

BAB III

ANALISIS SISTEM

3.1 Unit Kegiatan Kerja Praktik

Kegiatan kerja praktik dilakukan pada unit IT(Technology Information) dan akan membuat ulang sebuah *warehouse* berbasis web dengan tambahan fitur baru yaitu fitur *edit* bagi pengguna PT. Shima Prima Utama yang membutuhkan.

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 wawancara kebutuhan perusahaan.	
2.	26 Juni s.d 27 Juni 2023	Mempelajari <i>Codeigniter 3</i> dan SQL	
3.	1 Juli dan 3 Juli 2023	<i>Desain Front End</i>	

Tabel 3.2 Tabel Kegiatan Kerja Praktik (Lanjutan)

No	Tanggal	Kegiatan	Dokumentasi
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	<i>Desain UI Project</i>	
7.	11 Juli s.d 12 Juli 2023	<i>Desain Front End</i>	

Tabel 3.3 Tabel Kegiatan Kerja Praktik (Lanjutan)

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

3.4 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. Sehingga karyawan juga dapat menambahkan akun yang lebih baik hanya admin yang memiliki akses tersebut agar lebih terjaga keamanan data.

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 *re-design user interface* pada *website 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 mengubah data *supplier*, barang dan transaksi sesuai hak akses

3.5.3.2 Kebutuhan Non-Fungsional

Kebutuhan non – fungsional adalah kebutuhan sistem yang memiliki batasan. Beberapa kebutuhan non – fungsional pada perancangan *website 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 produk, artikel, kategori, dan akun.

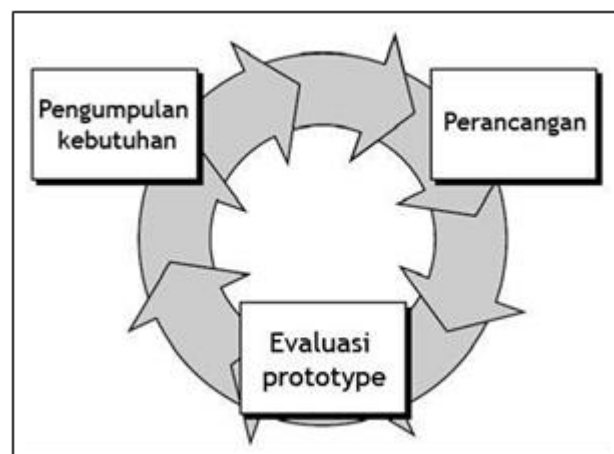
3.5.4 Prosedur Sistem Yang Diusulkan

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).

3.5 Metodologi Pengembangan Sistem

Metodelogi 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 akan dirancang.

2. Perancangan

Pada tahap ini dilakukan *re-design user interface* dan rancangan terhadap 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.

3.6 Perancangan Sistem Yang Diusulkan

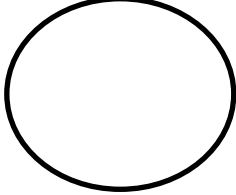

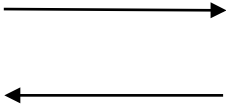

Perancangan *website* company profile ini terdiri dari pemodelan proses sistem yang dilakukan dengan *Data Flow Diagram* (DFD), pemodelan data sistem yang dilakukan dengan Entity Relationship Diagram (ERD) dan spesifikasi tabel-tabel *database* dan pemodelan sistem yang dilakukan dengan *Flowchart*. 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 (A.S., Rosa dan Shalahuddin, 2016).

Menurut Edward Yourdon dan Tom DeMarco, notasi – notasi pada DFD dapat dilihat pada tabel berikut ini (A.S., Rosa dan Shalahuddin, 2016):

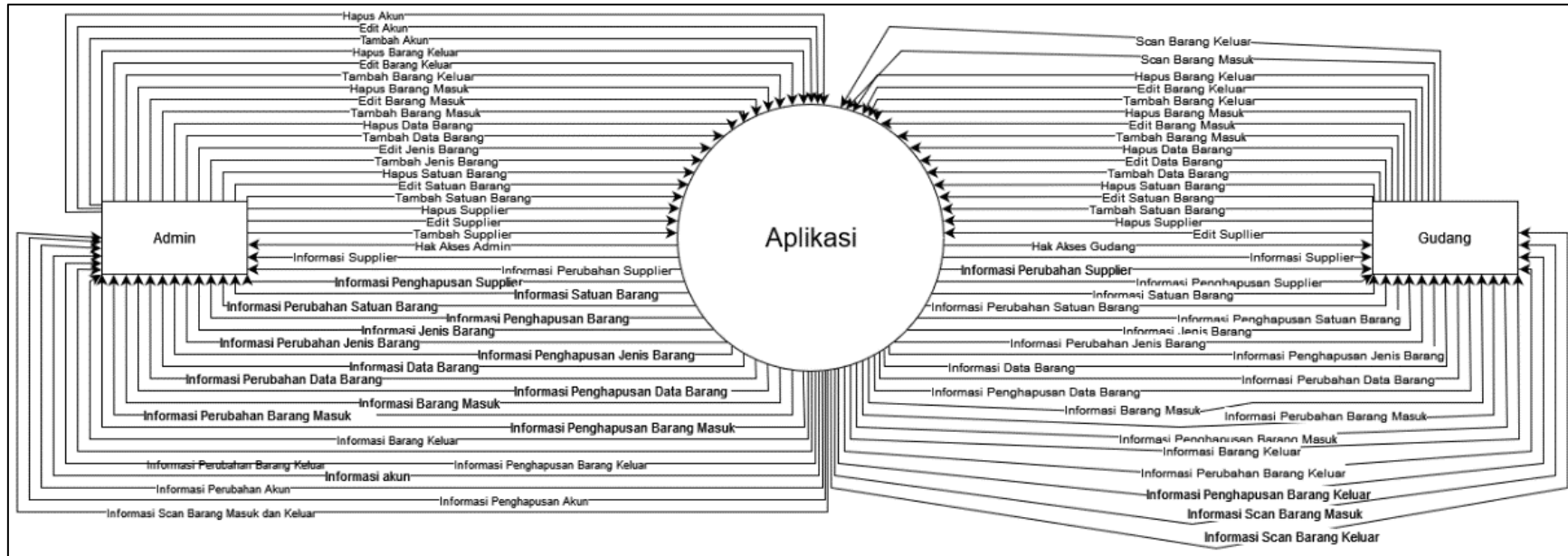
Tabel 3.4 Notasi – Notasi DFD
(Sumber: (A.S., Rosa dan Salahuddin, 2016))

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 – Tabel basis data yang dibutuhkan sesuai dengan <i>Entity Relationship Diagram</i> .

Berikut ini adalah tahapan – tahapan perancangan menggunakan *Data Flow Diagram* (DFD) (A.S., Rosa dan Shalahuddin, 2016):

3.6.1.1 DFD Level 0 (Diagram Konteks) Yang Diusulkan

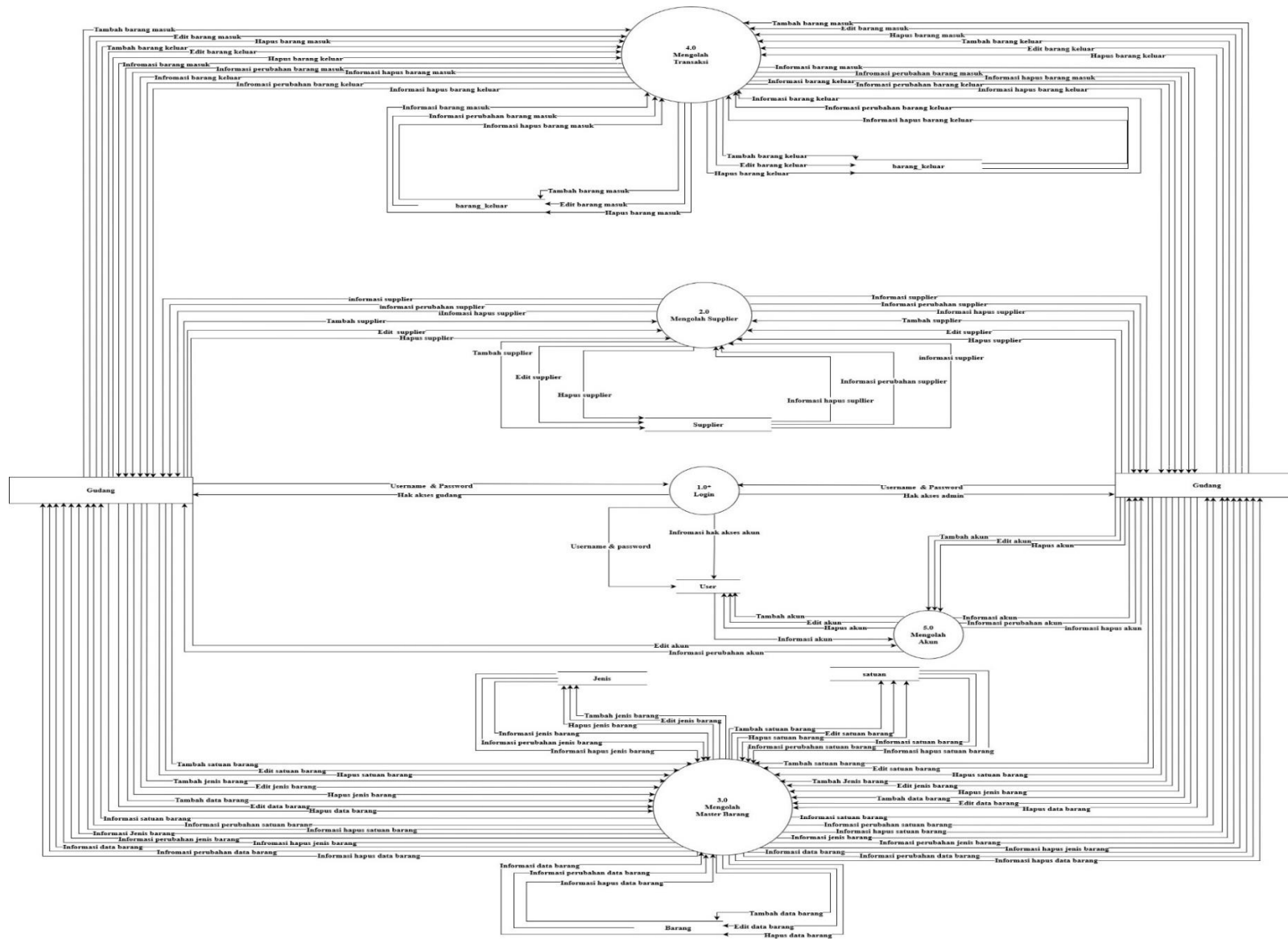
Diagram Konteks menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. Diagram konteks digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah 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

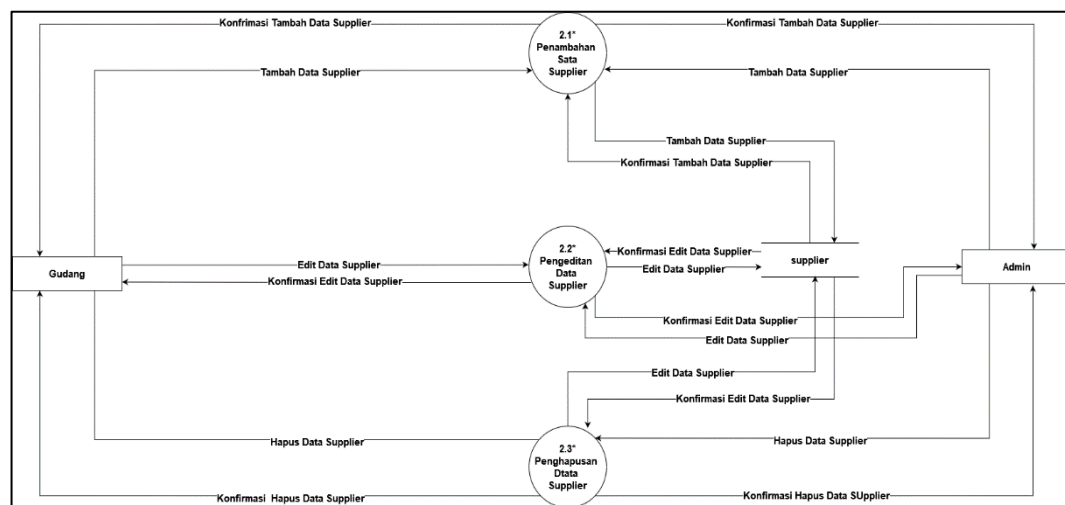
DFD Level 1 digunakan untuk menggambarkan modul– modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 1 dari sistem yang diusulkan:



Gambar 3. 4 DFD Level 1 yang diusulkan

3.6.1.3 DFD Level 2 Mengolah *Supplier*

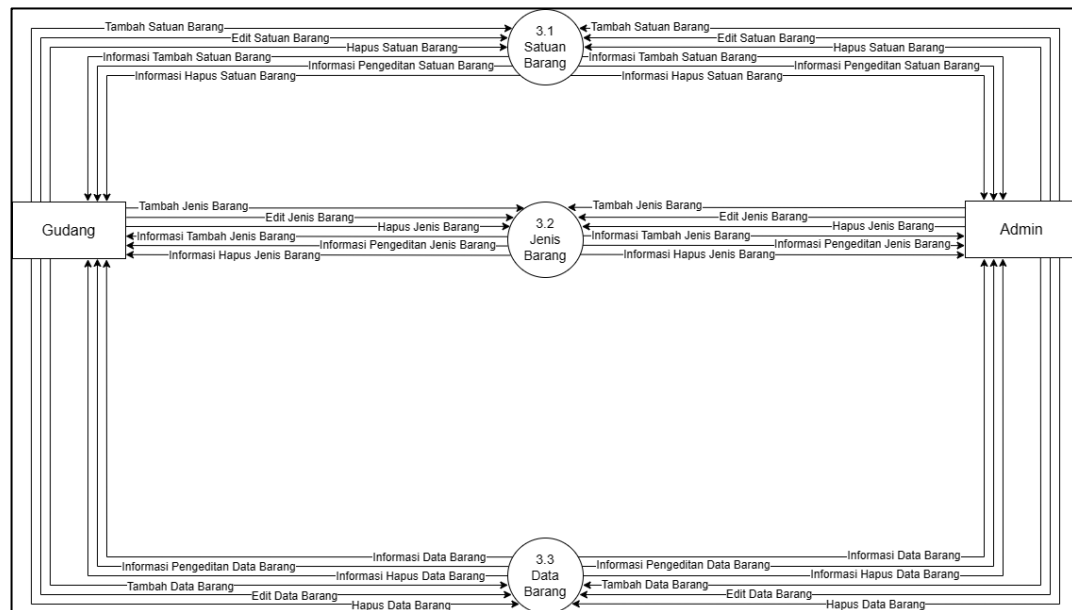
DFD Level 2 adalah hasil *breakdown* DFD Level 1 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 2 dalam proses mengolah *supplier* dari sistem yang diusulkan:



Gambar 3. 5 DFD Level 2 Mengolah *Supplier*

3.6.1.4 DFD Level 2 Mengolah Master Barang Yang Diusulkan

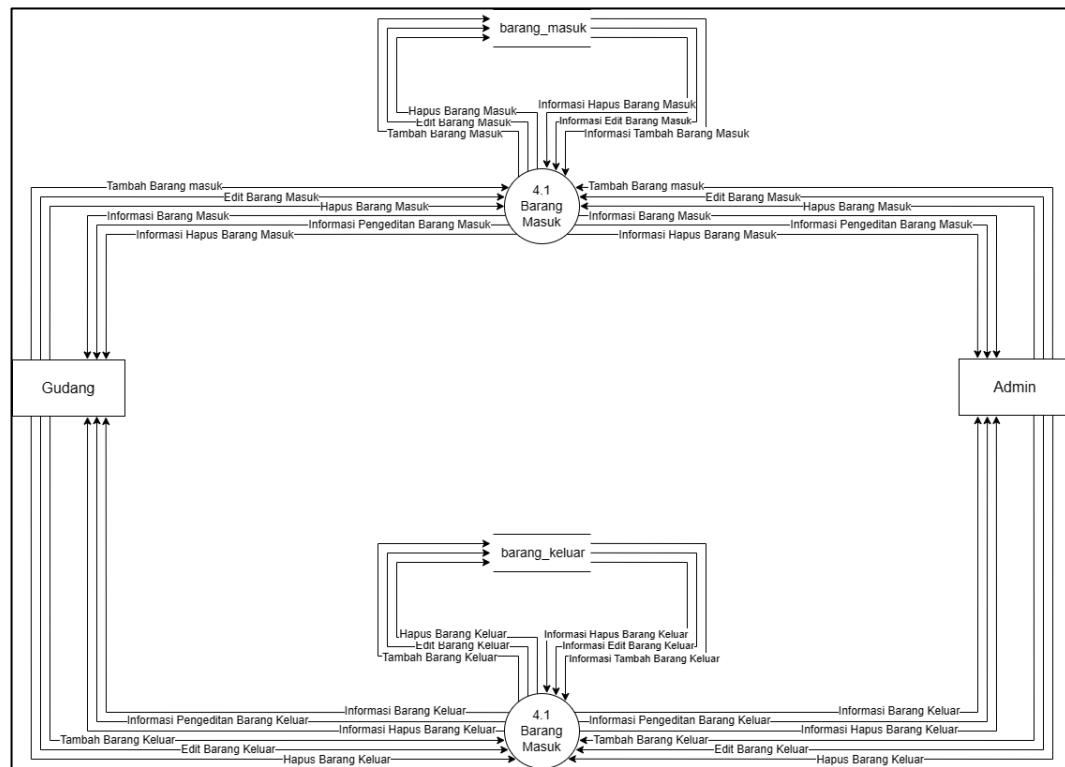
DFD Level 2 adalah hasil *breakdown* DFD Level 1 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 2 dalam proses mengolah master barang dari sistem yang diusulkan:



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

3.6.1.5 DFD Level 2 Mengolah Transaksi Yang Diusulkan

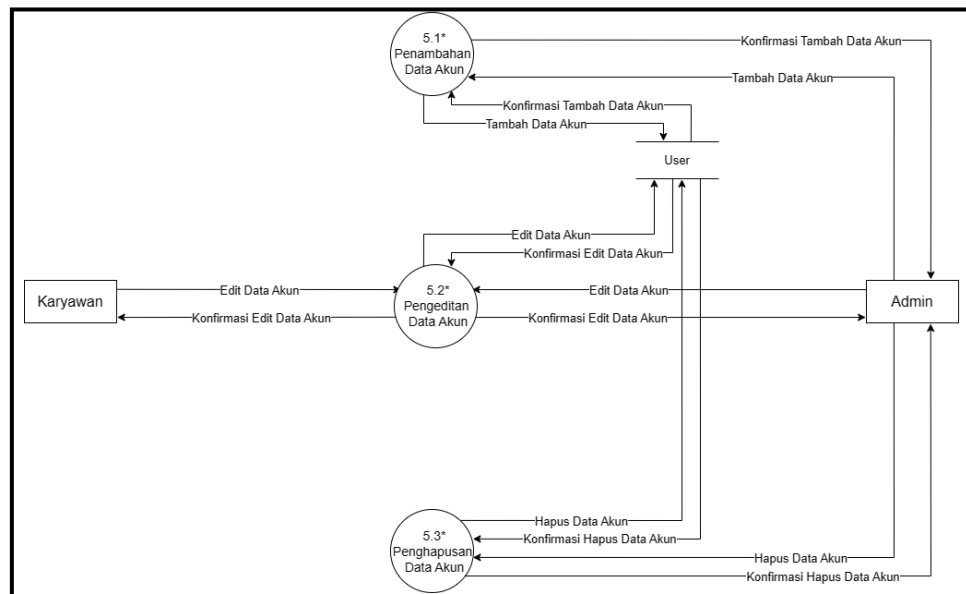
DFD Level 2 adalah hasil *breakdown* DFD Level 1 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 2 dalam proses mengolah transaksi dari sistem yang diusulkan:



Gambar 3. 7 DFD Level 2 Mengolah Transaksi

3.6.1.6 DFD Level 2 Mengolah Akun Yang Diusulkan

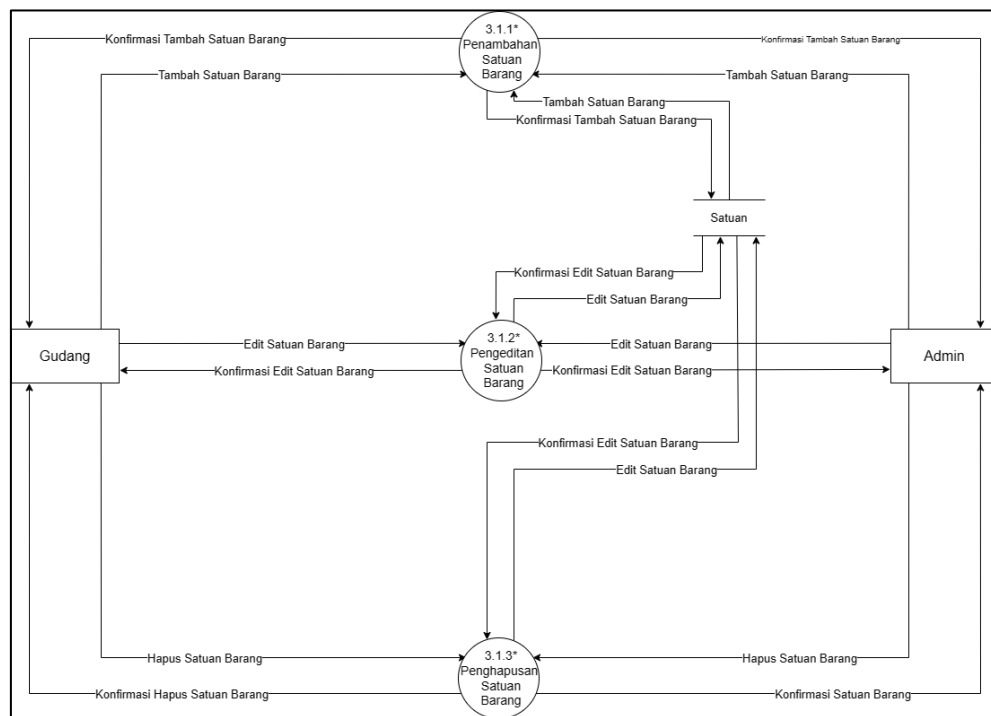
DFD Level 2 adalah hasil *breakdown* DFD Level 1 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 2 dalam proses mengolah akun dari sistem yang diusulkan:



Gambar 3. 8 DFD Level 2 Mengolah Akun

3.6.1.7 DFD Level 3 Mengolah Satuan Barang Yang Diusulkan

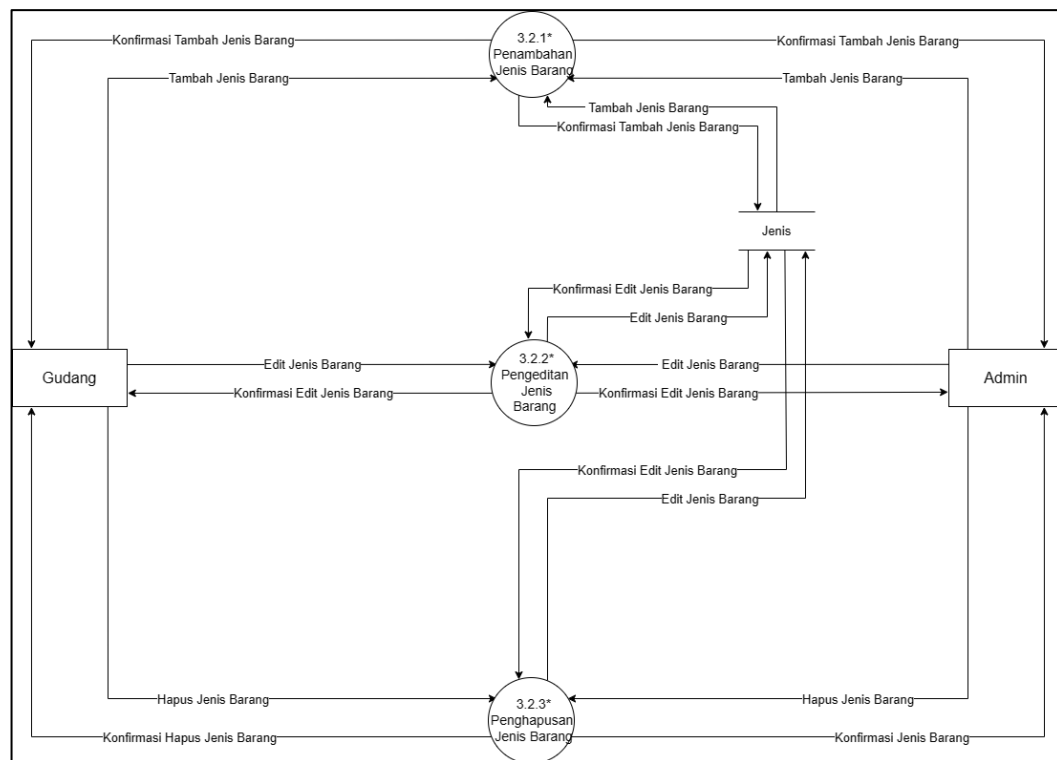
DFD Level 3 adalah hasil *breakdown* DFD Level 2 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 3 dalam proses mengolah Satuan Barang dari sistem yang diusulkan:



Gambar 3. 9 DFD Level 3 Mengolah Satuan Barang

3.6.1.8 DFD Level 3 Mengolah Jenis Barang Yang Diusulkan

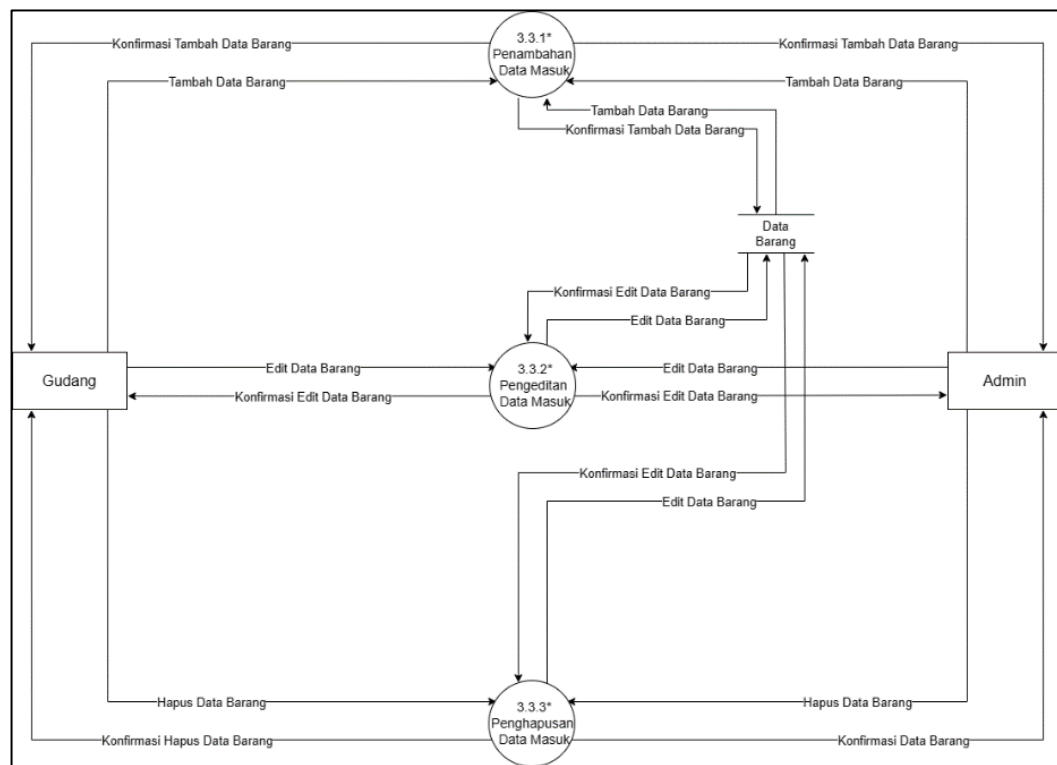
DFD Level 3 adalah hasil *breakdown* DFD Level 2 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 3 dalam proses mengolah Jenis Barang dari sistem yang diusulkan:



Gambar 3. 10 DFD Level 3 Mengolah Jenis Barang

3.6.1.9 DFD Level 3 Mengolah Data Barang Yang Diusulkan

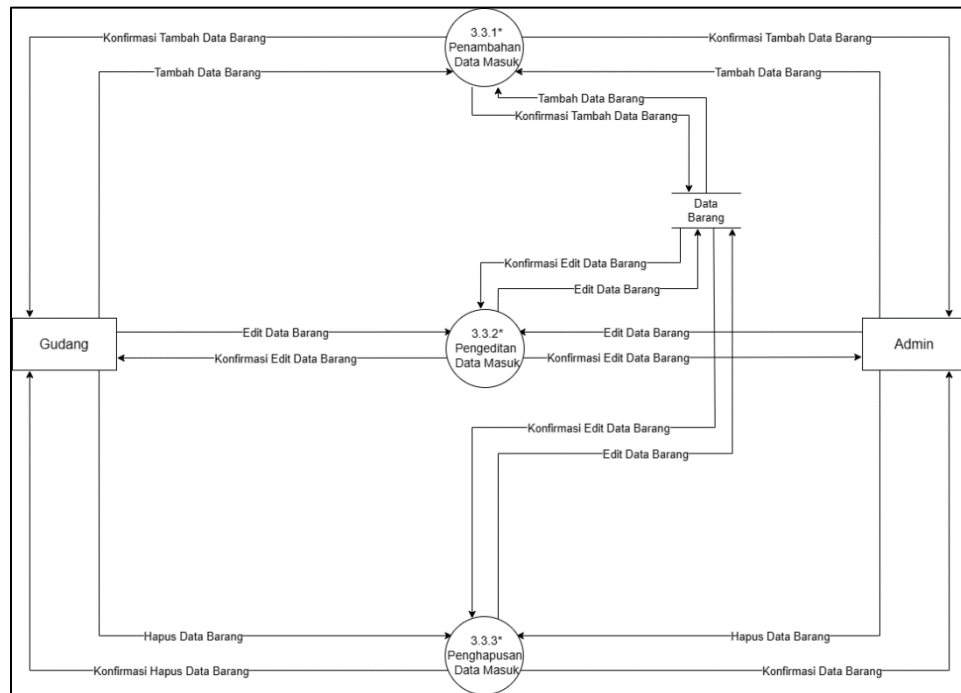
DFD Level 3 adalah hasil *breakdown* DFD Level 2 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 3 dalam proses mengolah Data Barang dari sistem yang diusulkan:



Gambar 3.11 DFD Level 3 Mengolah Data Barang

3.6.1.10 DFD Level 3 Mengolah Transaksi Barang Masuk Yang Diusulkan

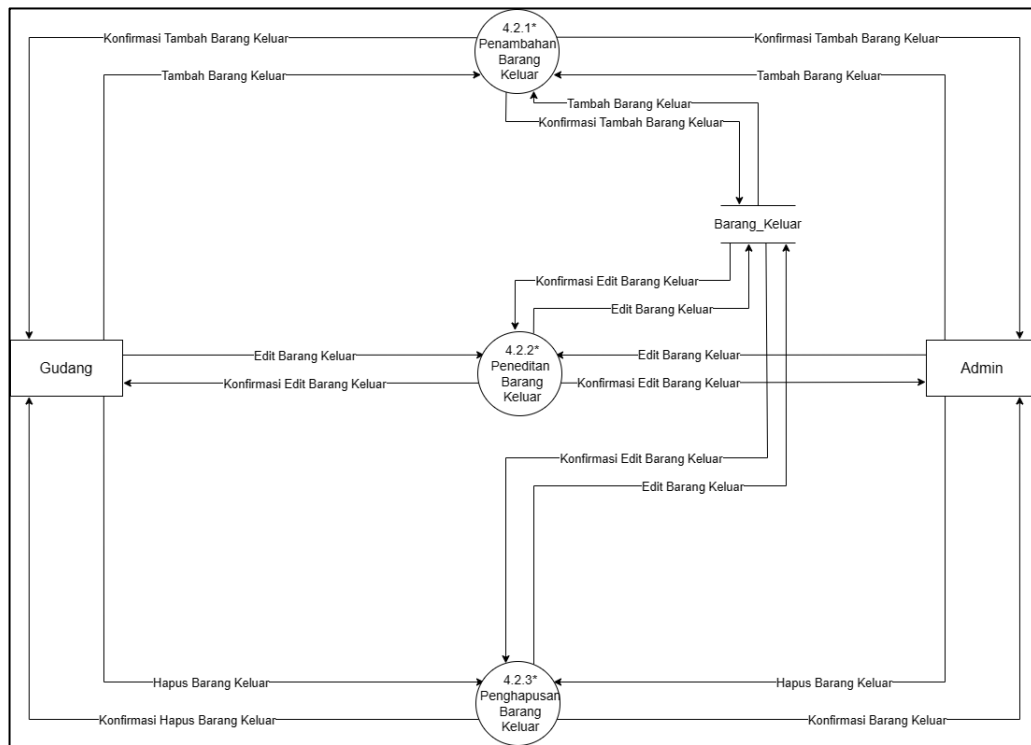
DFD Level 3 adalah hasil *breakdown* DFD Level 2 pada modul - modul yang harus di *breakdown* lebih detail tergantung tingkat kedetailan modul tersebut (A.S., Rosa dan Shalahuddin, 2016). Berikut ini adalah DFD Level 3 dalam proses mengolah Transaksi Barang Masuk dari sistem yang diusulkan:



Gambar 3. 12 DFD Level 3 Mengolah Data Barang

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

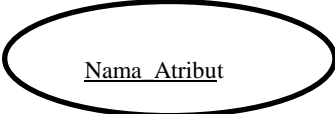

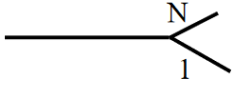
3.6.2 Data Pemodelan Data Sistem

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

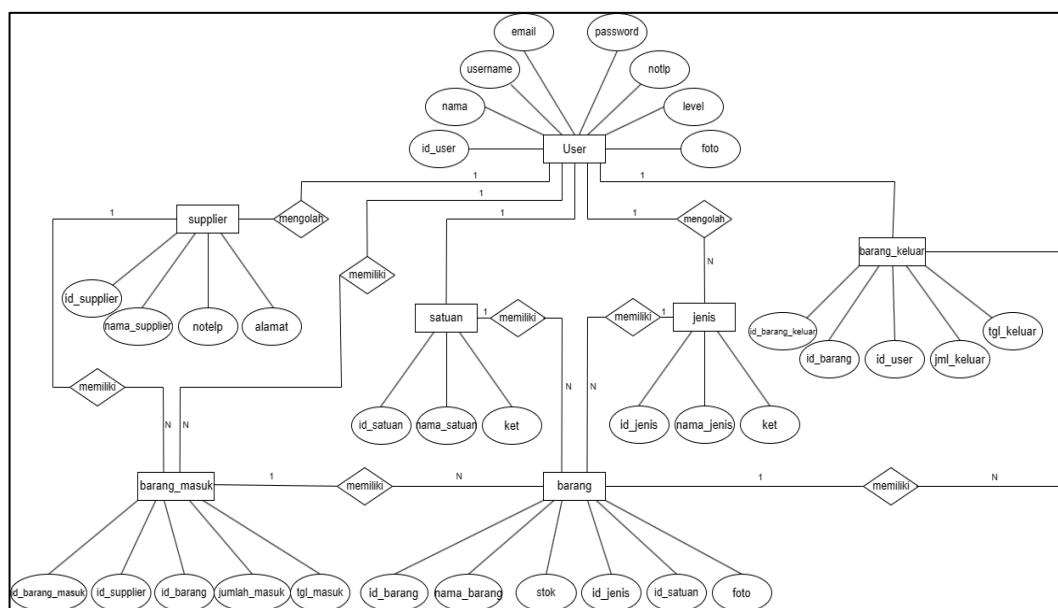
Tabel 3.3 Simbol ERD Notasi Chen

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

Simbol	Keterangan
<div style="border: 1px solid black; width: 100px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> Nama_entitas </div>	<i>Entitas</i> merupakan data inti yang akan disimpan, bakal tabel pada basis data.
<div style="border: 1px solid black; border-radius: 50%; width: 100px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> Nama_Atribut </div>	<i>Atribut</i> adalah kolom data yang butuh disimpan dalam suatu entitas.

Tabel 3.3 Simbol ERD Notasi Chen	
Simbol	Keterangan
	<p>Atribut kunci primer adalah kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.</p>
	<p>Relasi yang menghubungkan antar entitas dan biasanya diawali dengan kata kerja.</p>
	<p>Asosiasi 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.</p>

Berikut ini adalah pemodelan data sistem yang diusulkan:



Gambar 3.14 ERD

Database yang digunakan adalah sebagai berikut :

a. Tabel *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 5.3 Spesifikasi Tabel Data *User*

Field	Type	Size	Keterangan
<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>		
<i>foto</i>	<i>Varchar</i>	256	

b. Tabel *Supplier*

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

Tabel 3.6 Spesifikasi Tabel Data *Supplier*

Field	Type	Size	Keterangan
<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. Tabel Satuan

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

Tabel 3.7 Spesifikasi Tabel Data satuan

Field	Type	Size	Keterangan
<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. Tabel jenis

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

Tabel 3.8 Spesifikasi Tabel Data Nama Jenis

<i>Field</i>	<i>Type</i>	<i>Size</i>	Keterangan
id_jenis	<i>Int</i>	11	<i>Primary Key</i>
nama_jenis	<i>Varchar</i>	50	
ket	<i>Text</i>		

e. Tabel Barang

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

Tabel 3.9 Spesifikasi Tabel Barang

<i>Field</i>	<i>Type</i>	<i>Size</i>	Keterangan
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. Tabel 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 3.10 Spesifikasi Tabel Data Barang Masuk

<i>Field</i>	<i>Type</i>	<i>Size</i>	Keterangan
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. Tabel 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.

Tabel 3.11 Spesifikasi Tabel Data Jenis

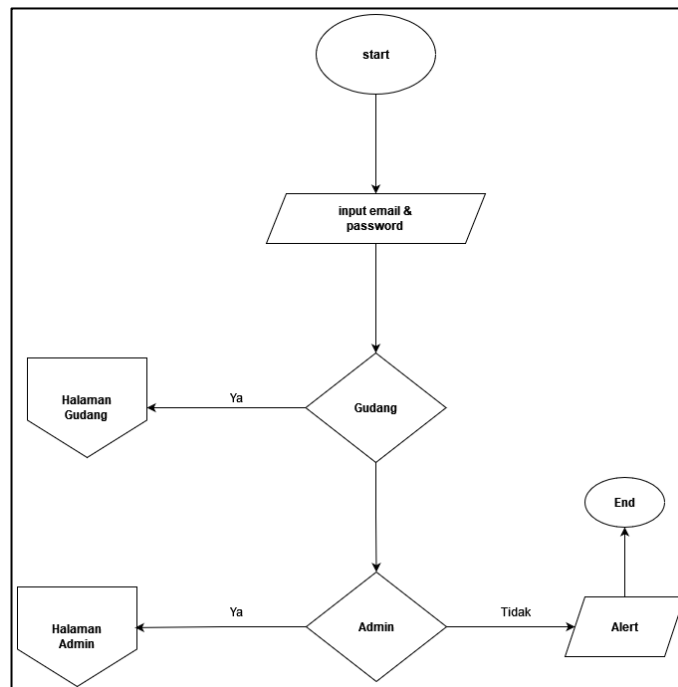
Field	Type	Size	Keterangan
id_barang_keluar	<i>Int</i>	11	<i>Primary Key</i>
id_barang	<i>Varchar</i>	11	<i>Forgein key</i>
id_user	<i>Varchar</i>	11	<i>Forgein key</i>
jumlah_keluar	<i>Varchar</i>	10	
tgl_keluar	<i>Varchar</i>	15	

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 User*

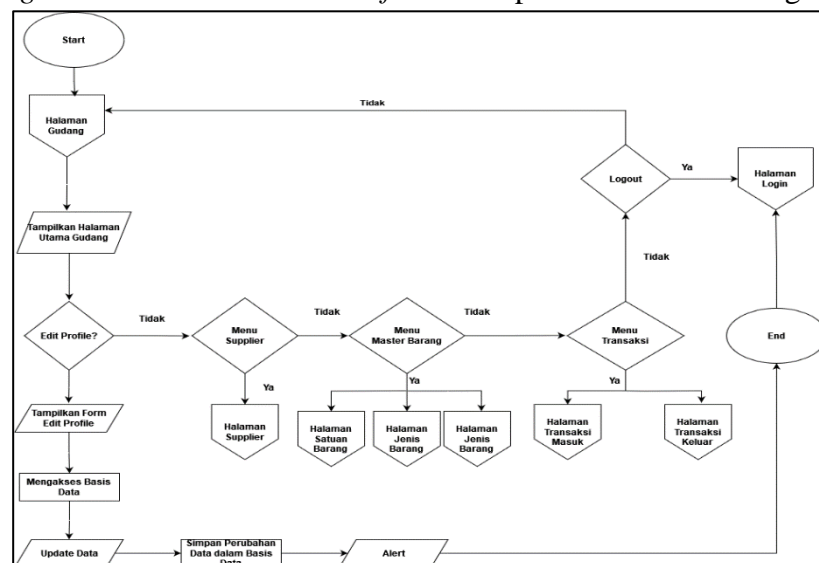
Pada saat aplikasi dijalankan aplikasi akan meminta *user* memasukkan *email* dan *password*. Apabila email dan password yang dimasukkan benar maka aplikasi akan mengkonfirmasi apakah *user* sebagai admin atau sebagai karyawan dan jika sudah terkonfirmasi *user* akan diarahkan ke halaman yang sesuai. 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 User

3.6.3.2 Flowchart Halaman Gudang

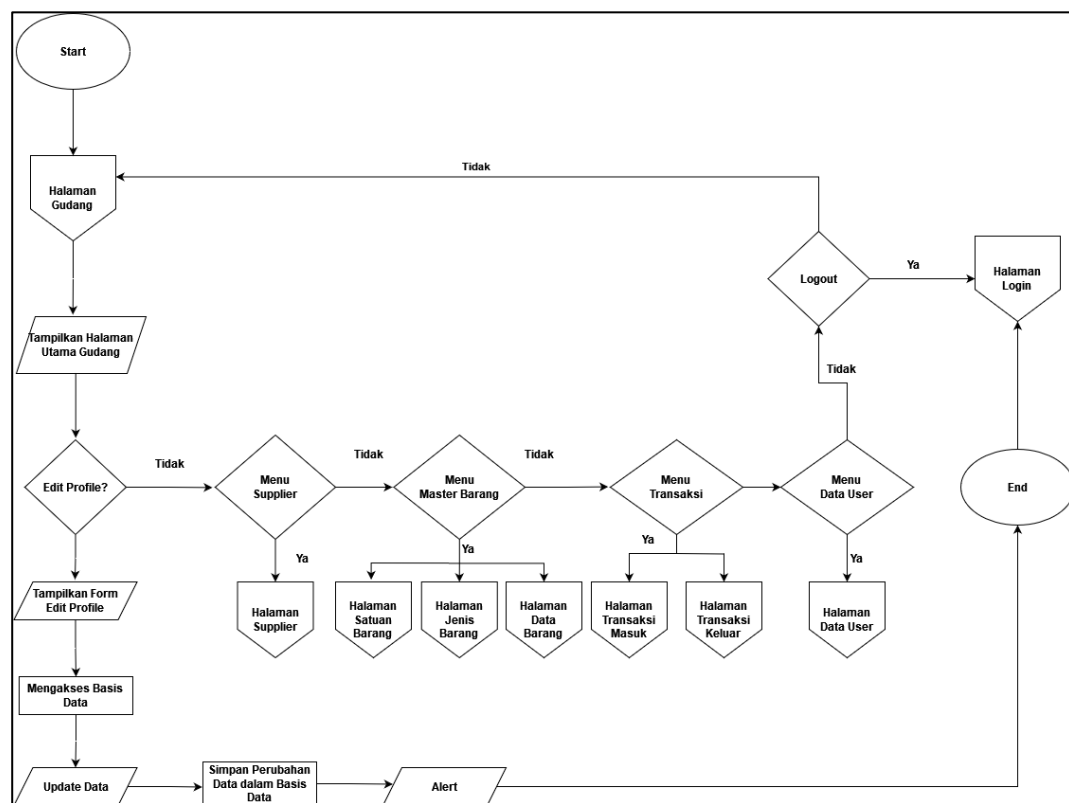
Pada halaman Gudang, Karyawan dapat mengolah data *supplier*, master barang, dan transaksi. Apabila karyawan memilih *logout* maka karyawan tersebut akan diarahkan menuju halaman *login* kembali. Berikut adalah *flowchart* pada halaman Gudang:



Gambar 3.16 Flowchart Halaman Karyawan

3.6.3.3 Flowchart Halaman Admin

Pada halaman admin, admin dapat mengolah data *supplier*, *master* barang, transaksi dan mendapatkan akses untuk mengolah data akun *user*. Apabila admin memilih *logout* maka tersebut akan diarahkan menuju halaman *login* kembali. Berikut adalah *flowchart* pada halaman admin:

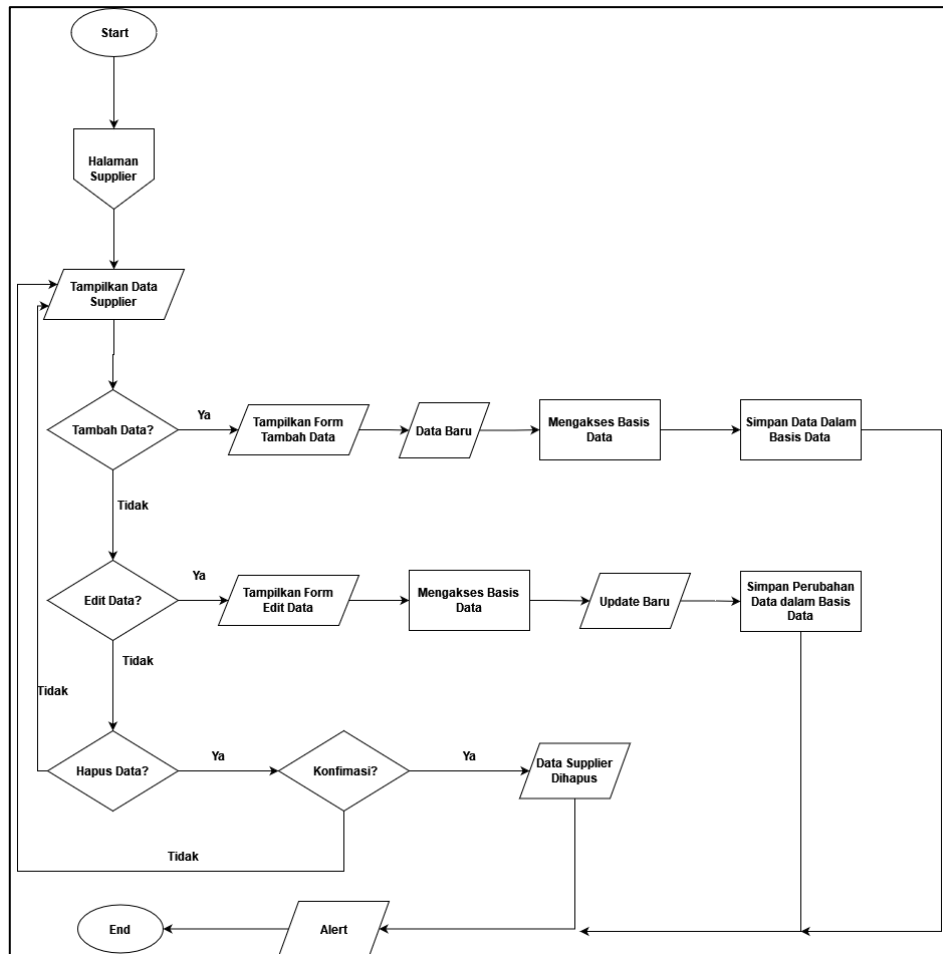


Gambar 3. 17 Flowchart Halaman Admin

3.6.3.4 Flowchart Halaman Menu Supplier

Pada menu *supplier*, admin dan karyawan bagian gudang dapat menambahkan, mengubah, dan menghapus data *supplier*. Jika data *supplier* telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel *supplier*. Jika data dihapus dan telah terkonfirmasi maka data akan dihapus dalam *database*. Seluruh *action* yang telah dilakukan akan menampilkan

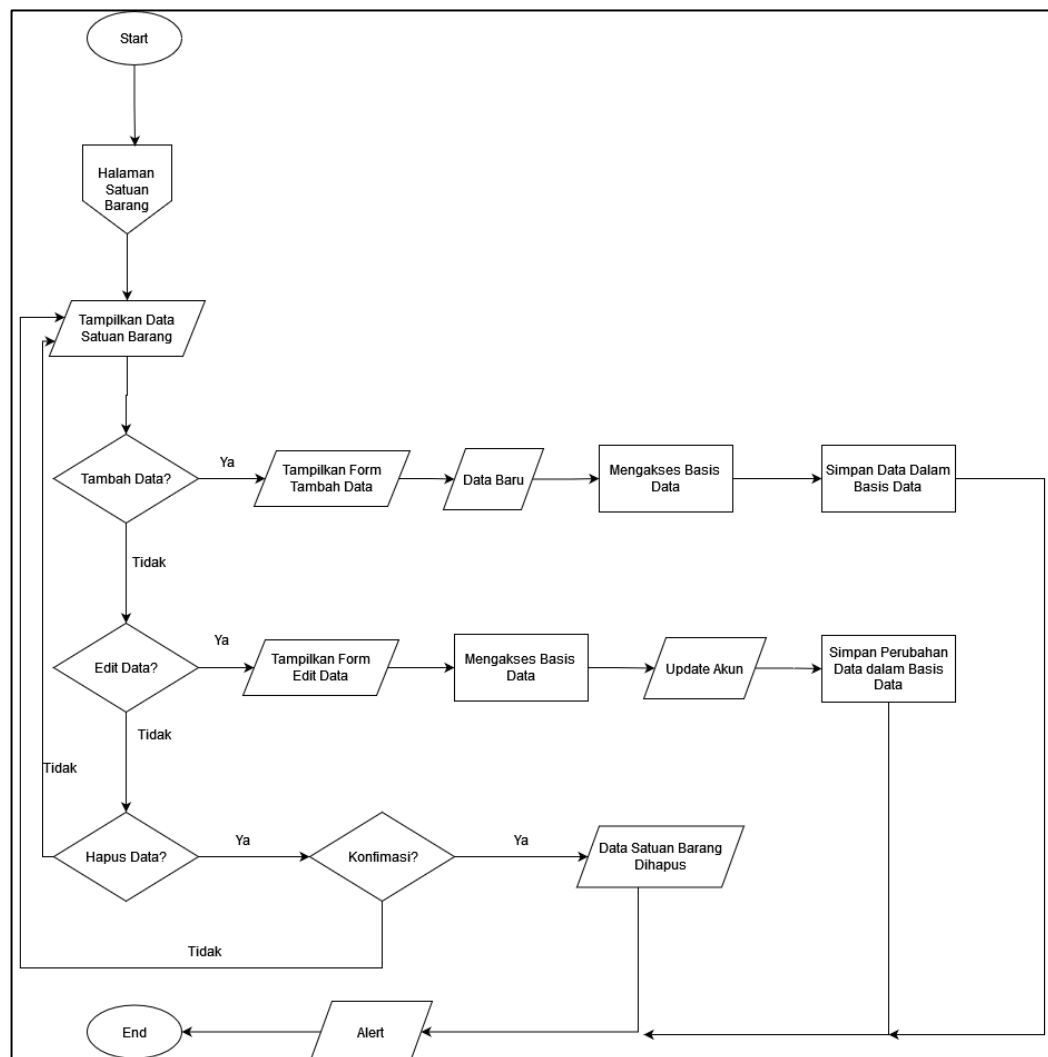
alert apabila *action* yang dilakukan berhasil.



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

3.6.3.5 *Flowchart* Halaman Satuan Barang

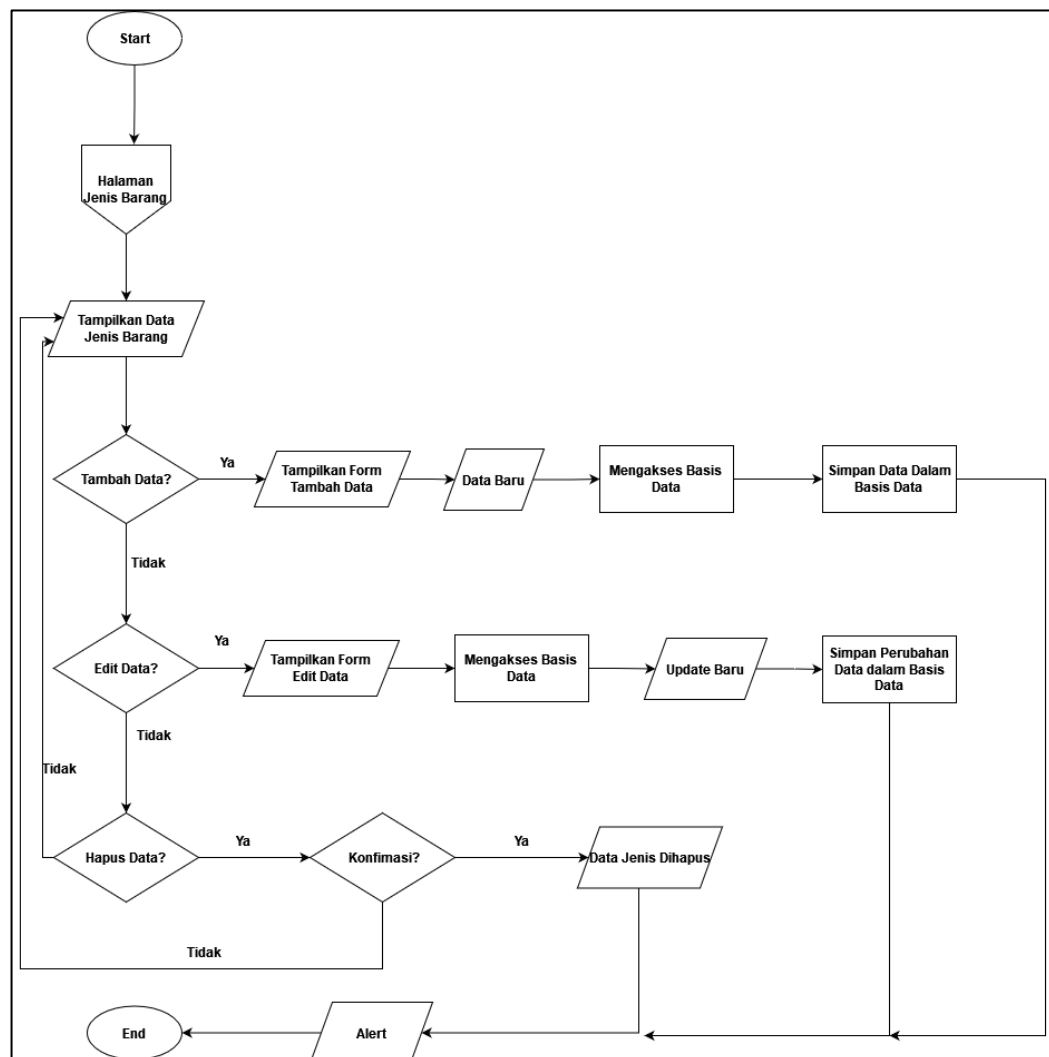
Pada menu satuan barang, admin dan karyawan bagian gudang dapat menambahkan, mengubah, dan menghapus data satuan barang. Jika data satuan barang telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel satuan. Jika data dihapus dan telah terkonfirmasi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



Gambar 3.19 Flowchart Halaman Menu Satuan Barang

3.6.3.6 Flowchart Halaman Jenis Barang

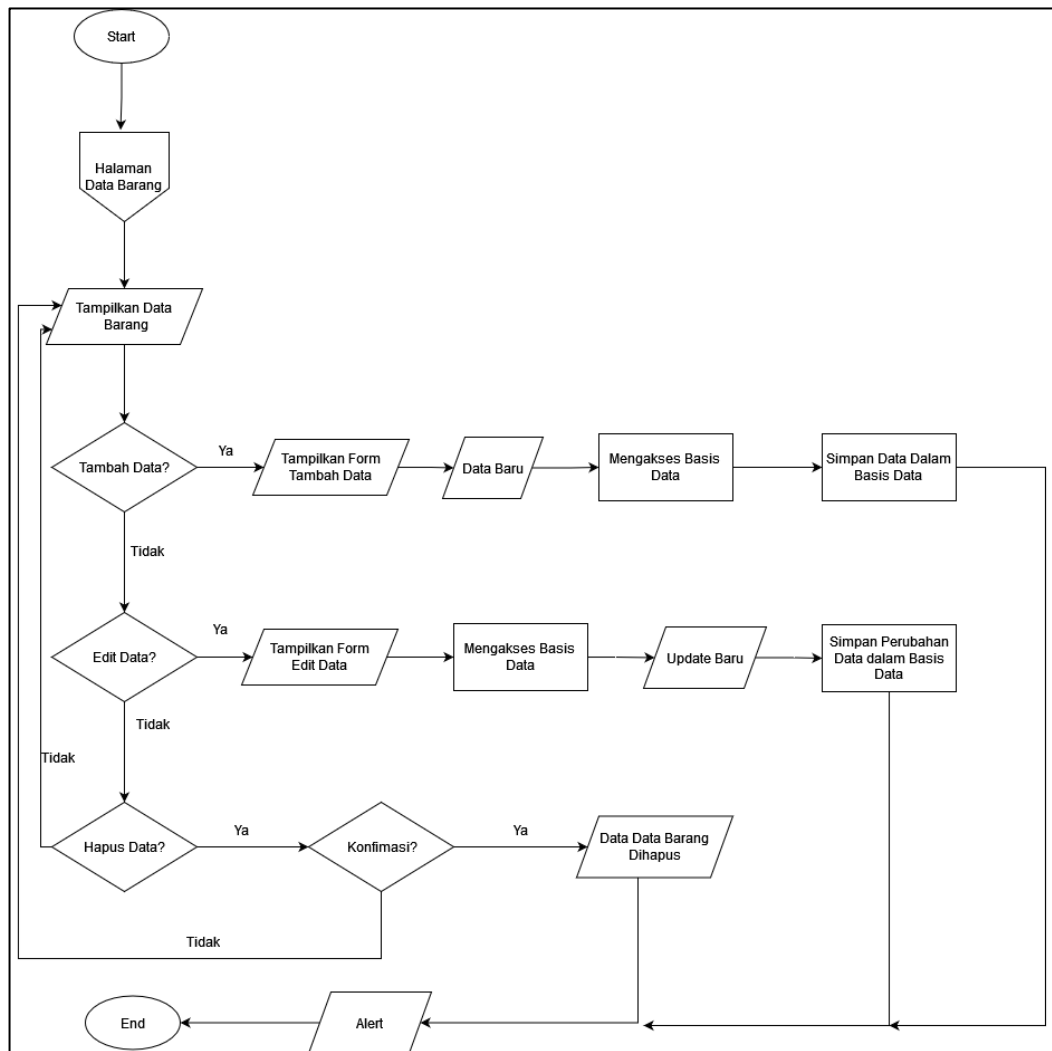
Pada menu jenis barang, admin dan karyawan bagian gudang dapat menambahkan, mengubah, dan menghapus data jenis barang. Jika data jenis barang telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel jenis. Jika data dihapus dan telah terkonfirmasi maka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



Gambar 3. 20 *Flowchart* Halaman Menu Jenis Barang

3.6.3.7 *Flowchart* Halaman Data Barang

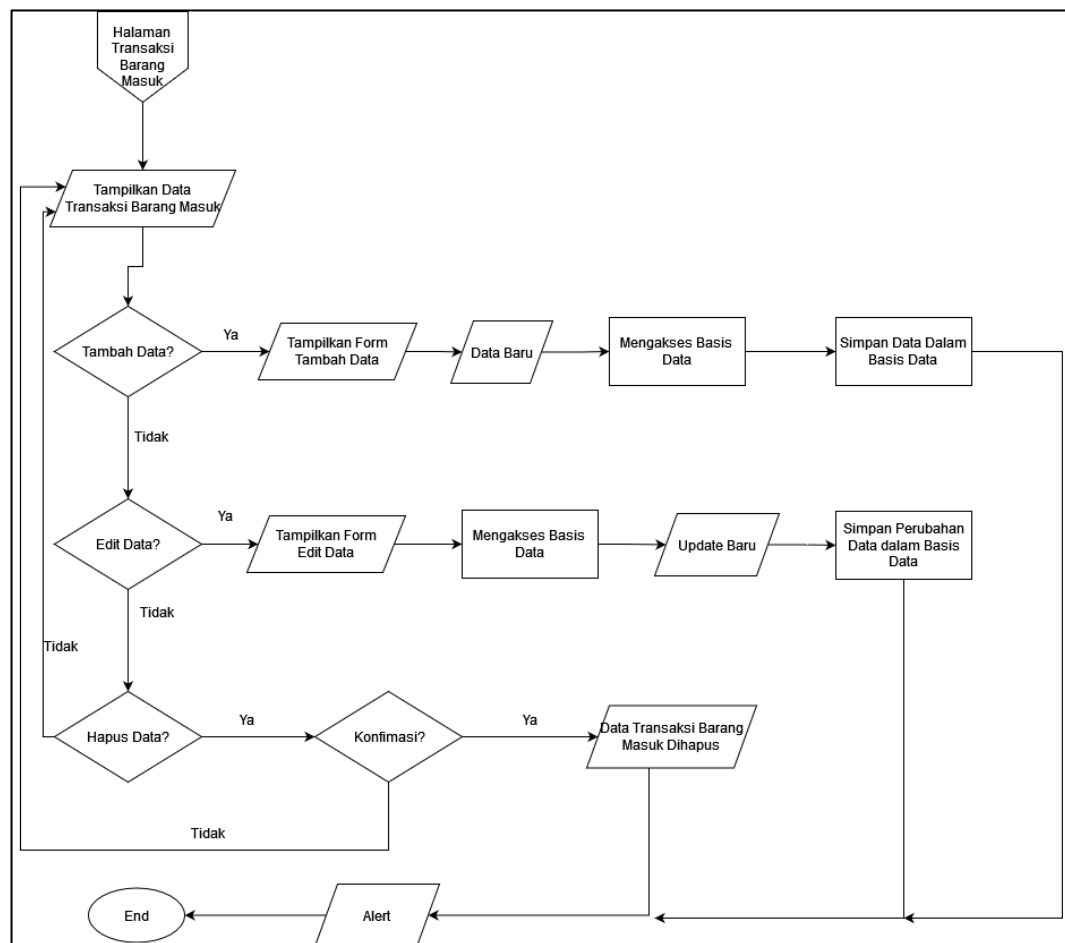
Pada menu data barang, admin dan karyawan bagian gudang yang dapat menambahkan, mengubah, dan menghapus data barang. Jika data barang telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel barang. Jika data dihapus dan telah terkonfirmasi maka data akan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



Gambar 3. 21 *Flowchart* Halaman Data Barang

3.6.3.8 *Flowchart* Halaman Transaksi Barang Masuk

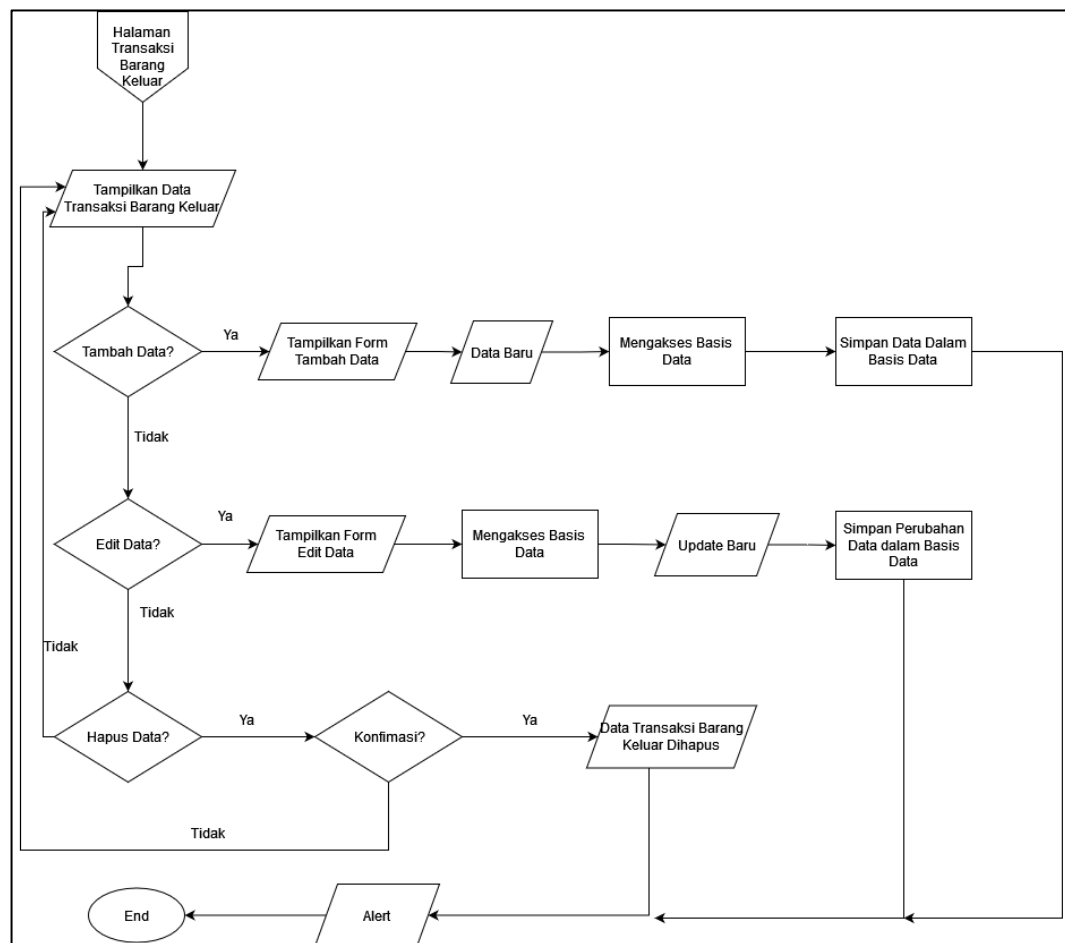
Pada menu akun transaksi barang masuk, admin dan karyawan bagian gudang yang dapat menambahkan, mengubah, dan menghapus data transaksi barang masuk. Jika data transaksi barang masuk telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel *transaksi_masuk*. 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. 22 Flowchart Halaman Menu Transaksi masuk

3.6.3.9 Flowchart Pencarian Transaksi Barang Keluar

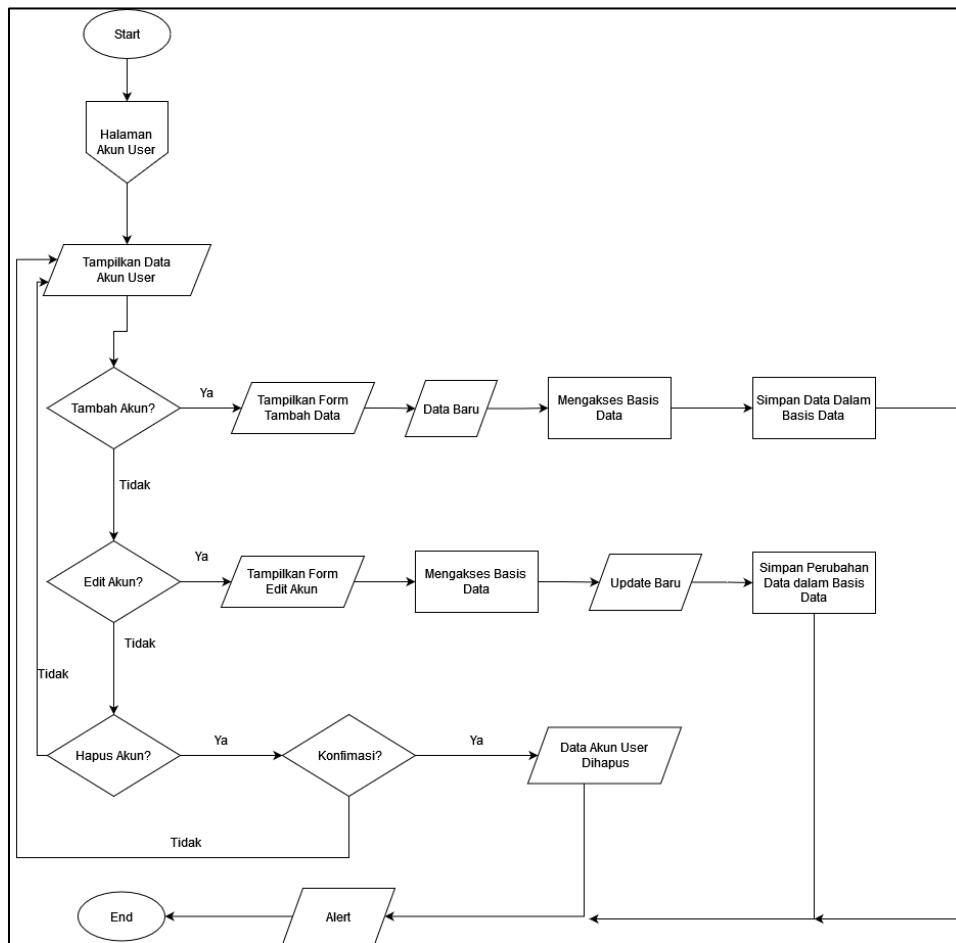
Pada menu akun transaksi barang keluar, admin dan karyawan bagian gudang yang dapat menambahkan, mengubah, dan menghapus data transaksi barang keluar. Jika data transaksi barang keluar telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel *transaksi_masuk*. 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. 23 Flowchart Halaman Menu Transaksi keluar

3.6.3.10 Flowchart Halaman Akun User

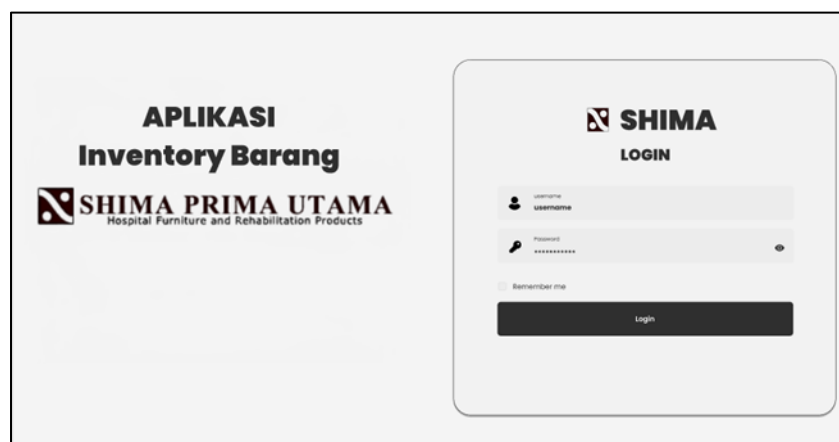
Pada menu akun *user*, hanya admin yang dapat menambahkan, mengubah, dan menghapus data *user*. Jika data *user* telah diinput atau perubahan yang dilakukan disimpan, maka data akan disimpan dalam *database* pada tabel *user*. Jika data dihapus dan telah terkonfirmasi maka data akan dihapus dalam *database*.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



Gambar 3. 24 Flowchart Halaman Menu Akun User

3.6.4 Perancangan Interface Sistem

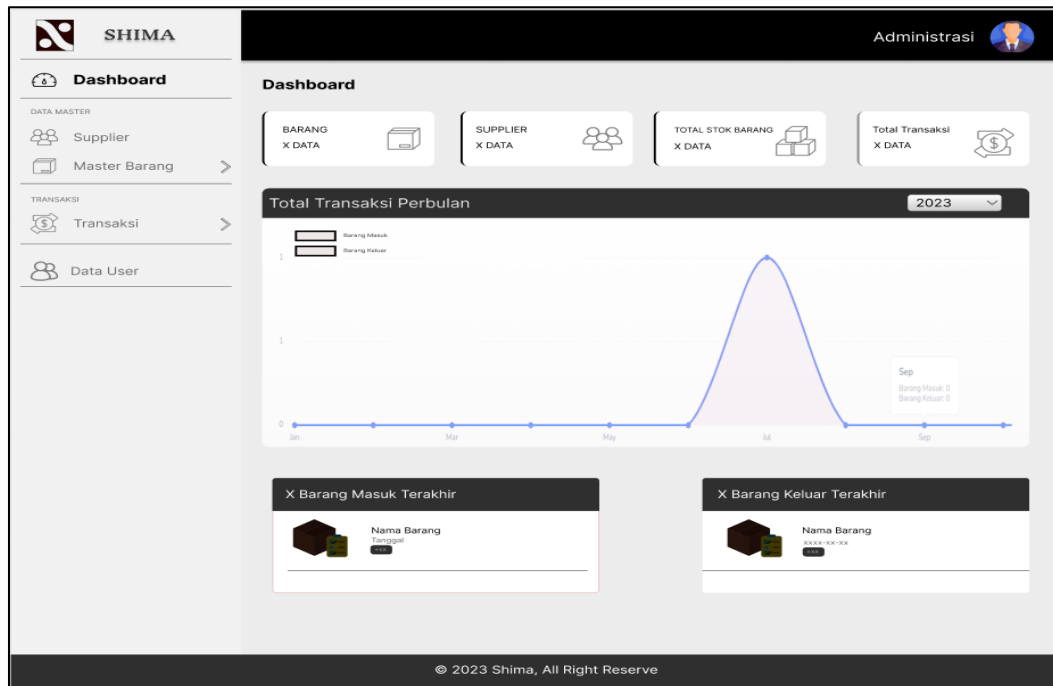
3.6.4.1 Perancangan Halaman Menu Login



Gambar 3. 25 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.

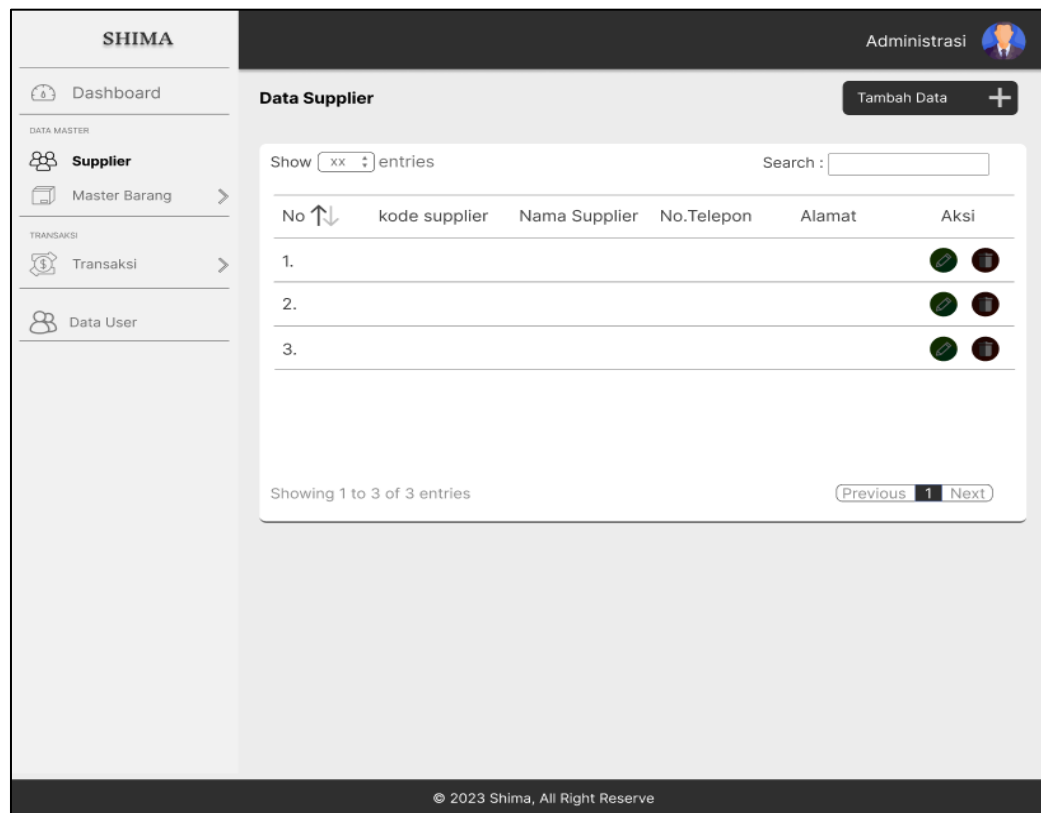
3.6.4.2 Perancangan Halaman Admin



Gambar 3. 26 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*



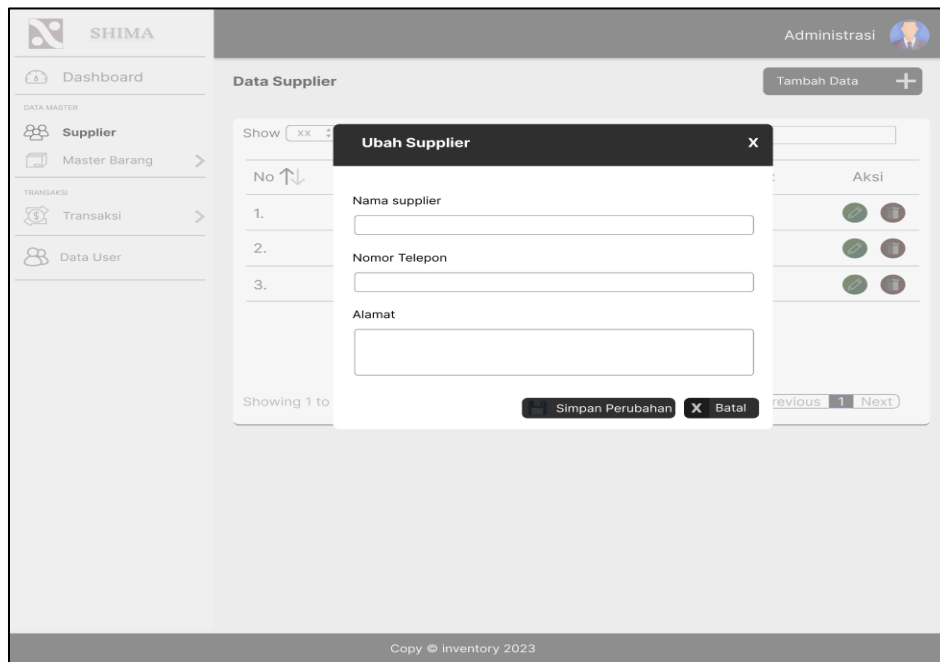
Gambar 3. 27 Halaman *Supplier*

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 screenshot displays the SHIMA web application interface. The top navigation bar includes the SHIMA logo, the title 'Administrasi', and a user profile icon. The left sidebar contains a menu with 'Dashboard', 'Supplier' (under DATA MASTER), 'Master Barang', 'Transaksi' (under TRANSAKSI), and 'Data User'. The main content area is titled 'Data Supplier' and features a 'Tambah Data' button with a plus sign. A modal window titled 'Tambah Supplier' is open, containing three input fields: 'Nama supplier', 'Nomor Telepon', and 'Alamat'. Below the modal, a table lists three suppliers with columns for 'No', 'Aksi', and 'previous 1 Next'. The table has a 'Showing 1 to' indicator at the bottom left and 'Simpan Data' and 'Batal' buttons at the bottom right. The footer of the application reads 'Copy © inventory 2023'.

Gambar 3. 28 Form Tambah Data *Supplier*

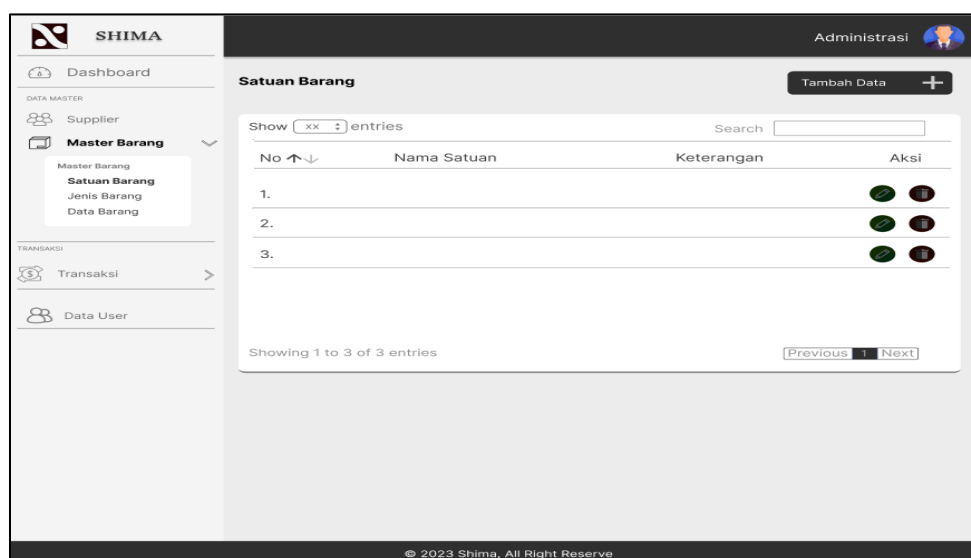
Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.



Gambar 3. 29 Form Edit Data Supplier

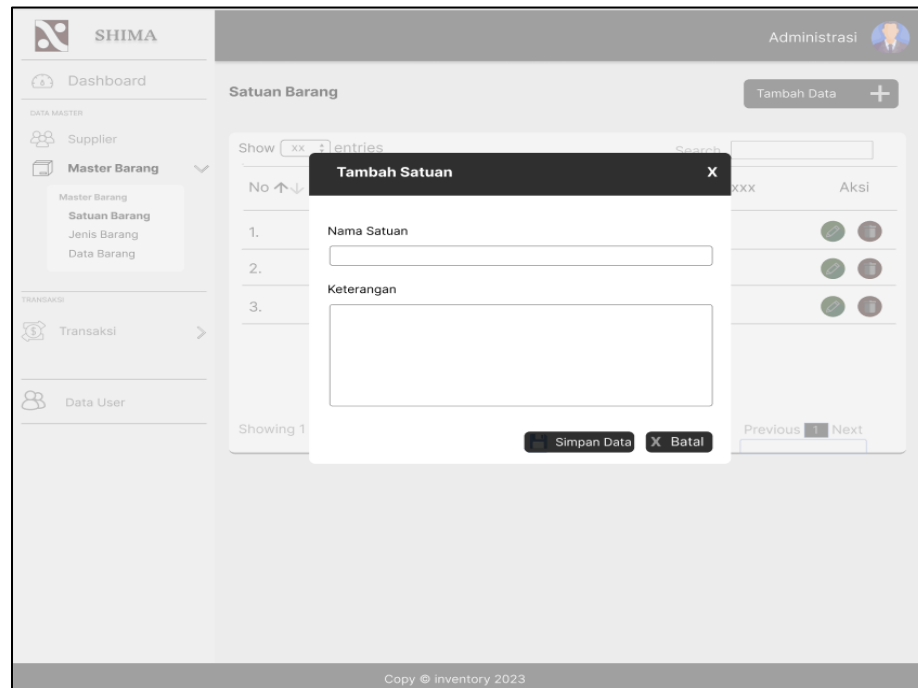
Gambar diatas menjelaskan desain form *edit* data, apabila *user* ingin mengubah data yang telah disimpan dalam *database*. Data yang telah ada sebelumnya akan ditampilkan dahulu. Kemudian *user* dapat mengisi data kembali, lalu bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

3.3.4.4 Perancangan Menu Master Barang



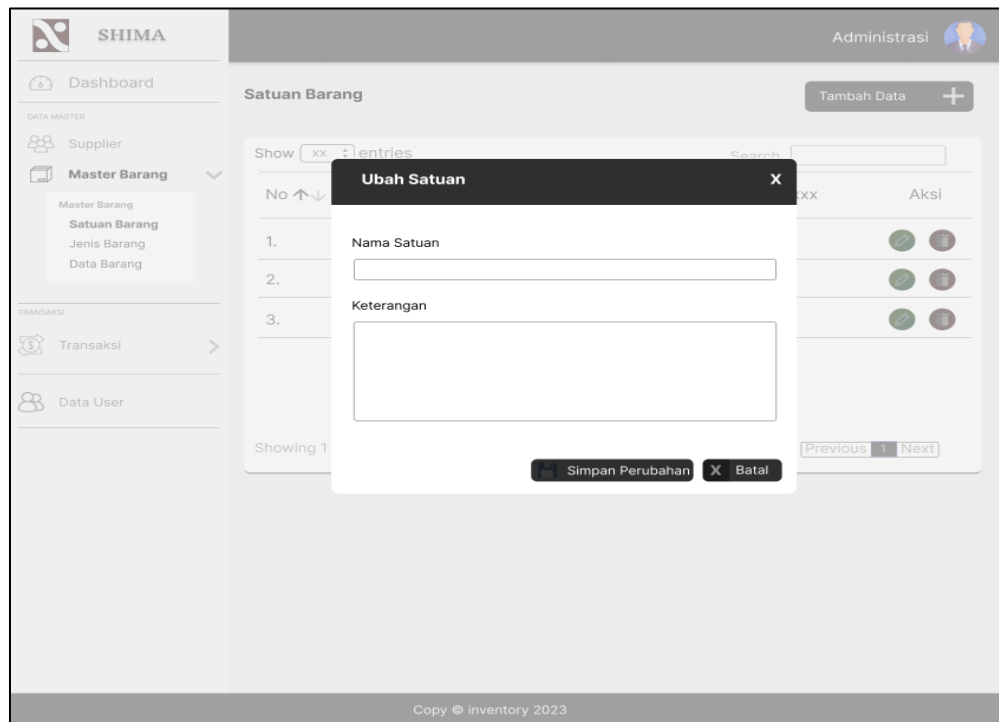
Gambar 3. 30 Tampilan Menu Satuan Barang

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.



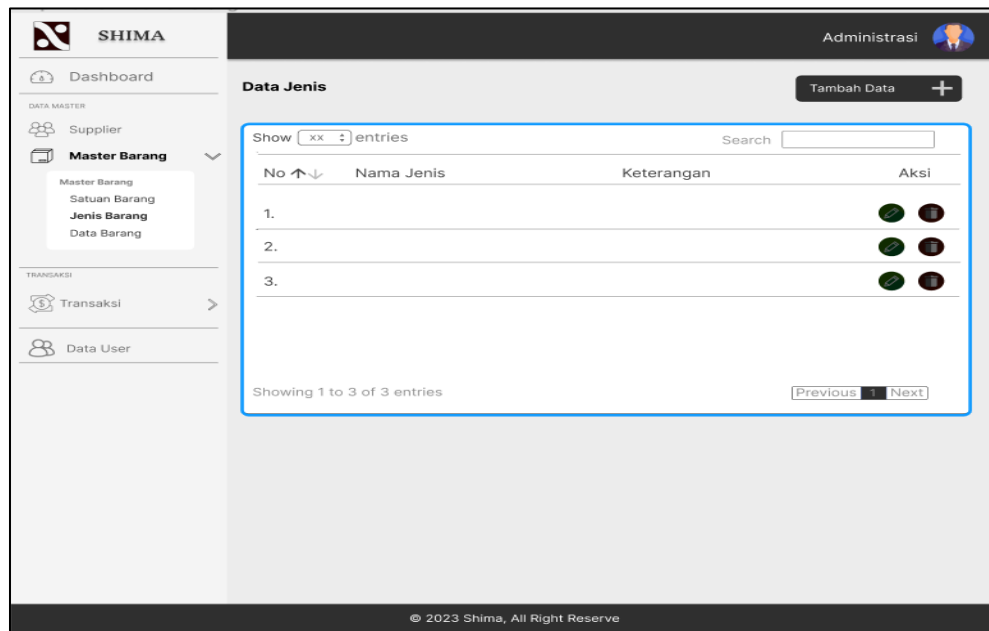
Gambar 3. 31 Form Tambah Data Satuan Barang

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*.



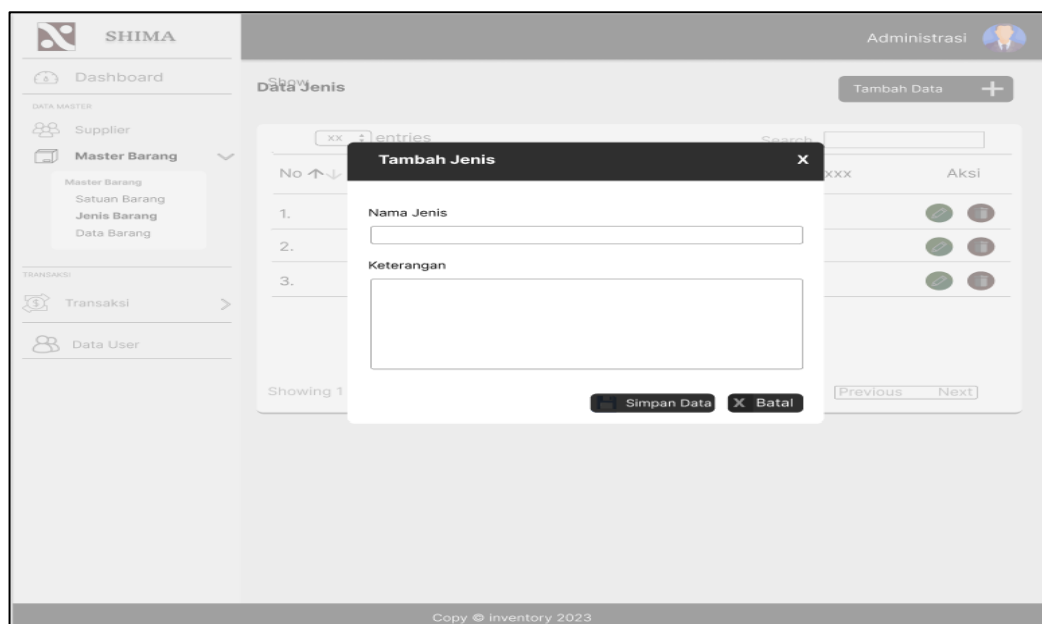
Gambar 3. 32 Form *Edit* Data Satuan Barang

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*.

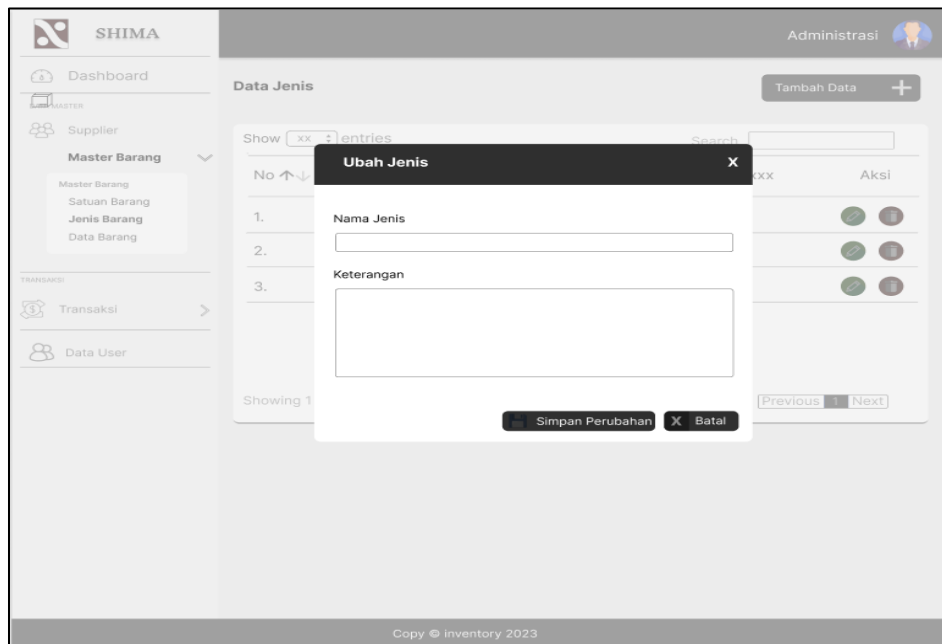


Gambar 3. 33 Tampilan Menu Jenis Barang

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.

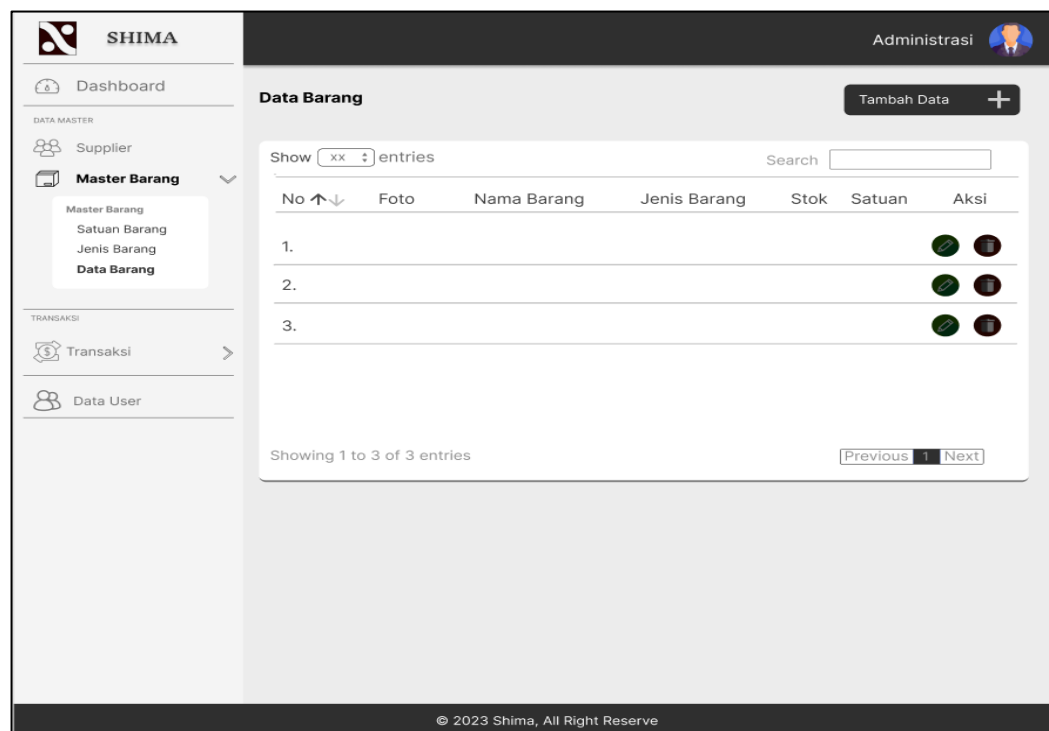


Gambar 3. 34 Form Tambah Data Jenis Barang



Gambar 3. 35 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. 36 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 screenshot displays the SHIMA web application interface. On the left is a sidebar menu with sections for 'DATA MASTER' (Supplier, Master Barang) and 'TRANSAKSI' (Transaksi, Data User). The main content area is titled 'Tambah Barang' and contains a 'Form Barang' with the following fields: 'Nama Barang' (text input), 'Stok Awal' (text input), 'Jenis Barang' (dropdown menu), and 'Satuan Barang' (dropdown menu). To the right of the form is a 'Foto' section with a 'Format' field (showing '.jpg .png .jpeg'), a preview image of a brown cube, and a 'Pilih File' button with a 'Browse' button. A 'Simpan Data' button is located at the top right of the form area. The footer of the application reads '© 2023 Shima, All Right Reserve'.

Gambar 3. 37 Form Tambah Data Barang

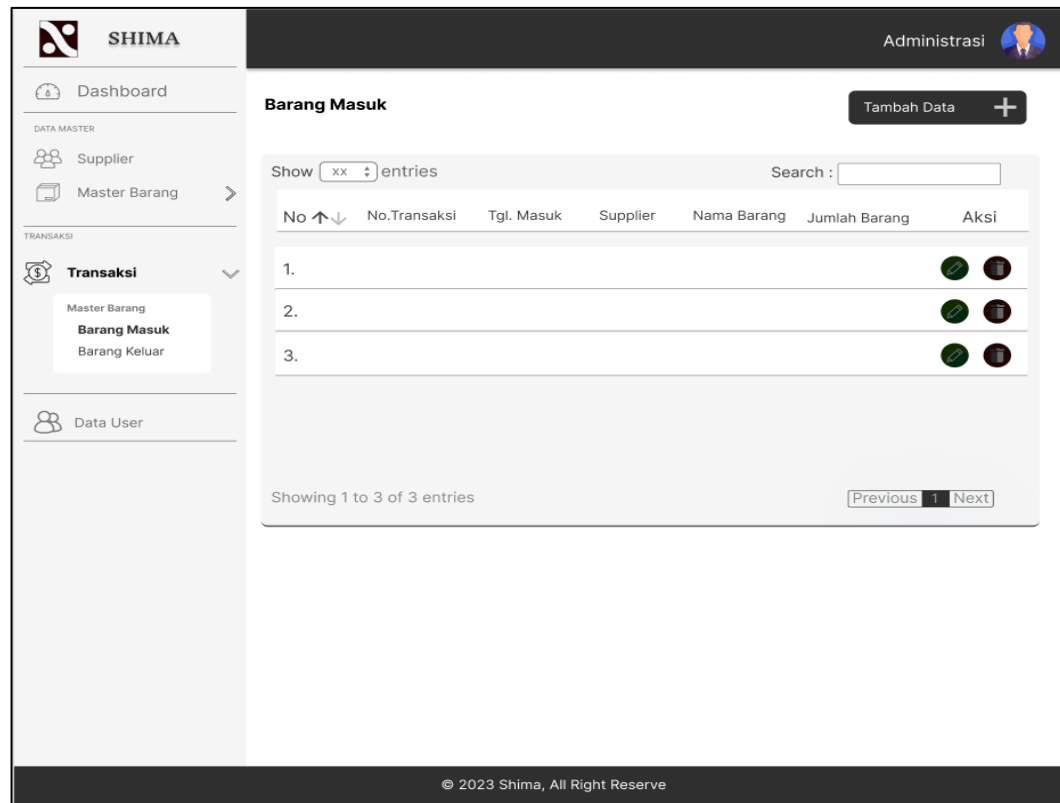
Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

The screenshot displays the 'Ubah Barang' (Edit Item) form in the SHIMA application. The interface is divided into a sidebar and a main content area. The sidebar on the left contains navigation options: Dashboard, Supplier, Master Barang (with sub-options: Master Barang, Satuan Barang, Jenis Barang, Data Barang), Transaksi, and Data User. The main content area is titled 'Ubah Barang' and features a 'Form Barang' section with the following fields: 'Nama Barang' (text input), 'Stok Awal' (text input), 'Jenis Barang' (dropdown menu with '--Pilih--'), and 'Satuan Barang' (dropdown menu with '--Pilih--'). To the right of the form is a 'Foto' section with a 'Format' field (showing '.jpg, .png, .jpeg'), a preview area with a 3D cube icon and a checklist, and a file upload section with 'Pilih File' and 'Browse' buttons. A 'Simpan Perubahan' button is positioned at the top right of the form area. The footer of the application reads '© 2023 Shima, All Right Reserve'.

Gambar 3. 38 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. 39 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.

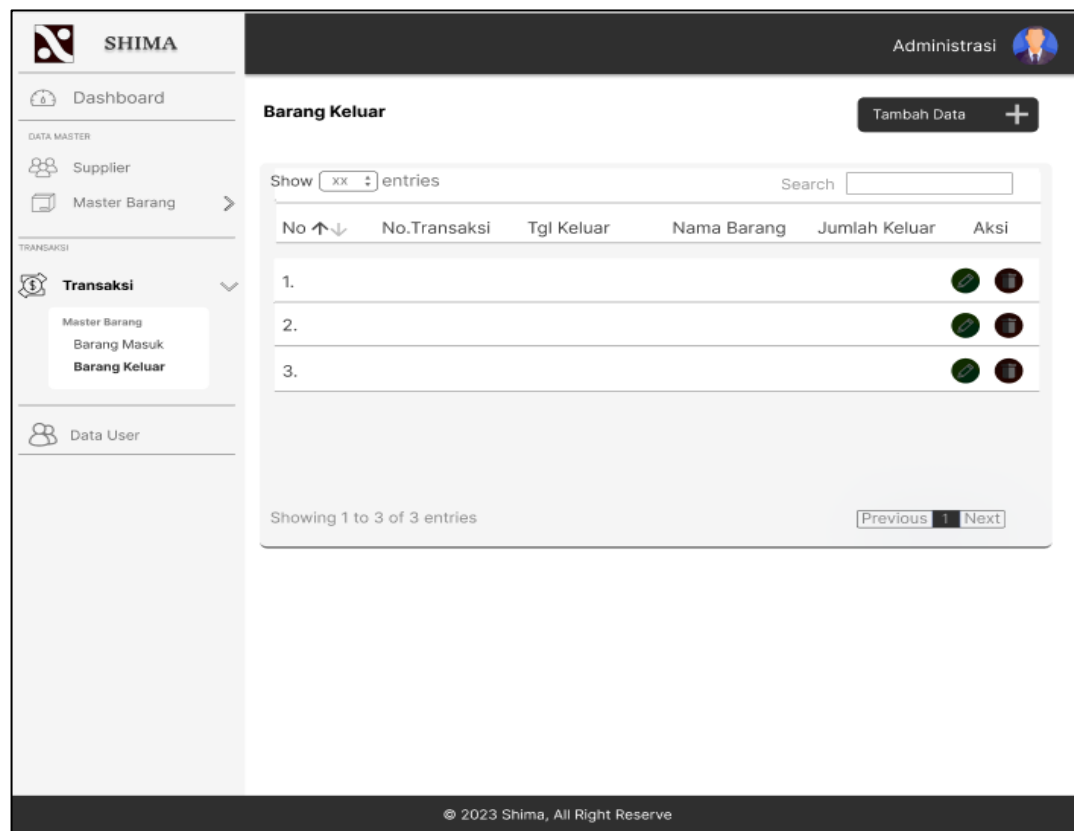
The screenshot displays the 'Tambah Barang Masuk' interface. On the left is a sidebar with navigation options: Dashboard, DATA MASTER (Supplier, Master Barang), TRANSAKSI (Transaksi, Master Barang, **Barang Masuk**, Barang Keluar), and Data User. The main content area features a 'Form Barang' with the following fields: ID Barang (text input), Tanggal Masuk (text input), Nama Barang (dropdown menu), Supplier (dropdown menu), and Jumlah Barang (text input with a spinner). To the right is a 'Preview' section showing a 3D box icon and two input fields for 'Nama Barang' and 'Stok Barang'. A 'Simpan Data' button is located at the top right of the main content area. The footer contains the text '© 2023 Shima, All Right Reserve'.

Gambar 3. 40 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*.

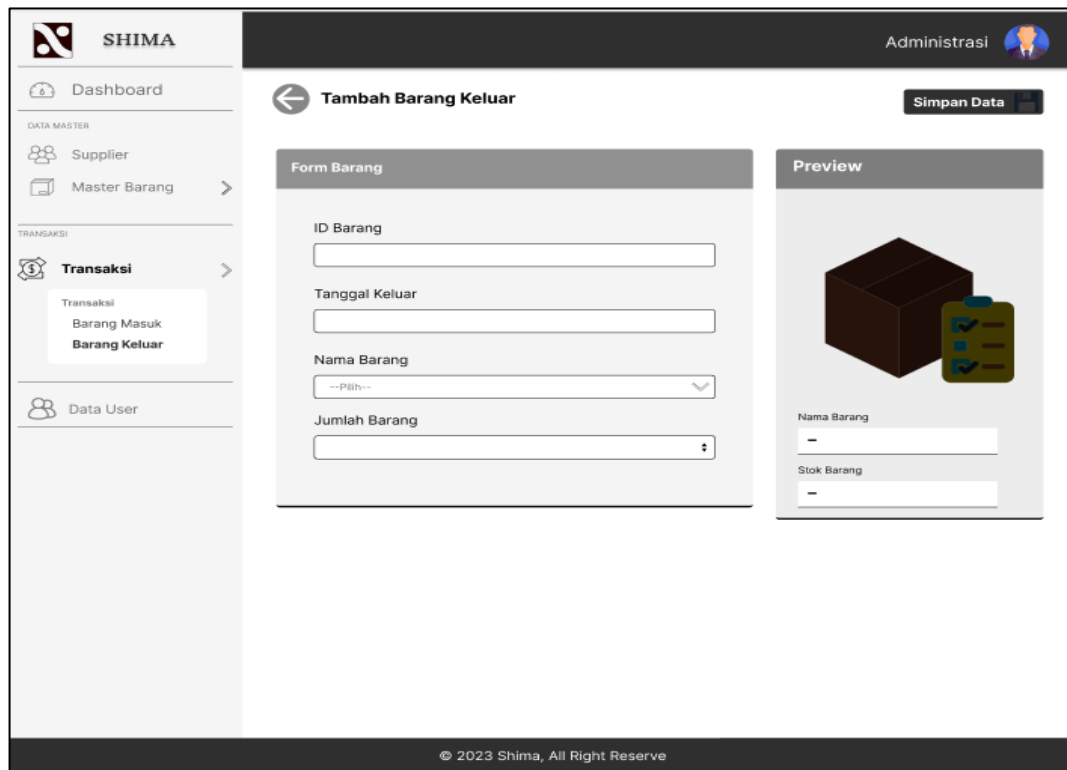
The screenshot displays the 'Ubah Barang Masuk' interface. The layout is identical to the 'Tambah Barang Masuk' form, but the 'Simpan Data' button is replaced by a 'Simpan Perubahan' button. The 'Form Barang' and 'Preview' sections remain the same. The footer contains the text '© 2023 Shima, All Right Reserve'.

Gambar 3. 41 Form *Edit* Data Transaksi Barang Masuk



Gambar 3. 42 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 screenshot displays the SHIMA web application interface. The top navigation bar includes the SHIMA logo, the user role 'Administrasi', and a user profile icon. The left sidebar contains a menu with 'Dashboard', 'DATA MASTER' (Supplier, Master Barang), 'TRANSAKSI' (Transaksi, Barang Masuk, Barang Keluar), and 'Data User'. The main content area is titled 'Tambah Barang Keluar' and features a 'Form Barang' section with the following fields: 'ID Barang' (text input), 'Tanggal Keluar' (text input), 'Nama Barang' (dropdown menu with '--Pilih--'), and 'Jumlah Barang' (text input with a spinner). A 'Simpan Data' button is located in the top right corner. To the right of the form is a 'Preview' section showing a 3D box icon and a checklist with three items, each with a checkmark. Below the preview, there are two text input fields labeled 'Nama Barang' and 'Stok Barang', both containing a hyphen. The footer of the application reads '© 2023 Shima, All Right Reserve'.

Gambar 3. 43 Form Tambah Data Transaksi Barang Keluar

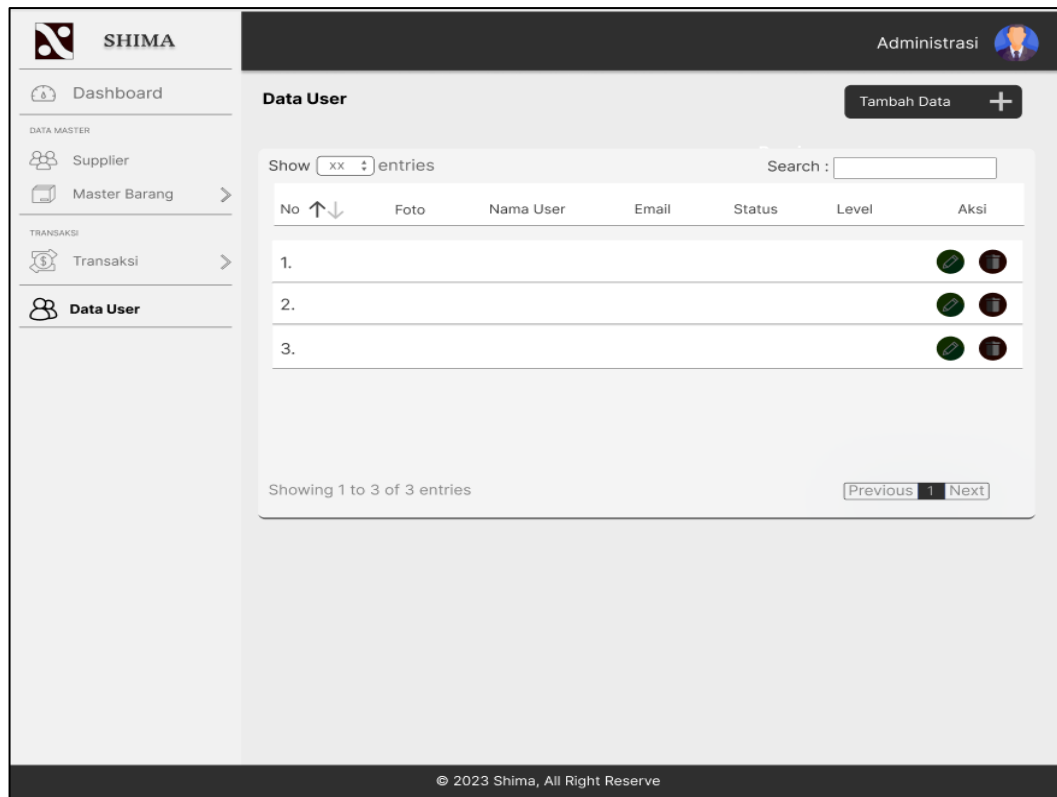
Gambar diatas menjelaskan desain form tambah data, dimana *user* harus mengisi data terlebih dahulu kemudian bisa menekan tombol simpan apabila ingin menyimpan data ke dalam *database*.

The screenshot displays the SHIMA web application interface for editing transaction data. The page title is "Ubah Barang Keluar". The interface includes a sidebar with navigation options: Dashboard, DATA MASTER (Supplier, Master Barang), TRANSAKSI (Transaksi, Master Barang, Barang Masuk, Barang Keluar), and Data User. The main content area is divided into two sections: "Form Barang" and "Preview". The "Form Barang" section contains input fields for ID Barang, Tanggal Masuk, Nama Barang (with a dropdown menu), Supplier (with a dropdown menu), and Jumlah Barang (with a spinner). The "Preview" section shows a 3D cube icon and fields for Nama Barang and Stok Barang. A "Simpan Perubahan" button is located at the top right of the main content area. The footer of the page reads "© 2023 Shima, All Right Reserve".

Gambar 3. 44 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 Master Barang



Gambar 3. 45 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 screenshot displays the 'Tambah Pengguna' (Add User) form in the SHIMA application. The interface includes a sidebar with navigation options like 'Dashboard', 'Supplier', 'Master Barang', 'Transaksi', and 'Data User'. The main content area is titled 'Tambah Pengguna' and features a 'Simpan Data' button. The form is divided into two sections: 'Form Pengguna' and 'Foto'. The 'Form Pengguna' section contains input fields for 'Nama Lengkap', 'Username', 'Nomor Telepon', 'Email', 'Level' (a dropdown menu), 'Password', and 'Konfirmasi Password'. The 'Foto' section includes a 'Format' dropdown (set to '.jpg .png .jpeg'), a placeholder image of a person, and a 'Browse' button for file selection. A 'Pilih File' button is also visible. The footer of the page reads '© 2023 Shima, All Right Reserve'.

Gambar 3. 46 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 screenshot displays the 'Ubah Pengguna' (Edit User) form in the SHIMA application. The interface is similar to the 'Tambah Pengguna' form, with a sidebar and a main content area titled 'Ubah Pengguna' featuring a 'Simpan Perubahan' button. The 'Form Pengguna' section includes input fields for 'Nama Lengkap', 'Username', 'Nomor Telepon', 'Email', 'Level' (a dropdown menu), 'Status' (a dropdown menu), and 'Password'. The 'Foto' section includes a 'Format' dropdown (set to '.jpg .png .jpeg'), a placeholder image of a person, and a 'Browse' button for file selection. A note below the photo section reads '*Kosongkan jika tidak ada ingin merubah' and includes a 'Pilih File' button. The 'Ubah Password' section includes a text input field with the note 'Kosongkan jika tidak ingin merubah!', and fields for 'Password' and 'Konfirmasi Password'. The footer of the page reads '© 2023 Shima, All Right Reserve'.

Gambar 3. 47 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*.