

## **BAB IV**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **4.1 ANALISIS SISTEM**

##### **4.1.1 Gambaran Umum Toko Hani Sejahtera**

Toko Hani Sejahtera adalah merupakan toko yang menjual perlengkapan kebutuhan masyarakat (sembako). Toko Hani Sejahtera didirikan pada tahun 2000. Yang berlokasi Jln. Tembesi-Sarulangun, Kelurahan Durian Luncuk Kecamatan Batin XXIV Kabupaten Batanghari. Dengan pemilik Muhammad Sukni bentuk usaha yaitu Toko yang Menjual perlengkapan kebutuhan masyarakat (sembako).

Dikarenakan persaingan yang semakin ketat dan semakin meningkatnya permintaan pelanggan Toko Hani Sejahtera berusaha untuk meningkatkan penjualan dan promosi guna memberikan kemudahan bagi para pelanggan dalam melakukan transaksi dengan pihak Toko Hani Sejahtera juga berusaha meningkatkan sistem pengolahan data yang lebih baik guna memperoleh informasi yang cepat dan akurat sehingga dapat mengurangi kesalahan dalam pengambilan keputusan.

##### **4.1.2. Analisis Sistem Yang Berjalan**

Toko Hani Sejahtera merupakan toko penjualan online yang berada di daerah Tembesi Sarulangun yang bergerak dalam penjualan perlengkapan kebutuhan masyarakat (sembako). Toko Hani Sejahtera dalam menjalankan

kegiatannya terutama dalam promosi dan penjualan masih menggunakan cara konvensional, yaitu :

1. Dalam Promosi

Toko Hani Sejahtera dalam memberikan promosi mengenai barang yang ditawarkan kepada pelanggan masih menggunakan promosi melalui pelanggan sehingga terkadang informasi mengenai nama barang, harga barang, dan keterangan mengenai produk sering terjadi kesalahan pemahaman antara pihak toko dengan pihak pelanggan.

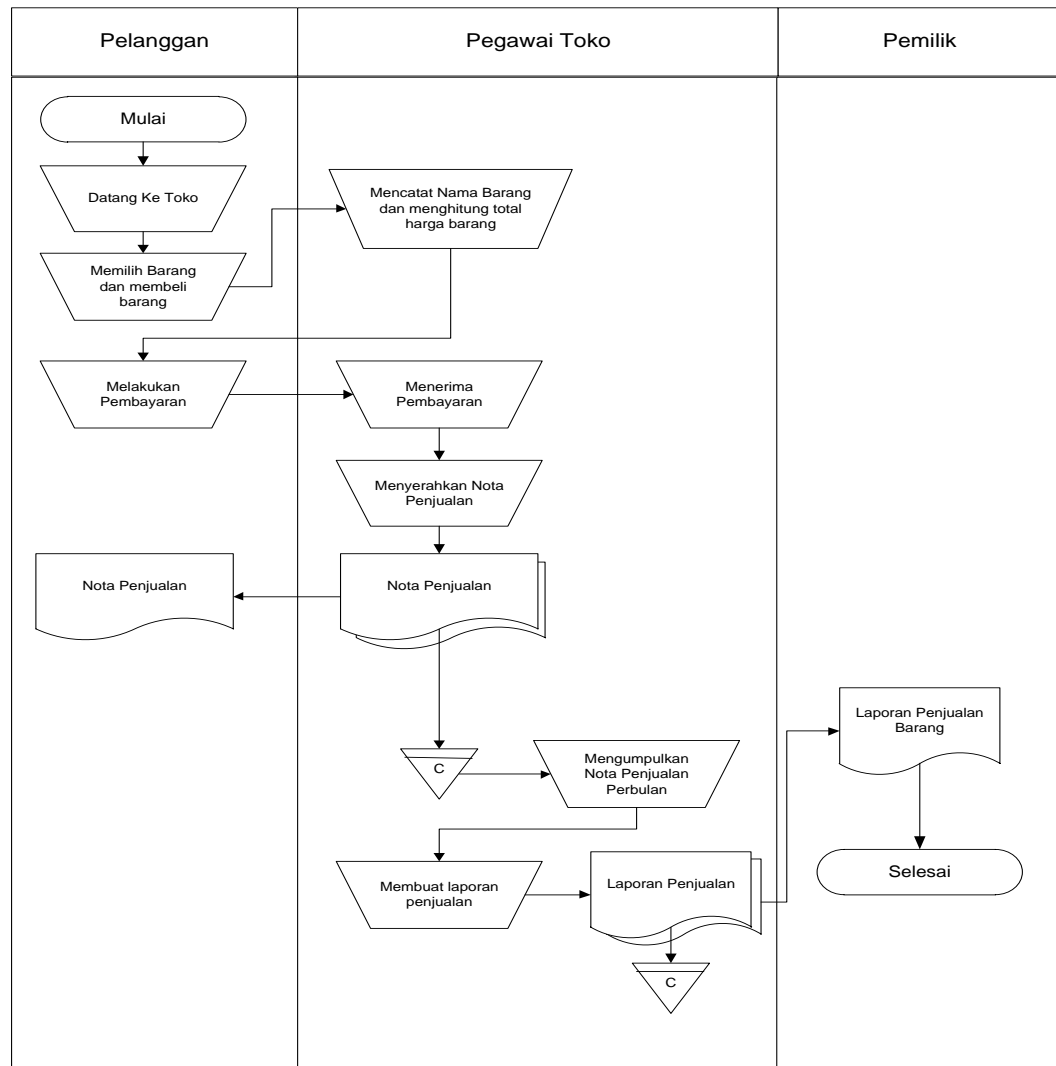
2. Dalam Penjualan

Toko Hani Sejahtera dalam melakukan transaksi penjualan, masih menunggu kedatangan pelanggan untuk membeli barang dan setelah pelanggan datang kemudian membelikan barang pihak pegawai toko mencatat nama barang yang dibeli dan menghitung total harga barang dengan kalkulator, selanjutnya menerima pembayaran dari pelanggan dan menyerahkan nota penjualan atau transfer sebagai bukti pembayaran. Dari setiap pemesanan akan dicatat menjadi laporan yang dibuat ke dalam agenda pemesanan dan nantinya akan dilihat pemilik toko setiap harinya.

Sehingga ditemukan beberapa kekurangan atau kelemahan dalam sistem yang sedang berjalan pada Toko Hani Sejahtera diantaranya yaitu :

1. Sulitnya meningkatkan omset penjualan dalam jumlah yang besar.
2. Kurangnya sarana promosi yang dapat dilakukan dengan biaya yang murah.

3. Kurangnya sarana dalam memberikan informasi yang tepat mengenai barang-barang yang ditawarkan kepada pelanggan.



**Gambar 4.1 Flowchart Sistem yang Sedang Berjalan**

#### 4.1.4 Solusi Pemecahan Masalah

Berdasarkan analisis permasalahan yang ada, dirancanglah sebuah sistem yang dapat memberi solusi dari masalah yang ada. Berikut adalah spesifikasi sistem baru yang dirancang :

1. Merancang sistem memudahkan pelanggan untuk memesan barang secara *online* yang mencakup seluruh wilayah Indonesia dan mengelola data produk yang ditawarkan di dalam *website*.
2. Merancang sistem dapat menampilkan informasi mengenai Toko Hani Sejahtera seperti cara menghubungi toko, dan email serta juga menampilkan barang barang yang dijual melalui *website* dan mudah diakses oleh pengunjung *website*.
3. Merancang sistem yang dapat menjelaskan secara detail dari barang barang yang ditawarkan dari nama barang, harga barang, kategori barang, detail barang sehingga informasi yang diterima oleh pelanggan lebih cepat dan akurat.

## **4.2 ANALISIS KEBUTUHAN PERANGKAT LUNAK**

### **4.2.1 Kebutuhan Fungsional Sistem**

Pemodelan fungsional sistem menggambarkan proses atau aktivitas layanan yang diberikan oleh sistem berdasarkan prosedur atau fungsi bisnis yang harus dikerjakan oleh sistem untuk melayani kebutuhan pengguna (*user*). Pada penelitian ini berdasarkan kebutuhan, maka fungsi utama yang harus di lakukan oleh sistem pada Toko Hani Sejahtera sebagai berikut :

#### 1. Admin

##### a) Fungsi *Login*

Fungsi admin untuk mengakses sistem

b) Fungsi Mengelola Data Admin

Fungsi ini digunakan oleh admin untuk menambah, mengubah dan menghapus data admin sesuai dengan kebutuhannya

c) Fungsi Mengelola Data Expedisi

Digunakan oleh admin untuk menambah, mengubah dan menghapus data Expedisi sesuai dengan kebutuhannya.

d) Fungsi Mengelola Data Pelanggan

Digunakan oleh admin untuk menambah, mengubah dan menghapus data Pelanggan sesuai dengan kebutuhannya.

e) Fungsi Mengelola Data Kategori

Fungsi ini digunakan oleh admin untuk menambah, mengubah dan menghapus data kategori sesuai dengan kebutuhannya.

f) Fungsi Mengelola Data Barang

Fungsi ini digunakan admin untuk menambah, mengubah dan menghapus data Barang sesuai dengan kebutuhannya

g) Fungsi Mengelola Data Informasi Web

Fungsi ini digunakan admin untuk menambah, mengubah dan menghapus data Informasi Web sesuai dengan kebutuhannya

h) Fungsi Mengelola Data Pemesanan

Digunakan oleh admin untuk mengubah data pemesanan dari sistem.

i) Fungsi Melihat Laporan

Digunakan oleh admin untuk melihat laporan transaksi pemesanan pada sistem.

j) Fungsi *Logout*

Digunakan oleh pelanggan untuk keluar dari sistem

2. Pengunjung

a) Fungsi Melihat Informasi Website

Digunakan oleh pengunjung untuk melihat data informasi yang ada pada *Website*

b) Fungsi Melakukan Pendaftaran

Digunakan oleh pengunjung untuk mendaftarkan diri menjadi pelanggan

3. Pelanggan

a) Fungsi *Login*

Fungsi pelanggan untuk masuk kedalam sistem dan mengakses sistem

b) Fungsi mengubah profil

Fungsi yang dapat digunakan pelanggan untuk merubah profil

c) Fungsi Melakukan Pemesanan

Digunakan oleh pelanggan untuk menambah, mengubah dan menghapus data pemesanan barang

d) Fungsi Mengelola Konfirmasi Pembayaran

Digunakan oleh pelanggan untuk mengkonfirmasi pemesanan barang yang telah dibayar

e) Fungsi *Logout*

Digunakan oleh pelanggan untuk keluar dari sistem

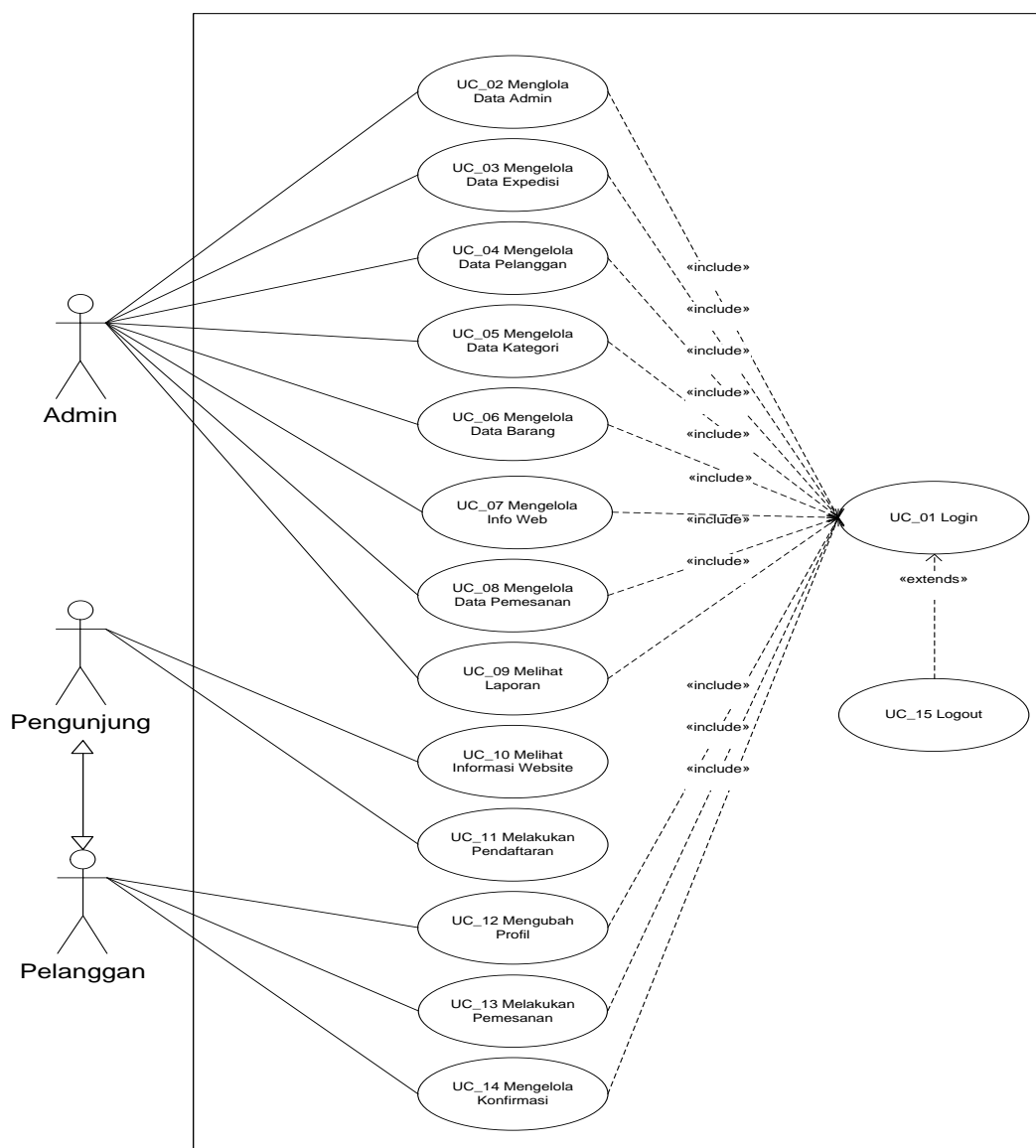
#### 4.2.2 Kebutuhan Non Fungsional Sistem

Kebutuhan non fungsional sistem mendefinisikan properties dan constraints dari sistem. Kebutuhan non fungsional sistem dapat menjadi lebih kritis dari fungsional sistem, dimana jika tidak terpenuhi maka sistem tidak dapat digunakan. Berdasarkan kebutuhan fungsional sistem yang telah dijelaskan sebelumnya, maka diharapkan sistem yang dirancang mampu memiliki hal-hal tersebut berikut:

1. *Usability*
  - a. Mudah digunakan oleh pengunjung, pelanggan dan admin dalam mengakses.
  - b. Informasi yang ditampilkan selalu diupdate oleh admin, sehingga mampu menampilkan informasi yang *uptodate*.
2. *Functionality*
  - a. Mempermudah akses informasi berdasarkan kata kunci (*keyword*)
  - b. Sistem mudah diakses oleh pengguna
  - c. Sistem dapat diakses dalam 24 jam sehari
3. *Security*
  - a. Informasi pelanggan bersifat privasi dan tidak ditampilkan ke *public* (umum)
  - b. Setiap admin dan pelanggan diberi *Username* dan *Password*

### 4.2.3 Use Case Diagram

*Use Case Diagram* merupakan *Diagram* yang menggambarkan metode atau fungsi-fungsi yang dapat dilakukan oleh sistem. Dibawah ini merupakan *Diagram Use Case*. Dari *Use Case* ini terdapat Aktor dan *Use Case* fungsi fungsi sistem. Admin, Pengunjung dan Pelanggan yang merupakan aktor yang berinteraksi dengan sistem secara langsung. Seperti terlihat pada gambar 4.2



**Gambar 4.2 Diagram Use Case Penjualan Toko Hani Sejahtera**



#### 4.2.4 Deskripsi Use Case

Berdasarkan *Use Case Diagram* yang dibahas sebelumnya maka dapat di deskripsikan *Use Case* pada sistem, yang dituangkan dalam bentuk tabel berikut:

##### 1. Deskripsi Use Case Login

Deskripsi *Use Case login* merupakan aktivitas yang dilakukan user dalam melakukan *login* yang berfungsi untuk melakukan pengolahan data pada *Website*, dan dideskripsikan pada *table 4.1* di bawah ini

**Tabel 4.1 Deskripsi Use Case Login**

<b>Nama</b>	<i>Login</i>	
<b>No Id Use Case</b>	UC_01	
<b>Aktor</b>	Admin dan Pelanggan	
<b>Deskripsi</b>	Aktor melakukan <i>login</i> dan masuk kehalaman utama	
<b>Exception</b>	<i>Username</i> dan <i>Password</i> tidak benar	
<b>Pre Condition</b>	<i>Username</i> dan <i>Password</i> sudah harus tersimpan dalam <i>database</i>	
<b>Aktor</b>		<b>Sistem</b>
<b>Skenario Normal</b>		
1. Aktor mengakses halaman <i>login</i>		
		2. Sistem menampilkan halaman <i>login</i> .
3. Aktor melakukan <i>login</i> dengan mengisi <i>Username</i> dan <i>Password</i> , kemudian meng-klik tombol <i>Login</i>		
		4. Sistem membuka koneksi ke <i>database</i>
		5. Sistem melakukan validasi <i>Username</i> dan <i>Password</i> benar
		6. Validasi berhasil, sistem menampilkan halaman utama
<b>Skenario Alternatif</b>		
1. Aktor masuk ke halaman <i>login</i>		
		2. Sistem menampilkan halaman <i>login</i> .
3. Aktor melakukan <i>login</i> dengan mengisi <i>Username</i> dan <i>Password</i> yang salah		
		4. Memberikan peringatan dan mengembalikan ke halaman form <i>login</i>
<b>Post Condition</b>	Aktor berhasil mengakses halaman utama	

## 2. Deskripsi *Use Case* Mengelola Data Admin

Deskripsi *Use Case* untuk mengelola data admin ini akan menjelaskan fungsional dari *Use Case* mengelola data admin terbagi atas menambah data, mengubah data dan menghapus data admin pada *database*. Seperti yang terlihat pada tabel 4.2 dibawah ini:

**Tabel 4.2 Deskripsi *Use Case* Mengelola Data Admin**

<b>Nama</b>	Mengelola Data Admin
<b>No Id Use Case</b>	UC-02
<b>Aktor</b>	Admin
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Admin pada <i>database</i>
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1. Aktor memilih menu admin	
	2. Sistem membuka koneksi ke <i>database</i>
	3. Sistem menampilkan data admin
<b>Sub Skenario</b>	
- Aktor menambah data admin maka skenario yang berlaku adalah S1	
- Aktor mengubah data admin maka skenario yang berlaku adalah S2	
- Aktor menghapus data admin maka skenario yang berlaku adalah S3	
<b>S1- Tambah Data Admin</b>	
1. Aktor mengklik tambah	
	2. Sistem menampilkan form inputan
3. Aktor menambahkan data admin	
4. Aktor mengklik simpan	
	5. Sistem memvalidasi data inputan dan menampilkan pesan.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang

	dinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Admin</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan dan menampilkan pesan.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang dinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Admin</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem mengkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data admin
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	
S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: jika aktor menjawab batal, sistem kembali menampilkan <i>table</i> admin	
<b>Post Condition</b>	Aktor berhasil mengelola data Admin untuk menambah, mengubah dan menghapus data admin

	pada <i>database</i>
--	----------------------

### 3. Deskripsi *Use Case* Mengelola Data Expedisi

Deskripsi *Use Case* untuk mengelola data Expedisi ini akan menjelaskan fungsional dari *Use Case* mengelola data Expedisi terbagi atas mengubah data dan menghapus data Expedisi pada *database*. Seperti yang terlihat pada tabel 4.3 dibawah ini :

**Tabel 4.3 Deskripsi *Use Case* Mengelola Data Expedisi**

<b>Nama</b>	Mengelola Data Expedisi
<b>No Id <i>Use Case</i></b>	UC-03
<b>Aktor</b>	Admin
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Expedisi pada <i>database</i>
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1. Aktor memilih menu Expedisi	
	2. Sistem membuka koneksi ke <i>database</i>
	3. Sistem menampilkan data Expedisi
<b>Sub Skenario</b>	
- Aktor menambah data Expedisi maka skenario yang berlaku adalah S1	
- Aktor mengubah data Expedisi maka skenario yang berlaku adalah S2	
- Aktor menghapus data Expedisi maka skenario yang berlaku adalah S3	
<b>S1- Tambah Data Expedisi</b>	
1. Aktor mengklik tambah	
	2. Sistem menampilkan form inputan
3. Aktor menambahkan data Expedisi	
4. Aktor mengklik simpan	
	5. Sistem memvalidasi data inputan benar.

	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Expedisi</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan benar.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang diinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Expedisi</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem menkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data Expedisi
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	
S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: Jika aktor menjawab batal, sistem kembali menampilkan tabel Expedisi	

<b>Post Condition</b>	Aktor berhasil mengolah data Expedisi untuk menambah, mengubah dan menghapus data Expedisi pada <i>database</i>
-----------------------	---

#### 4. Deskripsi *Use Case* Mengelola Data Pelanggan

Deskripsi *Use Case* untuk mengelola data Pelanggan ini akan menjelaskan fungsional dari *Use Case* mengelola data Pelanggan terbagi atas mengubah data dan menghapus data Pelanggan pada *database*. Seperti yang terlihat pada tabel 4.4 dibawah ini :

**Tabel 4.4 Deskripsi *Use Case* Mengelola Data Pelanggan**

<b>Nama</b>	Mengelola Data Pelanggan	
<b>No Id Use Case</b>	UC-04	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Pelanggan pada <i>database</i>	
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia	
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin	
<b>Aktor</b>		<b>Sistem</b>
<b>Skenario Normal</b>		
1. Aktor memilih menu Pelanggan		
		2. Sistem membuka koneksi ke <i>database</i>
		3. Sistem menampilkan data Pelanggan
<b>Sub Skenario</b>		
- Aktor menambah data Pelanggan maka skenario yang berlaku adalah S1		
- Aktor mengubah data Pelanggan maka skenario yang berlaku adalah S2		
- Aktor menghapus data Pelanggan maka skenario yang berlaku adalah S3		
<b>S1- Tambah Data Pelanggan</b>		
1. Aktor mengklik tambah		
		2. Sistem menampilkan form inputan
3. Aktor menambahkan data Pelanggan		
4. Aktor mengklik simpan		
		5. Sistem memvalidasi data

	inputan benar.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Pelanggan</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan benar.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang diinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Pelanggan</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem menkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data Pelanggan
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	

S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: Jika aktor menjawab batal, sistem kembali menampilkan tabel Pelanggan	
<b>Post Condition</b>	Aktor berhasil mengolah data Pelanggan untuk menambah, mengubah dan menghapus data Pelanggan pada <i>database</i>

#### 5. Deskripsi *Use Case* Mengelola Kategori

Deskripsi *Use Case* untuk mengelola data kategori ini akan menjelaskan fungsional dari *Use Case* mengelola data kategori terbagi atas menambah datan, mengubah data dan menghapus data kategori pada *database*. Seperti yang terlihat pada tabel 4.5 dibawah ini :

**Tabel 4.5 Deskripsi *Use Case* Mengelola Data Kategori**

<b>Nama</b>	Mengelola Data Kategori	
<b>No Id Use Case</b>	UC-05	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Kategori pada <i>database</i>	
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia	
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin	
<b>Aktor</b>	<b>Sistem</b>	
<b>Skenario Normal</b>		
1. Aktor memilih menu kategori		
	2. Sistem membuka koneksi ke <i>database</i>	
	3. Sistem menampilkan data kategori	
<b>Sub Skenario</b>		
- Aktor menambah data kategori maka skenario yang berlaku adalah S1		
- Aktor mengubah data kategori maka skenario yang berlaku adalah S2		
- Aktor menghapus data kategori maka skenario yang berlaku adalah S3		
<b>S1- Tambah Data Kategori</b>		
1. Aktor mengklik tambah		
	2. Sistem menampilkan form inputan	



3. Aktor menambahkan data kategori	
4. Aktor mengklik simpan	
	5. Sistem memvalidasi data inputan benar.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Kategori</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan benar.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang diinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Kategori</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem menkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data Barang
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	

S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	
S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: Jika aktor menjawab batal, sistem kembali menampilkan tabel kategori	
<b>Post Condition</b>	Aktor berhasil mengolah data Kategori untuk menambah, mengubah dan menghapus data Kategori pada <i>database</i>

#### 6. Deskripsi *Use Case* Mengelola Data Barang

Deskripsi *Use Case* untuk mengolah data Barang ini akan menjelaskan fungsional dari *Use Case* mengolah data Barang yang dilakukan oleh admin.

Seperti yang terlihat pada tabel 4.6 berikut :

**Tabel 4.6 Deskripsi *Use Case* Mengelola Barang**

<b>Nama</b>	Mengelola Data Barang
<b>No Id Use Case</b>	UC-06
<b>Aktor</b>	Admin
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Barang pada <i>database</i>
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1. Aktor memilih menu Barang	
	2. Sistem membuka koneksi ke <i>database</i>
	3. Sistem menampilkan data Barang
<b>Sub Skenario</b>	
- Aktor menambah data Barang maka skenario yang berlaku adalah S1	
- Aktor mengubah data Barang maka skenario yang berlaku adalah S2	
- Aktor menghapus data Barang maka skenario yang berlaku adalah S3	
<b>S1- Tambah Data Barang</b>	
1. Aktor mengklik tambah	
	2. Sistem menampilkan form inputan
3. Aktor menambahkan data Barang	

4. Aktor mengklik simpan	
	5. Sistem memvalidasi data inputan benar.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Barang</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan benar.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang diinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Barang</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem mengkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data Barang
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap	

dan proses penyimpanan data dibatalkan	
S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: Jika aktor menjawab batal, sistem kembali menampilkan tabel Barang	
<b>Post Condition</b>	Aktor berhasil mengolah data Barang untuk menambah, mengubah dan menghapus data Barang pada <i>database</i>

#### 7. Deskripsi *Use Case* Mengelola Data Info web

Deskripsi *Use Case* untuk mengelola data Info web ini akan menjelaskan fungsional dari *Use Case* mengelola data Info web terbagi atas mengubah data dan menghapus data Info web pada *database*. Seperti yang terlihat pada tabel 4.7 dibawah ini :

**Tabel 4.7 Deskripsi *Use Case* Mengelola Data Info web**

<b>Nama</b>	Mengelola Data Info web	
<b>No Id <i>Use Case</i></b>	UC-07	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Aktor menambah, mengubah, dan menghapus data Info web pada <i>database</i>	
<b>Exception</b>	Data yang diinputkan tidak lengkap, data yang dimodifikasi tidak lengkap dan data yang akan dihapus tidak tersedia	
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin	
<b>Aktor</b>		<b>Sistem</b>
<b>Skenario Normal</b>		
1. Aktor memilih menu Info web		
		2. Sistem membuka koneksi ke <i>database</i>
		3. Sistem menampilkan data Info web
<b>Sub Skenario</b>		
- Aktor menambah data Info web maka skenario yang berlaku adalah S1		
- Aktor mengubah data Info web maka skenario yang berlaku adalah S2		
- Aktor menghapus data Info web maka skenario yang berlaku adalah S3		
<b>S1- Tambah Data Info web</b>		
1. Aktor mengklik tambah		
		2. Sistem menampilkan form inputan

3. Aktor menambahkan data Info web	
4. Aktor mengklik simpan	
	5. Sistem memvalidasi data inputan benar.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan “Berhasil simpan data”
<b>S2- Ubah Data Info web</b>	
1. Aktor memilih data yang akan di ubah	
2. Aktor memilih tombol ubah	
	3. Menampilkan data yang akan di ubah
4. Aktor memodifikasi data	
5. Aktor memilih tombol simpan	
	6. Sistem memvalidasi data inputan benar.
	7. Sistem membuka koneksi <i>database</i>
	8. Sistem menyimpan data yang diinputkan.
	9. Sistem menampilkan pesan “Berhasil ubah data”
<b>S3- Hapus Data Info web</b>	
1. Aktor mengklik tombol Hapus.	
	2. Sistem menkonfirmasi hapus data
	3. Sistem menampilkan pesan konfirmasi untuk menghapus data.
4. Aktor mengklik pilihan ya.	
	5. Sistem membuka koneksi <i>database</i>
	6. Sistem menghapus data Info web
	7. Sistem menampilkan pesan “Berhasil hapus data”
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	

S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
S-2 6a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	
S-2 6b: Sistem memberikan kesempatan untuk mengulang kembali.	
S-3 3a: Jika aktor menjawab batal, sistem kembali menampilkan tabel Info web	
<b>Post Condition</b>	Aktor berhasil mengolah data Info web untuk menambah, mengubah dan menghapus data Info web pada <i>database</i>

#### 8. Deskripsi *Use Case* Mengelola Data Pemesanan

Deskripsi *Use Case* untuk mengelola data pemesanan ini akan menjelaskan fungsional dari *Use Case* mengelola data pemesanan yang dilakukan oleh admin. Seperti yang terlihat pada tabel 4.8 berikut :

**Tabel 4.8 Deskripsi *Use Case* Mengelola Pemesanan**

<b>Nama</b>	Mengelola Data Pemesanan	
<b>No Id <i>Use Case</i></b>	UC-08	
<b>Aktor</b>	Admin	
<b>Deskripsi</b>	Aktor melihat dan mengubah data pemesanan pada <i>database</i>	
<b>Exception</b>	Data yang dimodifikasi tidak lengkap	
<b>Pre Condition</b>	Aktor telah melakukan <i>login</i> sebagai admin	
<b>Aktor</b>		<b>Sistem</b>
<b>Skenario Normal</b>		
1. Aktor memilih menu pemesanan		2. Sistem menampilkan data pemesanan
3. Aktor memilih data yang akan diubah		
4. Aktor mengklik link ubah pesanan		
		5. Sistem menampilkan data pemesanan beserta form inputan
6. Aktor mengubah status pemesanan, input no resi		
7. Klik tombol simpan		
		8. Sistem memvalidasi data inputan benar.
		9. Sistem membuka koneksi <i>database</i>

	10. Sistem menyimpan data yang diinputkan.
	11. Sistem menampilkan pesan “Berhasil mengubah data”
<b>Skenario Alternatif</b>	
8a: Sistem akan menampilkan pesan apabila ada data yang tidak lengkap dan proses penyimpanan data dibatalkan	
8b: Sistem memberikan kesempatan untuk mengulang kembali.	
<b>Post Condition</b>	Aktor berhasil mengelola data Pemesanan pada <i>database</i>

#### 9. Deskripsi *Use Case* Melihat Laporan

Deskripsi *Use Case* untuk melihat laporan ini akan menjelaskan fungsional dari *Use Case* membuat laporan transaksi yang dilakukan oleh admin. Seperti yang terlihat pada tabel 4.9 berikut :

**Tabel 4.9 Deskripsi *Use Case* Melihat Laporan**

<b>Nama</b>	Melihat Laporan
<b>No Id Use Case</b>	UC-09
<b>Aktor</b>	Admin
<b>Deskripsi</b>	Aktor melihat laporan transaksi
<b>Exception</b>	-
<b>Pre Condition</b>	Aktor telah <i>login</i> ke dalam sistem sebagai admin
<b>Aktor</b>	<b>Sistem</b>
Skenario Normal	
1. Aktor mengklik menu laporan.	
	2. Sistem menampilkan halaman laporan transaksi pemesanan
3. Aktor memasukan periode transaksi	
4. Aktor mengklik tombol cari.	
	5. Sistem membuka koneksi ke database
	6. Sistem melakukan pencarian data pemesanan berdasarkan periode
	7. Sistem menampilkan data pemesanan
Skenario Alternatif	
-	
<b>Post Condition</b>	Aktor melihat laporan.

#### 10. Deskripsi *Use Case* Melihat Informasi *Website*

Deskripsi *Use Case* melihat Informasi *Website* merupakan aktivitas yang dilakukan pengunjung dan pelanggan dalam mengakses Informasi *Website* dari *Website* utama. Seperti yang terlihat pada tabel 4.10 berikut :

**Tabel 4.10 Deskripsi *Use Case* Melihat Informasi *Website***

<b>Nama</b>	Melihat Informasi <i>Website</i>
<b>No Id <i>Use Case</i></b>	UC-10
<b>Aktor</b>	Pengunjung dan Pelanggan
<b>Deskripsi</b>	Aktor melihat Informasi <i>Website</i> yang terdapat pada <i>Website</i>
<b>Exception</b>	-
<b>Pre Condition</b>	Aktor telah membuka web pada browser
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1 Aktor memilih menu pada <i>website</i>	
	2 Sistem menampilkan Informasi yang dipilih
<b>Skenario Alternatif</b>	
-	
<b>Post Condition</b>	Aktor berhasil melihat Informasi <i>Website</i>

#### 11. Deskripsi *Use Case* Melakukan Pendaftaran

Deskripsi *Use Case* melakukan Pendaftaran merupakan aktivitas yang dilakukan pengunjung untuk melakukan Pendaftaran. Seperti yang terlihat pada tabel 4.11 berikut :

**Tabel 4.11 Deskripsi *Use Case* Melakukan Pendaftaran**

<b>Nama</b>	Melakukan Pendaftaran
<b>No Id <i>Use Case</i></b>	UC-11
<b>Aktor</b>	Pengunjung
<b>Deskripsi</b>	Dilakukan oleh aktor untuk melakukan Pendaftaran
<b>Exception</b>	Data yang diinputkan tidak lengkap
<b>Pre Condition</b>	Aktor telah membuka web pada browser



Aktor	Sistem
<b>Skenario Normal</b>	
1. Aktor mengklik menu pendaftaran	
	2. Sistem menampilkan form untuk menambah data Pendaftaran
3. Aktor mengisi form Pendaftaran	
4. Aktor menekan tombol daftar	
	5. Sistem memvalidasi data inputan dan valid.
	6. Sistem membuka koneksi <i>database</i>
	7. Sistem menyimpan data yang diinputkan.
	8. Sistem menampilkan pesan "Pendaftaran berhasil silahkan melakukan <i>login</i> "
<b>Skenario Alternatif</b>	
S-1 5a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
S-1 5b: Sistem memberikan kesempatan untuk mengulang kembali	
<b>Post Condition</b>	Aktor telah melakukan Pendaftaran

## 12. Deskripsi *Use Case* Mengubah Profil

Deskripsi *Use Case* Mengubah Profil merupakan aktivitas yang dilakukan pelanggan untuk Mengubah Profil. Seperti yang terlihat pada tabel 4.12 berikut :

**Tabel 4.12 Deskripsi *Use Case* Mengubah Profil**

<b>Nama</b>	Mengubah Profil
<b>No Id <i>Use Case</i></b>	UC-12
<b>Aktor</b>	Pelanggan
<b>Deskripsi</b>	Proses yang dilakukan oleh aktor untuk Mengubah Profil
<b>Exception</b>	-
<b>Pre Condition</b>	Aktor telah <i>login</i> kedalam system
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	

1. Aktor memilih menu profil	
	2. Sistem menampilkan data profil
3. Aktor mengklik tombol Ubah Profil	
	4. Sistem menampilkan form yang siap di ubah
5. Aktor mengisi data profil baru	
6. Aktor menekan tombol Simpan	
	7. Sistem menyimpan data profil
<b>Skenario Alternatif</b>	
7a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
7b: Sistem memberikan kesempatan untuk mengulang kembali	
<b>Post Condition</b>	Aktor telah Mengubah Profil

### 13. Deskripsi *Use Case* Melakukan Pemesanan

Deskripsi *Use Case* melakukan pemesanan dapat dilihat pada tabel 4.13

berikut :

**Tabel 4.13 Deskripsi *Use Case* Melakukan Pemesanan**

<b>Nama</b>	Melakukan Pemesanan
<b>No Id <i>Use Case</i></b>	UC-12
<b>Aktor</b>	Pelanggan
<b>Deskripsi</b>	Proses yang dilakukan oleh aktor untuk melakukan pemesanan barang
<b>Exception</b>	Data yang diinputkan tidak lengkap
<b>Pre Condition</b>	Aktor telah <i>login</i> kedalam system
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1. Aktor menambahkan Barang ke keranjang	
	2. Sistem menyimpan Barang ke keranjang
3. Aktor membuka menu keranjang	
	4. Sistem menampilkan daftar Barang yang telah dipilih sebelumnya.
5. Aktor menekan tombol pesan	
	6. Sistem menampilkan ringkasan

	Barang beserta halaman pengirim
7. Aktor memastikan pesanan lalu mengisi data – data pengiriman dan menekan tombol Selesai	
	8. Sistem memvalidasi data dan menyimpan kedalam <i>database</i>
<b>Skenario Alternatif</b>	
8a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
8b: Sistem memberikan kesempatan untuk mengulang kembali	
<b>Post Condition</b>	Aktor telah melakukan pemesanan

#### 14. Deskripsi *Use Case* Mengelola Konfirmasi Pembayaran

Deskripsi *Use Case* Mengelola Konfirmasi Pembayaran dapat dilihat pada pada tabel 4.14 dibawah ini :

**Tabel 4.14 Deskripsi *Use Case* Mengelola Konfirmasi Pembayaran**

<b>Nama</b>	Mengelola Konfirmasi Pembayaran
<b>No Id <i>Use Case</i></b>	UC-13
<b>Aktor</b>	Pelanggan
<b>Deskripsi</b>	Proses yang dilakukan oleh aktor untuk melakukan Mengelola Konfirmasi Pembayaran
<b>Exception</b>	Konfirmasi Gagal
<b>Pre Condition</b>	Aktor telah <i>login</i> kedalam system
<b>Aktor</b>	<b>Sistem</b>
<b>Skenario Normal</b>	
1. Aktor memilih menu <i>History</i> Pemesanan	
	2. Sistem menampilkan daftar pemesanan
3. Aktor memilih pesanan yang dibuat sebelumnya	
	4. Sistem menampilkan form konfirmasi
5. Aktor mengisi data konfirmasi dan menekan tombol Simpan	
	6. Sistem melakukan validasi data
	7. Sistem menyimpan data konfirmasi ke <i>database</i>

<b>Skenario Alternatif</b>	
6a: Sistem akan menampilkan pesan apabila ada data yang tidak terisi dan proses penyimpanan data dibatalkan	
6b: Sistem memberikan kesempatan untuk mengulang kembali	
<b>Post Condition</b>	Aktor telah melakukan konfirmasi

#### 15. Deskripsi *Use Case Logout*

Rincian fungsionalitas dari *Use Case Logout* terlihat pada tabel 4.15 berikut :

**Tabel 4.15 Deskripsi *Use Case Logout***

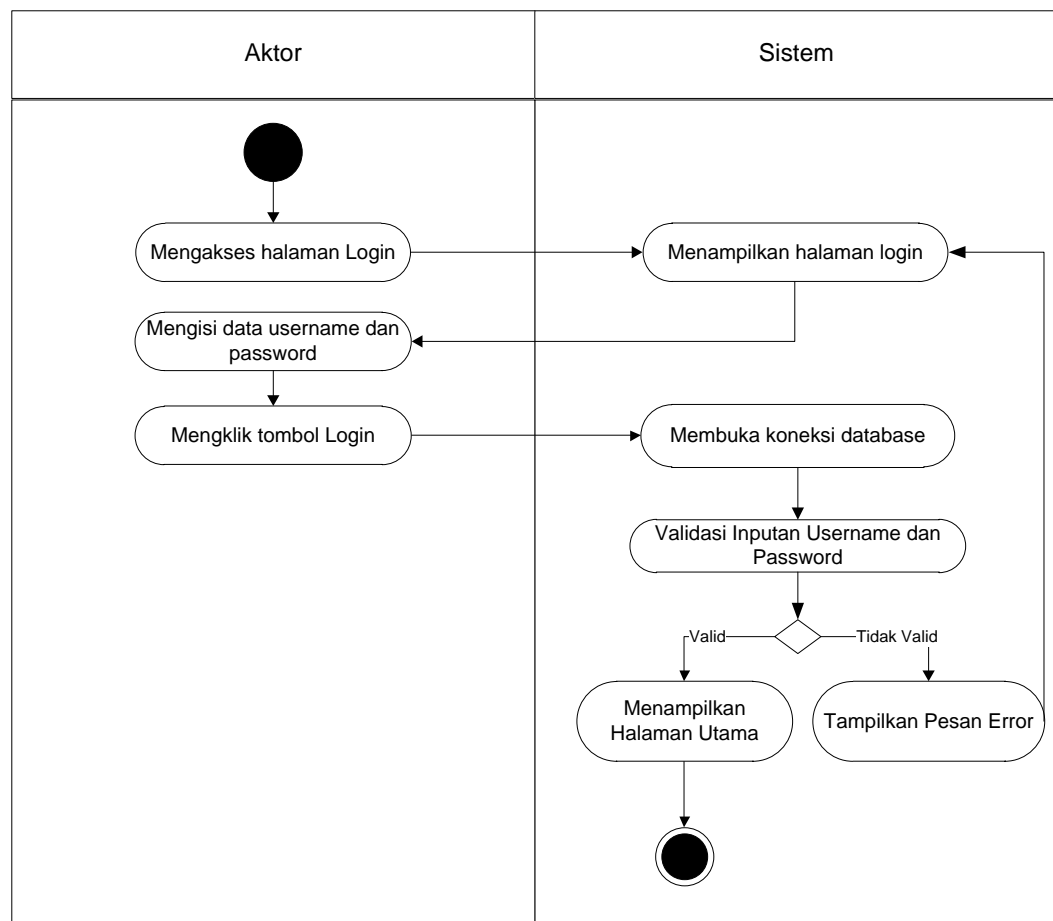
<b>Nama</b>	<i>Logout</i>	
<b>No Id Use Case</b>	UC-15	
<b>Aktor</b>	Pelanggan dan Admin	
<b>Deskripsi</b>	Proses yang dilakukan oleh aktor untuk melihat <i>logout</i>	
<b>Exception</b>	-	
<b>Pre Condition</b>	Aktor telah <i>login</i> kedalam system	
<b>Aktor</b>		<b>Sistem</b>
<b>Skenario Normal</b>		
1. Aktor memilih menu <i>Logout</i>		
		2. Sistem menghapus seluruh <i>session</i>
<b>Post Condition</b>	Aktor telah keluar dari system	

#### 4.2.5 *Activity Diagram*

*Activity Diagram* merupakan gambar proses bisnis dan urutan aktivitas dalam sebuah proses. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antara proses digambarkan dengan garis penghubung. Untuk memahami proses bisnis sistem yang di rancang ini, maka berikut ini beberapa *Activity Diagram* yang digunakan pada perancangan aplikasi yang lebih mendalam.

### 1. *Activity Diagram Login*

*Activity Diagram Login* menggambarkan aktifitas Aktor untuk melakukan *login*, dengan cara memasukan *user name* dan *name* dan *password* setelah memasukkan *username* dan *password*, sistem membuka koneksi ke database sistem memvalidasi *username* dan *password*, jika *username* dan *password* valid maka sistem menampilkan menu utama, jika tidak valid maka tampilan pesan peringatan dan aktor memasukkan kembali *username* dan *password*. *Activity Diagram Login* ditunjukkan pada Gambar 4.3.



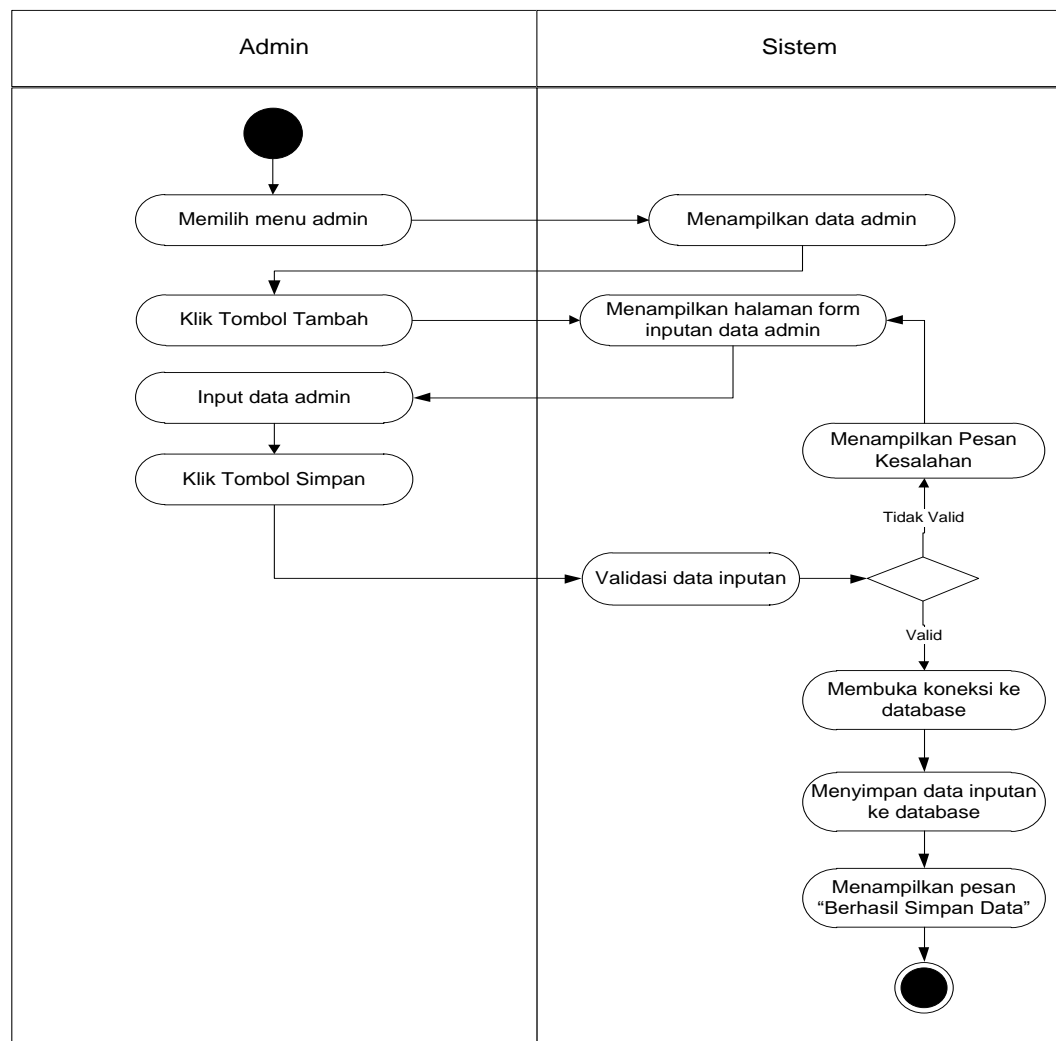
**Gambar 4.3** *Activity Diagram Login*

## 2. Activity Diagram Mengelola Data Admin

Activity Diagram mengelola data Admin yang ditunjukkan pada Gambar 4.4 sampai dengan Gambar 4.6 di bawah ini menjelaskan aktivitas sistem saat menambah, mengubah dan menghapus data Admin.

### a. Activity Diagram Menambah Data Admin.

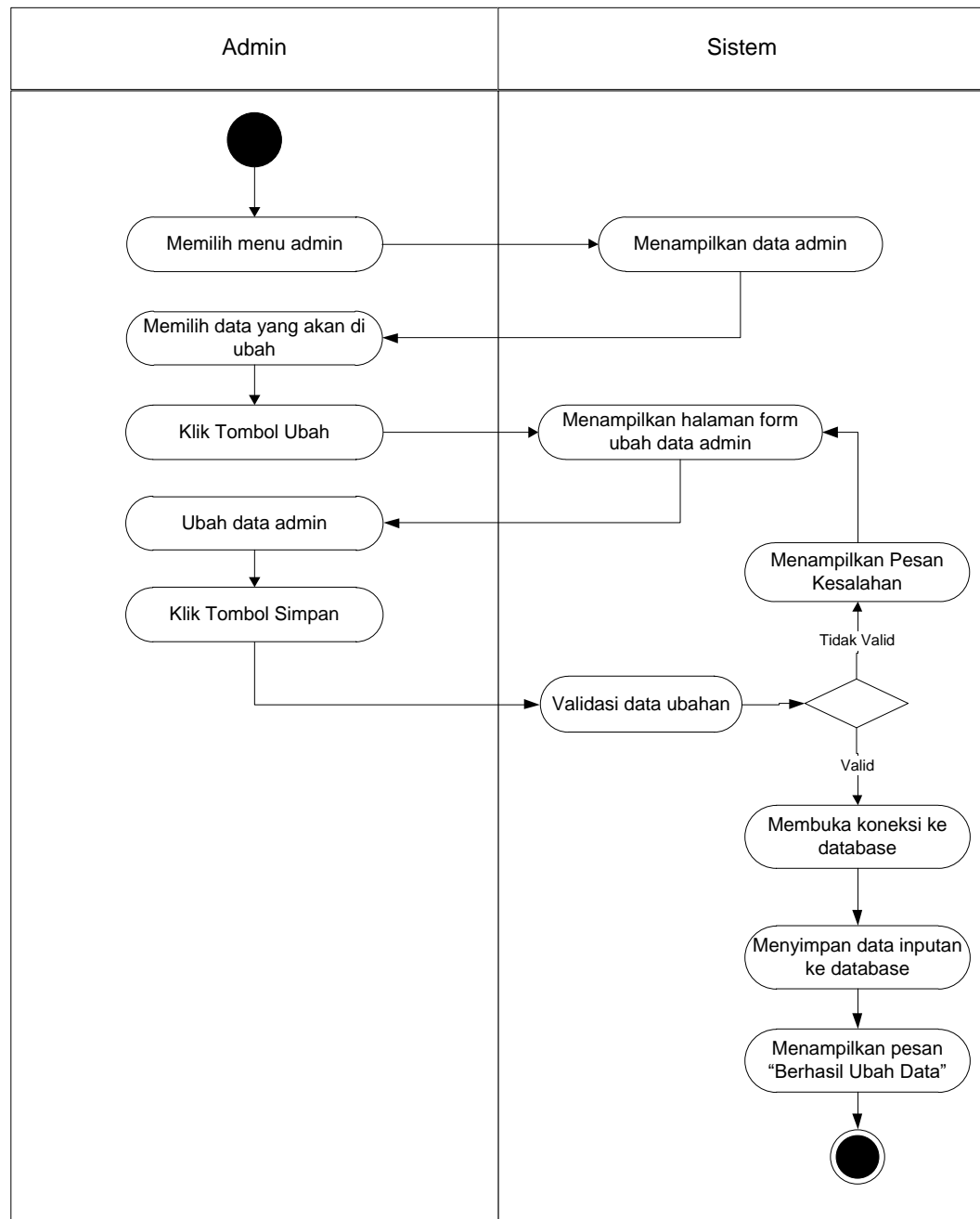
Activity Diagram menambah data admin menggambarkan aktifitas Admin dalam proses menambah data admin baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.4



**Gambar 4.4 Activity Diagram Menambah Data Admin**

b. *Activity Diagram* Mengubah Admin

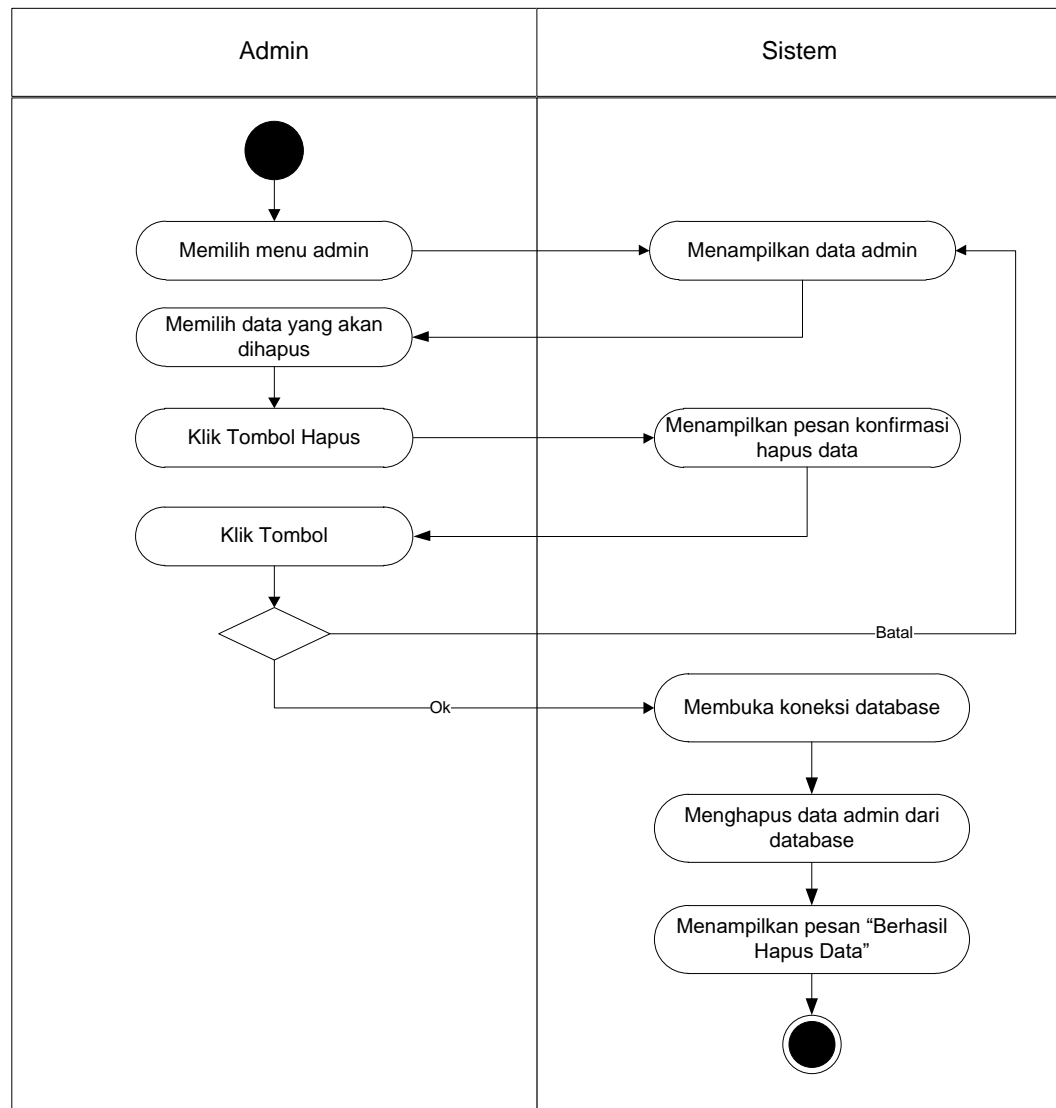
*Activity Diagram* mengubah admin merupakan langkah – langkah dari Admin untuk mengubah admin dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.5.



**Gambar 4.5** *Activity Diagram* Mengubah Data Admin

c. *Activity Diagram* Menghapus Data Admin

*Activity Diagram* menghapus admin merupakan langkah – langkah dari Admin untuk menghapus data admin dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.6



**Gambar 4.6** *Activity Diagram* Menghapus Data Admin

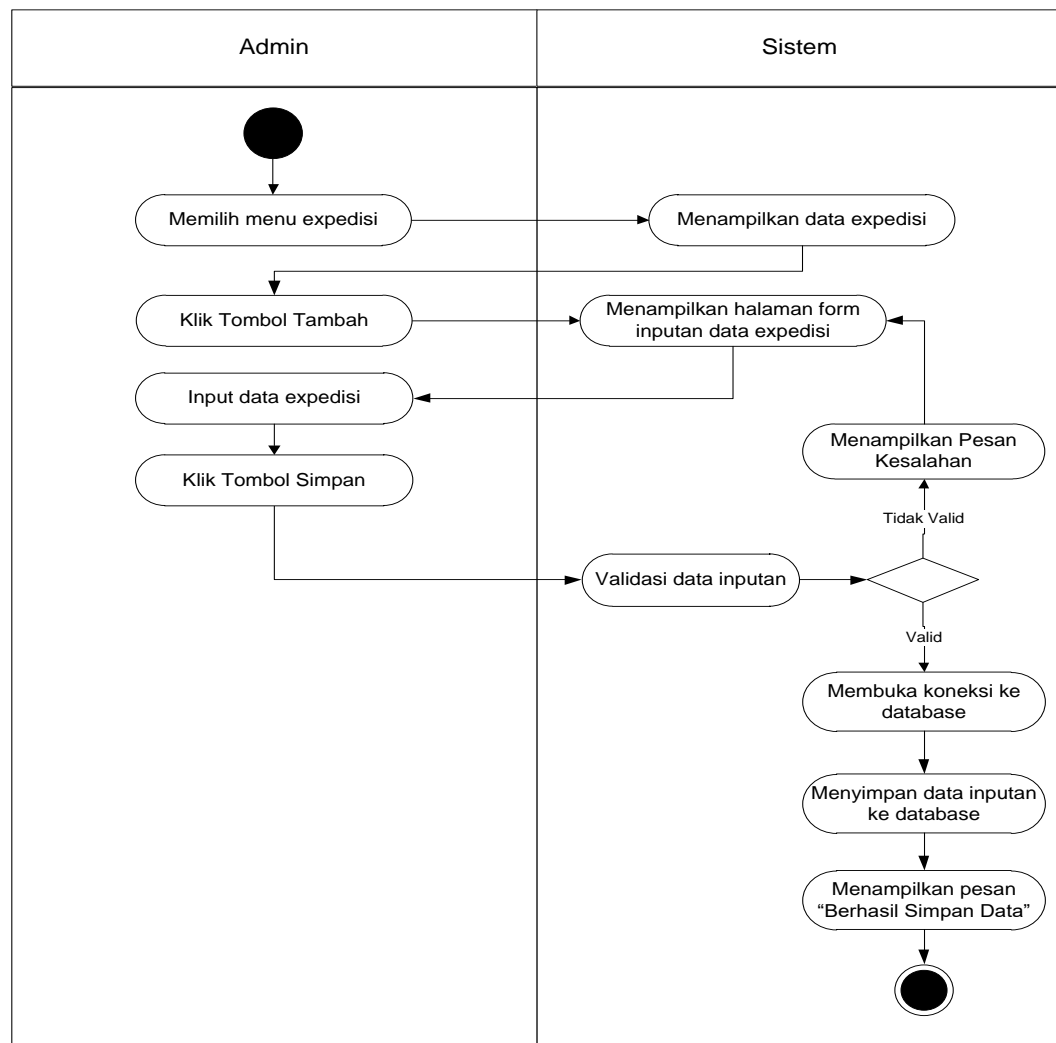


### 3. *Activity Diagram* Mengelola Data Expedisi

*Activity Diagram* mengeloladata Expedisi yang ditunjukkan pada Gambar 4.7 sampai dengan Gambar 4.9 di bawah ini menjelaskan aktivitas sistem saat menambah dan menghapus data Expedisi.

#### a. *Activity Diagram* Menambah Data Expedisi

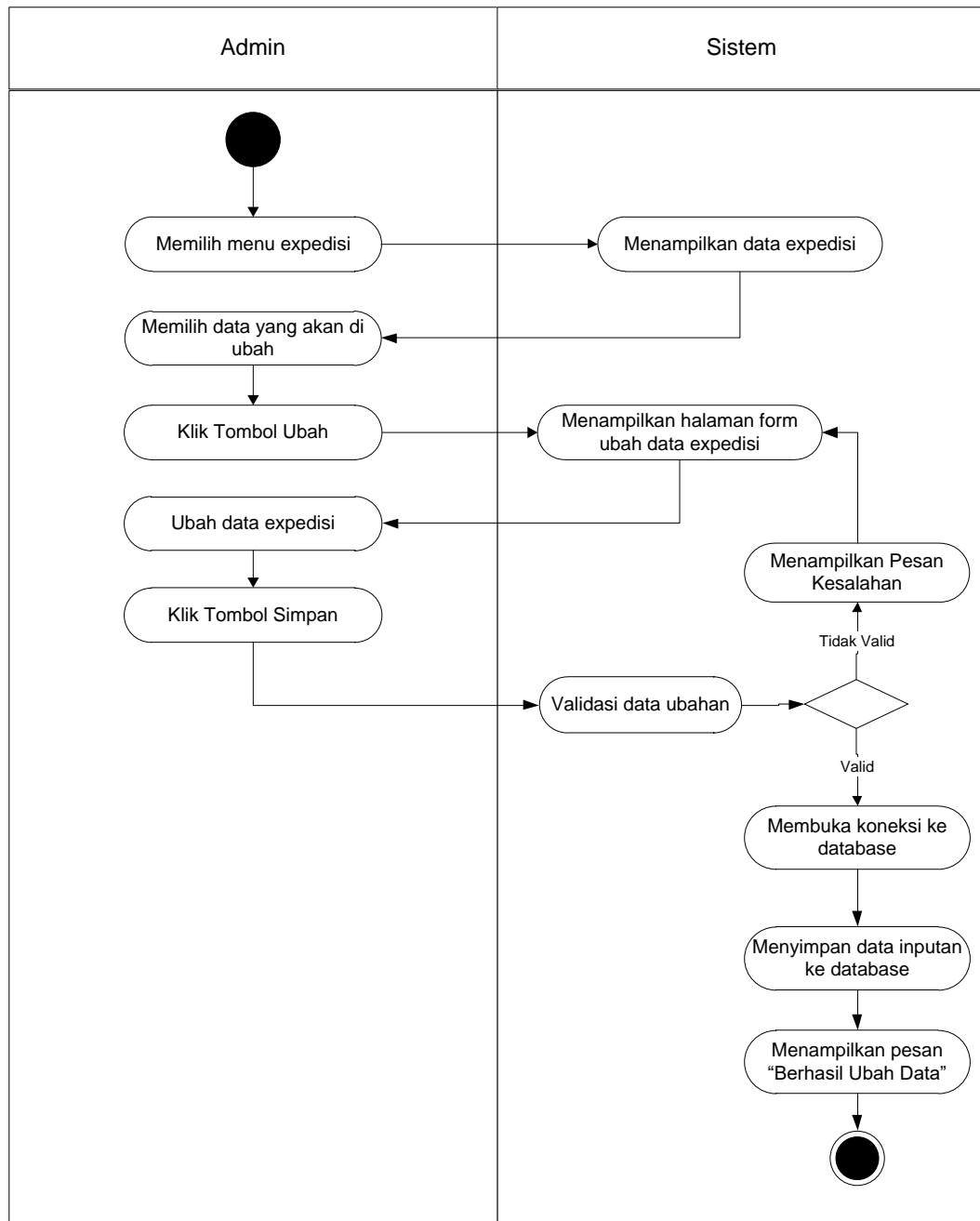
*Activity Diagram* menambah data Expedisi menggambarkan aktifitas Admin dalam proses menambah data Expedisi baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.7



**Gambar 4.7 *Activity Diagram* Menambah Data Expedisi**

b. *Activity Diagram* Mengubah Data Expedisi

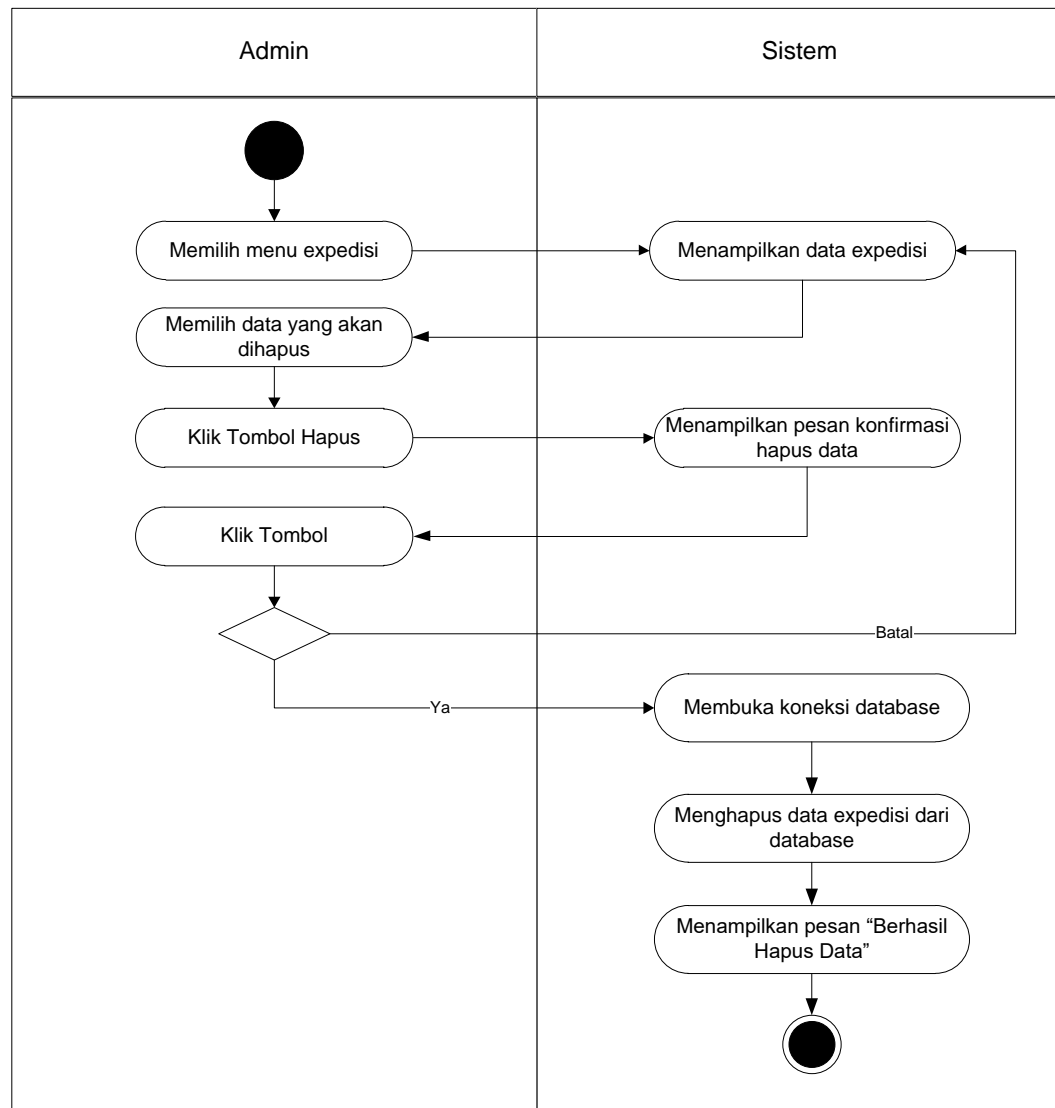
*Activity Diagram* mengubah Expedisi merupakan langkah – langkah dari Admin untuk mengubah Expedisi dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.8.



**Gambar 4.8** *Activity Diagram* Mengubah Data Expedisi

c. *Activity Diagram* Menghapus Data Expedisi

*Activity Diagram* menghapus Expedisi merupakan langkah – langkah dari Admin untuk menghapus data Expedisi dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.9



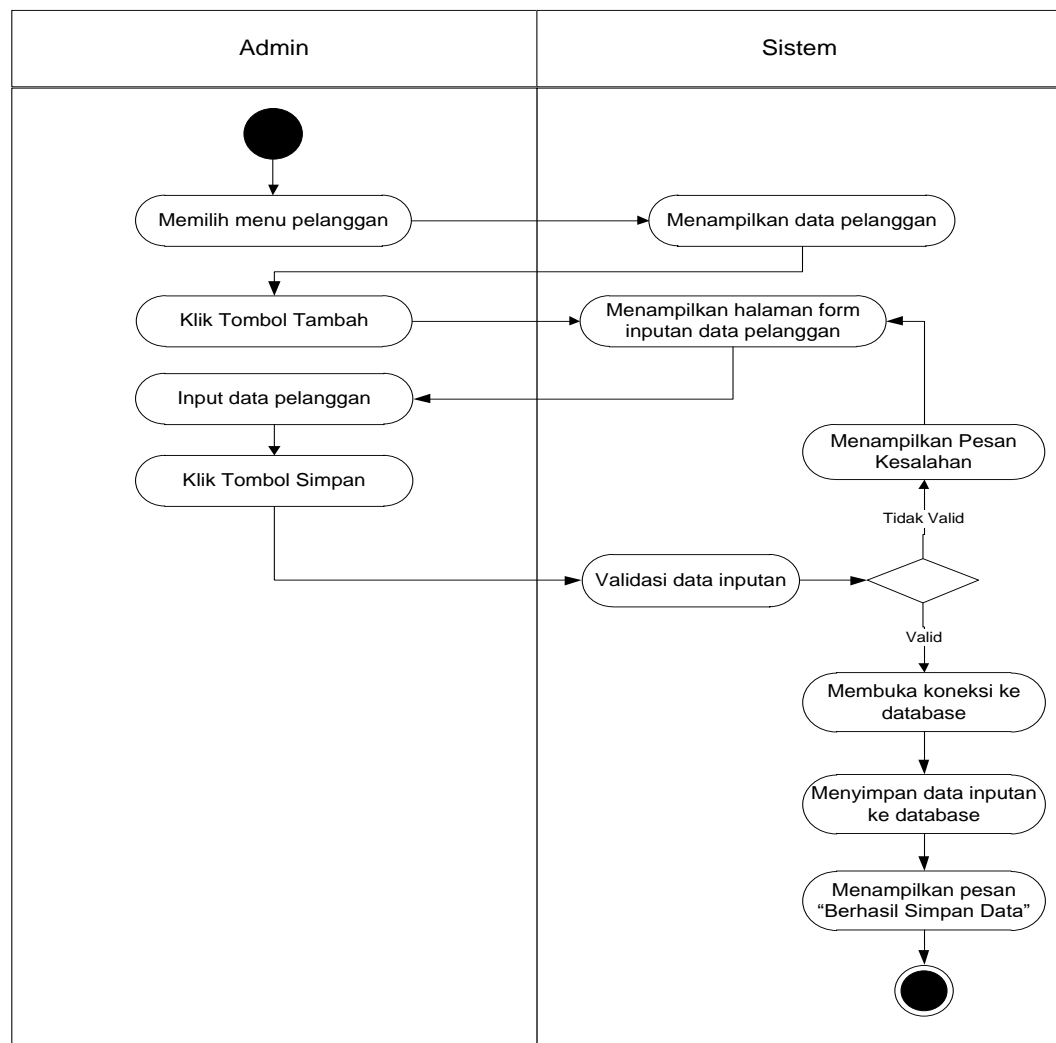
**Gambar 4.9** *Activity Diagram* Menghapus Data Expedisi

#### 4. *Activity Diagram* Mengelola data Pelanggan

*Activity Diagram* mengelola data Pelanggan yang ditunjukkan pada Gambar 4.10 sampai dengan Gambar 4.12 di bawah ini menjelaskan aktivitas sistem saat menambah dan menghapus data Pelanggan.

##### a. *Activity Diagram* Menambah Data Pelanggan

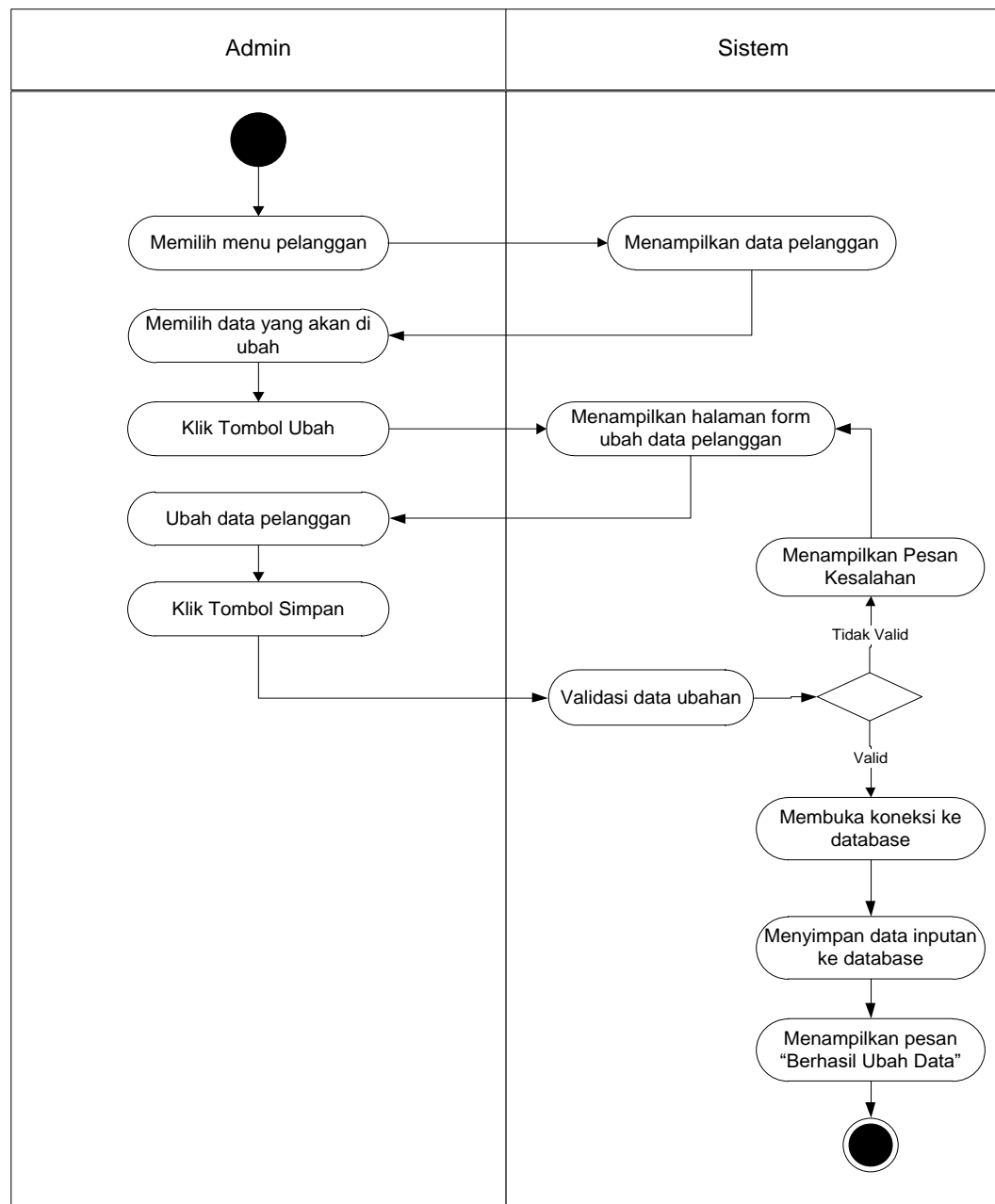
*Activity Diagram* menambah data Pelanggan menggambarkan aktifitas Admin dalam proses menambah data Pelanggan baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.10



**Gambar 4.10 *Activity Diagram* Menambah Data Pelanggan**

b. *Activity Diagram* Mengubah Data Pelanggan

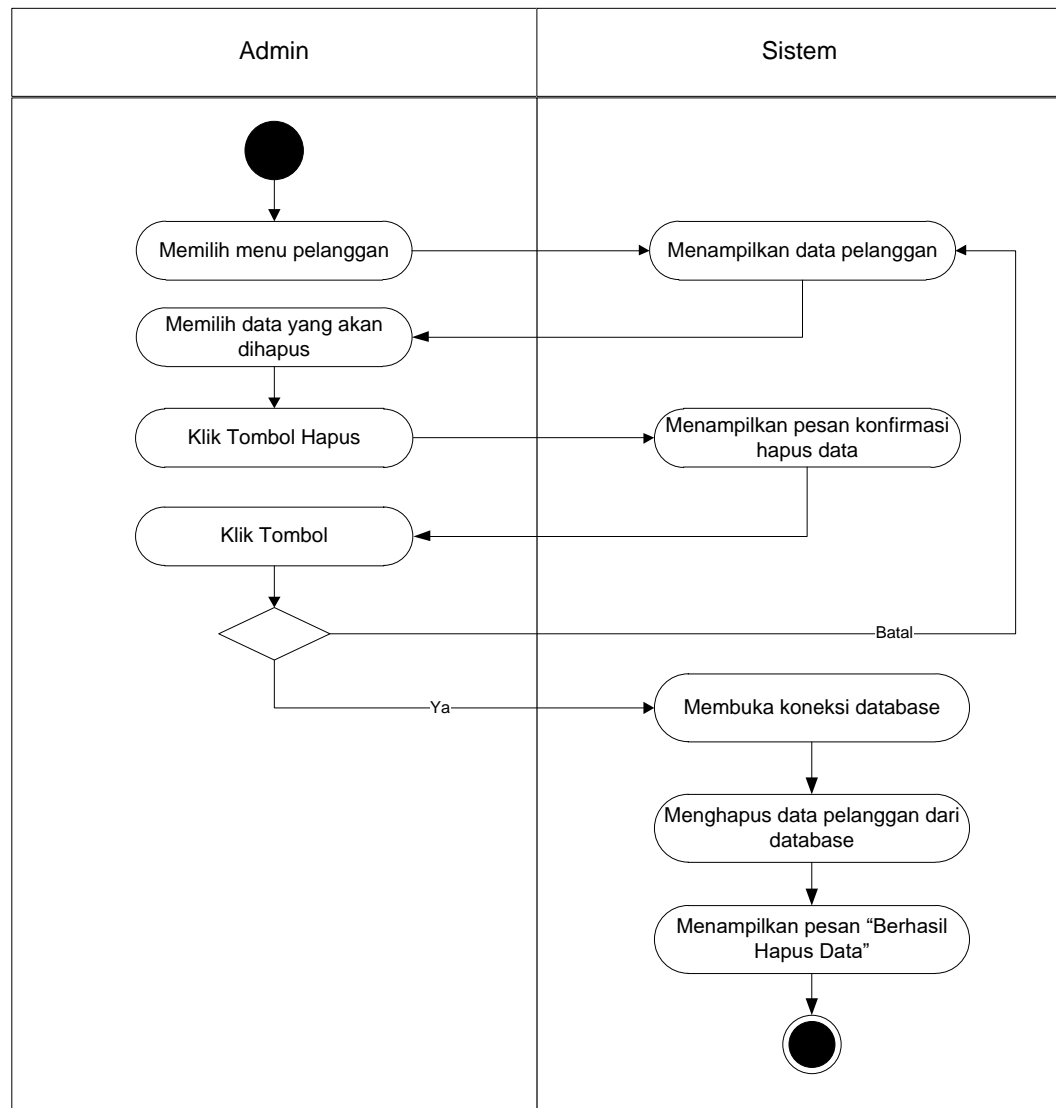
*Activity Diagram* mengubah Pelanggan merupakan langkah – langkah dari Admin untuk mengubah Pelanggan dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.11



**Gambar 4.11** *Activity Diagram* Mengubah Data Pelanggan

c. *Activity Diagram* Menghapus Data Pelanggan

*Activity Diagram* menghapus Pelanggan merupakan langkah – langkah dari Admin untuk menghapus data Pelanggan dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.12



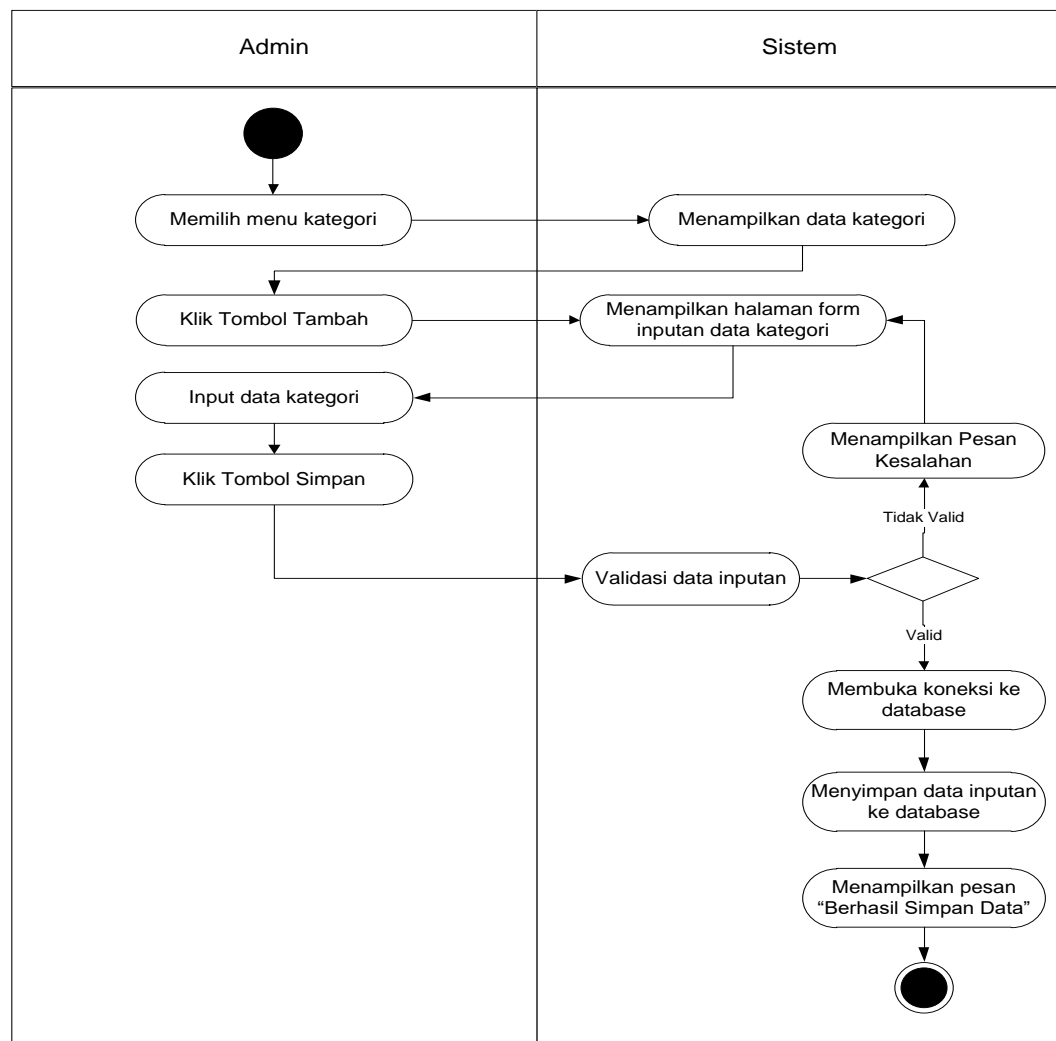
**Gambar 4.12** *Activity Diagram* Menghapus Data Pelanggan

## 5. Activity Diagram Mengelola Data Kategori

*Activity Diagram* mengelola data kategori yang ditunjukkan pada Gambar 4.13 sampai dengan Gambar 4.15 di bawah ini menjelaskan aktivitas sistem saat menambah, mengubah dan menghapus data kategori.

### a. Activity Diagram Menambah Data kategori

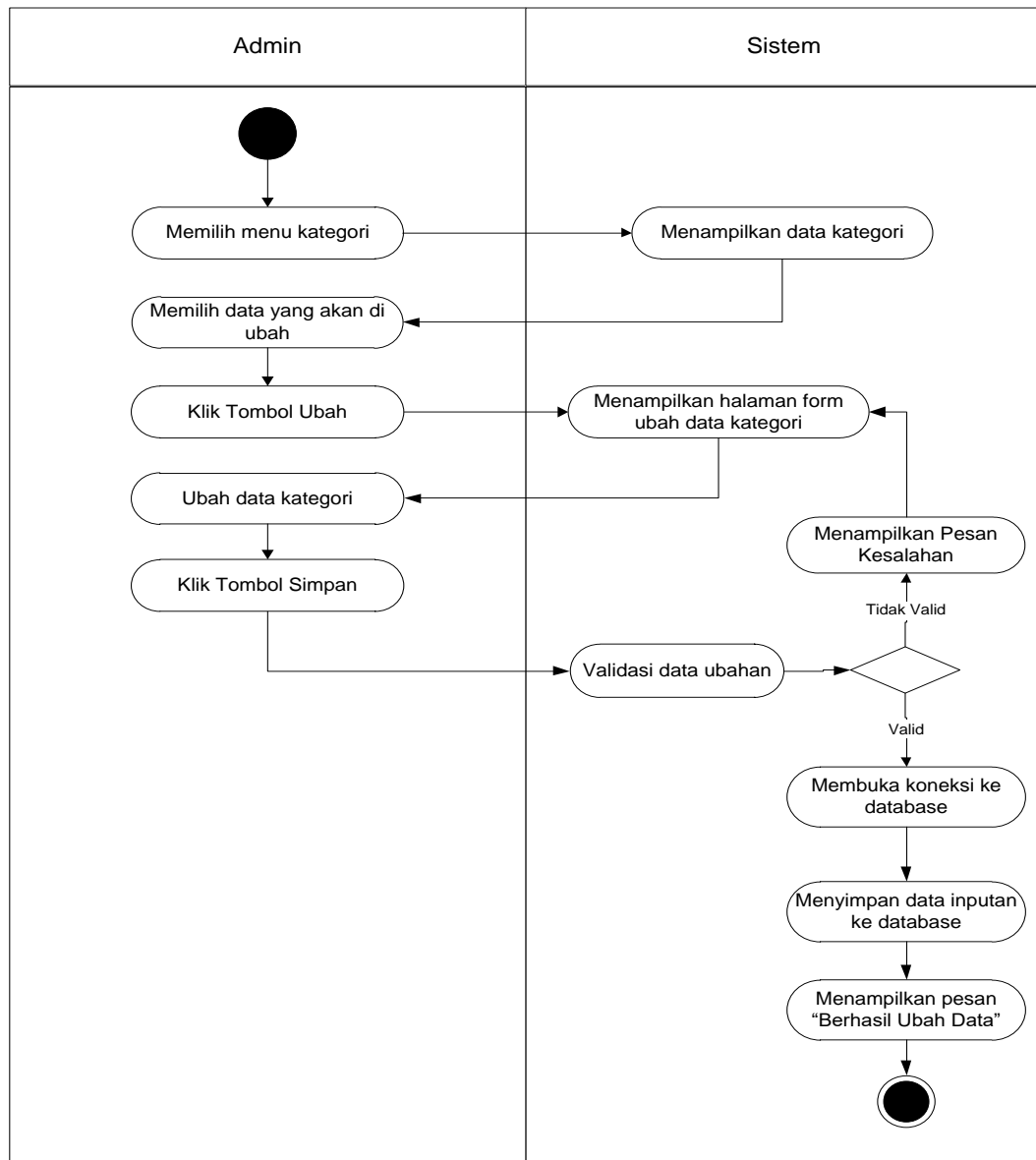
*Activity Diagram* menambah data kategori menggambarkan aktifitas Admin dalam proses menambah data kategori baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.13



**Gambar 4.13 Activity Diagram Menambah Data Kategori**

b. *Activity Diagram* Mengubah Data Kategori

*Activity Diagram* mengubah kategori merupakan langkah – langkah dari Admin untuk mengubah kategori dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.14

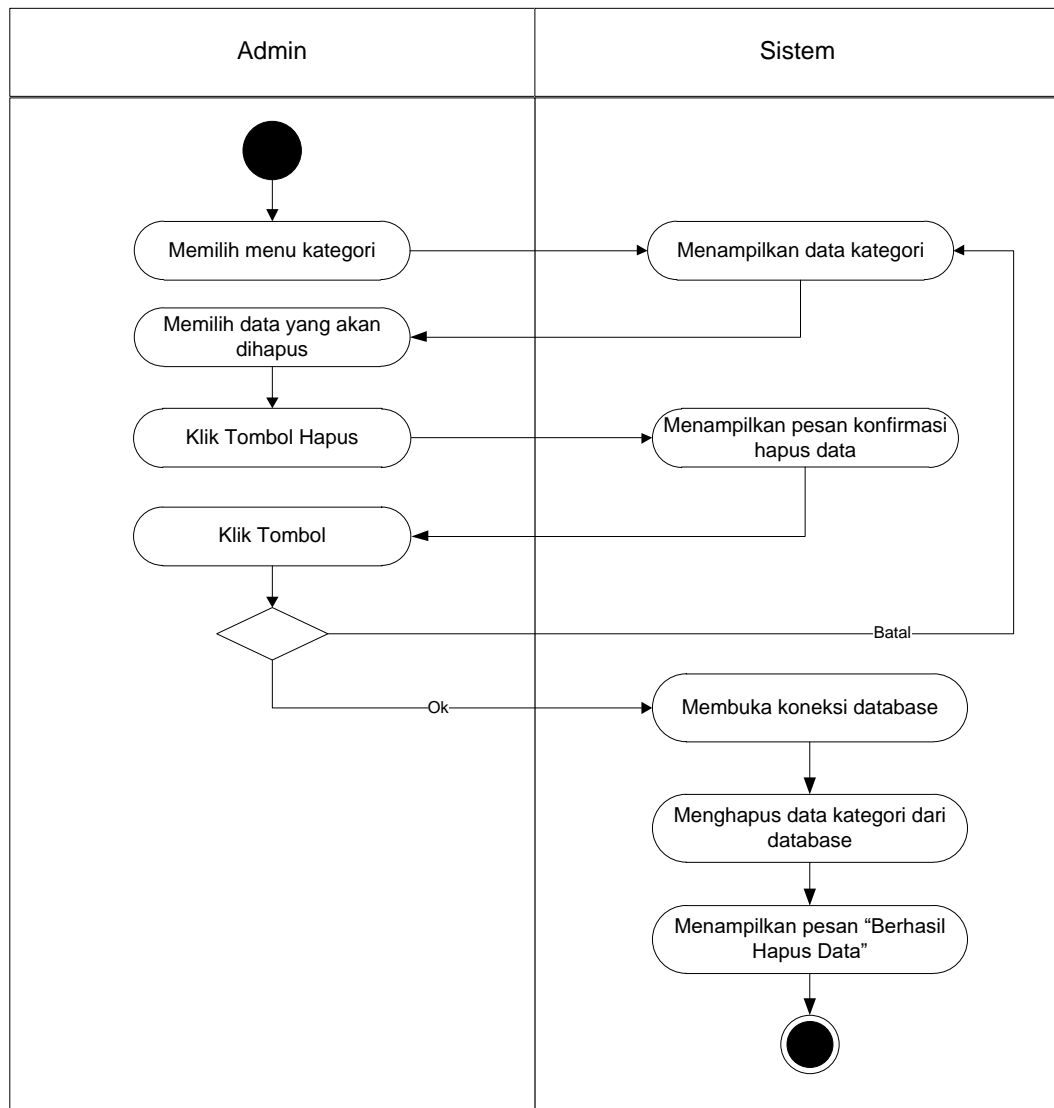


**Gambar 4.14** *Activity Diagram* Mengubah Data Kategori



c. *Activity Diagram* Menghapus Data Kategori

*Activity Diagram* menghapus kategori merupakan langkah – langkah dari Admin untuk menghapus data kategori dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.15



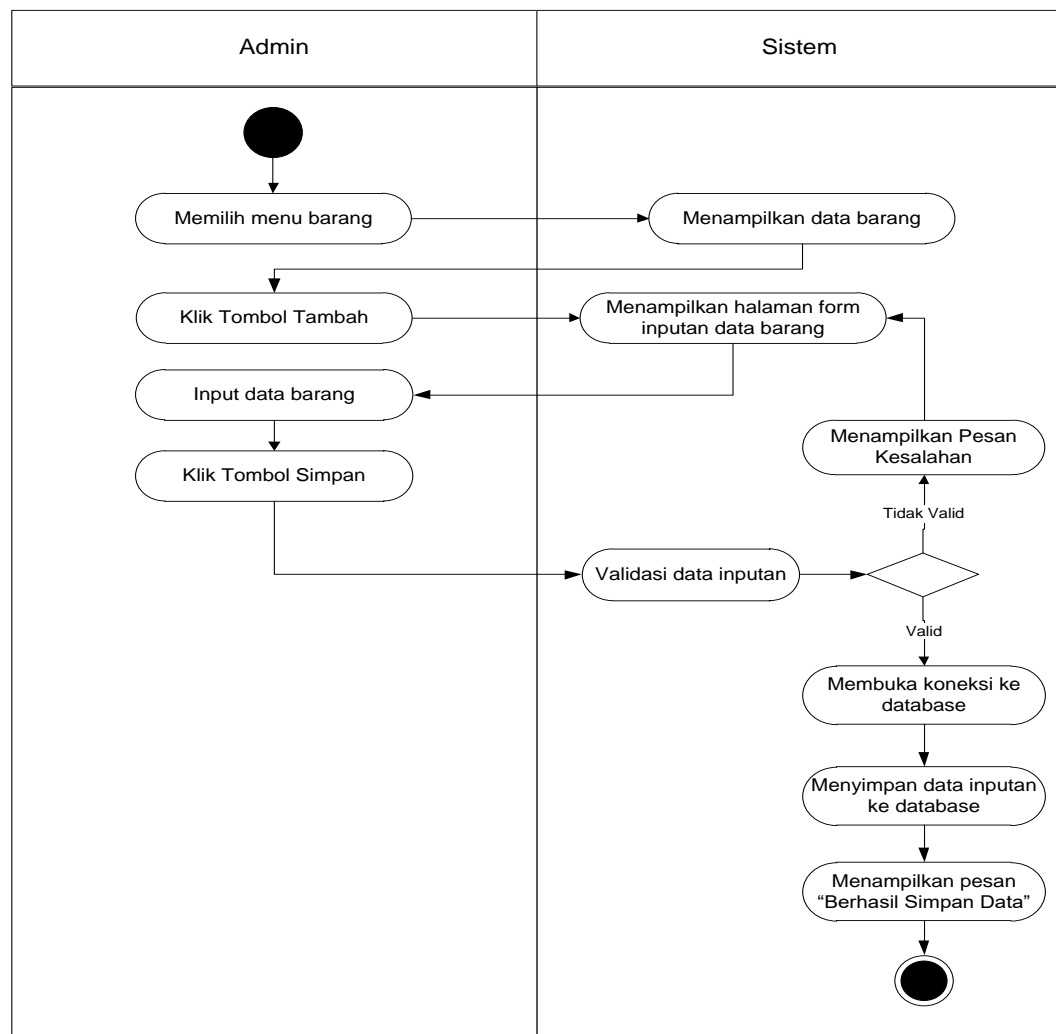
**Gambar 4.15** *Activity Diagram* Menghapus Data Kategori

## 6. *Activity Diagram* Mengelola Data Barang

*Activity Diagram* mengelola data Barang yang ditunjukkan pada Gambar 4.16 sampai dengan Gambar 4.18 di bawah ini menjelaskan aktivitas sistem saat menambah dan menghapus data Barang.

### a. *Activity Diagram* Menambah Data Barang

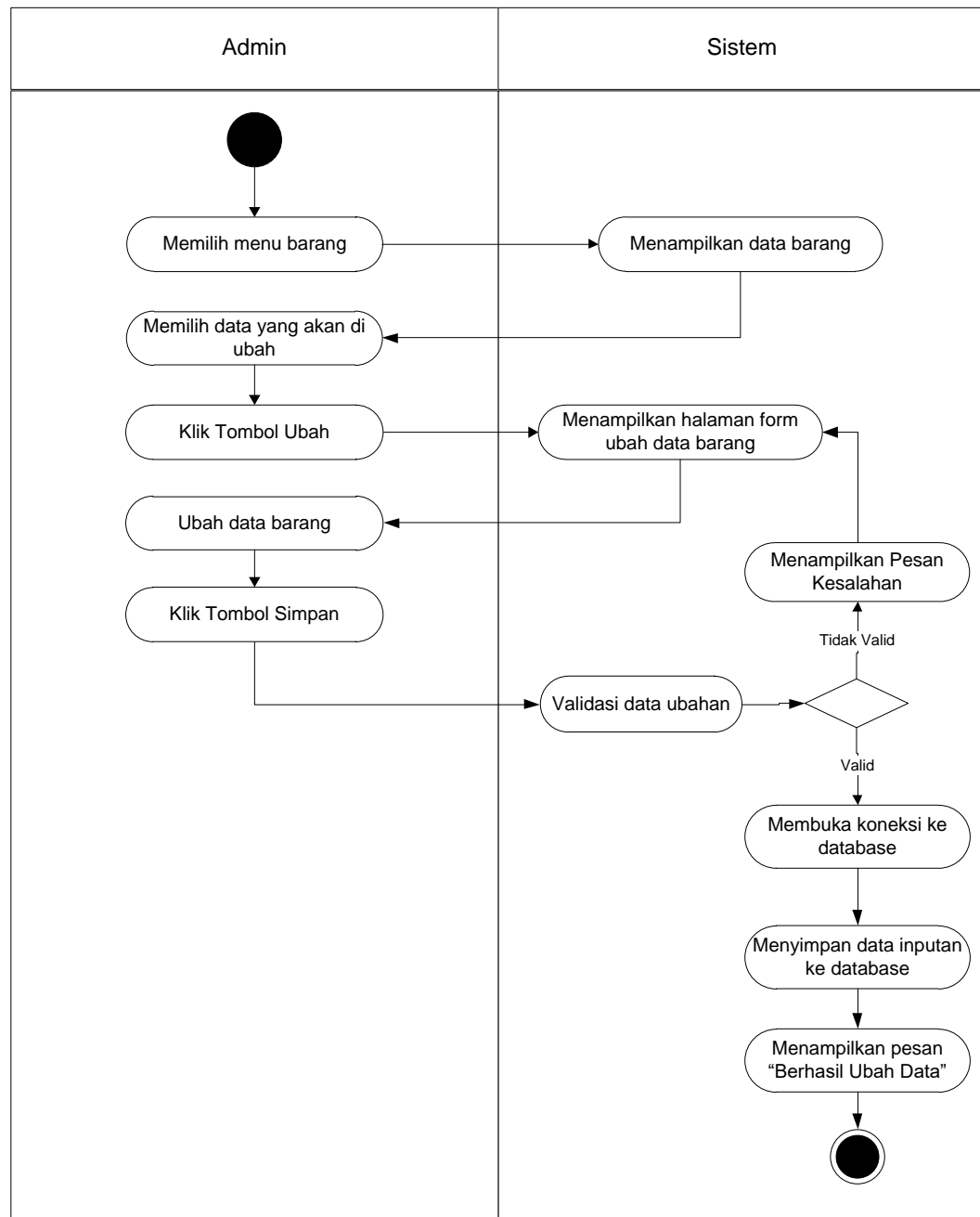
*Activity Diagram* menambah data barang menggambarkan aktifitas Admin dalam proses menambah data barang baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.16



**Gambar 4.16** *Activity Diagram* Menambah Data Barang

b. *Activity Diagram* Mengubah Data Barang

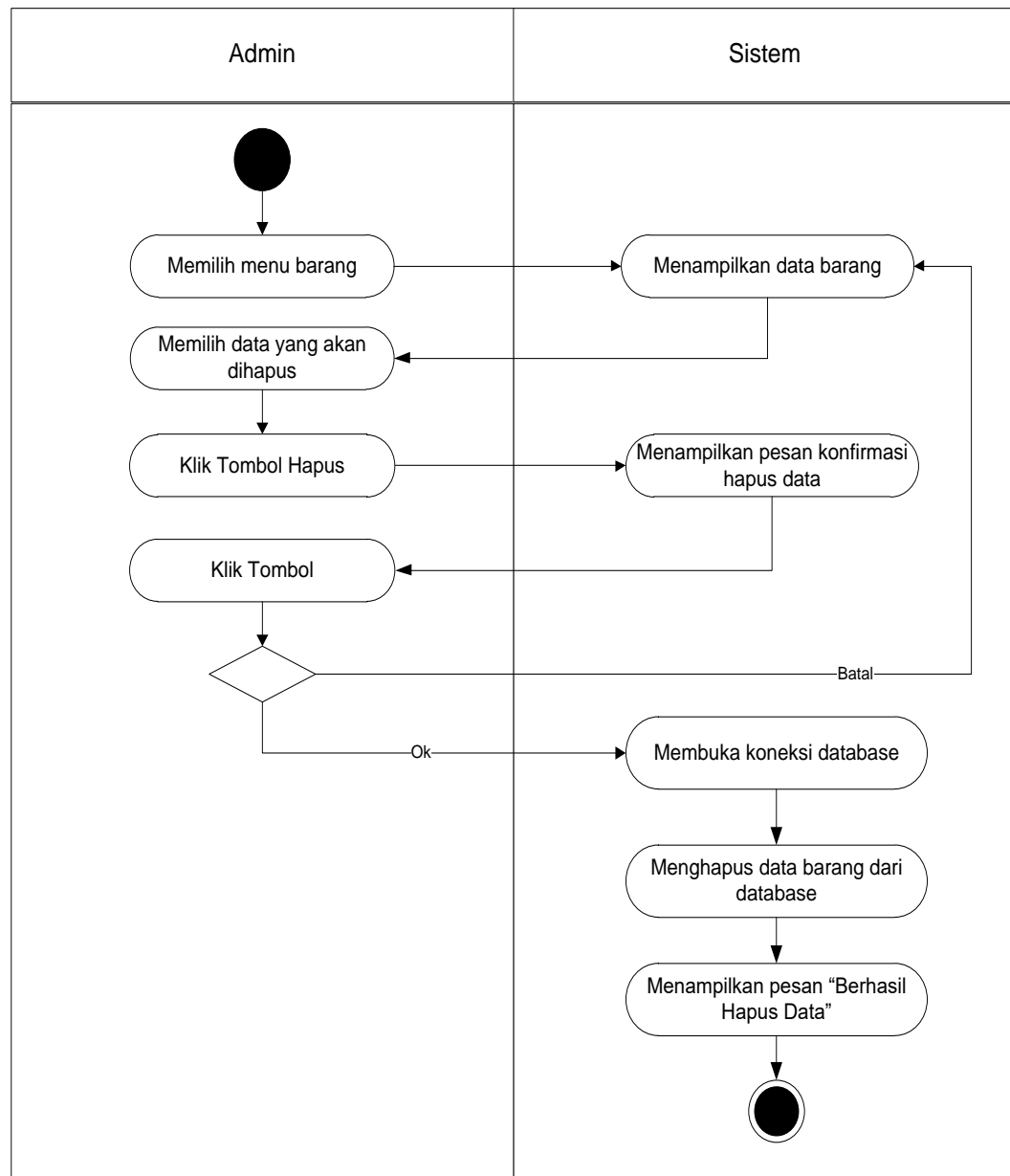
*Activity Diagram* mengubah barang merupakan langkah – langkah dari Admin untuk mengubah barang dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.17



**Gambar 4.17** *Activity Diagram* Mengubah Data Barang

c. *Activity Diagram* Menghapus Data Barang

*Activity Diagram* menghapus barang merupakan langkah – langkah dari Admin untuk menghapus data barang dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.18



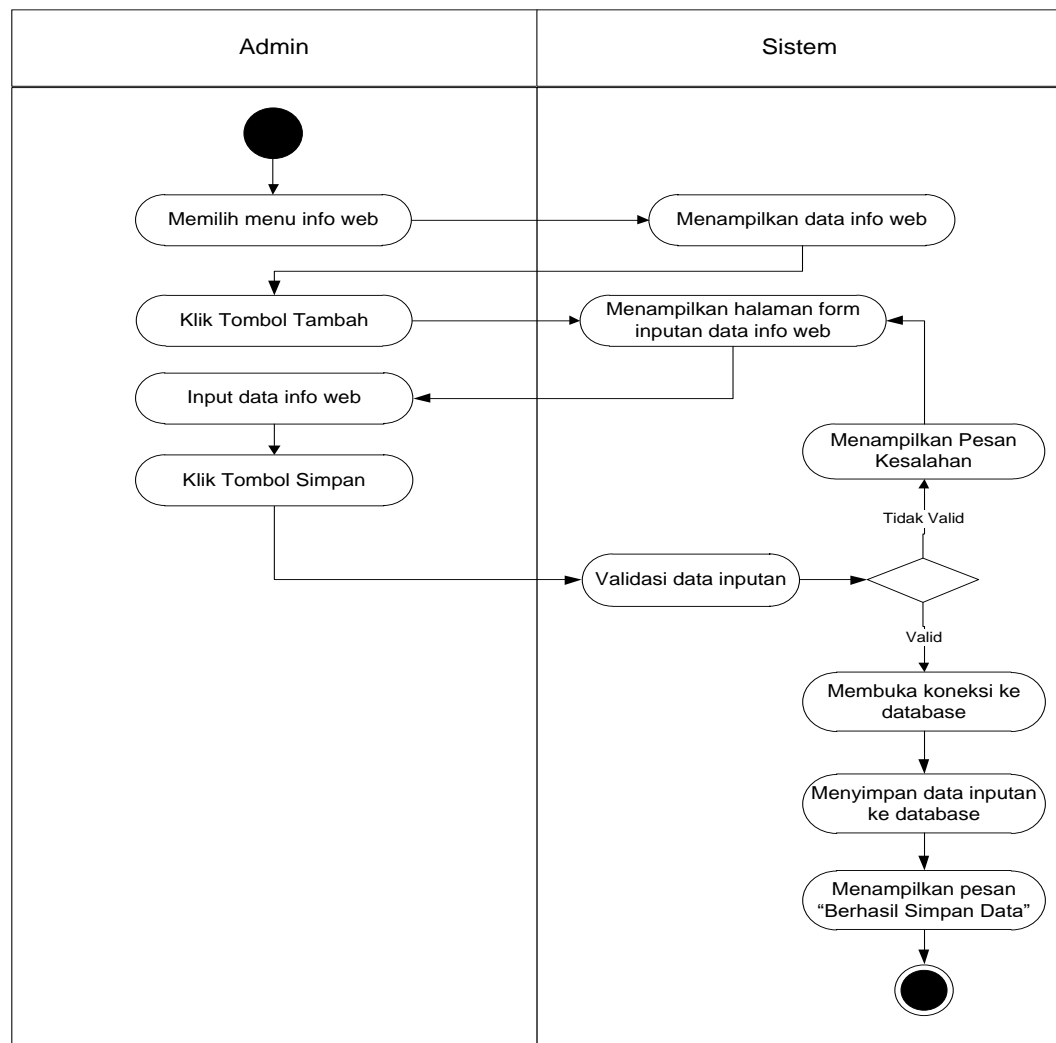
**Gambar 4.18** *Activity Diagram* Menghapus Data Barang

## 7. Activity Diagram Mengelola Data Info Web

Activity Diagram mengeloladata Info Web yang ditunjukkan pada Gambar 4.19 sampai dengan Gambar 4.21 di bawah ini menjelaskan aktivitas sistem saat menambah dan menghapus data Info Web.

### a. Activity Diagram Menambah Data Info Web

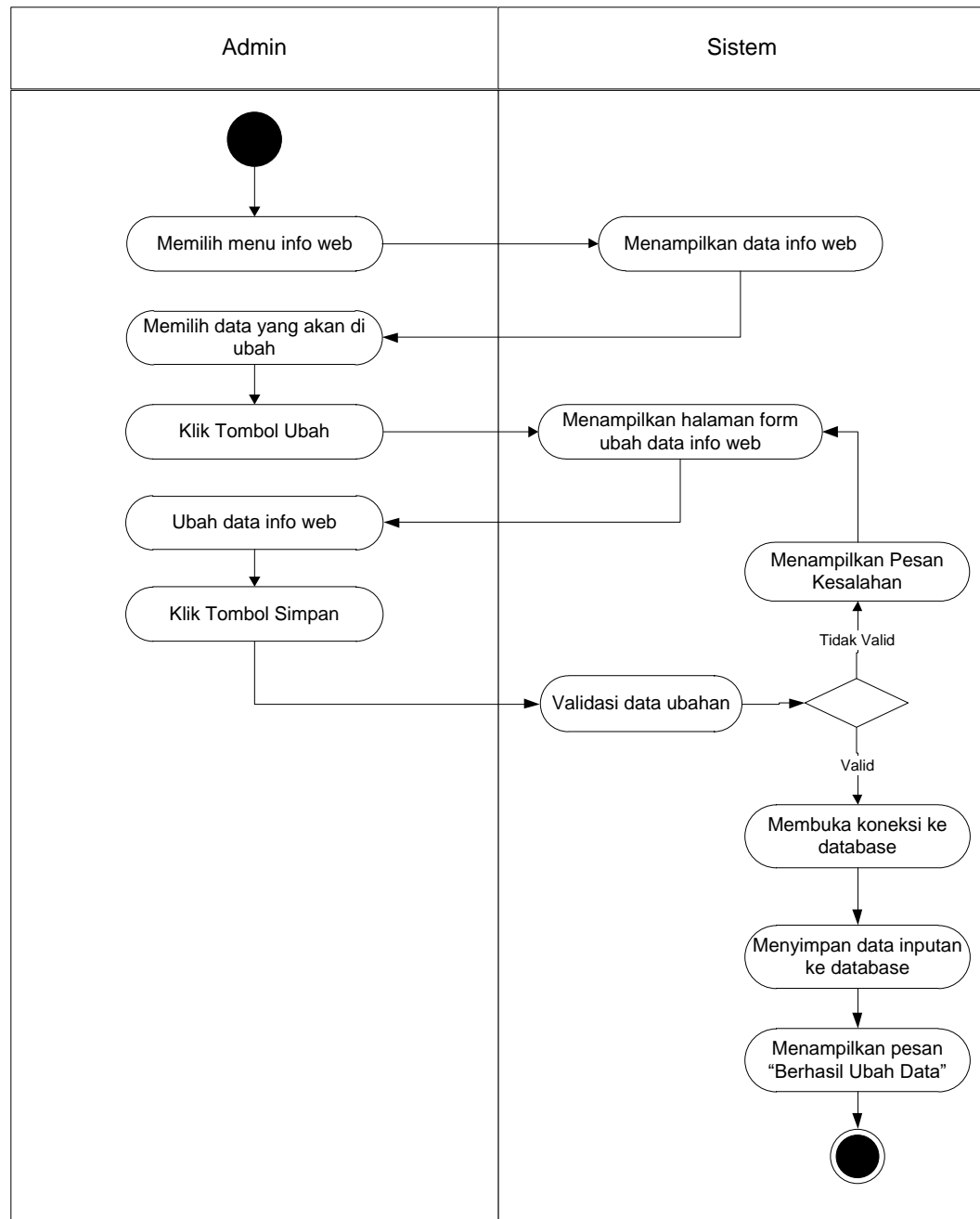
Activity Diagram menambah data Info Web menggambarkan aktifitas Admin dalam proses menambah data Info Web baru ke dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.19



**Gambar 4.19 Activity Diagram Menambah Data Info Web**

b. *Activity Diagram* Mengubah Data Info Web

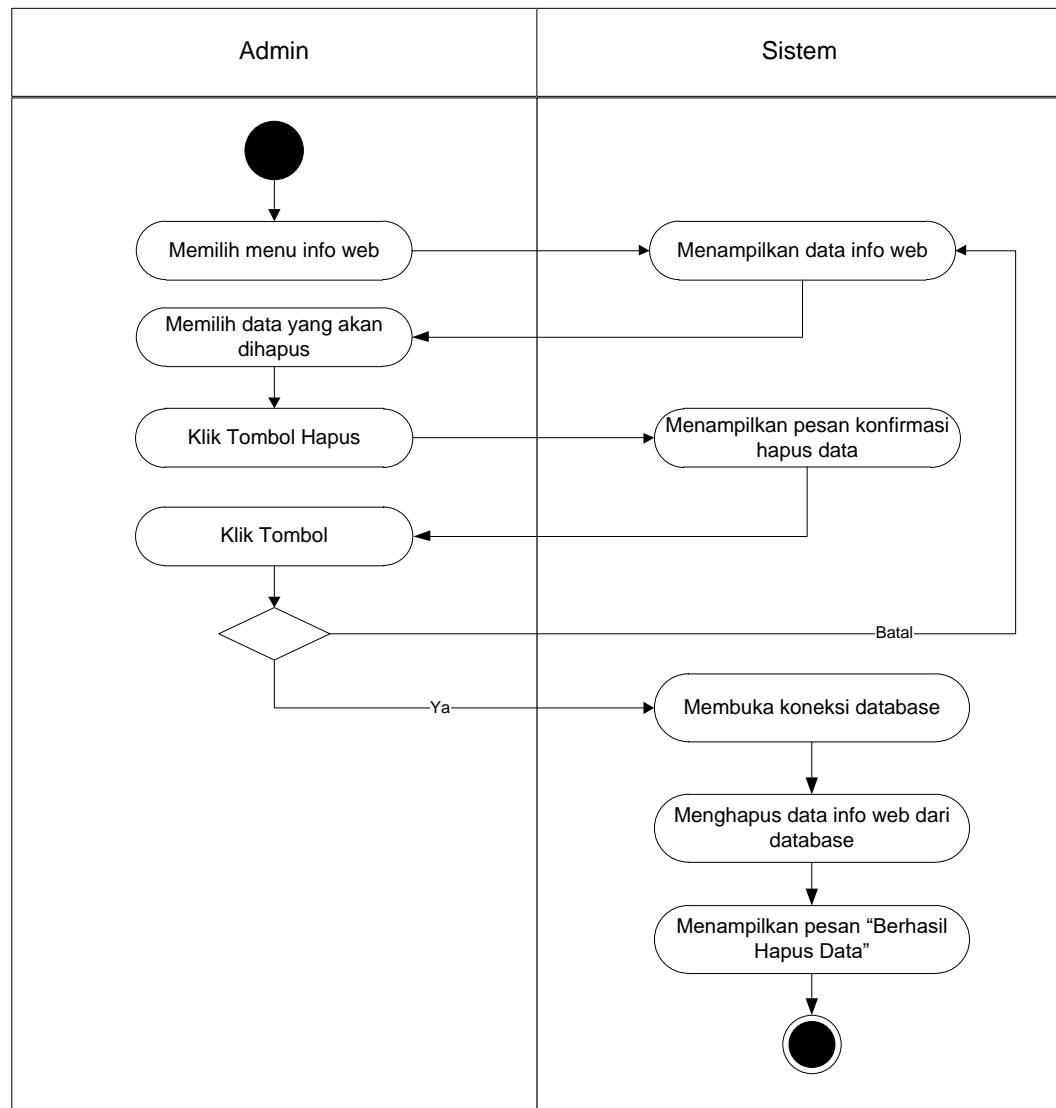
*Activity Diagram* mengubah Info Web merupakan langkah – langkah dari Admin untuk mengubah Info Web dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.20



**Gambar 4.20** *Activity Diagram* Mengubah Data Info Web

c. *Activity Diagram* Menghapus Data Info Web

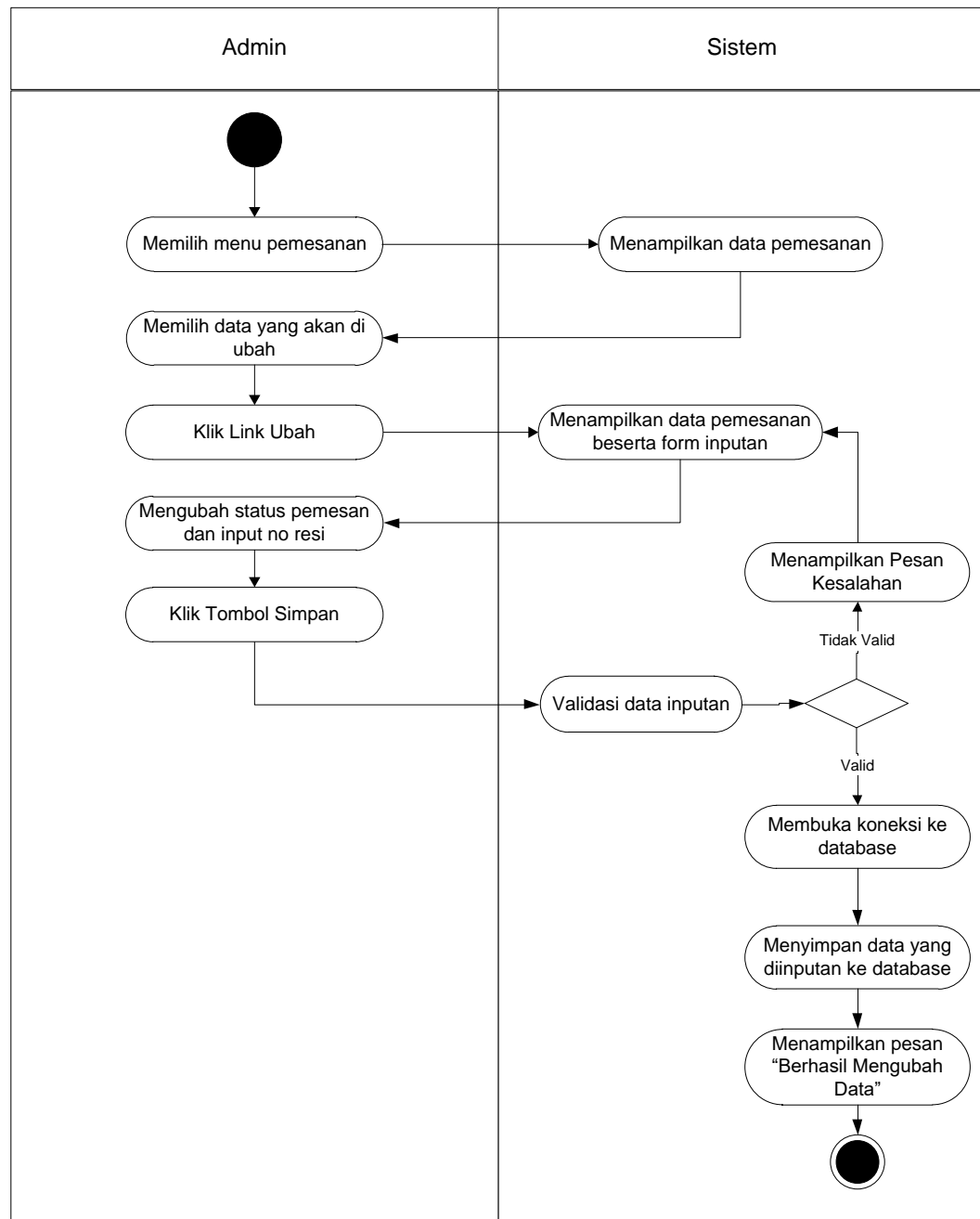
*Activity Diagram* menghapus Info Web merupakan langkah – langkah dari Admin untuk menghapus data Info Web dari dalam sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.21



**Gambar 4.21** *Activity Diagram* Menghapus Data Info Web

## 8. Activity Diagram Mengelola Data Pemesanan

*Activity Diagram* mengelola data pemesanan menjelaskan aktivitas sistem saat admin mengelola data pemesanan. Seperti ditunjukkan pada Gambar 4.22

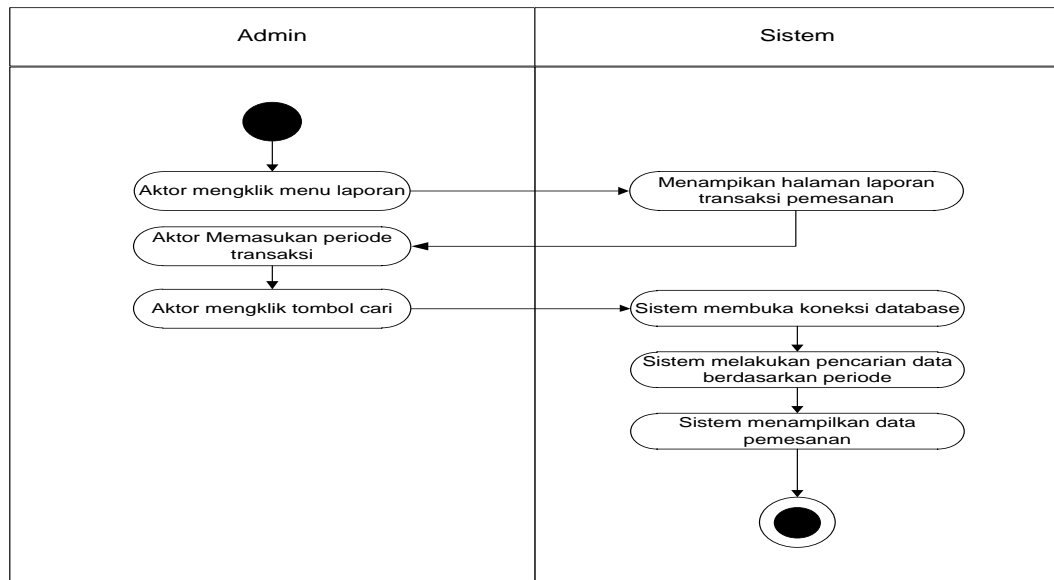


**Gambar 4.22 Activity Diagram Mengubah Data Pemesanan**



### 9. *Activity Diagram* Melihat laporan

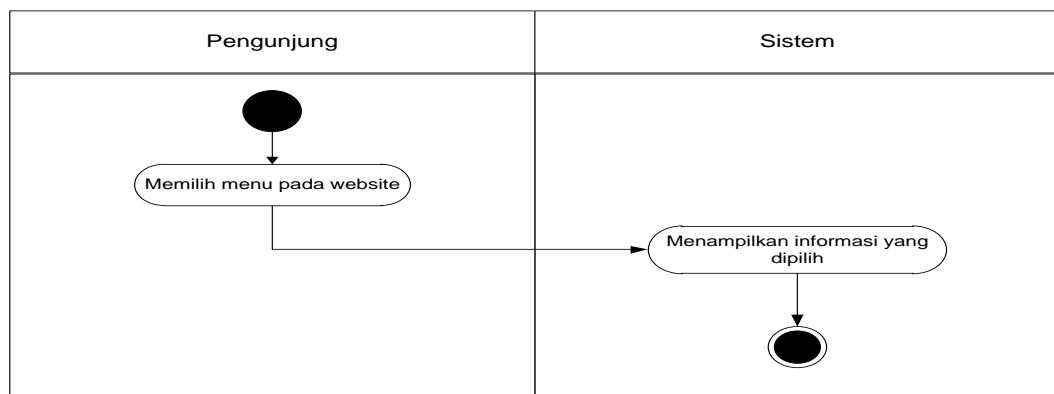
*Activity Diagram* melihat laporan ini menjelaskan aktivitas sistem saat lihat laporan. Seperti yang terlihat pada gambar 4.23



**Gambar 4.23 *Activity Diagram* Melihat laporan**

### 10. *Activity Diagram* melihat Informasi Website

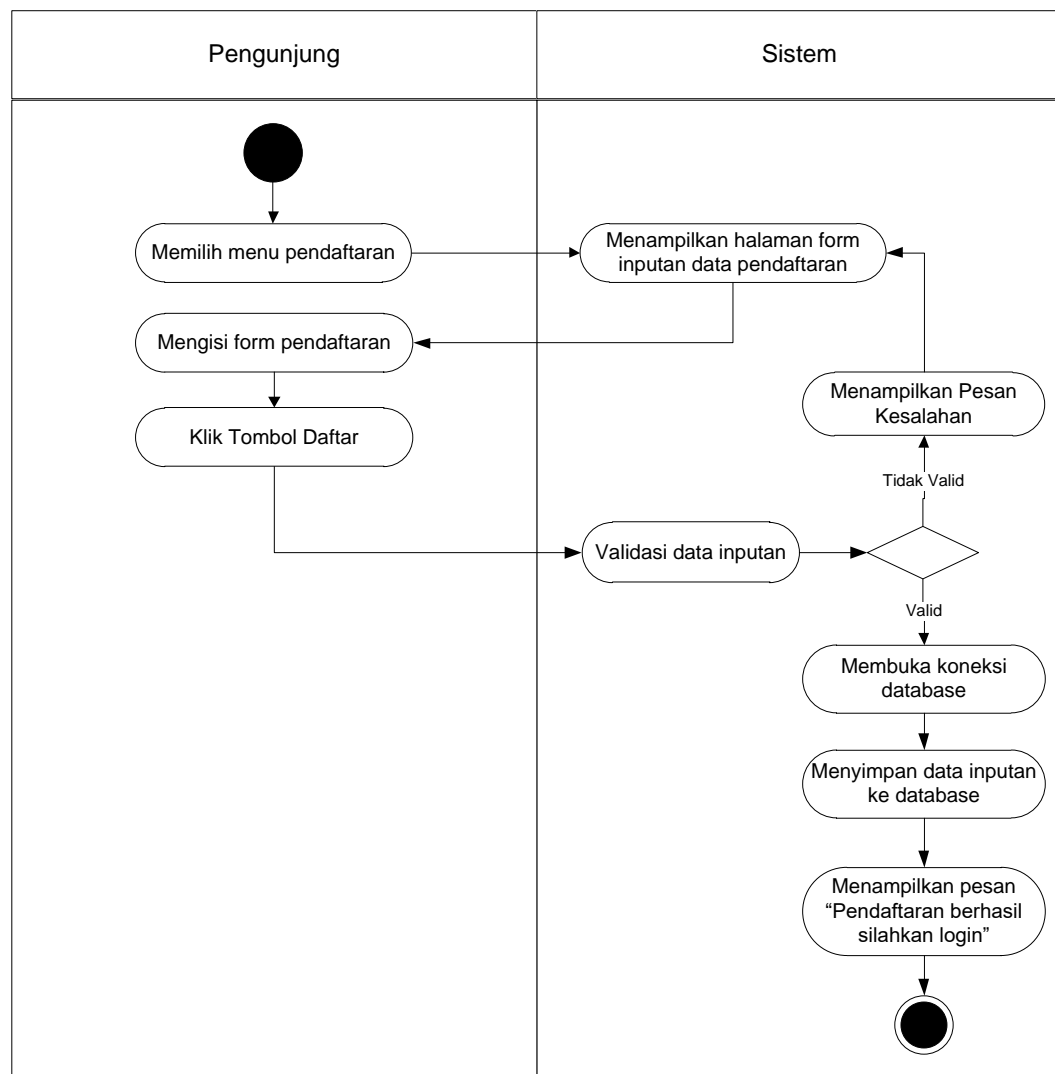
*Activity Diagram* melihat Informasi Website ini menjelaskan aktivitas pengunjung atau pelanggan saat melihat informasi pada website. Seperti yang terlihat pada gambar 4.24



**Gambar 4.24 *Activity Diagram* Melihat Informasi Website**

### 11. *Activity Diagram* melakukan Pendaftaran

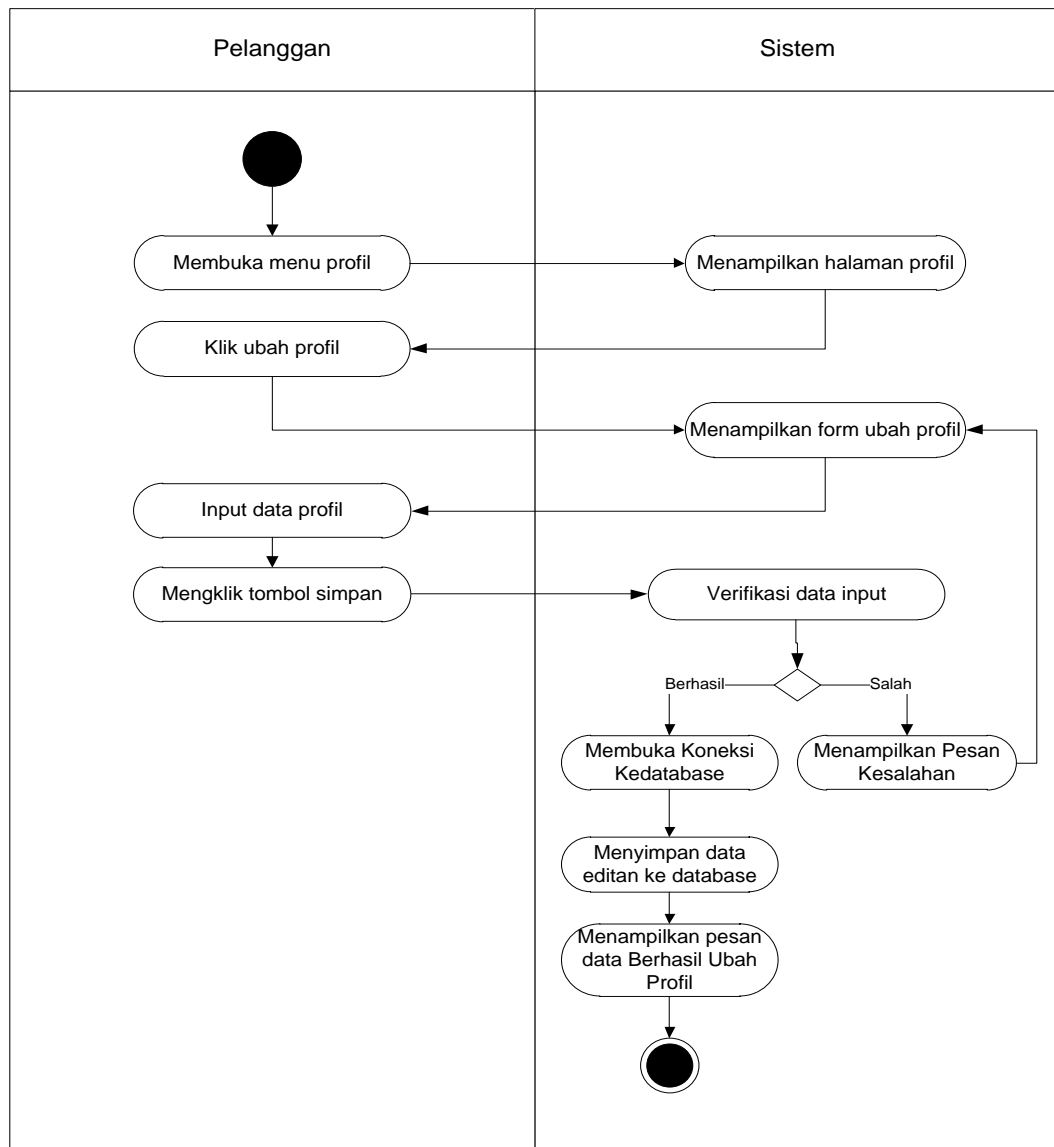
*Activity Diagram* melakukan Pendaftaran ini menjelaskan aktivitas sistem saat melakukan Pendaftaran. Seperti yang terlihat pada gambar 4.25



**Gambar 4.25 *Activity Diagram* Melakukan Pendaftaran**

### 12. *Activity Diagram* Mengubah Profil

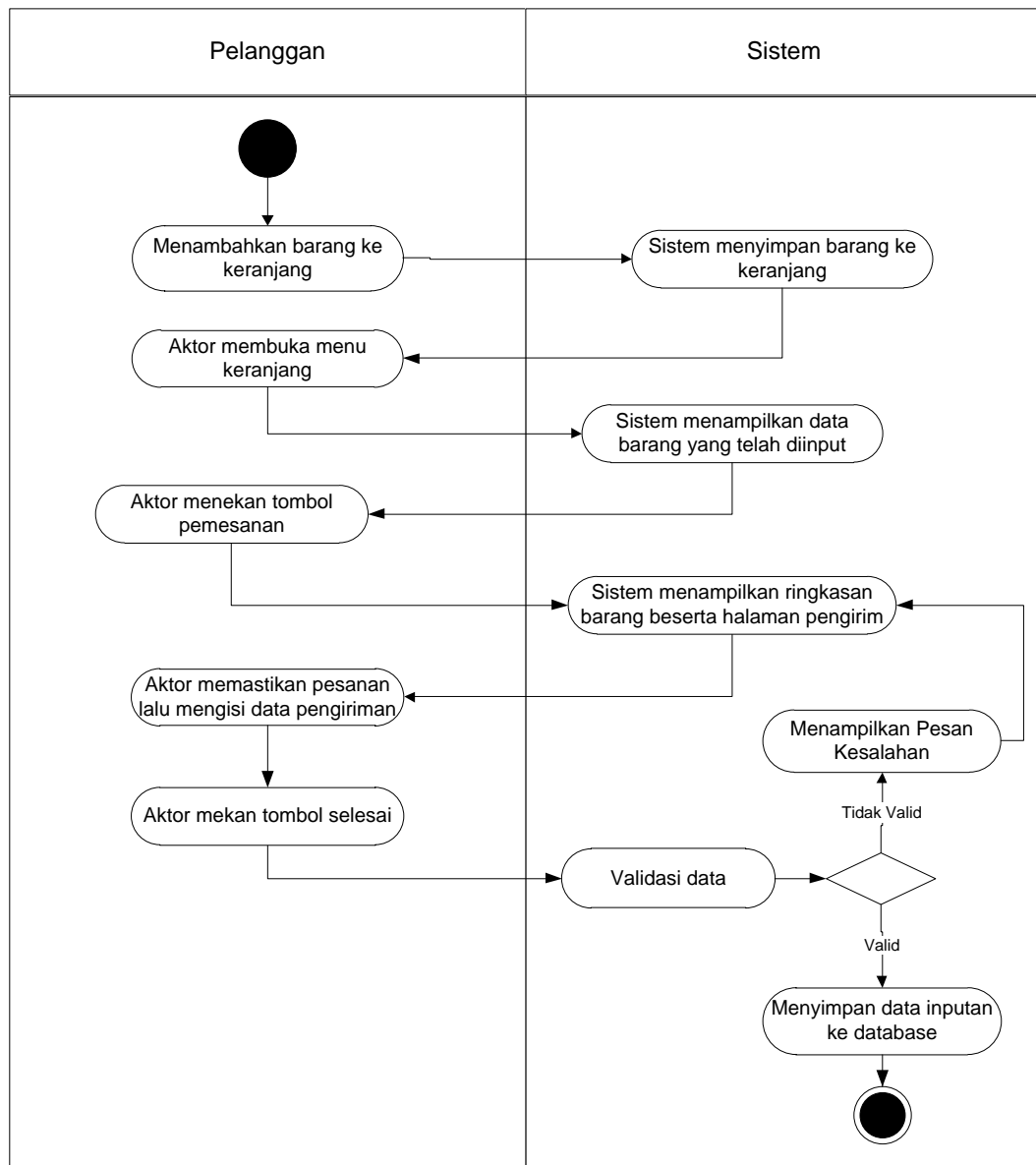
*Activity Diagram* Mengubah Profil ini menjelaskan aktivitas sistem saat melakukan ubah data profil. Seperti yang terlihat pada gambar 4.26



**Gambar 4.26 Activity Diagram Mengubah Profil**

13. *Activity Diagram* melakukan Pemesanan

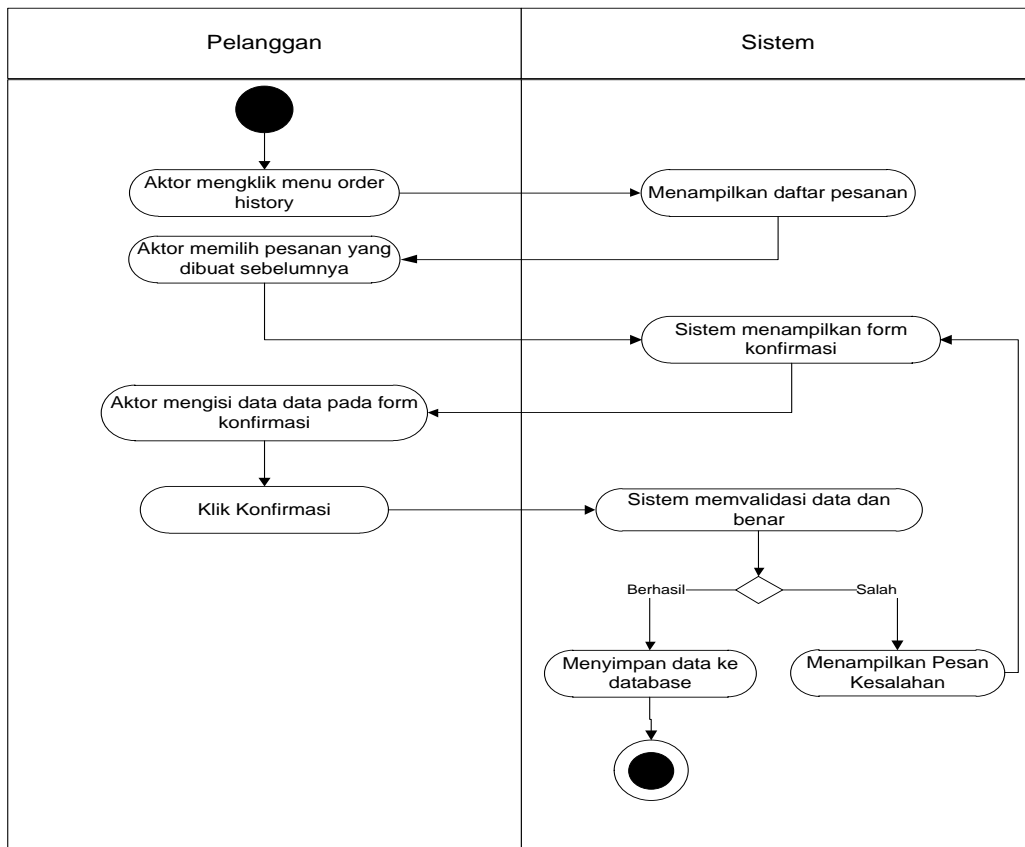
*Activity Diagram* melakukan Pemesanan ini menjelaskan aktivitas sistem pelanggan saat melakukan pemesanan, pelanggan menambah barang yang akan dipesan selanjutnya barang yang dipesan di simpan pada menu keranjang belanja, kemudian pelanggan dapat melakukan pemesanan dengan menekan tombol pemesanan. Seperti yang terlihat pada gambar 4.27



**Gambar 4.27 Activity Diagram Melakukan Pemesanan**

#### 14. Activity Diagram Mengelola Konfirmasi Pembayaran

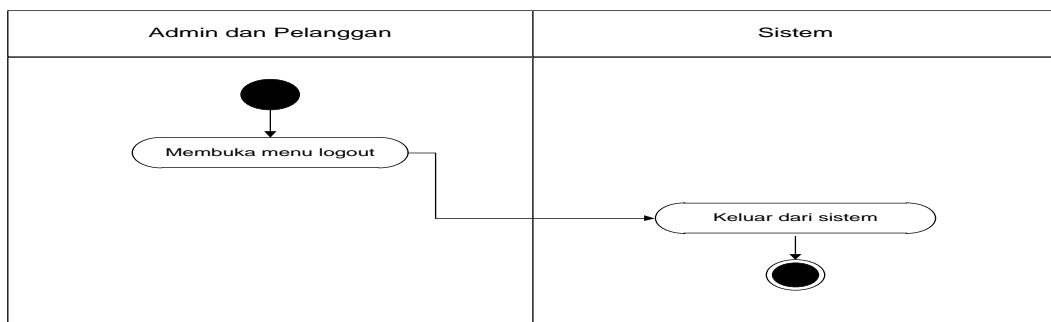
*Activity Diagram* melakukan Mengelola Konfirmasi Pembayaran ini menjelaskan aktivitas sistem saat melakukan Mengelola Konfirmasi Pembayaran. Seperti yang terlihat pada gambar 4.28.



**Gambar 4.28 Activity Diagram Melakukan Konfirmasi Pembayaran**

15. Activity Diagram Logout

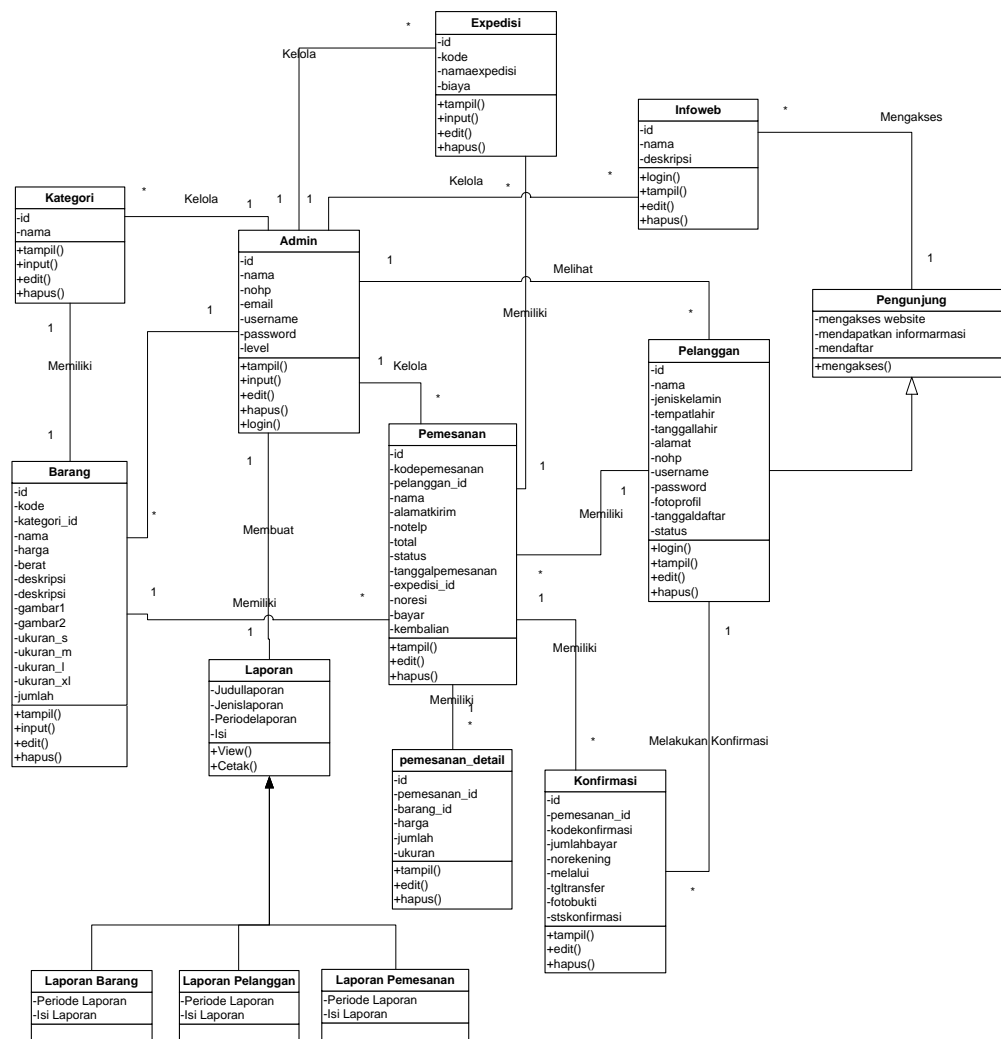
Activity Diagram logout yang dilakukan untuk keluar dari sistem dan mengunci akses terhadap pengolahan data. Seperti yang terlihat pada gambar 4.29.



**Gambar 4.29 Activity Diagram Logout**

### 4.2.6 Class Diagram

Class Diagram menggambarkan perilaku dan keadaan dengan menghubungkan antara kelas-kelas. Tampilan dari class diagram dapat dilihat pada gambar 4.30



Gambar 4.30 Class Diagram Penjualan Toko Hani Sejahtera

### 4.3 PERANCANGAN PERANGKAT LUNAK

Perancangan perangkat lunak merupakan rancangan dari form-form masukan dari sistem dan keluaran dari sistem. Berikut adalah rancangan *input* dan *output* :

#### 4.3.1 Rancangan *Input*

Rancangan rancangan tampilan *input* dari sistem yang akan dibangun adalah sebagai berikut:

##### 1. Rancangan Halaman Pendaftaran

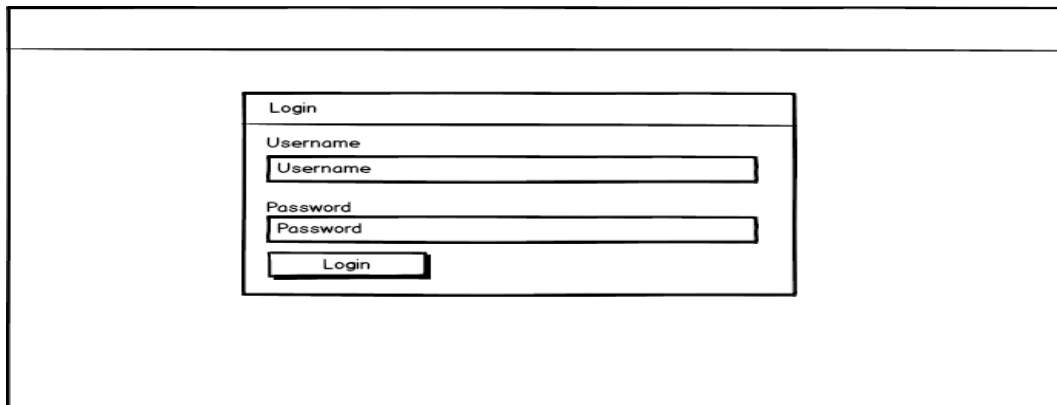
Berikut adalah rancangan dari halaman Pendaftaran pengunjung yang ingin menjadi pelanggan. Seperti terlihat pada gambar 4.31 :

The image shows a wireframe of a web registration page. At the top, there's a browser window with a search bar and a 'Login' dropdown. Below that is a navigation bar with 'Home', 'Cara Pesan', and 'Hubungi Kami', along with another search bar and a 'Home Daftar' button. The main content area is split into two columns. The left column has a placeholder for a logo and a 'KATEGORI' list with five items. The right column is the 'REGISTRASI' form, which includes fields for Name, Gender, Birthplace, Birthdate, Address, Phone Number, Username, and Password, and a 'Daftar' button. The footer contains placeholder text for store name, categories, and payment methods, and a copyright notice for 2021.

Gambar 4.31 Rancangan Halaman Pendaftaran

## 2. Rancangan Halaman *Login Admin*

Berikut adalah rancangan dari halaman *login admin*. Seperti terlihat pada gambar 4.32 :

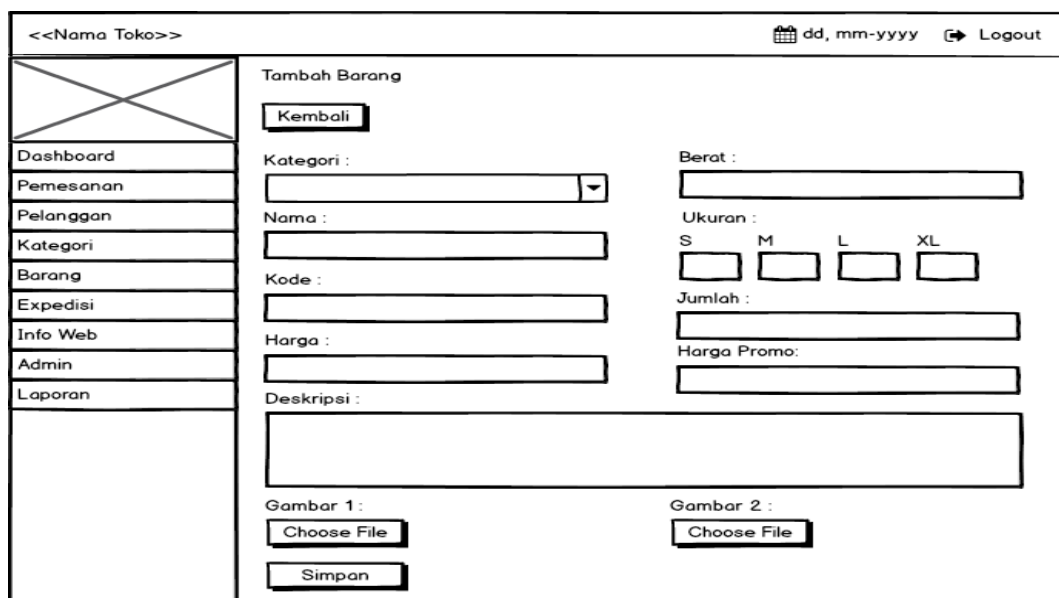


The image shows a simple login form titled "Login". It contains three input fields: "Username" with the placeholder text "Username", "Password" with the placeholder text "Password", and a "Login" button below them.

**Gambar 4.32 Rancangan Halaman *Login Admin***

## 3. Rancangan Halaman Form Data Barang

Rancangan halaman form data Barang merupakan halaman yang dapat diakses oleh admin untuk menambah dan mengubah Barang. Seperti terlihat pada gambar 4.33 :



The image shows a web page for adding a new item. At the top, there is a navigation bar with "<<Nama Toko>>" on the left, a date "dd, mm-yyyy" in the center, and a "Logout" button on the right. On the left side, there is a vertical menu with links: Dashboard, Pemesanan, Pelanggan, Kategori, Barang, Expedisi, Info Web, Admin, and Laporan. The main content area is titled "Tambah Barang" and includes a "Kembali" button. The form fields are: "Kategori:" (a dropdown menu), "Nama:" (a text input), "Kode:" (a text input), "Harga:" (a text input), "Deskripsi:" (a large text area), "Berat:" (a text input), "Ukuran:" (four radio buttons labeled S, M, L, XL), "Jumlah:" (a text input), and "Harga Promo:" (a text input). At the bottom, there are two "Gambar 1:" and "Gambar 2:" sections, each with a "Choose File" button, and a "Simpan" button.

**Gambar 4.33 Rancangan Halaman Form Data Barang**



#### 4. Rancangan Halaman Form Data Kategori

Rancangan halaman form data kategori merupakan halaman yang dapat diakses oleh admin untuk menambah dan mengubah kategori. Seperti terlihat pada gambar 4.34 :

**Gambar 4.34 Rancangan Halaman Form Kategori**

#### 5. Rancangan Halaman Form Data Expedisi

Rancangan halaman form data Expedisi merupakan halaman yang dapat diakses oleh admin untuk menambah data Expedisi. Seperti terlihat pada gambar 4.35 :

**Gambar 4.35 Rancangan Halaman Form Data Expedisi**

## 6. Rancangan Halaman Form Data Admin

Rancangan halaman form data Admin merupakan halaman yang dapat diakses oleh admin untuk menambah data Admin. Seperti terlihat pada gambar 4.36 :

The screenshot shows a web application interface for adding an admin. At the top, there is a header with the store name '<<Nama Toko>>', a date 'dd, mm-yyyy', and a 'Logout' button. On the left, there is a sidebar menu with the following items: Dashboard, Pemesanan, Pelanggan, Kategori, Barang, Expedisi, Info Web, Admin, and Laporan. The main content area is titled 'Tambah Admin' and contains a 'Kembali' button at the top. Below it are several input fields: 'Nama :', 'No Hp :', 'Email :', 'Username :', 'Password', and 'Level :'. The 'Level' field is a dropdown menu. At the bottom of the form is a 'Simpan' button.

**Gambar 4.36 Rancangan Halaman Form Data Admin**

## 7. Rancangan Halaman Form Data Info Web

Rancangan halaman form data Info Web merupakan halaman yang dapat diakses oleh admin untuk menambah data Info Web. Seperti terlihat pada gambar 4.37 :

The screenshot shows a web application interface for adding website information. At the top, there is a header with the store name '<<Nama Toko>>', a date 'dd, mm-yyyy', and a 'Logout' button. On the left, there is a sidebar menu with the following items: Dashboard, Pemesanan, Pelanggan, Kategori, Barang, Expedisi, Info Web, Admin, and Laporan. The main content area is titled 'Tambah Info Web' and contains a 'Kembali' button at the top. Below it are two input fields: 'Nama :' and 'Deskripsi :'. At the bottom of the form is a 'Simpan' button.

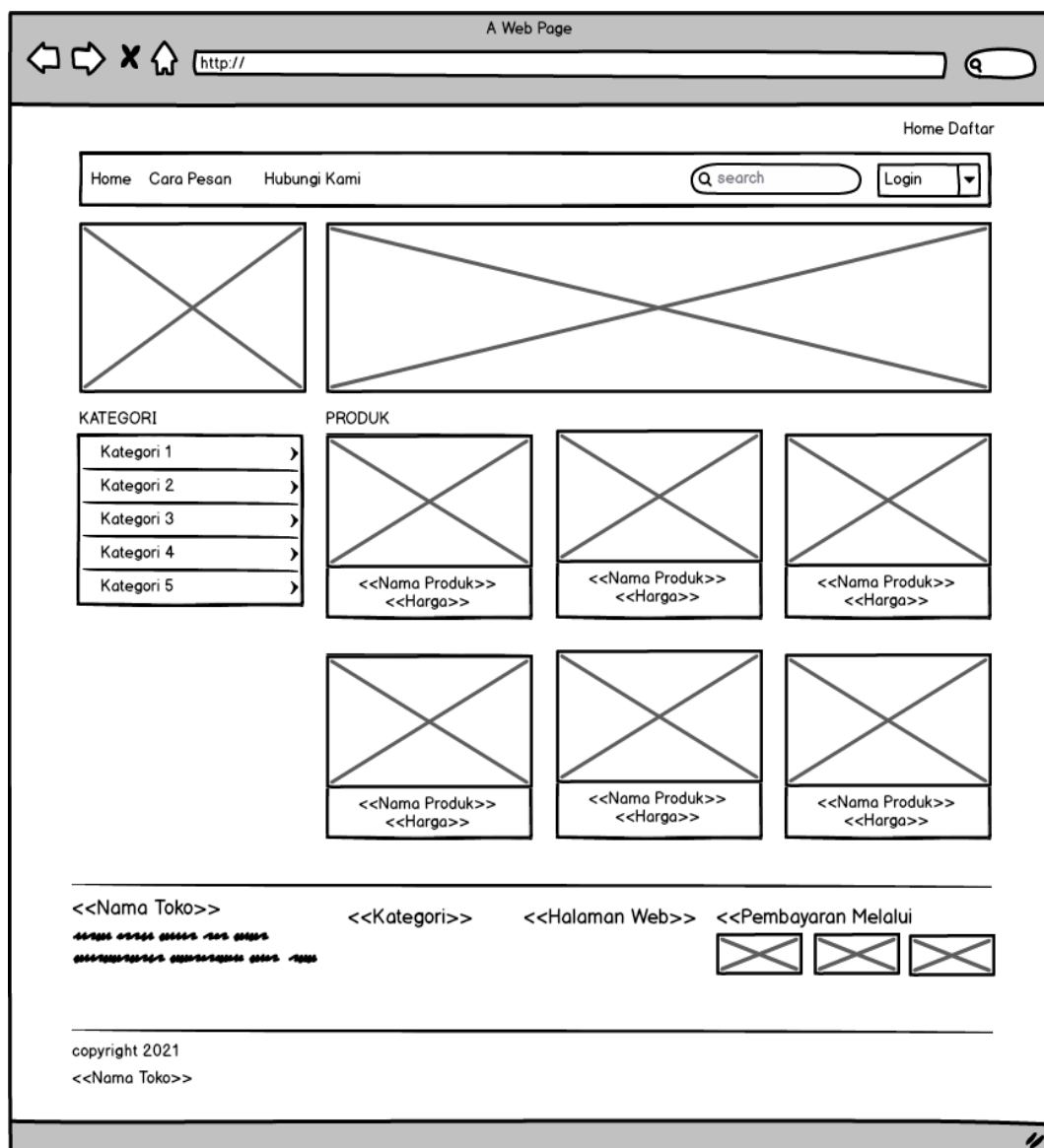
**Gambar 4.37 Rancangan Halaman Form Data Info web**

### 4.3.2 Rancangan Output

Rancangan rancangan tampilan *output* dari sistem yang akan dibangun adalah sebagai berikut:

#### 1. Rancangan Halaman Utama *Website*

Berikut halaman utama *Website* yang pertama kali dilihat ketika membuka *Website*. Seperti yang terlihat pada Gambar 4.38:



Gambar 4.38 Rancangan Halaman Utama

## 2. Rancangan Halaman Keranjang

Berikut adalah halaman keranjang pemesanan pelanggan, didalam halaman ini terdapat data data pemesanan yang telah ditambah ke keranjang pemesanan. Seperti terlihat pada gambar 4.39 :

Home Daftar

Home Cara Pesan Hubungi Kami

[Home](#) > List Cart

List Cart

Hapus	Barang	Ukuran	Harga	Jumlah	Total

KATEGORI

- Kategori 1 >
- Kategori 2 >
- Kategori 3 >
- Kategori 4 >
- Kategori 5 >

Data Pengiriman	Total Biaya Keseluruhan
Nama Penerima : <input type="text"/>	Total Berat : <<999.999>>
Alamat Penerima : <input type="text"/>	Sub Total : <<999.999>>
No HP : <input type="text"/>	
Jasa Pengiriman : <input type="text"/>	

<<Nama Toko>> <<Kategori>> <<Halaman Web>> <<Pembayaran Melalui

copyright 2021 <<Nama Toko>>

**Gambar 4.39 Rancangan Halaman Keranjang Pemesanan**

### 3. Rancangan Halaman Mengelola Data Barang

Rancangan mengelola data Barang merupakan halaman yang dapat diakses oleh admin untuk menambah, mengubah dan menghapus data Barang. Seperti terlihat pada gambar 4.40 :

**Gambar 4.40 Rancangan Halaman Mengelola Data Barang**

### 4. Rancangan Halaman Mengelola Data Kategori

Rancangan mengelola data kategori merupakan halaman yang dapat diakses oleh admin untuk menambah, mengubah dan menghapus data kategori. Seperti terlihat pada gambar 4.41 :

**Gambar 4.41 Rancangan Halaman Data Kategori**

## 5. Rancangan Halaman Mengelola Data Pelanggan

Rancangan mengelola data pelanggan merupakan halaman yang dapat diakses oleh admin untuk menambah, mengubah dan menghapus data pelanggan. Seperti terlihat pada gambar 4.42 :

**Gambar 4.42 Rancangan Halaman Data Pelanggan**


## 6. Rancangan Halaman Mengelola Data Expedisi

Rancangan mengelola data Expedisi merupakan halaman yang dapat diakses oleh admin untuk mengubah Expedisi. Seperti terlihat pada gambar 4.43 :

**Gambar 4.43 Rancangan Halaman Mengelola Expedisi**

7. Rancangan Laporan Barang

Rancangan laporan Barang ini berfungsi ketika admin ingin melihat data laporan semua Barang. Tampilan halaman laporan Barang dapat dilihat pada gambar 4.44:



<< NAMA TOKO >>

<< Alamat Toko >>

---

LAPORAN DATA BARANG

No	Kode	Nama	Kategori	Harga	Stok	Ukuran S	Ukuran M	Ukuran L	Ukuran XL	Jumlah	Opsi

Jambi, dd-mm-yyyy

Mengetahui


  

Pimpinan

Gambar 4.44 Rancangan Halaman Laporan Barang

8. Rancangan Laporan Pelanggan

Rancangan laporan pelanggan ini berfungsi ketika admin ingin melihat data laporan pelanggan. Tampilan halaman laporan pelanggan dapat dilihat pada gambar 4.45:



<< NAMA TOKO >>

<< Alamat Toko >>

---

LAPORAN DATA PELANGGAN

No	Nama	No Telp	Alamat

Jambi, dd-mm-yyyy

Mengetahui


  

Pimpinan

Gambar 4.45 Rancangan Halaman Laporan Pelanggan

## 9. Rancangan Laporan Pemesanan Perbulan

Rancangan laporan pemesanan perbulan ini berfungsi ketika admin ingin melihat data laporan pemesanan perbulan. Tampilan halaman laporan pemesanan perbulan dapat dilihat pada gambar 4.46:

	<< NAMA TOKO >> << Alamat Toko >>				
LAPORAN PEMESANAN					
Bulan : mm Tahun : yyyy					
No	Kode Pemesanan	Nama	Total	Expedisi	Total Biaya
Total					99.999.999
Jambi, dd-mm-yyyy Pimpinan					
Pimpinan					

**Gambar 4.46 Rancangan Halaman Laporan Pemesanan Perbulan**

## 4.4 RANCANGAN STRUKTUR DATA

Struktur data yang digunakan dalam sistem yang baru, dirancang dalam beberapa tabel sebagai berikut :

### 1. Tabel Admin

Tabel admin digunakan untuk menyimpan data-data admin. Nama tabel admin dalam *database* adalah admin. Seperti yang terlihat pada Tabel 4.16:



**Tabel 4.16 Rancangan Tabel Admin**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	6	Id admin ( <i>Primary key</i> )
2	Nama	<i>Varchar</i>	30	Nama lengkap dari admin
3	Nohp	<i>Varchar</i>	12	No Hp
4	<i>Email</i>	<i>Varchar</i>	30	<i>Email</i>
5	<i>Username</i>	<i>Varchar</i>	20	<i>Username</i> dari admin
6	<i>Password</i>	<i>Varchar</i>	10	<i>Password</i>
7	Level	<i>Varchar</i>	20	Level dari admin

## 2. Tabel Pelanggan

Tabel pelanggan dikelola oleh Admin untuk menyimpan data-data pelanggan. Nama tabel pelanggan dalam *database* adalah pelanggan. Seperti yang terlihat pada Tabel 4.17 :

**Tabel 4.17 Rancangan Struktur Tabel Pelanggan**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	5	Id Pelanggan ( <i>Primary key</i> )
2	Nama	<i>Varchar</i>	30	Nama Pelanggan
3	Jeniskelamin	<i>Enum</i>	L,P	Jenis Kelamin
4	Tempatlahir	<i>Varchar</i>	20	Tempat lahir
5	Tanggallahir	<i>Date</i>	-	Tanggal lahir
6	Alamat	<i>Varchar</i>	100	Alamat
7	Nohp	<i>Varchar</i>	15	No hp
8	<i>Username</i>	<i>Varchar</i>	20	Username
9	<i>Password</i>	<i>Varchar</i>	10	Password
10	Fotoprofil	<i>Varchar</i>	50	Link foto profil
11	Tanggaldaftar	<i>Datetime</i>	-	Tanggal daftar
12	Status	<i>Varchar</i>	10	Status pelanggan

## 3. Tabel Barang

Tabel Barang dikelola oleh admin untuk menyimpan data Barang. Nama tabel Barang dalam *database* adalah Barang. Seperti yang terlihat pada Tabel 4.18 :

**Tabel 4.18 Rancangan Struktur Tabel Barang**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	5	Id barang ( <i>Primary key</i> )
2	Kode	<i>Varchar</i>	10	Kode barang
3	Kategori_id	<i>Int</i>	5	Id kategori ( <i>Foreign key</i> )
4	Nama	<i>Varchar</i>	50	Nama Barang
5	Harga	<i>Int</i>	7	Harga
6	Berat	<i>Int</i>	3	Berat
7	Deskripsi	<i>Text</i>	-	Deskripsi barang
8	Gambar1	<i>Varchar</i>	-	Gambar 1
9	Gambar2	<i>Varchar</i>	-	Gambar 2
10	Ukuran_s	<i>Int</i>	3	Ukuran S
11	Ukuran_m	<i>Int</i>	3	Ukuran M
12	Ukuran_l	<i>Int</i>	3	Ukuran L
13	Ukuran_xl	<i>Int</i>	3	Ukuran XL
14	Jumlah	<i>Int</i>	3	Jumlah barang
15	Promo	<i>Int</i>	6	Harga promo

#### 4. Tabel Kategori

Tabel kategori dikelola oleh admin untuk menyimpan data kategori barang. Nama tabel kategori dalam *database* adalah kategori. Seperti yang terlihat pada Tabel 4.19 :

**Tabel 4.19 Rancangan Struktur Tabel Kategori**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	3	Id Kategori ( <i>Primary key</i> )
2	Nama	<i>Varchar</i>	20	Nama
3	Jenisukuran	<i>Enum</i>	'Banyak', 'Satu'	Jenis Ukuran

#### 5. Tabel Pemesanan

Tabel Pemesanan merupakan tabel yang digunakan oleh pelanggan untuk melakukan pemesanan Barang. Nama tabel Pemesanan dalam *database* adalah pemesanan. Seperti yang terlihat pada Tabel 4.20 :

**Tabel 4.20 Rancangan Struktur Tabel pemesanan**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	5	Id pemesanan ( <i>Primary key</i> )
2	Kodepemesanan	<i>Varchar</i>	10	Kode pemesanan
3	Pelanggan_id	<i>Int</i>	5	Id pelanggan ( <i>Foreign key</i> )
4	Nama	<i>Varchar</i>	30	Nama pelanggan
5	Alamatkirim	<i>Varchar</i>	50	Alamat kirim
6	Notelp	<i>Varchar</i>	15	No telp
7	Total	<i>Int</i>	8	Total
8	Status	<i>Varchar</i>	20	Status pemesanan
9	Tanggalpemesanan	<i>Datetime</i>	-	Tanggal pemesanan
10	Expedisi_id	<i>Int</i>	5	Id ekspedisi ( <i>Foreign key</i> )
11	Noresi	<i>Varchar</i>	20	No resi
12	Bayar	<i>Int</i>	7	Bayar
13	Kembalian	<i>Int</i>	7	Kembalian

#### 6. Tabel Detail Pemesanan

Tabel Detail Pemesanan dikelola oleh Admin untuk menyimpan data detail pemesanan. Nama tabel detail pemesanan dalam *database* adalah detail pemesanan. Seperti yang terlihat pada Tabel 4.21:

**Tabel 4.21 Rancangan Struktur Tabel Detail Pemesanan**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	5	Id detail ( <i>Primary Key</i> )
2	Pemesanan_id	<i>Int</i>	5	Id Pemesanan( <i>Foreign Key</i> )
3	Barang_id	<i>Int</i>	5	Id Barang ( <i>Foreign Key</i> )
4	Harga	<i>Int</i>	8	Harga Barang
5	Jumlah	<i>Int</i>	5	Jumlah Barang yang dibeli
6	Ukuran	<i>Varchar</i>	2	Ukuran

#### 7. Tabel Konfirmasi

Tabel Konfirmasi dikelola oleh Admin untuk menyimpan data konfirmasi. Nama tabel konfirmasi dalam *database* adalah konfirmasi. Seperti yang terlihat pada Tabel 4.22:

**Tabel 4.22 Rancangan Struktur Tabel Konfirmasi**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	5	Id konfirmasi ( <i>Primary Key</i> )
2	Pemesanan_id	<i>Int</i>	5	Id Pemesanan( <i>Foreign Key</i> )
3	Kodekonfirmaasi	<i>Int</i>	10	Kode konfirmasi
4	Jumlahbayar	<i>Int</i>	8	Jumlah bayar
5	Norekening	<i>Varchar</i>	30	Norekening
6	Melalui	<i>Varchar</i>	30	transfer melalui
7	Tgltransfer	<i>Date</i>	-	Tanggal transfer
8	Fotobukti	<i>Varchar</i>	50	Foto bukti transfer
9	Stskonfirmasi	<i>Enum</i>	'B','S'	Status konfirmasi

#### 8. Tabel Expedisi

Tabel Expedisi dikelola oleh Admin untuk menyimpan data Expedisi.

Nama tabel Expedisi dalam *database* adalah Expedisi. Seperti yang terlihat pada Tabel 4.23:

**Tabel 4.23 Rancangan Struktur Tabel Expedisi**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	4	Id expedisi ( <i>Primary key</i> )
2	Kode	<i>Varchar</i>	10	Kode expedisi
3	Namaexpedisi	<i>Varchar</i>	40	Nama expedisi
4	Biaya	<i>Int</i>	5	Biaya pengiriman

#### 9. Tabel Info Web

Tabel Info web dikelola oleh Admin untuk menyimpan data Informasi

Website. Nama tabel info web dalam *database* adalah info web. Seperti yang terlihat pada Tabel 4.24:

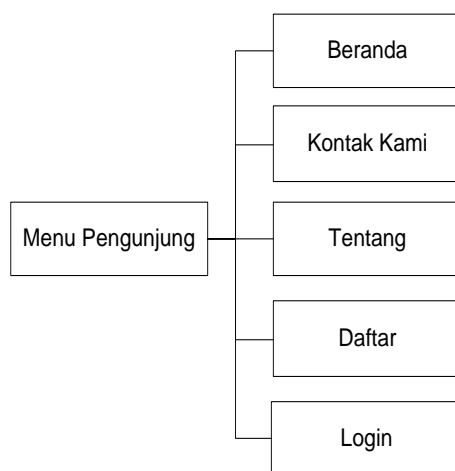
**Tabel 4.24 Rancangan Struktur Tabel Info Web**

No	Field Name	Type	Length	Keterangan
1	Id	<i>Int</i>	4	Id expedisi ( <i>Primary key</i> )
2	Nama	<i>Varchar</i>	30	Nama
3	Deskripsi	<i>Text</i>	-	Deskripsi

## 4.7 RANCANGAN STRUKTUR PROGRAM

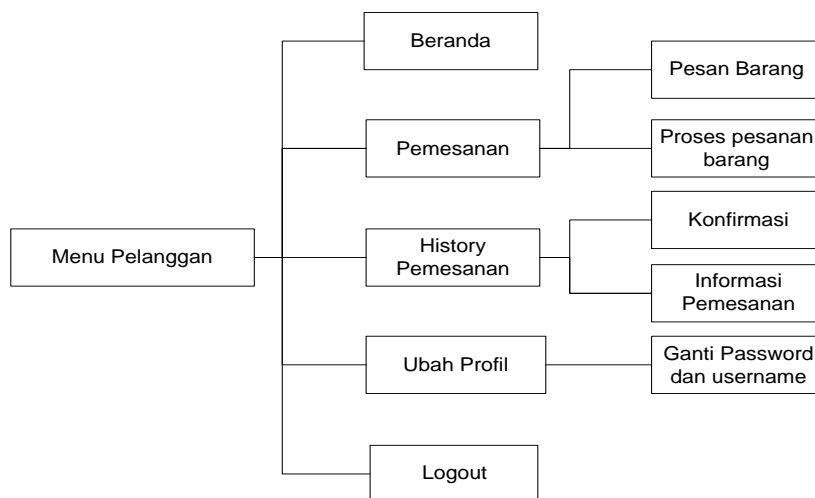
Rancangan struktur program merupakan bayangan dari sebuah program dimana nanti akan menjelaskan bagian besar struktur program yang akan dibangun oleh penulis, rancangannya terdapat pada gambar dibawah ini.

### 1. Rancangan struktur Program Pengunjung



**Gambar 4.47 Rancangan Struktur Program Pengunjung**

### 2. Rancangan Struktur Program Pelanggan



**Gambar 4.48 Rancangan Struktur Program Pelang**

