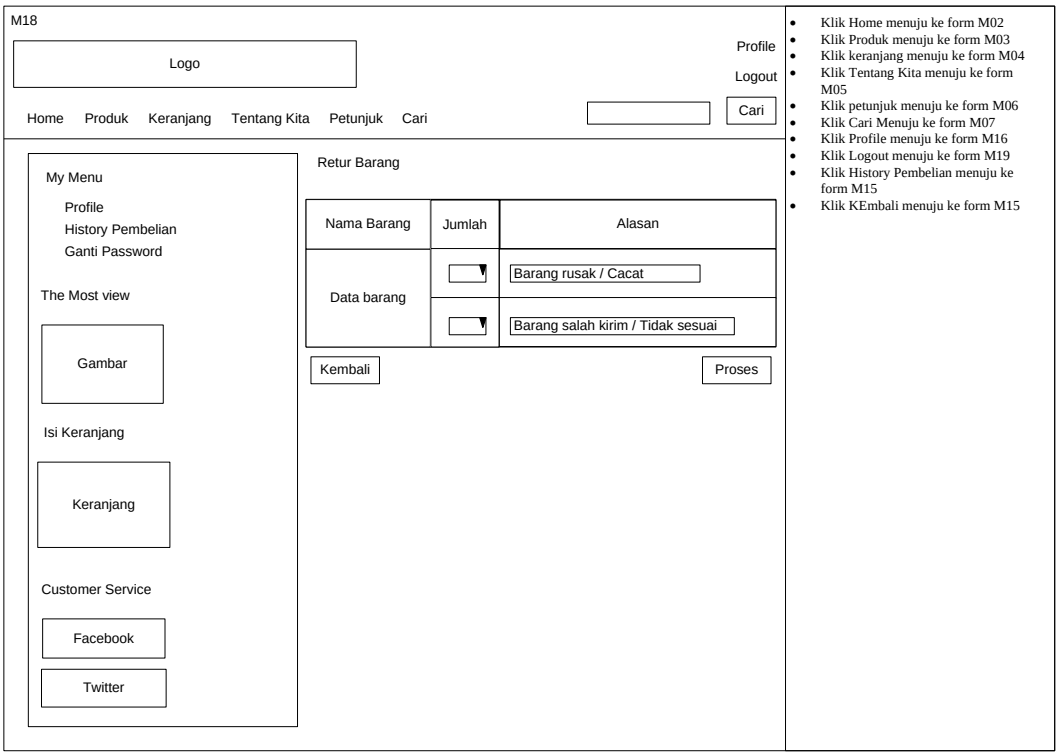

**Gambar 3.50 Form Ubah *Profile*.**

17. Form Ganti *Password*




**Gambar 3.51 Form Ganti *Password*.**

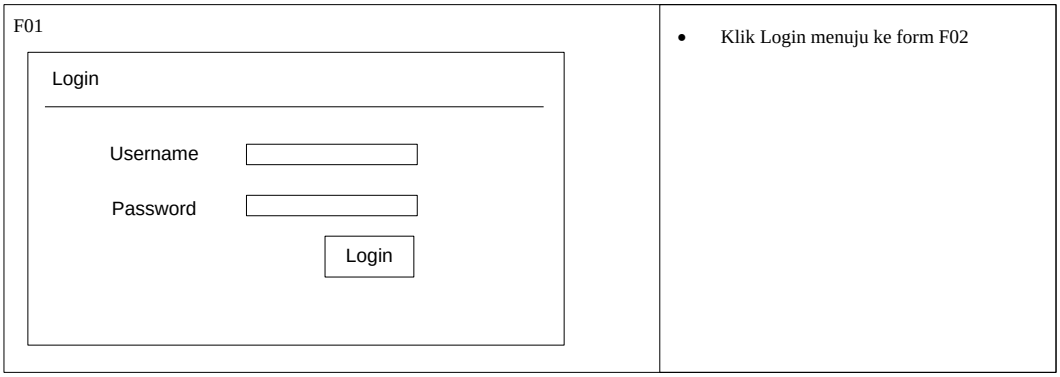
18. Form Retur




**Gambar 3.52 Form Retur.**

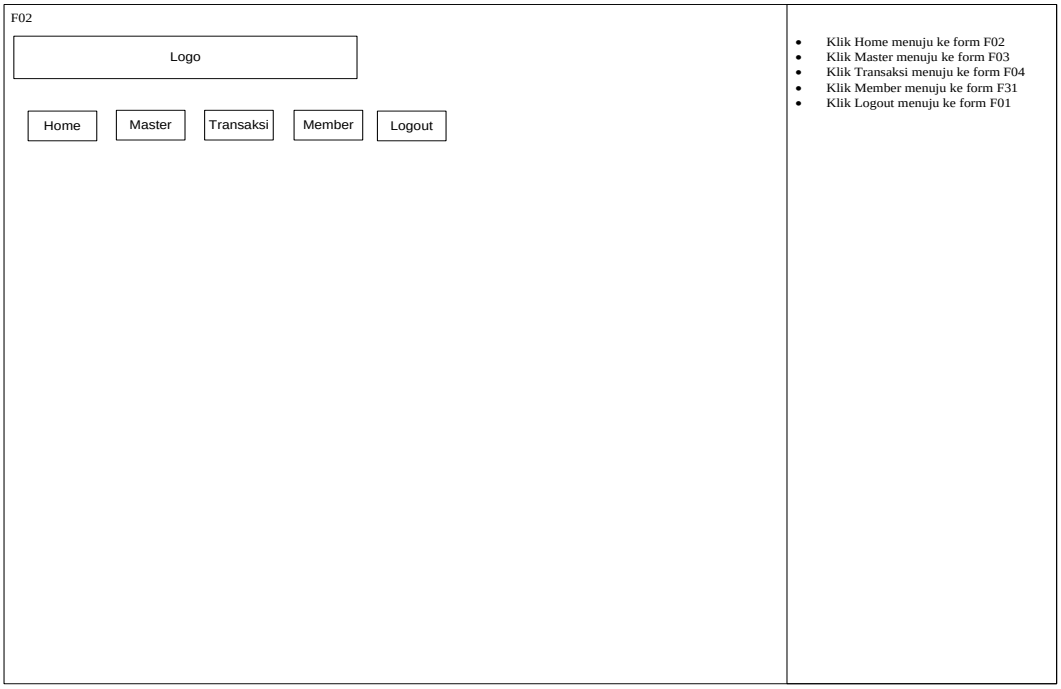
### 3.2.2.4 Perancangan Antarmuka Operator

1. Form *Login*



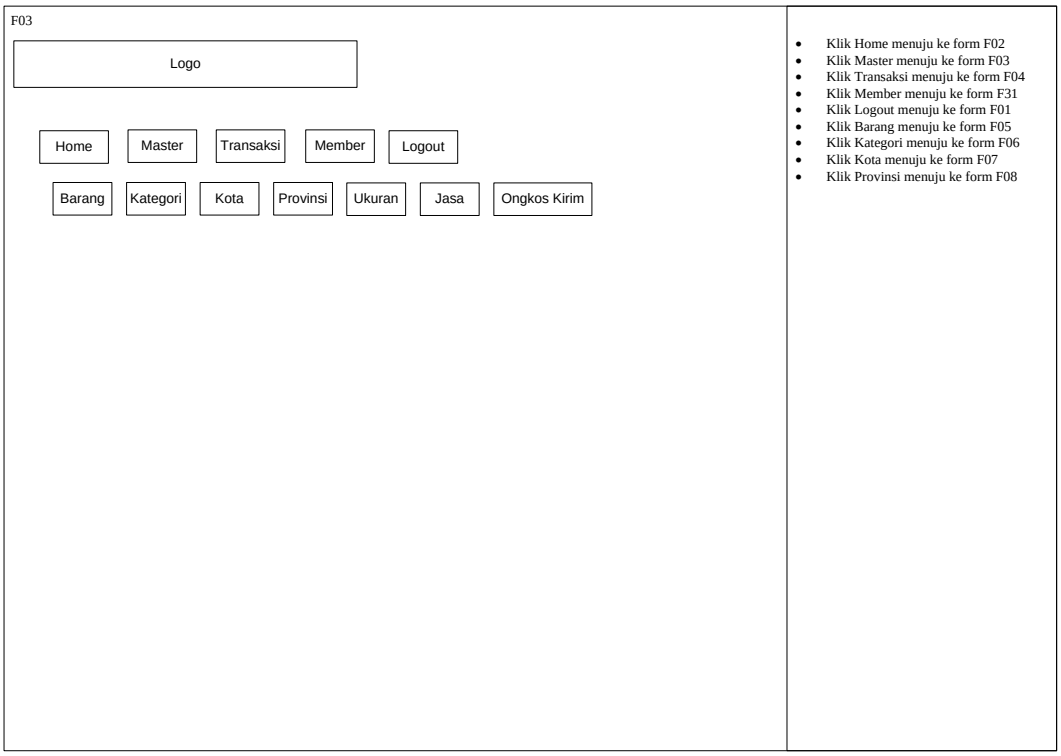

**Gambar 3.53 Form *Login*.**

2. Form *Home*



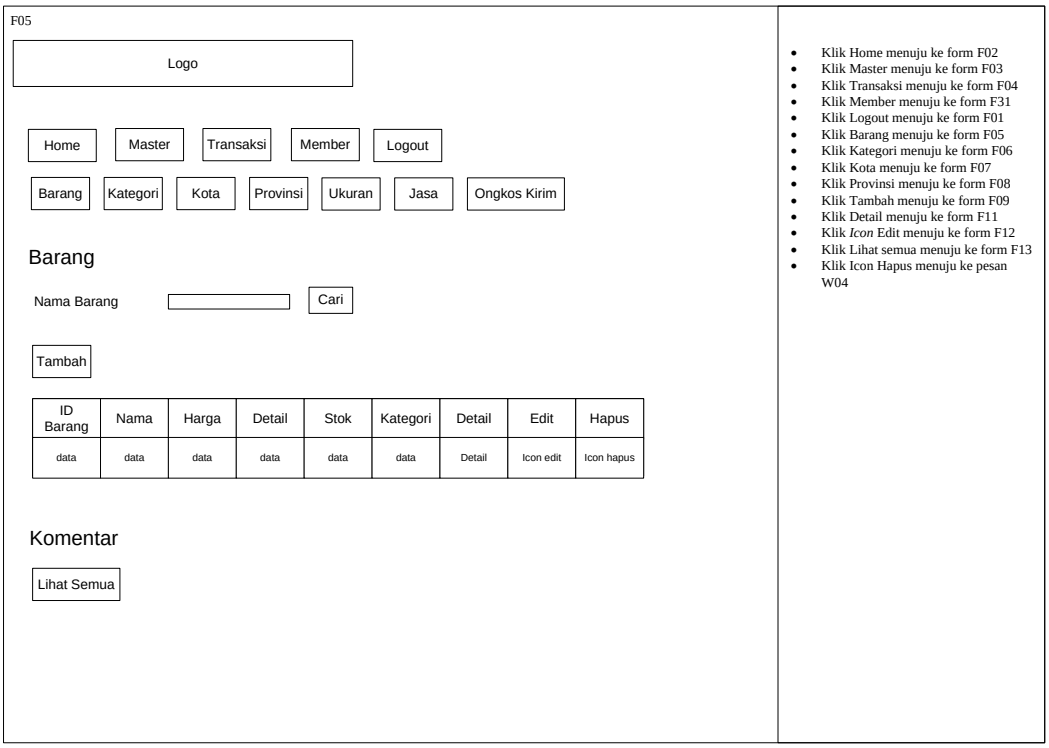

**Gambar 3.54 Form *Home*.**

3. Form Master



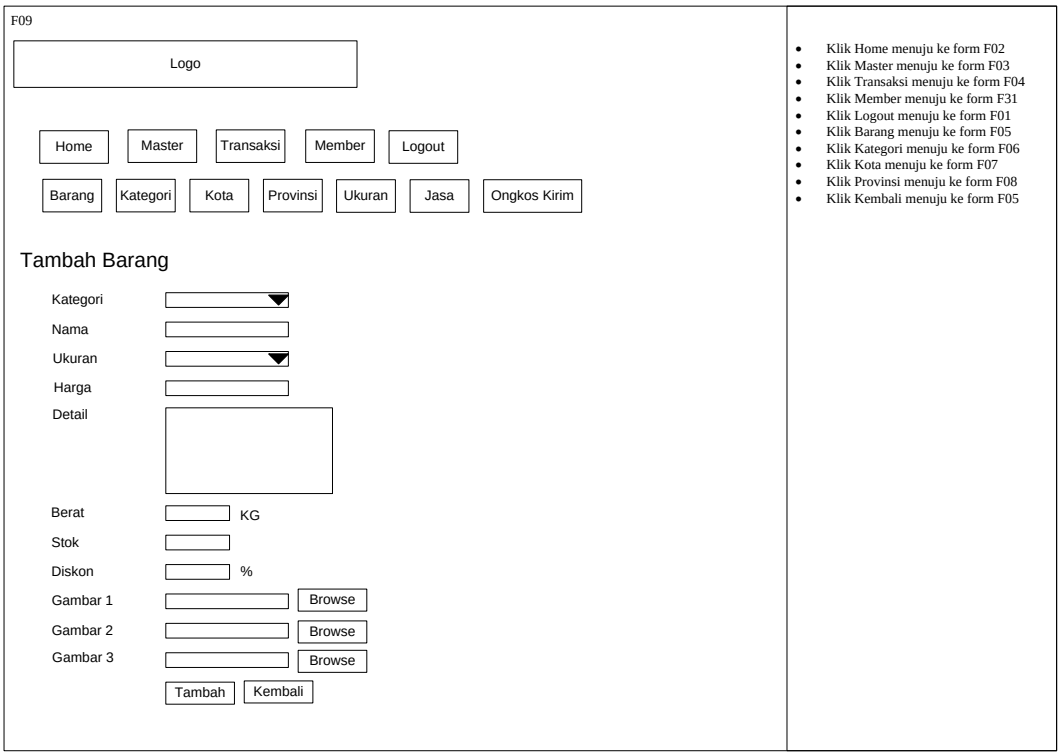

**Gambar 3.55 Form Master.**

4. Form Barang



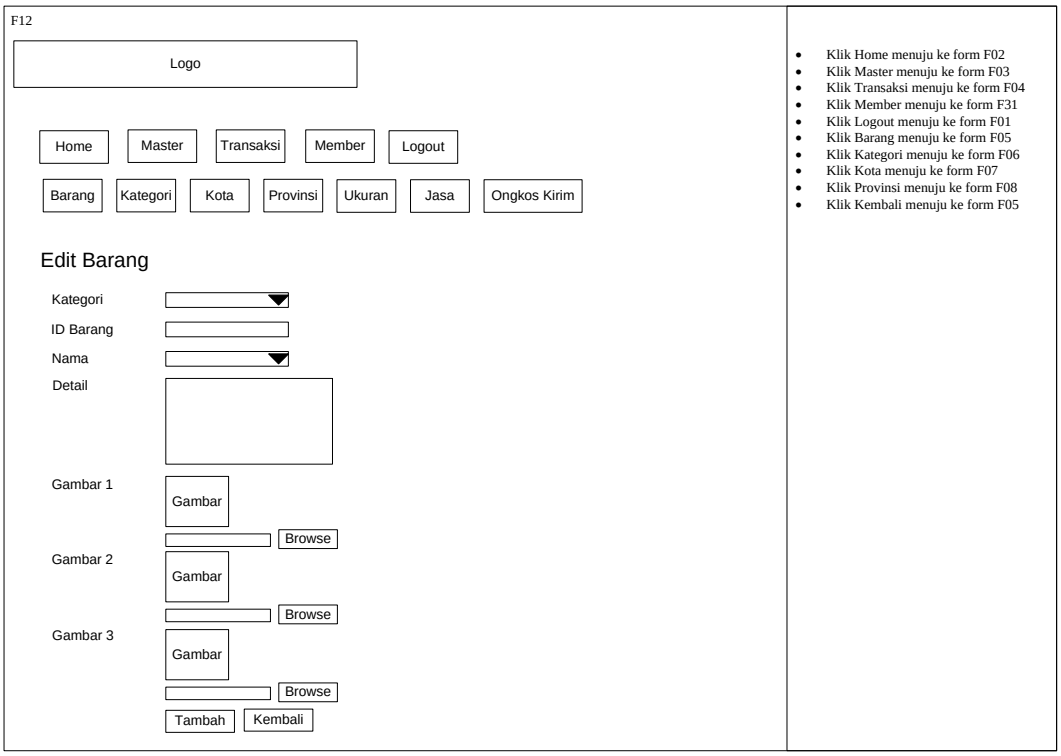

**Gambar 3.56 Form Barang.**

5. Form Tambah Barang



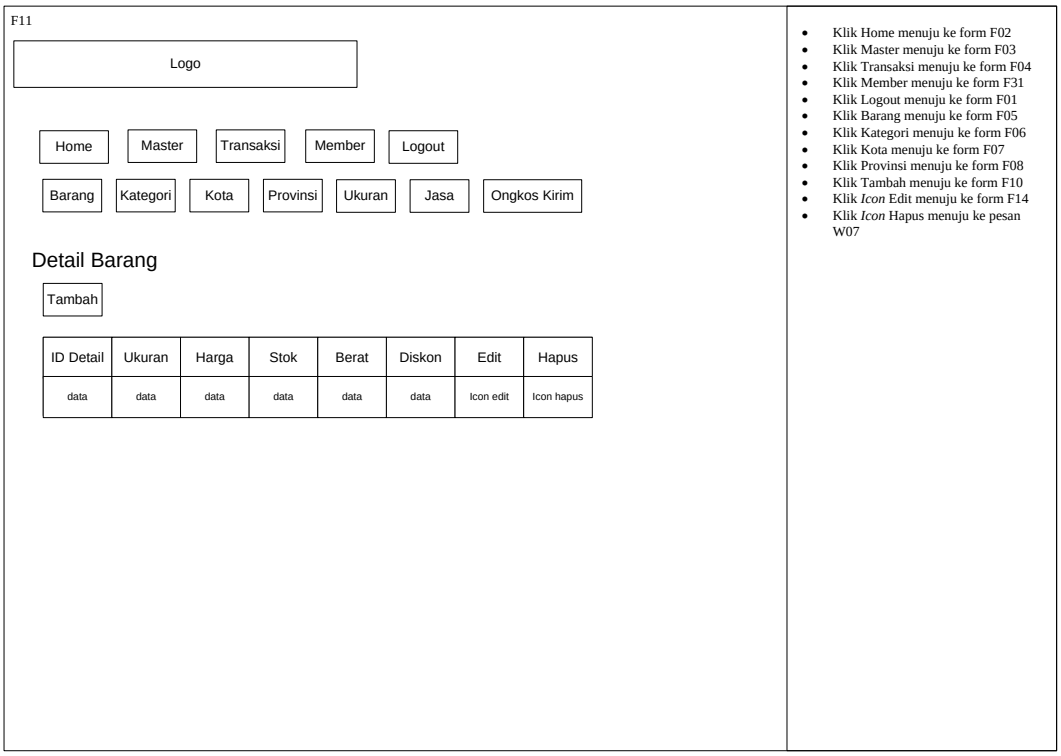

**Gambar 3.57 Form Tambah Barang.**

6. Form Edit Barang



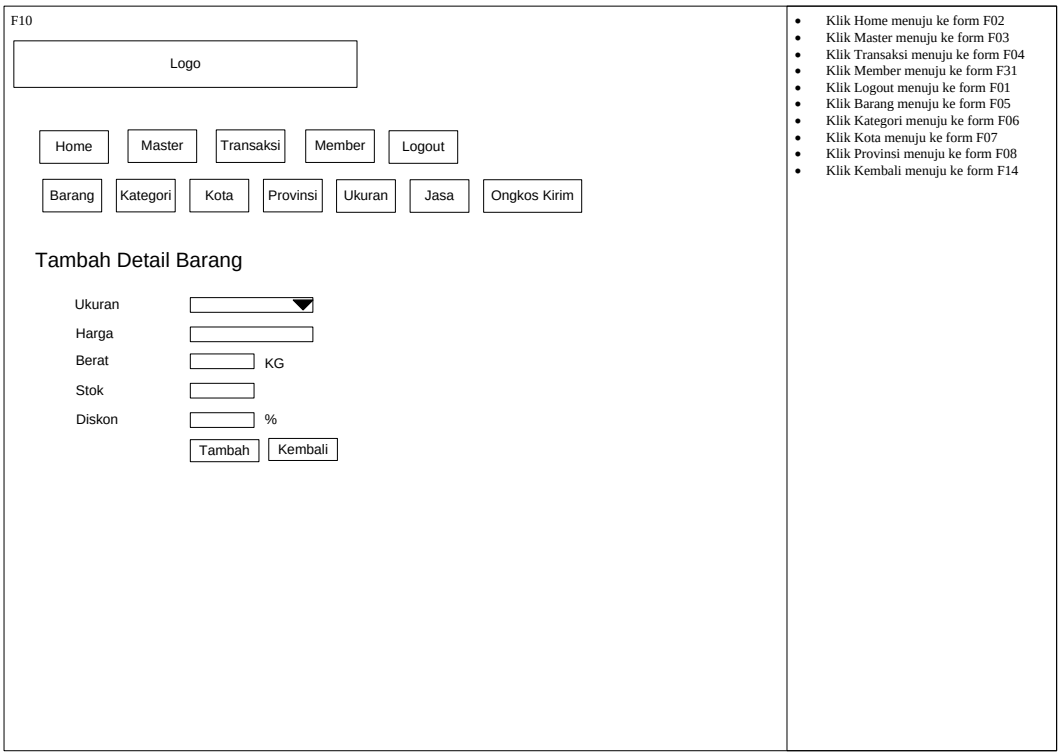

**Gambar 3.58 Form Edit Barang.**

7. Form Detail Barang



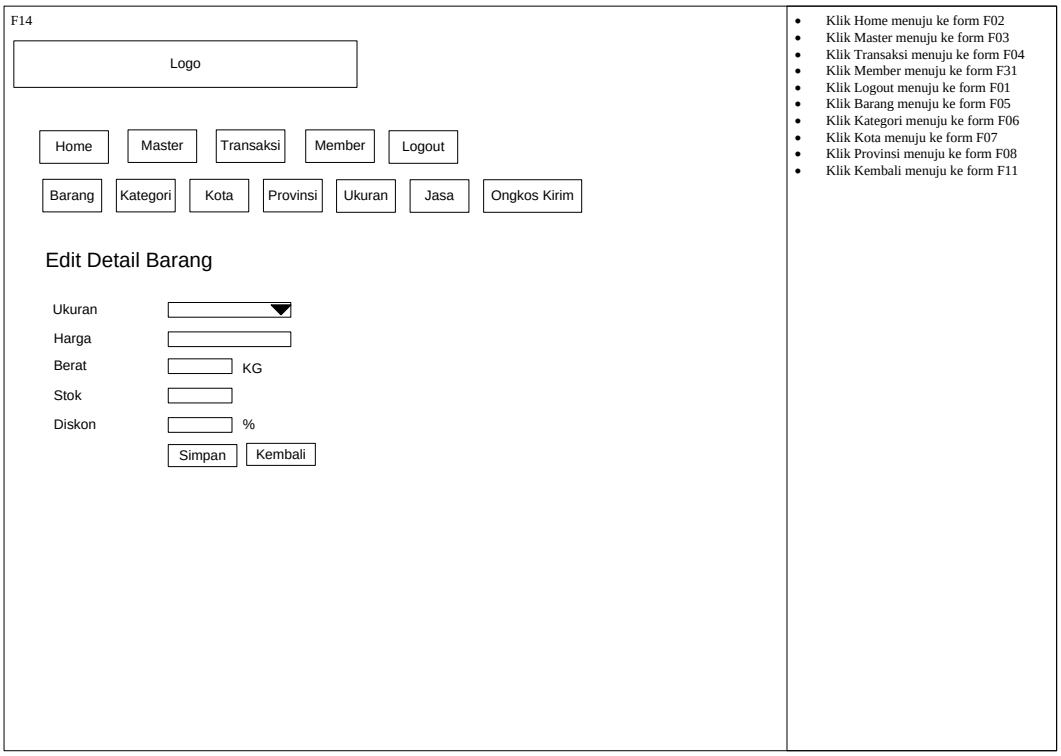

**Gambar 3.59 Form Detail Barang.**

8. Form Tambah Detail Barang



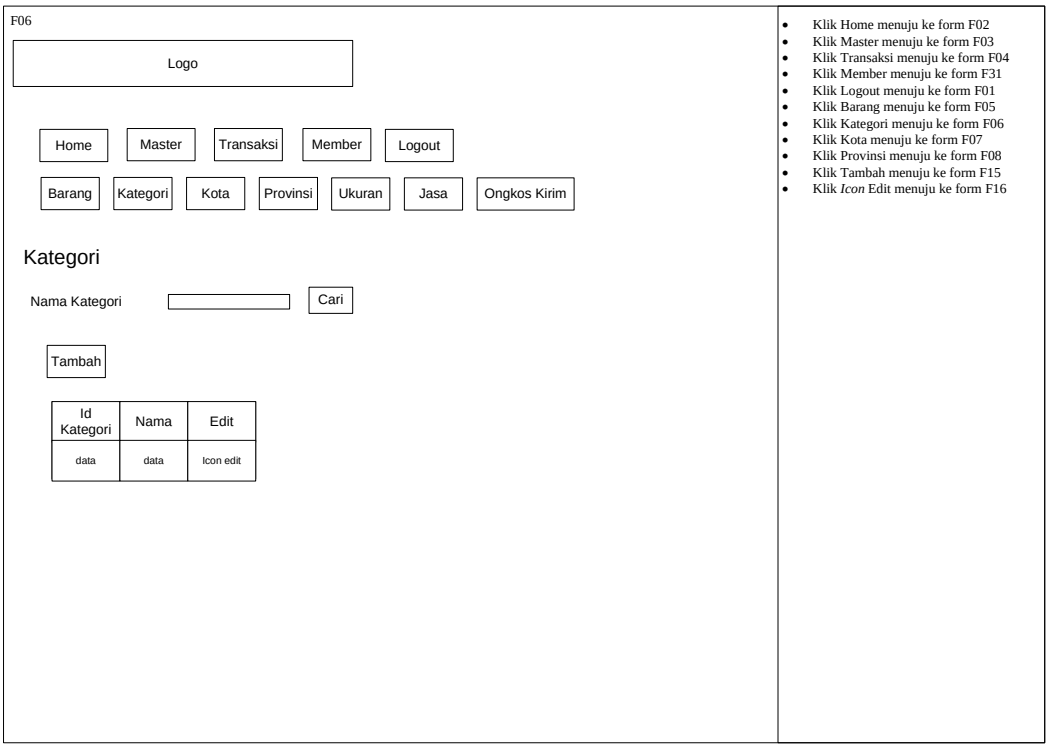

**Gambar 3.60 Form Tambah Detail Barang.**

9. Form Edit Detail Barang



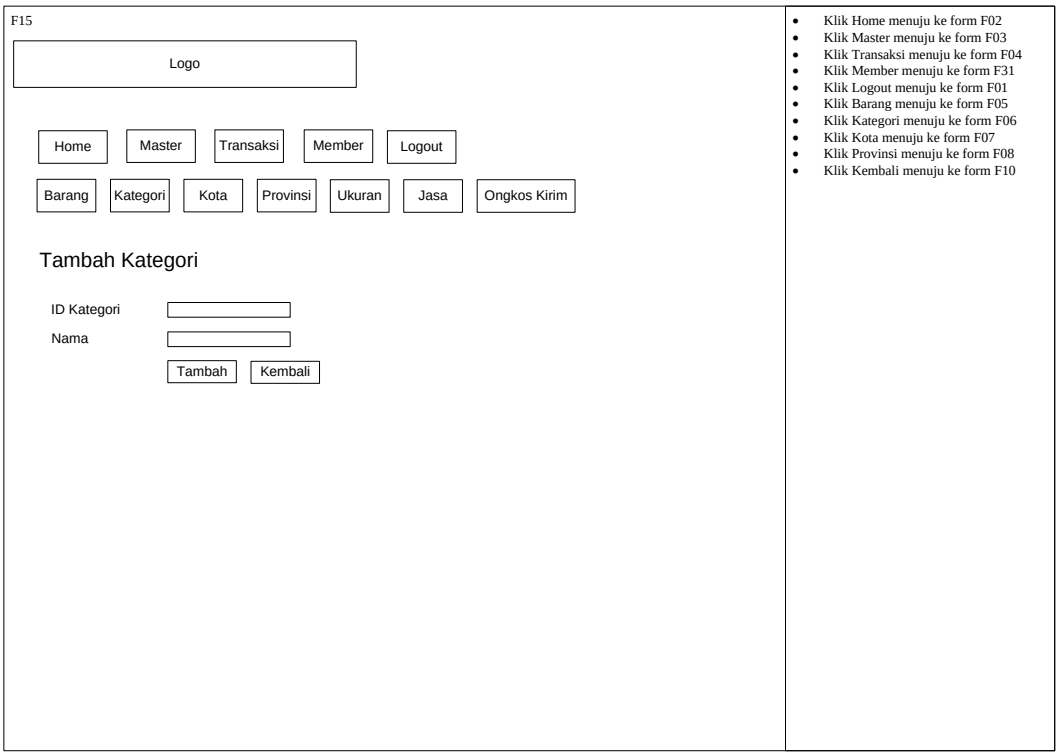

**Gambar 3.61 Form Edit Detail Barang.**

10. Form Kategori



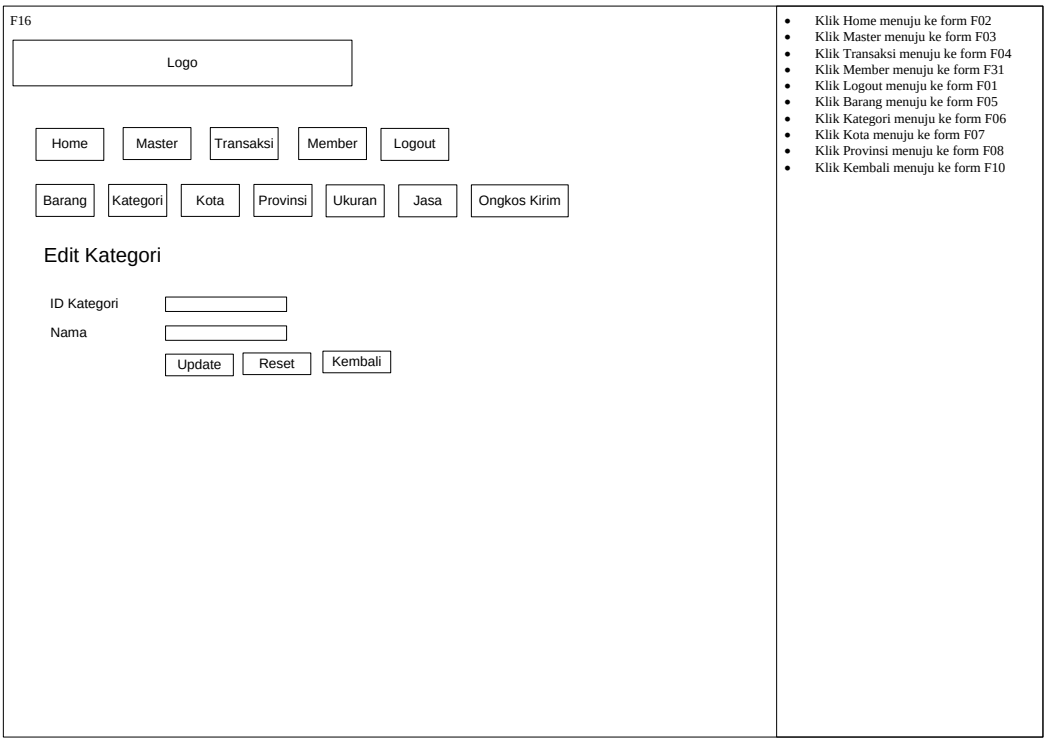

**Gambar 3.62 Form Kategori.**

11. Form Tambah Kategori



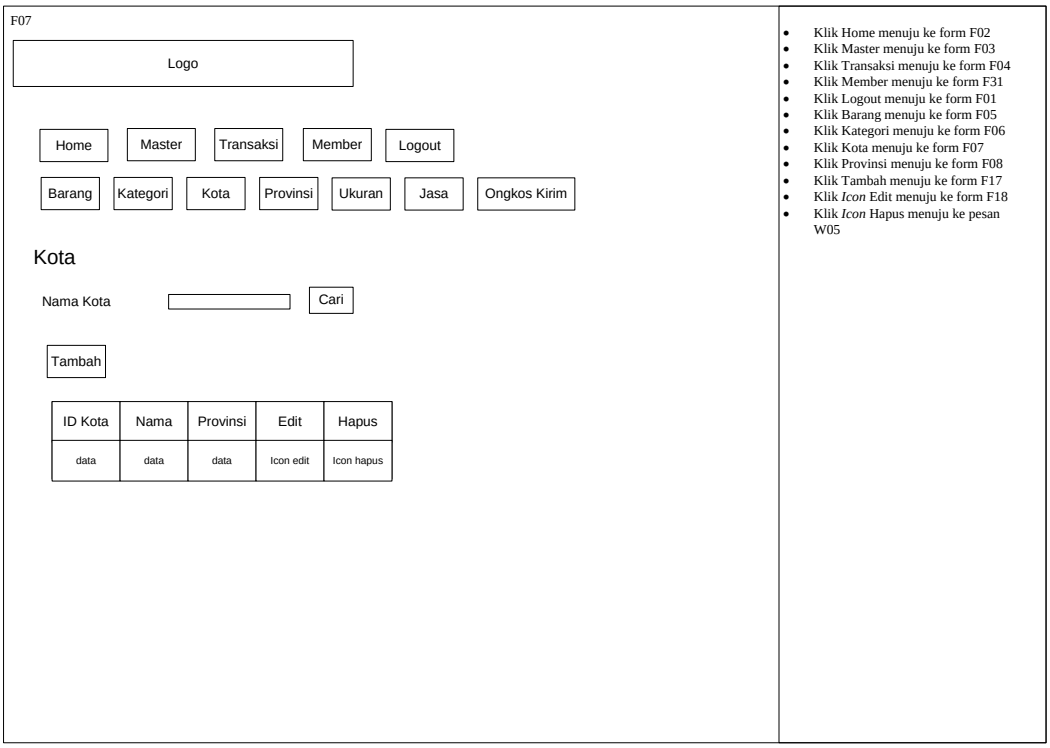

**Gambar 3.63 Form Tambah Kategori.**

12. Form Edit Kategori



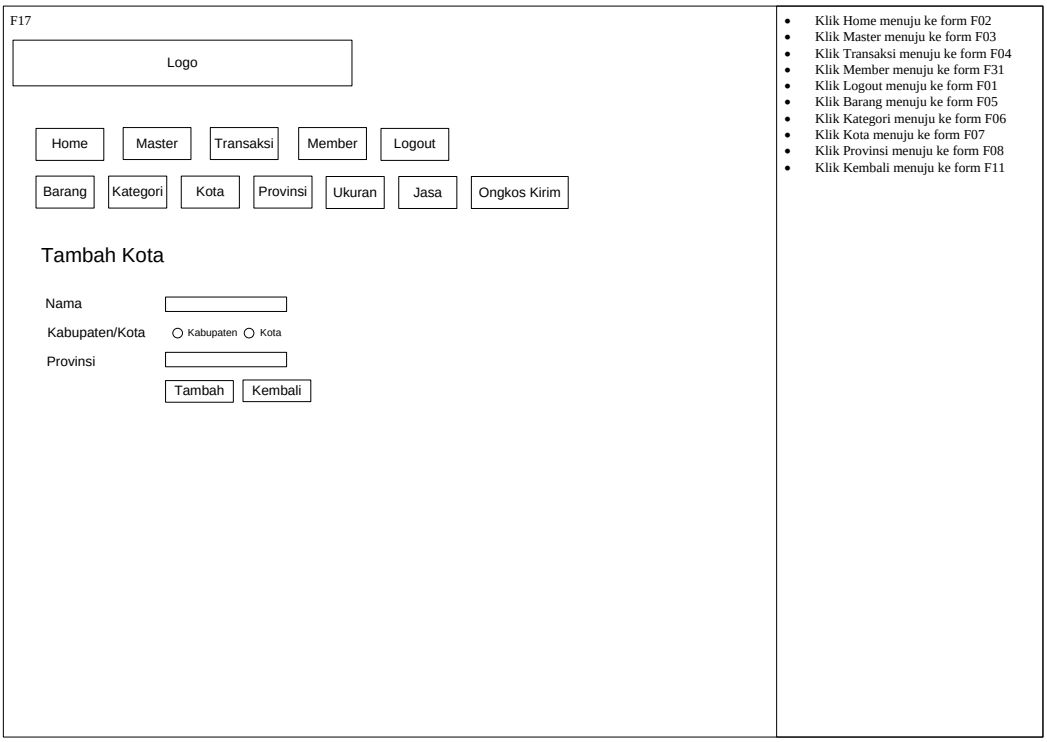

**Gambar 3.64 Form Edit Kategori.**

13. Form Kota



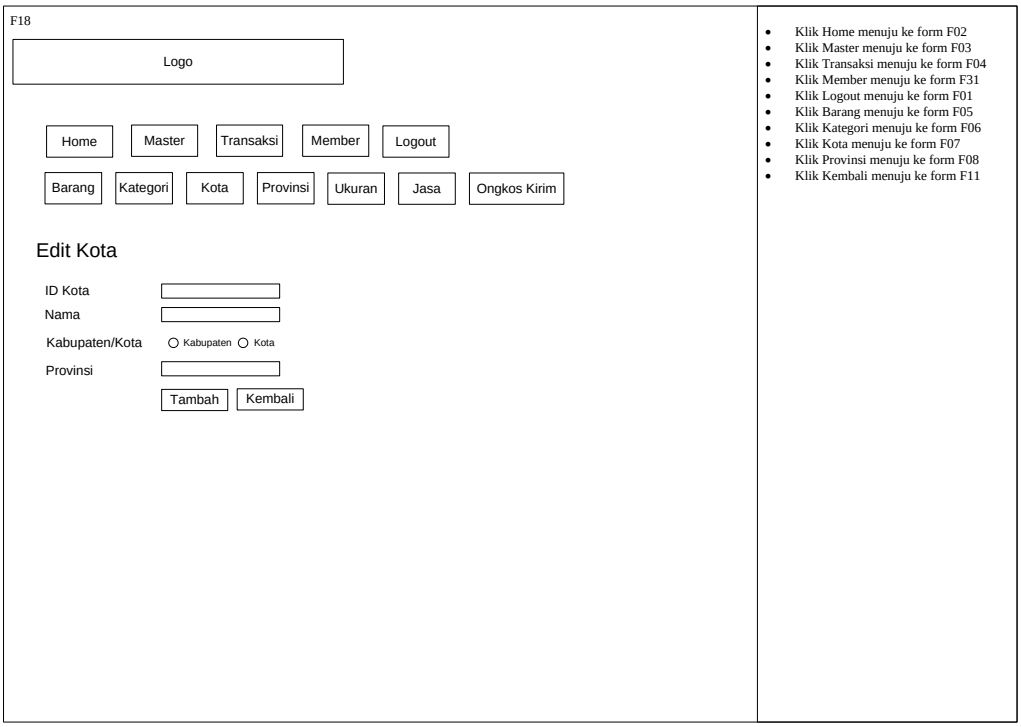

**Gambar 3.65 Form Kota.**

14. Form Tambah Kota



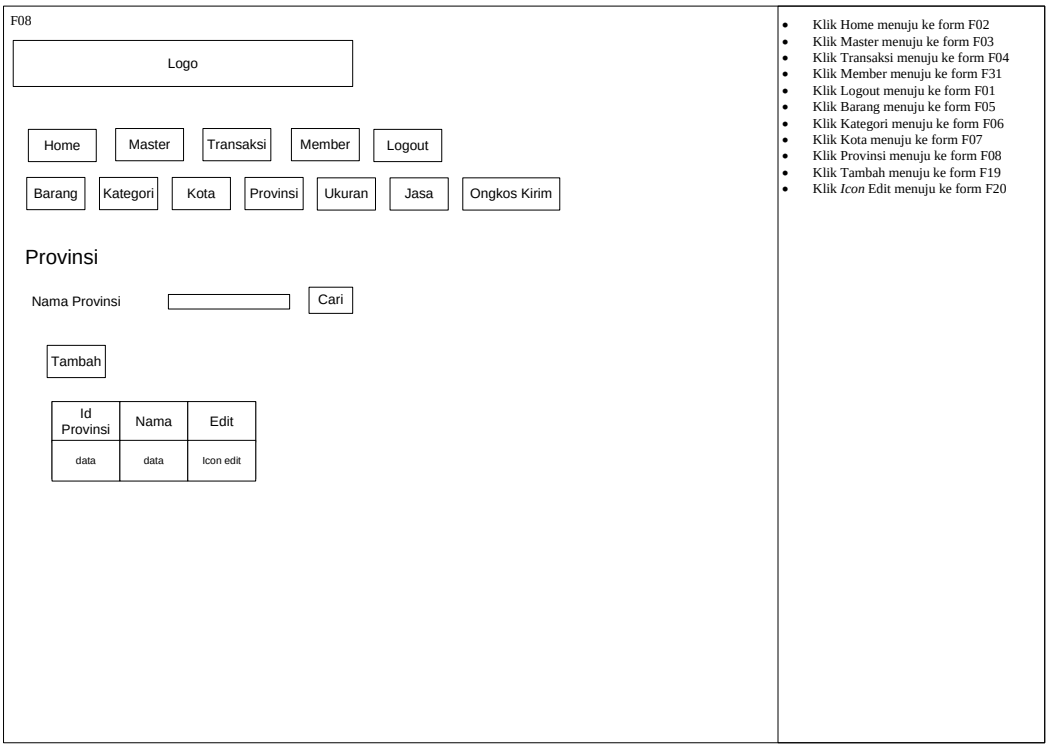

**Gambar 3.66 Form Tambah Kota.**

15. Form Edit Kota



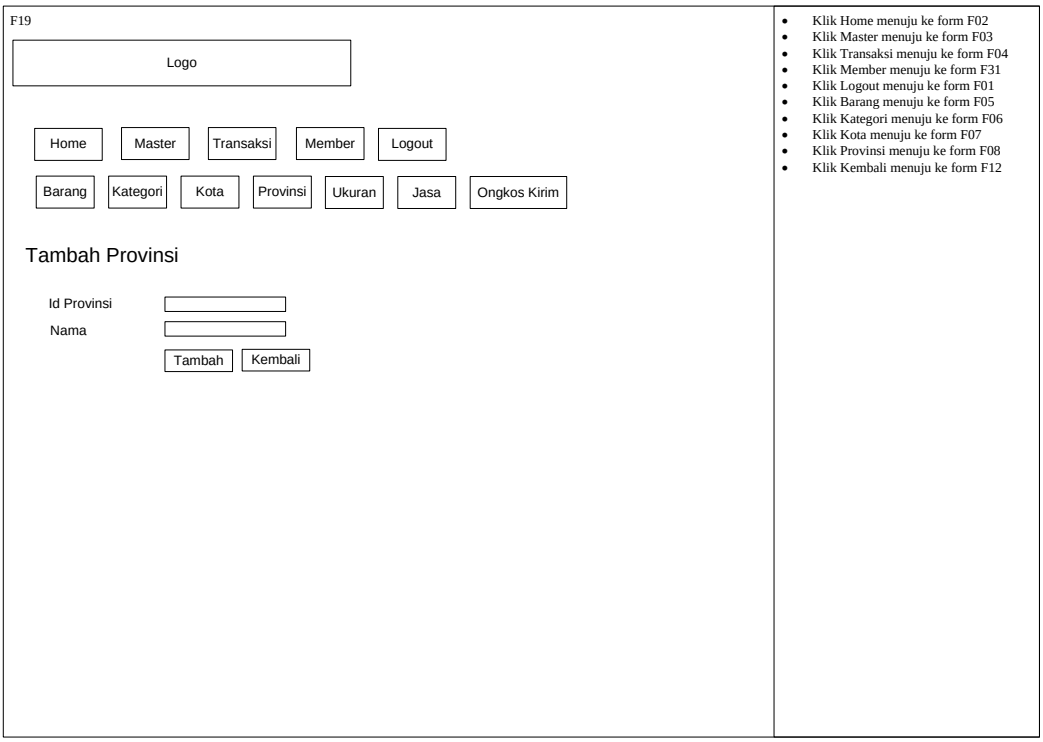

**Gambar 3.67 Form Edit Kota.**

16. Form Provinsi



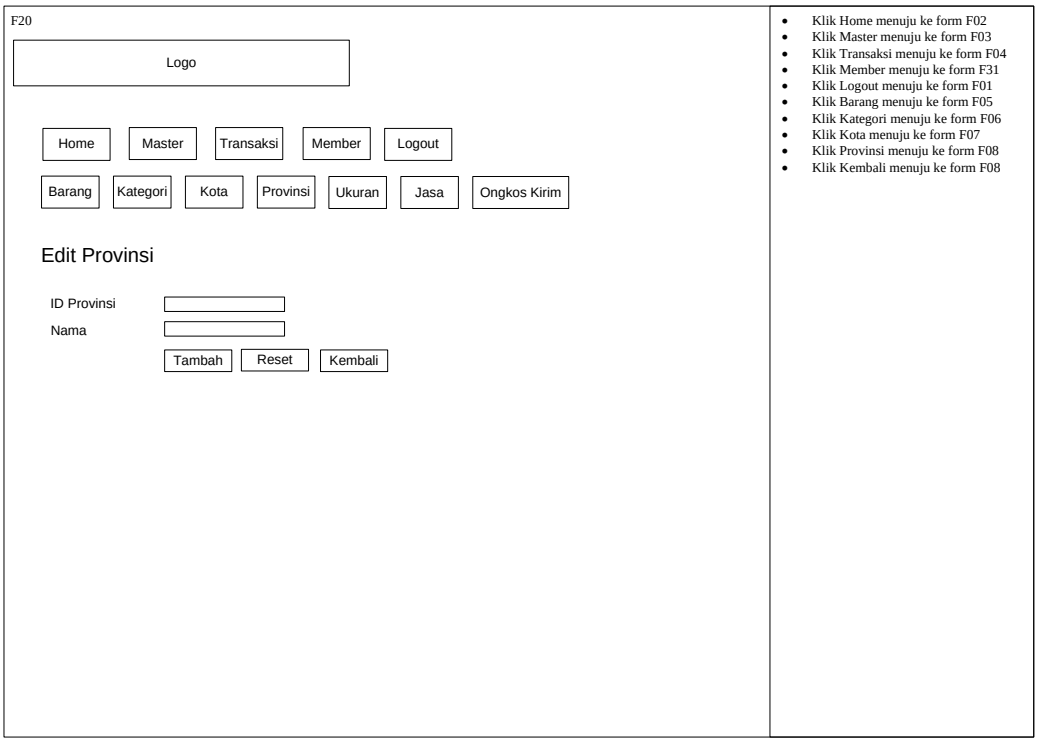

**Gambar 3.68 Form Provinsi.**

17. Form Tambah Provinsi



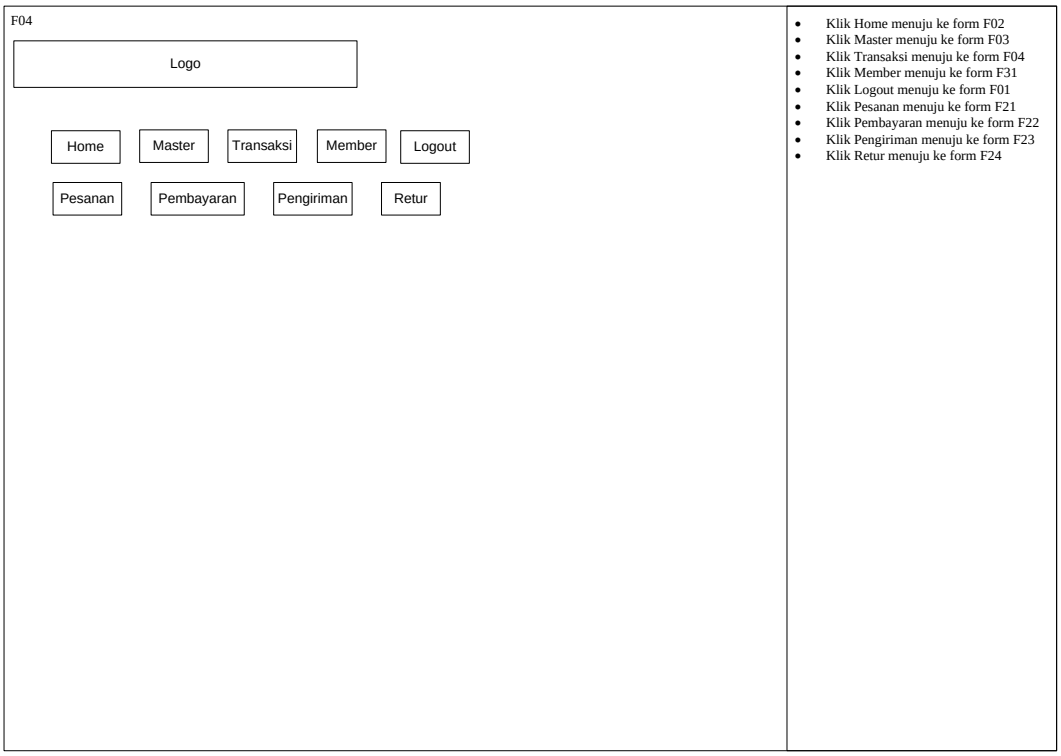

**Gambar 3.69 Form Tambah Provinsi.**

18. Form Edit Provinsi



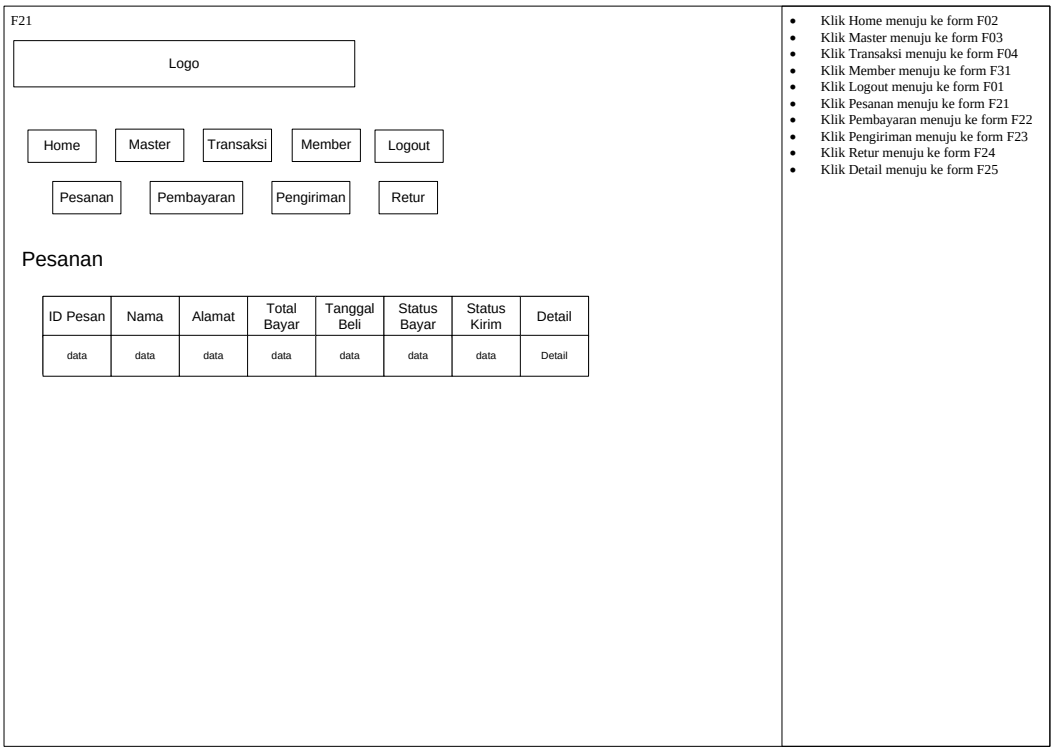

**Gambar 3.70 Form Edit Provinsi.**

19. Form Transaksi



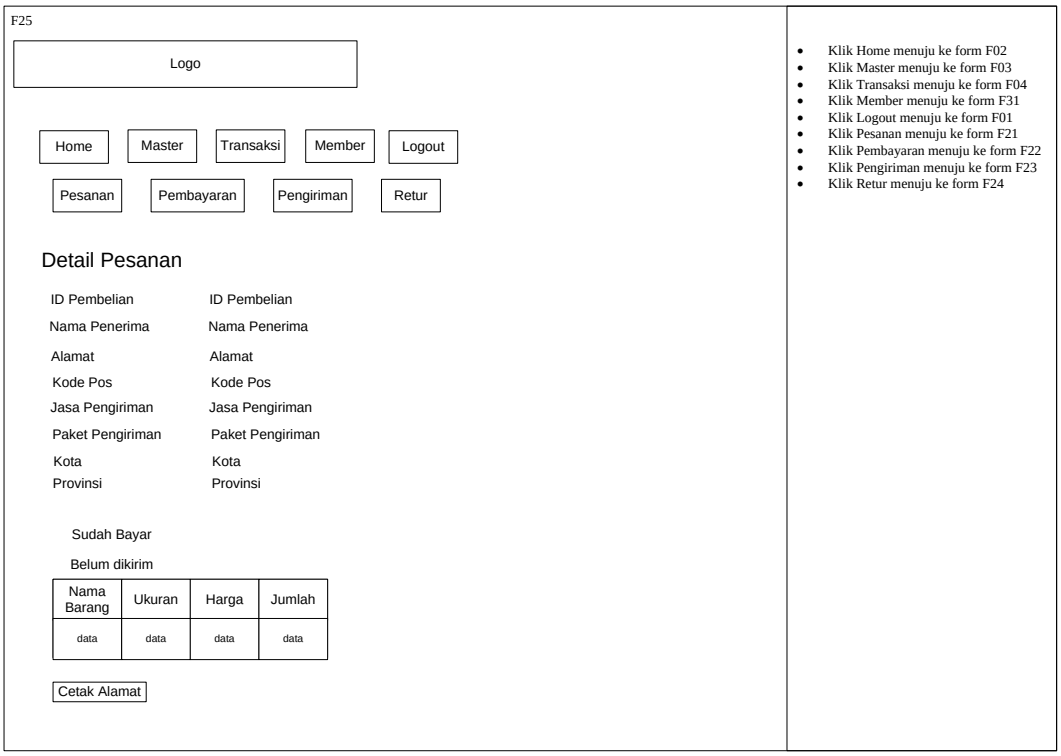

**Gambar 3.71 Form Transaksi.**

20. Form Pemesanan



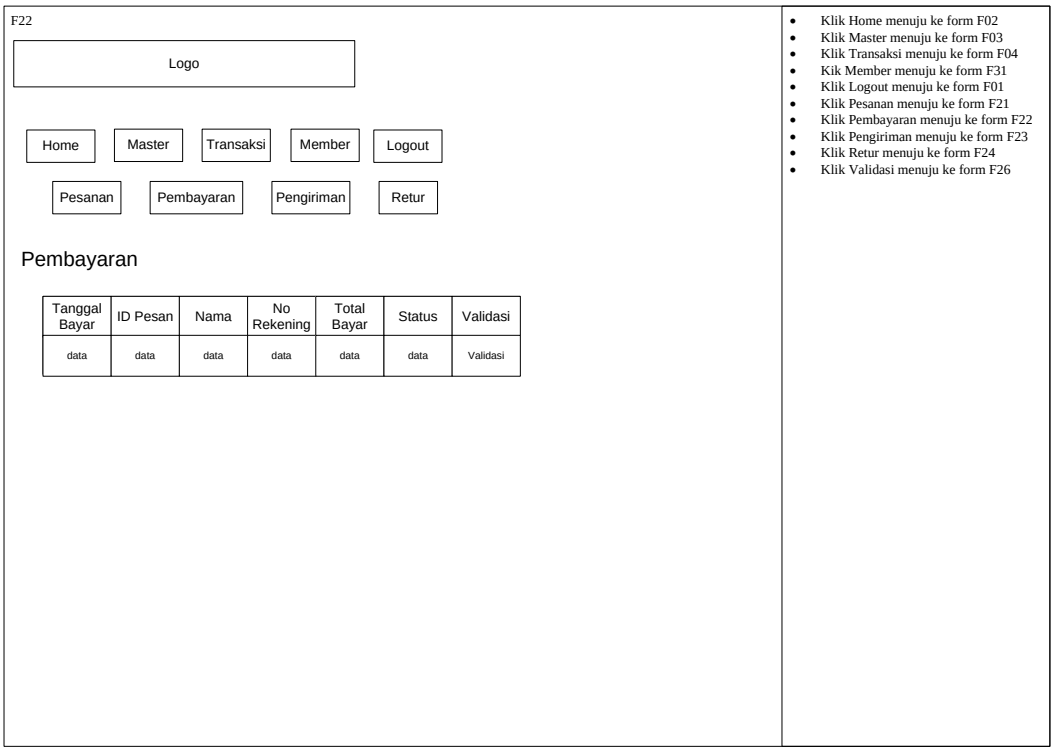

**Gambar 3.72 Form Pemesanan.**

21. Form Detail Pemesanan



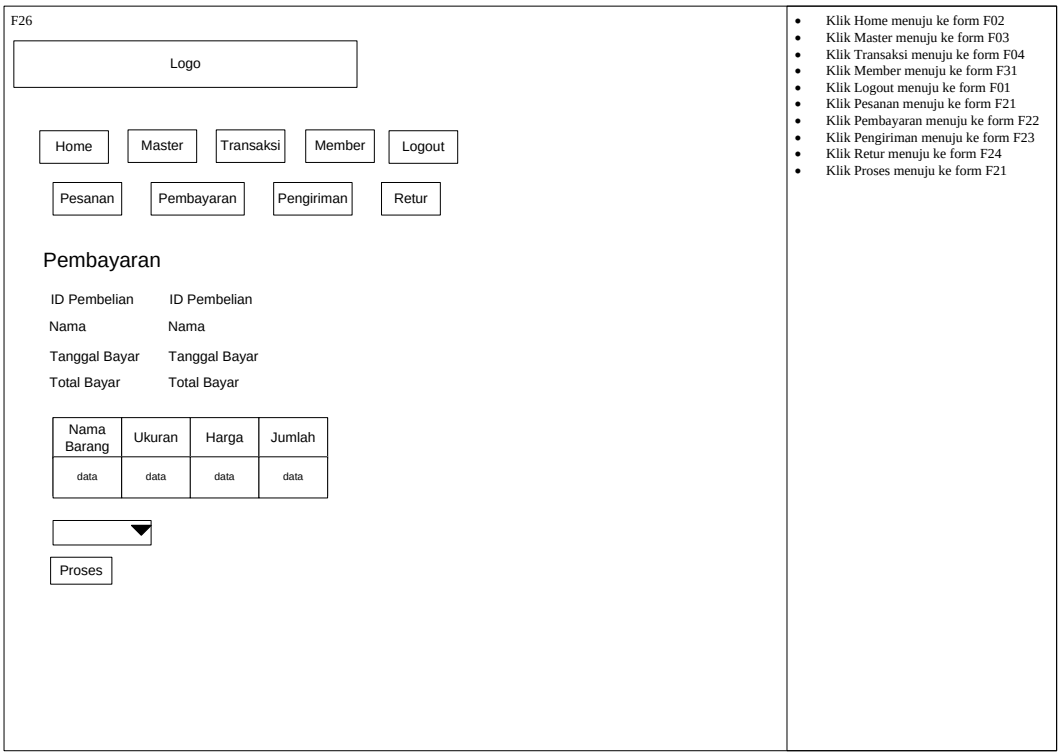

**Gambar 3.73 Form Detail Pemesanan.**

22. Form Pembayaran



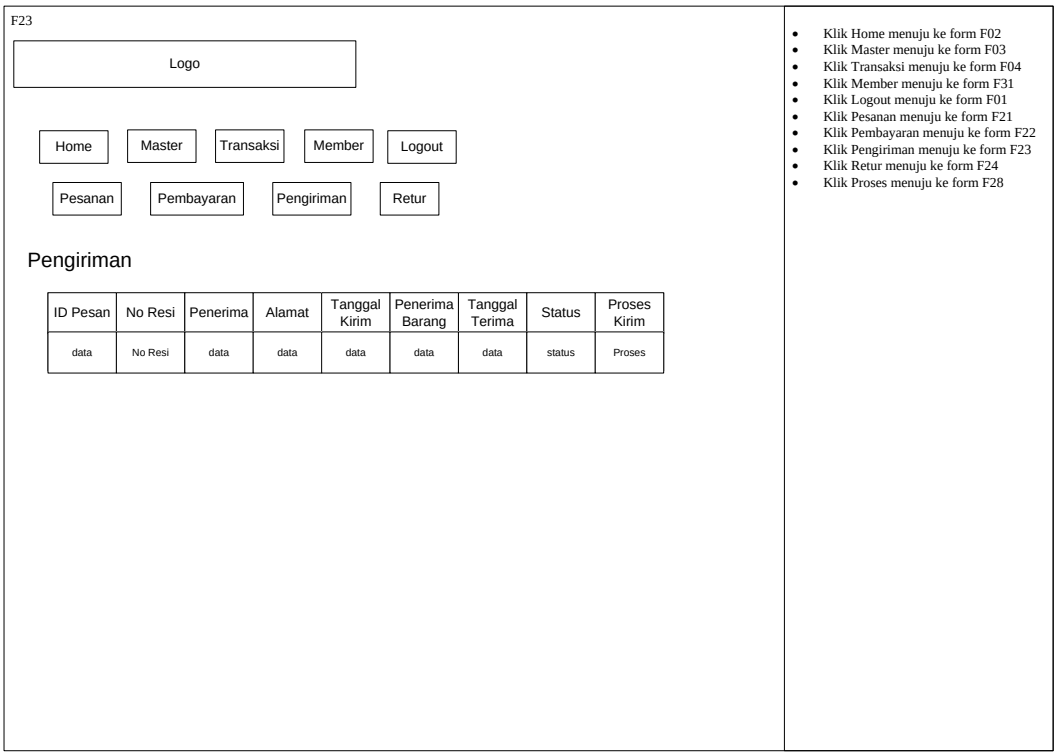

**Gambar 3.74 Form Pembayaran.**

23. Form *Valid*asi Pembayaran



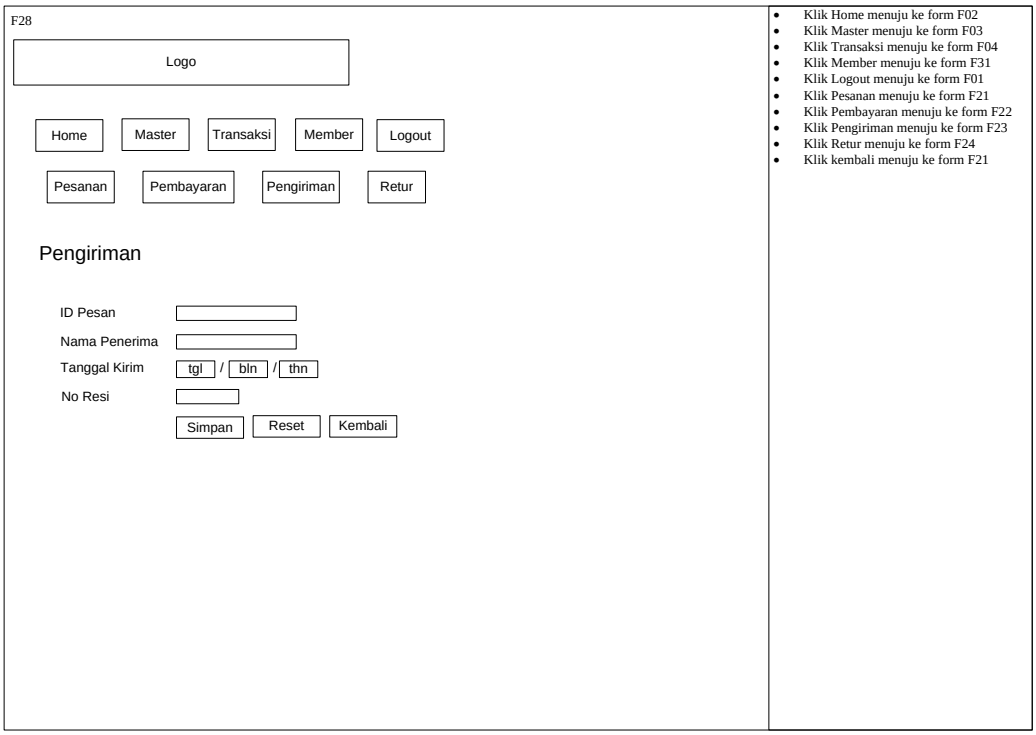

**Gambar 3.75 Form *Valid*asi Pembayaran.**

24. Form Pengiriman



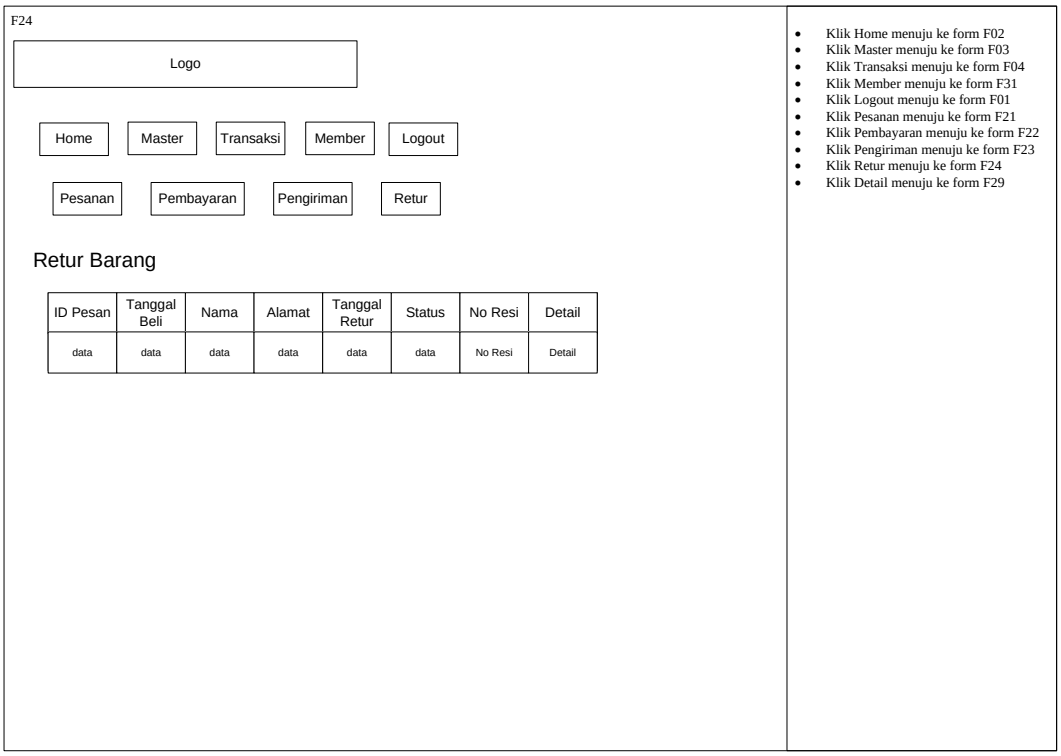

**Gambar 3.76 Form Pengiriman.**

25. Form Proses Pengiriman



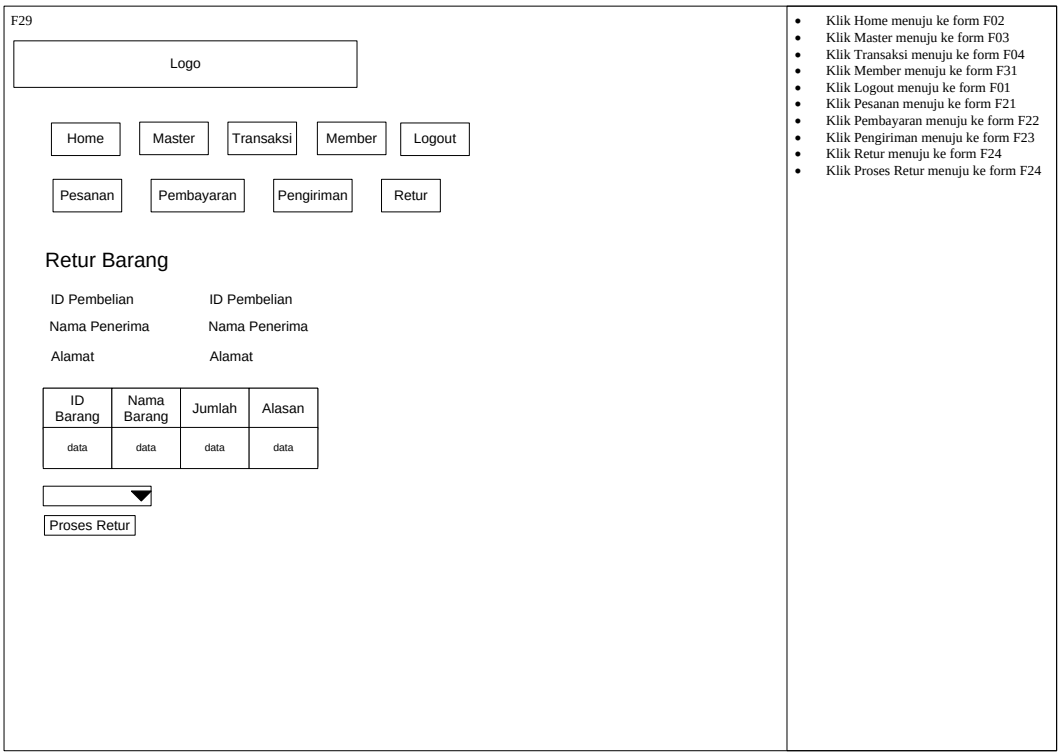

**Gambar 3.77 Form Proses Pengiriman.**

26. Form Retur



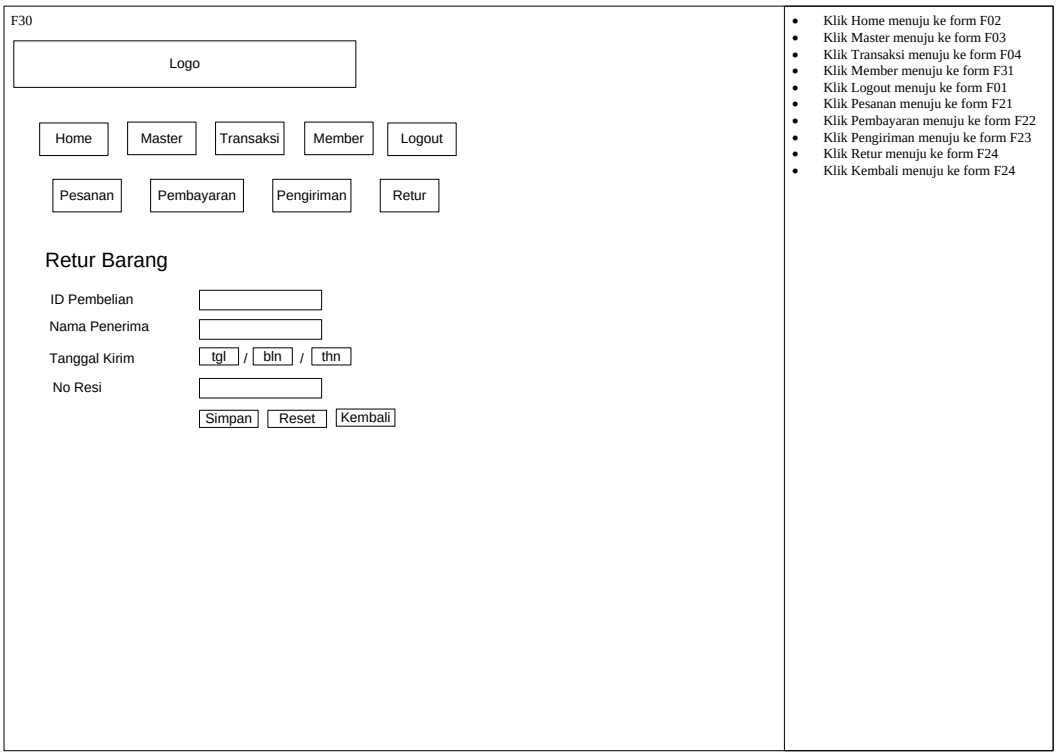

**Gambar 3.78 Form Retur.**

27. Form Proses Retur



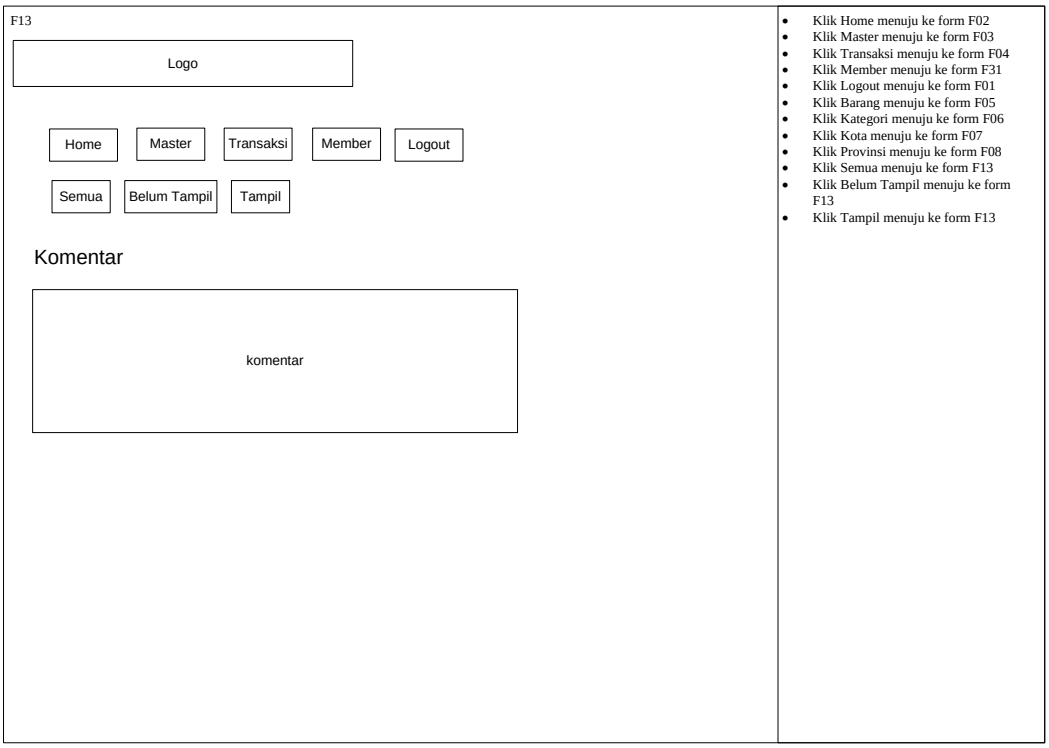

**Gambar 3.79 Form Proses Retur.**

28. Form Pengiriman Retur



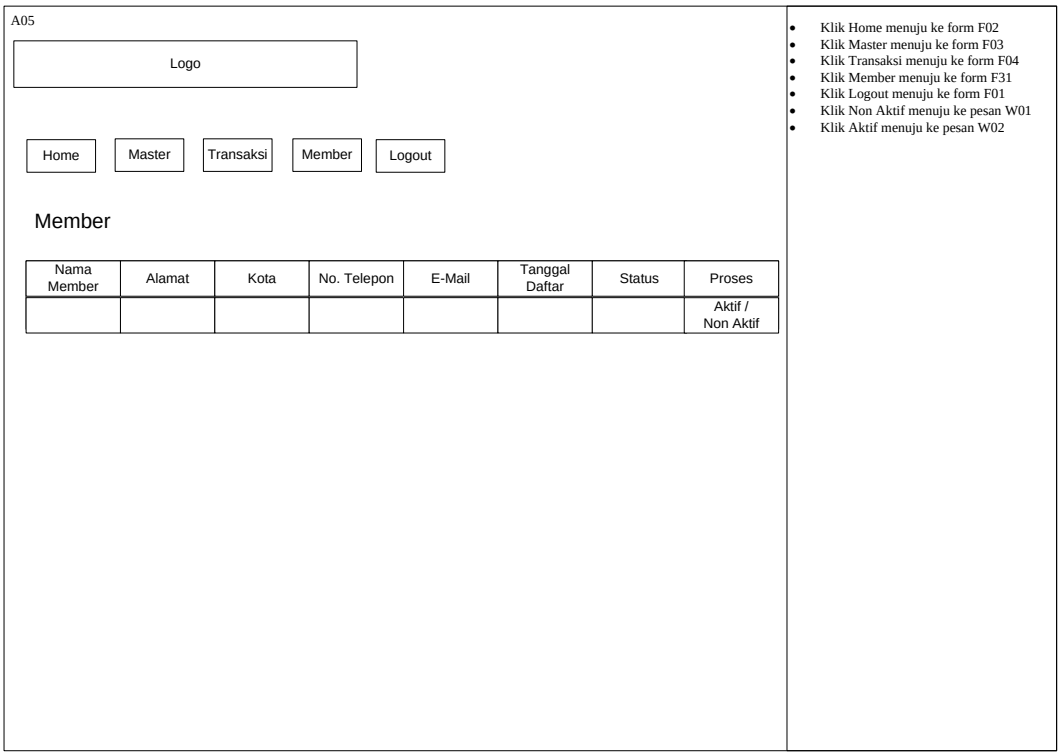

**Gambar 3.80 Form Pengiriman Retur.**

29. Form Komentar




**Gambar 3.81 Form Komentar.**

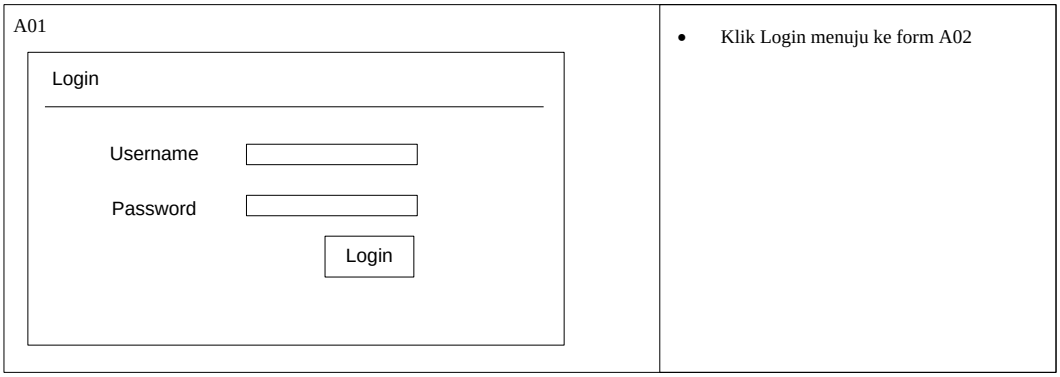
30. Form *Member*




**Gambar 3.82 Form *Member*.**

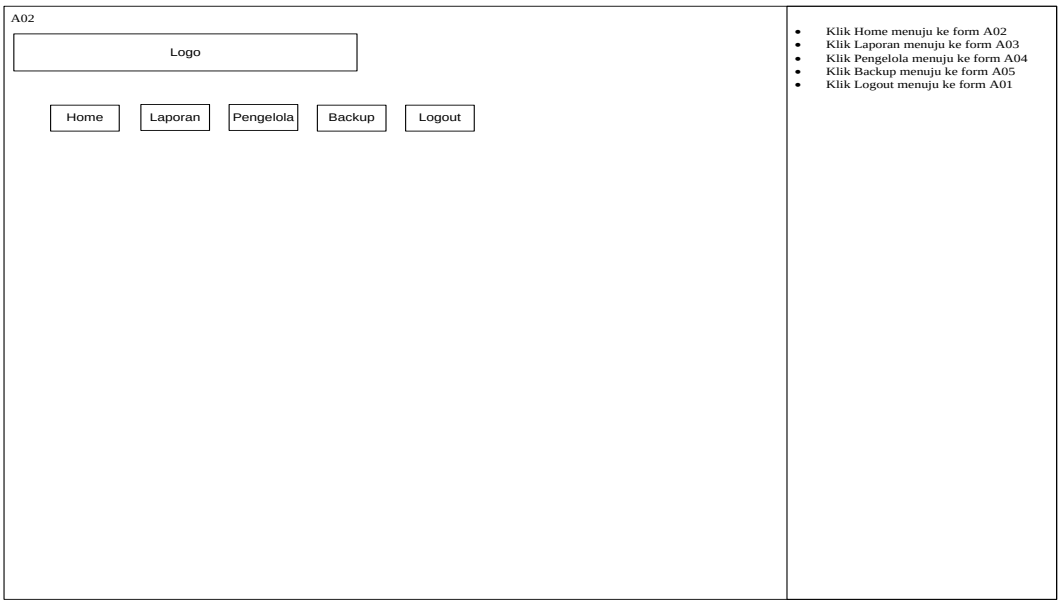
### 3.2.2.5 Perancangan Antarmuka Admin

1. Form *Login*



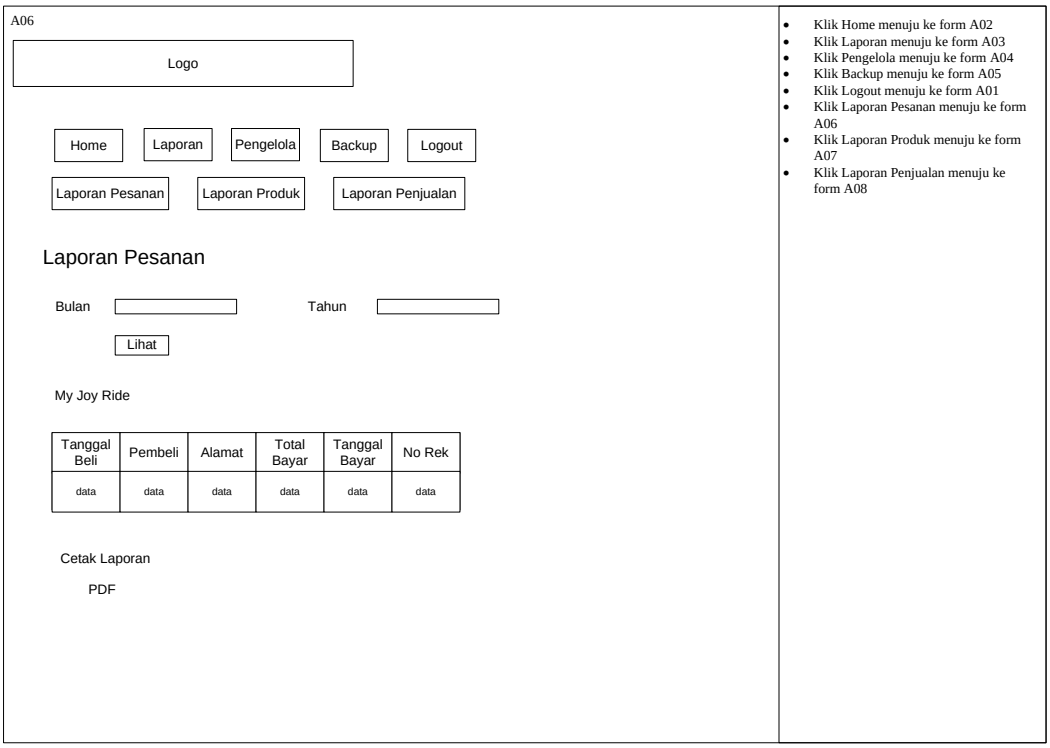

**Gambar 3.83 Form *Login*.**

2. Form *Home*



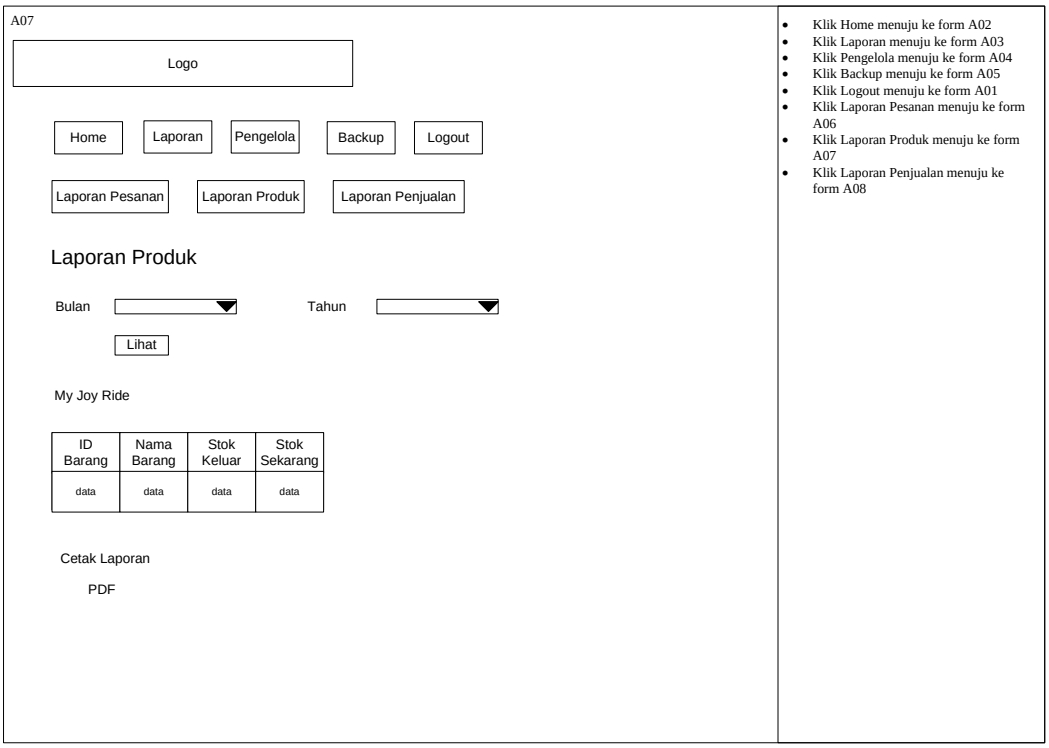

**Gambar 3.84 Form *Home*.**

3. Form Laporan Pemesanan



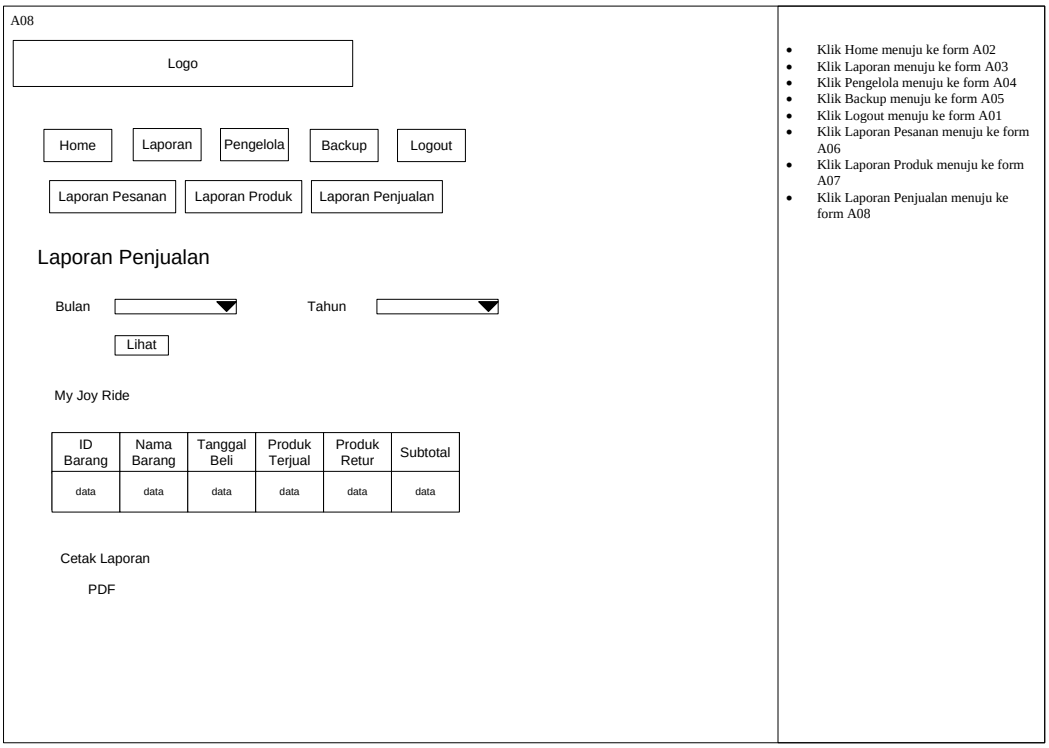

**Gambar 3.85 Form Laporan Pemesanan.**

4. Form Laporan Produk



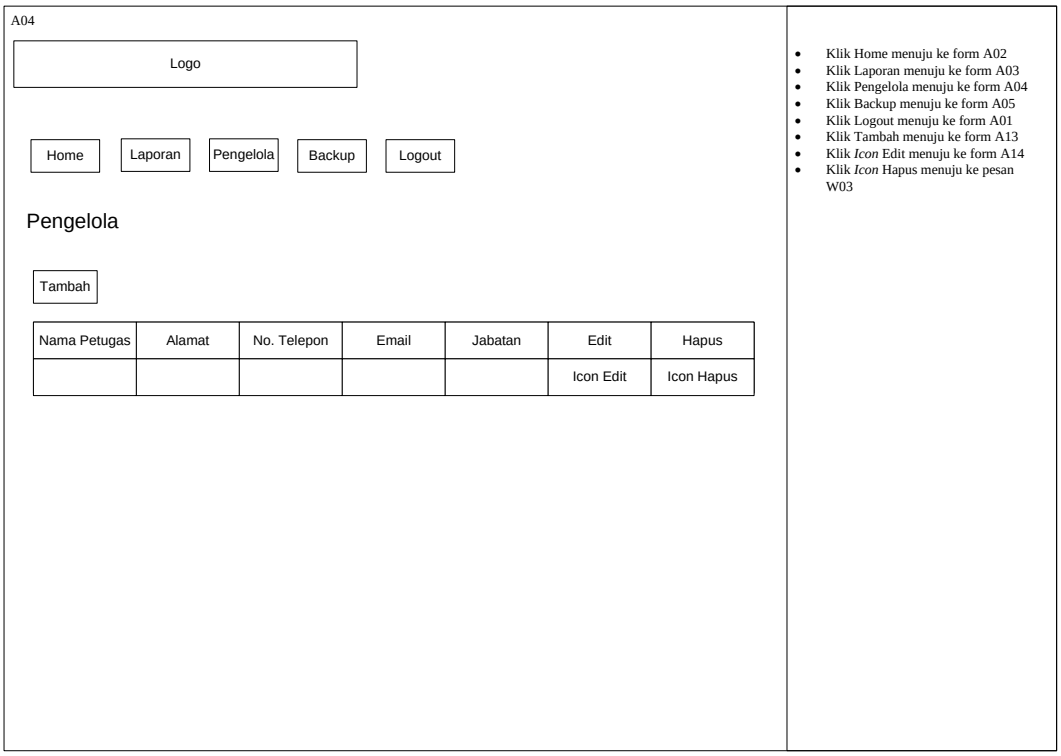

**Gambar 3.86 Form Laporan Produk.**

5. Form Laporan Penjualan



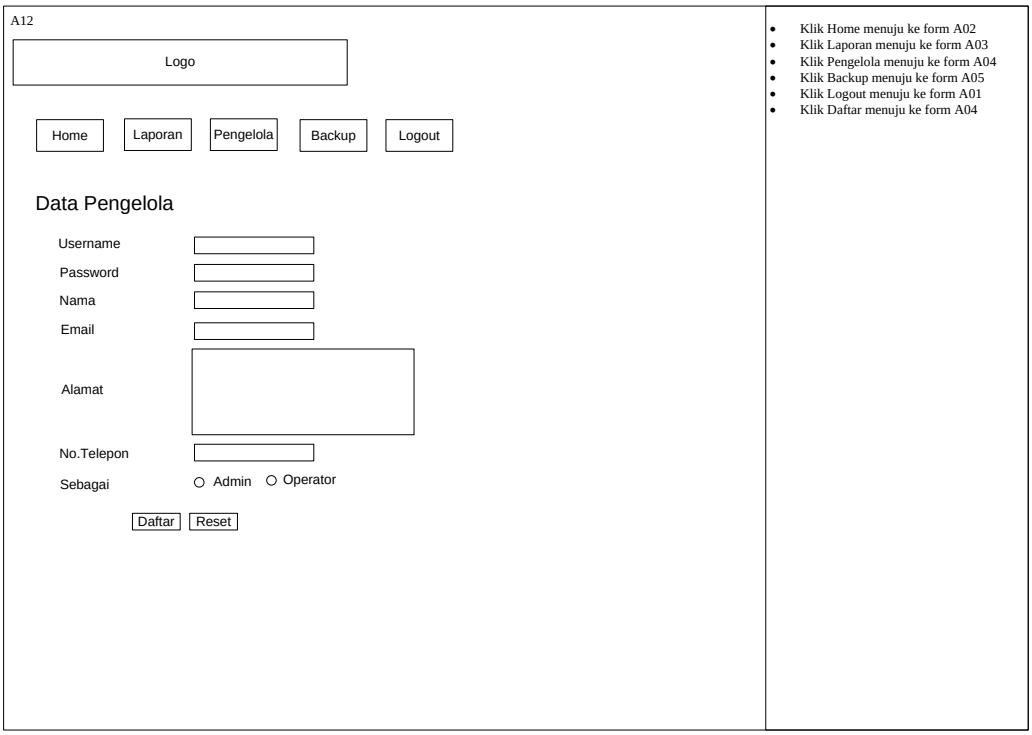

**Gambar 3.87 Form Laporan Penjualan.**

6. Form Pengelola



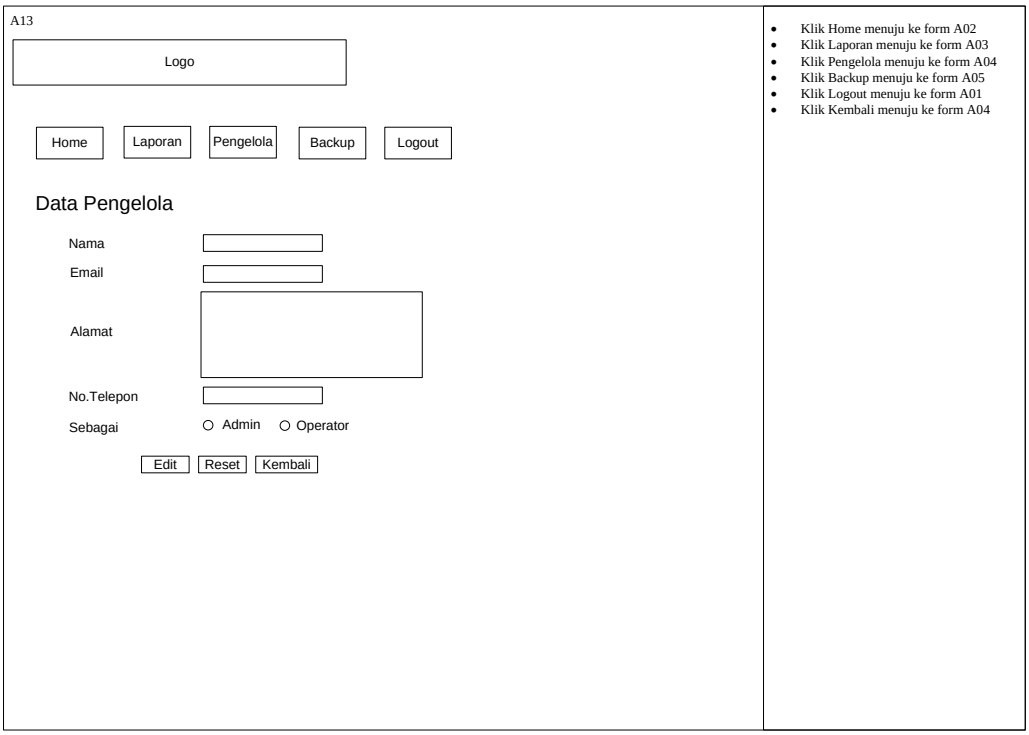

**Gambar 3.88 Form Pengelola.**

7. Form Tambah Pengelola




**Gambar 3.89 Form Tambah Pengelola.**

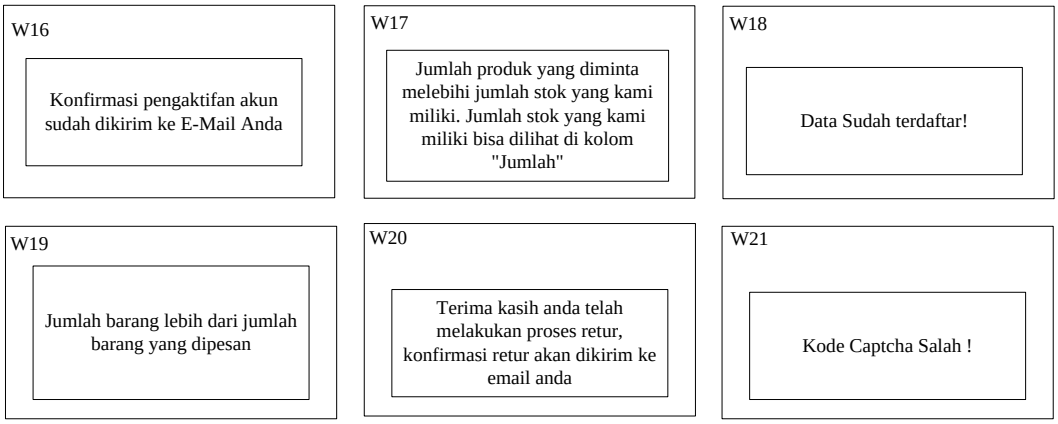
8. Form Edit Pengelola




**Gambar 3.90 Form Edit Pengelola.**

### 3.2.2.6 Perancangan Antarmuka Pesan.



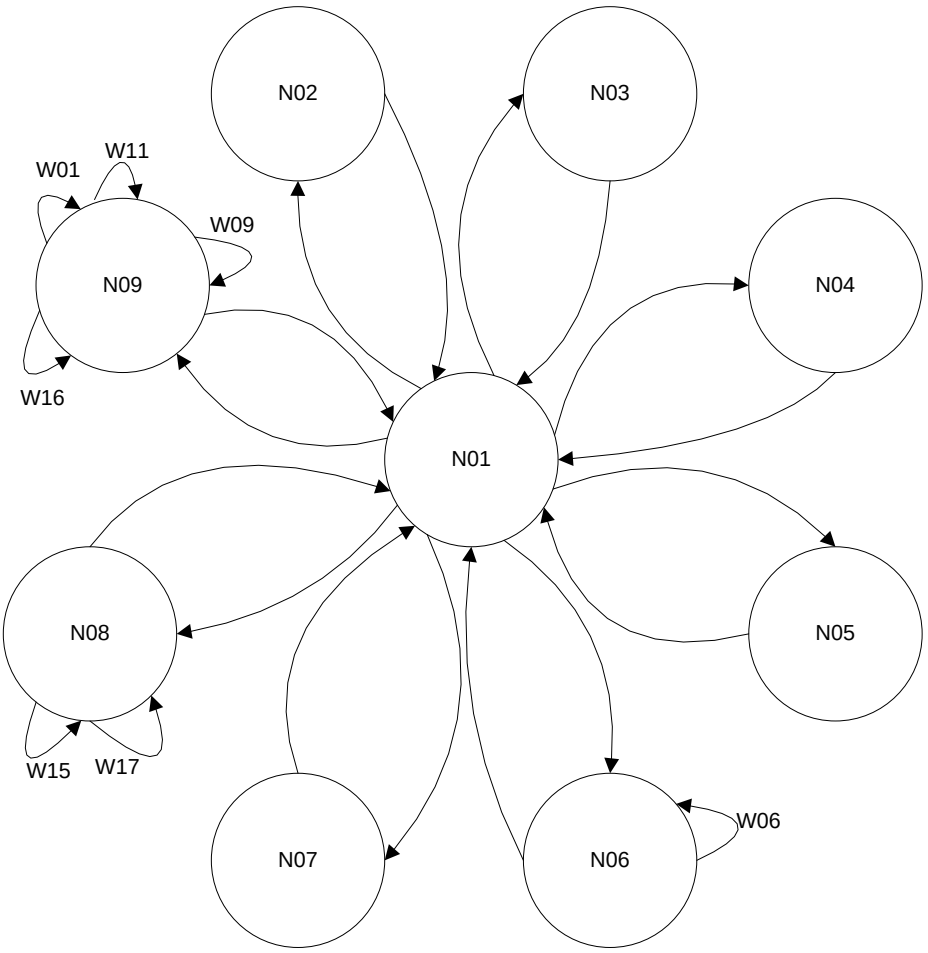



**Gambar 3.91 Perancangan antarmuka Pesan.**

### 3.2.2.7 Jaringan Semantik

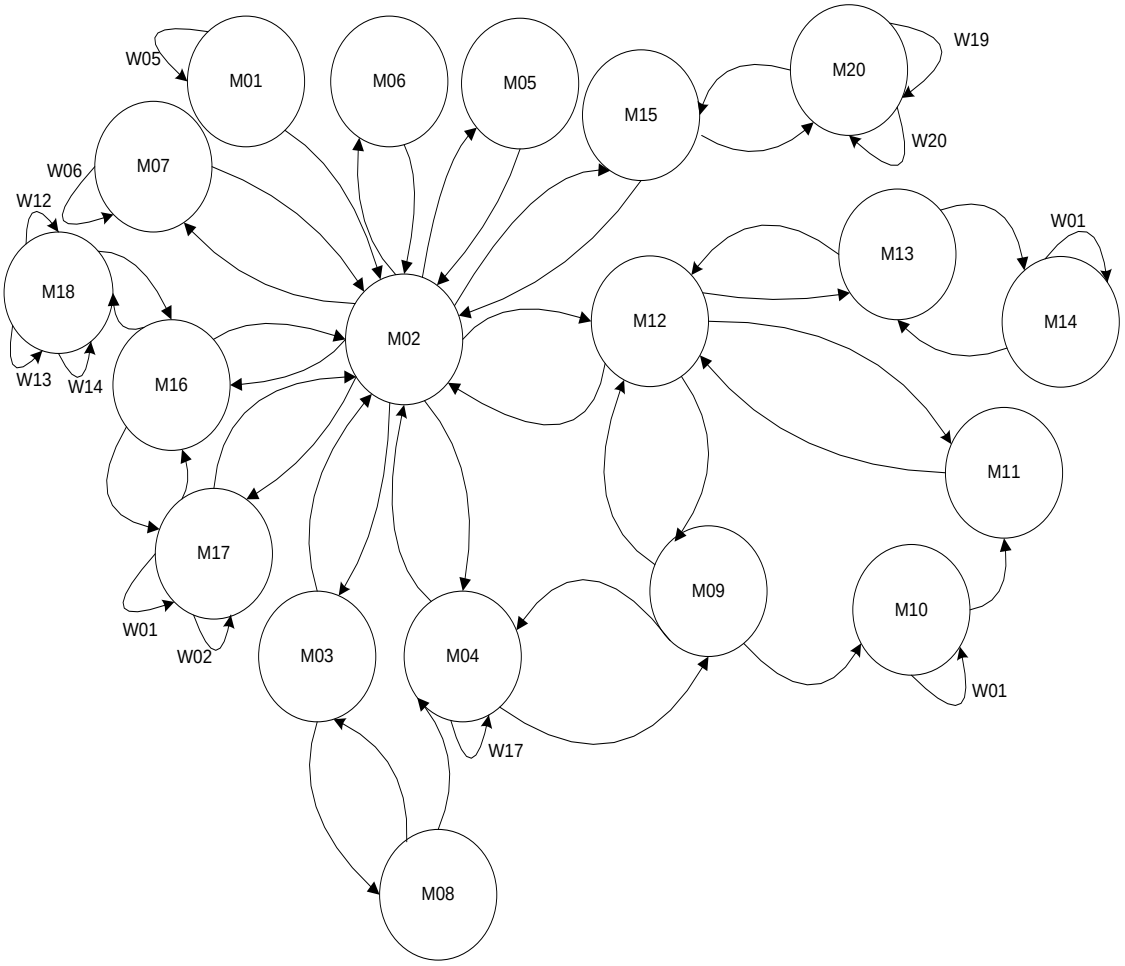
Berikut ini adalah gambar aliran dari menu-menu yang ada dalam aplikasi.

1. Jaringan Semantik Pengunjung Non *Member*



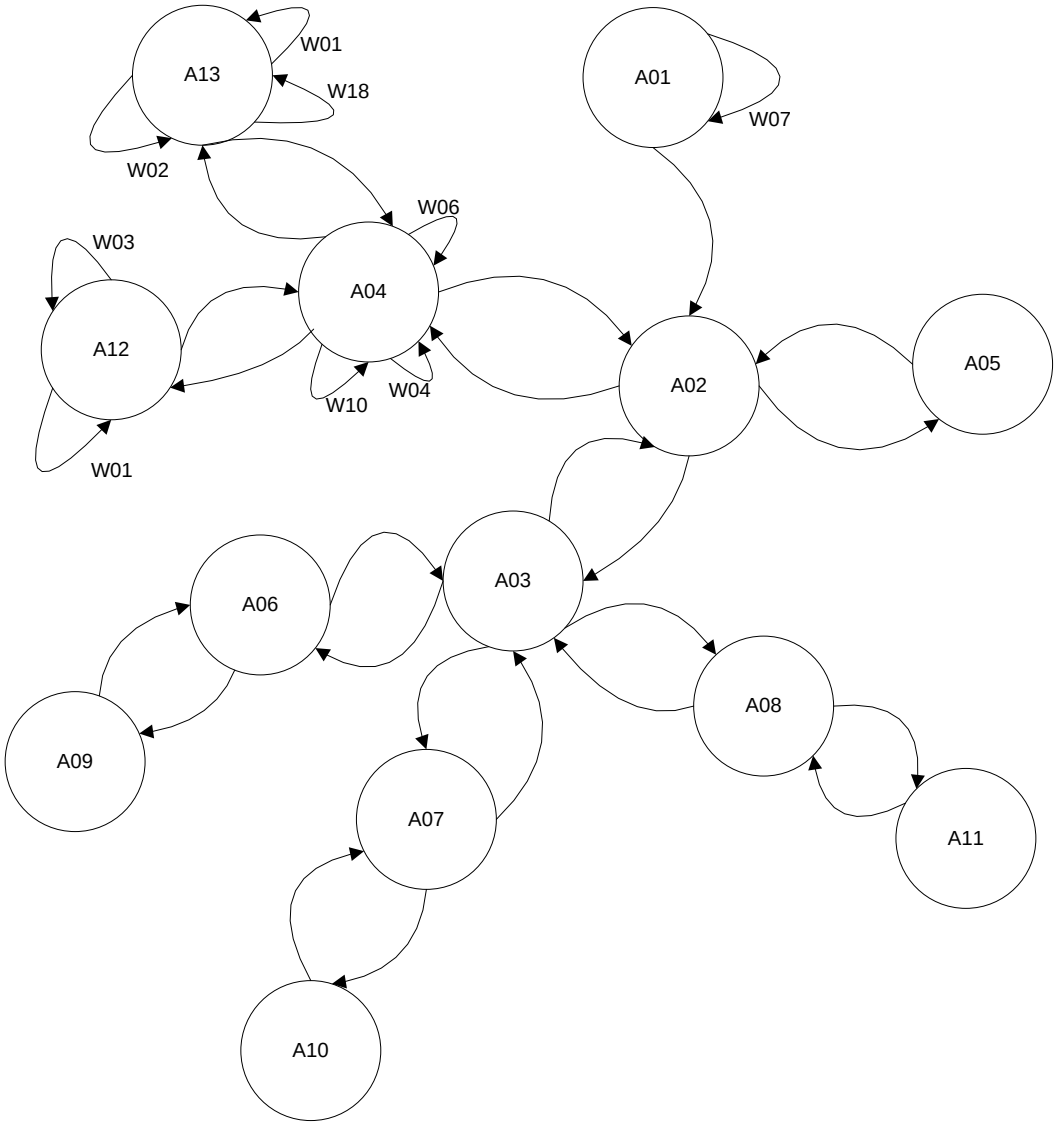

**Gambar 3.92 Jaringan Semantik Pengunjung Non *Member*.**

2. Jaringan Semantik Pengunjung *Member*



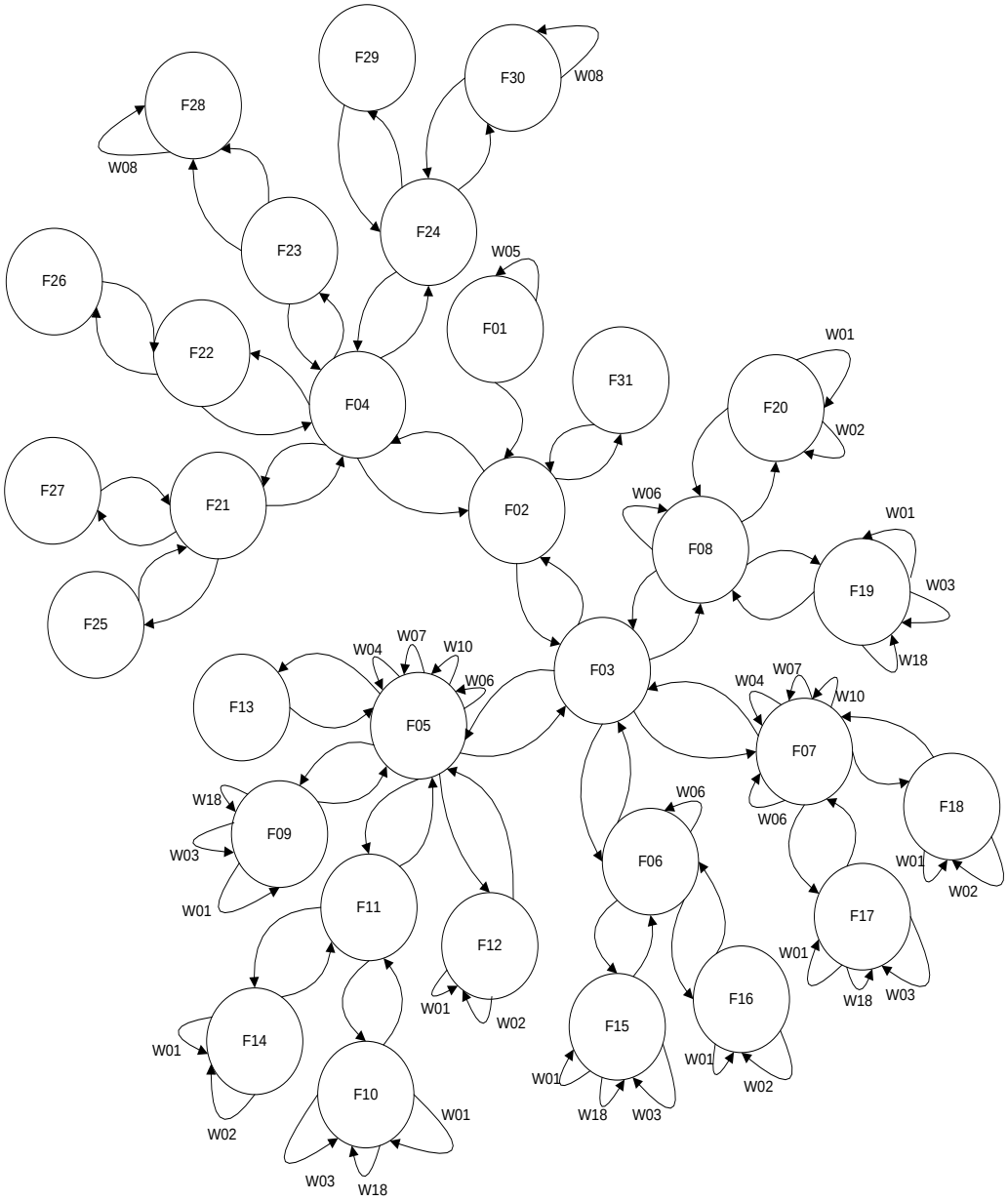

**Gambar 3.93 Jaringan Semantik Pengunjung *Member*.**

3. Jaringan Semantik Admin




**Gambar 3.94 Jaringan Semantik Admin.**

4. Jaringan Semantik Operator



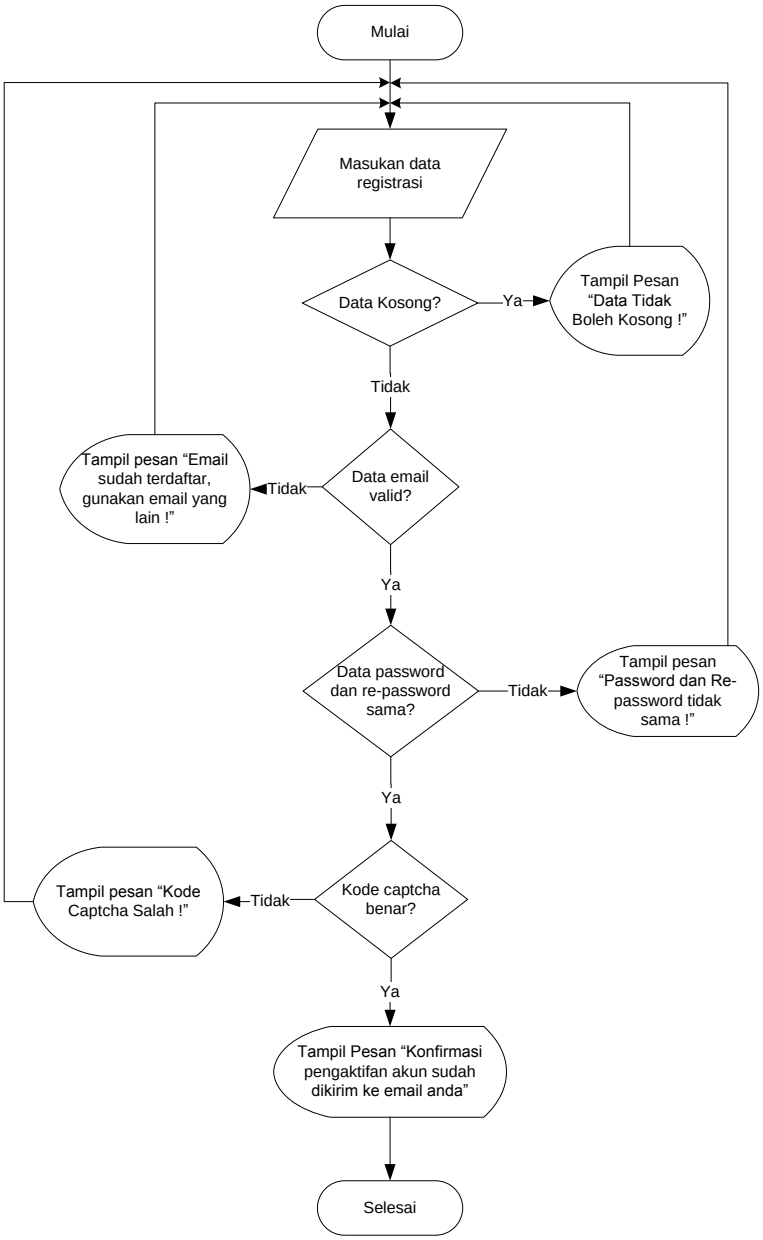

**Gambar 3.95 Jaringan Semantik Operator.**

### 3.1.2 Perancangan Prosedural

Perancangan prosedural merupakan perancangan yang dilakukan untuk menetapkan algoritma yang akan dinyatakan ke dalam suatu program. Adapun perancangan prosedural untuk pembangunan aplikasi *e-commerce* yang akan dibangun adalah sebagai berikut :

1. Prosedural Pendaftaran

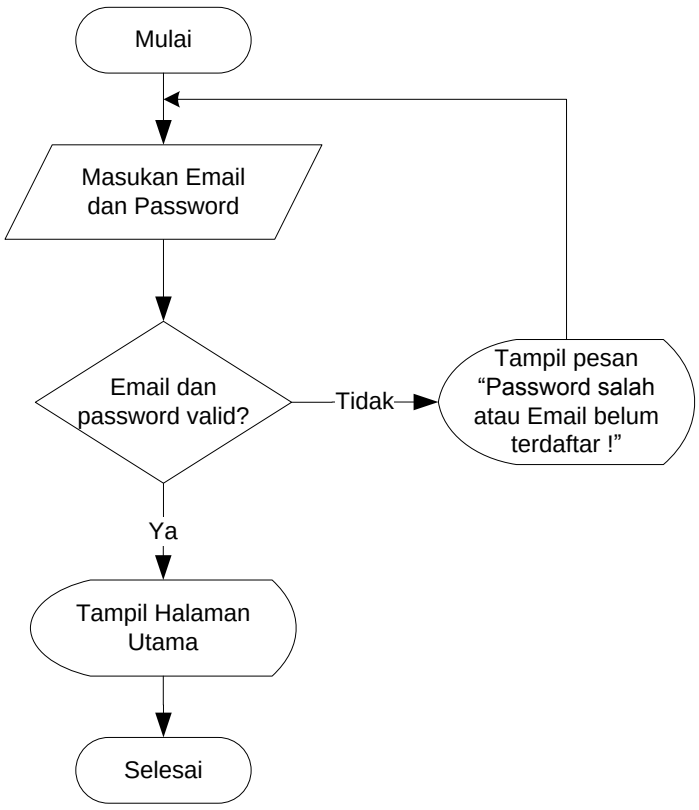
   Prosedural pendaftaran merupakan prosedur yang terjadi ketika pengguna melakukan pendaftaran. Prosedural pendaftaran dapat dilihat pada gambar 3.96 di bawah ini.




**Gambar 3.96 *Flowchart* Pendaftaran**

2. Prosedural Login

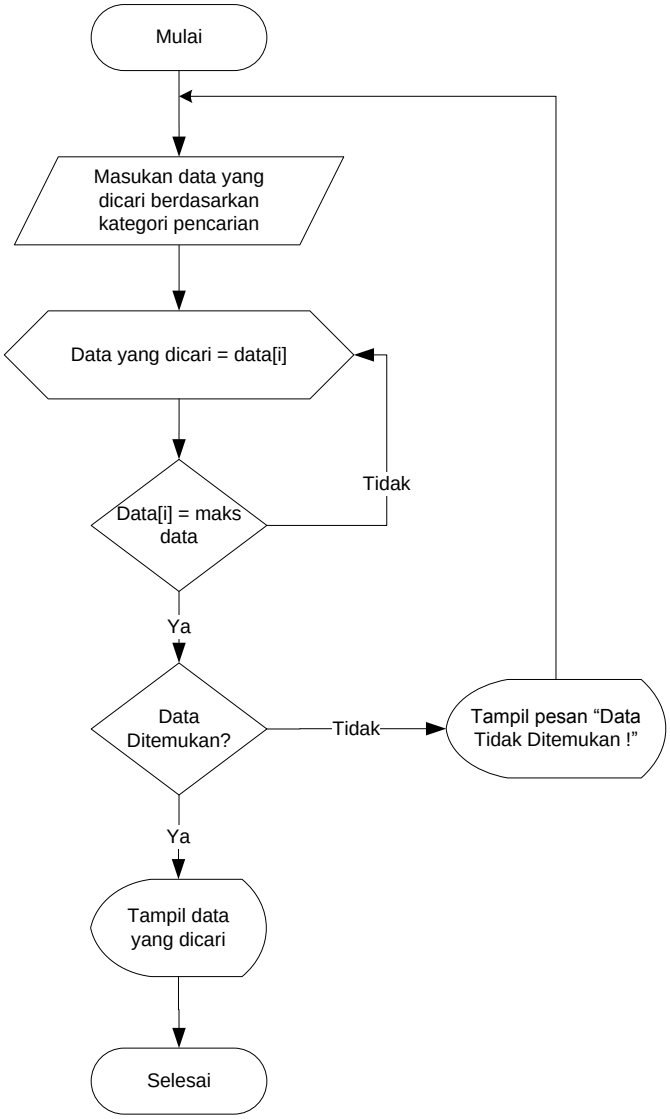
Prosedural login merupakan prosedur yang terjadi ketika pengguna melakukan login. Prosedural login dapat dilihat pada gambar 3.97 di bawah ini.




**Gambar 3.97 *Flowchart* Login**

3. Prosedural Pengubahan Data

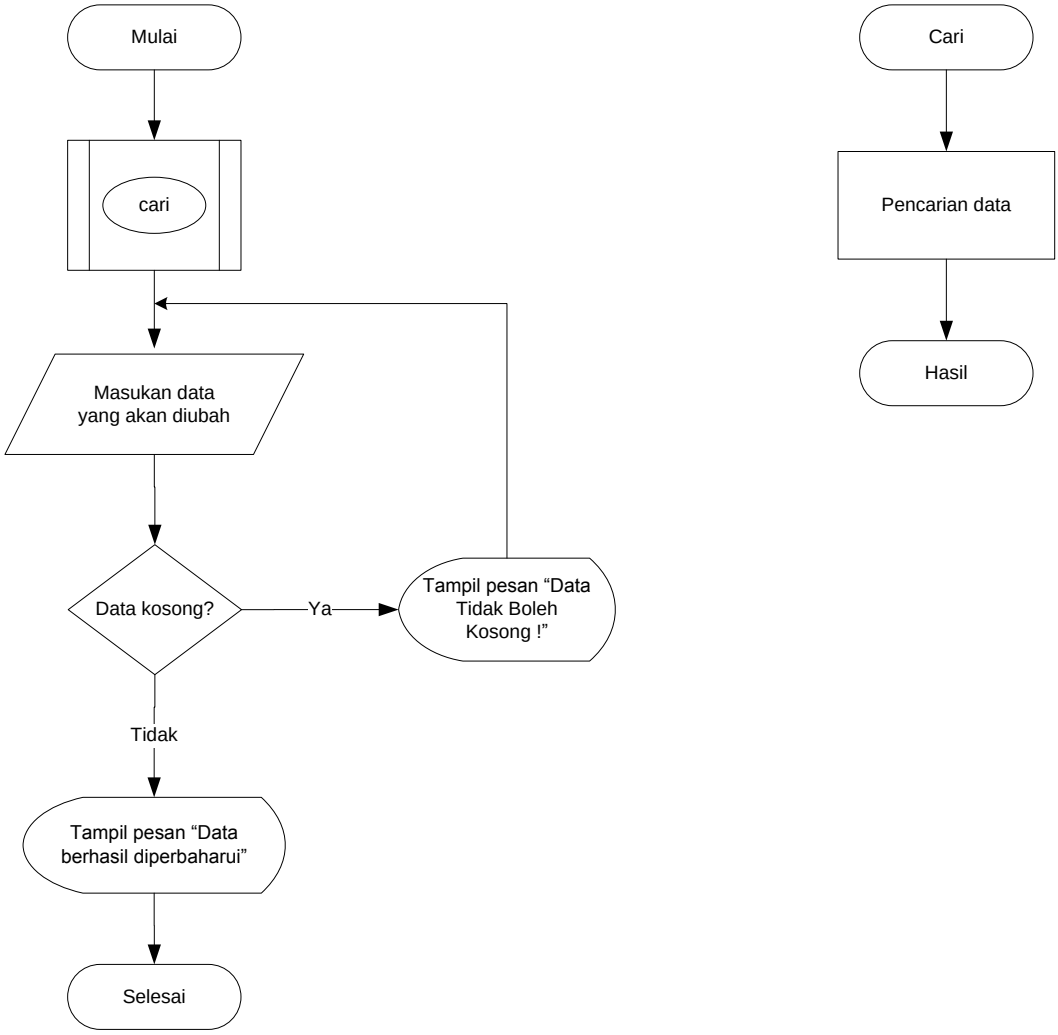
Prosedural pencarian data merupakan prosedur yang terjadi ketika pengguna ingin mencari data. Prosedural pencarian dapat dilihat pada gambar 3.98 di bawah ini.




**Gambar 3.98 *Flowchart* Pencarian Data**

4. Prosedural Pengubahan Data

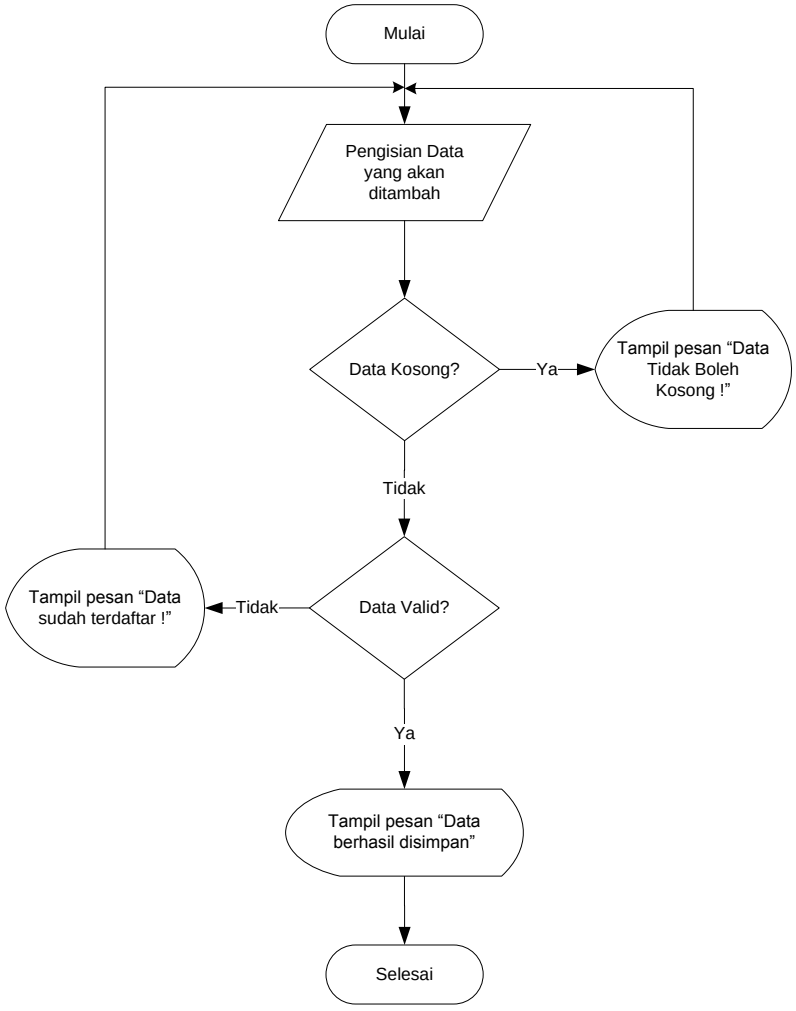
Prosedural pengubahan data merupakan prosedur yang terjadi ketika pengguna ingin mengubah data. Prosedural pengubahan dapat dilihat pada gambar 3.99 di bawah ini.




**Gambar 3.99 *Flowchart* Pengubahan Data**

5. Prosedural Penambahan Data

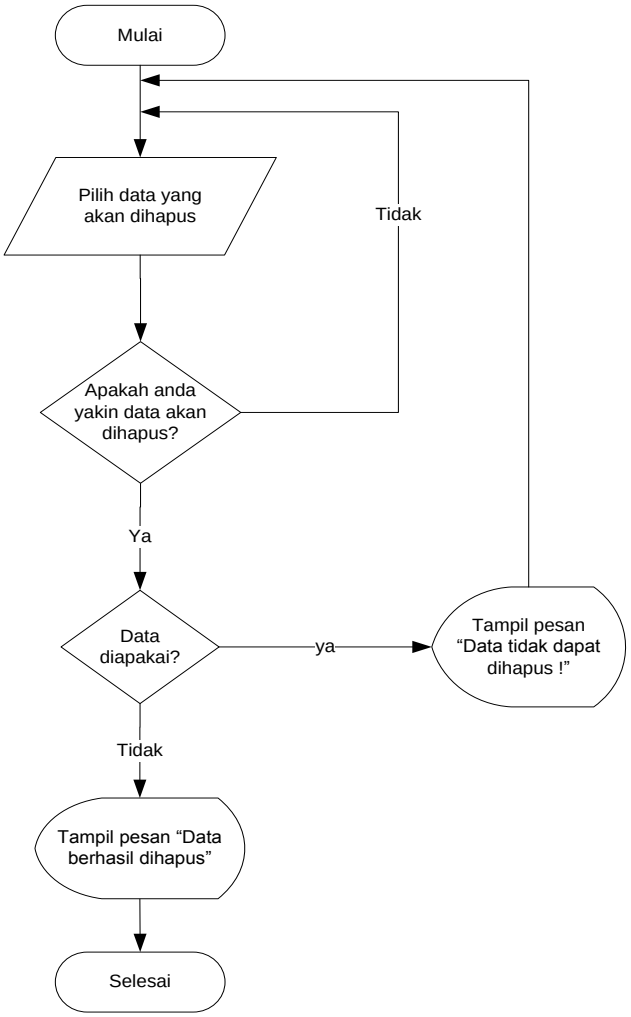
Prosedural penambahan data merupakan prosedur yang terjadi ketika pengguna ingin menambah data. Prosedural penambahan dapat dilihat pada gambar 3.100 di bawah ini.




**Gambar 3.100 *Flowchart* Penambahan Data**

6. Prosedural Penghapusan Data

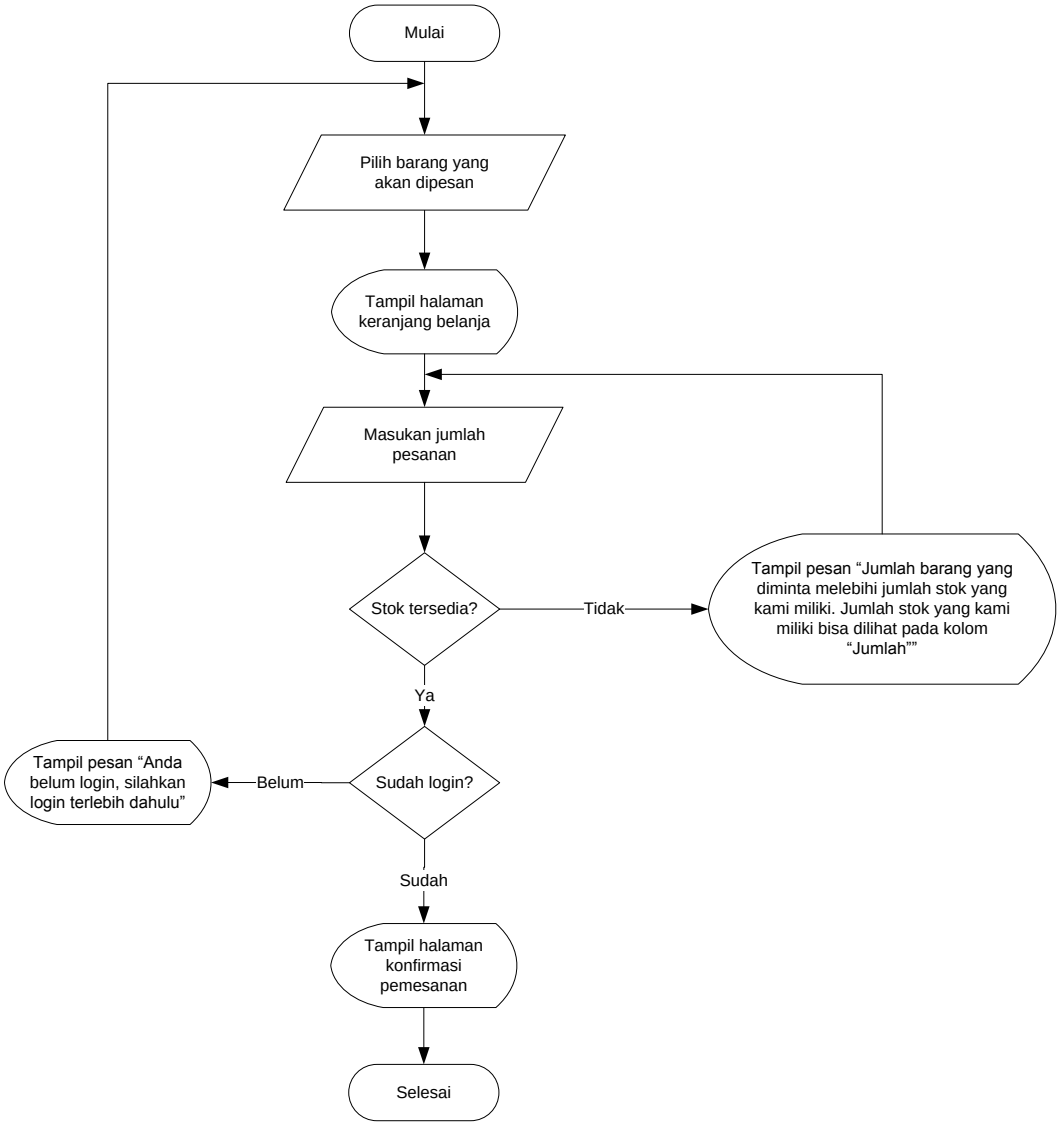
Prosedural penghapusan data merupakan prosedur yang terjadi ketika pengguna ingin menghapus data. Prosedural penghapusan dapat dilihat pada gambar 3.101 di bawah ini.




**Gambar 3.101 *Flowchart* Penghapusan Data**

7. Prosedural Pemesanan

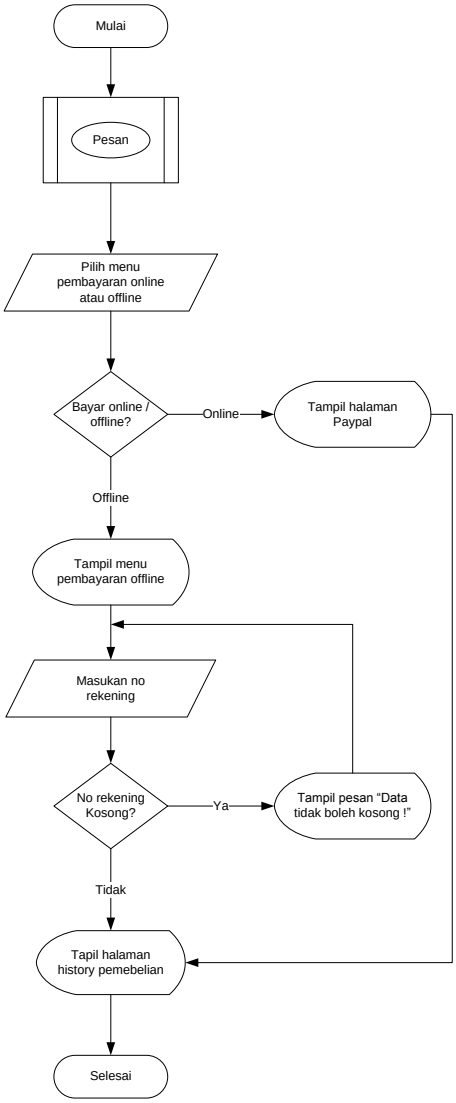
Prosedural pemesanan merupakan prosedur yang terjadi ketika *member* ingin melakukan pemesanan. Prosedural pemesanan dapat dilihat pada gambar 3.102 di bawah ini.




**Gambar 3.102 *Flowchart* Pemesanan**

8. Prosedural Pembayaran

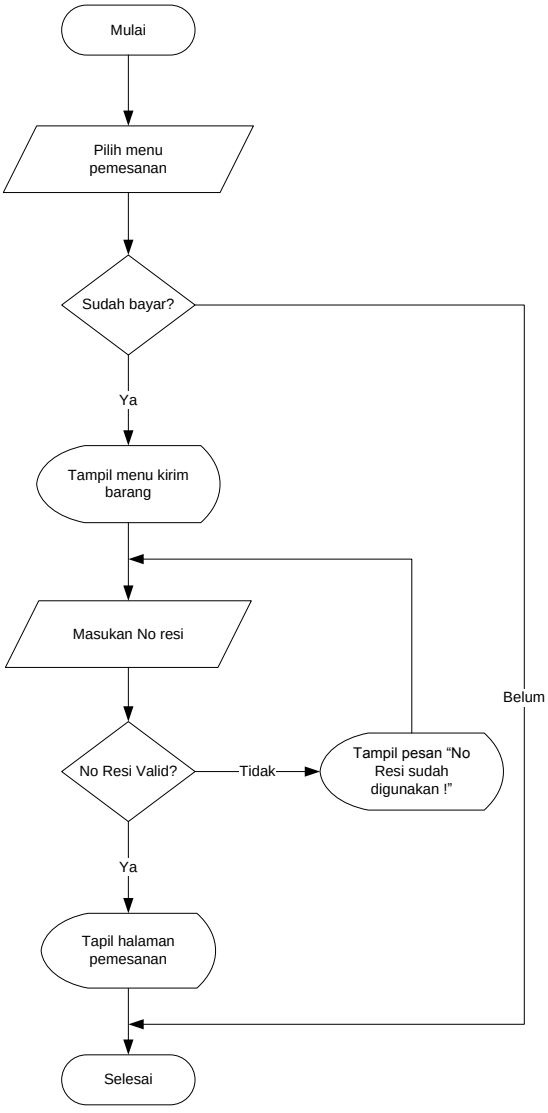
Prosedural pembayaran merupakan prosedur yang terjadi ketika *member* ingin melakukan pembayaran. Prosedural pembayaran dapat dilihat pada gambar 3.103 di bawah ini.




**Gambar 3.103 *Flowchart* Pembayaran**

9. Prosedural Pengiriman

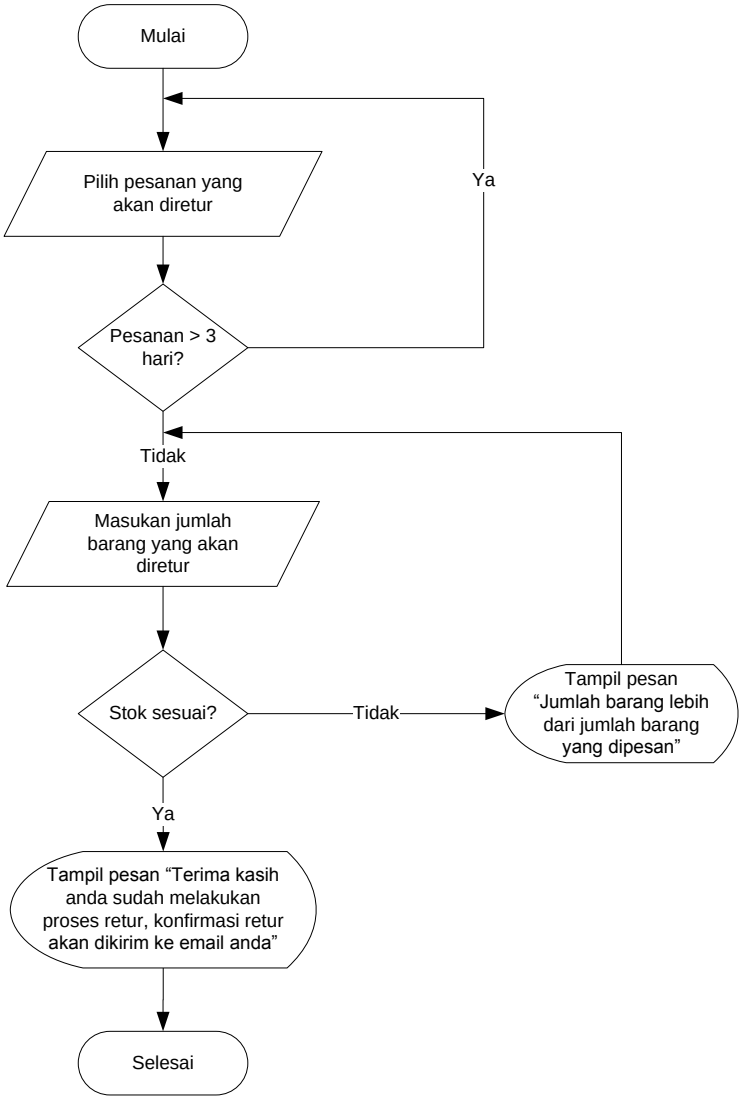
Prosedural pengiriman merupakan prosedur yang terjadi ketika operator ingin melakukan pengiriman. Prosedural pengiriman dapat dilihat pada gambar 3.104 di bawah ini.




**Gambar 3.104 *Flowchart* Pengiriman**

10. Prosedural Retur

Prosedural retur merupakan prosedur yang terjadi ketika *member* ingin melakukan retur. Prosedural retur dapat dilihat pada gambar 3.105 di bawah ini.




**Gambar 3.105 *Flowchart* Retur**