

## BAB 3

### METODE PENELITIAN

Pada bab ini, peneliti akan menguraikan secara rinci metode penelitian yang diterapkan dalam pengembangan sistem informasi inventori untuk PT Surya Prima Sakti. Pendekatan yang digunakan dalam penelitian ini menggunakan metode *Waterfall*. Tahapan yang dilakukan dapat dilihat pada flowchart Gambar 3.1 di bawah ini:



Gambar 3.1 Alur Metode Penelitian

#### 3.1 Identifikasi Masalah

Identifikasi masalah menjadi landasan penting dalam penelitian ini, karena pada tahap awal ini, peneliti menyorot permasalahan yang menjadi fokus untuk

dicari solusinya. Peneliti berbicara langsung kepada bagian *support* dari PT Surya Prima Sakti untuk mengetahui permasalahan yang dihadapi terkait sistem inventori mereka saat ini yang masih terbatas oleh instalasi sistem harus memiliki akses internet pada *server* kantor. Informasi di bawah ini didapatkan oleh penulis setelah melakukan pembicaraan:

1. Sistem yang dikembangkan akan memiliki 4 jumlah *role* yang mencakupi *role Admin*, *role Gudang*, *role Sales*, dan *role Purchasing*.
2. Fungsi utama pada sistem yang diinginkan oleh PT Surya Prima Sakti adalah dapat mengetahui jumlah barang yang dimiliki perusahaan dan pembuatan berbagai pelaporan, seperti pelaporan barang masuk dan pelaporan barang keluar.

Setelah memperoleh informasi dari *client*, di dapatkan informasi mengenai alur kerja inventori pada perusahaan, ilustrasi alur dapat dilihat pada Gambar 3.2 di bawah ini:



Gambar 3.2 Alur Inventaris Barang PT Surya Prima Sakti

Langkah selanjutnya penulis melakukan tinjauan pustaka dengan mencari literatur yang berfokus pada penelitian serupa yang telah dilakukan sebelumnya. Penelitian sebelumnya yang cukup serupa dijadikan penulis sebagai rujukan. Selain itu, penulis juga mencari literatur terkait lainnya untuk mendalami pemahaman terkait metode dan teori yang berhubungan dengan topik penelitian.

### 3.2 Analisis Kebutuhan

Setelah melakukan identifikasi kebutuhan dan menyusuri literatur terkait, langkah selanjutnya adalah menerapkan analisis kebutuhan. Pada tahap ini, penulis merinci proses analisis yang mendalam terhadap kebutuhan sistem yang akan dikembangkan dan memisahkan kebutuhan menjadi kebutuhan fungsional dan non fungsional.

#### 3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan aspek penting dalam pengembangan sistem informasi, merinci elemen-elemen esensial yang harus dipenuhi agar sistem



dapat beroperasi sesuai dengan harapan dan tujuan yang telah ditetapkan (Mujib, 2023). Pada tahap ini, penelitian mengulas secara mendalam setiap kebutuhan fungsional yang telah diidentifikasi sebelumnya, membahas secara rinci fungsi-fungsi yang menjadi fondasi utama dalam mendukung operasional sistem. Memahami dan merinci kebutuhan fungsional bertujuan untuk memastikan bahwa sistem yang dikembangkan dapat memberikan solusi yang responsif, efektif, dan sesuai dengan ekspektasi pengguna serta pemangku kepentingan. Tabel 3.1 di bawah ini menjelaskan kebutuhan fungsional yang digunakan pada sistem:

Tabel 3.1 Kebutuhan Fungsional

No	Kode Fitur	Fitur	Keterangan
1	KF01	<i>Login</i>	Setiap pengguna dapat melakukan login sebelum mengakses fitur pada sistem.
2	KF03	Mengolah Laporan Barang Masuk	<i>Role purchasing</i> dapat melihat, <i>admin</i> dan gudang dapat membuat, mengedit dan menghapus laporan untuk barang baru yang masuk.
3	KF04	Mencetak Laporan Barang Masuk	<i>Role admin</i> , gudang dan <i>purchasing</i> dapat mencetak daftar laporan barang masuk dalam bentuk PDF.
4	KF05	Mencetak Individu Laporan Barang Masuk	<i>Role admin</i> , gudang dan <i>purchasing</i> dapat mencetak laporan barang masuk dalam bentuk PDF secara individu.
5	KF06	Melihat <i>List</i> Laporan Barang Masuk	<i>Role admin</i> , gudang dan <i>purchasing</i> dapat melihat laporan untuk barang baru yang masuk.

No	Kode Fitur	Fitur	Keterangan
6	KF07	Mengolah Laporan Barang Keluar	<i>Role sales</i> dapat melihat, <i>admin</i> dan gudang dapat membuat, mengedit dan menghapus laporan untuk barang yang keluar.
7	KF08	Mencetak Laporan Barang Keluar	<i>Role admin.</i> gudang dan <i>sales</i> dapat mencetak daftar laporan barang keluar dalam bentuk PDF.
8	KF09	Mencetak Individu Laporan Barang Keluar	<i>Role admin</i> , gudang, dan <i>sales</i> dapat mencetak laporan barang keluar dalam bentuk PDF secara individu.
9	KF10	Melihat <i>List</i> Laporan Barang Keluar	<i>Role admin</i> , gudang, dan <i>sales</i> dapat melihat laporan untuk barang baru yang keluar.
10	KF11	Melihat Daftar Seluruh Barang	Semua <i>role</i> dapat melihat informasi stok barang yang sedang dimiliki oleh perusahaan.
11	KF12	Mencetak <i>List</i> Stok Barang	Semua <i>role</i> dapat mencetak list stok barang dalam bentuk PDF.
12	KF13	Mengolah Stok Barang	<i>Role admin</i> dan gudang dapat melihat, menambahkan, mengedit dan menghapus informasi stok barang yang dimiliki perusahaan.
13	KF02	<i>Register</i>	<i>Role admin</i> dapat membuat akun baru untuk <i>user</i> .
14	KF14	Mengolah Data Kategori	<i>Role admin</i> dan gudang dapat mengolah data kategori barang.



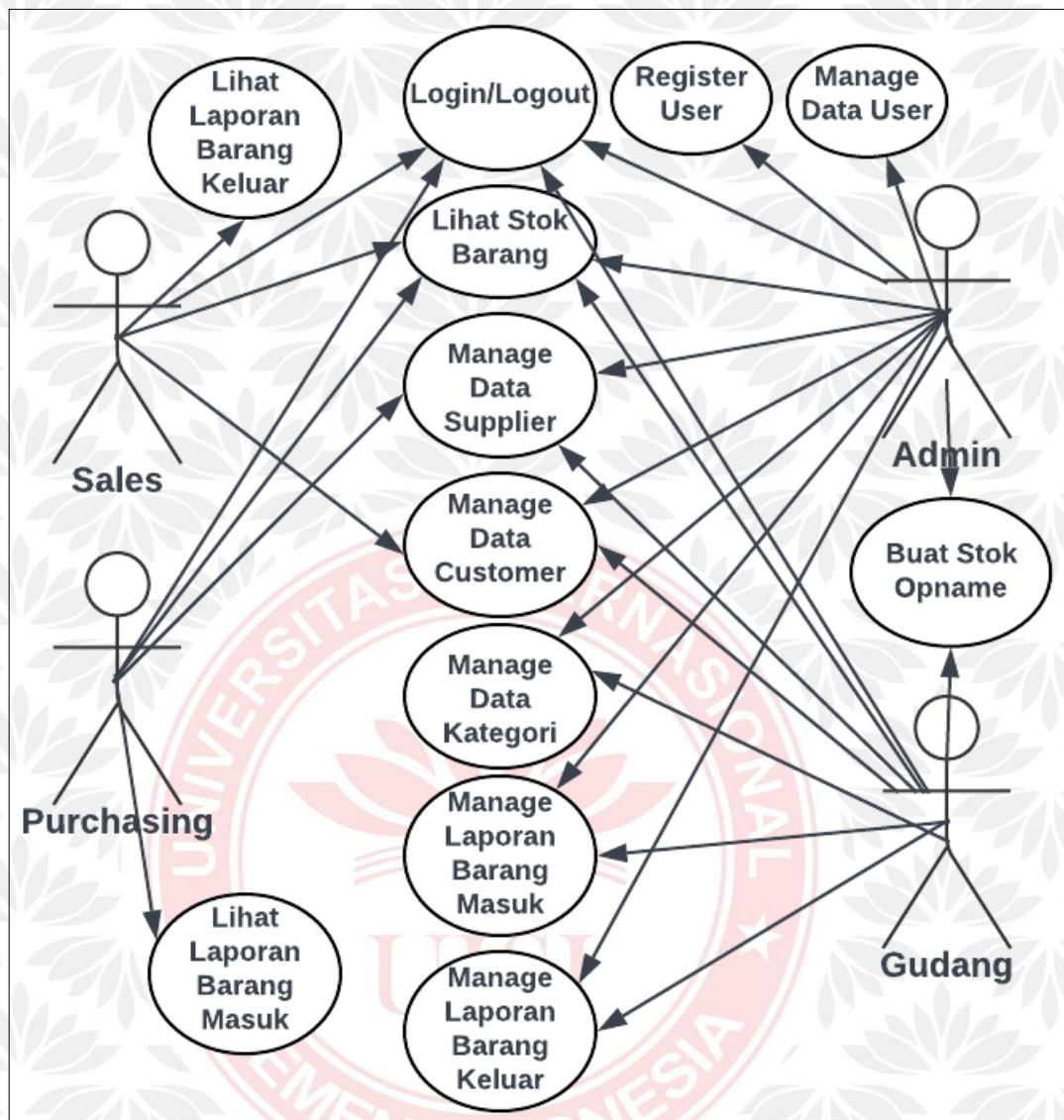
No	Kode Fitur	Fitur	Keterangan
15	KF15	Melihat Daftar Seluruh Kategori	<i>Role admin</i> dan gudang dapat melihat daftar informasi <i>customer</i> perusahaan.
16	KF16	Melihat Daftar Seluruh <i>Customer</i>	<i>Role admin</i> , gudang dan <i>sales</i> dapat melihat daftar informasi <i>customer</i> perusahaan.
17	KF17	Mencetak Data <i>Customer</i>	<i>Role admin</i> , gudang dan <i>sales</i> dapat mencetak daftar <i>customer</i> dalam bentuk PDF.
18	KF18	Mengolah Data <i>Customer</i>	<i>Role admin</i> , gudang, dan <i>sales</i> dapat melihat, menambahkan, mengedit dan menghapus segala informasi data untuk <i>customer</i> .
19	KF19	Melihat Daftar Seluruh <i>Supplier</i>	<i>Role admin</i> , gudang, dan <i>purchasing</i> dapat melihat daftar informasi <i>supplier</i> barang pada perusahaan.
20	KF20	Mengolah Data <i>Supplier</i>	<i>Role admin</i> , gudang, dan <i>purchasing</i> dapat melihat, menambahkan, mengedit dan menghapus segala informasi data untuk <i>supplier</i> .
21	KF21	Mencetak Data <i>Supplier</i>	<i>Role admin</i> , gudang, dan <i>purchasing</i> dapat mencetak data <i>supplier</i> dalam bentuk PDF.
22	KF22	Melihat <i>List User</i>	<i>Role admin</i> dapat melihat, menambahkan, mengedit dan

No	Kode Fitur	Fitur	Keterangan
			menghapus segala informasi data <i>user</i> pada sistem.
23	KF23	Mengola Data <i>User</i>	<i>Role admin</i> dapat melihat informasi dari setiap <i>user</i> yang terdaftar pada sistem.
24	KF24	<i>Filter</i> Laporan Barang Sesuai Tanggal	Semua <i>role</i> dapat menyaring laporan barang keluar atau masuk dengan memasukkan tanggal yang diinginkan.
25	KF25	Tambah Stok Opname	<i>Role admin</i> dan gudang dapat menulis pengecekan stok setiap bulan dan tidak dapat dihapus atau di ubah.
26	KF26	<i>Logout</i>	Setiap pengguna dapat <i>logout</i> dari sistem.

#### A. Use Case Diagram

Dari fitur-fitur di atas, penulis menggunakan use case diagram untuk memvisualisasikan setiap role dan fitur yang dapat diakses dalam sistem. Diagram use case mengilustrasikan hubungan antara fitur-fitur dari sistem dan aktor-aktor yang berinteraksi dengan sistem tersebut. Aktor dapat berupa individu, perangkat, atau sistem lain yang terlibat dalam interaksi dengan sistem yang sedang dikembangkan. Use case menunjukkan fungsionalitas sistem atau persyaratan yang harus dipenuhi agar sistem dapat memenuhi kebutuhan pengguna secara optimal (Setiawan and Khairuzzaman, 2017). Dengan demikian, use case diagram memberikan gambaran yang jelas tentang bagaimana berbagai elemen dalam sistem berinteraksi satu sama lain dan dengan aktor eksternal. Pada Gambar 3.3 di bawah ini adalah diagram use case pada sistem yang diteliti:





Gambar 3.3 Use Case Diagram

Berdasarkan gambar *use case* di atas, terdapat 4 aktor atau *role* dalam sistem ini yaitu aktor *admin*, gudang, *sales*, dan *purchasing*. Keempat *role* diatas memiliki akses masing-masing pada sistem. Berikut pada Tabel 3.2 adalah penjelasan akses pada masing-masing *role*:

Tabel 3.2 Akses *Role*

No	Aktor	Akses Pada Sistem
1	<i>Admin</i>	- <i>Manage &amp; Print Data Customer</i>

No	Aktor	Akses Pada Sistem
		<ul style="list-style-type: none"> <li>- <i>Manage &amp; Print Data Supplier</i></li> <li>- <i>Manage Data Kategori</i></li> <li>- <i>Manage Data User</i></li> <li>- <i>Lihat &amp; Print Stok Barang</i></li> <li>- <i>Manage &amp; Print Laporan Barang Masuk</i></li> <li>- <i>Manage &amp; Print Laporan Barang Keluar</i></li> <li>- <i>Buat Stok Opname</i></li> <li>- <i>Login &amp; Logout</i></li> <li>- <i>Register User</i></li> </ul>
2	Gudang	<ul style="list-style-type: none"> <li>- <i>Manage &amp; Print Data Customer</i></li> <li>- <i>Manage &amp; Print Data Supplier</i></li> <li>- <i>Manage Data Kategori</i></li> <li>- <i>Lihat &amp; Print Stok Barang</i></li> <li>- <i>Manage &amp; Print Laporan Barang Masuk</i></li> <li>- <i>Manage &amp; Print Laporan Barang Keluar</i></li> <li>- <i>Login &amp; Logout</i></li> <li>- <i>Buat Stok Opname</i></li> </ul>
3	Purchasing	<ul style="list-style-type: none"> <li>- <i>Manage &amp; Print Data Supplier</i></li> <li>- <i>Lihat &amp; Print Stok Barang</i></li> <li>- <i>Lihat &amp; Print Laporan Barang Masuk/Pembelian</i></li> </ul>
4	Sales	<ul style="list-style-type: none"> <li>- <i>Manage &amp; Print Data Customer</i></li> <li>- <i>Lihat &amp; Print Stok Barang</i></li> <li>- <i>Lihat &amp; Print Laporan Barang Keluar/Penjualan</i></li> </ul>

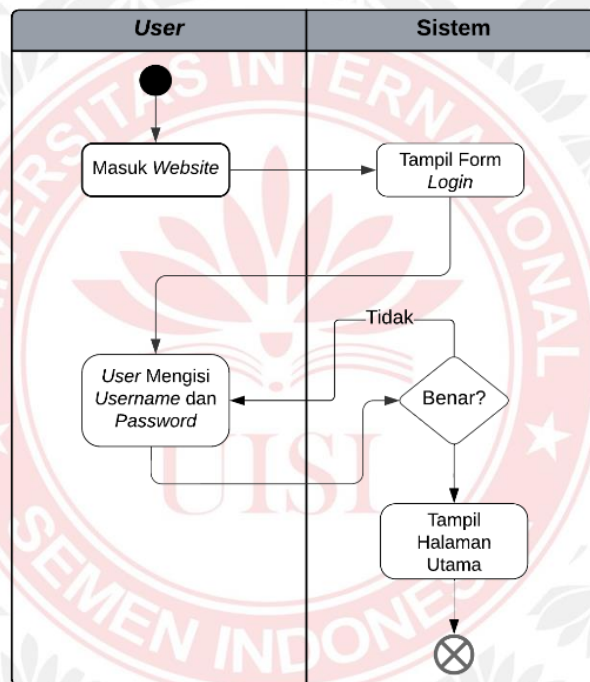
## B. Desain Activity Diagram



*Diagram* aktifitas adalah sebuah teknik yang menggambarkan aktivitas atau aliran kerja dari suatu sistem, desain ini cukup banyak digunakan karena dianggap mudah dipahami bagi sebagian orang. Aktivitas diagram untuk setiap fitur pada sistem adalah sebagai berikut:

#### 1. Aktifitas *Login* (AD01)

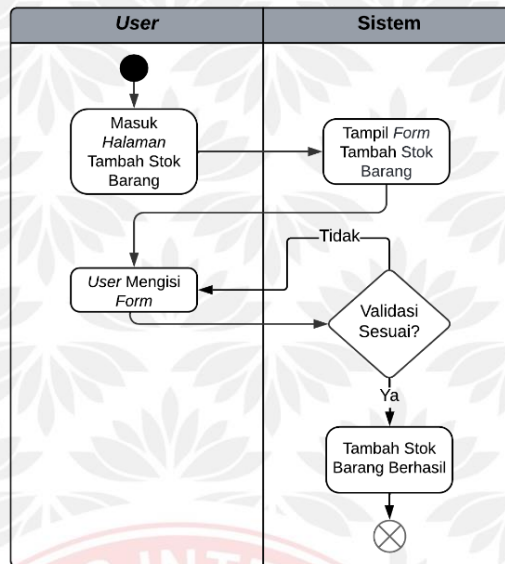
Di bawah ini adalah Gambar 3.4 dengan alur sistem *login* pada *website* sistem *inventory* SI SUPRAS, dimana aktifitas dimulai dengan *user* memasukkan *username* dan *password* yang telah di daftarkan sebelumnya oleh *role admin*:



Gambar 3.4 Activity Diagram Login

#### 2. Aktifitas *Diagram* Tambah Barang (AD20)

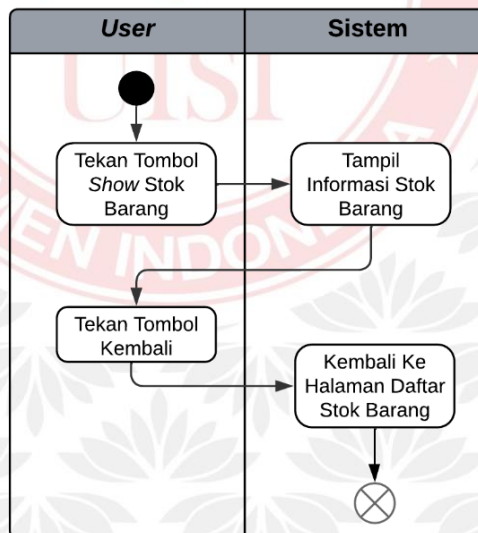
Jika perusahaan memiliki barang barang yang belum pernah dimiliki, sebelum *user* membuat laporan barang masuk *user* terlebih dahulu harus membuat stok baru dengan mengisi nama barang, berikut adalah alur aktifitas diagram untuk menambahkan data stok barang, dimulai dari *user* menekan tombol tambah stok barang, kemudian *user* di arahkan untuk mengisi informasi stok barang pada *form* yang telah disediakan, seperti pada Gambar 3.5 di bawah ini:



Gambar 3.5 Activity Diagram Tambah Data Stok Barang

### 3. Aktifitas Diagram Show Stok Barang (AD21)

Di bawah ini merupakan Gambar 3.6 alur yang dilakukan *user* agar dapat melihat informasi stok barang secara individu, informasi yang tersedia yakni nama barang, kategori, satuan dan jumlah stok. Dimulai dari *user* masuk ke halaman daftar kategori, lalu tekan tombol simbol mata berwarna biru:

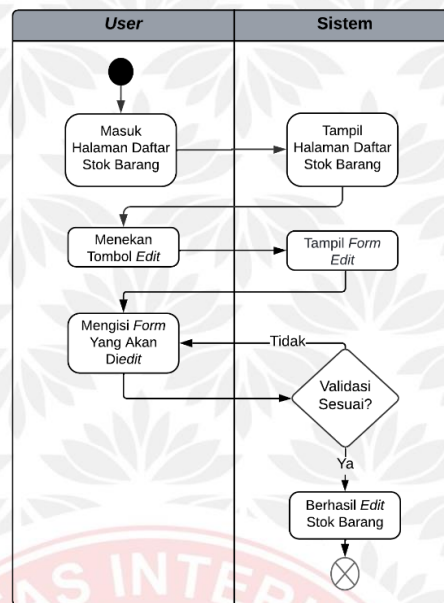


Gambar 3.6 Activity Diagram Show Stok Barang

### 4. Aktifitas Diagram Edit Data Stok Barang (AD22)

Berikut adalah Gambar 3.7 untuk alur bagaimana *user* dapat mengedit informasi stok barang. Dimulai dari *user* menekan tombol dengan simbol pensil berwarna hijau, lalu *user* dapat mengganti informasi stok barang sesuai yang dibutuhkan:

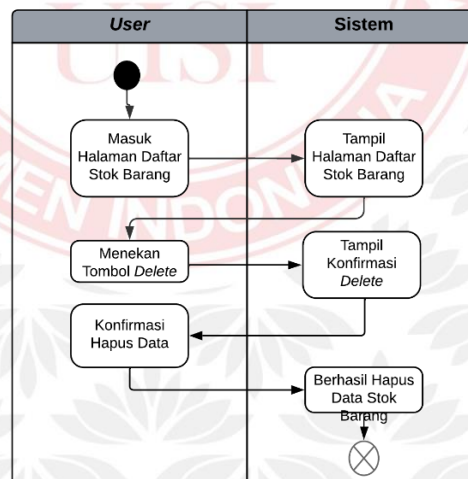




Gambar 3.7 Activity Diagram Edit Data Stok Barang

##### 5. Aktifitas Diagram Hapus Data Stok Barang (AD23)

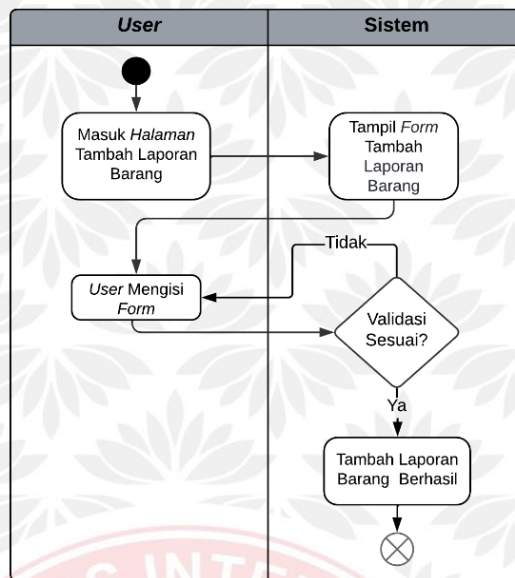
Berikut adalah Gambar 3.8 alur diagram untuk menghapus data stok barang, dimulai dari *user* menekan tombol dengan simbol sampah berwarna merah pada halaman daftar stok barang lalu *user* akan diberikan pesan konfirmasi untuk memastikan *user* akan menghapus data stok barang yang dipilih:



Gambar 3.8 Activity Diagram Hapus Data Stok Barang

##### 6. Aktifitas Diagram Tambah Laporan Barang (AD25)

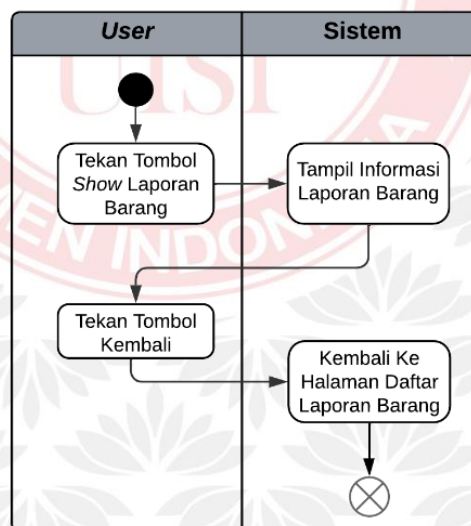
Berikut adalah alur aktifitas diagram untuk menambahkan data barang masuk dan keluar, dimulai dari *user* menekan tombol tambah laporan, lalu *user* di arahkan untuk mengisi informasi laporan barang masuk dan keluar pada *form* yang telah disediakan dan di ilustrasikan pada Gambar 3.9 ini:



Gambar 3.9 Activity Diagram Tambah Data Laporan Barang

#### 7. Aktivitas Diagram Lihat Informasi Laporan Barang (AD26)

Di bawah ini adalah alur diagram untuk dapat melihat informasi barang masuk dan keluar yang telah ditambahkan sebelumnya, dimulai dari *user* menekan tombol dengan simbol mata berwarna biru pada halaman daftar laporan barang masuk dan keluar dan di ilustrasikan pada Gambar 3.10 ini:

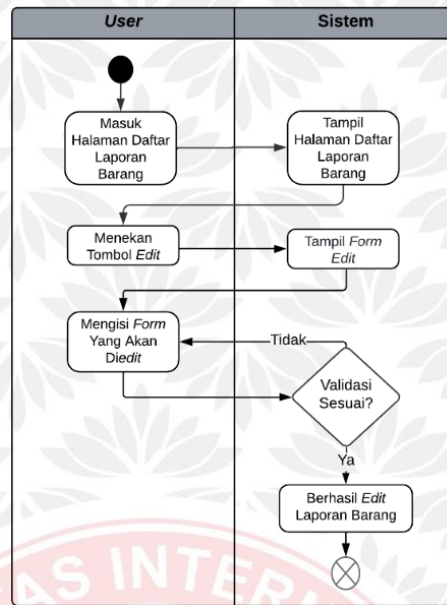


Gambar 3.10 Activity Diagram Show Informasi Laporan Barang

#### 8. Aktivitas Diagram Edit Data Laporan Barang (AD27)

Berikut adalah Gambar 3.11 ilustrasi alur bagaimana *user* dapat mengedit informasi barang masuk dan keluar. Dimulai dari *user* menekan tombol dengan simbol pensil berwarna hijau, lalu *user* dapat mengganti informasi laporan barang masuk dan keluar sesuai yang dibutuhkan:

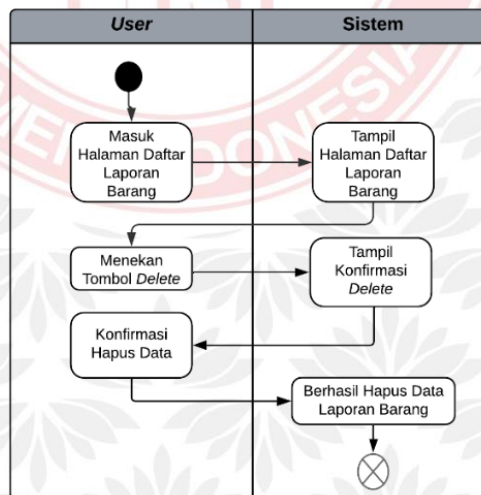




Gambar 3.11 Activity Diagram Edit Data Laporan Barang

#### 9. Aktivitas Diagram Hapus Data Laporan Barang (AD28)

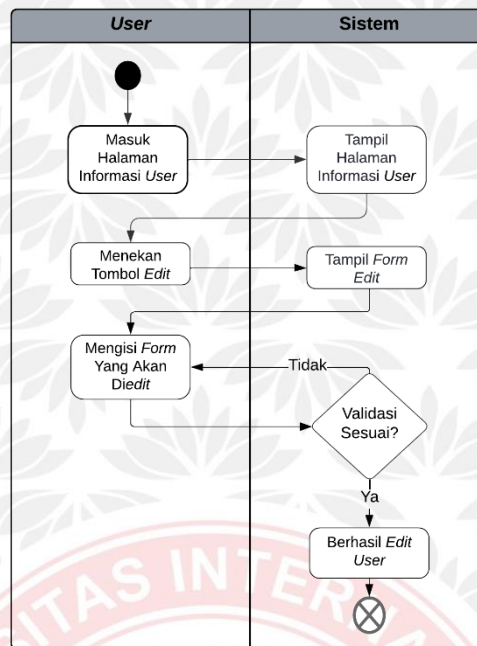
Berikut adalah Gambar 3.12 ilustrasi alur diagram untuk menghapus data barang masuk dan keluar, dimulai dari *user* menekan tombol dengan simbol sampah berwarna merah pada halaman daftar laporan barang masuk dan keluar lalu *user* akan diberikan pesan konfirmasi untuk memastikan *user* akan menghapus data laporan yang dipilih:



Gambar 3.12 Activity Diagram Delete Data Laporan Barang

#### 10. Aktivitas Edit User (AD04)

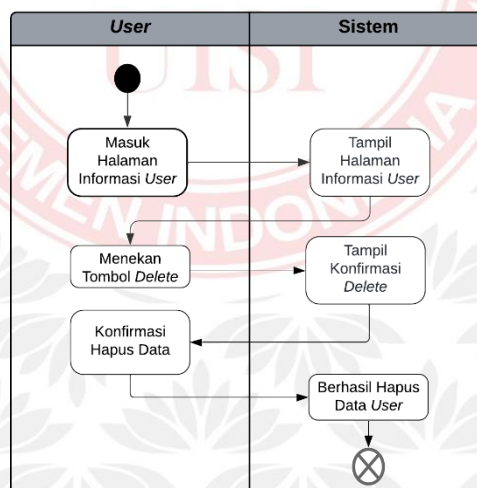
Berikut adalah Gambar 3.13 ilustrasi alur diagram untuk mengubah data *user*, dimulai dari *admin* masuk pada halaman khusus *admin* lalu memilih data *user* mana yang akan di ubah:



Gambar 3.13 Activity Diagram Edit Data User

#### 11. Aktifitas Delete User (AD05)

Berikut adalah Gambar 3.14 ilustrasi alur diagram untuk menghapus data *user*, dimulai dari *admin* masuk pada halaman khusus *admin* lalu memilih data *user* mana yang akan di hapus:

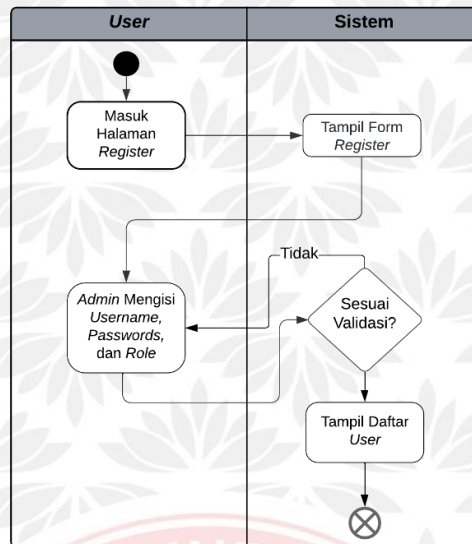


Gambar 3.14 Activity Diagram Delete Data User

#### 12. Aktifitas Register (AD02)

Berikut adalah Gambar 3.15 ilustrasi alur sistem registrasi *user* baru yang hanya dapat di akses oleh *user* dengan *role admin* dimana *user admin* harus masuk ke dalam *admin page* terlebih dahulu agar dapat mengakses halaman *register*:

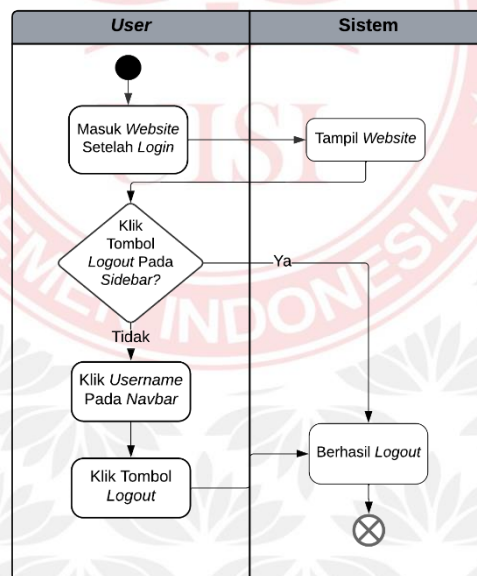




Gambar 3.15 Activity Diagram Register User

### 13. Aktivitas Logout (AD03)

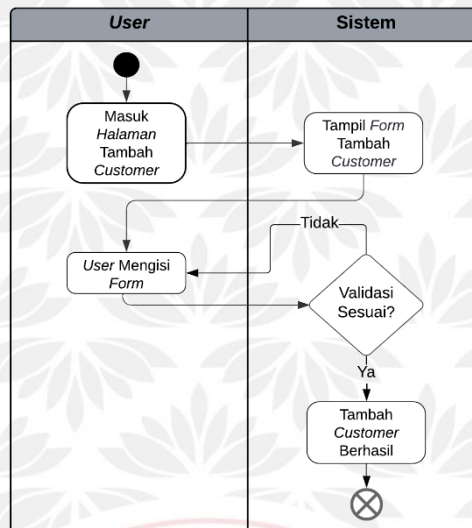
Di bawah ini adalah Gambar 3.16 untuk alur yang ditempuh *user* agar dapat keluar akun dari sistem dengan menekan tombol *logout* pada *username* di *navbar*



Gambar 3.16 Activity Diagram Logout User

### 14. Aktivitas Tambah Data Customer (AD06)

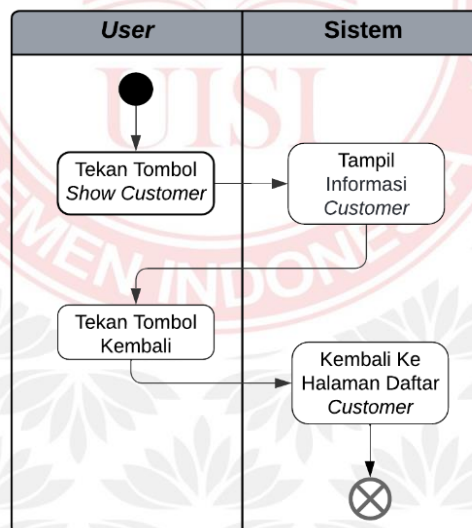
Berikut adalah Gambar 3.17 ilustrasi alur aktivitas diagram untuk menambahkan data *customer*, dimulai dari *user* menekan tombol tambah *supplier*, lalu *user* di arahkan untuk mengisi informasi *customer* pada *form* yang telah disediakan:



Gambar 3.17 Activity Diagram Tambah Data Customer

#### 15. Aktivitas Lihat Informasi Customer (AD07)

Di bawah ini adalah Gambar 3.18 ilustrasi alur diagram untuk dapat melihat informasi *customer* yang telah ditambahkan sebelumnya, dimulai dari *user* menekan tombol dengan simbol mata berwarna biru pada halaman daftar - *customer*:

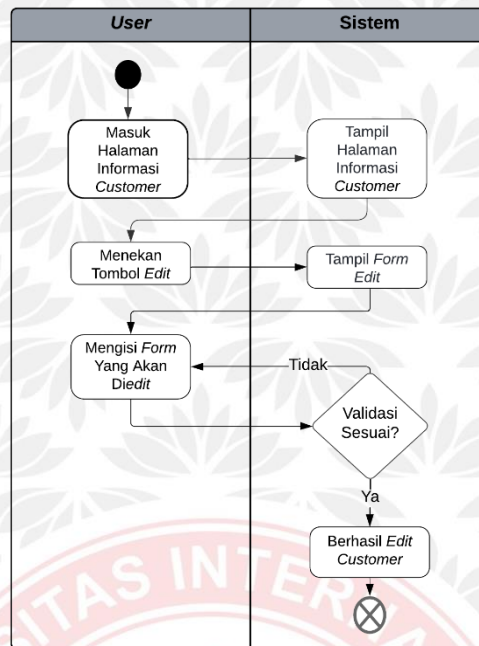


Gambar 3.18 Activity Diagram Lihat Informasi Data Customer

#### 16. Aktivitas Edit Data Customer (AD08)

Berikut adalah Gambar 3.19 ilustrasi alur bagaimana *user* dapat mengedit informasi *customer*. Dimulai dari *user* menekan tombol dengan simbol pensil berwarna hijau, lalu *user* dapat mengganti informasi *customer* sesuai yang dibutuhkan:

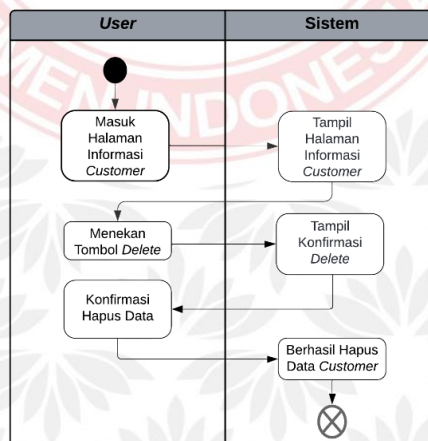




Gambar 3.19 Activity Diagram Edit Data Customer

#### 17. Aktivitas Diagram Hapus Data Customer (AD09)

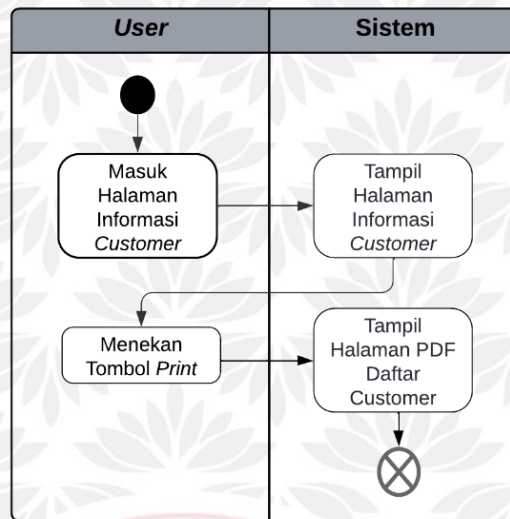
Berikut adalah Gambar 3.20 ilustrasi alur diagram untuk menghapus data *customer*, dimulai dari *user* menekan tombol dengan simbol sampah berwarna merah pada halaman daftar *customer* lalu *user* akan diberikan pesan konfirmasi untuk memastikan *user* akan menghapus data *customer* yang dipilih:



Gambar 3.20 Activity Diagram Hapus Data Customer

#### 18. Aktivitas Diagram Cetak Daftar Customer (AD10)

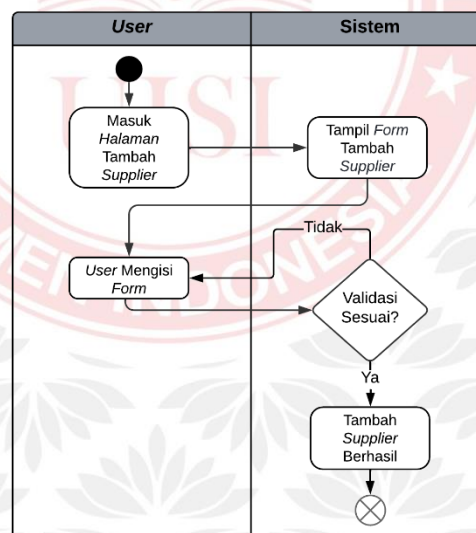
Alur di bawah ini adalah Gambar 3.21 ilustrasi yang di lalui oleh *user* dan sistem untuk mencetak atau mengunduh seluruh daftar *customer* menjadi file PDF:



Gambar 3.21 Activity Diagram Cetak Daftar Customer

#### 19. Aktivitas Diagram Tambah Data Supplier (AD11)

Berikut adalah Gambar 3.22 ilustrasi alur aktifitas diagram untuk menambahkan data *supplier*, dimulai dari *user* menekan tombol tambah *supplier*, lalu *user* di arahkan untuk mengisi informasi *supplier* pada *form* yang telah disediakan:

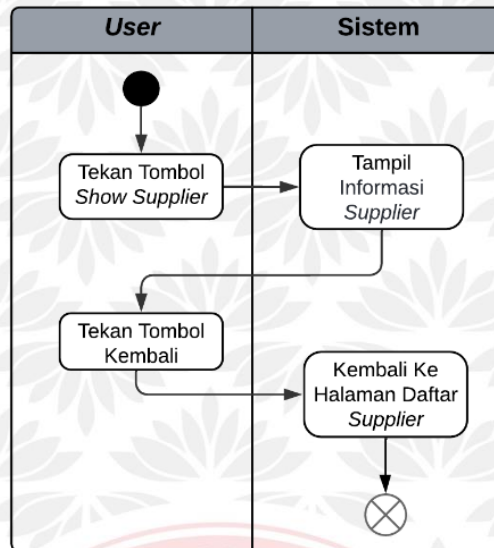


Gambar 3.22 Activity Diagram Tambah Data Supplier

#### 20. Aktivitas Diagram Lihat Informasi Supplier (AD12)

Di bawah ini adalah Gambar 3.23 ilustrasi alur diagram untuk dapat melihat informasi *supplier* yang telah ditambahkan sebelumnya, dimulai dari *user* menekan tombol dengan simbol mata berwarna biru pada halaman daftar *supplier*:

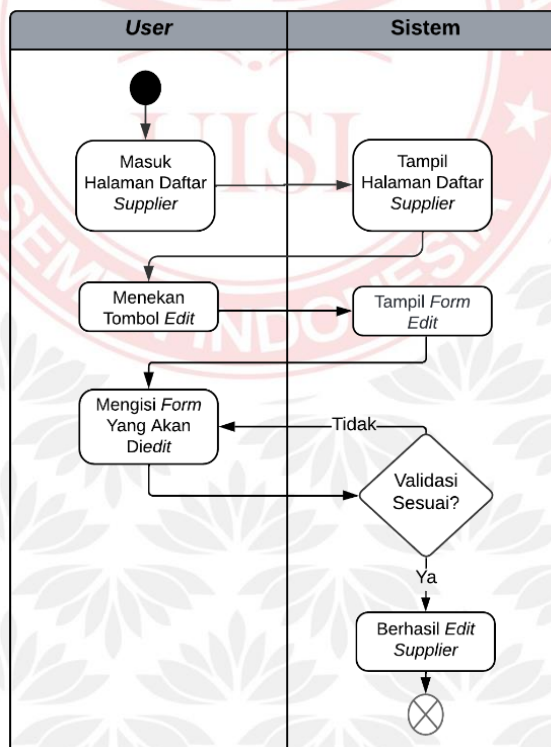




Gambar 3.23 Activity Diagram Lihat Informasi Data Supplier

## 21. Aktivitas Diagram Edit Data Supplier (AD13)

Berikut adalah Gambar 3.24 alur bagaimana *user* dapat mengedit informasi *supplier*. Dimulai dari *user* menekan tombol dengan simbol pensil berwarna hijau, lalu *user* dapat mengganti informasi *supplier* sesuai yang dibutuhkan:

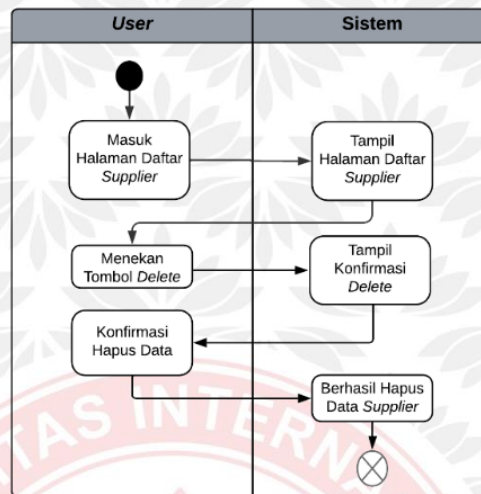


Gambar 3.24 Activity Diagram Edit Data Supplier

## 22. Aktivitas Diagram Hapus Data Supplier (AD14)

Berikut adalah Gambar 3.25 alur diagram untuk menghapus data *supplier*,

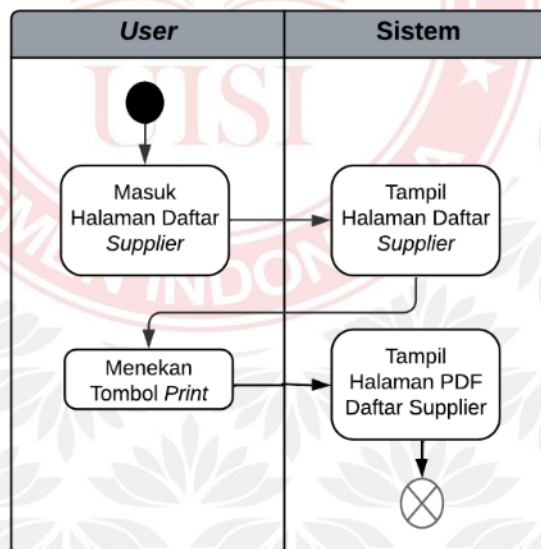
dimulai dari *user* menekan tombol dengan simbol sampah berwarna merah pada halaman daftar *supplier* lalu *user* akan diberikan pesan konfirmasi untuk memastikan *user* akan menghapus data *supplier* yang dipilih:



Gambar 3.25 Activity Diagram Hapus Data Supplier

#### 23. Aktivitas Diagram Cetak Daftar Supplier (AD15)

Alur di bawah ini adalah Gambar 3.26 yang di lalui oleh *user* dan sistem untuk mencetak atau mengunduh seluruh daftar *supplier* menjadi file PDF:

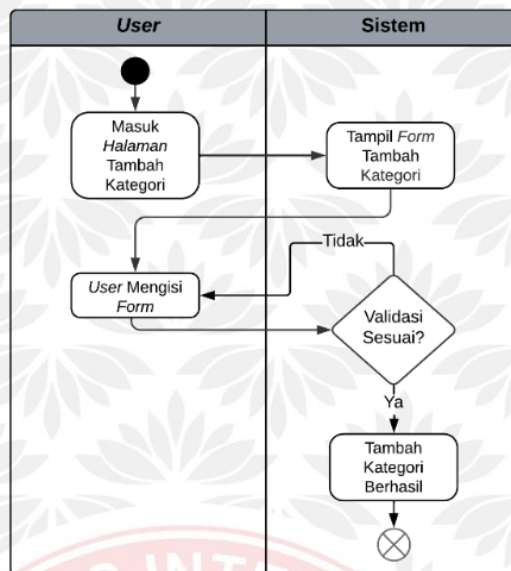


Gambar 3.26 Activity Diagram Cetak Daftar Supplier

#### 24. Aktivitas Diagram Tambah Data Kategori (AD16)

Berikut adalah Gambar 3.27 ilustrasi alur aktifitas diagram untuk menambahkan data kategori, dimulai dari *user* menekan tombol tambah kategori, lalu *user* di arahkan untuk mengisi informasi kategori pada *form* yang telah disediakan:

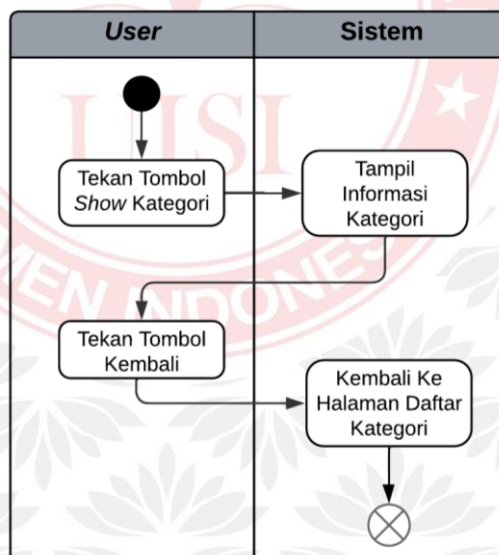




Gambar 3.27 Activity Diagram Tambah Data Kategori

## 25. Aktivitas Diagram Show Data Kategori (AD17)

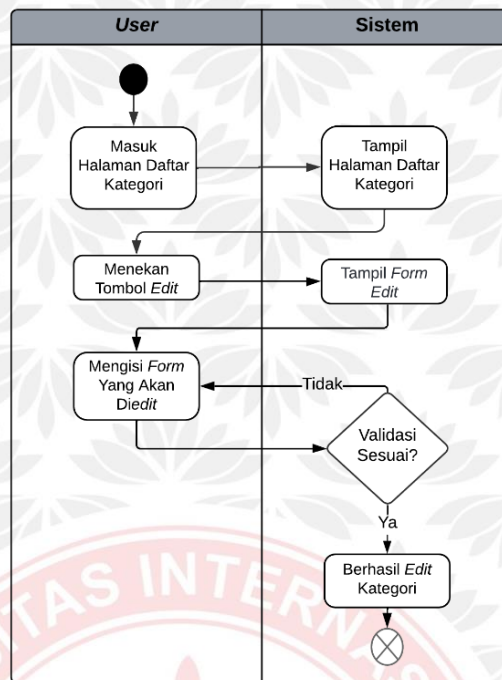
Di bawah ini merupakan Gambar 3.28 ilustrasi alur yang dilakukan *user* agar dapat melihat informasi kategori secara individu. Dimulai dari *user* masuk ke halaman daftar kategori, lalu tekan tombol simbol mata berwarna biru:



Gambar 3.28 Activity Diagram Show Kategori

## 26. Aktivitas Diagram Edit Data Kategori (AD18)

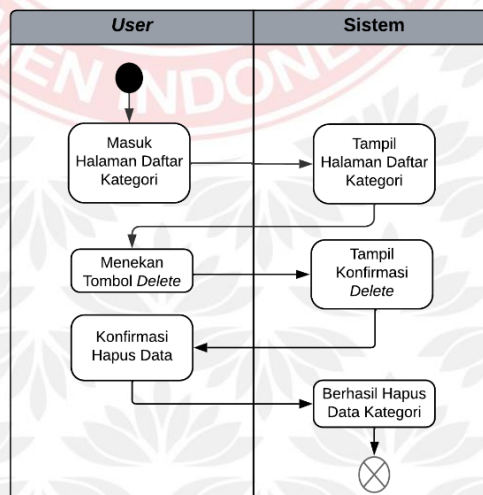
Berikut adalah Gambar 3.29 ilustrasi alur bagaimana *user* dapat mengedit informasi kategori. Dimulai dari *user* menekan tombol dengan simbol pensil berwarna hijau, lalu *user* dapat mengganti informasi kategori sesuai yang dibutuhkan:



Gambar 3.29 Activity Diagram Edit Kategori

## 27. Aktifitas Diagram Hapus Data Kategori (AD19)

Berikut adalah Gambar 3.30 ilustrasi alur diagram untuk menghapus data kategori, dimulai dari *user* menekan tombol dengan simbol sampah berwarna merah pada halaman daftar kategori lalu *user* akan diberikan pesan konfirmasi untuk memastikan *user* akan menghapus data stok barang yang dipilih:



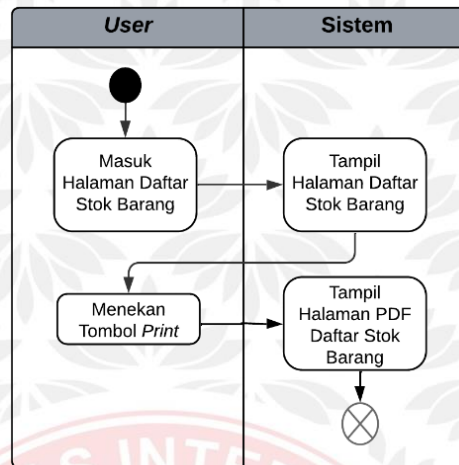
Gambar 3.30 Activity Diagram Hapus Data Kategori

## 28. Aktifitas Diagram Cetak Daftar Stok Barang (AD24)

Alur di bawah ini adalah Gambar 3.31 ilustrasi yang di lalui oleh *user* dan



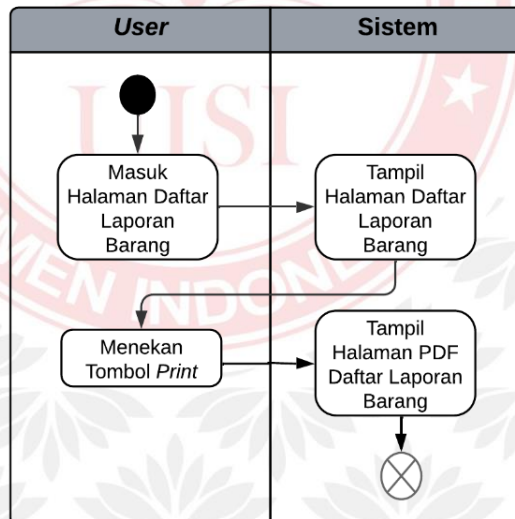
sistem untuk mencetak atau mengunduh seluruh daftar stok barang menjadi file PDF:



Gambar 3. 31 Activity Diagram Cetak Stok Barang

#### 29. Aktifitas Diagram Cetak Daftar Laporan Barang (AD29)

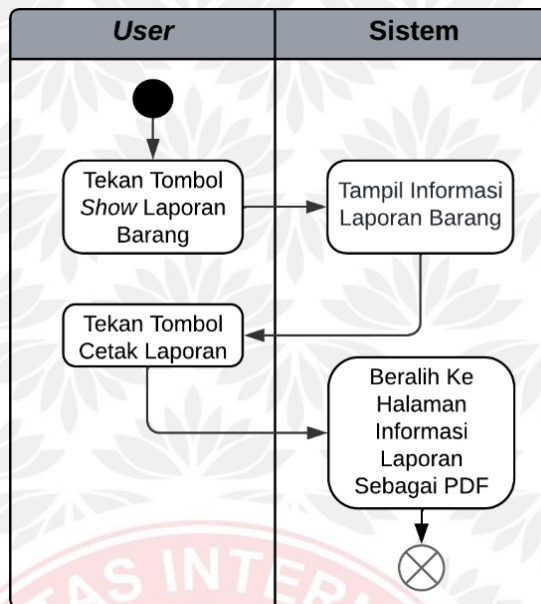
Alur di bawah ini adalah Gambar 3.32 ilustrasi yang di lalui oleh *user* dan sistem untuk mencetak atau mengunduh seluruh daftar laporan barang masuk dan keluar menjadi file PDF:



Gambar 3.32 Activity Diagram Cetak Daftar Laporan Barang

#### 30. Aktifitas Diagram Cetak Laporan Barang Individu (AD30)

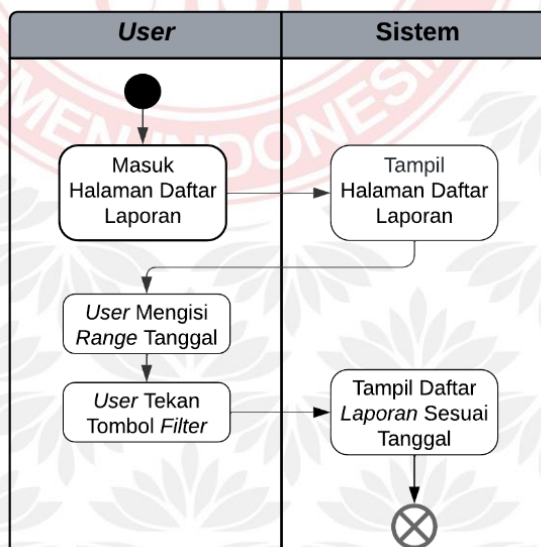
Berikut adalah Gambar 3.33 ilustrasi alur yang di lakukan *user* untuk mencetak atau mengunduh laporan barang barang secara individu sebagai *file* dengan format PDF. Dimulai dari *user* memilih laporan mana yang akan dicetak atau unduh, lalu tekan tombol simbol mata berwarna biru kemudia tekan tombol cetak laporan.



Gambar 3.33 Activity Diagram Cetak Laporan Individu

### 31. Aktivitas Diagram *Filter* Laporan Dengan Tanggal (AD31)

Di bawah ini adalah Gambar 3.34 ilustrasi alur untuk *user* agar dapat melihat laporan barang tertentu dengan memasukkan *range* tanggal yang diinginkan, dimulai dari *user* mengisi kolom tanggal pertama untuk jarak awal dan kolom tanggal kedua untuk jarak akhir dan selanjutnya menekan tombol *Filter* untuk dapat menyaring informasi laporan:



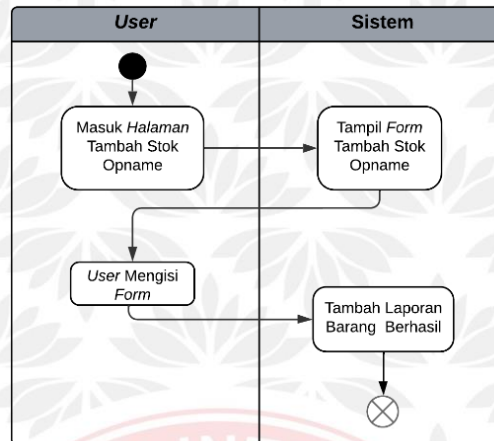
Gambar 3.34 Activity Diagram Cetak Laporan Individu

### 32. Aktivitas Diagram Tambah Stok Opname (AD32)

Di bawah ini adalah Gambar 3.35 ilustrasi alur untuk *user* agar dapat



menambahkan data stok opname, dimulai dari *user* masuk ke halaman daftar stok opname dan menekan tombol tambah stok opname:



Gambar 3. 35 Activity Diagram Tamnbah Stok Opname

### 3.2.2 Kebutuhan Non Fungsional

Kriteria dari kualitas atau kinerja yang harus terpenuhi oleh sistem perangkat lunak dapat dianggap sebagai kebutuhan non fungsional, dan ini menjadi aspek utama yang perlu diperhatikan selama tahap pengembangan sistem (Aziza and Fadhilah, 2020). Dengan merinci kebutuhan non-fungsional, pengembang dapat memastikan bahwa sistem yang dikembangkan tidak hanya memenuhi kriteria fungsional, tetapi juga memenuhi standar kualitas dan kinerja yang diinginkan oleh pengguna dan pemangku kepentingan. Berikut adalah kebutuhan non fungsional pada sistem ini:

1. *Usability*

Sistem ini harus memiliki antarmuka yang baik, nyaman dipandang dan mudah digunakan. Pengguna juga harus dengan mudah memahami dan menavigasi melalui fitur-fitur yang ada di sistem ini.

2. *Availability*

Sistem yang dikembangkan berbasis *website*, maka dari itu sistem dapat diakses secara fleksibel.

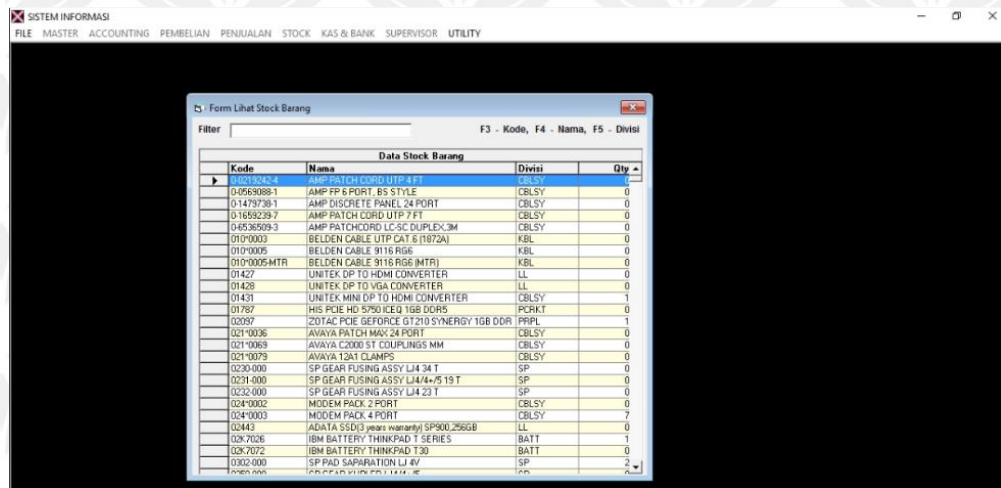
3. *Scalability*

Sistem harus dapat ditingkatkan atau dikembangkan untuk dapat lebih mempermudah pengguna melakukan aktivitas pekerjaan dalam sistem.

### 3.3 Desain Sistem

#### 3.3.1 Desain Interface

Desain *interface* memegang peranan krusial dalam mencapai keberhasilan implementasi sistem yang akan dibangun. Gambar 3.13 di bawah ini merupakan contoh tampilan antarmuka sedang digunakan oleh sistem inventory yang lama:



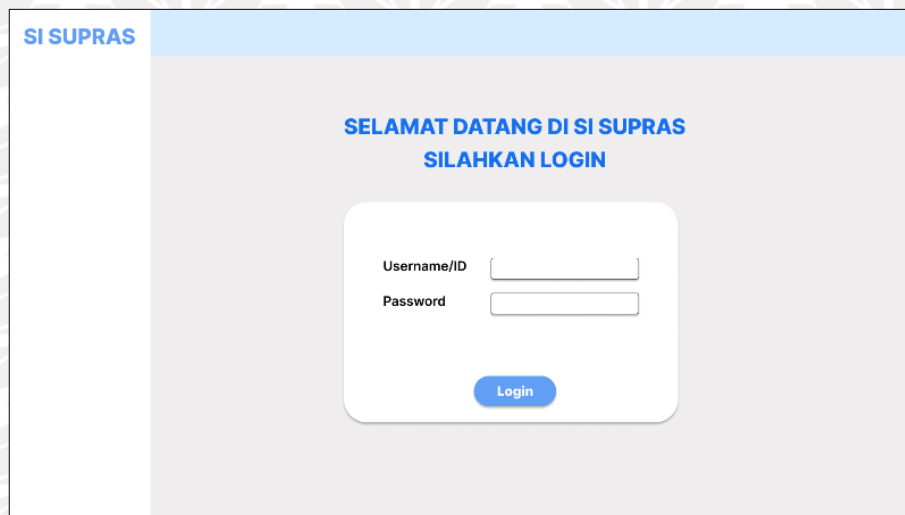
Gambar 3.36 Tampilan Sistem Lama

Seperti gambar di atas, tampilan dari salah satu fitur pada sistem yang saat ini digunakan terkesan kurang moder, dengan beberapa tombol dengan ukuran tulisan yang cukup kecil sehingga mempersulit pengguna untuk membaca setiap menu dan dapat mempersulit pengguna ketika akan melakukan tugasnya. Maka dari itu, peneliti akan memperkenalkan desain *wireframe high fidelity* yang telah dikembangkan sebagai bagian integral dari proses perancangan antarmuka. *Wireframe* adalah suatu struktur yang digunakan untuk mengorganisir elemen-elemen dalam sebuah halaman *website* atau aplikasi. Umumnya, *wireframe* dibuat sebelum sistem sebenarnya dikembangkan. *Wireframe high fidelity* memiliki Elemen-elemen yang tidak terlalu sederhana, memiliki beberapa tombol dengan warna dan tulisan yang jelas.

#### 1. Desain Tampilan Halaman Login

Pada halaman *login*, *user* diwajibkan untuk memasukkan *username* dan *password* yang sebelumnya telah didaftarkan untuk mengakses sistem. Halaman *Login* dapat dilihat pada Gambar 3.14 berikut:

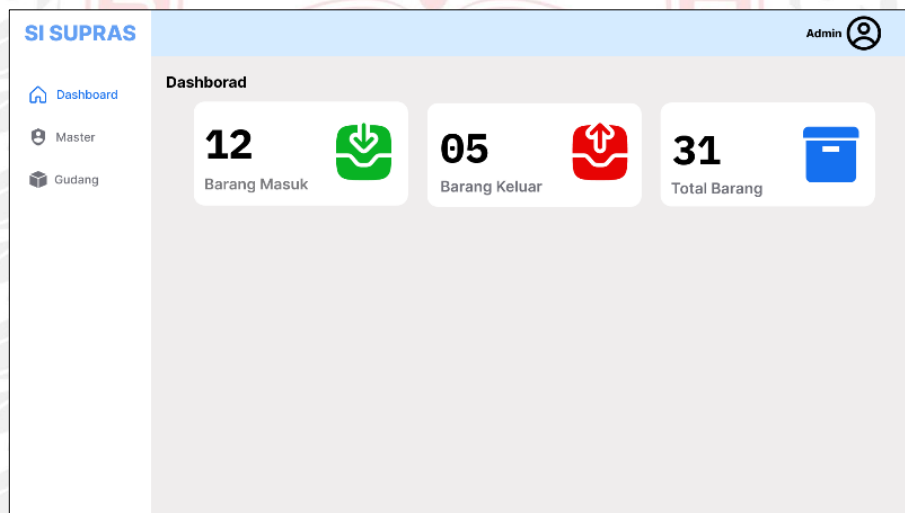




Gambar 3.37 Wireframe Halaman Login

## 2. Desain Tampilan Halaman Dashboard

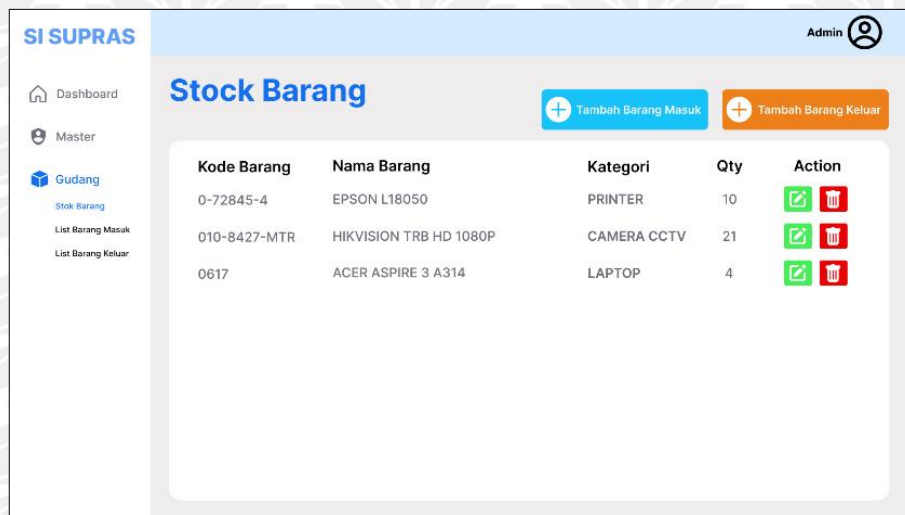
Pada halaman dashboard sistem akan menampilkan rangkuman terkait aktivitas barang pada sistem, seperti jumlah barang keluar, barang masuk dan total barang yang dimiliki perusahaan. Halaman Dashboard dapat dilihat pada Gambar 3.15 berikut:



Gambar 3.38 Wireframe Halaman Dashboard

## 3. Desain Tampilan Halaman Informasi Stok

Pada halaman informasi stok terdapat tabel yang menampilkan seluruh stok barang yang dimiliki perusahaan. Pengguna juga dapat menambahkan, menghapus, dan mengedit. Tampilan halaman stok dapat dilihat pada Gambar 3.16 berikut:



Gambar 3.39 Wireframe Halaman Stok

#### 4. Desain Tampilan Pembuatan Laporan Barang Masuk dan Keluar

Pada halaman ini, pengguna dapat membuat berbagai laporan secara otomatis. Pengguna akan dapat membuat laporan barang masuk dan barang keluar. Setiap laporan akan memiliki *form* yang berbeda-beda. Tampilan dari halaman laporan dapat dilihat pada Gambar 3.17 di bawah ini:

Gambar 3.40 Wireframe Halaman Laporan

#### 5. Desain Tampilan Informasi Barang Masuk dan Keluar

Pada Halaman ini pengguna dapat melihat riwayat informasi laporan barang masuk ataupun barang keluar yang telah dibuat sebelumnya, pada halaman ini pengguna juga dapat tombol cetak laporan untuk didownload dalam perangkat Tampilan dari halaman informasi laporan dapat dilihat pada Gambar 3.18 di bawah ini:

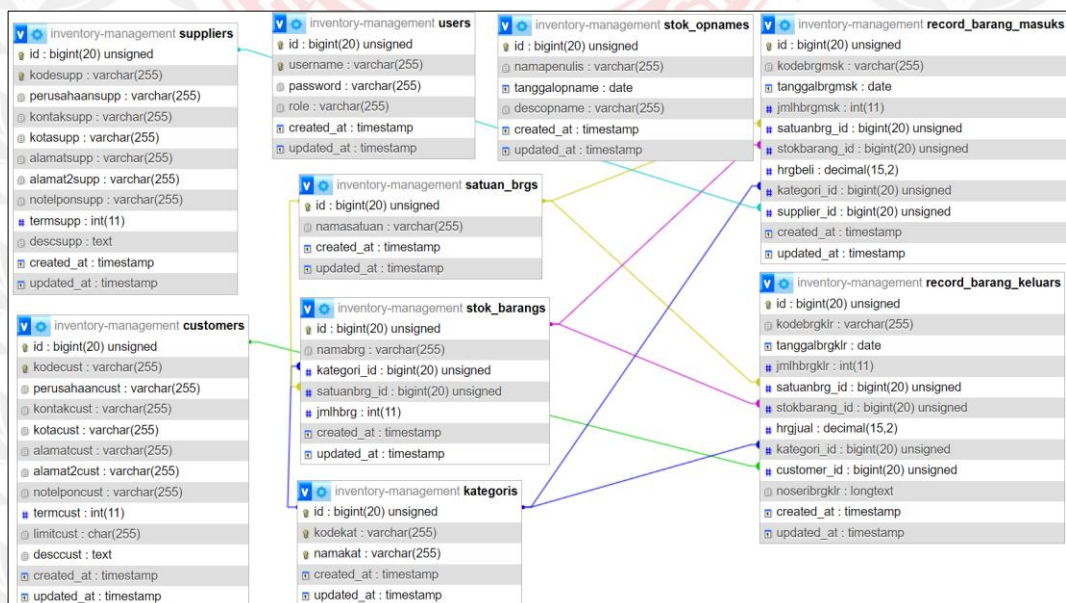




Gambar 3.41 Wireframe Halaman Informasi Laporan

### 3.3.2 Desain Database

Setelah alur sistem telah disepekat, selanjutnya peneliti memasuki tahap desain *database*. Salah satu langkah dalam desain database adalah dengan membuat *Entity Relationship Diagram*, ERD adalah sebuah diagram yang digunakan untuk merancang database, menunjukkan hubungan atau keterkaitan antara objek atau entitas beserta atribut-atributnya. Gambar 3.19 di bawah ini merupakan ERD dari sistem inventori yang dikembangkan memiliki 9 tabel yang masing-masing memiliki fungsinya sendiri, penulis membuat ERD dengan bantuan dari perangkat lunak phpmyadmin.



Gambar 3.42 Desain ERD SI SUPRAS

Berikut ini adalah Tabel 3.3 hingga Tabel 3.11 dengan penjelasan tentang data dan relasi yang dimiliki dalam setiap tabel *database* pada sistem yang akan dibangun:

#### 1. Tabel *User*

Tabel 3.3 Daftar Kolom Tabel *User*

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	username	<i>Varchar</i>	255
3	password	<i>Varchar</i>	255
4	role	<i>TinyInt</i>	1

Pada tabel ini berisi segala informasi *user* mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, *username* merupakan kolom yang dapat menyimpan nama *user*, *password* adalah kolom yang menyimpan kata sandi *user* dan *role* merupakan kolom yang dapat menentukan akses *user* sesuai tugas masing-masing. Tabel ini tidak memiliki relasi dengan tabel yang lain.

#### 2. Tabel Satuan Barang

Tabel 3.4 Daftar Kolom Tabel Satuan Barang

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	namasatuan	<i>Varchar</i>	255

Pada tabel ini berisi segala informasi satuan barang mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, dan *namasatuan* yang dapat menyimpan segala jenis satuan barang. Tabel ini memiliki relasi *one-to many* dengan tabel Stok Barang, *Record* Barang Masuk dan *Record* Barang Keluar.

#### 3. Tabel Stok Barang

Tabel 3.5 Daftar Kolom Tabel Stok Barang

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	namabrg	<i>Varchar</i>	255
3	kategori_id	<i>BigInt</i>	20



No	Nama Kolom	Tipe Data	Length
4	satuanbrg_id	<i>BigInt</i>	20
5	jmlhbrg	<i>int</i>	11

Pada tabel ini berisi segala informasi stok barang mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, namabarang yang dapat menyimpan nama-nama barang dari *user* dan harus berbeda dari yang lain, jmlhbrg merupakan kolom yang dapat menyimpan jumlah barang pada perusahaan, kategori\_id dan satuanbrg\_id dimana mengambil *foreign key* dari tabel kategori dan tabel satuan barang. Tabel ini memiliki relasi *one to many* dengan tabel *Record* Barang Masuk dan *Record* Barang Keluar, sedangkan dengan tabel satuan barang memiliki relasi *many to one*.

#### 4. Tabel *Record* Barang Masuk

Tabel 3.6 Daftar Kolom Tabel Laporan Barang Masuk

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	kodebrgmsk	<i>Varchar</i>	255
3	tanggalbrgmsk	<i>date</i>	
4	jmlhbrgmsk	<i>int</i>	11
5	satuanbrg_id	<i>BigInt</i>	20
6	stokbarang_id	<i>BigInt</i>	20
7	hrghbeli	<i>decimal</i>	15,2
8	kategori_id	<i>BigInt</i>	20
9	supplier_id	<i>BigInt</i>	20

Pada tabel ini berisi segala informasi laporan barang masuk mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, kodebrgmsk merupakan kode laporan yang diisi manual dan harus berbeda dari yang lain, jmlhbrgmsk merupakan kolom yang dapat menyimpan jumlah barang yang masuk dan dapat ditambahkan pada stok barang perusahaan, tanggalbrgmsk akan menyimpan tanggal kapan barang masuk tiba, kolom hrghbeli menyimpan harga yang dikeluarkan perusahaan dalam pembelian barang yang akan ditampilkan setelah dikalikan dengan jumlah barang masuk dan tabel ini memiliki *foreign key*

pada tabel *supplier\_id*, *kategori\_id*, *satunbrg\_id* dan *stokbarang\_id*. Tabel ini memiliki relasi *many to one* dengan tabel *Suppliers*, Satuan Barang, Kategori dan Stok Barang.

## 5. Tabel *Record* Barang Keluar

Tabel 3.7 Daftar Kolom Tabel Laporan Barang Keluar

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	kodebrgklr	<i>Varchar</i>	255
3	tanggalbrgklr	<i>date</i>	
4	jmlhbrgklr	<i>int</i>	11
5	satunbrg_id	<i>BigInt</i>	20
6	stokbarang_id	<i>BigInt</i>	20
7	hrgjual	<i>decimal</i>	15,2
8	kategori_id	<i>BigInt</i>	20
9	customer_id	<i>BigInt</i>	20
10	noseribrgklr	<i>longtext</i>	

Pada tabel ini berisi segala informasi laporan barang keluar mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, kodebrgmklr merupakan kode laporan yang diisi secara manual dan harus berbeda dari yang lain, jmlhbrgklr merupakan kolom yang dapat menyimpan jumlah barang yang keluar dan dapat mengurangi jumlah stok barang perusahaan, tanggalbrgklr akan menyimpan tanggal kapan barang keluar dan tidak bisa di ubah, kolom hrgjual menyimpan harga yang dikeluarkan perusahaan dalam penjualan barang yang akan ditampilkan setelah dikalikan dengan jumlah barang keluar, kolom nomor seri akan bertambah tergantung dari jumlah barang keluar yang telah diisi sebelumnya oleh *user*, dan tabel ini memiliki *foreign key* pada tabel *supplier\_id*, *kategori\_id*, *satunbrg\_id* dan *stokbarang\_id*. Tabel ini memiliki relasi *many to one* dengan tabel beberapa tabel lain seperti tabel *Suppliers*, tabel Satuan Barang, tabel Kategori dan tabel Stok Barang.



## 6. Tabel *Customer*

Tabel 3.8 Daftar Kolom Tabel *Customer*

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	kodecust	<i>Varchar</i>	255
3	perusahaancust	<i>Varchar</i>	255
4	kontakcust	<i>Varchar</i>	255
5	kotacust	<i>Varchar</i>	255
6	alamatcust	<i>Varchar</i>	255
7	alamat2cust	<i>Varchar</i>	255
8	notelponcust	<i>Varchar</i>	255
9	termcust	<i>int</i>	11
10	limitcust	<i>char</i>	255
11	desccust	<i>text</i>	

Pada tabel ini berisi segala informasi *customer* mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, kodecust merupakan kode perusahaan yang diisi manual dan harus berbeda dari yang lain, perusahaancust merupakan kolom yang dapat menyimpan nama perusahaan *customer*, kontak *customer* menyimpan nama *person in contact* pada perusahaan *customer*, kotacust menyimpan alamat kota *customer*, alamatcust memiliki 2 kolom dimana kolom kedua adalah *nullable* yang dapat diisi jika *customer* memiliki alamat gudang yang berbeda dari alamat kantor, termcust adalah kolom yang menyimpan batas waktu pembayaran oleh *customer*, limitcust menyimpan batasan harga yang dapat dimiliki *customer* sebelum pembayaran berikutnya dan descust adalah kolom *nullable* yang menyimpan keterangan *customer*. Tabel ini memiliki relasi *one to many* dengan tabel Record Barang Keluar.

## 7. Tabel *Supplier*

Tabel 3.9 Daftar Kolom Tabel *Supplier*

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	kodesupp	<i>Varchar</i>	255

No	Nama Kolom	Tipe Data	Length
3	perusahaansupp	Varchar	255
4	kontaksupp	Varchar	255
5	kotasupp	Varchar	255
6	alamat1supp	Varchar	255
7	alamat2supp	Varchar	255
8	notelponsupp	Varchar	255
9	termsupp	int	11
10	descsupp	text	

Pada tabel ini berisi segala informasi *supplier* mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, kodesupp merupakan kode perusahaan yang diisi manual dan harus berbeda dari yang lain, perusahaansupp merupakan kolom yang dapat menyimpan nama perusahaan *supplier*, kontaksupp menyimpan nama *person in contact* pada perusahaan *supplier*, kotasupp menyimpan alamat kota *supplier*, alamat1supp memiliki 2 kolom dimana kolom kedua adalah *nullable* yang dapat diisi jika *supplier* memiliki alamat gudang yang berbeda dari alamat kantor, termsupp adalah kolom yang menyimpan batas waktu pembayaran oleh *supplier*, dan descsupp adalah kolom *nullable* yang menyimpan keterangan *supplier*. Tabel ini memiliki relasi *one to many* dengan tabel Record Barang Masuk.

#### 8. Tabel Stok Opname

No	Nama Kolom	Tipe Data	Length
1	id	BigInt	20
2	namapenulis	string	255
3	tanggalopname	date	
4	descopname	Varchar	255

Pada tabel ini berisi segala informasi *stok opname* mulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, nama penulis merupakan kolom yang dapat menyimpan nama penulis opname, tanggal opname adalah kolom yang menyimpan tanggal berapa stok opname dilakukan dan *descopname* merupakan



kolom yang menyimpan keterangan pengecekan stok, seperti mungkin pada pengecekan stok terdapat beberapa barang yang jumlahnya kurang sesuai dengan jumlah pada sistem, pada tabel ini tidak memiliki nama pengecek karena hanya terdapat 1 karyawan yang bertugas untuk melakukan pengecekan. Tabel ini tidak memiliki relasi dengan tabel yang lain.

## 9. Tabel Kategori

Tabel 3.10 Daftar Kolom Tabel Kategori

No	Nama Kolom	Tipe Data	Length
1	id	<i>BigInt</i>	20
2	kodekat	<i>Varchar</i>	255
3	namakat	<i>Varchar</i>	255

Pada tabel ini berisi setiap informasi mengenai kategori, dimulai dari id yang dimana merupakan *primary key* dan tidak bisa di ubah, kodekat merupakan kode dari kategori dan harus tidak boleh sama, begitu pun sama dengan kolom namakat yang berfungsi untuk menyimpan nama kategori.

### 3.4 Implementasi

Implementasi sistem merupakan tahap penting dalam pengembangan sistem. Pada sub bab ini, fokus akan diberikan pada dua aspek implementasi yang mendasar, yaitu spesifikasi perangkat keras yang digunakan dan pilihan bahasa pemrograman yang diadopsi selama proses pengembangan sistem informasi inventori berbasis web.

Pemilihan perangkat lunak yang digunakan menjadi elemen kunci yang akan memengaruhi fleksibilitas dan kehandalan sistem yang dikembangkan, penulis berencana akan menggunakan bahasa pemrograman HTML, PHP dan CSS dengan menggunakan *framework* Laravel dan Bootstrap.

### 3.5 Pengujian Sistem

Langkah berikutnya yang dilakukan setelah proses pembuatan sistem selesai adalah melakukan pengujian. Tahap pengujian ini merupakan hal yang penting untuk memverifikasi bahwa sistem tidak mengandung kesalahan dan sudah sesuai dengan kebutuhan pengguna. Dalam melakukan pengujian, penulis

melakukan tiga metode pengujian yaitu *System Usability Scale (SUS)*, *User Acceptance Testing (UAT)* dan *Maze Testing*.

Proses pelaksanaan pengujian yang akan dilakukan melibatkan 4 total responden dari karyawan PT Surya Prima Sakti dengan perincian: 2 responden memiliki peran sebagai *role admin* dan 2 responden memiliki peran sebagai *role Gudang*. Untuk mendapatkan hasil pengujian SUS, penulis menggunakan bantuan dari *google form* agar mendapatkan jawaban-jawaban dari responden, sedangkan hasil *maze testing* menggunakan *tools* dari maze sendiri, untuk *User Acceptance Testing* penulis akan memberikan langkah-langkah untuk apa yang harus dilakukan oleh setiap responden yang nantinya akan di dokumentasikan melalui video. Website pada pengujian dapat di akses melalui tautan ini <https://naufalyuwan.my.id/>

