BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1. Analisis Sistem Yang Berjalan

Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikan-perbaikan. Langkah-langkah analisis sistem antara lain :

1. *Identify*, yaitu memahami masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze,* yaitu menganalisis sistem.
4. *Report,* yaitu membuat laporan hasil analisis.

Tahap analisis ini sangat penting karena dalam tahap ini apabila terdapat kesalahan, maka akan menyebabkan kesalahan terhadap tahap selanjutnya. Maka perlu tingkat ketelitian dan kecermatan yang tinggi untuk mendapatkan kualitas kerja sistem yang baik. Analisis yang sedang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem untuk dijadikan landasan usulan perancangan sistem.

4.1.1. Analisis Dokumen

Analisis Dokumen ini dilakukan guna untuk mengetahui dokumen apa saja yang menjadi *input*, proses, dan *output* dari sistem ini. Dokumen *input* adalah

dokumen yang akan diproses oleh sistem yang biasanya dilakukan oleh entitas luar sistem. Dokumen Proses adalah dokumen yang diperlukan oleh sistem dalam melakukan kegiatan pendafataran dan pengajuan, sedangkan yang menjadi Dokumen *Output* adalah dokumen yang dihasilkan oleh proses olahan sistem, yang berasal dari dalam sistem ke entitas luar. Dan dokumen-dokumen yang digunakan sebenarnya cukup banyak, baik itu yang berupa data dalam bentuk simpanan (*database*) maupun yang bisa dijadikan laporan maupun dokumen dalam bentuk fisik (masukan dan keluaran). Namun yang akan dibahas dan dijelaskan dalam analisa ini adalah dokumen yang dirasa cukup sering digunakan dan diperlukan dalam aktivitas penjualan di toko ikan hias TAN Karawang, dokumen tersebut diantaranya adalah :

1. Pembelian barang
2. Penjualan barang

Tabel 4.1.

Analisis Dokumen

No	Nama Dokumen	Uraian
1	Pembelian Barang	a. Deskripsi : Dokumen ini merupakan dokumen yang berisikan informasi membuat list barang yang ingin di pesan b. Fungsi : Sebagai kumpulan informasi yang bertujuan untuk membantu bagian pembelian dalam mengecek barang yang datang dari suplier dan membuat laporan barang yang sudah masuk

		untuk di teruskan pada pemilik toko c. Sumber : *Suplier* d. Distribusi : Bagian Pembelian e. Atribut : nomor_barang, satuan, qty, harga_satuan, diskon, harga, sub_total, ekstra_diskon, cash_diskon, total, ppn, ongkos_angkut, grand_total, terbilang, catatan
2	Penjualan barang	a. Deskripsi : Dokumen ini merupakan dokumen yang berisikan informasi menginput barang yang di di jual kepada konsumen b. Fungsi : membuat laporan penjualan barang yang terjual ke konsumen, membuat laporan keuangan penjualan barang, membuat struk penjualan c. Sumber : Konsumen d. Distribusi : *Kasir* e. Atribut : nomor_barang, nama_barang, satuan, qty, harga_satuan, diskon, harga, sub_total, ekstra_diskon, cash_diskon, total, ppn, ongkos_angkut, grand_total, terbilang, catatan

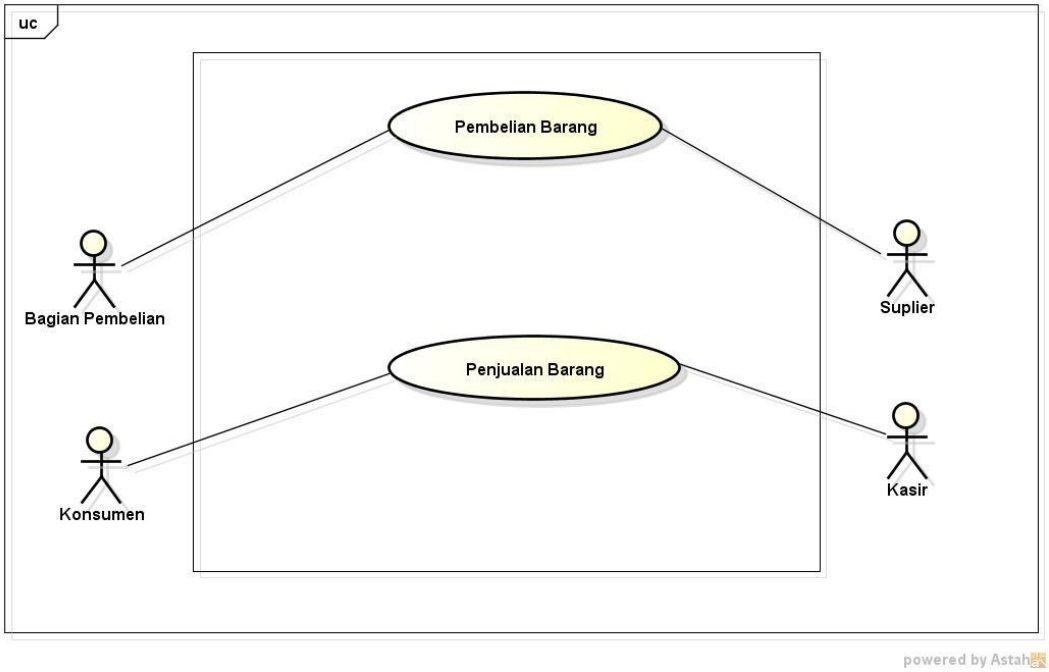
4.1.2. Analisis Prosedur Yang Sedang Berjalan

Analisis sistem yang sedang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem.

Analisis sistem ini dapat dijadikan sebagai suatu landasan usulan perancangan sistem yang sedang berjalan yang dilakukan berdasarkan urutan kejadian yang ada serta dari urutan kejadian tersebut. Berikut merupakan analisis sistem yang sedang berjalan.

4.1.2.1. *Usecase Diagram*

Usecase Diagram dari sistem yang sedang berjalan, digambarkan sebagai berikut :



Gambar 4.1. *Usecase Diagram* Sistem Yang Sedang Berjalan

Dalam *usecase diagram* diatas, dapat terlihat terdapat 2 proses inti yang sedang berjalan di Toko Ikan Hias TAN Karawang. Untuk lebih jelasnya dapat dijabarkan sebagai berikut :

1. Proses pertama, yaitu pembelian barang dimana Bagian pembelian sebagai *Actor utama* yang membuat list barang yang ingin di pesan dari *Suplier*, peran dari *actor* berikut nya *Suplier* yaitu mengkonfirmasikan barang yang ada dan memberikan langsung barang yang dipesan ke Bagian Pembelian. Lalu Bagian Pembelian menerima barang dan mengecek barang yang datang, lalu Bagian pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko.
2. Proses kedua, yaitu penjualan barang dimana Konsumen memberikan barang yang ingin di beli kepada *Kasir* sebagai *actor* yang menginputkan data barang yang dijual, lalu menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen. Lalu Konsumen menerima informasi dan melakukan pembayaran. Lalu *Kasir* membuat struk penjualan. Lalu Konsumen menerima struk penjualan.

4.1.2.2. Definisi Aktor dan Deskripsinya

Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Berikut adalah deskripsi pendefinisian aktor pada Sistem Informasi Penjualan Ikan Hias di Toko TAN Karawang.

Tabel 4.2

Definisi Aktor dan Deskripsinya

NO	AKTOR	DESKRIPSI
1	Bagian Pembelian	Orang yang membuat daftar permintaan barang kepada supplier dan mengecek barang yang masuk.
2	Supplier	Orang yang menyediakan barang untuk di jual ke toko.
3	Kasir	Orang yang bertugas menerima daftar barang yang dibeli oleh konsumen dan mengurus pembayaran barang yang dibeli konsumen.
4	Konsumen	Orang yang membeli barang dari toko.

4.1.2.3. Definisi Use Case dan Deskripsinya

Use Case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Berikut adalah deskripsi pendefinisian *use case* pada Sistem Informasi Penjualan Ikan Hias di Toko TAN Karawang.

Table 4.3

Definisi Use Case dan Deskripsinya

No	Use Case	Deskripsi
1	Pembelian Barang	Merupakan proses barang masuk dari supplier yang akan di jual di toko ini.
2	Penjualan Barang	Merupakan proses barang keluar atau barang yang terjual di toko kepada konsumen.

4.1.2.4. Skenario *Activity Diagram*

a) Skenario *Activity Diagram* Sistem Pembelian Barang Yang Sedang Berjalan

1. Nama *Usecase* : Pembelian Barang
2. Aktor : Bagian Pembelian dan *Suplier*
3. Tipe : *Primary*
4. Tujuan : Membuat laporan barang yang sudah masuk untuk di teruskan kepada pemilik toko

Tabel 4.4.

Skenario Pembelian Barang Yang Sedang Berjalan

No : A01	
Nama	Pembelian barang
Tujuan	membuat laporan barang yang sudah masuk untuk di teruskan kepada pemilik toko
Deskripsi	
Tipe	*Primary*
Aktor	Bagian Pembelian dan *Suplier*
Skenario Utama	
Kondisi Awal	-
Aksi Aktor	**Reaksi Sistem**
1. Bagian Pembelian membuat list barang yang ingin dipesan	
2. Bagian Pembelian memberikan data list barang yang dipesan pada *Suplier*	

3. *Suplier* mengkonfirmasi barang yang ada dan memberikan langsung barang yang dipesan ke Bagian Pembelian	
4. Bagian Pembelian menerima barang tersebut dan mengecek barang yang datang	
5. Bagian pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko	
Kondisi Akhir	Pembelian Barang terselesaikan dengan baik

b) Skenario *Activity Diagram* Sistem Penjualan BarangYang Sedang Berjalan

1. Nama *Usecase* : Penjualan Barang
2. Aktor : *Kasir,* Konsumen
3. Tipe : *Primary*
4. Tujuan : Agar laporan barang yang dijual terinputkan dengan baik

Tabel 4.5.

Skenario Penjualan Barang Yang Sedang Berjalan

No : A02	
Nama	Penjualan Barang
Tujuan	Agar laporan barang yang dijual terinputkan dengan baik

Deskripsi		
Tipe	*Primary*	
Aktor	*Kasir,* Konsumen	
Skenario Utama		
Kondisi Awal	-	
Aksi Aktor		**Reaksi Sistem**
1. Konsumen memberikan barang yang ingin dibeli		
2. *Kasir* menginputkan barang yang dijual		
3. *Kasir* menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen		
4. Konsumen menerima informasi dan melakukan pembayaran		
5. *Kasir* membuat struk penjualan		
6. Kosumen menerima struk penjualan		
Kondisi Akhir		Laporan penjualan barang terinputkan dengan baik

4.1.2.5. *Activity Diagram*

1. *Activity Diagram* Sistem Pengolahan Pembelian Barang Yang Sedang Berjalan

Proses pengolahan Pembelian Barang di Toko Ikan Hias TAN Karawang secara umum dapat dijelaskan sebagai berikut :

a) Bagian Pembelian akan membuat list barang.

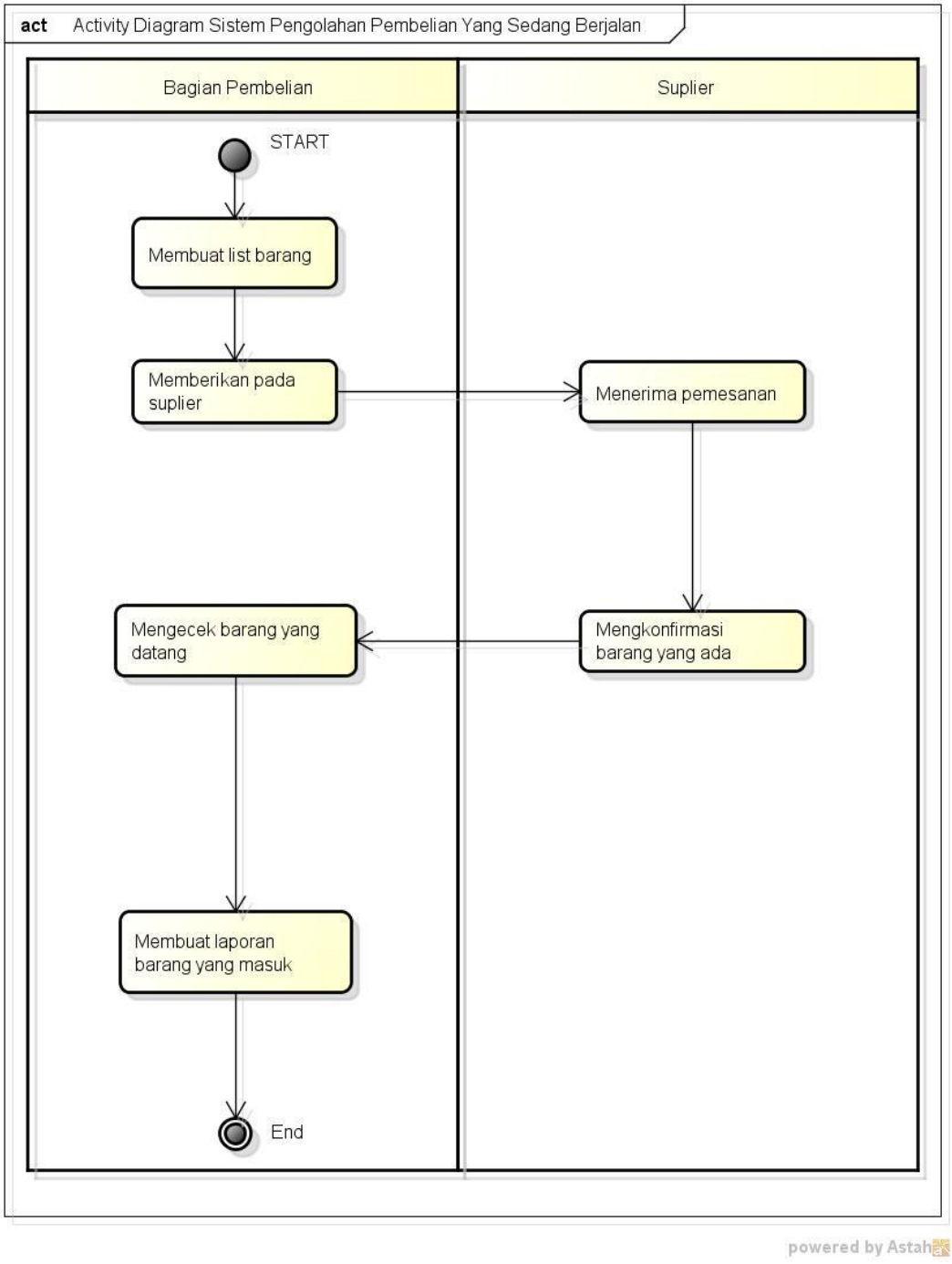
b) Lalu Bagian Pembelian memberikan kepada *Suplier*.

c) Lalu *Suplier* menerima pemesanan.

d) Dan kemudian *Suplier* akan mengkonfirmasi.

e) Lalu Bagian Pembelian mengecek barang yang datang.

f) Dan kemudian Bagian Pembelian memebuat laporan barang yang masuk.

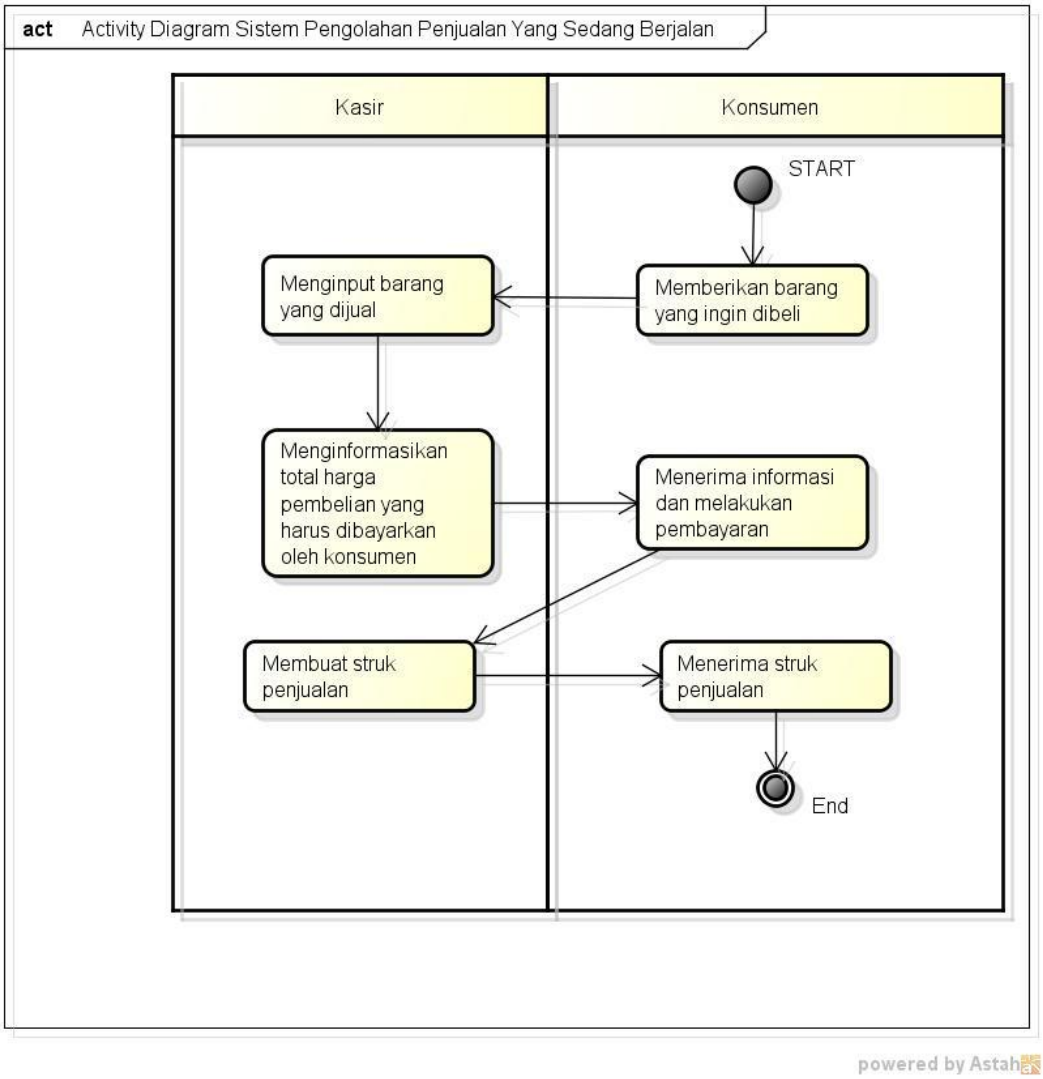


Gambar 4.2. *Activity Diagram* Pengolahan Pembelian Yang Sedang Berjalan

2. *Activity Diagram* Sistem Pengolahan Penjualan Yang Sedang Berjalan

Proses pengolahan penjualan di Toko Ikan Hias TAN Karawang secara umum dapat dijelaskan sebagai berikut :

a) Konsumen memberikan barang yang ingin dibeli
b) Lalu *Kasir* menginput barang yang dijual
c) Dan kemudian *Kasir* menginformasikan total harga pembelian yang harus dibayar oleh konsumen
d) Lalu Konsumen menerima informasi dan melakukan pembayaran
e) Lalu *Kasir* membuat struk penjualan
f) Lalu Konsumen menerima struk penjualan



Gambar 4.3. *Activity Diagram* Pengolahan Penjualan Yang Sedang Berjalan

4.1.3. Kesimpulan Analisis Prosedur Yang Sedang Berjalan

Berdasarkan penjabaran analisis mengenai sistem yang sedang berjalan diatas, dapat terlihat bahwa sistem-sistem tersebut masih sangat sederhana dan berjalan secara manual. Manual yang di maksud pengerjaan laporan dari mulai pembelian sampai penjulan masih menggunakan sistem tulis tangan. Sehingga kendalanya ada pada *human erorr* yaitu kesalahan dari manusia nya itu sendiri

mencatat segala jenis laporan, sehingga dapat mengganggu proses pembelian dan penjualan di Toko Ikan Hias TAN Karawang.

Belum adanya sistem komputerisasi yang dapat mengefisienkan waktu serta ke akuratan mengolah dan menginputkan data laporan pembelian dan penjualan barang. Solusi nya dengan menerapkan Program Aplikasi khusus yang mampu menangani sistem-sistem tersebut agar terintegrasi dengan baik di Toko Ikan Hias TAN Karawang.

Dari analisis tersebut, maka perlunya dibuatkan suatu Program Aplikasi yang dapat mengintegrasikan sistem pengolahan dan penginputkan data secara komputerisasi agar dapat mengefisienkan waktu dan mengakuratkan semua jenis laporan. Program Aplikasi tersebut akan memudahkan Bagian Pembelian dan *Suplier* dalam mengolah data pembelian barang dapat terkontrol dengan baik serta memudahkan Konsumen memilih membeli barang yang ingin dibeli dan *Kasir* mengolah data penjualan barang. Semua kegiatan sistem-sistem yang sedang berjalan tersebut dapat tercakup, diakses dan diselesaikan melalui Program Aplikasi yang akan dibuat.

4.2. Perancangan Sistem

Adapun tujuan dan gambaran dari perancangan sistem yang dikembangkan, yaitu:

4.2.1. Tujuan Perancangan

Perancangan sistem ini dilakukan setelah tahap analisis sistem berjalan dikerjakan, dalam hal ini telah didapatkan gambaran dengan jelas apa yang harus dikerjakan. Pada dasarnya tahap perancangan ini bertujuan untuk:

1. Untuk memenuhi kebutuhan kepada pemakai sistem.
2. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemogram komputer dan ahli-ahli teknik lainnya seputar program aplikasi yang akan dibuat.

4.2.2. Gambaran Umum Sistem Yang Diusulkan

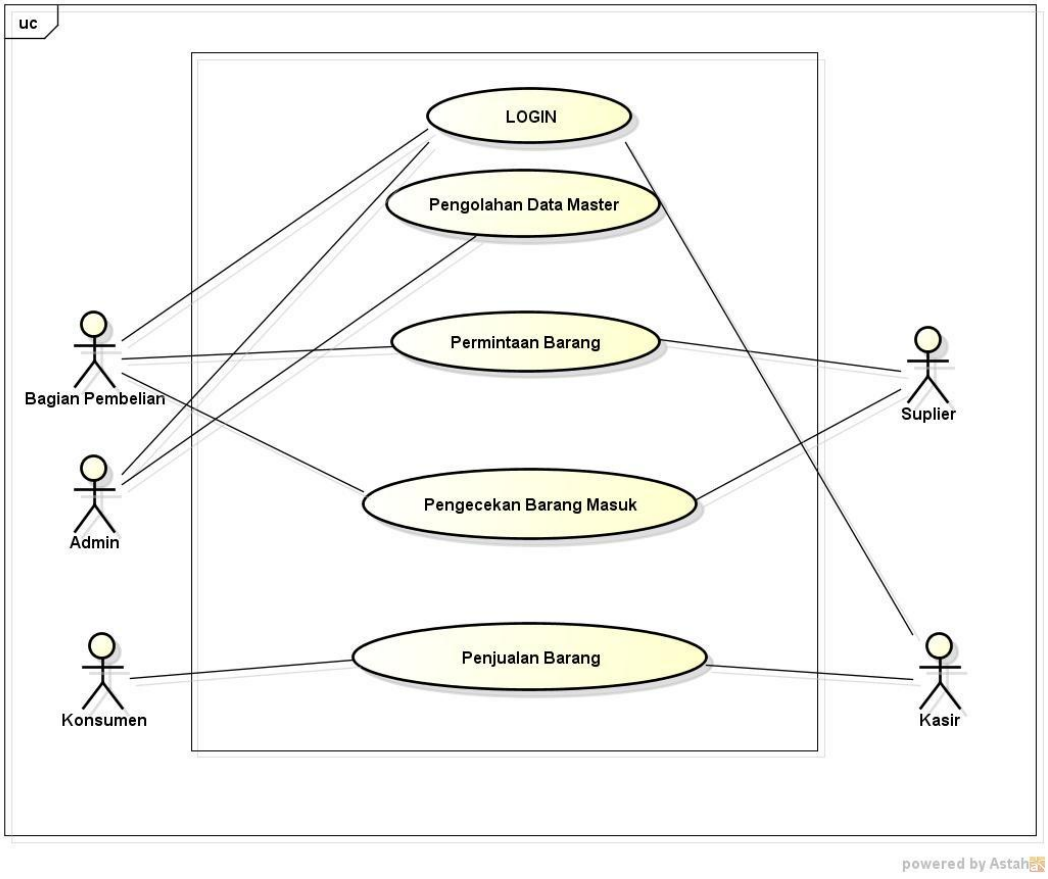
Perancangan aplikasi pembantu dan pendukung kinerja karyawan ini menggunakan bahasa UML (*Unified Modeling Language*). Mulai dari pembuatan rancangan *Usecase* nya hingga *Deployment* Diagramnya. Secara umum, proses tersebut dimulai dari penentuan arsitektur utama dari sistem yang ingin dirancang dan dibuat diagram alur proses pendaftaran sebagai Admin (Administrator/Operator) yang mana dalam kasus ini adalah Admin itu sendiri dan dilanjutkan alur proses utamanya, yang berupa pengolahan data master, pembelian barang, penjualan barang, retur pembelian, retur penjualan.

4.2.3. Perancangan Prosedur Yang Diusulkan

Usulan atau rancangan dari analisis sistem yang akan dikembangkan adalah sebagai berikut :

4.2.3.1. *Usecase Diagram*

Usecase diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah *Usecase* mempresentasikan sebuah interaksi antara Aktor dengan sistem. Berikut adalah perancangan *Usecase* nya:



Gambar 4.4. *Usecase Diagram* Sistem Yang Diusulkan

Dalam *usecase diagram* diatas, dapat terlihat terdapat 5 proses inti yang akan diusulkan di Toko Ikan Hias TAN Karawang. Untuk lebih jelasnya dapat dijabarkan sebagai berikut :

3. Proses pertama, yaitu aktivitas *login* yang terlebih dahulu harus dilakukan oleh Admin, Bagian Pembelian, dan Kasir untuk dapat masuk ke Program Aplikasi ini.
4. Proses kedua, yaitu pengolahan data dimana Admin sebagai *Actor* yang intinya Admin menginputkan data-data master yaitu data master barang, data karyawan, data konsumen, dan data suplier
5. Proses ketiga, yaitu yaitu permintaan barang dimana Bagian pembelian sebagai *Actor utama* yang membuat list permintaan barang sebagai informasi bagian pembelian yang akan diminta ke banyak supplier. peran dari *actor* berikut nya *Suplier* yaitu mengkonfirmasikan barang yang ada dan memberikan langsung barang yang dipesan ke Bagian Pembelian. Lalu Bagian Pembelian menerima barang dan mengecek barang yang datang, lalu Bagian pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko.
6. Proses keempat, yaitu pengecekan barang masuk, mengecek data dari list permintaan barang sama atau tidak dengan barang masuk dari suplier
7. Dan terakhir yaitu proses kelima, yaitu penjualan barang dimana Konsumen memberikan barang yang ingin di beli kepada *Kasir* sebagai *actor* yang menginputkan data barang yang dijual, lalu menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen. Lalu Konsumen menerima

informasi dan melakukan pembayaran. Lalu *Kasir* membuat struk penjualan. Lalu Konsumen menerima struk penjualan.

4.2.3.2. Definisi Aktor Yang di Usulkan

Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Berikut adalah deskripsi pendefinisian aktor yang di usulkan pada Sistem Informasi Penjualan Ikan Hias di Toko TAN Karawang.

Tabel 4.6.
Definisi Aktor dan Deskripsinya Yang Diusulkan

NO	AKTOR	DESKRIPSI
1	Admin / Pemilik	Orang yang mengelola data-data seperti data barang, data karyawan, data supplier yang akan di gunakan di program yang di usulkan.
2	Bag. Pembelian	Orang yang membuat daftar permintaan barang kepada supplier dan mengecek barang yang masuk.
3	Supplier	Orang yang menyediakan barang untuk di jual ke toko.
4	Kasir / Bag. Penjualan	Orang yang bertugas menerima daftar barang yang dibeli oleh konsumen dan mengurus pembayaran barang yang dibeli konsumen.
5	Konsumen	Orang yang membeli barang dari toko.

4.2.3.3. Definisi Use Case Yang di Usulkan

Use Case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Berikut adalah deskripsi pendefinisian *use case* yang di usulkan pada Sistem Informasi Penjualan Ikan Hias di Toko TAN Karawang.

Tabel 4.6.

Definisi Aktor dan Deskripsinya Yang Diusulkan

NO	USE CASE	DESKRIPSI
1	Pengolahan Data Master	Merupakan proses penginputan data-data seperti data barang, data karyawan, data supplier yang akan di gunakan di program yang di usulkan.
2	Permintaan Barang	Merupakan proses penginputan daftar barang yang ingin di beli dari supplier.
3	Pengecekan Barang Masuk	Merupakan proses pengecekan barang yang masuk ke gudang dari supplier.
4	Penjualan Barang	Merupakan proses barang keluar atau barang yang terjual di toko kepada konsumen.

4.2.3.4. Skenario *Diagram*

Skenario yang diusulkan adalah sebagai berikut :

1. Skenario *Diagram Login* Yang Diusulkan
   a. Nama *Usecase* : *Login.*
   b. Aktor : Admin, Bagian Pembelian, Kasir.
   c. Tipe : *Primary.*
   d. Tujuan : Agar dapat masuk dan mengakses Program Aplikasi.

Tabel 4.7.

Skenario *Login* Yang Diusulkan

No : A01		
Nama	*Login*	
Tujuan	Agar dapat masuk dan mengakses Program Aplikasi	
Deskripsi		
Tipe	*Primary*	
Aktor	Admin, Bagian Pembelian, dan Kasir	
Skenario Utama		
Kondisi Awal	-	
Aksi Aktor		**Reaksi Sistem**
6. Admin, Bagian Pembelian, dan Kasir memasukkan *username* dan *password*		
		7. Sistem akan

	mengecek *username* dan *password*
8. Mendapatkan pesan dari sistem : a) Apabila *username / password* tidak *valid*, maka memasukkan *username/ password* lagi atau keluar dari program aplikasi b) Apabila *username/ password valid*, maka Admin, Bagian Pembelian, dan Kasir dapat mengakses Program Aplikasi	
Kondisi Akhir	Admin, Bagian Pembelian, dan Kasir dapat masuk dan mengakses Program Aplikasi

2. Skenario *Diagram* Sistem Pengolahan Data Master Yang Diusulkan

a. Nama *Usecase* : Pengolahan Data Master.

b. Aktor : Admin.

c. Tipe : *Primary.*

d. Tujuan : Menginputkan data-data master yaitu data master barang, data karyawan, data konsumen, dan data suplier.

Tabel 4.8.

Skenario Pengolahan Data Master Yang Diusulkan

<table>
<tr><td colspan="3">No : A02</td></tr>
<tr><td>Nama</td><td colspan="2">Pengolahan Data Master</td></tr>
<tr><td>Tujuan</td><td colspan="2">Menginputkan data-data master yaitu data master barang, data karyawan, data konsumen, dan data suplier.</td></tr>
<tr><td colspan="3">Deskripsi</td></tr>
<tr><td>Tipe</td><td colspan="2">Primary</td></tr>
<tr><td>Aktor</td><td colspan="2">Admin</td></tr>
<tr><td colspan="3">Skenario Utama</td></tr>
<tr><td>Kondisi Awal</td><td colspan="2">-</td></tr>
<tr><td colspan="2">Aksi Aktor</td><td>Reaksi Sistem</td></tr>
<tr><td colspan="2">1. Admin menginputkan data-data master di program aplikasi.</td><td></td></tr>
<tr><td colspan="2"></td><td>2. Sistem dapat menyimpan, mengubah,menghapus, mencari data-data master.</td></tr>
<tr><td colspan="2">Kondisi Akhir</td><td>Data Master tersimpan dengan baik.</td></tr>
</table>

3. Skenario *Diagram* Permintaan Barang Yang Diusulkan

   a. Nama *Usecase* : Permintaan Barang.

   b. Aktor : Bagian Pembelian dan *Suplier.*

   c. Tipe : *Primary.*

   d. Tujuan : Membuat list permintaan barang yang ingin di pesan kepada banyak suplier.

Tabel 4.9.

Skenario Permintaan Barang Yang Diusulkan

No : A03	
Nama	Permintaan Barang
Tujuan	Membuat list permintaan barang yang ingin di pesan kepada banyak suplier
Deskripsi	
Tipe	*Primary*
Aktor	Bagian Pembelian dan *Suplier*
Skenario Utama	
Kondisi Awal	-
Aksi Aktor	**Reaksi Sistem**
7. Bagian Pembelian membuat list barang yang ingin dipesan (inputkan ke program)	
	8. Sistem akan menyimpan

	data tersebut dan akan mengeluarkan nya sebagai laporan
9. Bagian Pembelian memberikan data list barang yang dipesan pada *Suplier*	
Kondisi Akhir	List permintaan barang terdaftar dengan baik

4. Skenario Diagram Pengecekan Barang Masuk Yang Diusulkan

   a. Nama *Usecase* : Pengecekan Barang Masuk.

   b. Aktor : Bagian Pembelian dan *Suplier.*

   c. Tipe : *Primary.*

   d. Tujuan : Mengecek barang yang masuk sama atau tidak dengan list permintaan barang.

Tabel 4.10.

Skenario Pengecekan Barang Masuk Yang Diusulkan

No : A04	
Nama	Pengecekan Barang Masuk
Tujuan	Mengecek barang yang masuk sama atau tidak dengan list permintaan barang
Deskripsi	
Tipe	*Primary*

Aktor	Bagian Pembelian dan *Suplier*
Skenario Utama	
Kondisi Awal	-
Aksi Aktor	**Reaksi Sistem**
1. Suplier mengkonfirmasi barang yang ada dan meberikan langsung barang yang di pesan Bagian Pembelian	
2. Bagian Pembelian menerima barang yang ada dan mengecek barang yang datang	
3. Bagian pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko	
	4. Sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan
Kondisi Akhir	Barang yang dipesan masuk ke stock gudang

5. Skenario *Diagram* Penjualan Barang Yang Diusulkan

a. Nama *Usecase* : Penjualan Barang.

b. Aktor : Konsumen dan *Kasir*.

c. Tipe : *Primary*.

d. Tujuan : Agar laporan yang dijual terinputkan dengan baik.

Tabel 4.11.

Skenario Penjualan Barang Yang Diusulkan

No : A05		
Nama	Penjualan Barang	
Tujuan	Agar laporan yang dijual terinputkan dengan baik	
Deskripsi		
Tipe	*Primary*	
Aktor	Konsumen dan *Kasir*	
Skenario Utama		
Kondisi Awal	-	
Aksi Aktor		**Reaksi Sistem**
1. Konsumen memberikan barang yang ingin dibeli		
2. *Kasir* menginputkan barang yang dijual		

3. *Kasir* menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen	
4. Konsumen menerima informasi dan melakukan pembayaran	
	5. Sistem akan menyimpan data penjualan dan akan mengeluarkan struk
	6. Konsumen Menerima struk penjualan
Kondisi Akhir	Laporan penjualan barang terinputkan dengan baik

4.2.3.5. *Activity Diagram*

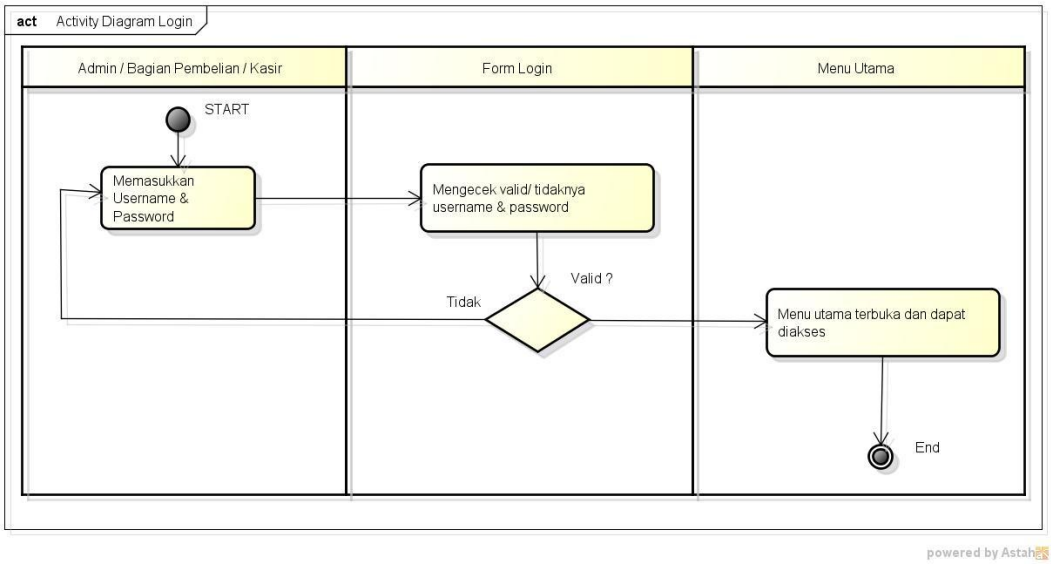
Activity Diagram yang diusulkan adalah sebagai berikut :

1. *Activity Diagram Login* Yang Diusulkan

Proses login yang diusulkan secara umum dapat dijelaskan sebagai berikut:

a. Admin, Bagian Pembelian, dan Kasir memasukkan *username* dan *password*
b. Lalu di Form Login, Sistem akan mengecek *username* dan *password*
c. Apabila tidak valid, form login akan memberikan pesan, dan *Project Leader* atau karyawan dapat memasukkan kembali *Username* dan *Password* atau dapat keluar dari Program Aplikasi.

d. Apabila *valid*, maka masuk ke manu utama dan dapat mengakses Program Aplikasi tersebut.



Gambar 4.5. *Activity Diagram* Login Yang Diusulkan

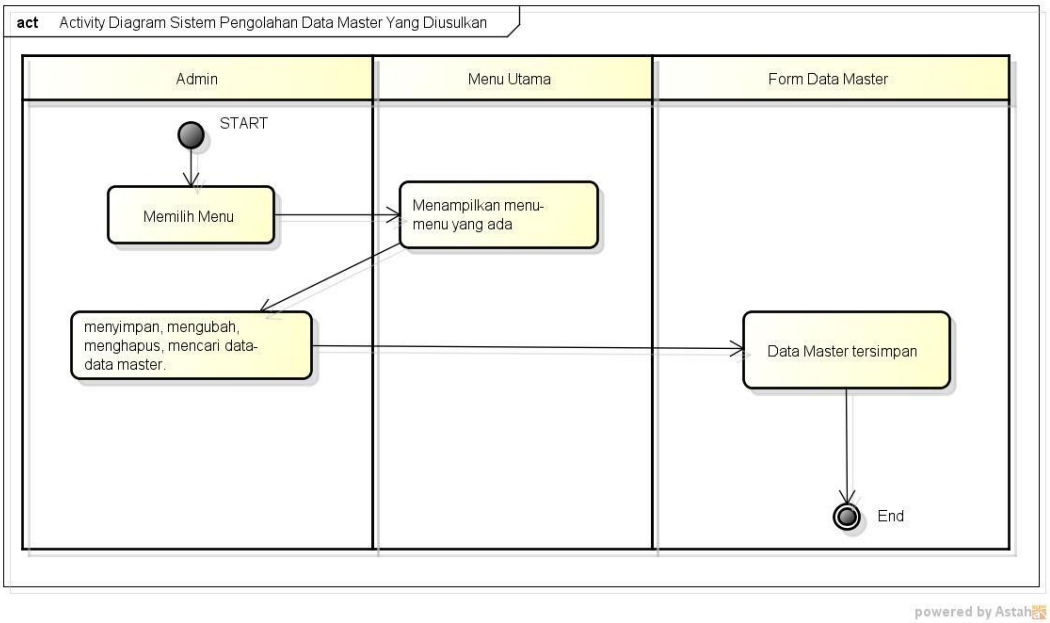
2. Activity Diagram Sistem Pengolahan Data Master Yang Diusulkan

Proses Pengolahan Data Master yang diusulkan secara umum dapat dijelaskan sebagai berikut:

a. Admin menginputkan data-data master di program aplikasi.

b. Sistem dapat menyimpan, mengubah,menghapus, mencari data-data master.

c. Data master tersimpan.

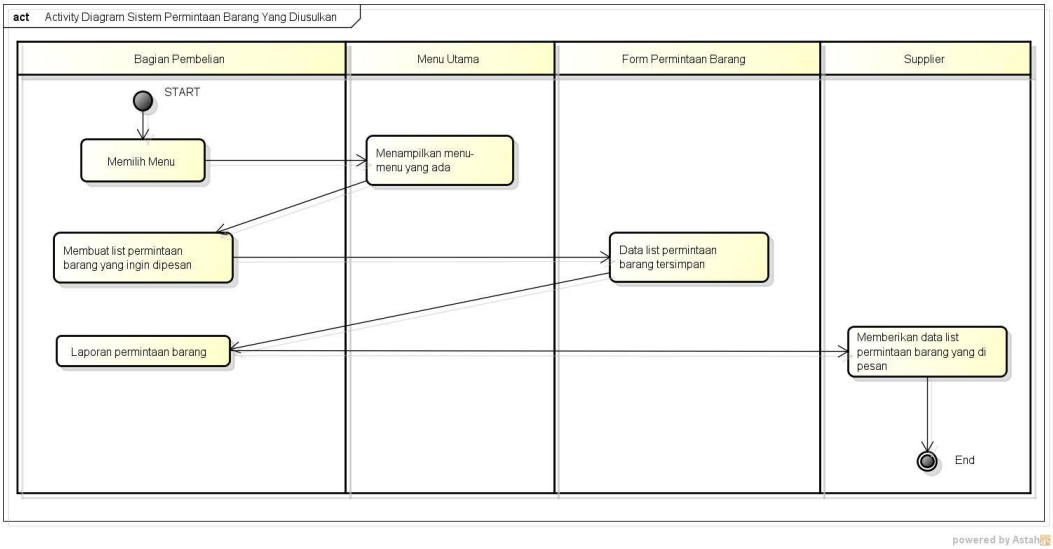


Gambar 4.6. *Activity Diagram* Pengolahan Data Master Yang Diusulkan

3. *Activity Diagram* Sistem Permintaan Barang Yang Diusulkan

Proses Permintaan Barang yang diusulkan secara umum dapat dijelaskan sebagai berikut:

a. Bagian Pembelian membuat list barang yang ingin dipesan (inputkan ke program).
b. Sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan.
c. Bagian Pembelian memberikan data list barang yang dipesan pada *Suplier*.

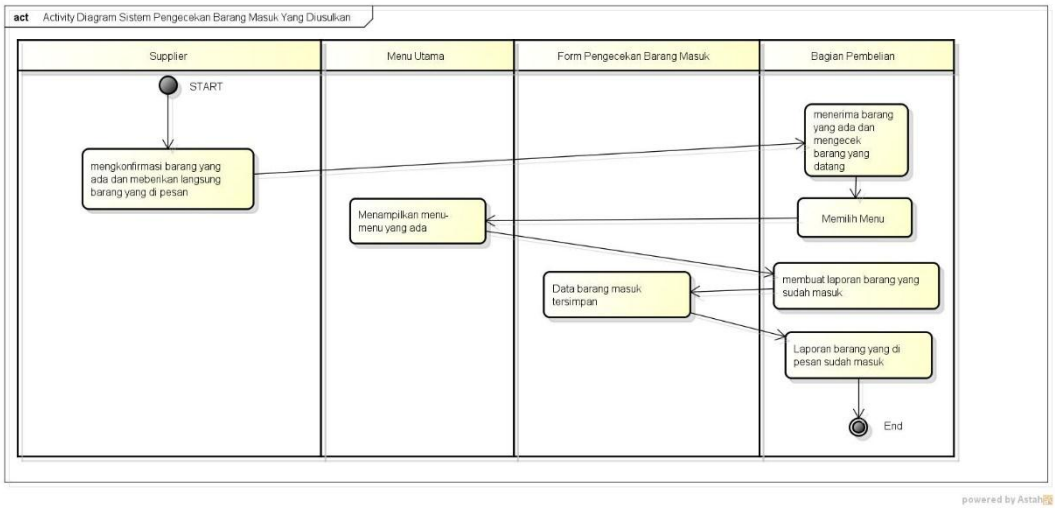


Gambar 4.7. *Activity Diagram* Permintaan Barang Yang Diusulkan

4. *Activity Diagram* Sistem Pengecekan Barang Masuk Yang Diusulkan

Proses Penegecekan Barang Masuk yang diusulkan secara umum dapat dijelaskan sebagai berikut:

a. Suplier mengkonfirmasi barang yang ada dan meberikan langsung barang yang di pesan Bagian Pembelian
b. Bagian Pembelian menerima barang yang ada dan mengecek barang yang datang
c. Bagian pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko.
d. Sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan.

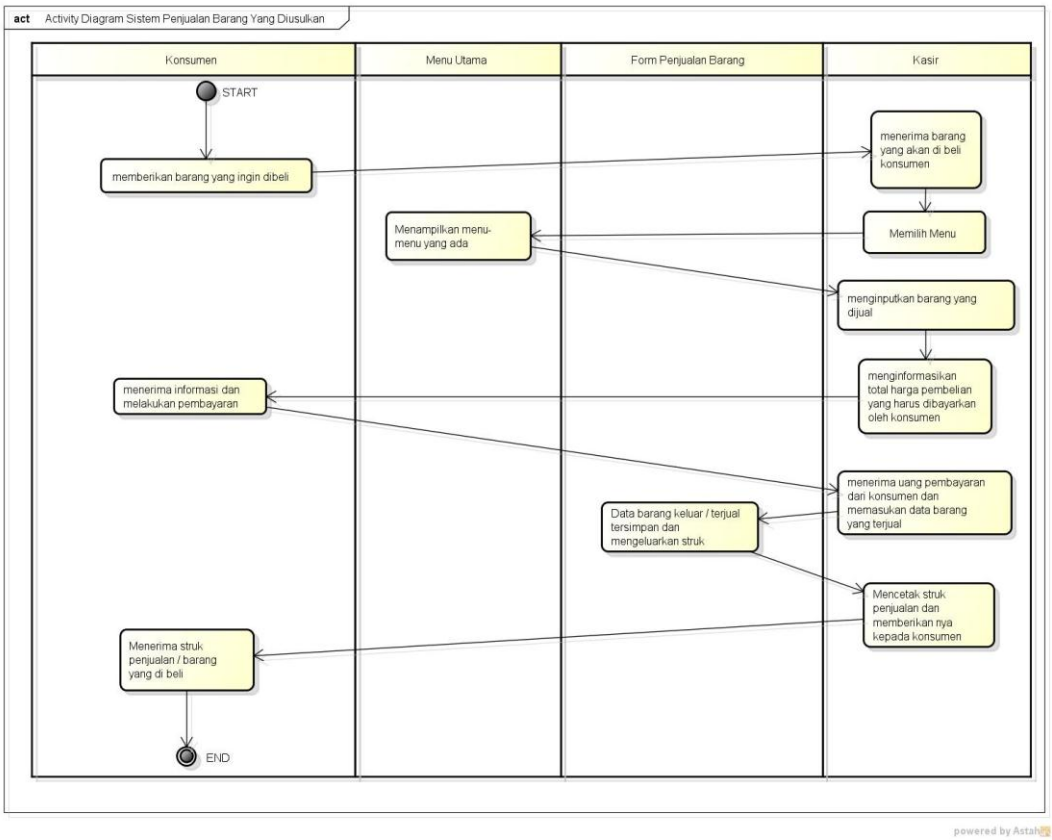


Gambar 4.8. *Activity Diagram* Pengecekan Barang Masuk Yang Diusulkan

5. *Activity Diagram* Sistem Penjualan Barang Yang Diusulkan

Proses Penjualan Barang yang diusulkan secara umum dapat dijelaskan sebagai berikut:

a. Konsumen memberikan barang yang ingin dibeli.
b. *Kasir* menginputkan barang yang dijual.
c. *Kasir* menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen.
d. Konsumen menerima informasi dan melakukan pembayaran.
e. Sistem akan menyimpan data penjualan dan akan mengeluarkan struk.
f. Konsumen Menerima struk penjualan.



Gambar 4.9. *Activity Diagram* Penjualan Barang Yang Diusulkan

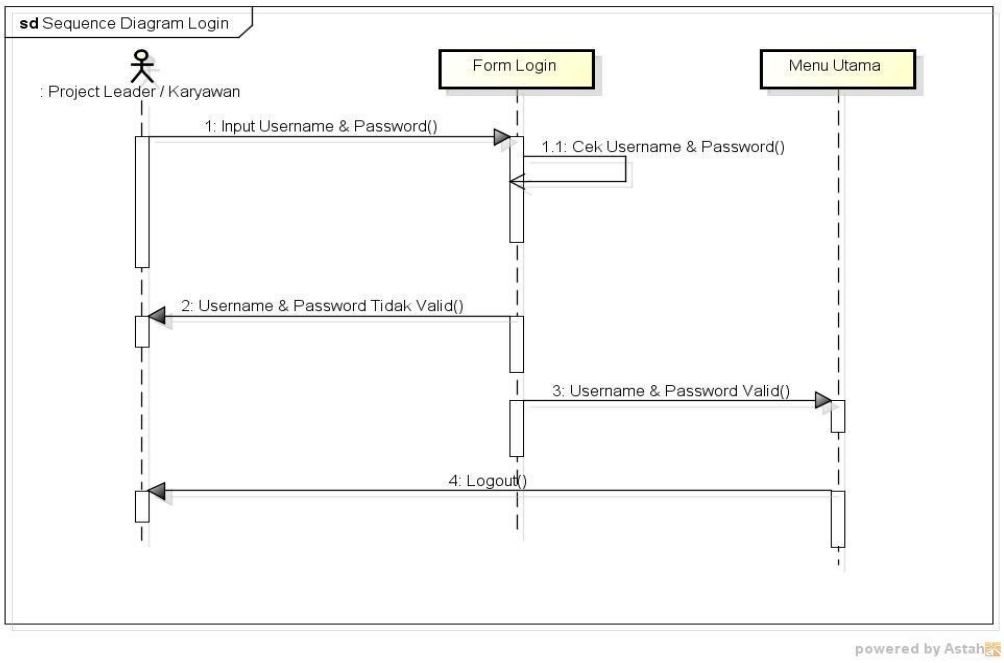
4.2.3.6. *Sequence Diagram*

Sequence Diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan *output* tertentu. *Diagram* ini menunjukkan sejumlah contoh obyek dan *message* yang di letakkan di antara objek-objek di dalam usecase. Komponen utama *sequence diagram* terdiri dari objek yang di tulisakan dengan kotak segi empat bernama. *Message* di wakili oleh garis dengan tanda panah dan waktu yang di tunjukkan dengan progress *vertical*. Diawali dari apa yang

mentriggerkan aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

1. *Sequence Diagram Login*

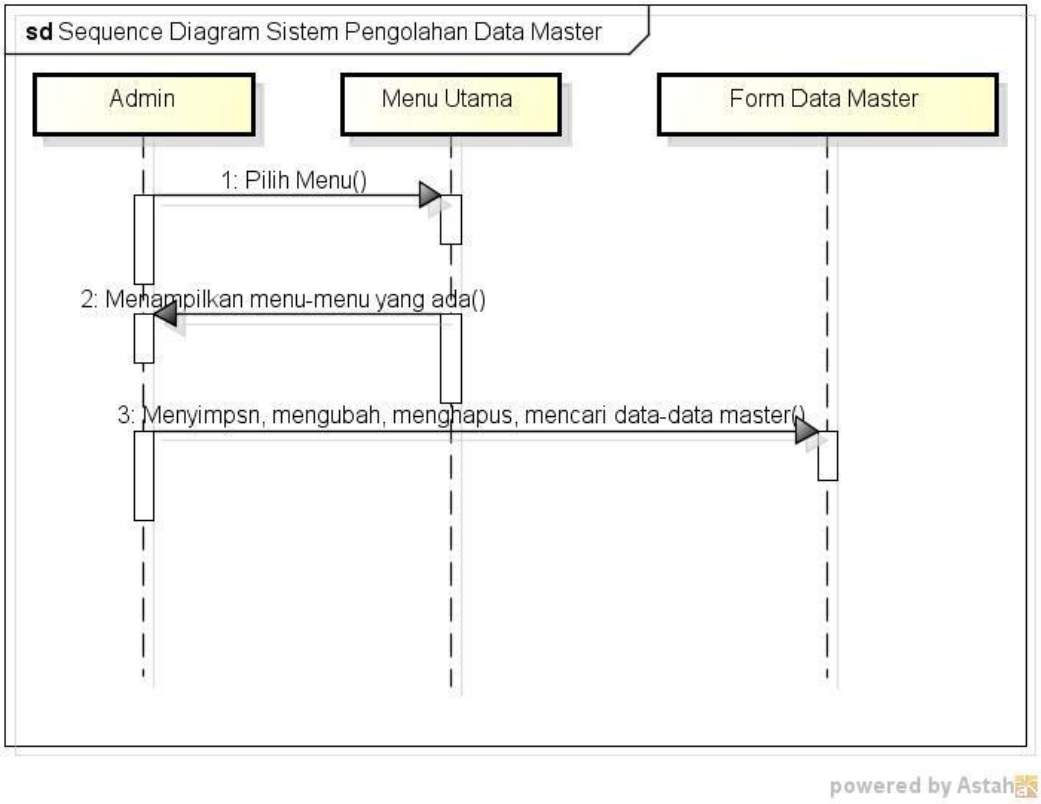
Gambar dibawah ini merupakan *Sequence Diagram Login* yang menggambarkan proses yang sedang terjadi dalam *Login* Admin, bagian pembelian atau kasir. Prosesnya dimulai dengan admin masuk kedalam *form login* terlebih dahulu, kemudian selanjutnya Admin, bagian pembelian atau kasir dapat melakukan *login* dengan memasukkan *Username* dan *Password* untuk dapat masuk kedalam sistem/ menu utama. Proses *Login* itu sendiri diikuti juga validasi apabila Admin, bagian pembelian atau kasir salah memasukan *Username* dan *Password.* Dan diakhirkan dengan proses *logout*.



Gambar 4.10. *Sequence Diagram Login* Yang Diusulkan

2. *Sequence Diagram* Pengolahan Data Master

Gambar dibawah ini merupakan *Sequence Diagram* pengolahan data master, dimana Admin menginputkan data-data master di program aplikasi, lalu sistem dapat menyimpan, mengubah, mencari data-data master. Apabila sudah data master akan tersimpan.

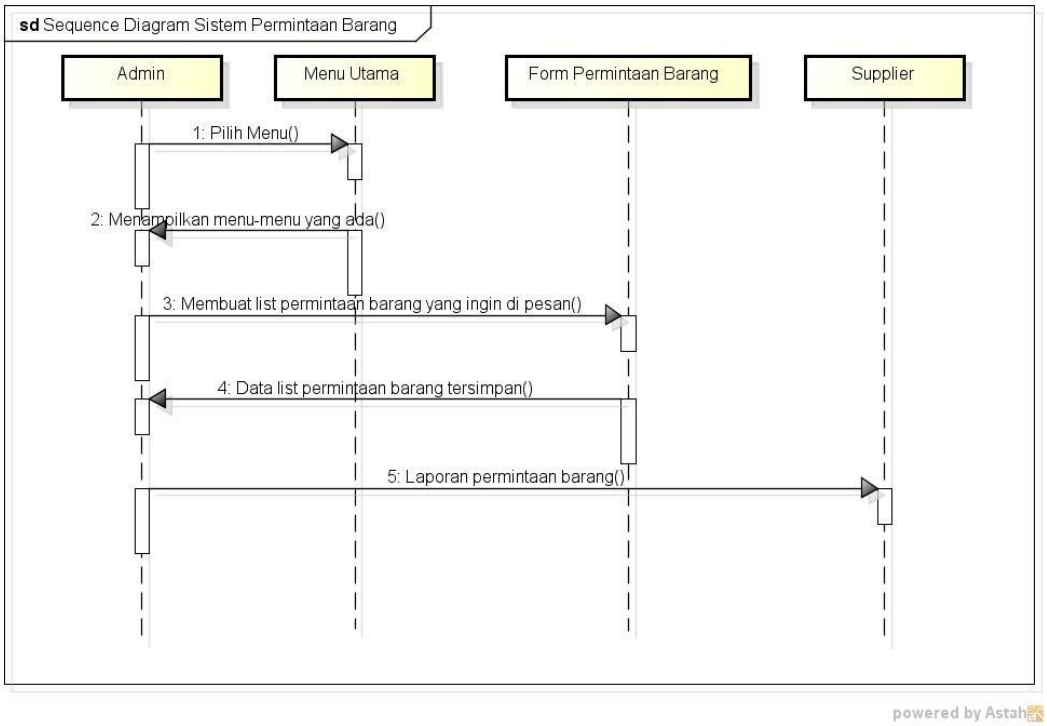


Gambar 4.11. *Sequence Diagram* Pengolahan Data Master Yang Diusulkan

3. *Sequence Diagram* Permintaan Barang

Gambar dibawah ini merupakan *Sequence Diagram* Permintaan Barang, dimana Bagian Pembelian membuat list barang yang ingin di pesan lalu di inputkan ke program, selanjutnya sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan. Apabila data tersebut sudah menjadi laporan

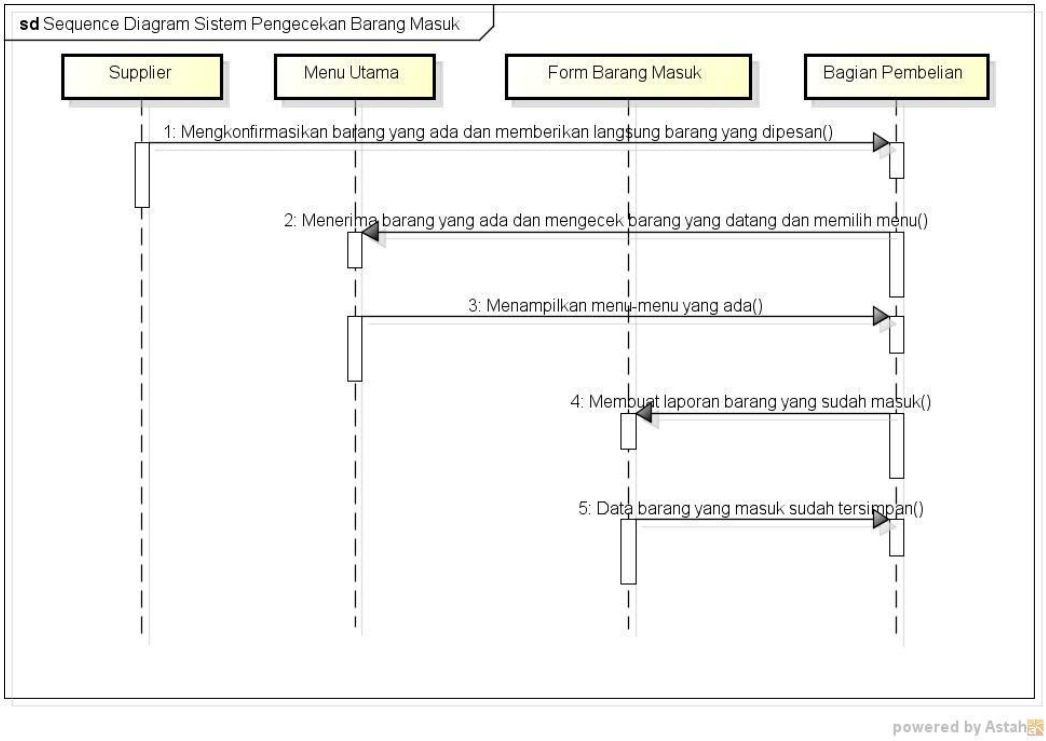
maka Bagian Pembelian akan memberikan data list barang yang dipesan kepada *Supplier*.



Gambar 4.12 *Sequence Diagram* Permintaan Barang Yang Diusulkan

4. *Sequence Diagram* Pengecekan Barang Masuk

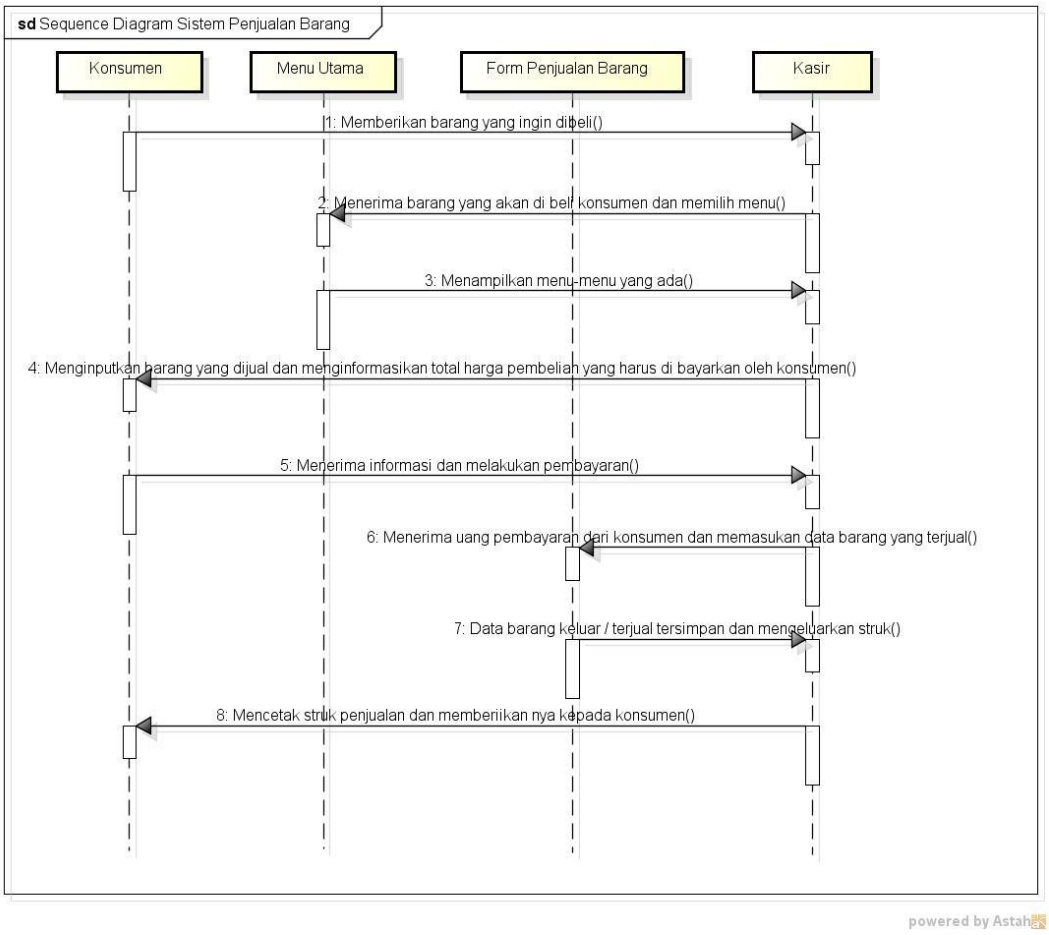
Gambar dibawah ini merupakan *Sequence Diagram* Ppengecekan barang masuk, dimana *Supplier* mengkonfirmasikan barang yang ada dan memberikan langsung barang yang dipesan Bagian Pembelian, lalu kemudian Bagian Pembelian menerima barang yang ada dan mengecek barang yang datang, setelah itu Bagian Pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko. Apabila laporan barang yang sudah masuk sudah di berikan kepada pemilik toko maka sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan.



Gambar 4.13. *Sequence Diagram* Pengecekan Barang Masuk Yang Diusulkan

5. *Sequence Diagram* Penjualan Barang

Gambar dibawah ini merupakan *Sequence Diagram* penjualan barang, dimana konsumen memberikan barang yang ingin di beli, lalu kasir menginputkan barang yang dijual dan menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen, lalu konsumen menerima informasi itu dan melakukan pembayaran. Apalbila konsumen telah melakukan pembayaran maka sistem akan menyimpan data penjualan dan akan mengeluarkan struk, setelah itu konsumen akan menerima struk penjualan barang yang dibeli nya.



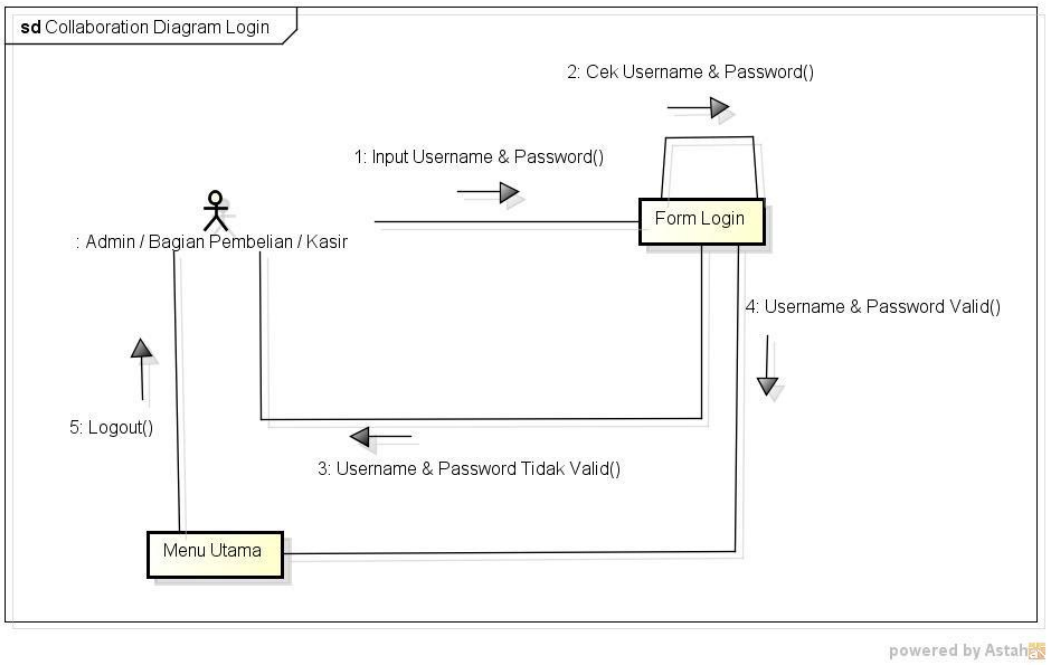
Gambar 4.14. *Sequence Diagram* Penjualan Barang Yang Diusulkan

4.2.3.7. *Collaboration Diagram*

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masingmasing objek dan bukan pada waktu penyampaian message. Setiap message memiliki sequence number, di mana *message* dari level tertinggi memiliki nomor 1. *Messages* dari level yang sama memiliki prefiks yang sama.

1. *Collaboration Diagram Login*

Collaboration diagram Login, menggambarkan skenario dari Aplikasi pembantu dan pendukung kinerja karyawan di Toko Ikan Hias TAN Karawang antara Admin / Bagian Pembelian / Kasir dengan sistem dalam proses *login.* Dalam hal ini Admin / Bagian Pembelian / Kasir harus menginputka *Username* dan *Password* terlebih dahulu untuk masuk kedalam sistem. Berikut gambarnya :

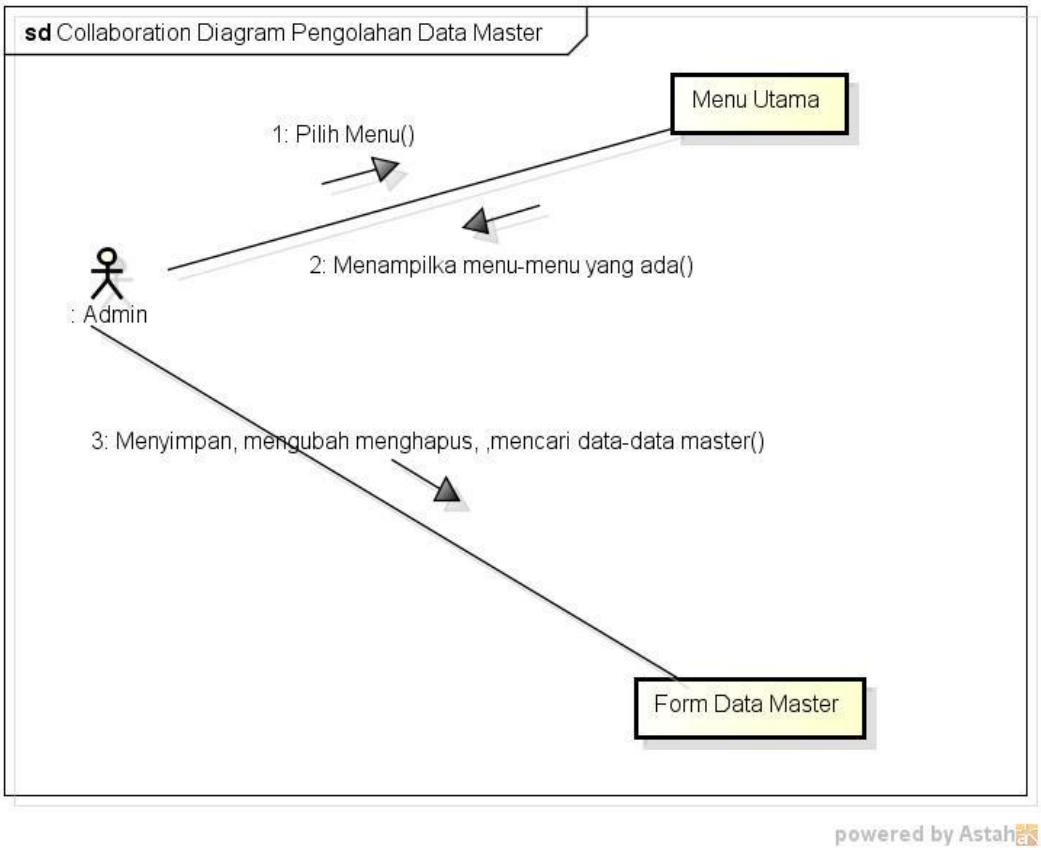


Gambar 4.15. *Collaboration Diagram Login* Yang Diusulkan

2. *Collaboration Diagram* Pengolahan Data Master

Collaboration diagram Pengolahan Data Master menggambarkan skenario dari Aplikasi pembantu dan pendukung kinerja karyawan di Toko Ikan Hias TAN Karawang antara Admin dengan sistem. Dalam hal ini Admin menginputkan data-data master di program aplikasi, lalu sistem dapat menyimpan, mengubah,

mencari data-data master. Apabila sudah data master akan tersimpan. Berikut gambarnya :

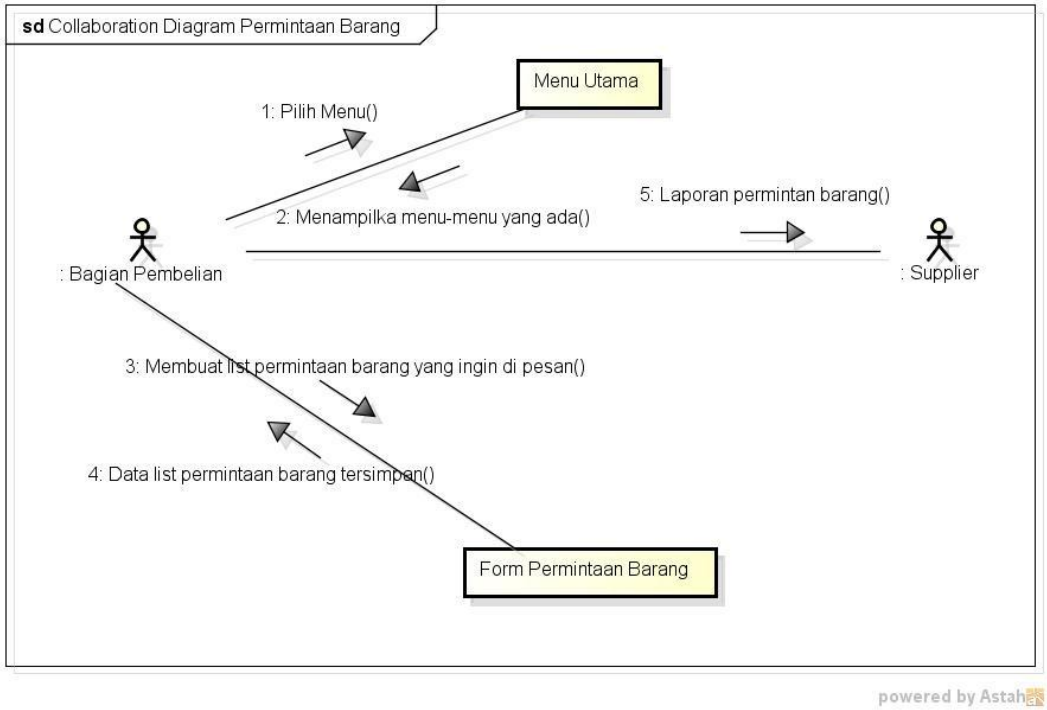


Gambar 4.16. *Collaboration Diagram* Pengolahan Data Master Yang Diusulkan

3. *Collaboration Diagram* Permintaan Barang

Collaboration diagram permintaan barang, menggambarkan skenario dari Aplikasi pembantu dan pendukung kinerja karyawan di Toko Ikan Hias TAN Karawang antara Bagian Pembelian, Supplier dengan sistem. Dalam hal ini Bagian Pembelian membuat list barang yang ingin di pesan lalu di inputkan ke program, selanjutnya sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan. Apabila data tersebut sudah menjadi laporan

maka Bagian Pembelian akan memberikan data list barang yang dipesan kepada *Supplier*. Berikut gambarnya :

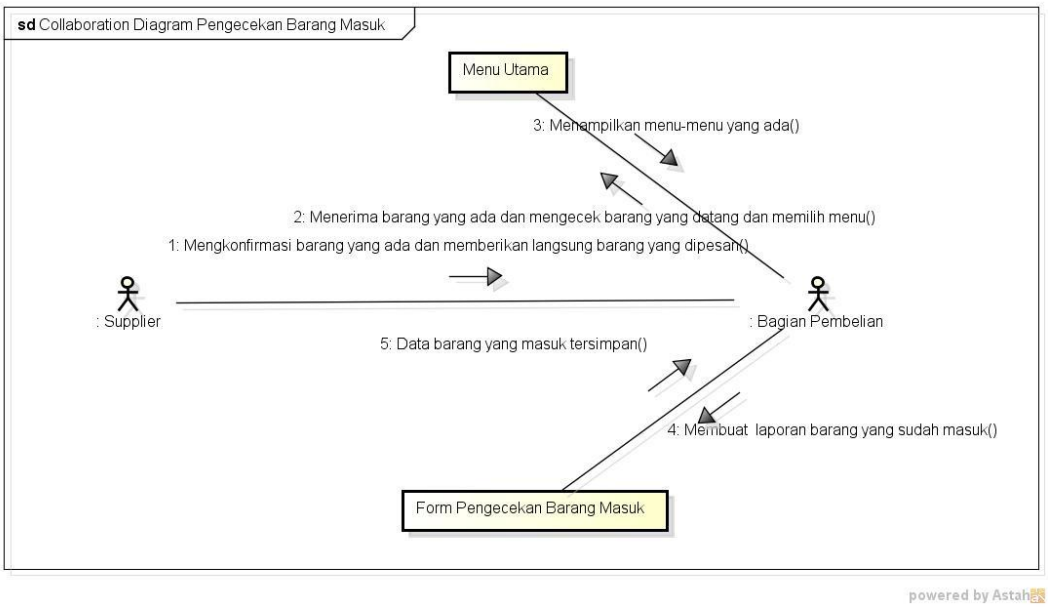


Gambar 4.17. *Collaboration Diagram* Permintaan Barang Yang Diusulkan

4. *Collaboration Diagram* Pengecekan Barang Masuk

Collaboration diagram pengecekan barang masuk, menggambarkan skenario dari Aplikasi pembantu dan pendukung kinerja karyawan di Toko Ikan Hias TAN Karawang antara *Supplier,* Bagian Pembelian dengan sistem. Dalam hal ini *Supplier* mengkonfirmasikan barang yang ada dan memberikan langsung barang yang dipesan Bagian Pembelian, lalu kemudian Bagian Pembelian menerima barang yang ada dan mengecek barang yang datang, setelah itu Bagian Pembelian membuat laporan barang yang sudah masuk tersebut untuk di teruskan kepada pemilik toko. Apabila laporan barang yang sudah masuk sudah di berikan

kepada pemilik toko maka sistem akan menyimpan data tersebut dan akan mengeluarkan nya sebagai laporan. Berikut gambarnya :

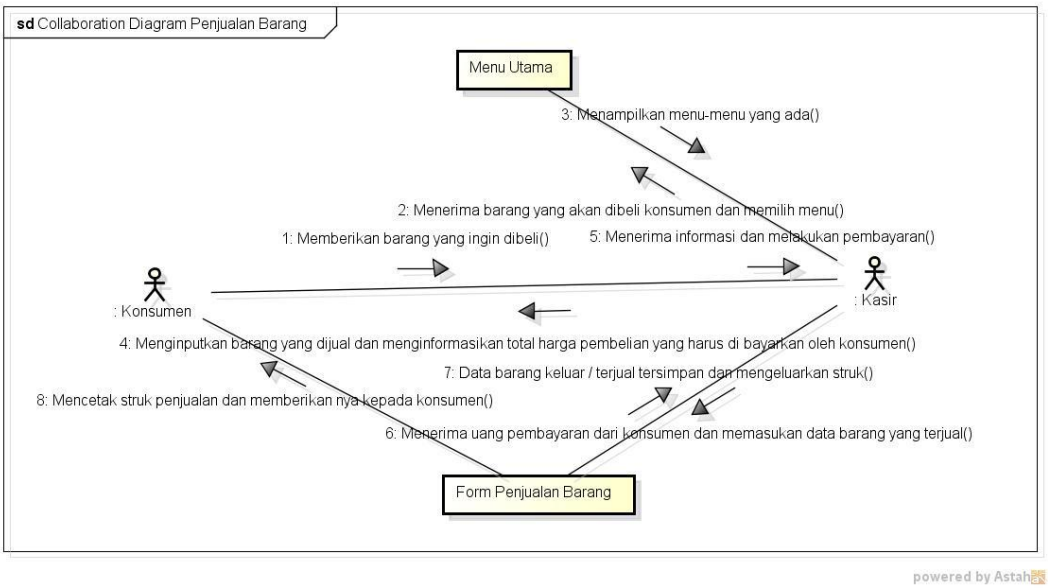


Gambar 4.18. *Collaboration Diagram* Pengecekan Barang Masuk Yang Diusulkan

5. *Collaboration Diagram* Penjualan Barang

Collaboration diagram Penjualan Barang, menggambarkan skenario dari Aplikasi pembantu dan pendukung kinerja karyawan di Toko Ikan Hias TAN Karawang antara Konsumen, Kasir dengan sistem. Dalam hal ini konsumen memberikan barang yang ingin di beli, lalu kasir menginputkan barang yang dijual dan menginformasikan total harga pembelian yang harus dibayarkan oleh konsumen, lalu konsumen menerima informasi itu dan melakukan pembayaran. Apalbila konsumen telah melakukan pembayaran maka sistem akan menyimpan

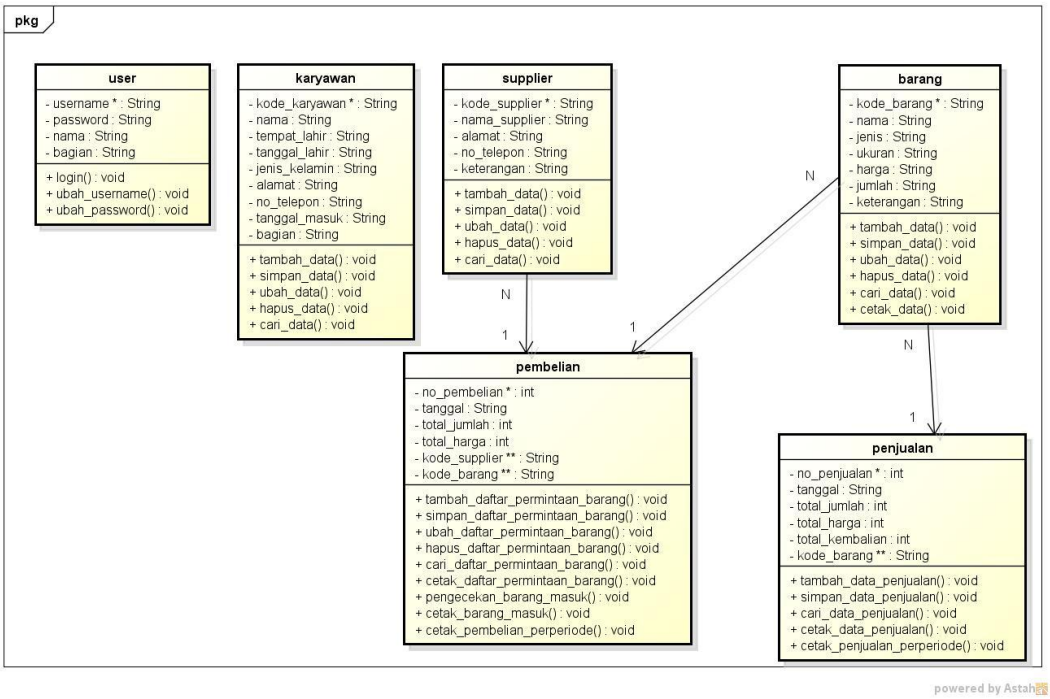
data penjualan dan akan mengeluarkan struk, setelah itu konsumen akan menerima struk penjualan barang yang dibeli nya. Berikut gambarnya :



Gambar 4.19. *Collaboration Diagram* Penjualan Barang Yang Diusulkan

4.2.3.8. *Class Diagram*

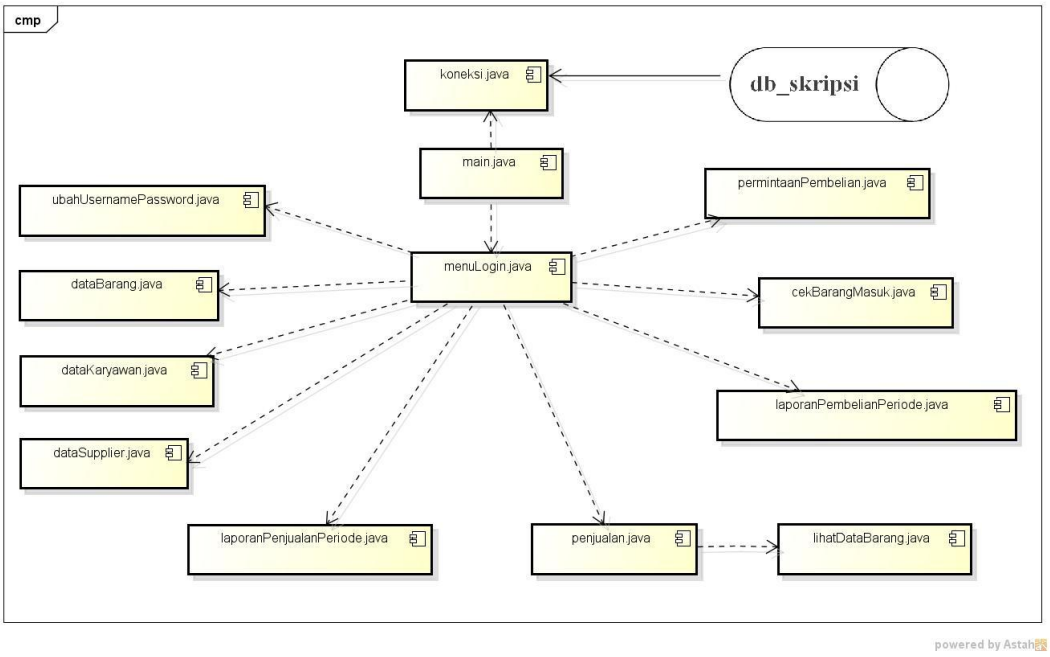
Class Diagram adalah sebuah spesifikasi yang jika di instansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class Diagram* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaam tersebut (metoda/fungsi). *Class Diagram* menggambarkan struktur dan deskripsi *Class*, *Package*, dan dan *Object* beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Berikut adalah *Class Diagram* yang dibutuhkan dalam perancangan Program Aplikasi Sistem Penjualan Ikan Hias di Toko Ikan Hias TAN Karawang :



Gambar 4.20. *Class Diagram* Yang Diusulkan

4.2.3.9. *Component Diagram*

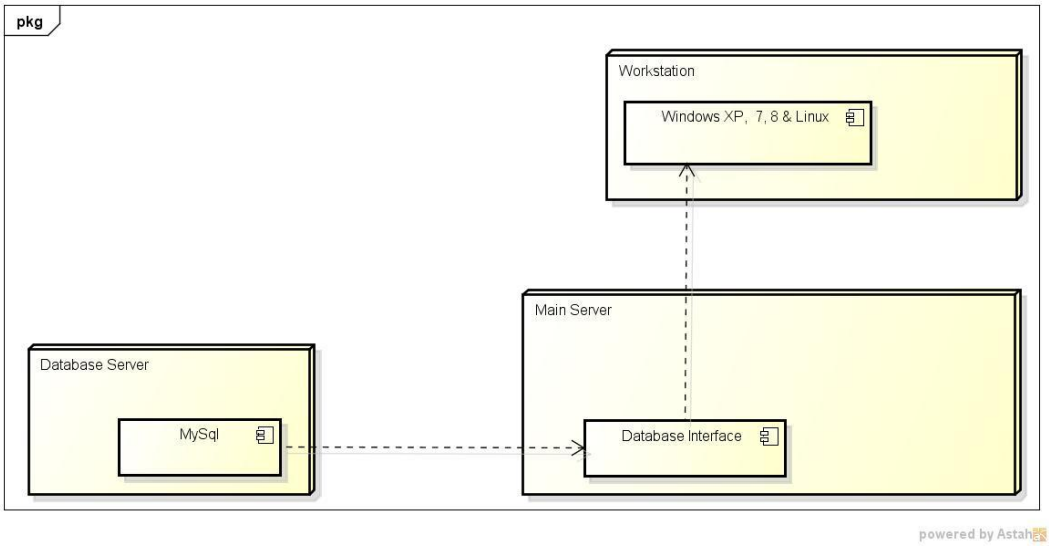
Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun run *time*. Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain. Berikut gambar *Component diagram* yang diusulkan :



Gambar 4.21. *Component Diagram* Yang Diusulkan

4.2.3.10. *Deployment Diagram*

Deployment/ physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisikal. Sebuah *node* adalah *server*, *workstation*, atau piranti keras lain yang digunakan untuk men-deploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya *TCP*/ *IP*) dan *requirement* dapat juga didefinisikan dalam diagram ini. . Berikut gambar *Deployment diagram* yang diusulkan :



Gambar 4.22 *Deployment Diagram* Yang Diusulkan

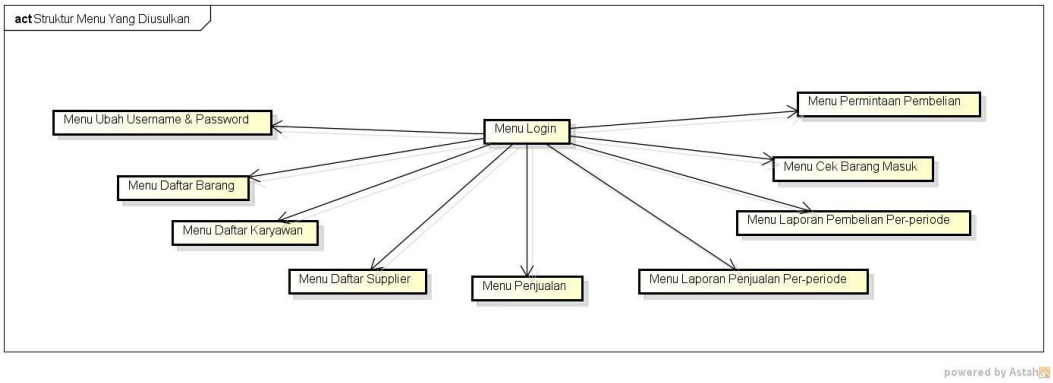
4.2.4. Perancangan Antar Muka

Pada perancangan antar muka atau *interface* ini akan menggambarkan rancangan tampilan yang ada dalam sistem yang diusulkan. Adapun yang akan dibahas adalah perancangan input dan perancangan *output*.

4.2.4.1. Struktur *Menu*

Struktur *menu* adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer sehingga pada saat menjalankan program dikomputer, *user* tidak mengalami kesulitan lagi dalam memilih menu nyang diinginkan.

Struktur menu yang diusulkan adalah sebagai berikut :



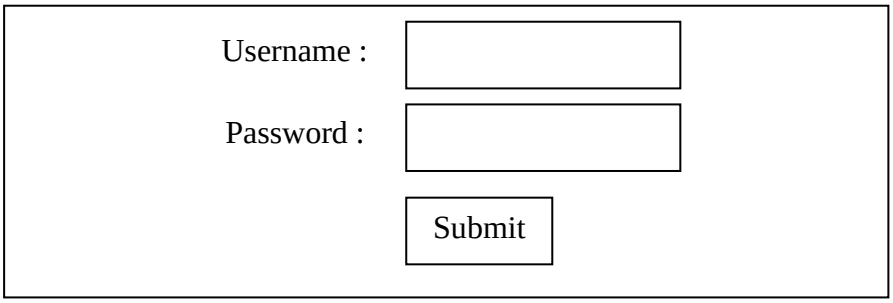
Gambar 4.23. Struktur *Menu* Yang Diusulkan

4.2.4.2. Perancangan *Input*

Perancangan ini dipergunakan bagi pengguna yang berkepentingan untuk menggunakan program aplikasi. Bagi yang menggunakan program aplikasi ini maka terlebih dahulu harus memasukkan *username* dan *password*.

1. Login

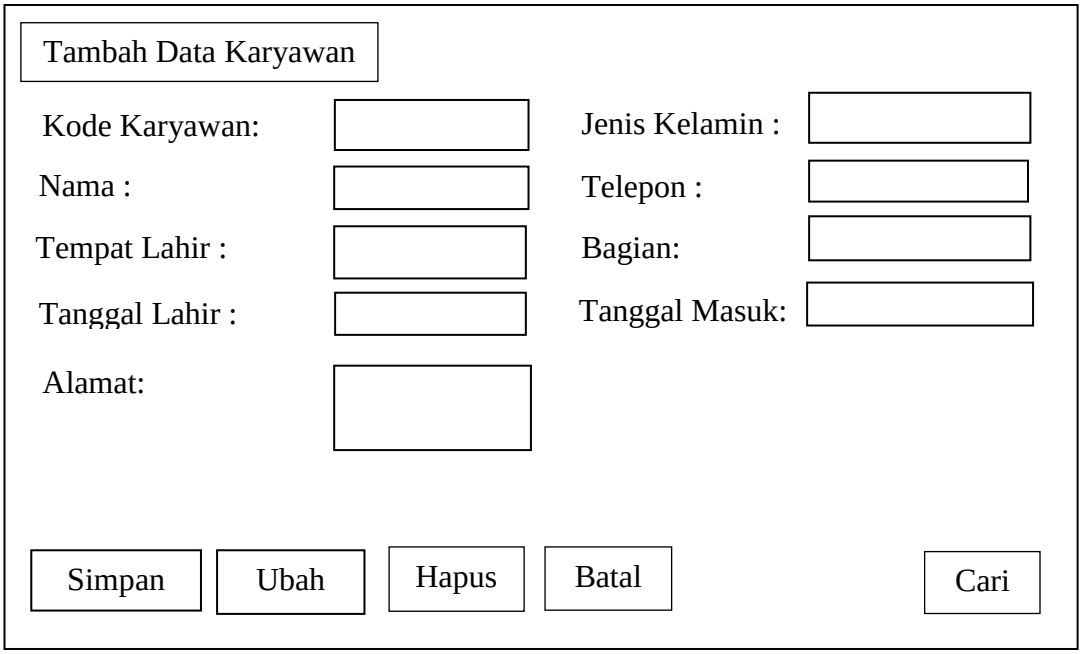
Tombol *Submit* di *Form* Inputan *Login* akan menvalidasi langsung ke *database* sistem, dimana tombol *Submit* diatas akan menampilkan pesan validasi *username* dan *password* dari *user*.



Gambar 4.24. *Form* Inputan *Login* Yang Diusulkan

2. Pendaftaran Data Karyawan

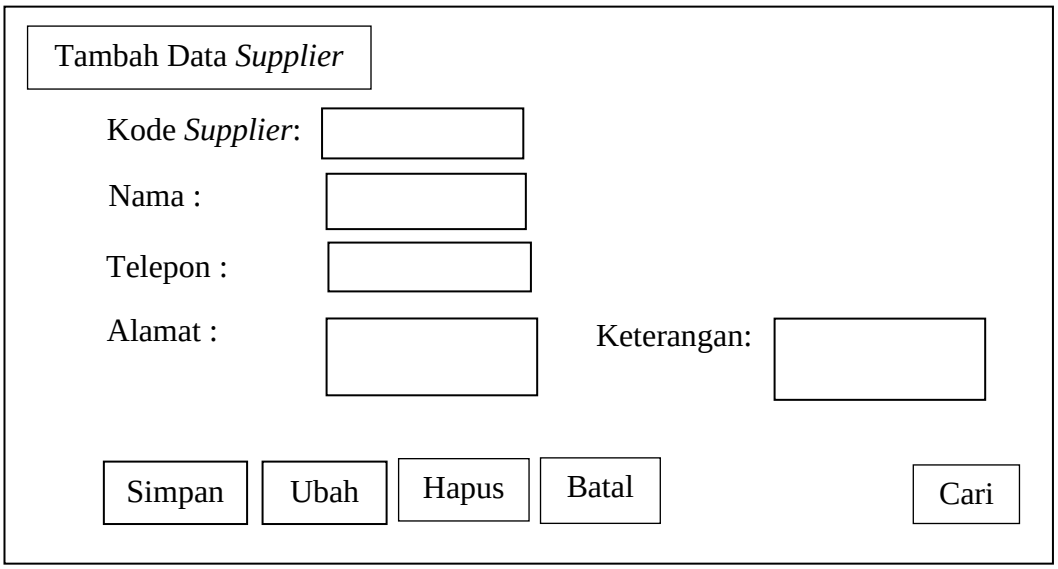
Tombol *Save* akan menyimpan data Data Karyawan langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apablia *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan.



Gambar 4.25. *Form* Inputan Data Karyawan Yang Diusulkan

3. Pendaftaran Data *Supplier*

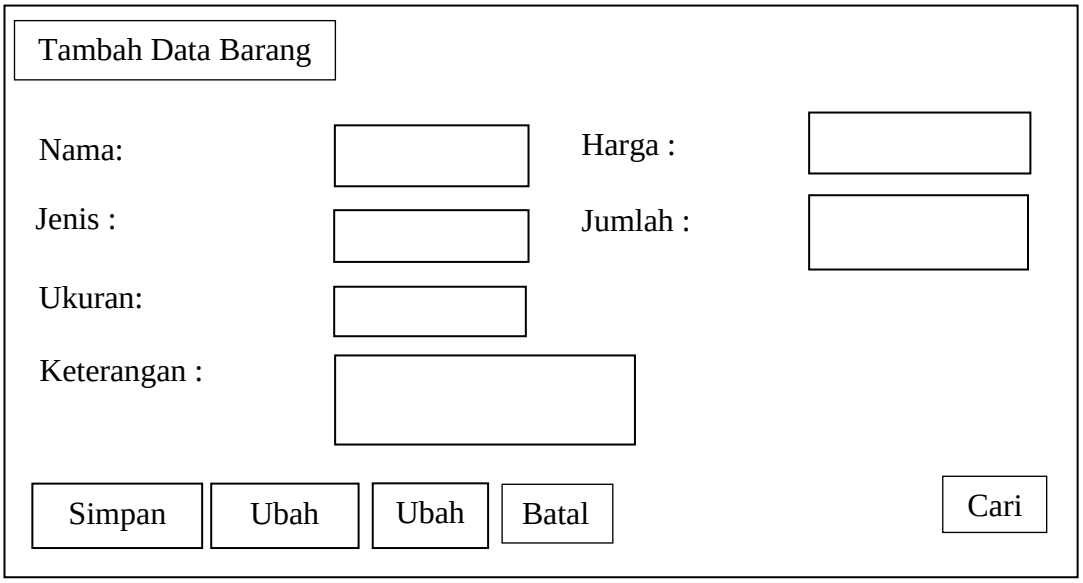
Tombol *Save* akan menyimpan data Data *Supplier* langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apabila *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan.



Gambar 4.26. *Form* Inputan Data *Supplier* Yang Diusulkan

4. Pendaftaran Data Barang

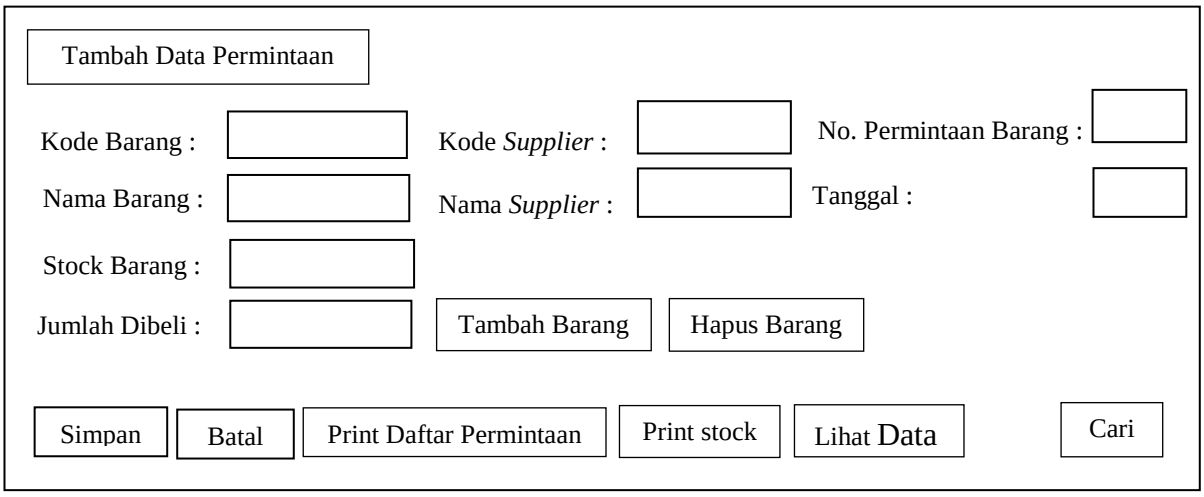
Tombol *Save* diatas akan menyimpan data Data Barang langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apablia *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan.



Gambar 4.27. *Form* Inputan Data Barang Yang Diusulkan

5. Pendaftaran Permintaan Pembelian

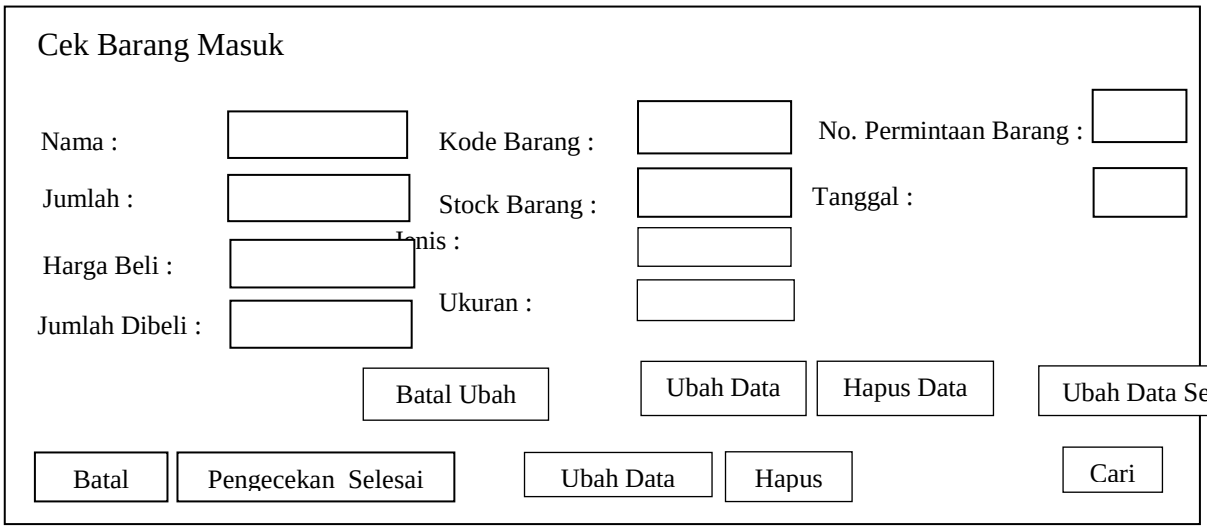
Tombol *Save* akan menyimpan data permintaan pembelian langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apablia *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan



Gambar 4.28. *Form* Pendaftaran Permintaan Pembelian Yang Diusulkan

6. Pendaftaran Cek Barang Masuk

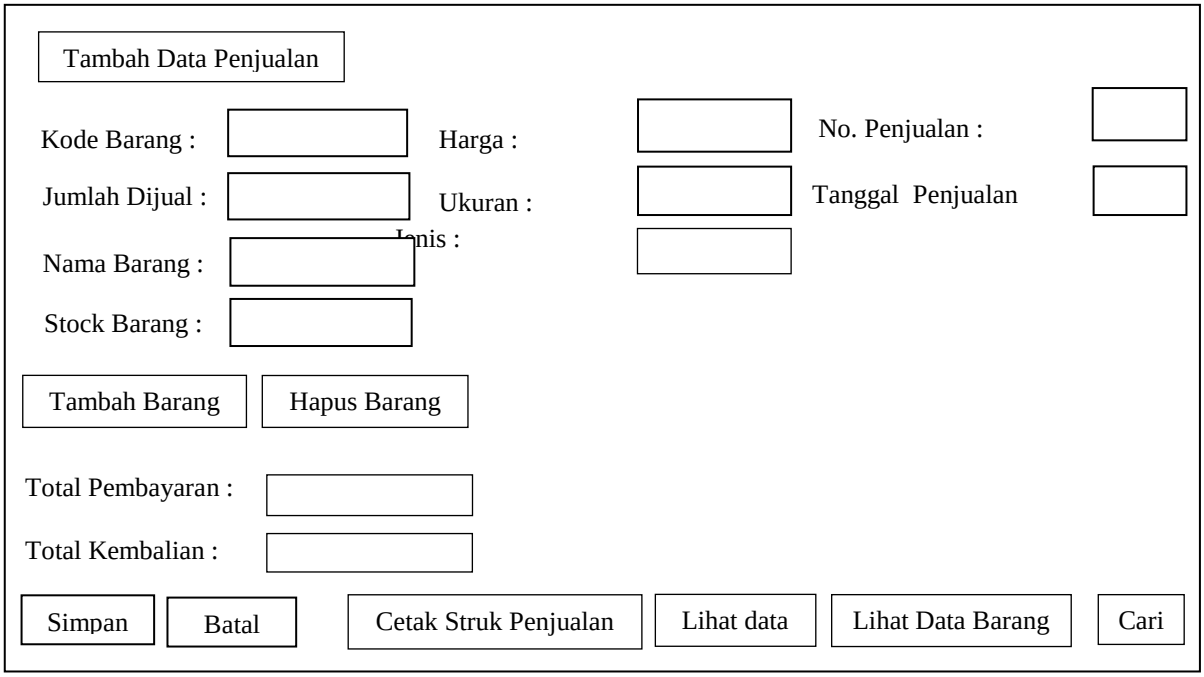
Tombol *Save* akan menyimpan data cek barang masuk langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apablia *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan.



Gambar 4.29. *Form* Pendaftaran Cek Barang Masuk Yang Diusulkan

7. Pendaftaran Transaksi Penjualan

Tombol *Save* akan menyimpan data transaksi penjualan langsung ke *database* sistem, sedangkan tombol *Reset* diatas akan mengulang pengisian *record*, apablia *user* salah memasukkan data/ ingin mengganti data, dengan mengklik tombol *Reset* tersebut, maka sistem otomatis akan menghapus inputan.



Gambar 4.30. *Form* Transaksi Penjualan Yang Diusulkan

4.2.4.3. Perancangan *Output*

Rancangan *output* adalah produk dari sistem informasi yang dihasilkan setelah input dilaksanakan. Hasil dari rancangan input data terlihat dari media keluaran. Dalam rancangan *output*, sistem informasi ini juga memanfaatkan file database mengandung data-data yang terbilang cukup banyak sehingga membutuhkan tempat yang cukup luas untuk menggambarkan bagaimana sebenarnya rancangan *output* pada aplikasi yang akan dibuat.

1. Laporan Penerimaan Barang

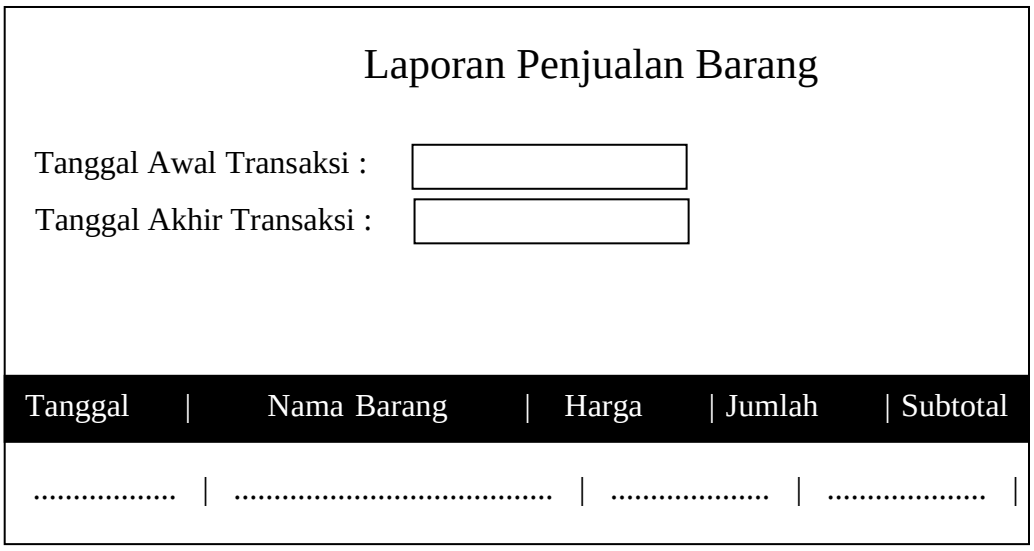
Perancangan *output* ini bertujuan menampilkan data rekapan Laporan Penerimaan Barang yang diajukan oleh Bagian Pembelian. Dimana *output* ini berbentuk laporan berformat .docx yang dapat langsung diunduh pada *Form* Laporan Penerimaan Barang di Program Aplikasinya.



Gambar 4.31. *Form Output* Laporan Penerimaan Barang Yang Diusulkan

2. Laporan Penjualan

Perancangan *output* ini bertujuan menampilkan data pejualan barang yang diajukan oleh konsumen. Dimana *output* ini berbentuk laporan berformat .pdf yang dapat langsung diunduh pada *Form* Penjualan Barang di Program Aplikasinya.

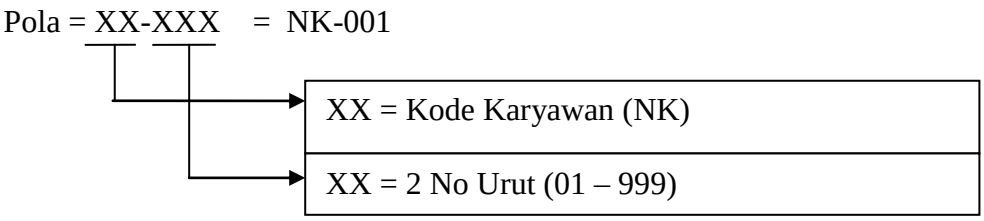


Gambar 4.32. *Form Output* Laporan Penjualan Barang Yang Diusulkan

4.2.4.4. Kodefikasi

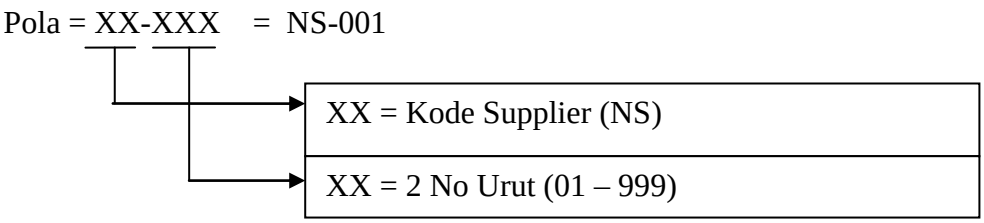
Merupakan pola kode otomatis yang dibentuk oleh sistem, dimana inputan yang dimasukkan akan otomatis digenerate oleh sistem. Contoh kedefikasi yang ada di Program Aplikasi ini adalah :

1. Kode Karyawan



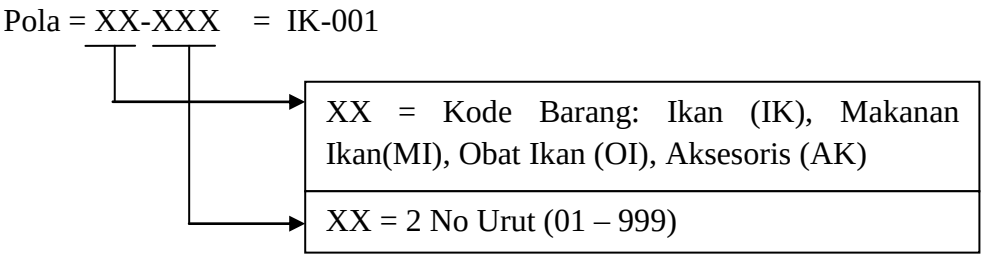
Gambar 4.33. Struktur Kodefikasi Kode Karyawan

2. Kode Supplier



Gambar 4.34. Struktur Kodefikasi Kode Supplier

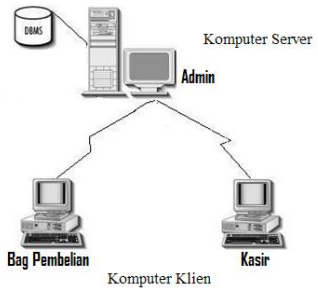
3. Kode Barang



Gambar 4.35. Struktur Kodefikasi Kode Barang

4.2.4.5. Perancangan Arsitektur Jaringan

Adapun arsitertur jaringan sebagai lingkungan pengembangan dan implementasi dari lingkungan dari aplikasi secara ringkas diilustrasikan oleh gambar berikut :



Gambar 4.36. Arsitektur Jaringan Yang Diusulkan

Secara umum, arsitektur tersebut dapat dijelaskan sebagai berikut :

1. Program Aplikasi ini akan di *install* di komputer *server*, yang mana akan menggunakan DBMS (*Database Management System*). Komputer server di jalankan oleh admin untuk memonitori semua laporan penjualan dan pembelian.
2. Masing-masing komputer klien akan mengakses aplikasi tersebut dengan menggunakan sistem jaringan komputer. Komputer klien ada dua masing-masing di jalankan oleh bagian pembelian untuk proses pembelian barang ke *supplier* dan cek barang masuk, sedangkan kasir untuk transaksi penjualan.
3. Topologi yang digunakan adalah topologi bintang/ *star* yaitu suatu cara untuk menghubungkan antara komputer satu dengan komputer yang lainnya sehingga dapat membentuk jaringan berupa bentuk bintang / *star*.