

## BAB III ANALISIS SISTEM

### 3.1 Unit Kegiatan Kerja Praktik

Kegiatan kerja praktik dilakukan pada unit IT (*Technology Information*) dan akan membuat sebuah perancangan aplikasi warehouse berbasis android dengan fitur scan Barcode dan fitur edit bagi admin gudang PT. Shima Prima Utama.

### 3.2 Prosedur pada Unit Kerja


Dalam menjalankan tugasnya, bagian IT telah memiliki prosedur kerja yang telah ditetapkan oleh PT. Shima Prima Utama. Prosedur kerja yang dilakukan adalah sebagai berikut :

1. Bagian IT melakukan pengecekan rutin terhadap infrastruktur TI (*Technology Information*) baik itu hardware & Software untuk menunjang operasional perusahaan.
2. Bagian IT bertanggung jawab atas inovasi dalam hal teknologi yang dapat digunakan untuk menunjang semua pekerjaan dalam perusahaan.

### 3.3 Kegiatan Kerja Praktik yang dilakukan

Kegiatan kerja yang dilakukan sesuai dengan prosedur yang berlaku di perusahaan tersebut. Untuk rincian mengenai kegiatan yang dilakukan di PT. Shima Prima Utama dapat dilihat pada tabel berikut.

**Tabel 3.1 Tabel Kegiatan Kerja Praktik**

No	Tanggal	Kegiatan	Dokumentasi
1.	17 Mei 2023	Perkenalan perusahaan dan wawancara kebutuhan perusahaan.	

Tabel 3.1 Tabel Kegiatan Kerja Praktik (Lanjutan)

No	Tanggal	Kegiatan	Dokumentasi
2.	26 Juni s.d 27 Juni 2023	Mempelajari <i>Codeigniter 3</i> dan SQL	
3.	1 Juli dan 3 Juli 2023	Desain <i>Front End</i>	
4.	4 Juli 2023	Meminta informasi sejarah, visi dan misi perusahaan PT.ShimaPrima Utama	
5.	5 Juli 2023	Diskusi kebutuhan perusahaan	
6.	6 Juli s.d 28 Juli 2023	Desain UI Project	

Tabel 3.1 Tabel Kegiatan Kerja Praktik (Lanjutan)

No	Tanggal	Kegiatan	Dokumentasi
7.	11 Juli s.d 12 Juli 2023	Desain <i>Front End</i>	
8.	14 Juli s.d 21 Juli 2023	Membuat <i>Back End</i>	
9.	22 Juli 2023	Diskusi Desain Project	
10.	27 Juli 2023	Diskusi Progres KP	
11.	28 Juli s.d 1 Agustus 2023	Hosting Website Shima	
12.	2 Agustus s.d 4 Agustus 2023	Penyerahan surat penilaian + selesai kp	

### **3.4 Metodologi Pengumpulan Data dan Pengembangan Sistem**

#### **3.4.1 Observasi**

Pada tahap observasi ini penulis mengamati dan memahami sistem warehouse yang sudah ada bagaimana proses barang masuk dan barang keluar. Observasi ini penulis lakukan kurang lebih 1 Minggu di PT Shima Prima Utama selama melakukan kerja praktik di perusahaan tersebut.

#### **3.4.2 Wawancara**

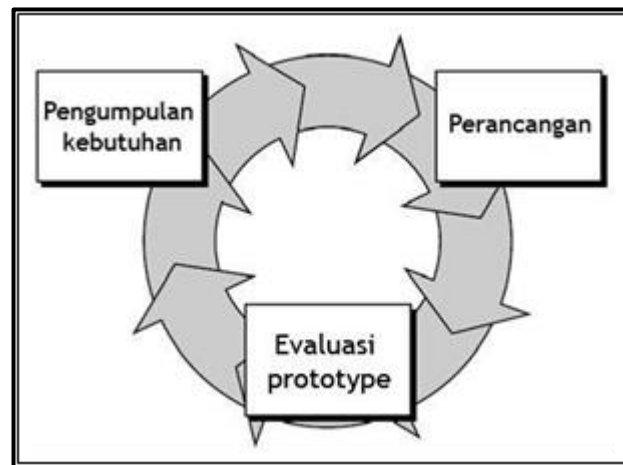
Pada metode wawancara ini penulis melakukan wawancara kepada bagian IT (*Technology Information*) pihak yang mengetahui dan memahami proses sistem yang sedang berjalan di PT Shima Prima Utama. Penulis juga melakukan wawancara untuk mengetahui apa saja yang dibutuhkan dalam proses perancangan sistem pada nantinya.

#### **3.4.3 Studi Pustaka**

Pengumpulan data dilakukan dengan cara mengumpulkan dan mempelajari dari buku, skripsi, jurnal, dan artikel dari internet yang berhubungan dengan masalah yang dibahas.

#### **3.4.4 Metodologi Pengembangan Sistem**

Metodologi pengembangan sistem yang digunakan dalam perancangan website ini adalah Model Prototype. Prototyping Model adalah metode yang digunakan untuk mendefinisikan serangkaian sasaran umum bagi perangkat lunak serta mengidentifikasi kebutuhan input, pemrosesan, ataupun output detail (Pressman, 2010). Tahap model ini digambar pada gambar dibawah ini.:



**Gambar 3. 2 Model Prototype**

1. Pengumpulan Kebutuhan

Pada tahap ini dilakukan konsultasi dan wawancara dengan pihak PT.Shima Prima Utama di bidang IT yang bertujuan untuk mengetahui alur kerja dan sistem yang sudah ada pada perusahaan. Dari sistem tersebut maka akan digunakan dalam perancangan fitur yang akandirancang.

2. Perancangan

Pada tahap ini merancang fitur scan Barcode dan fitur *edit* yang sesuai dengan permintaan yang didapat dari konsultasi dan wawancara. Perancangan yang di buat dengan metode *Data Flow Diagram (DFD)*, *Entity Relationship Diagram (ERD)*, dan *flowchart* yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika.

3. Evaluasi Prototype (Evaluasi Perancangan)

Pada tahap ini akan dilakukan konsultasi kembali dengan bagian IT mengenai rancangan program yang telah dibuat. Evaluasi dilakukan untuk memeriksa apakah rancangan tersebut telah sesuai dengan keinginan perusahaan.

### 3.5 Analisis Sistem

Untuk membangun aplikasi yang tepat guna, langkah-langkah

menganalisis sistem yang sedang berjalan perlu dilakukan dengan menggunakan metode riset lapangan, observasi, wawancara langsung kepada bagian IT di PT. Shima Prima Utama.

### **3.5.1 Prosedur Sistem Yang Sedang Berjalan**

Prosedur dalam website warehouse pada bagian website masih belum memiliki fitur edit yang memudahkan pelanggan dalam mencari produk yang diinginkan. Sehingga karyawan hanya dapat menginput data. Kemudian karyawan dan admin memiliki hak akses yang sama dalam mengolah seluruh data yang ada.

### **3.5.2 Identifikasi Permasalahan**

Identifikasi masalah bertujuan mengidentifikasi permasalahan yang ada pada prosedur kerja supaya terciptanya prosedur baru dengan harapan dapat diterapkan dengan baik. Setelah dilakukan observasi dan menurut wawancara dengan PT.Shima Prima Utama, dibutuhkan perancangan aplikasi warehouse PT.Shima.

### **3.5.3 Kebutuhan Sistem Yang Diusulkan**

Analisis sistem terdapat 2 kebutuhan yaitu kebutuhan fungsional dan kebutuhan non – fungsional. Kebutuhan fungsional berisi fitur-fitur yang dibutuhkan sistem. Kebutuhan non – fungsional adalah kebutuhan sistem yang memiliki batasan.

#### **3.5.3.1 Kebutuhan Fungsional**

Kebutuhan fungsional adalah kebutuhan yang berisi fitur–fitur yang harus ada pada sistem. Kebutuhan fungsional pada PT. Shima Prima Utama, meliputi:

1. Sistem dapat memasukkan data supplier, barang dan transaksi sesuai hak akses

2. Sistem dapat menampilkan data supplier, barang dan transaksi
3. Sistem dapat menghapus data supplier, barang dan transaksi sesuai hak akses
4. Sistem dapat memasukan data Barcode

### **3.5.3.2 Kebutuhan Non-Fungsional**

Kebutuhan non – fungsional adalah kebutuhan sistem yang memiliki batasan. Beberapa kebutuhan non – fungsional pada perancangan Aplikasi warehouse pada PT. Shima Prima Utama adalah sebagai berikut:

#### **1. Keamanan**

Sistem bisa digunakan oleh karyawan dan admin yang memiliki email dan password sehingga perubahan data supplier, barang, transaksi, dan akun user (karyawan atau admin) dapat dilakukan tracing data.

#### **2. Informasi**

Informasi yang ditujukan untuk karyawan dan admin yaitu agar memudahkan tracing data atas terjadinya perubahan data.

### **3.5.4 Prosedur Sistem Yang Diusulkan**

Setiap karyawan yang memiliki hak akses dapat menginput, mengubah, melihat informasi dan menghapus data supplier, barang dan transaksi. Akan tetapi, untuk karyawan hanya dapat melihat informasi dan mengubah data akun atau profil dari karyawan tersebut. Sedangkan bagian administrator mendapat akses tambahan yaitu menginput, mengubah, melihat dan menghapus data akun(admin dan karyawan).

## **3.6 Perancangan Sistem Yang Diusulkan**

Perancangan Aplikasi Warehouse ini terdiri dari pemodelan proses sistem yang dilakukan dengan *Data Flow Diagram* (DFD), pemodelan data

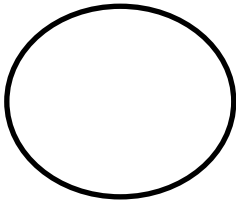


sistem yang dilakukan dengan *Entity Relationship Diagram* (ERD) dan spesifikasi tabel-tabel database dan pemodelan sistem yang dilakukan dengan Flowchart. Kemudian perancangan user interface pada website.

### 3.6.1 Pemodelan Proses Sistem

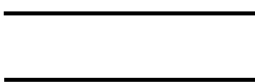
*Data Flow Diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari input/masukan dan output/keluaran. DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi dan dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail (Akik Hidayat 2021) (A.S., Rosa dan Salahuddin).

Menurut Edward Yourdon dan Tom DeMarco, notasi – notasi pada DFD dapat dilihat pada tabel berikut ini:

**Tabel 3.2 Notasi – Notasi DFD**  
(Sumber: (A.S., Rosa dan Salahuddin))

Notasi	Keterangan
	Proses atau fungsi atau prosedur pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program
	Entitas luar atau masukan atau keluaran atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan
	Aliran data merupakan data yang dikirim antar proses dari penyimpanan ke proses,

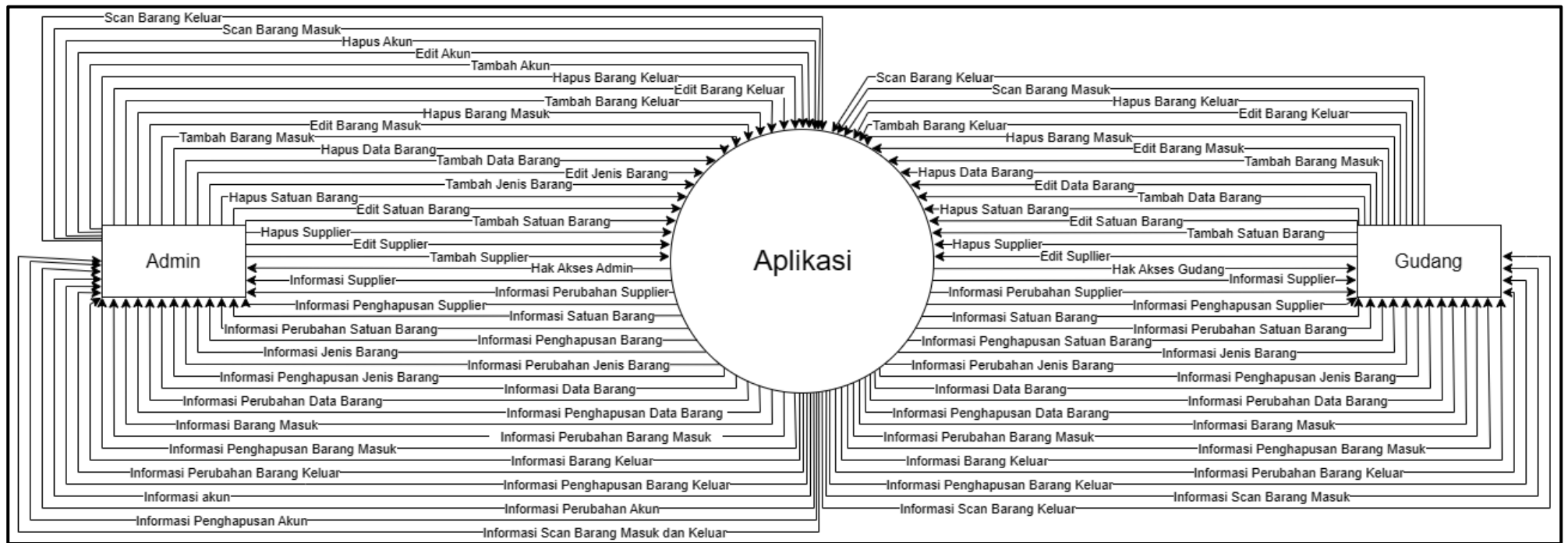


←	atau dari proses ke masukan atau keluaran
	Penyimpanan berisi tabel – table basis data yang dibutuhkan sesuai dengan <i>Entity Relationship Diagram</i> .

Berikut ini adalah tahapan – tahapan perancangan menggunakan *Data Flow Diagram* (DFD) (Akik Hidayat 2021):

### 3.6.1.1 DFD Level 0 (Diagram Konteks) Yang Diusulkan

Diagram Konteks menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. Diagram konteks digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar. Berikut ini adalah DFD Level 0 dari sistem yang diusulkan.

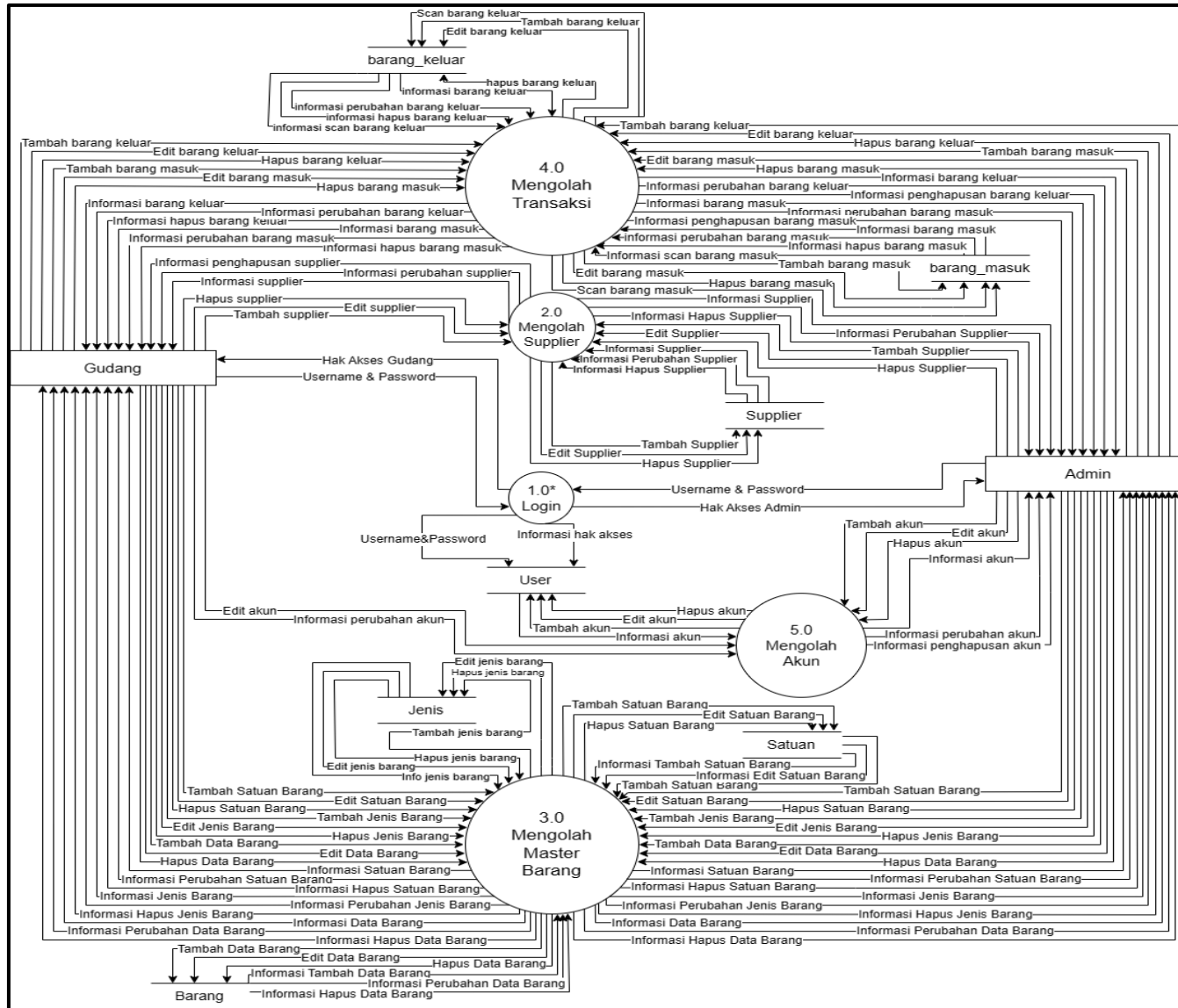


**Gambar 3. 3 DFD Level 0 Yang Diusulkan**

### 3.6.1.2 DFD Level 0 (Diagram Konteks) Yang Diusulkan

DFD Level 1 digunakan untuk menggambarkan modul– modul yang ada dalam sistem yang akan dikembangkan.

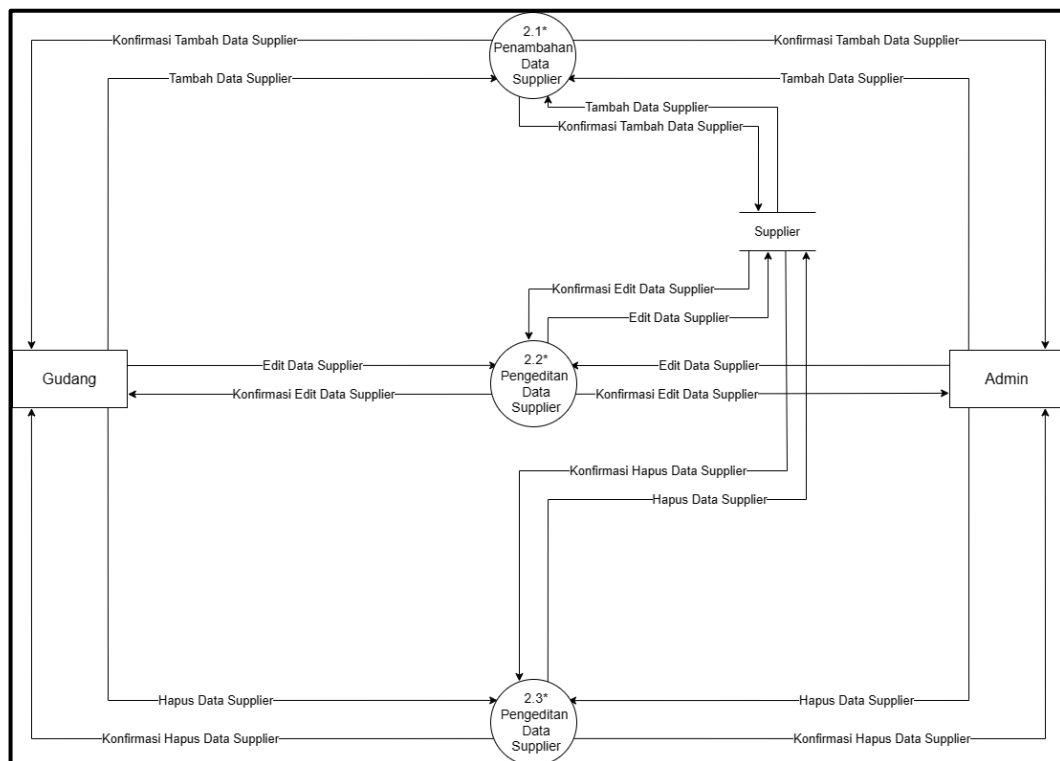
DFD Level 1 merupakan hasil breakdown DFD Level. Berikut ini adalah DFD Level 1 dari sistem yang diusulkan



Gambar 3. 4 DFD Level 1 yang diusulkan

### 3.6.1.3 DFD Level 2 Mengolah Supplier

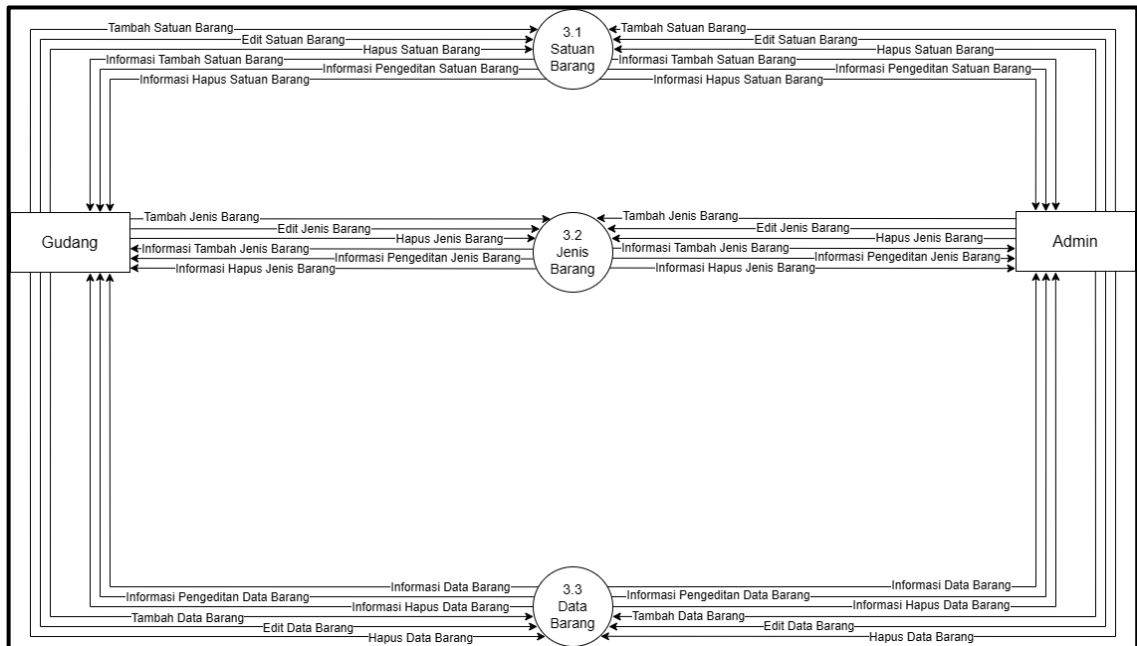
DFD Level 2 adalah hasil breakdown DFD Level 1 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 2 dalam proses mengolah supplier dari sistem yang diusulkan:



**Gambar 3. 5 DFD Level 2 Mengolah Supplier**

### 3.6.1.4 DFD Level 2 Mengolah Master Barang Yang Diusulkan

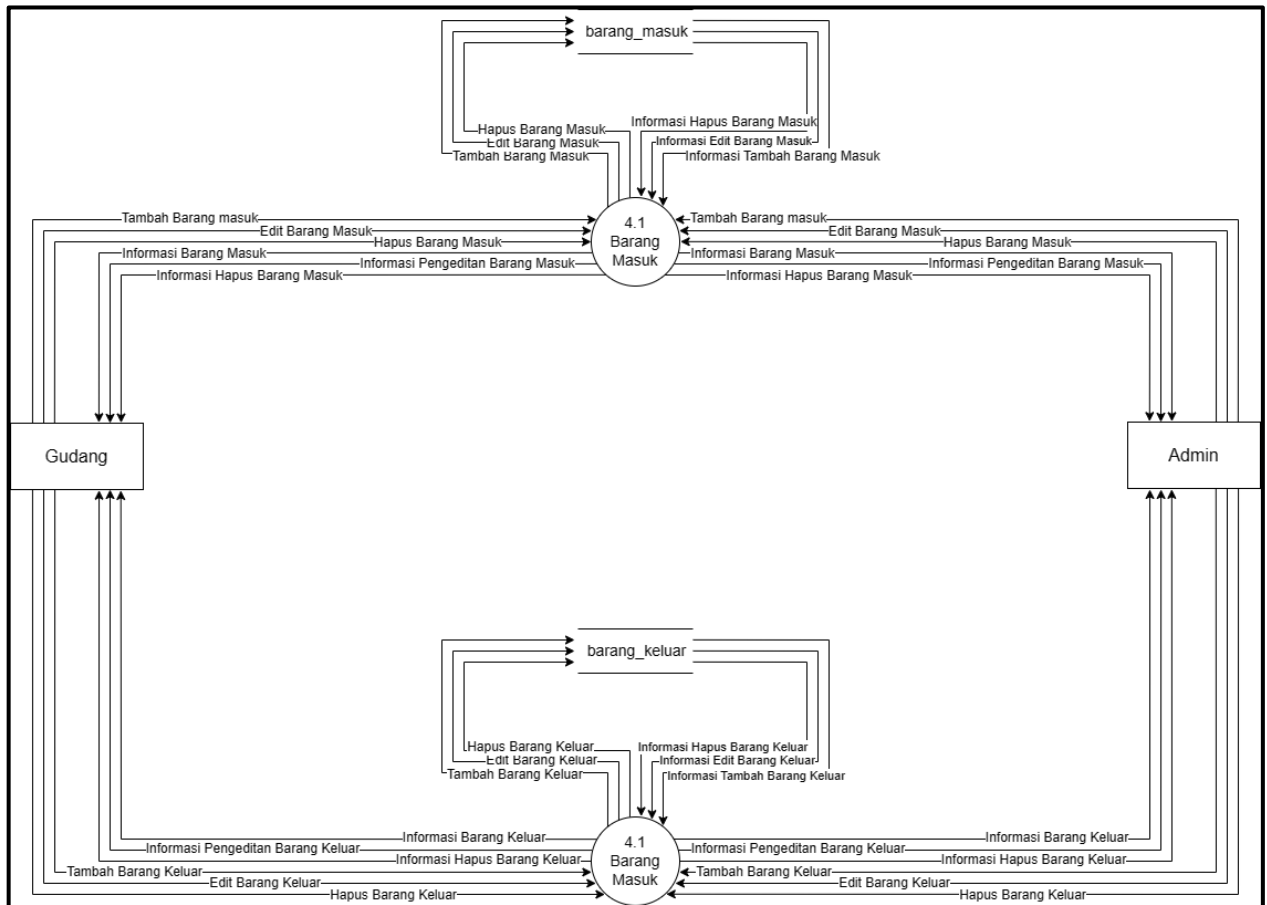
DFD Level 2 adalah hasil breakdown DFD Level 1 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut . Berikut ini adalah DFD Level 2 dalam proses mengolah master barang dari sistem yang diusulkan:



**Gambar 3. 6 DFD Level 2 Mengolah Master Barang Yang Diusulkan**

### 3.6.1.5 DFD Level 2 Mengolah Transaksi Yang Diusulkan

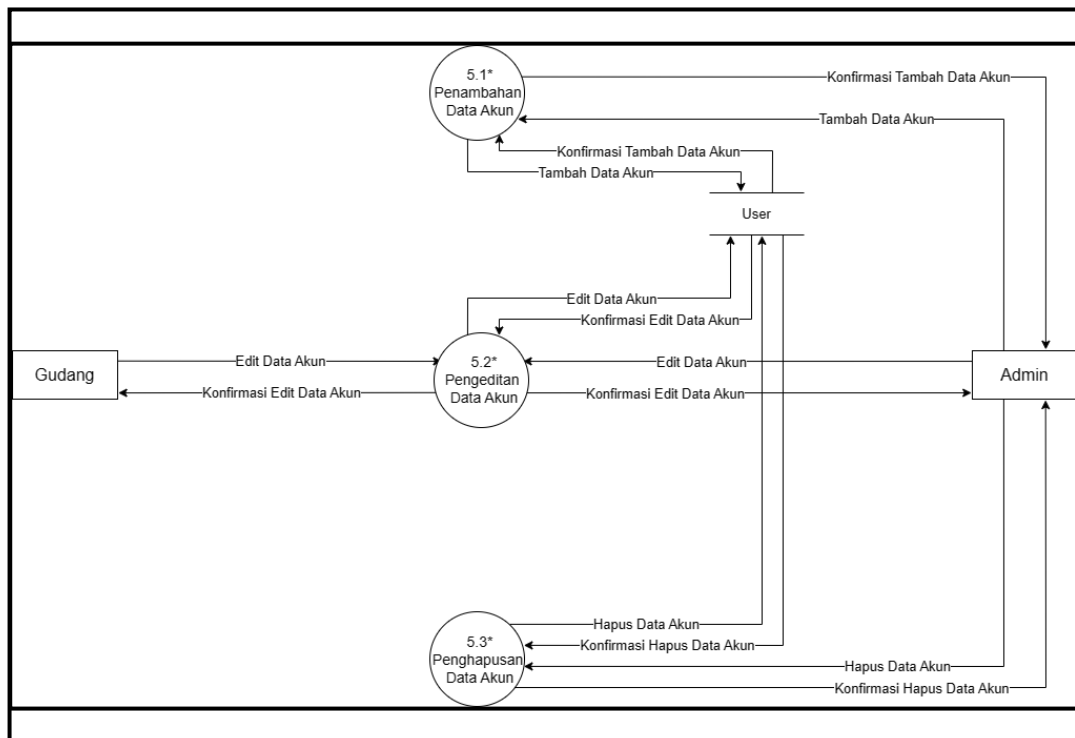
DFD Level 2 adalah hasil breakdown DFD Level 1 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 2 dalam proses mengolah transaksi dari sistem yang diusulkan:



**Gambar 3. 7 DFD Level 2 Mengolah Transaksi**

### 3.6.1.6 DFD Level 2 Mengolah Akun Yang Diusulkan

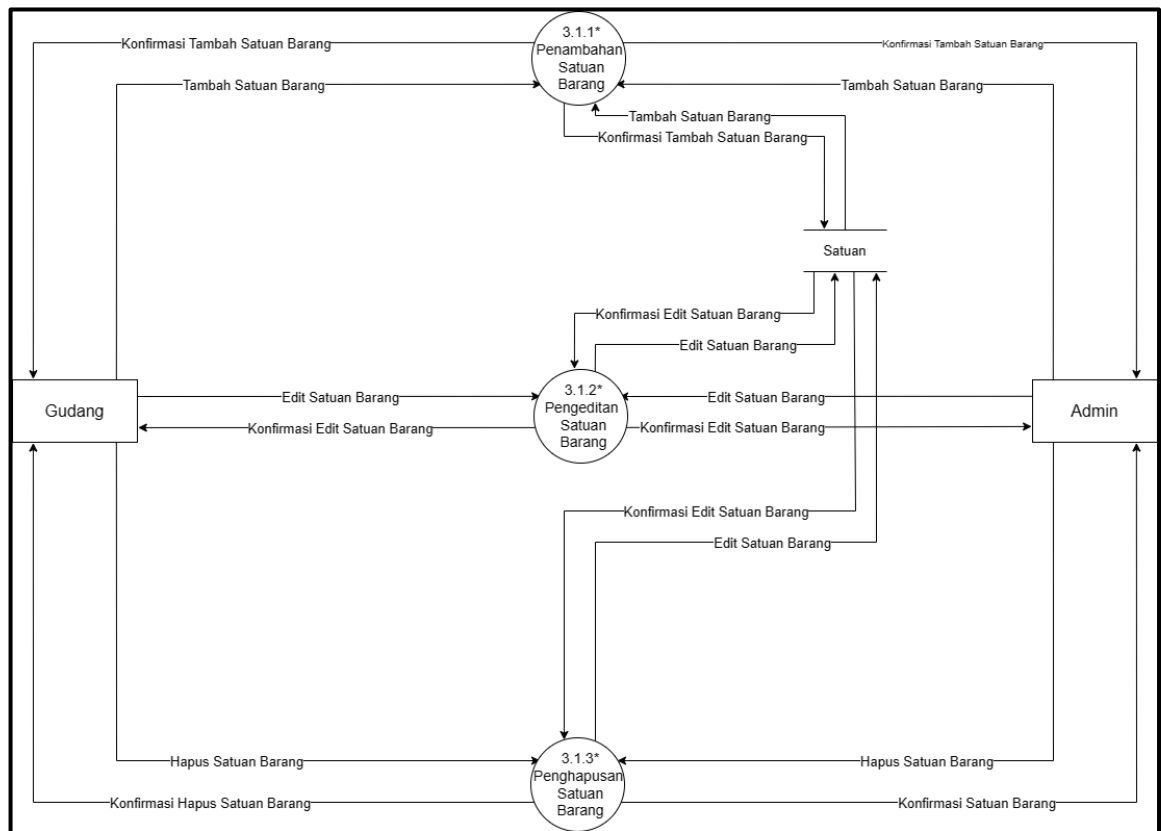
DFD Level 2 adalah hasil breakdown DFD Level 1 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 2 dalam proses mengolah akun dari sistem yang diusulkan:



**Gambar 3. 8 DFD Level 2 Mengolah Akun**

### 3.6.1.7 DFD Level 3 Mengolah Satuan Barang Yang Diusulkan

DFD Level 3 adalah hasil breakdown DFD Level 2 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 3 dalam proses mengolah Satuan Barang dari sistem yang diusulkan:

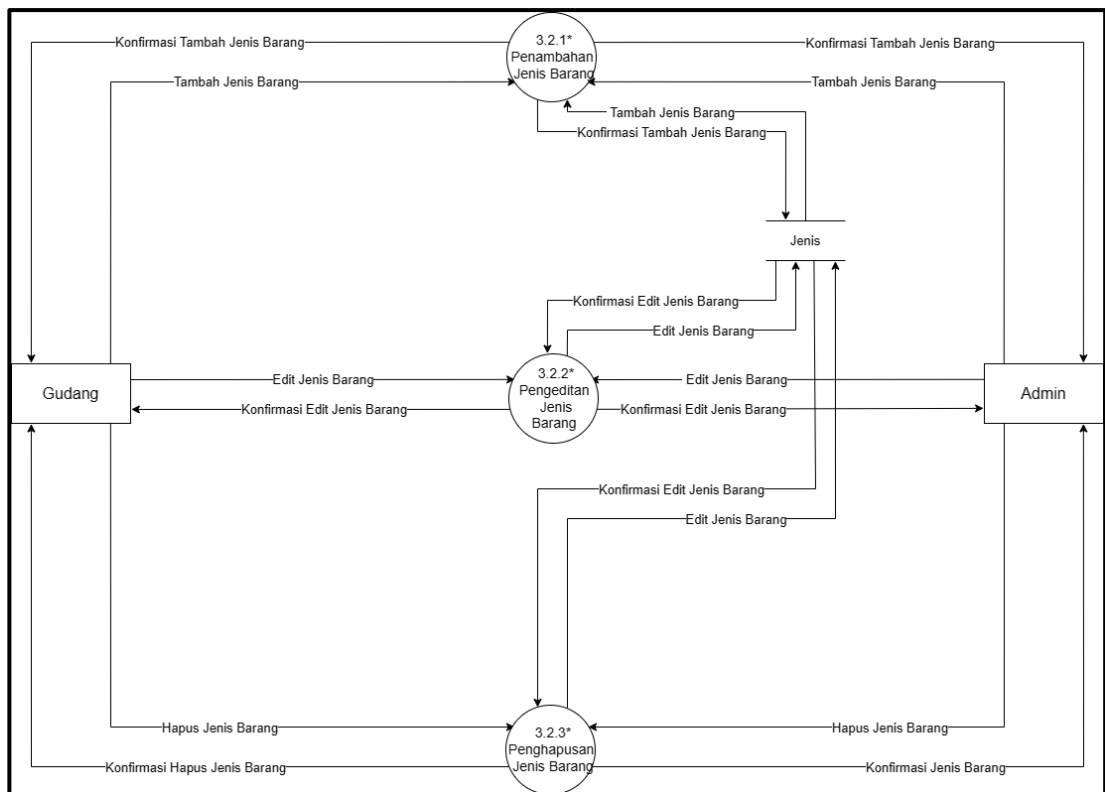


**Gambar 3. 9 DFD Level 3 Mengolah Satuan Barang**

### **3.6.1.8 DFD Level 3 Mengolah Jenis Barang Yang Diusulkan**

DFD Level 3 adalah hasil breakdown DFD Level 2 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 3 dalam proses mengolah Jenis Barang dari sistem yang diusulkan:

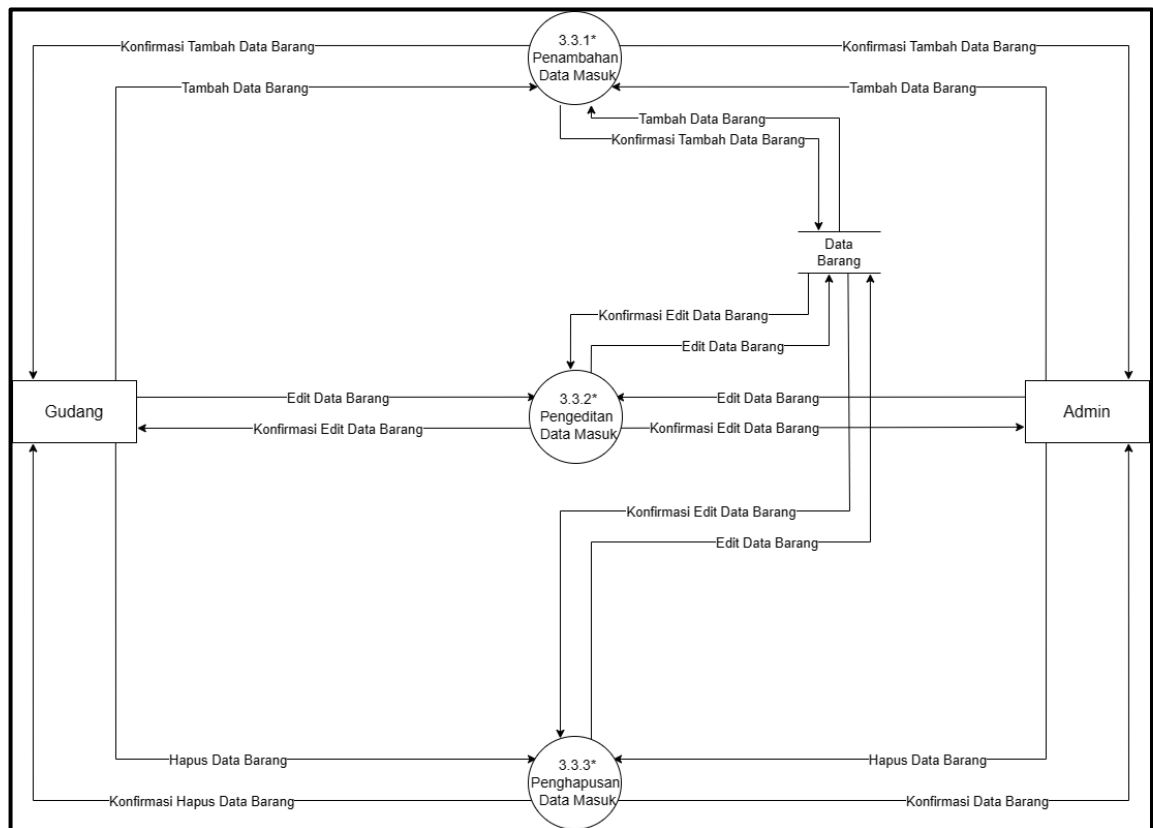




**Gambar 3. 10 DFD Level 3 Mengolah Jenis Barang**

### 3.6.1.9 DFD Level 3 Mengolah Data Barang Yang Diusulkan

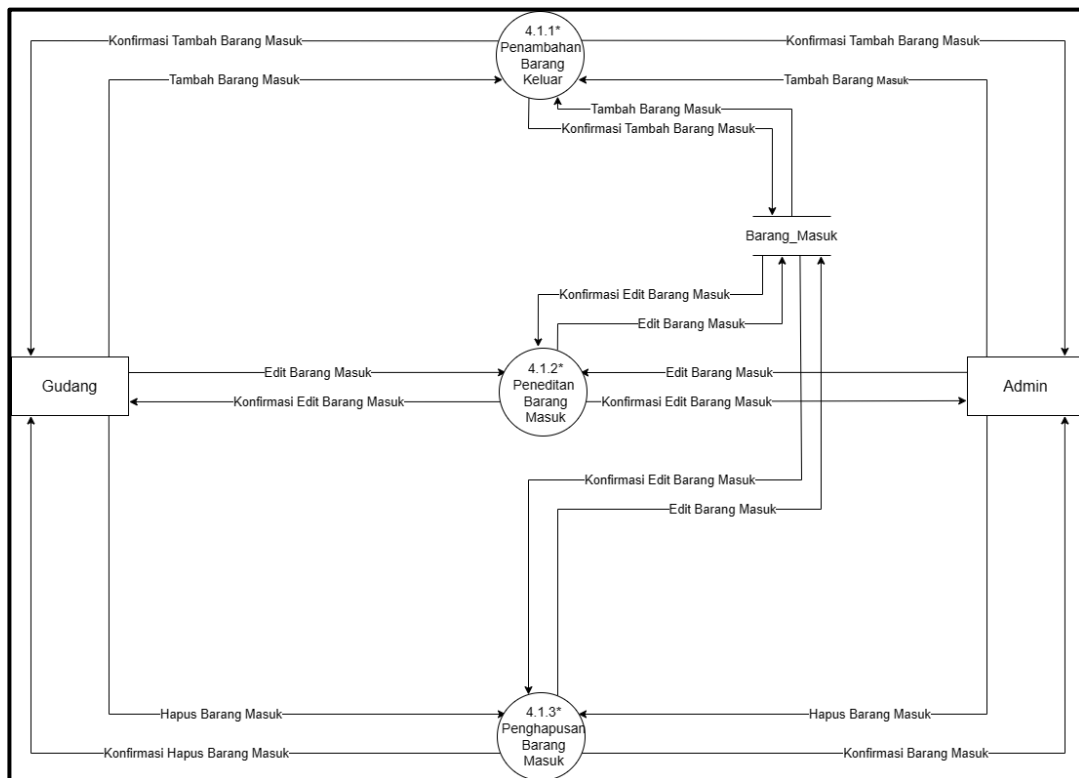
DFD Level 3 adalah hasil breakdown DFD Level 2 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul. Berikut ini adalah DFD Level 3 dalam proses mengolah Data Barang dari sistem yang diusulkan:



**Gambar 3. 11 DFD Level 3 Mengolah Data Barang**

### **3.6.1.10 DFD Level 3 Mengolah Transaksi Barang Masuk Yang Diusulkan**

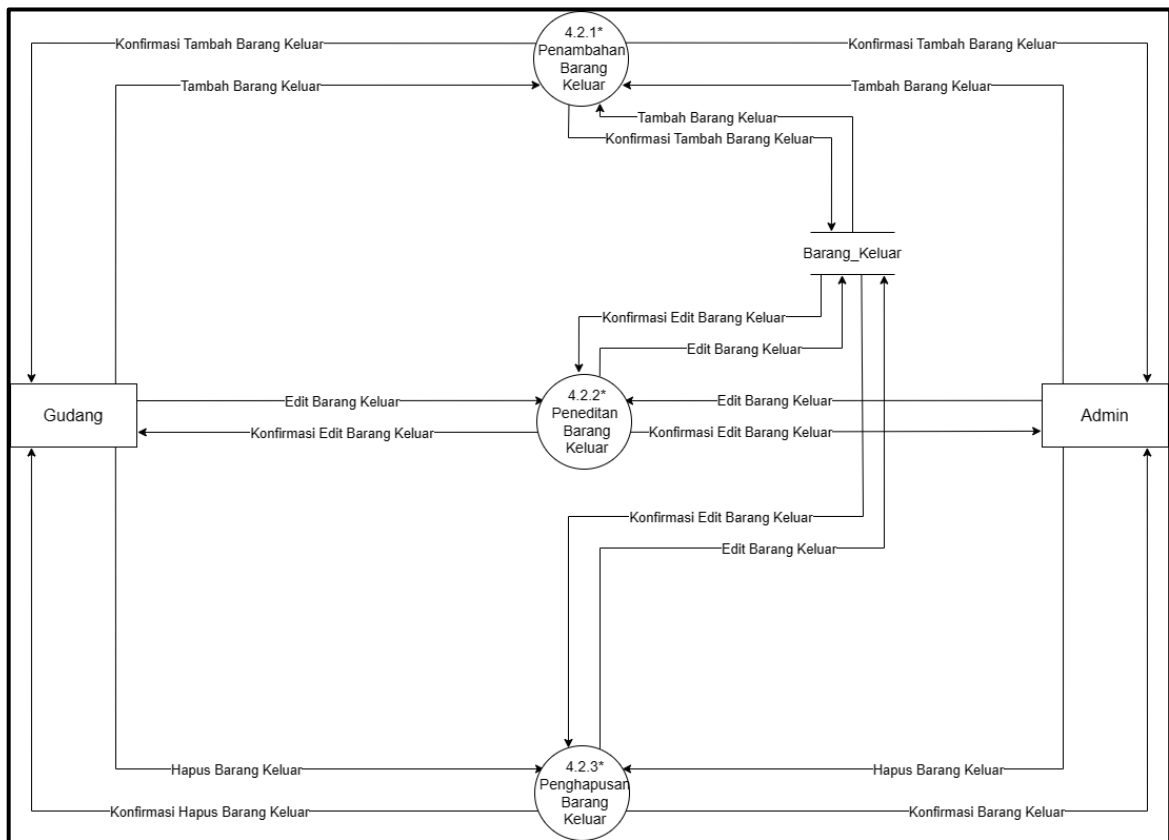
DFD Level 3 adalah hasil breakdown DFD Level 2 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul. Berikut ini adalah DFD Level 3 dalam proses mengolah Transaksi Barang Masuk dari sistem yang diusulkan:



**Gambar 3. 12 DFD Level 3 Mengolah Transaksi Barang Masuk**

### 3.6.1.11 DFD Level 3 Mengolah Transaksi Barang Keluar Yang Diusulkan

DFD Level 3 adalah hasil breakdown DFD Level 2 pada modul - modul yang harus di breakdown lebih detail tergantung tingkat kedetailan modul tersebut. Berikut ini adalah DFD Level 3 dalam proses mengolah Transaksi Barang Keluar dari sistem yang diusulkan:



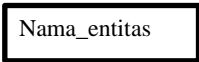
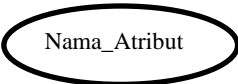
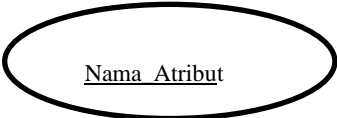

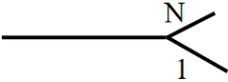
**Gambar 3. 13 DFD Level 3 Mengolah Transaksi Barang Keluar**

### 3.6.2 Data Pemodelan Data Sistem

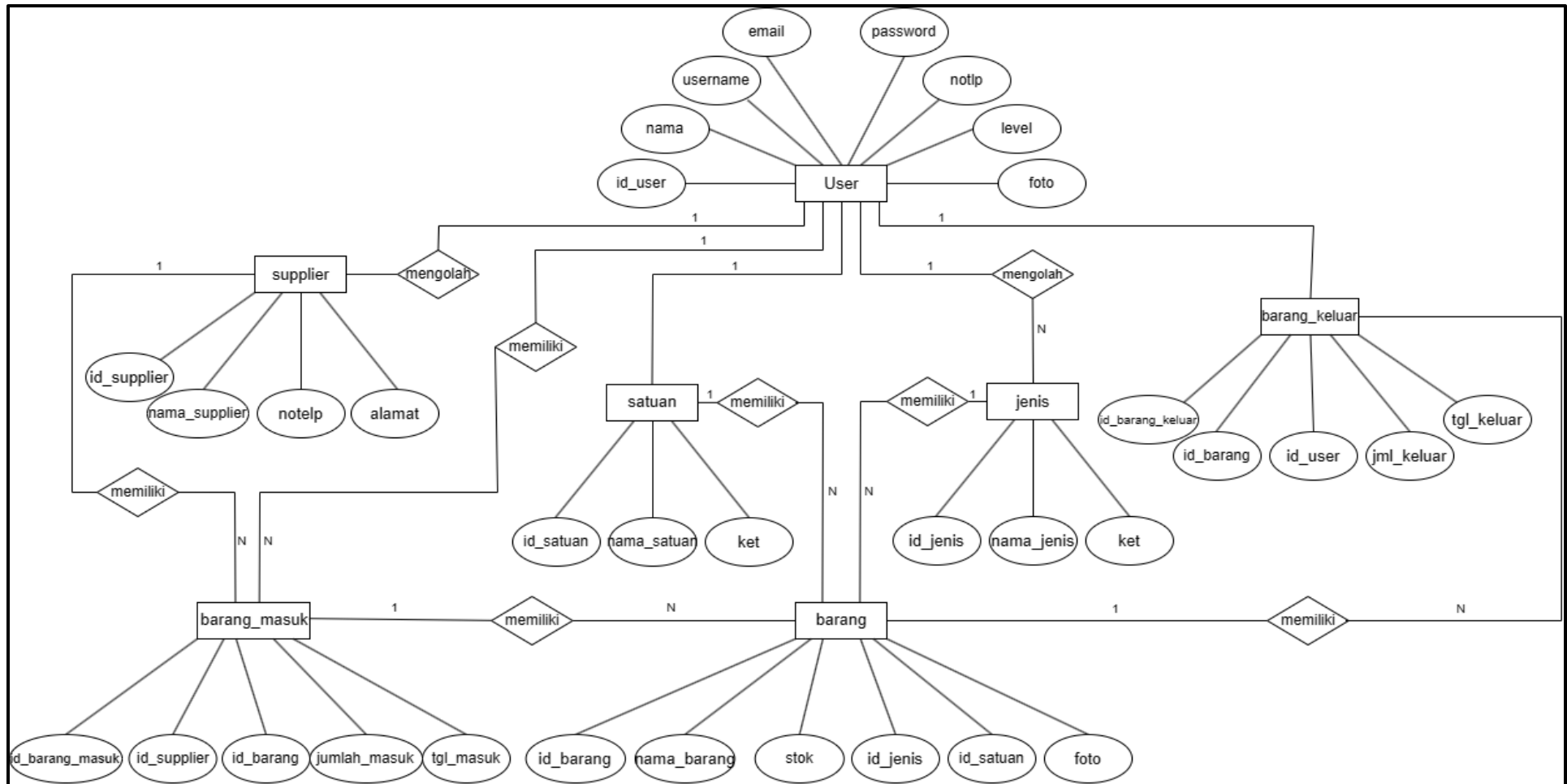
Pemodelan data sistem menggunakan *Entity Relationship Diagram* (ERD). ERD digunakan untuk pemodelan basis data relasional dan dikembangkan berdasarkan teori himpunan dalam bidang matematika. Notasi yang paling banyak digunakan adalah notasi dari Chen. Berikut ini adalah notasi dari Chen:

**Tabel 3.3 Simbol ERD Notasi Chen**

(Sumber: (A.S., Rosa dan Shalahuddin, 2016))

Simbol	Keterangan
	<b>Entitas</b> merupakan data inti yang akan disimpan, bakal tabel pada basis data.
	<b>Atribut</b> adalah kolom data yang butuh disimpan dalam suatu entitas.
	<b>Atribut kunci primer</b> adalah kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.
	<b>Relasi</b> yang menghubungkan antar entitas dan biasanya diawali dengan kata kerja.
	<b>Asosiasi</b> adalah penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas.

Berikut ini adalah pemodelan data sistem yang diusulkan:



Gambar 3. 14 ERD

Database yang digunakan adalah sebagai berikut :

a. Table *User*

Tabel user merupakan penyimpanan data – data akun user termasuk email dan password untuk akun karyawan dan admin. Berikut ini adalah atribut – atribut tabel user:

**Tabel 3.4 Spesifikasi Tabel Data User**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
<i>id_user</i>	<i>Int</i>	11	<i>Primary Key</i>
<i>nama</i>	<i>varchar</i>	50	
<i>Username</i>	<i>varchar</i>	50	
<i>email</i>	<i>varchar</i>	50	
<i>password</i>	<i>varchar</i>	50	
<i>notelp</i>	<i>varchar</i>	15	
<i>level</i>	<i>Enum</i>	Gudang, Admin	
<i>foto</i>	<i>varchar</i>	256	

b. Table *Supplier*

Tabel supplier merupakan tabel basis data yang digunakan untuk menyimpan data-data supplier. Berikut ini adalah atribut-atribut tabel supplier.

**Tabel 3.5 Spesifikasi Tabel Data Supplier**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
<i>id_supplier</i>	<i>Int</i>	11	<i>Primary Key</i>
<i>nama_supplier</i>	<i>varchar</i>	50	
<i>notelp</i>	<i>varchar</i>	50	
<i>alamat</i>	<i>text</i>		

c. Table *Satuan*

Tabel satuan merupakan tabel basis data yang digunakan untuk menyimpan data-data satuan barang. Berikut ini adalah atribut- atribut tabel satuan.

**Tabel 3.6 Spesifikasi Tabel Data satuan**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
<i>id_satuan</i>	<i>Int</i>	11	<i>Primary Key</i>
<i>nama_satuan</i>	<i>varchar</i>	50	
<i>ket</i>	<i>text</i>		

## d. Table jenis

Tabel jenis merupakan tabel basis data yang digunakan untuk menyimpan data-data jenis barang. Berikut ini adalah atribut-atribut tabel jenis.

**Tabel 3.7 Spesifikasi Tabel Data Jenis**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
id_jenis	<i>Int</i>	11	<i>Primary Key</i>
nama_jenis	<i>varchar</i>	50	
ket	<i>text</i>		

## e. Table Barang

Tabel barang merupakan tabel basis data yang digunakan untuk menyimpan data-data barang. Berikut ini adalah atribut-atribut tabel barang .

**Tabel 3.8 Spesifikasi Tabel Barang**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
id_barang	<i>Int</i>	11	<i>Primary Key</i>
Nama_barang	<i>varchar</i>	11	
stok	<i>varchar</i>	11	
id_satuan	<i>varchar</i>	11	<i>Forgein key</i>
id_jenis	<i>varchar</i>	11	<i>Forgein key</i>
foto	<i>varchar</i>	256	

## f. Table Barang Masuk

Tabel barang masuk merupakan tabel basis data yang digunakan untuk menyimpan data-data barang masuk. Berikut ini adalah atribut-atribut tabel barang masuk.

**Tabel 3.9 Spesifikasi Tabel Data Barang Masuk**

<i>Field</i>	<i>Type</i>	<i>Size</i>	<b>Keterangan</b>
id_barang_masuk	<i>Int</i>	11	<i>Primary Key</i>
id_supplier	<i>varchar</i>	11	<i>Forgein key</i>
id_barang	<i>varchar</i>	11	<i>Forgein key</i>
id_user	<i>varchar</i>	11	<i>Forgein key</i>
jumlah_masuk	<i>int</i>	10	
tgl_masuk	<i>varchar</i>	15	



g. Table Barang Keluar

Tabel barang keluar merupakan tabel basis data yang digunakan untuk menyimpan data-data barang keluar. Berikut ini adalah atribut-atribut tabel barang keluar.

**Table 3.10 Spesifikasi Tabel Data Jenis**

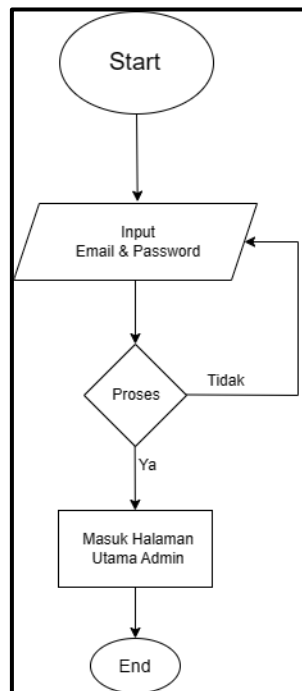
Field	Type	Size	Keterangan
id_barang_keluar	Int	11	Primary Key
id_barang	varchar	11	Forgein key
id_user	varchar	11	Forgein key
jumlah_keluar	varchar	10	
tgl_keluar	varchar	15	

### 3.6.3 Pemodelan Proses

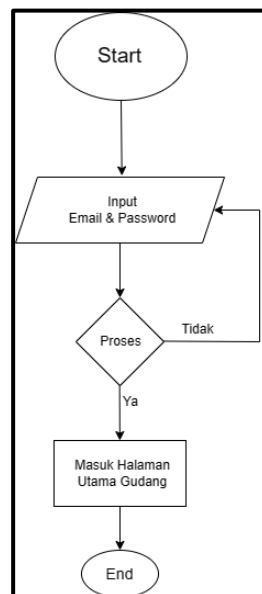
Pemodelan proses dilakukan dengan menggunakan *flowchart*. *Flowchart* adalah suatu bagian yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir (Indahyanti & Rahmawati, 2020). Berikut ini adalah simbol – simbol flowchart menurut ANSI (*American National Standard Institute*):

#### 3.6.3.1 Flowchart Login

Pada saat aplikasi dijalankan aplikasi akan meminta user memasukkan email dan password. Apabila email dan password yang dimasukkan benar maka aplikasi akan mengkonfirmasi apakah user sebagai admin atau sebagai karyawan dan jika sudah terkonfirmasi user akan diarahkan ke halaman yang sesuai. Apabila email dan password yang dimasukkan salah maka akan kembali pada halaman login dan diberikan sebuah alert bahwa email atau password tidak sesuai. Setelah itu proses selesai. Flowchart dapat dilihat pada Gambar berikut.



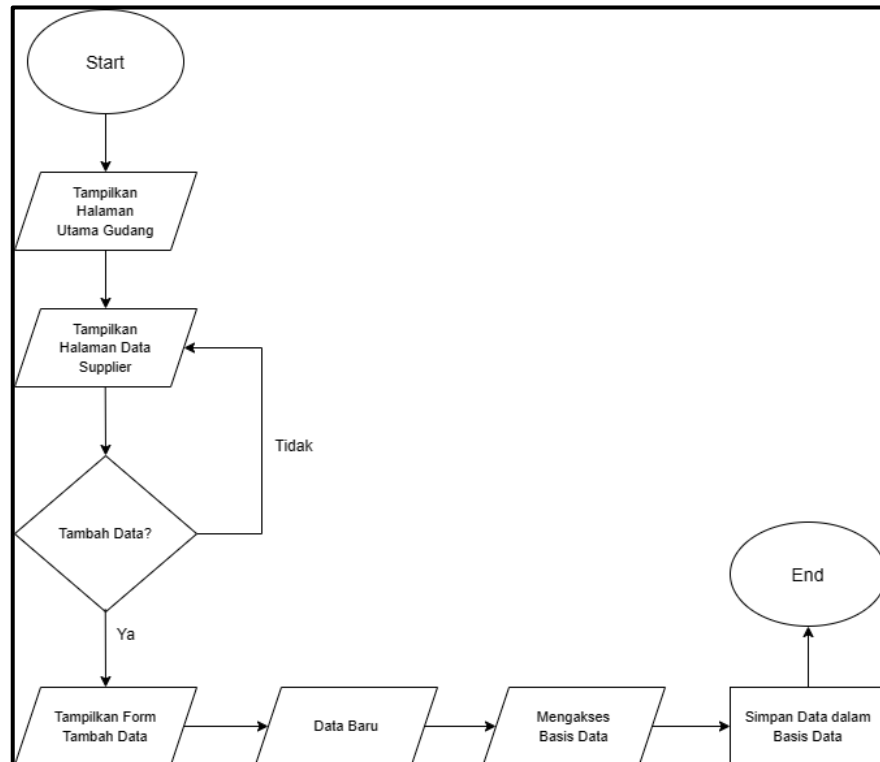
**Gambar 3. 15** *Flowchart Login Admin*



**Gambar 3. 16** *Flowchart Login Gudang*

### 3.6.3.2 Flowchart Halaman Menu Supplier Tambah Data

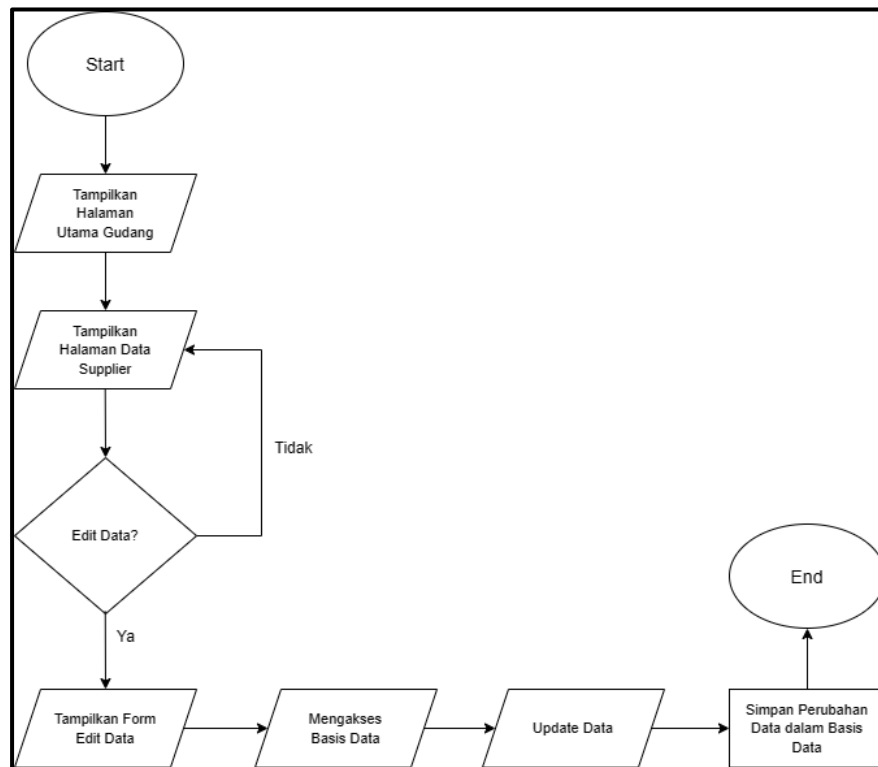
Pada halaman Menu Supplier, Karyawan dapat menambah data supplier. Apabila karyawan memilih Batal maka karyawan tersebut akan diarahkan menuju halaman Menu Supplier kembali. Berikut adalah flowchart pada halaman Menu Supplier Tambah Data:



**Gambar 3. 17 Flowchart Halaman Menu Supplier Tambah Data**

### 3.6.3.3 Flowchart Halaman Menu Supplier Edit Data

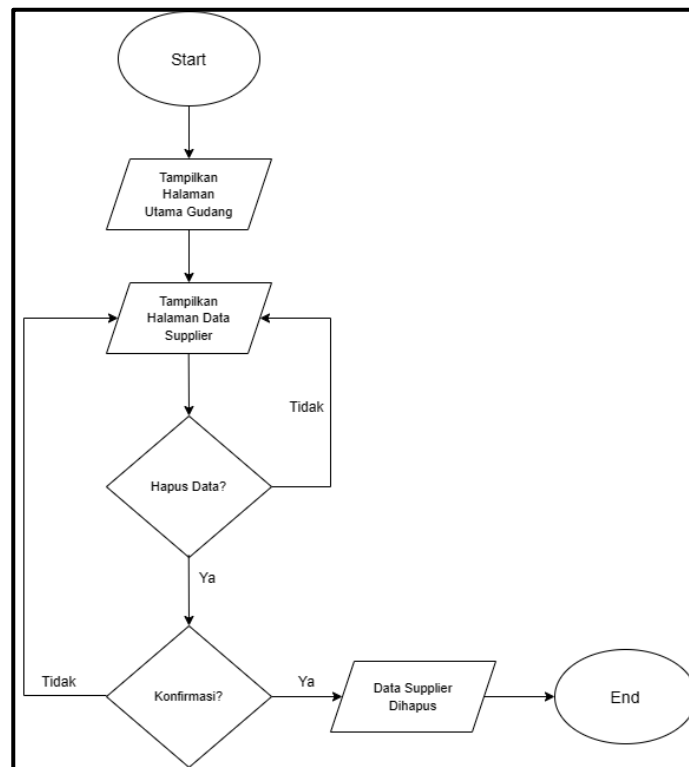
Pada menu supplier, admin dan karyawan bagian gudang dapat mengubah data supplier. Jika data supplier terdapat perubahan, maka data akan disimpan dalam database pada tabel supplier.



**Gambar 3. 18** *Flowchart* Halaman Menu Supplier Edit Data

#### 3.6.3.4 Flowchart Halaman Menu Supplier Hapus Data

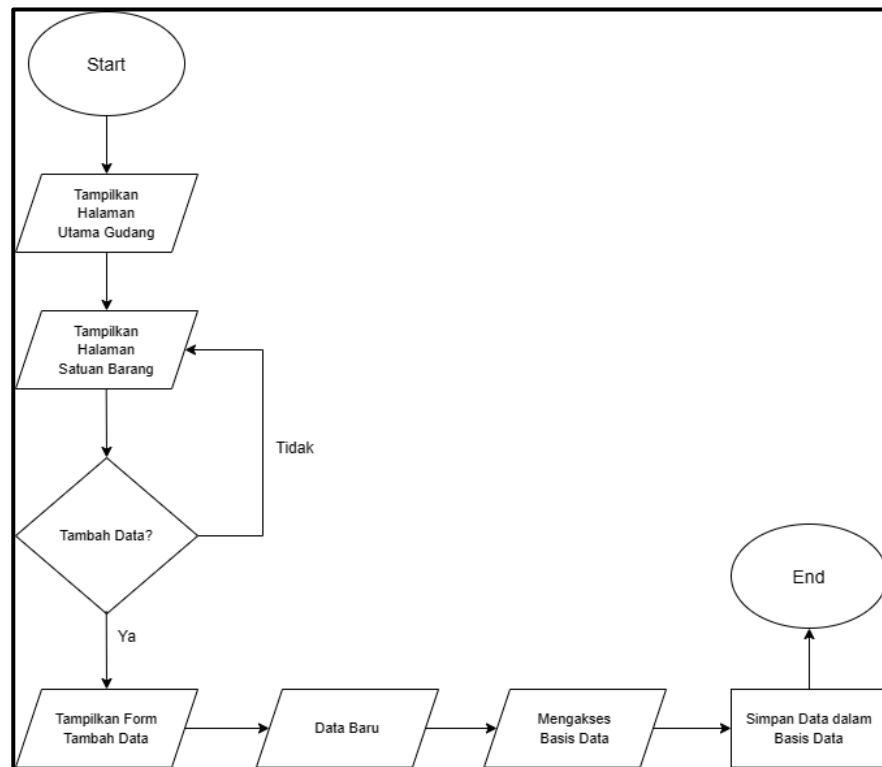
Pada menu supplier, admin dan karyawan bagian gudang dapat menghapus data supplier. Jika data supplier telah dihapus, maka data akan dihapus dalam database pada tabel supplier.



**Gambar 3. 19** *Flowchart* Halaman Menu Supplier Hapus Data

### 3.6.3.5 *Flowchart* Halaman Satuan Barang Tambah Data

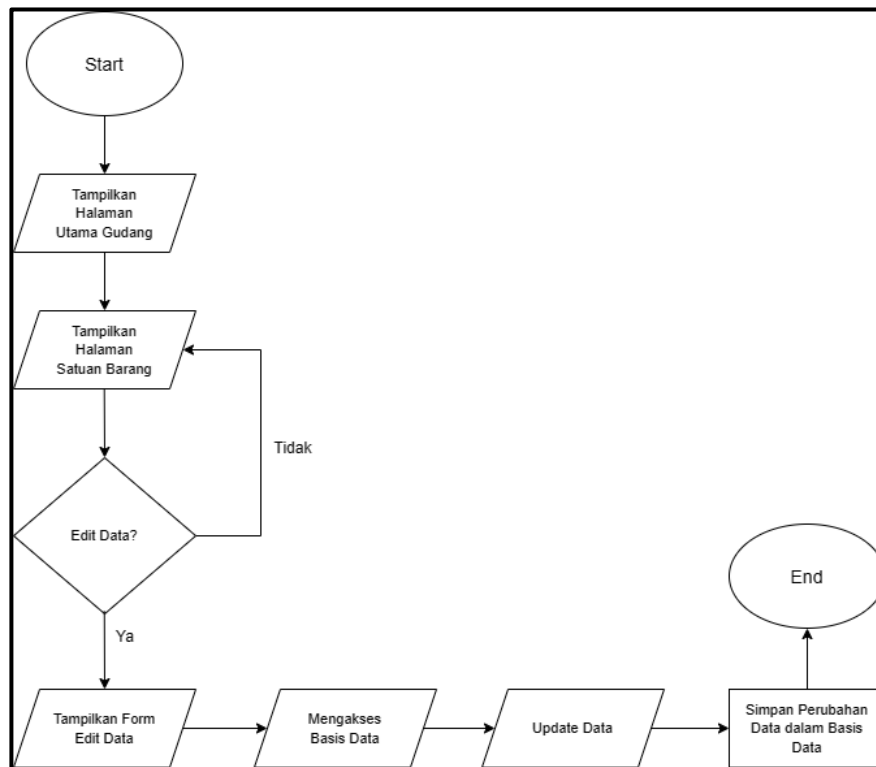
Pada menu satuan barang, admin dan karyawan bagian gudang dapat menambahkan data satuan barang. Jika data satuan barang telah diinput, maka data akan disimpan dalam database pada tabel satuan.



**Gambar 3. 20** *Flowchart* Halaman Menu Tambah Satuan Barang

### 3.6.3.6 *Flowchart* Halaman Edit Satuan Barang

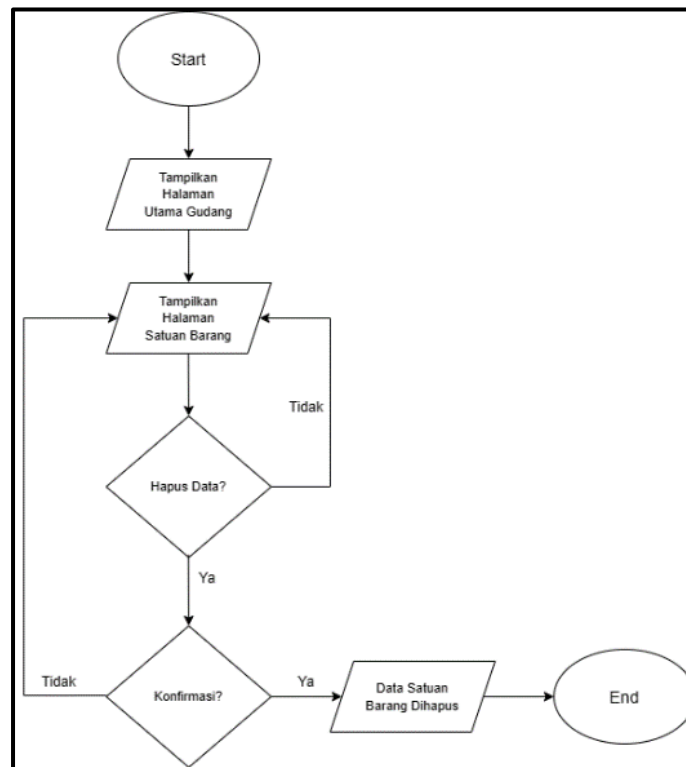
Pada menu satuan barang, admin dan karyawan bagian gudang dapat mengubah data satuan barang. Jika data satuan barang terdapat perubahan, maka data akan disimpan dalam database pada tabel satuan.



**Gambar 3. 21** *Flowchart* Halaman Edit Satuan Barang

### 3.6.3.7 Flowchart Halaman Hapus Satuan Barang

Pada menu satuan barang, admin dan karyawan bagian gudang dapat menghapus data satuan barang. Jika data satuan barang dihapus, maka data akan dihapus dalam database pada tabel satuan.

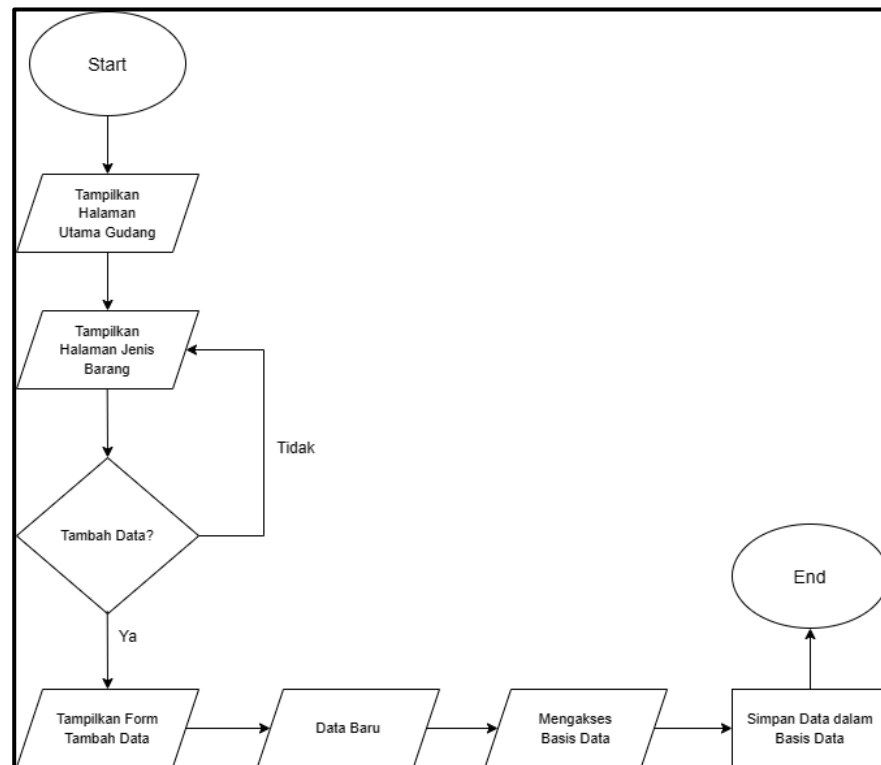


**Gambar 3. 22** *Flowchart* Halaman Hapus Satuan Barang

### 3.6.3.8 *Flowchart* Halaman Tambah Jenis Barang

Pada menu jenis barang, admin dan karyawan bagian gudang dapat menambahkan data jenis barang. Jika data jenis barang telah diinput, maka data akan disimpan dalam database pada tabel jenis.

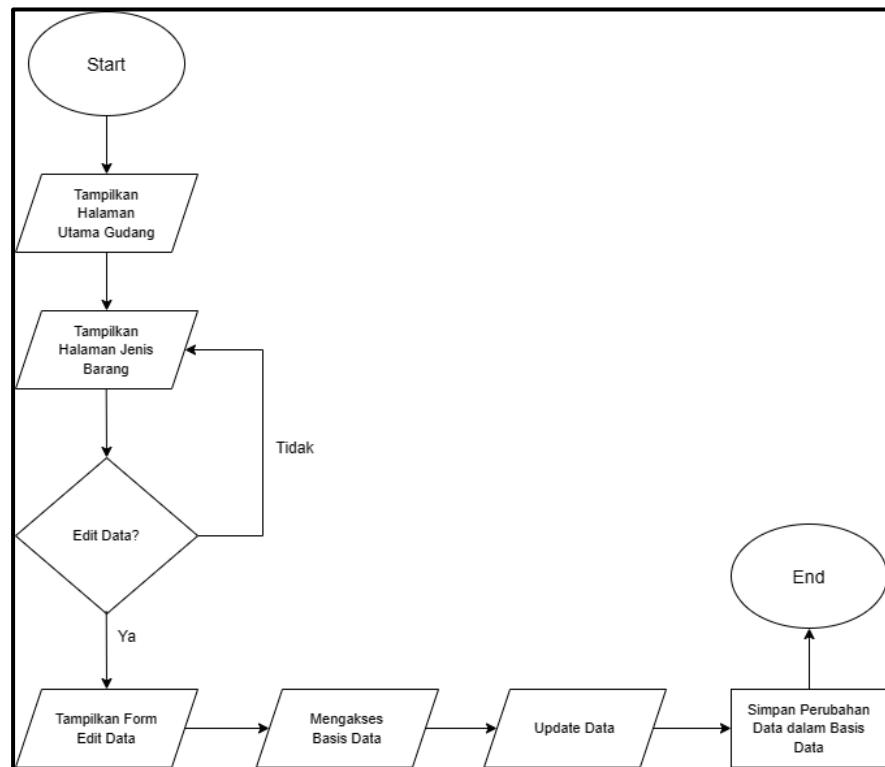




**Gambar 3. 23** *Flowchart* Halaman Menu Tambah Jenis Barang

### 3.6.3.9 Flowchart Halaman Edit Jenis Barang

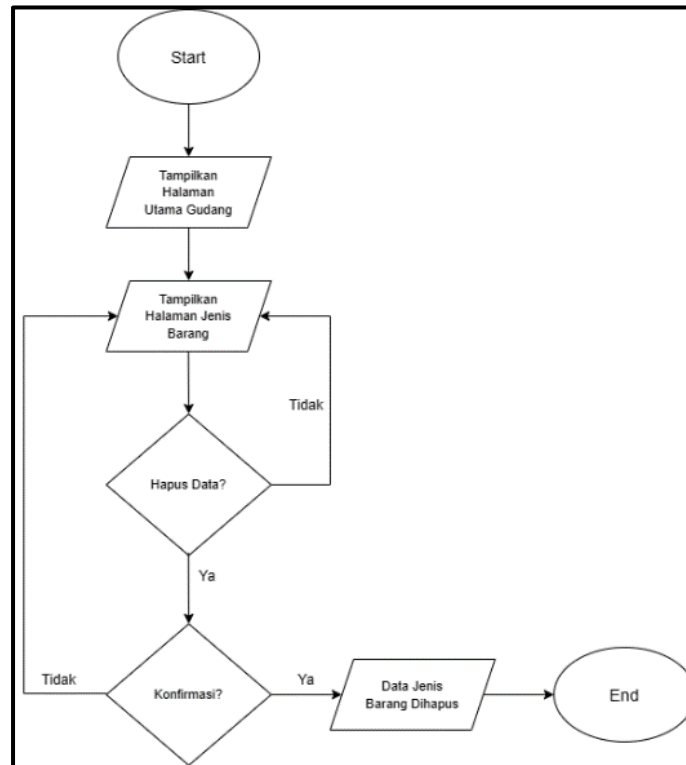
Pada menu jenis barang, admin dan karyawan bagian gudang dapat mengubah data jenis barang. Jika data jenis barang terdapat perubahan, maka data akan disimpan dalam database pada tabel jenis.



**Gambar 3. 24** *Flowchart* Halaman Menu Edit Jenis Barang

### 3.6.3.10 *Flowchart* Halaman Hapus Jenis Barang

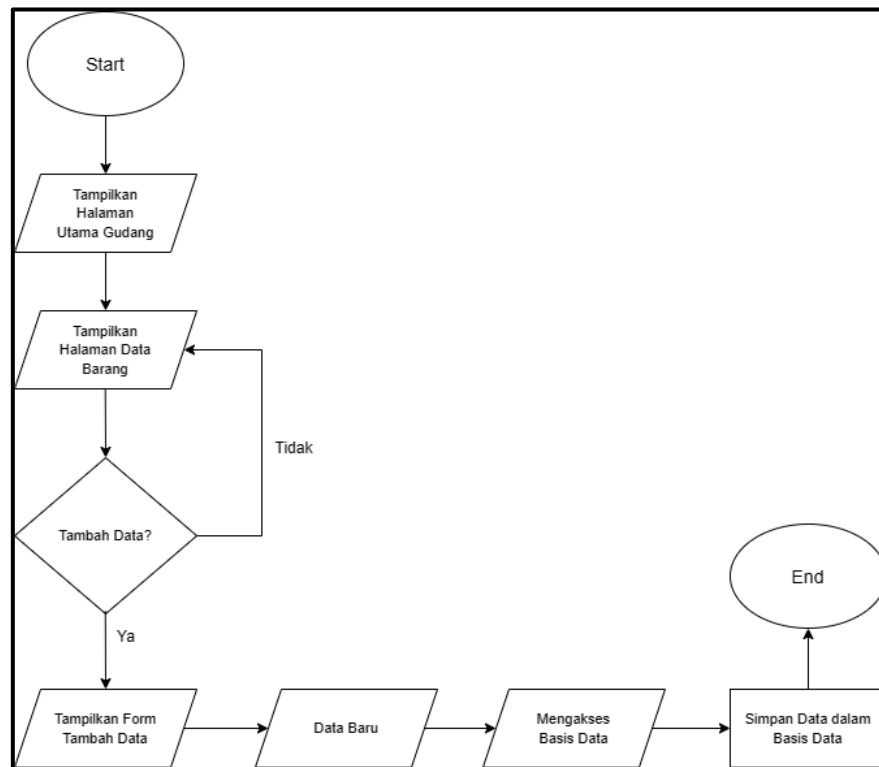
Pada menu akun user, hanya admin yang dapat menambahkan, mengubah, dan menghapus data *user*. Jika data user telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam database pada tabel *user*. Jika data dihapus dan telah terkonfirmasi maka data akan dihapus dalam *database*. Seluruh *action* yang telah dilakukan akan menampilkan *alert* apabila *action* yang dilakukan berhasil.



**Gambar 3. 25 Flowchart Halaman Menu Hapus Jenis Barang**

### **3.6.3.11 Flowchart Halaman Tambah Data Barang**

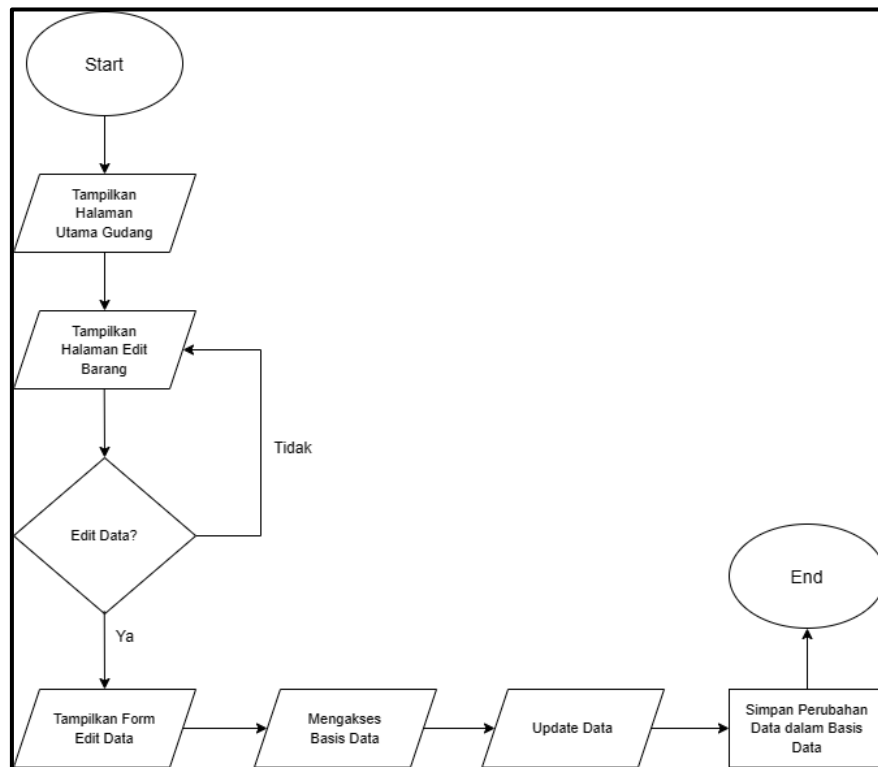
Pada menu data barang, admin dan karyawan bagian gudang yang dapat menambahkan data barang. Jika data barang telah diinput, maka data akan disimpan dalam database pada tabel barang.



**Gambar 3. 26** *Flowchart* Halaman Menu Tambah Data Barang

### 3.6.3.12 Flowchart Halaman Edit Data Barang

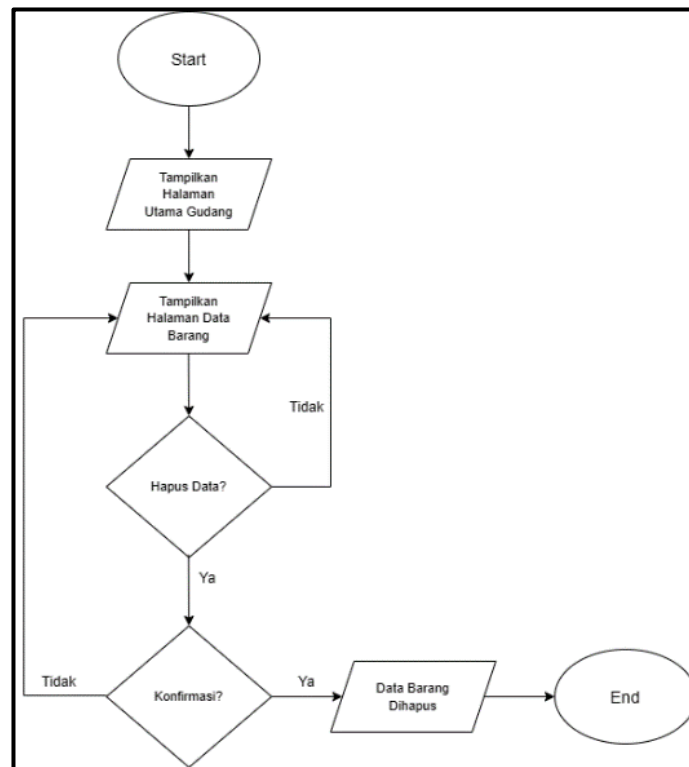
Pada menu data barang, admin dan karyawan bagian gudang yang dapat mengubah data barang. Jika data barang terdapat perubahan, maka data akan disimpan dalam database pada tabel barang.



**Gambar 3. 27 Flowchart Halaman Menu Edit Data Barang**

### 3.6.3.13 Flowchart Halaman Hapus Data Barang

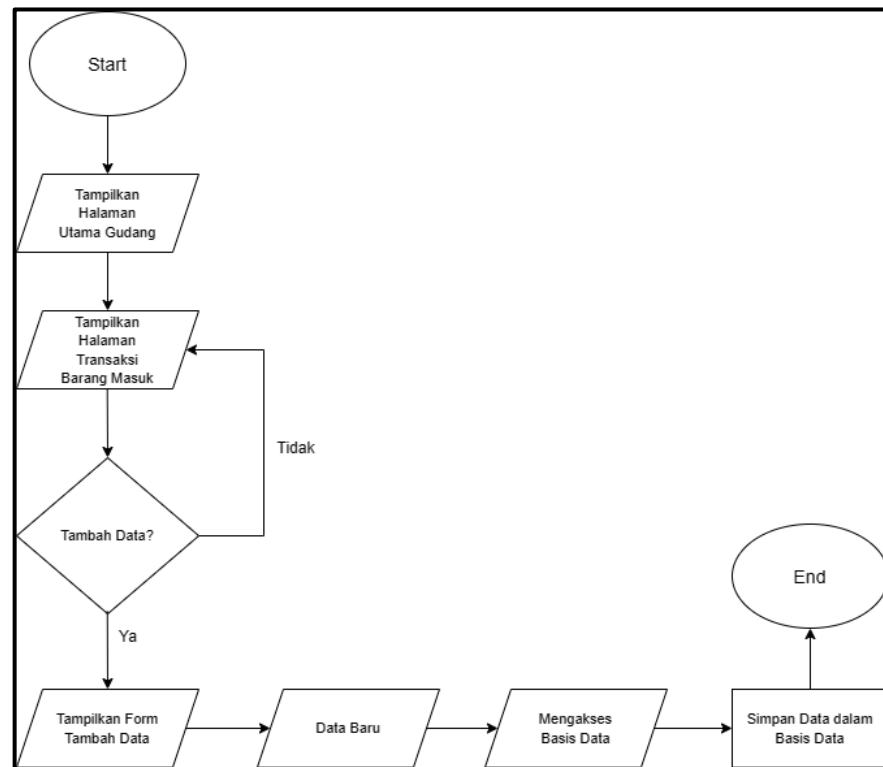
Pada menu data barang, admin dan karyawan bagian gudang yang dapat menghapus data barang. Jika data barang telah dihapus, maka data akan dihapus dalam database pada tabel barang.



**Gambar 3. 28 Flowchart Halaman Menu Hapus Data Barang**

### 3.6.3.14 Flowchart Halaman Tambah Transaksi Masuk

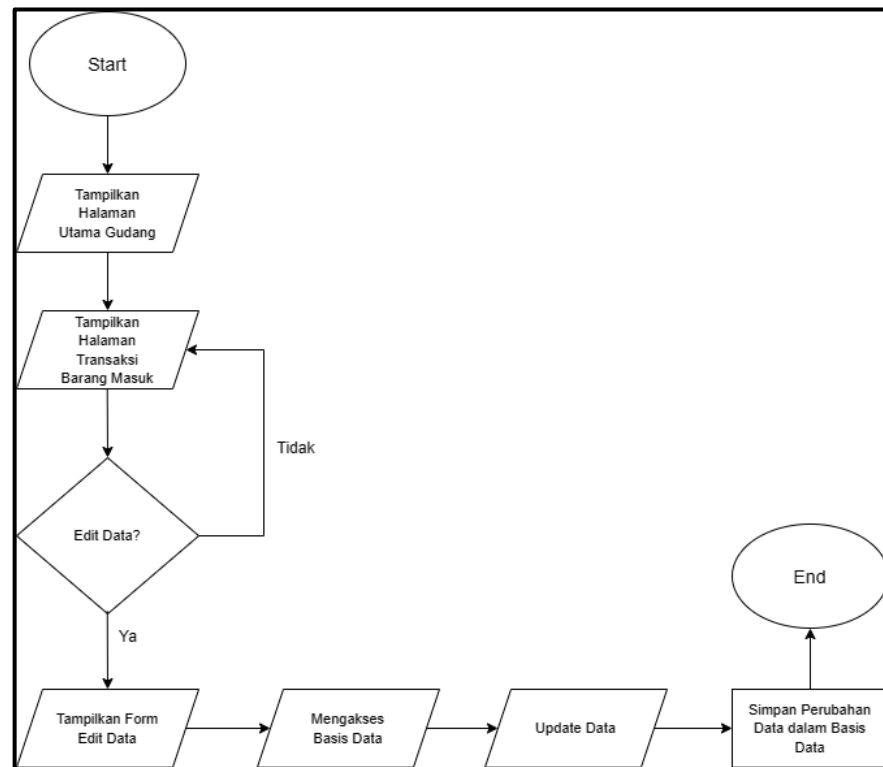
Pada menu akun transaksi barang masuk, admin dan karyawan bagian gudang yang dapat menambahkan data transaksi barang masuk. Jika data transaksi barang masuk telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam database pada tabel `barang_masuk`.



**Gambar 3. 29** *Flowchart* Halaman Menu Tambah Transaksi Masuk

### 3.6.3.15 Flowchart Halaman Edit Transaksi Masuk

Pada menu akun transaksi barang masuk, admin dan karyawan bagian gudang yang dapat mengubah data transaksi barang masuk. Jika data transaksi barang masuk telah terdapat perubahan, maka perubahan disimpan dalam database pada tabel `barang_masuk`.

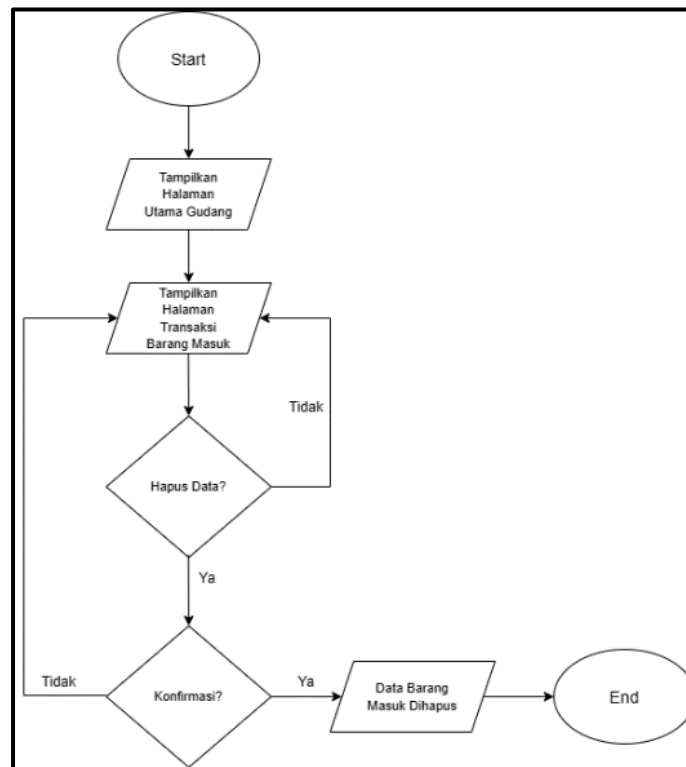


**Gambar 3. 30 Flowchart Halaman Menu Edit Transaksi Masuk**

### 3.6.3.16 Flowchart Halaman Hapus Transaksi Masuk

Pada menu akun transaksi barang masuk, admin dan karyawan bagian gudang dapat menghapus data transaksi barang masuk. Jika data transaksi barang masuk dihapus, maka data akan dihapus dalam database pada tabel `barang_masuk`.

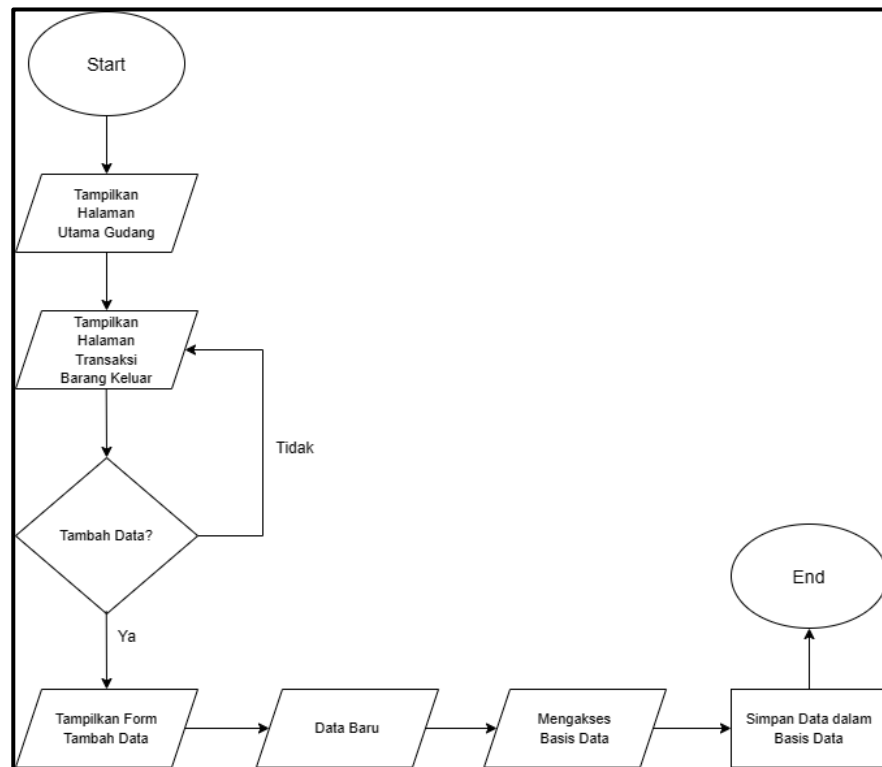




**Gambar 3. 31 Flowchart Halaman Menu Hapus Transaksi Masuk**

### 3.6.3.17 Flowchart Halaman Tambah Transaksi Keluar

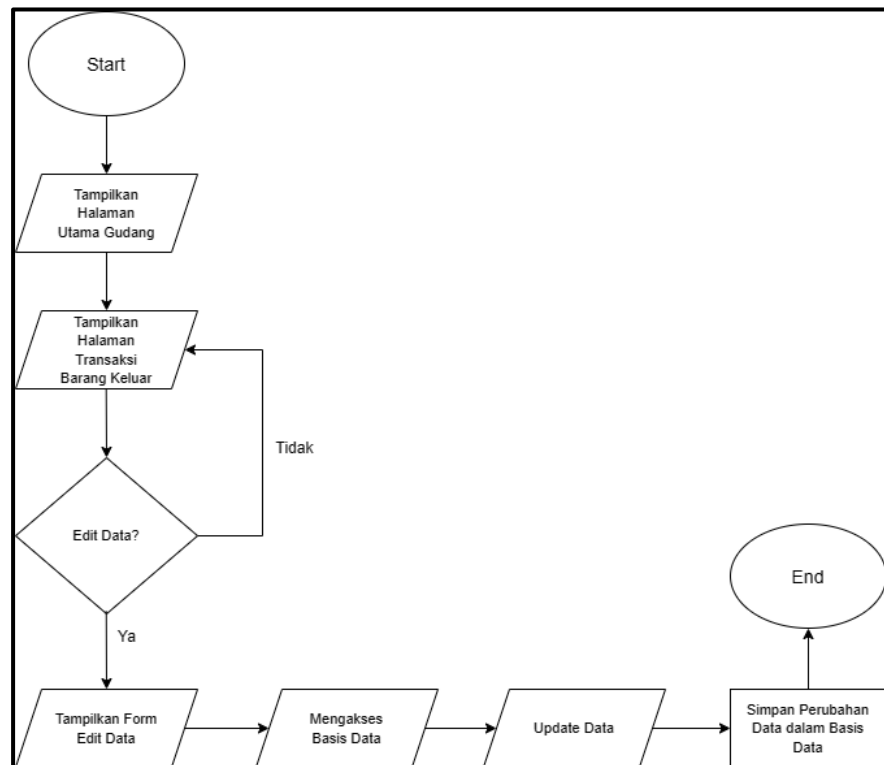
Pada menu akun transaksi barang keluar, admin dan karyawan bagian gudang yang dapat menambahkan transaksi barang keluar. Jika data transaksi barang keluar telah diinput dilakukan disimpan, maka data akan disimpan dalam database pada tabel `barang_keluar`.



**Gambar 3. 32 Flowchart Halaman Menu Tambah Transaksi Keluar**

### 3.6.3.18 Flowchart Halaman Edit Transaksi Keluar

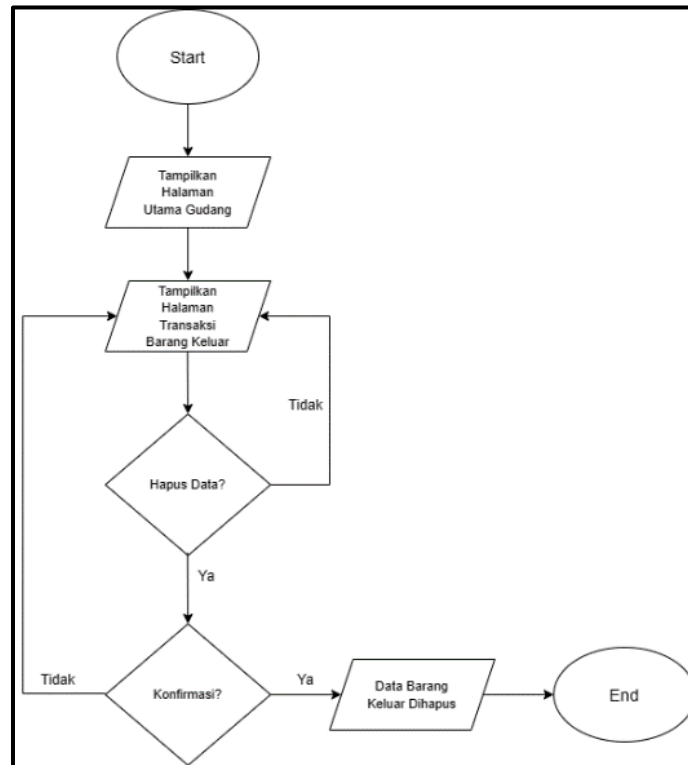
Pada menu akun transaksi barang keluar, admin dan karyawan bagian gudang yang dapat mengubah data transaksi barang keluar. Jika data transaksi barang keluar terdapat perubahan yang dilakukan disimpan, maka data akan disimpan dalam database pada tabel `barang_keluar`.



**Gambar 3. 33 Flowchart Halaman Menu Edit Transaksi Keluar**

### 3.6.3.19 Flowchart Halaman Hapus Transaksi Keluar

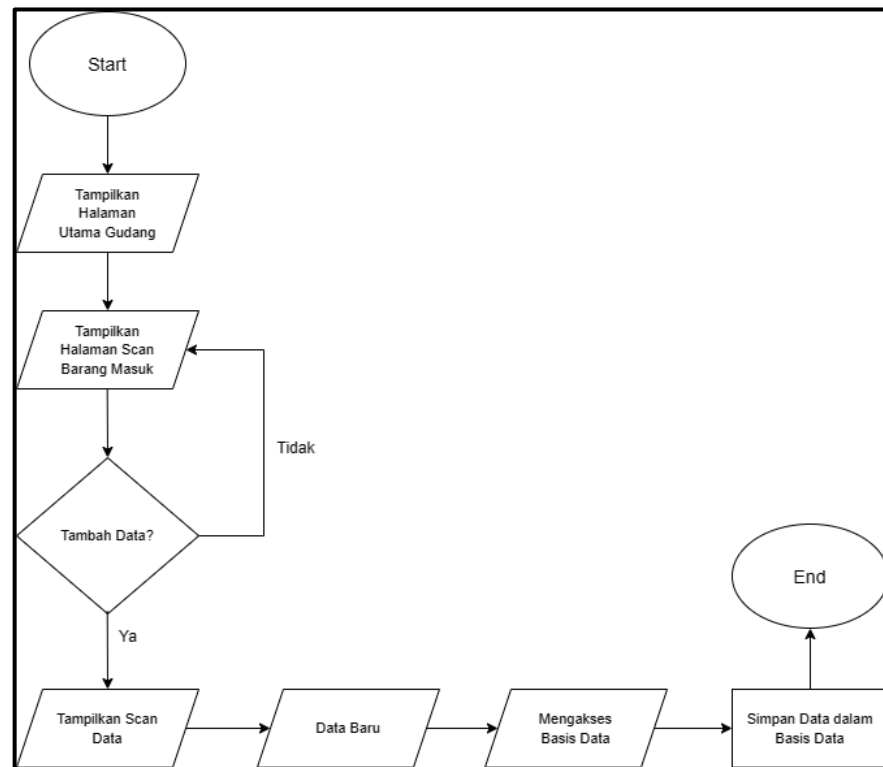
Pada menu akun transaksi barang keluar, admin dan karyawan bagian gudang yang dapat menghapus data transaksi barang keluar. Jika data dihapus dan telah terkonfirmasi maka data akan dihapus dalam database.



**Gambar 3. 34 Flowchart Halaman Menu Hapus Transaksi Keluar**

### 3.6.3.20 Flowchart Halaman Scan Barang Masuk

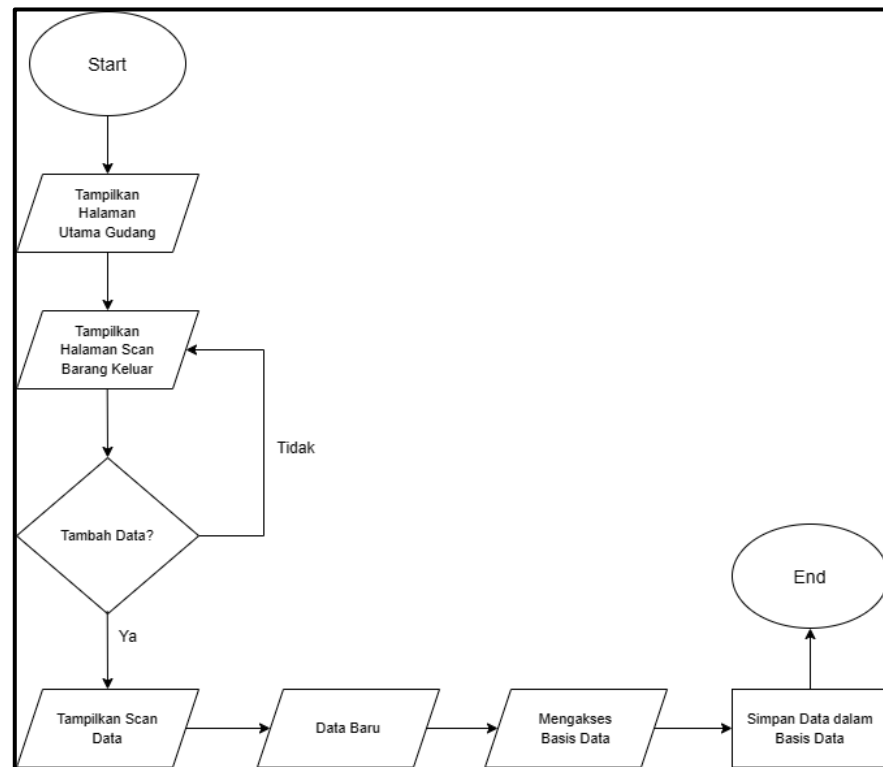
Pada menu scan barang masuk, admin dan karyawan bagian gudang yang dapat menambahkan barang masuk dengan menggunakan fitur scan. Jika barang masuk telah discan maka data akan disimpan dalam database pada tabel barang\_keluar.



**Gambar 3. 35 Flowchart Halaman Menu Scan Barang Masuk**

### 3.6.3.21 Flowchart Halaman Scan Barang Keluar

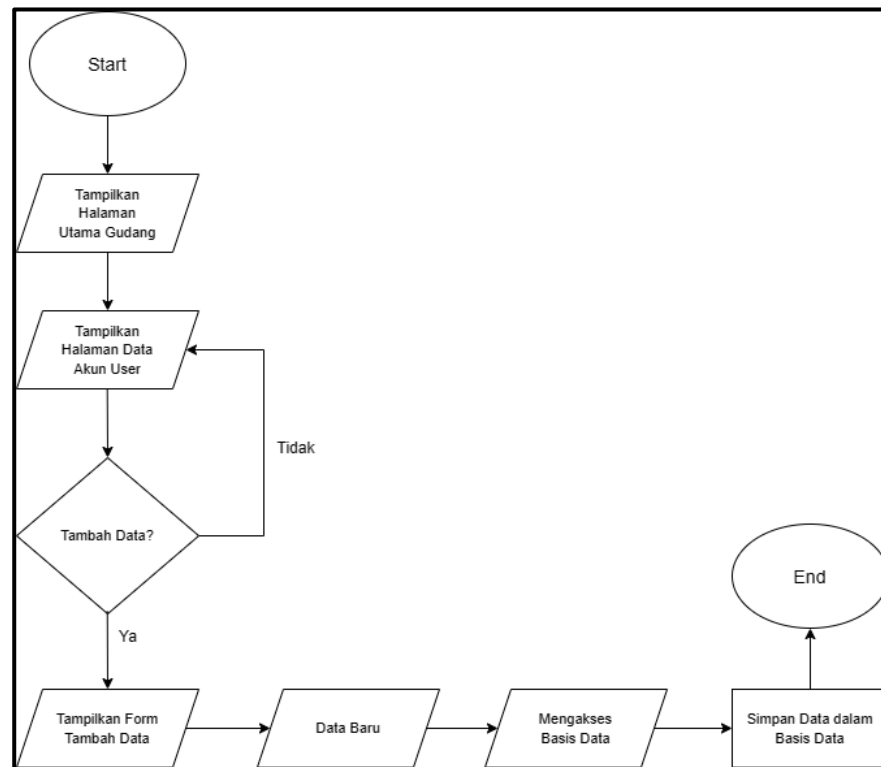
Pada menu scan barang keluar, admin dan karyawan bagian gudang yang dapat menambahkan barang keluar dengan menggunakan fitur scan. Jika barang keluar telah discan maka data akan disimpan dalam database pada tabel `barang_keluar`.



**Gambar 3. 36** *Flowchart* Halaman Menu Scan Barang Keluar

### 3.6.3.22 Flowchart Halaman Tambah Data User

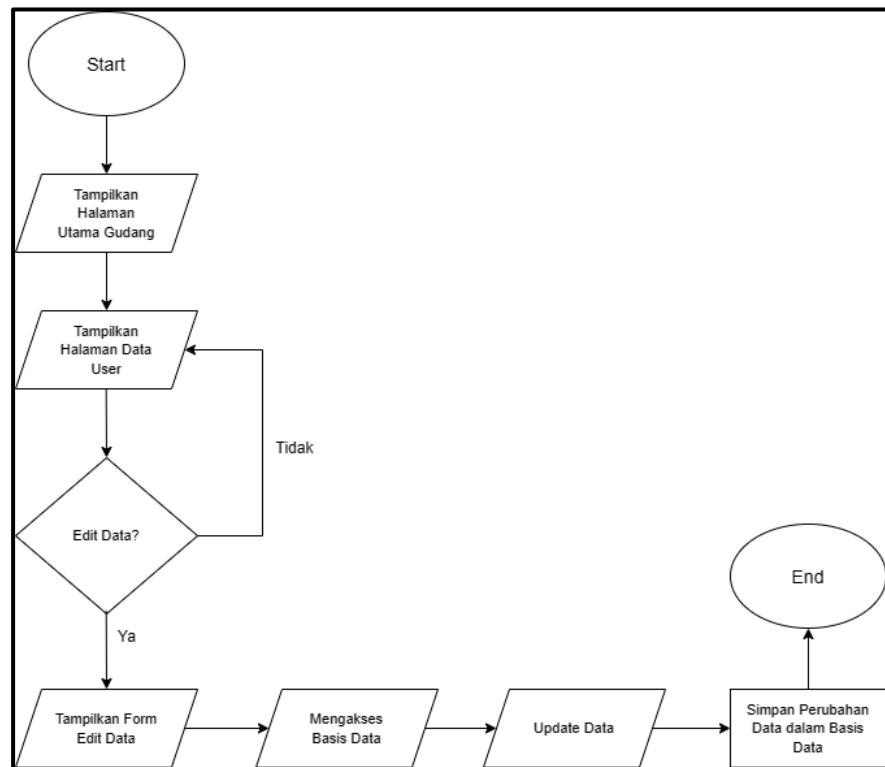
Pada menu akun user, hanya admin yang dapat menambahkan data *user*. Jika data user telah diinput, maka data akan disimpan dalam database pada tabel *user*.



**Gambar 3. 37 Flowchart Halaman Menu Tambah Data User**

### 3.6.3.23 Flowchart Halaman Edit Data Akun

Pada menu akun user, hanya admin yang dapat mengubah data *user*. Jika data user terdapat perubahan, maka data akan disimpan dalam database pada tabel *user*.

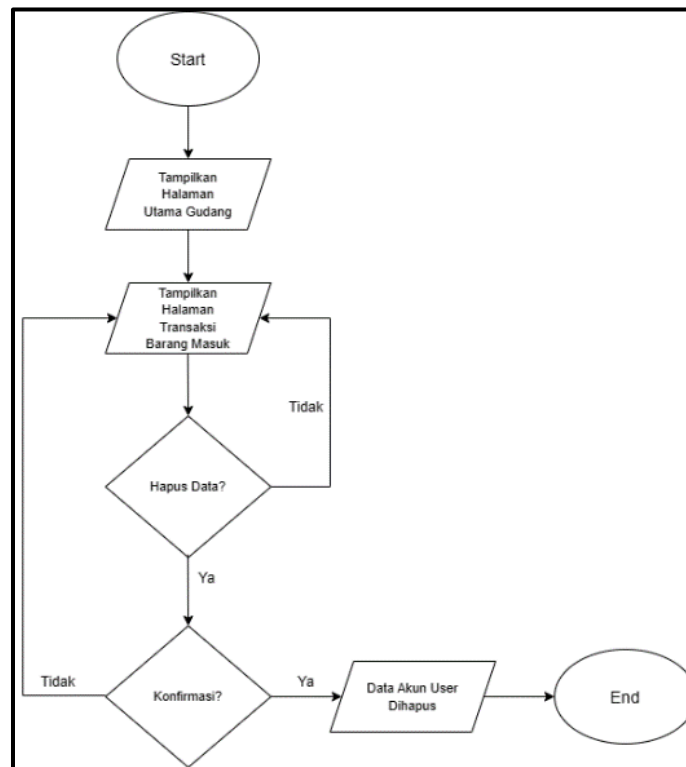


**Gambar 3. 38** *Flowchart* Halaman Menu Edit Data User

### 3.6.3.24 Flowchart Halaman Hapus Data

Pada menu akun user, hanya admin yang dapat menghapus data *user*. Jika data dihapus dan telah terkonfirmasi maka data akan dihapus dalam *database*. Seluruh *action* yang telah dilakukan akan menampilkan *alert* apabila *action* yang dilakukan berhasil.

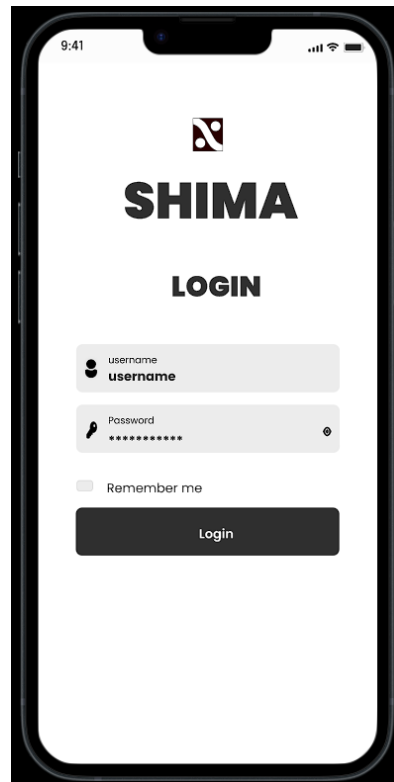




**Gambar 3. 39** *Flowchart* Halaman Menu Hapus Data User

### 3.6.4 Perancangan *Interface* Sistem

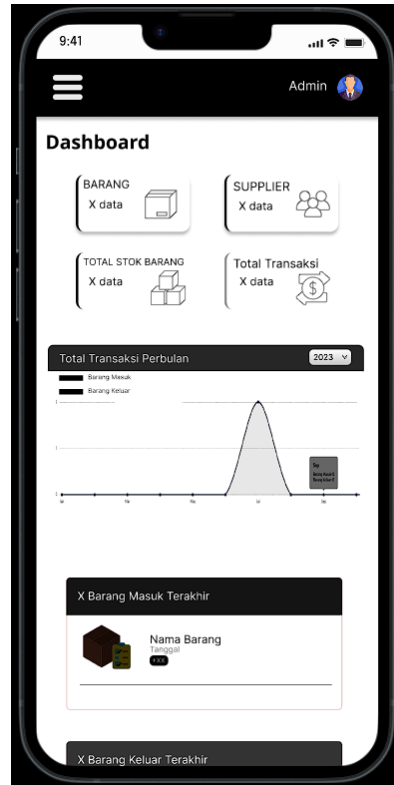
#### 3.6.4.1 Perancangan Halaman Menu Login



**Gambar 3. 40** Halaman Menu Login

Gambar diatas menjelaskan desain halaman menu login, dimana terdapat 2 textbox yaitu *email* dan *password* yang harus diisi untuk memiliki hak akses masuk ke halaman karyawan atau admin.

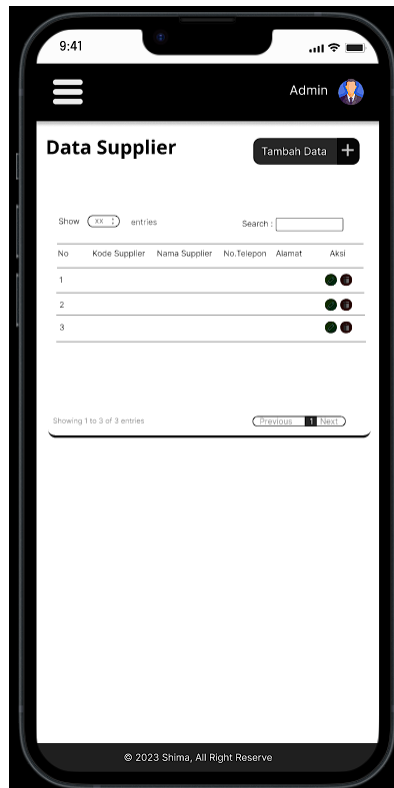
### 3.6.4.2 Perancangan Halaman Admin



**Gambar 3. 41 Halaman Admin**

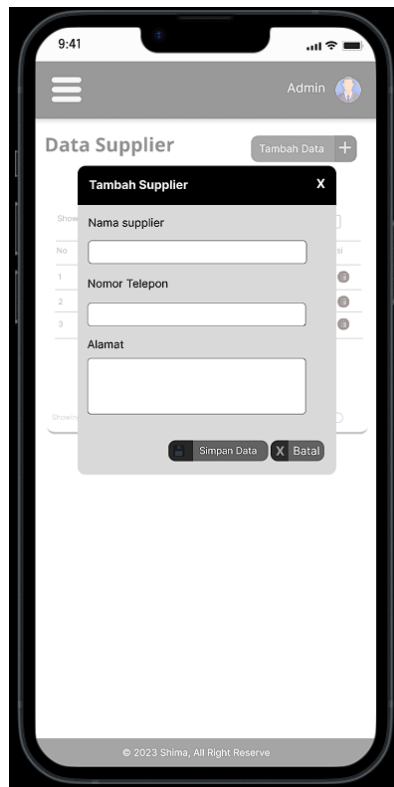
Gambar diatas menjelaskan desain tampilan halaman utama admin yang terdapat menu sesuai hak akses admin seperti menu supplier, master barang, transaksi, laporan dan akun *user*. Selanjutnya, terdapat menu profile dan logout pada bagian kanan atas yang mengarahkan kembali ke menu login.

### 3.3.4.3 Perancangan Halaman Supplier



**Gambar 3. 42 Halaman Supplier**

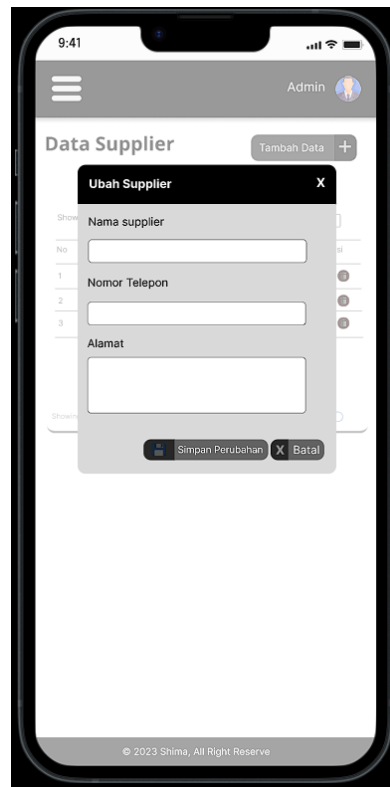
Gambar diatas menjelaskan desain dari menu supplier, data supplier dalam database akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan user ke form tambah atau edit data. Lalu tombol hapus apabila *user* ingin menghapus data.



The image shows a mobile application interface. At the top, the status bar displays the time 9:41, signal strength, and battery level. The app's header includes a menu icon, the user name 'Admin', and a profile picture. The main screen is titled 'Data Supplier' and features a 'Tambah Data +' button. A modal form titled 'Tambah Supplier' is overlaid on the screen. This form contains the following fields: 'Nama supplier', 'No', 'Nomor Telepon' (with sub-headers 1, 2, and 3), and 'Alamat'. At the bottom of the modal, there are two buttons: 'Simpan Data' and 'Batal'. A copyright notice '© 2023 Shima, All Right Reserve' is visible at the very bottom of the screen.

**Gambar 3. 43 Form Tambah Data Supplier**

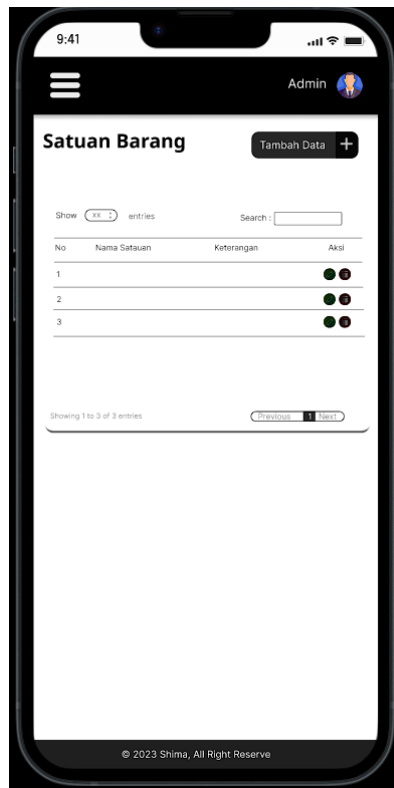
Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam database.



**Gambar 3. 44 Form Edit Data Supplier**

Gambar diatas menjelaskan desain form edit data, apabila *user* ingin mengubah data yang telah disimpan dalam database. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *user* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam database.

### 3.3.4.4 Perancangan Menu Master Barang



**Gambar 3. 45 Tampilan Menu Satuan Barang**

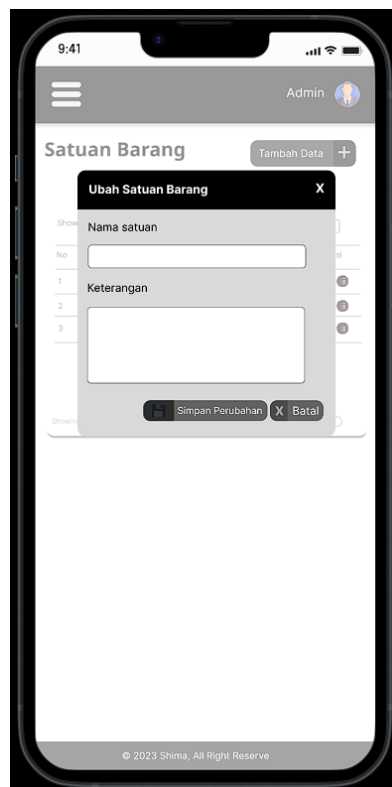
Gambar diatas menjelaskan desain dari menu satuan barang, data satuan barang dalam database akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan user ke form tambah atau edit data. Lalu tombol hapus apabila user ingin menghapus data.

The image shows a mobile application interface for managing units. The main screen is titled "Satuan Barang" and features a "Tambah Data +" button. A modal form titled "Tambah Satuan Barang" is displayed in the foreground. This form includes a "Nama satuan" label and a text input field, a "Keterangan" label, and a text area. At the bottom of the modal, there are "Simpan Data" and "Batal" buttons. The background screen shows a table with columns "No", "Nama", and "Keterangan", and a "Show" button. The time is 9:41 and the user is "Admin".

**Gambar 3. 46 Form Tambah Data Satuan Barang**

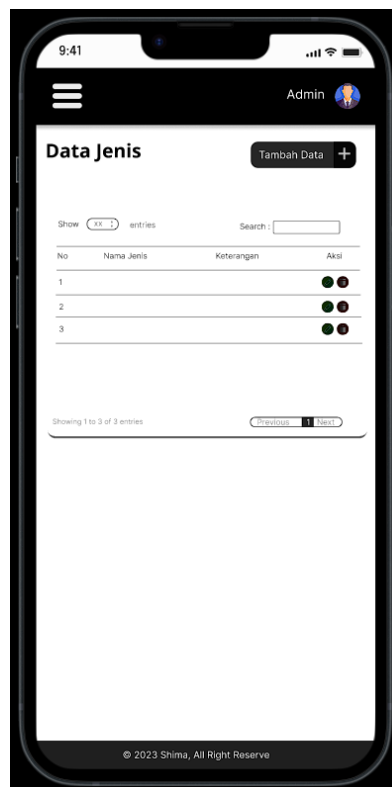
Gambar diatas menjelaskan desain form tambah data, dimana user harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam database.





**Gambar 3. 47 Form Edit Data Satuan Barang**

Gambar diatas menjelaskan desain form edit data, apabila user ingin mengubah data yang telah disimpan dalam *database*. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *user* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



**Gambar 3. 48 Tampilan Menu Jenis Barang**

Gambar diatas menjelaskan desain dari menu jenis barang, data jenis barang dalam database akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan *user* ke form tambah atau edit data. Lalu tombol hapus apabila *user* ingin menghapus data.



The screenshot displays a mobile application interface for managing goods. At the top, the status bar shows the time as 9:41 and the user is logged in as 'Admin'. The main header is 'Data Jenis' with a 'Tambah Data +' button. A modal window titled 'Tambah Jenis Barang' is open, featuring a 'Nama Jenis' label and a text input field. Below this is a 'Keterangan' label and a larger text area. At the bottom of the modal, there are two buttons: 'Simpan Data' and 'Batal'. The background shows a list of items with columns for 'No', 'Nama', and 'Keterangan', and a 'Show' button on the left.

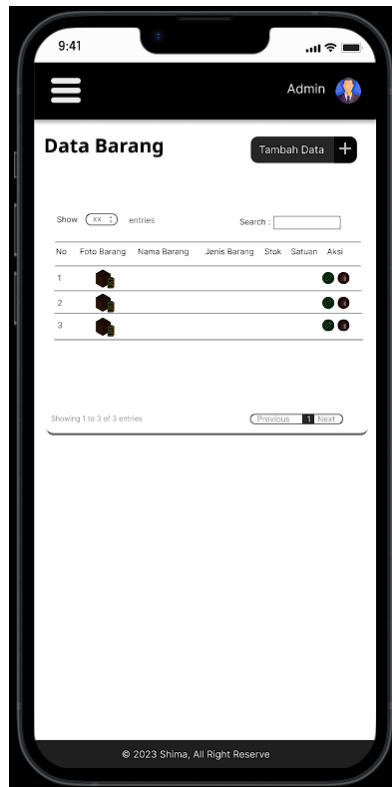
**Gambar 3. 49 Form Tambah Data Jenis Barang**



The screenshot displays the same mobile application interface as in the previous image, but with a modal window titled 'Ubah Jenis Barang' open. This modal contains the same 'Nama Jenis' and 'Keterangan' fields as the 'Tambah' modal. However, the bottom buttons are 'Simpan Perubahan' and 'Batal'. The background interface remains identical to the previous screenshot.

**Gambar 3. 50 Form Edit Data Jenis Barang**

Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



**Gambar 3. 51 Tampilan Menu Data Barang**

Gambar diatas menjelaskan desain dari menu data barang, data barang dalam database akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan *user* ke form tambah atau edit data. Lalu tombol hapus apabila user ingin menghapus data.

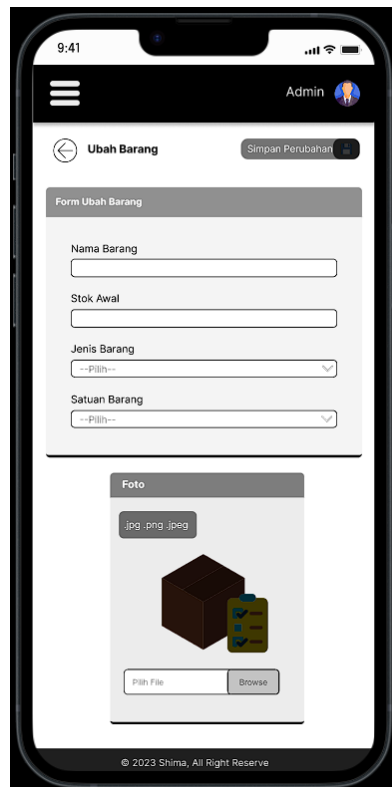
The image shows a mobile application interface for adding a new item. The screen is titled "Tambah Barang" (Add Item) and features a "Simpan Data" (Save Data) button. The form includes the following fields:

- Nama Barang (Item Name)
- Stok Awal (Initial Stock)
- Jenis Barang (Item Type) - dropdown menu
- Satuan Barang (Item Unit) - dropdown menu

Below the form is a "Foto" (Photo) section with a file upload area and a "Browse" button. The bottom of the screen displays the copyright notice: "© 2023 Shima, All Right Reserve".

**Gambar 3. 52 Form Tambah Data Barang**

Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



The image shows a mobile application interface for editing item data. The screen is titled "Ubah Barang" (Edit Item) and features a "Simpan Perubahan" (Save Changes) button. The form includes the following fields:

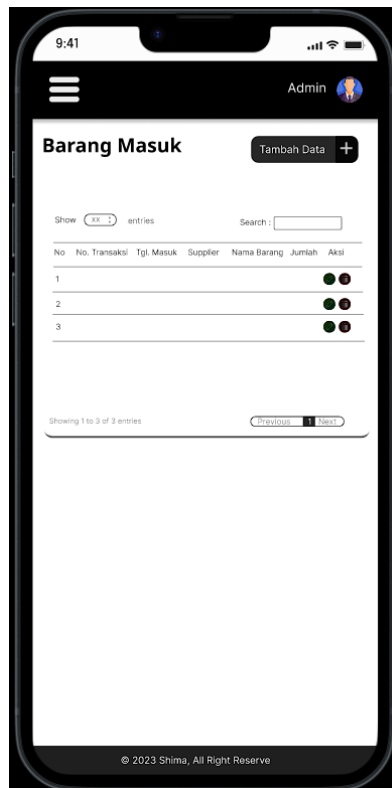
- Nama Barang (Item Name)
- Stok Awal (Initial Stock)
- Jenis Barang (Item Type) with a dropdown menu showing "--Pilih--"
- Satuan Barang (Item Unit) with a dropdown menu showing "--Pilih--"

Below the form is a "Foto" (Image) section with a file upload area showing ".jpg .png .jpeg" and a "Browsar" (Browse) button. The interface is displayed on a smartphone screen with a status bar at the top showing the time 9:41 and the user "Admin".

**Gambar 3. 53 Form Edit Data Barang**

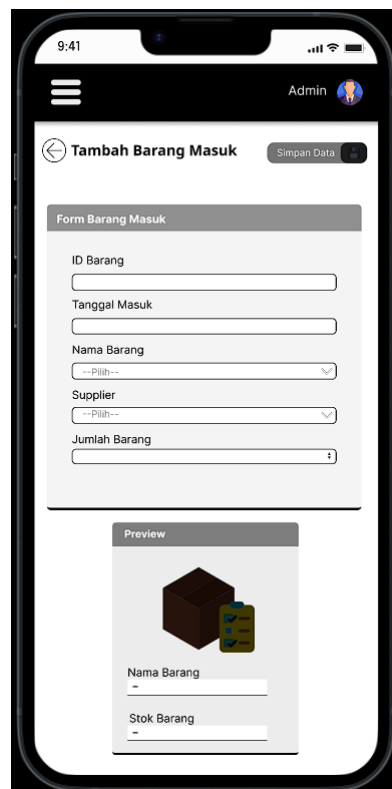
Gambar diatas menjelaskan desain form edit data, apabila *user* ingin mengubah data yang telah disimpan dalam database. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *user* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

### 3.3.4.5 Perancangan Halaman Transaksi



**Gambar 3. 54 Tampilan Menu Transaksi**

Gambar diatas menjelaskan desain dari menu transaksi barang, data satuan barang dalam *database* akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan *user* ke form tambah atau edit data. Lalu tombol hapus apabila *user* ingin menghapus data.




The image shows a mobile application interface for adding incoming goods data. The screen is titled "Tambah Barang Masuk" (Add Incoming Goods) and features a "Simpan Data" (Save Data) button in the top right corner. The form is divided into two sections: "Form Barang Masuk" and "Preview".

**Form Barang Masuk**

- ID Barang:
- Tanggal Masuk:
- Nama Barang:
- Supplier:
- Jumlah Barang:

**Preview**

- Image: 
- Nama Barang:
- Stok Barang:

**Gambar 3. 55 Form Tambah Data Transaksi Barang Masuk**

Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



9:41

Admin

← Ubah Barang Masuk Simpan Perubahan

**Form Barang Masuk**

ID Barang

Tanggal Masuk

Nama Barang

Supplier

Jumlah Barang

**Preview**

Nama Barang

Stok Barang

**Gambar 3. 56 Form Edit Data Transaksi Barang Masuk**

9:41

Admin

**Barang Keluar** Tambah Data

Show  entries Search:

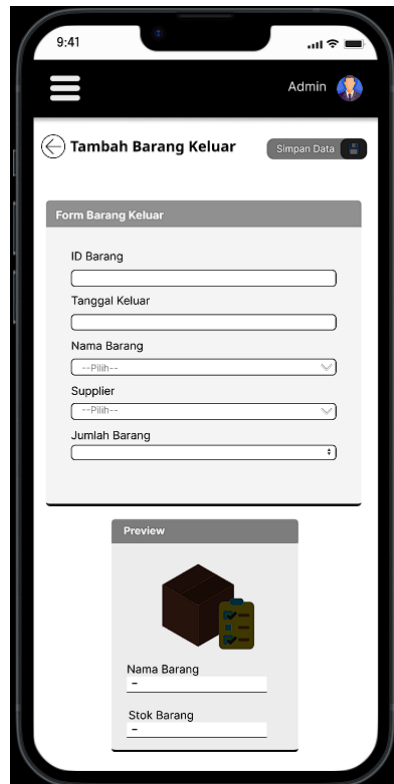
No	No. Transaksi	Tgl. Keluar	Supplier	Nama Barang	Jumlah	Aksi
1						
2						
3						

Showing 1 to 3 of 3 entries Previous 1 Next

© 2023 Shimo, All Right Reserve

**Gambar 3. 57 Tampilan Menu Transaksi Barang Keluar**

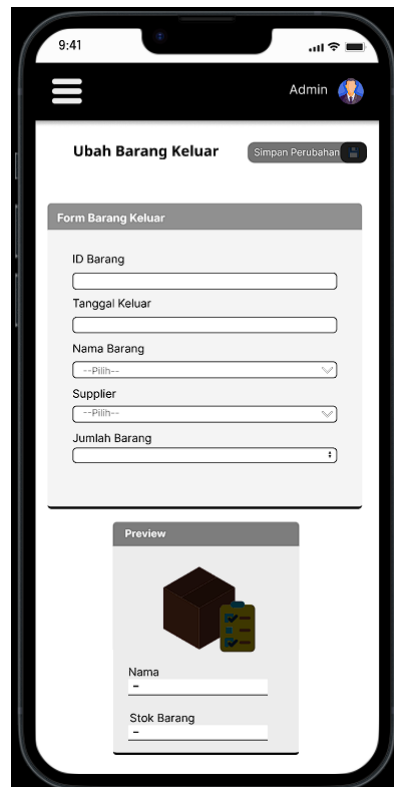
Gambar diatas menjelaskan desain dari menu transaksi barang keluar, data transaksi barang keluar dalam *database* akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan *user* ke *form* tambah atau edit data. Lalu tombol hapus apabila *user* ingin menghapus data.



The image shows a mobile application interface for adding an outgoing transaction. At the top, the status bar shows the time 9:41 and signal strength. The app header includes a hamburger menu icon, the user name 'Admin', and a profile picture. The main title is 'Tambah Barang Keluar' with a back arrow on the left and a 'Simpan Data' button on the right. The form is titled 'Form Barang Keluar' and contains the following fields: 'ID Barang' (text input), 'Tanggal Keluar' (text input), 'Nama Barang' (dropdown menu with '--Pilih--'), 'Supplier' (dropdown menu with '--Pilih--'), and 'Jumlah Barang' (text input with a numeric keypad icon). Below the form is a 'Preview' section with a 3D box icon and two fields: 'Nama Barang' and 'Stok Barang', both with dashes indicating they are empty.

**Gambar 3. 58 Form Tambah Data Transaksi Barang Keluar**

Gambar diatas menjelaskan desain form tambah data, dimana user harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



The screenshot shows a mobile application interface for editing outgoing goods data. The title is "Ubah Barang Keluar" (Edit Outgoing Goods) with a "Simpan Perubahan" (Save Changes) button. The form is titled "Form Barang Keluar" and contains the following fields:

- ID Barang: Text input field
- Tanggal Keluar: Date input field
- Nama Barang: Text input field
- Supplier: Dropdown menu (Pilih--)
- Jumlah Barang: Text input field with a numeric keypad icon

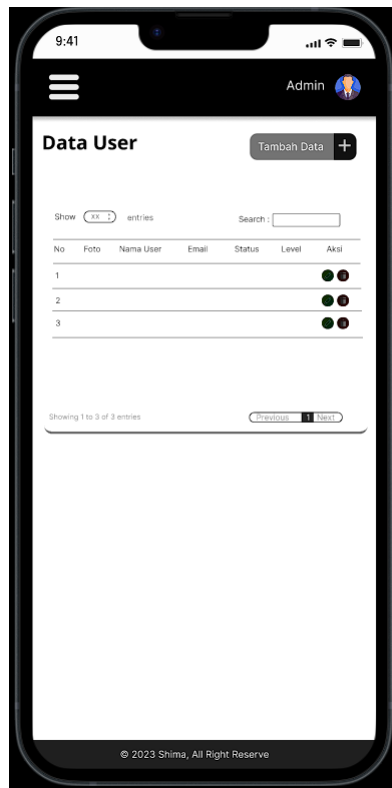
Below the form is a "Preview" section showing a 3D box icon and the following fields:

- Nama: Text input field
- Stok Barang: Text input field

**Gambar 3. 59 Form Edit Data Transaksi Barang Keluar**

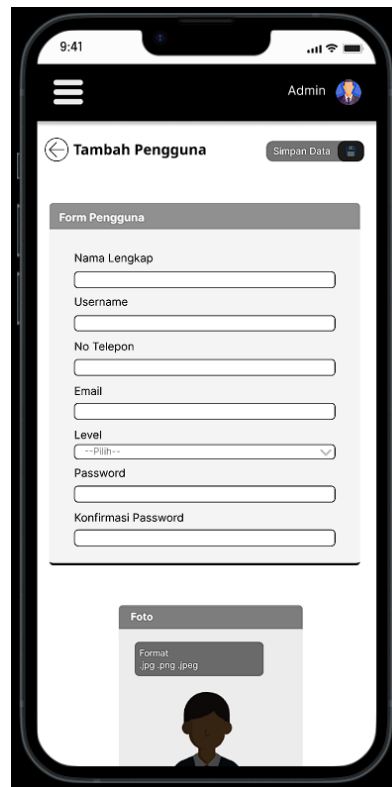
Gambar diatas menjelaskan desain form edit data, apabila *user* ingin mengubah data yang telah disimpan dalam *database*. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *user* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

### 3.3.4.6 Perancangan Menu Data User



**Gambar 3. 60 Tampilan Menu Data User**

Gambar diatas menjelaskan desain dari menu data *user* yang hanya dapat diakses oleh admin, data *user* dalam *database* akan ditampilkan. Kemudian terdapat tombol tambah data dan edit yang akan mengarahkan admin ke form tambah atau edit data. Lalu tombol hapus apabila *user* ingin menghapus data.



The image shows a mobile application interface for adding a user. The screen is titled "Tambah Pengguna" (Add User) and features a "Simpan Data" (Save Data) button in the top right corner. The form is organized into two main sections: "Form Pengguna" and "Foto".

The "Form Pengguna" section contains the following fields:

- Nama Lengkap (Full Name)
- Username
- No Telepon (Phone Number)
- Email
- Level (Level, with a dropdown menu showing "--Pilih--")
- Password
- Konfirmasi Password (Confirm Password)

The "Foto" section includes a placeholder image of a person and a label indicating the supported image formats: "Format: .jpg, .png, .jpeg".

**Gambar 3. 61 Form Tambah Data User**

Gambar diatas menjelaskan desain form tambah data, dimana admin harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

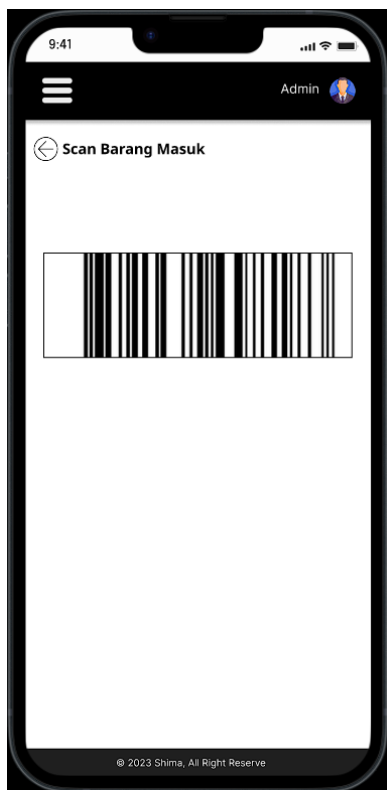
The image shows a mobile application interface for editing user data. The form is displayed on a smartphone screen and contains the following elements:

- No Telepon:** A text input field.
- Email:** A text input field.
- Level:** A dropdown menu with "--Pilih--" and a downward arrow.
- Status:** A dropdown menu with "--Pilih--" and a downward arrow.
- Foto:** A section for uploading a profile picture. It includes a "Format" label with ".jpg, .png, .jpeg" below it, a placeholder image of a person, and two buttons: "Pilih File" and "Browse".
- Ubah Password:** A section for changing the password. It includes a dark grey button with the text "Kosongkan jika tidak ingin merubah!". Below this are three input fields: "Password", "Konfirmasi Password", and a third empty input field.
- Footer:** A small copyright notice at the bottom: "© 2023 Shima, All Right Reserve".

**Gambar 3. 62 Form Edit Data User**

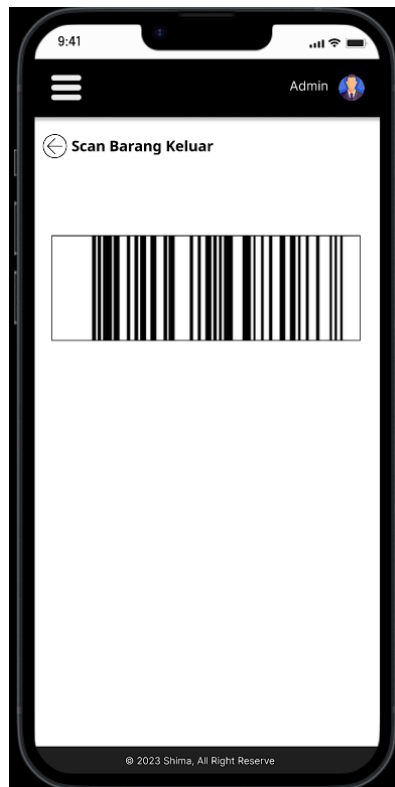
Gambar diatas menjelaskan desain form edit data, apabila admin ingin mengubah data yang telah disimpan dalam *database*. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *admin* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

### 3.3.4.7 Perancangan Menu Scan Barang



**Gambar 3. 63 Form Scan Barang Masuk**

Gambar diatas desain form scan untuk barang masuk yang akan disimpan dalam database.



**Gambar 3. 64 Form Scan Barang Keluar**

Gambar diatas desain form scan untuk barang keluar yang akan diupdate dalam *database*.