BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem Yang Sedang Berjalan

Analisis merupakan langkah awal yang dilakukan dalam perancangan sistem sebagai acuan agar perancangan sistem lebih terarah dan sistem dapat terselesaikan sesuai dengan kebutuhan.

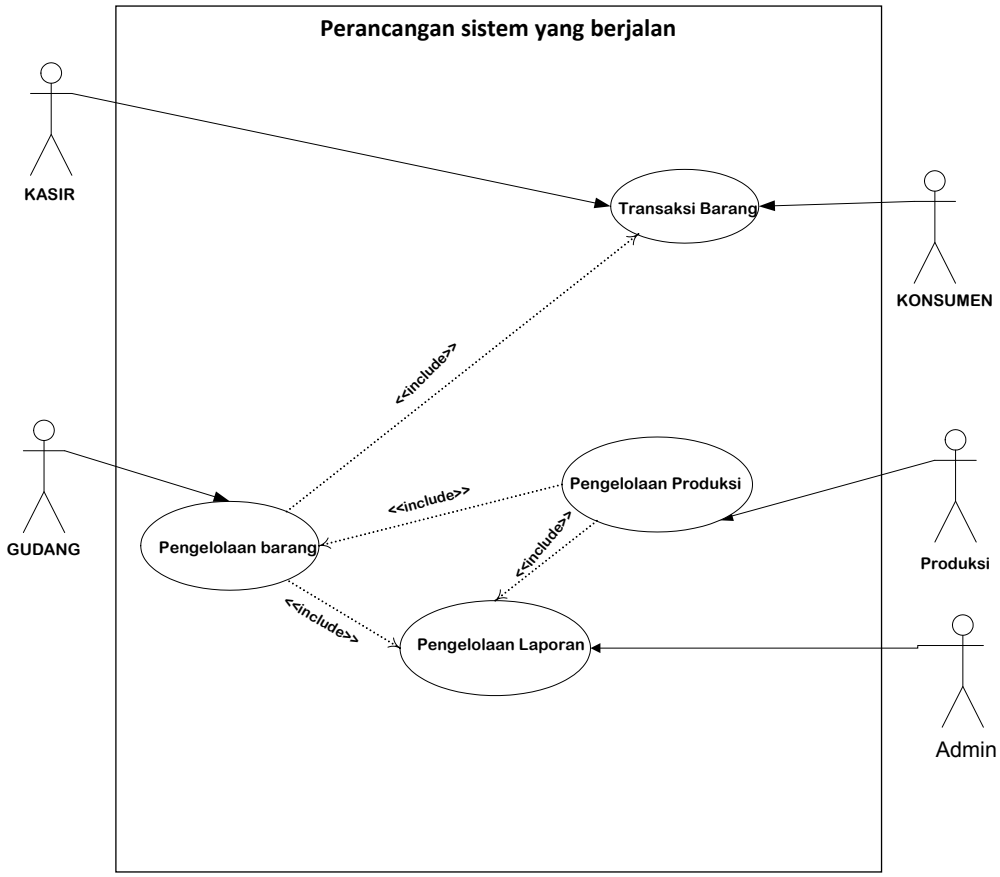
4.2 Analisis Prosedur Yang Sedang Berjalan

Dalam kegiatan sehari-harinya, Boutique JASMINE memiliki prosedur-prosedur dalam pelaksanaannya. Maka, untuk pengembangan sistem informasi yang akan dibuat, diperlukan analisis mengenai prosedur-prosedur yang sedang berjalan sebagai gambaran untuk sistem yang akan dikembangkan. adapun alat bantu analisis yang digunakan adalah sebagai berikut :

4.2.1 Use Case Diagram

Use Case Diagram merupakan hal yang sangat penting dilakukan pada tahap analisis. Dengan menggunakan *Use Case Diagram* kita akan mendapatkan banyak informasi yang sangat penting yang berkaitan dengan aturan-aturan bisnis yang coba kita tangkap. Dalam hal ini, setiap objek yang berinteraksi dengan sistem/perangkat lunak merupakan *actor* untuk sistem/perangkat lunak kita, sementara *Use Case* merupakan deskripsi lengkap tentang interaksi yang terjadi antara

para *actor* dengan sistem/perangkat lunak yang sedang kita kembangkan. Adapun *Use Case Diagram* proses bisnis yang ada di Boutique JASMINE yaitu :



Gambar 4.1 *Use Case* Proses bisnis yang sedang berjalan pada BOUTIQUE Jasmine

4.2.1.1 Skenario Use Case

Setiap *Use Case Diagram* harus kita deskripsikan skenarionya untuk lebih memperinci lagi perilaku sistem untuk masing-masing *use case* yang ada.

1. Nama *Use Case* : Pengelolaan Barang

   Aktor Utama : Gudang, Produksi

   Aktor Tambahan : -

   Tujuan : Proses pencatatan barang

Tabel 4.1 Skenario *Use Case* Pencatatan barang yang berjalan

Produksi	**Gudang**
1. Bagian produksi memberikan barang yang telah jadi	
	2. Bagian gudang menerima barang
	3. Bagian gudang mencatat barang ke stok barang

2. Nama *Use Case* : Transaksi Barang

   Aktor Utama : Kasir

   Aktro Tambahan : Konsumen

   Tujuan : Konsumen membeli barang

Tabel 4.2 Skenario *Use Case* Transaksi barang yang berjalan

Konsumen	**Kasir**
1. Konsumen memilih	

barang	
	2. Kasir mencari barang
	3. Kasir mencatat barnag ke nota
	4. Kasir menghitung total bayar
5. Konsumen melakukan pembayaran	

3. Nama *Use Case* : Pembuatan Laporan

   Aktor Utamaa :Admin, Gudang, Kasir, Produksi

   Aktor Tambahan : -

   Tujuan : Memberi laporan data produksi, barang dan penjualan kepada admin

Tabel 4.3 Skenario *Use Case* Pembuatan laporan yang berjalan

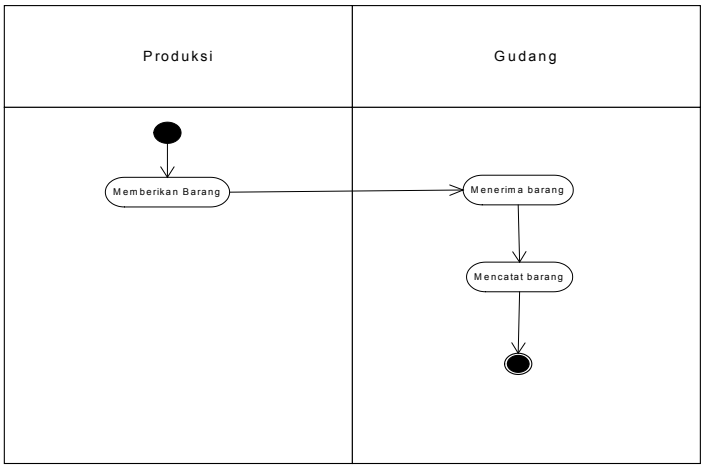
Gudang, Kasir, Produksi	**Admin**
1. Bagian produksi, gudang, kasir memberikan laporan data produksi, barang, penjualan kepada admin	
	2. Admin menerima

	laporan

4.2.2 Activity Diagram

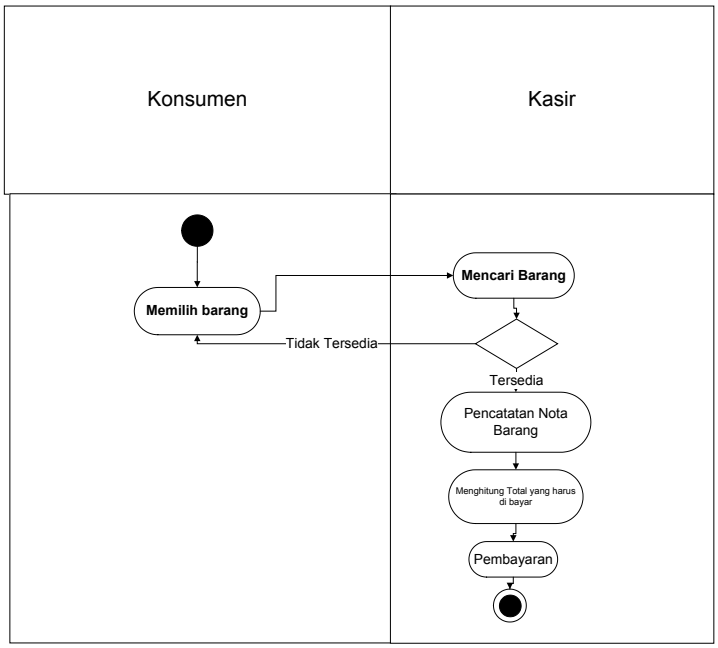
Untuk lebih memperinci *Use Case Diagram* proses bisnis yang berjalan, maka perlu dibuat *activity diagram* sebagai berikut :

1. *Activity Diagram* Pengelolaan Barang



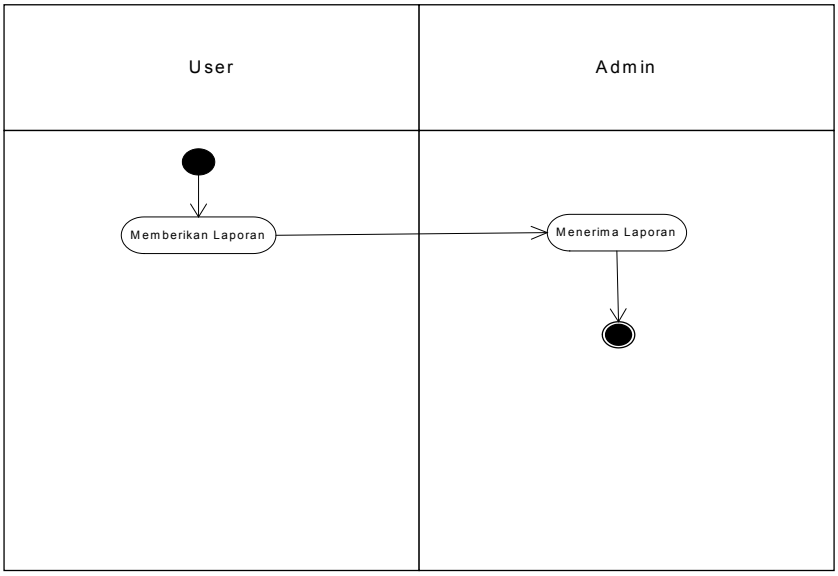
Gambar 4.2. *Activiti Diagram* pengelolaan barang yang berjalan

2. *Activity Diagram* Transaksi



Gambar 4.3. *Activity Diagram* Transaksi Yang Sedang Berjalan

3. *Activity Diagram* Laporan



Gambar 4.4. *Activity Diagram* Laporan yang berjalan

4.2.3 Evaluasi Sistem Yang Sedang Berjalan

Setelah melakukan analisis terhadap proses bisnis yang sedang berjalan, penulis menemukan beberapa masalah yang dapat mengakibatkan penurunan pelayanan terhadap konsumen. Setelah melakukan identifikasi terhadap penyebab permasalahan yang ada, penulis juga melakukan pengamatan dengan maksud untuk mengumpulkan fakta atau data yang menjadi permasalahan pada sistem ini. Adapun masalah-masalahnya adalah sebagai berikut:

1. Proses Penjualan barang di Boutique JASMINE didalam pengolahan data produksi, data barang, proses penjualan barang, dan laporan penjualan masih menggunakan buku catatan yang mana dalam proses menulisnya memakan waktu yang lama dan dapat terjadi duplikasi data.
2. Proses pencarian informasi data mengalami hambatan karena untuk mencari data dilakukan dengan cara membuka buku catatan sehingga waktu pencarian menjadi lama.
3. Masih banyak penggunaan dokumen berupa kertas sehingga memungkinkan terjadinya kerusakan atau kehilangan dokumen tersebut.

4.3 Perancangan Sistem

Sebelum memulai perancangan, terlebih dahulu dilakukan perencanaan untuk melakukan pengembangan sistem. Perencanaan sistem ini menyangkut estimasi dari kebutuhan-kebutuhan fisik dan tenaga kerja yang dibutuhkan

untuk mendukung pengembangan sistem serta untuk mendukung operasinya setelah diterapkan. Perancangan sistem dapat diartikan sebagai :

1. Tahap setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian atas kebutuhan-kebutuhan fungsional.
3. Persiapan untuk rancang bangun implementasi.
4. Menggambarkan sistem yang akan dibentuk.
5. Konfigurasi komponen *software* dan *hardware* sistem.

4.3.1 Tujuan Perancangan Sistem

Perancangan sistem ini bertujuan untuk memberikan gambaran secara umum dan terperinci kepada pengguna tentang sistem yang akan dibangun dan mengidentifikasikan komponen-komponen sistem informasi yang akan dirancang secara *detail*. Tujuan perancangan sistem informasi penjualan ini akan diuraikan sebagai berikut :

1. Merubah proses-proses pencatatan kedalam buku besar yang menyangkut pengelolaan data menjadi terkomputerisasi.
2. Meningkatkan pelayanan terhadap konsumen dan memberikan kemudahan dalam proses pengelolaan data.
3. Pembuatan laporan data barang dan transaksi barang menjadi terkomputerisasi

4.3.2 Gambaran Umum Sistem yang Diusulkan

Pada tahap perancangan sistem informasi berbasis *Client-Server* di rancang dengan tujuan sebagai alat komunikasi antara pemakai (*user*) atau kasir dan bagian gudang dengan pembuat program guna mendapatkan sistem aplikasi yang sesuai dengan apa yang dibutuhkan. Sistem Aplikasi ini mempunyai pelayanan dan fungsi yaitu mengelola data barang, transaksi barang dan laporan data barang.

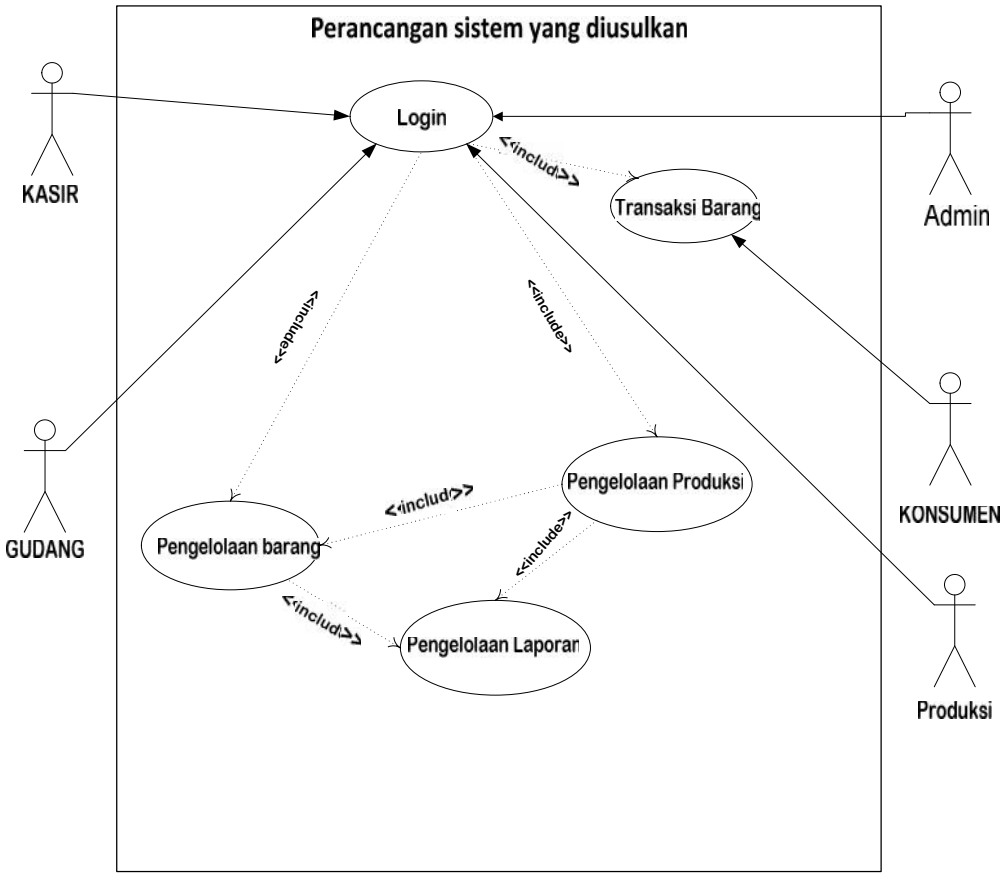
Untuk fungsi-fungsi yang terdapat di dalam sistem informasi berbasis *client-server* ini dibagi kedalam 2 bagian. Pada bagian kasir, tidak perlu sulit dan buang-buang waktu lagi untuk menghitung dengan kalkulator dan menuliskan data transaksi pada buku besar, karena sistem ini mempunyai fungsi pelayanan transaksi penjualan. Serta bagian gudang tidak perlu buang-buang waktu untuk memeriksa stok barang yang ada digudang dan menghitungnya satu persatu karena sistem ini bisa menghitung secara otomatis barang yang masuk dan keluar. Sistem informasi *client-server* ini diharapkan dapat membantu mengatasi masalah yang ada.

4.3.3 Perancangan Prosedur Yang Diusulkan

Perancangan ini menggunakan alat bantu berupa *use case diagram*, *activity diagram, sequence diagram, class diagram, object diagram, diagram component* dan *diagram deployment* untuk menghasilkan sistem yang baik dan sesuai kebutuhan.

4.3.3.1 Use Case Diagram

Berikut adalah *use case* diagram perancangan sistem yang diusulkan pada Boutique JASMINE :



Gambar 4.5. *Use case* diagram sistem yang diusulkan pada Boutique Jasmine

4.3.3.1.1 Skenario Use case

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Berikut skenario use case yang diusulkan :

1. Nama Use Case : *Login User*

   Aktor Utama : Kasir, Gudang dan Admin

   Aktor Tambahan : -

   Tujuan : Proses Validasi Akun *User*

Tabel 4.4. Skenario *Use Case Login User*

User	**Sistem**
1. Masuk ke halaman *login user*	
2. *M*engisi *username* dan *password* pada form *login* akun member.	
	3. Verifikasi *Username* dan *password* pada sistem database.
	4. Jika *account* terdaftar, maka secara otomatis akan dapat mengakses menu utama, jika *account* tidak terdaftar maka sistem akan memberikan peringatan 'Username atau Password tidak valid' dan kembali ke halaman *login user.*

2. Nama Use Case : Transaksi Barang

   Aktor Utama : Kasir

   Aktor Tambahan :Konsumen

   Tujuan : Melakukan proses transaksi barang

Tabel 4.5. Skenario *Use Case* Transaksi Barang

Konsumen	Kasir	Sistem
1. Permintaan Barang		
	2. Bagian kasir menerima permintaan barang.	
	3. Bagian Kasir melakukan proses transaksi masuk ke menu transaksi barang	
		4. Menampilkan halaman transaksi untuk menginput data barang dan menampilkan nomor transaksi secara otomatis
	5. Bagian Kasir menginputkan nama barang untuk ditampilkan	
		6. Menampilkan kode, harga, ukuran dan jumlah barang yang tersedia tanpa harus diinputkan lewat *keyboard*.
		7. Apabila stok barang masih tersedia di gudang, maka transaksi

		dilanjutkan, tetapi apabila stok habis maka transaksi dibatalkan dan permintaan dikembalikan kepada konsumen.
	8. Bagian Kasir menginputkan jumlah barang yang dibeli konsumen.	
	9. Bagian Kasir menambahkan ke daftar beli	
		10. Verifikasi data pembelian barang
	11. Bagian Kasir memproses dengan 'BAYAR'	
		12. Menghitung secara otomatis semua total pembelian barang.
		13. Konfirmasi berhasil akan muncul tampilan 'Input sukses'
		14. Menyimpan data yang sudah *valid* di *database*
		15. Menampilkan struk penjualan barang untuk dicetak kasir
	16. Mencetak struk pembayaran	

17. Menerima Struk		

3. Nama Use Case : Pengelolaan Produksi

   Aktor Utama : Produksi

   Aktor Tambahan : -

   Tujuan : Proses Penambahan barang

Tabel 4.6. Skenario *Use Case* pengelolaan produksi

Produksi	**Sistem**
1. Bagian produksi menambahkan Jenis barang baru	
	2. Menampilkan tambah jenis barang untuk input jenis barang baru
3. Menginputkan seluruh data jenis barang baru yang di perlukan	
4. Konfirmasi ‘SIMPAN’	
	5. Menampilkan ‘input suskes’ apabila berhasil apabila tidak kembali ke input data jenis barang
	6. Menyimpan data yang sudah *valid* di *database*
7. Bagian produksi menambahkan barang baru	
	8. Menampilkan tambah barang untuk input barang baru
9. Menginputkan seluruh data barang baru yang di perlukan	
10. Konfirmasi ‘SIMPAN’	

	11. Menampilkan ‘input suskes’ apabila berhasil apabila tidak kembali ke input data barang
	12. Menyimpan data yang sudah *valid* di *database*
13. Bagian Produksi akan mengganti detail produk lama maka bagian produksi akan *edit* Barang lama tersebut dengan cara mengubahnya	
	14. Menampilkan tampilan *edit* Barang
15. Bagian Produksi mencari barang berdasarkan kode barang yang akan di update datanya atau secara langsung memilih data barang yang sudah ada pada tabel barang	
16. Memulai pengubahan dengan ‘Edit’	
	17. Menampilkan Data-data barang yang akan di update secara otomatis
18. Mengubah data barang yang akan di edit	
19. konfirmasi dengan‘*UPDATE’*	
	20. Proses pengeditan barang
	21. Apabila barang berhasil di edit akan menampilkan edit berhasil, dan bila gagal kembali ke edit data barang
	22. Menyimpan data *valid* di *database*
23. Bagian Produksi menerima laporan minimal barang	
24. Masuk ke *form* tambah barang lama	

	25. Menampilkan *form* barang lama
26. Masukan nama barang sesuai laporan	
	27. Menampilkan data barang
28. Menambah stok barang	
\	**29.** Menampilkan 'input suskes' apabila berhasil apabila tidak kembali ke input data barang
	30. Menyimpan data *valid* di *database*

4. Nama Use Case : Pengelolaan Barang

   Aktor Utama : Gudang

   Aktor Tambahan : -

   Tujuan : Proses Pemantauan barang yang ada di Boutique Jasmine

Tabel 4.7. Skenario *Use Case* pengelolaan barang

Gudang	**Sistem**
31. Bagian gudang menambahkan barang yang diberi bagian produksi	
	32. Menampilkan tambah barang untuk input barang baru
33. Menginputkan kode produksi	
	34. Menampilkan data barang berdasarkan kode produksi
35. Memasukan nama barang baru	

	36. Menampilkan detail barang baru
37. Konfirmasi ‘SIMPAN’	
	38. Menampilkan ‘input suskes’ apabila berhasil apabila tidak kembali ke input data barang
	39. Menyimpan data yang sudah *valid* di *database*
40. Bagian Gudang akan mengganti detail produk lama maka bagian gudang akan *update* Barang lama tersebut dengan cara mengubahnya	
	41. Menampilkan tampilan *edit* Barang
42. Bagian gudang mencari barang berdasarkan kode barang yang akan di update datanya atau secara langsung memilih data barang yang sudah ada pada tabel barang	
43. Memulai pengubahan dengan ‘Edit’	
	44. Menampilkan Data-data barang yang akan di update secara otomatis
45. Mengubah data barang yang akan di edit	
46. konfirmasi dengan‘*UPDATE’*	
	47. Proses pengeditan barang
	48. Apabila barang berhasil di edit akan menampilkan edit berhasil, dan bila gagal kembali ke edit data barang
	49. Menyimpan data *valid* di *database*

50. Bagian Gudang mengecek minimum stok barang	
	51. Menampilkan stok barang
52. Meceklis minimum stok	
	53. Menampilkan stok barang < 3
54. Mencetak stok barang untuk di tambah ke bagian produksi	
	55. Mencetak laporan minimal stok

4. Nama Use Case : Pengelolaan Laporan

   Aktor Utama :Admin, Gudang, dan Kasir

   Aktor Tambahan : -

   Tujuan :Membuat laporan data barang dan transaksi barang

Table 4.8 Skenario *Use Case* Pengelolaan Laporan

User	Sistem
1. Bagian produksi,gudan,kasir,admin masuk ke menu laporan	
	2. Menampilkan menu laporan
3. Memasukan parameter tanggal yang akan di cetak	
	4. Mensortir data sesuai tanggal yang sudah di tentukan

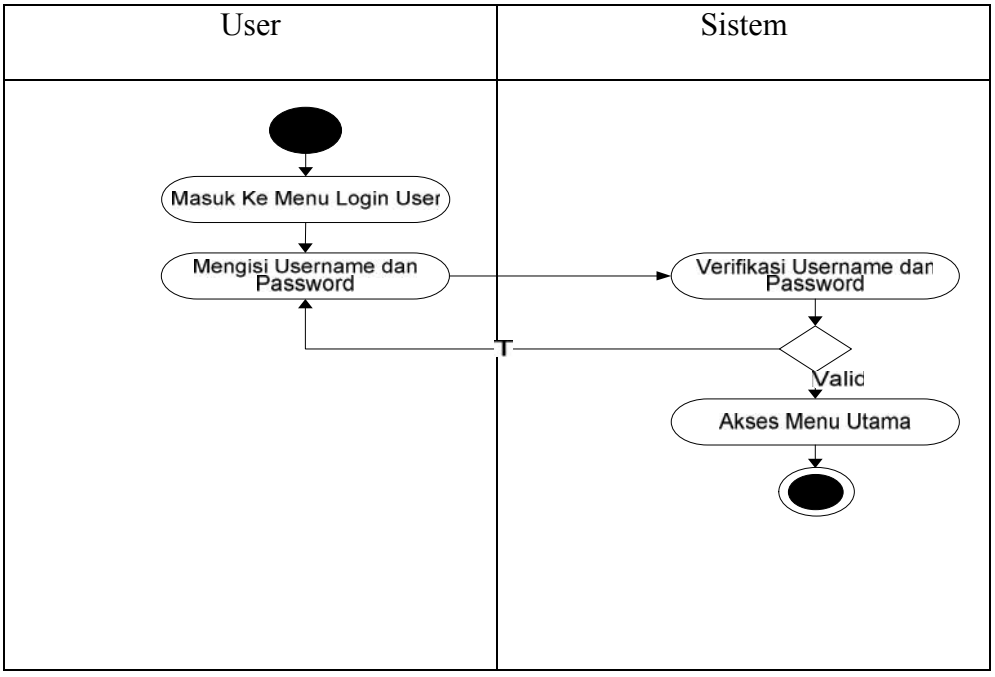
	5. Apabila data yang sesuai tanggal ada akan di tampilkan,dan bila tidak akan kembali ke menu laporan
	6. Cetak laporan

4.3.3.2 Activity Diagram

Activity diagram berikut ini memperlihatkan secara rinci aliran data secara logika tanpa mempertimbangkan lingkungan fisik dimana data mengalir. User diharuskan untuk login terlebih dahulu jika ingin menggunakan dan menjalankan pelayanan-pelayanan yang diberikan.

1. Deskripsi

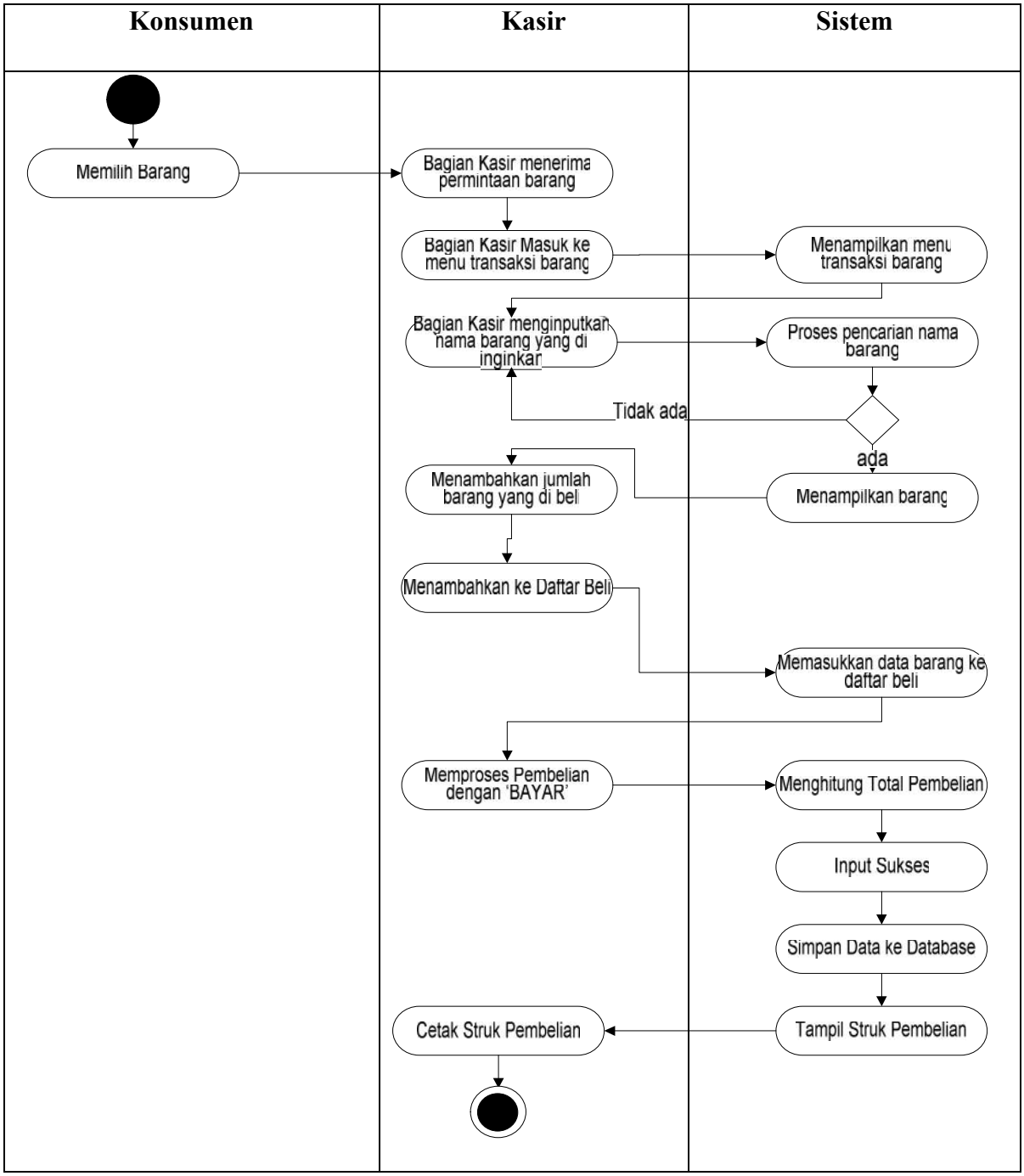
User melakukan login, Jika *account* terdaftar, maka secara otomatis akan masuk ke halaman utama, jika tidak ada data memberikan peringatan ‘User Tidak ada’ dan kembali ke halaman *login.*



Gambar 4.6. *Activity Diagram* Login User pada Boutique JASMINE Yang Diusulkan

2. Deskripsi

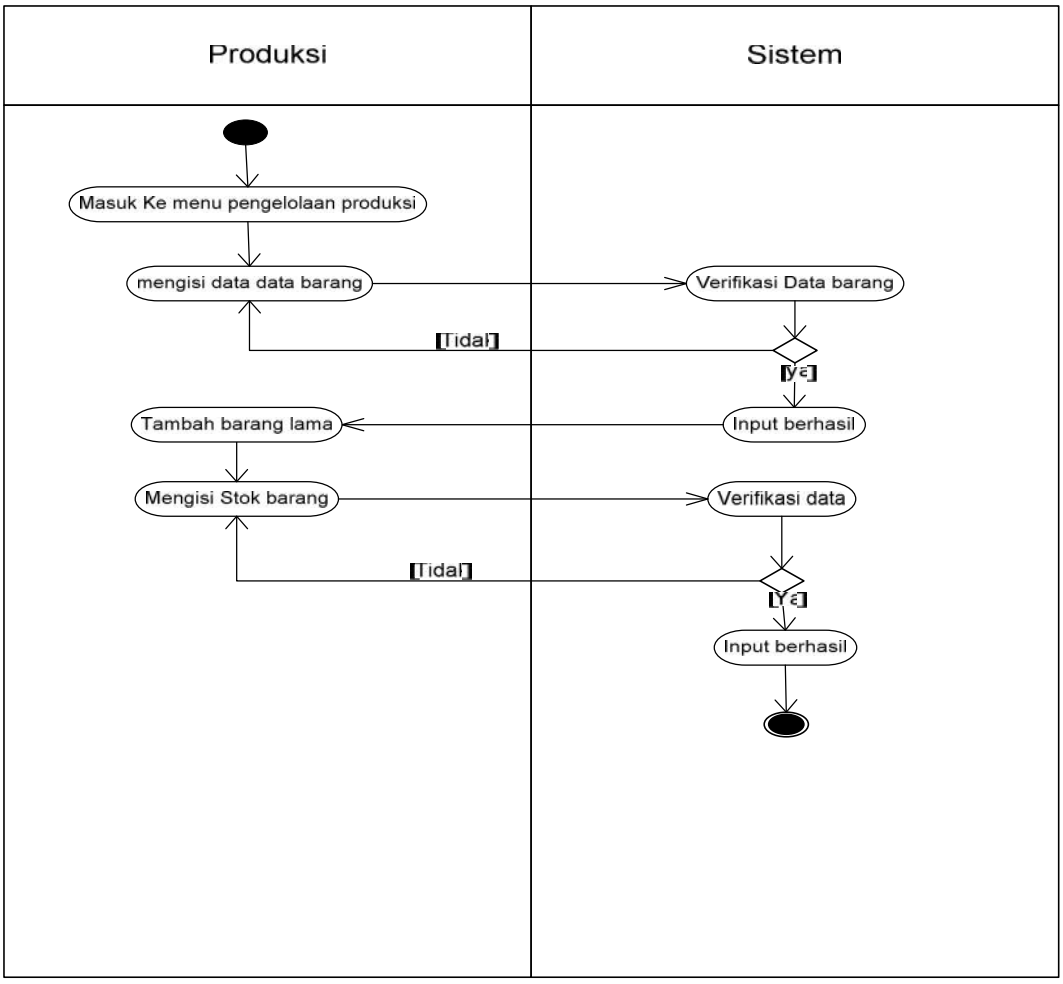
Kasir Masuk Ke Menu Transaksi Barang lalu memasukan data-data yang diperlukan untuk transaksi, melalui proses otomatis pencarian data sehingga kasir tidak perlu banyak mengetik inputan setelah itu hitung dengan otomatis dan secara langsung tersimpan di database.



Gambar 4.7. *Activity Diagram* Transaksi Barang pada Boutique JASMINE Yang Diusulkan

3. Deskripsi

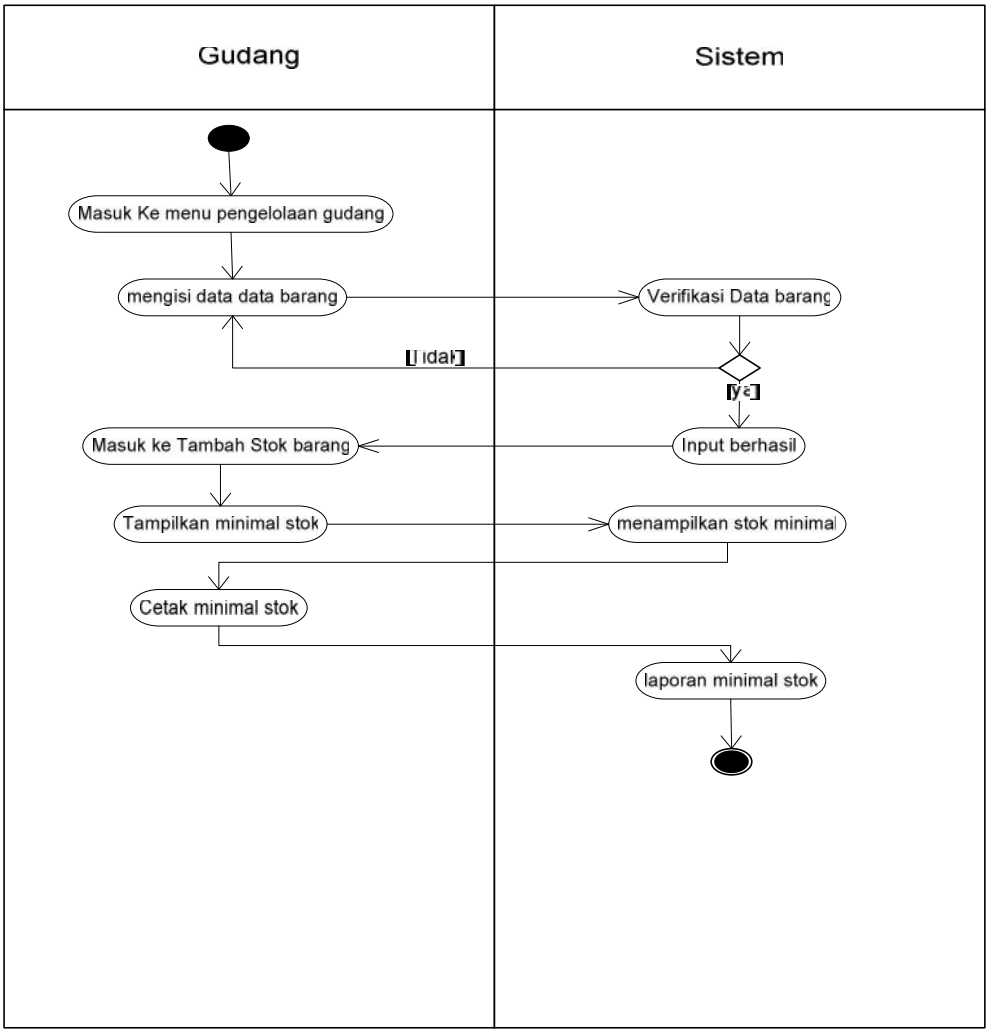
Bagian Gudang dapat melakukan pengelolaan barang yang mencakupi tambah jenis barang,tambah barang,edit barang,tambah stok barang yang akan di simpan pada database



Gambar 4.8. *Activity Diagram* Pengelolaan Produksi pada Boutique JASMINE Yang Diusulkan

4. Deskripsi

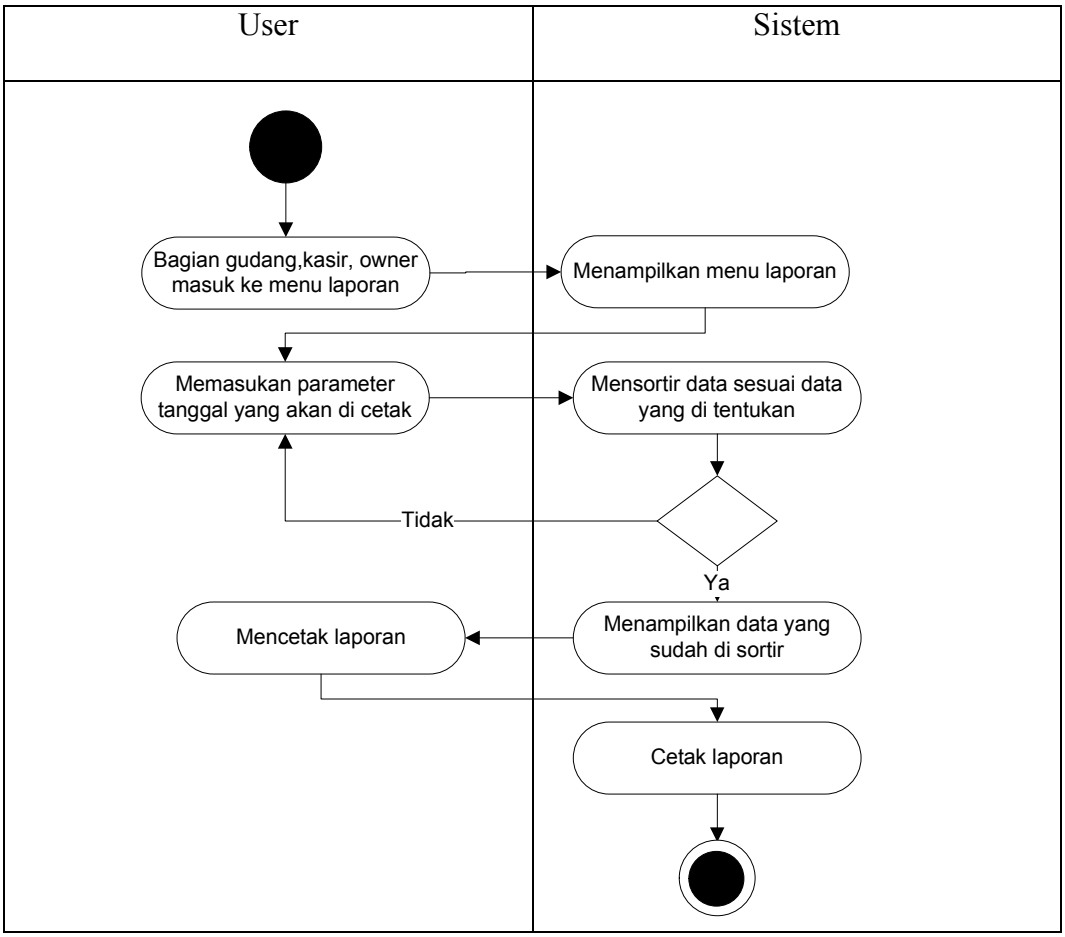
Bagian Gudang dapat melakukan pengelolaan barang yang mencakupi tambah jenis barang,tambah barang,edit barang,tambah stok barang yang akan di simpan pada database



Gambar 4.9. *Activity Diagram* Pengelolaan Barang pada Boutique JASMINE Yang Diusulkan

5. Deskripsi

User dapat membuat laporan data barang dan transaksi penjualan secara otomatis

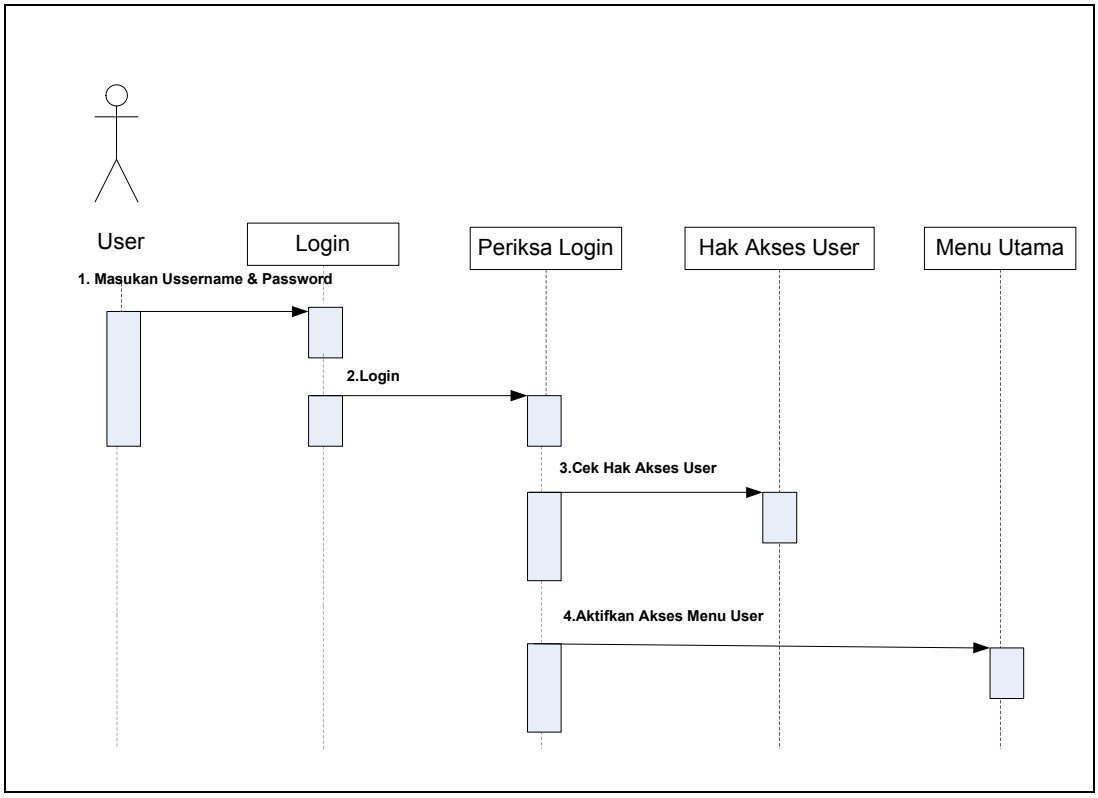


Gambar 4.10. *Activity Diagram* Pengelolaan Laporan pada Boutique JASMINE Yang Diusulkan

4.3.3.3 Sequence Diagram

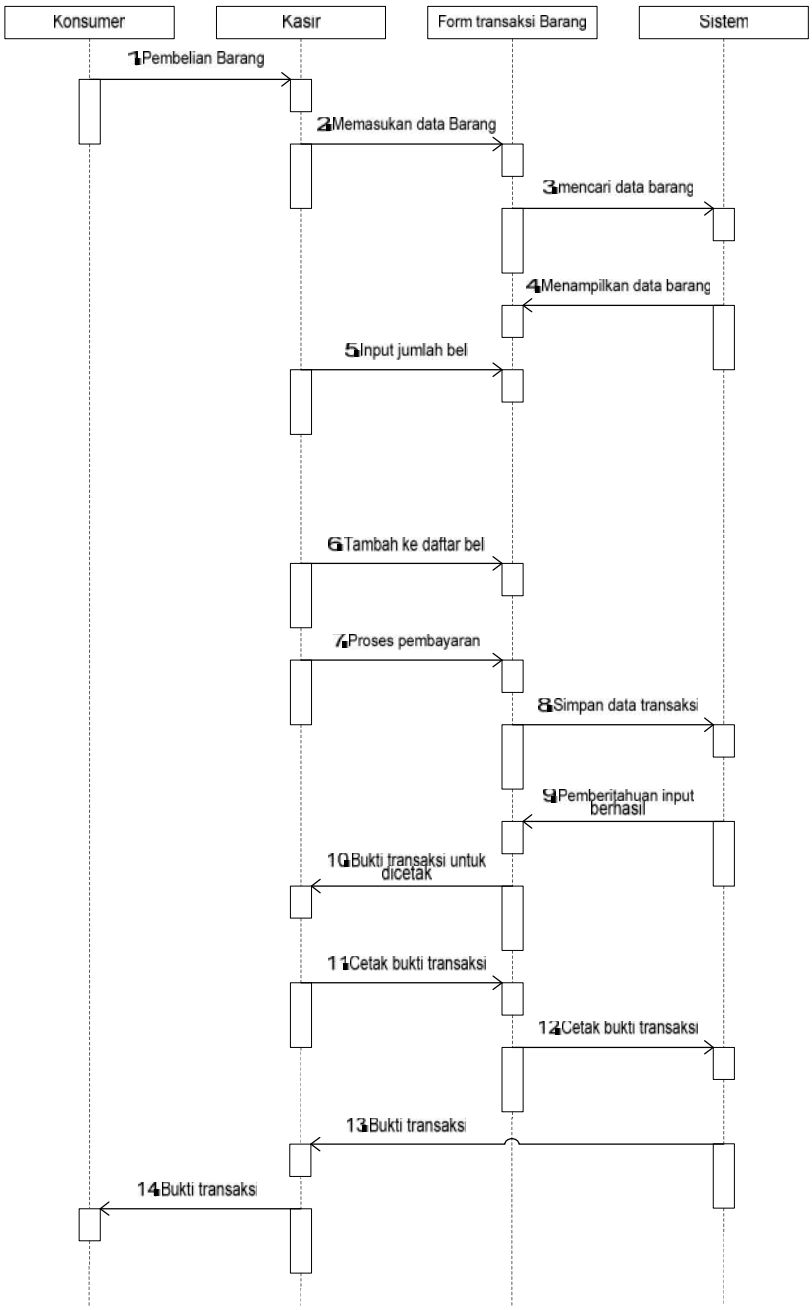
Menggambarkan kolaborasi dinamis antara sejumlah *object* . kegunaanya untuk menunjukan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object* , sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

1. *Sequence Diagram* Login User



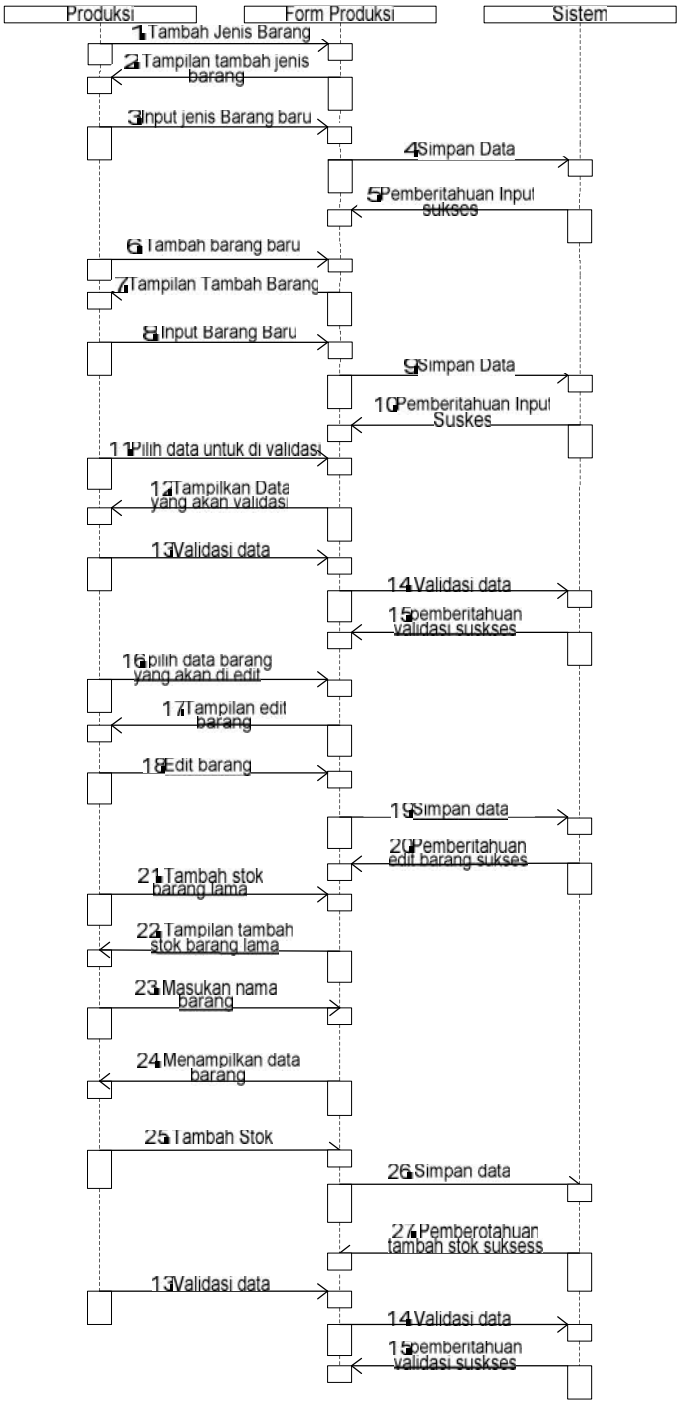
Gambar 4.11. *Sequence Diagram* Login User pada Boutique JASMINE Yang Diusulkan

2. *Sequence Diagram* Transaksi Barang



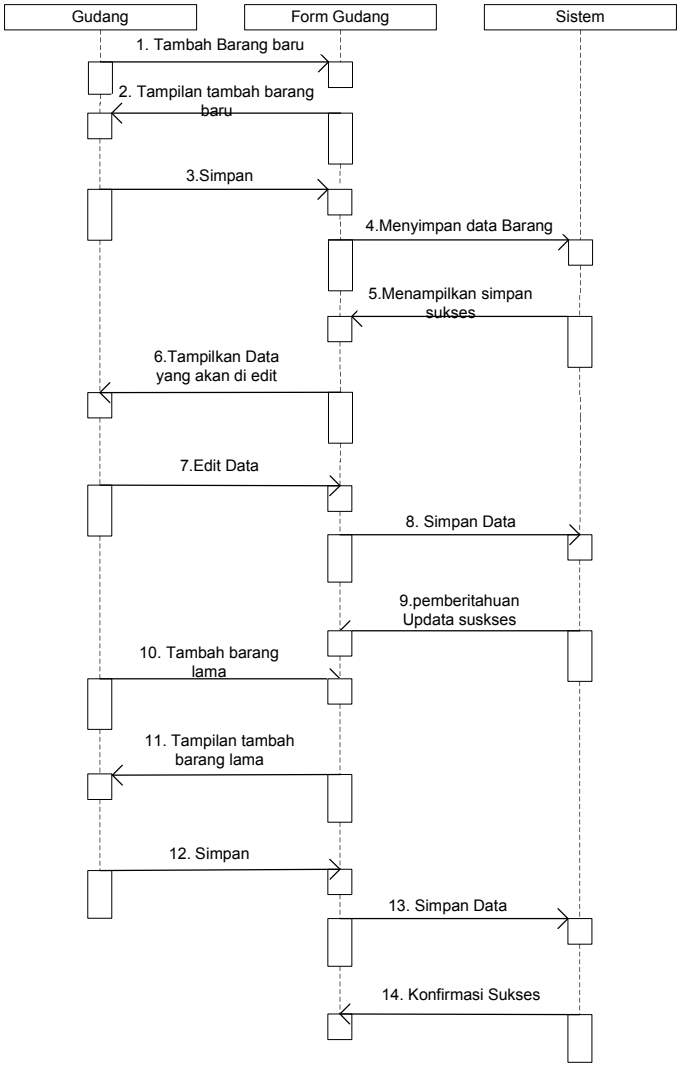
Gambar 4.12. *Sequence Diagram* Transaksi Barang pada Boutique JASMINE Yang Diusulkan

3. *Sequence Diagram* Pengelolaan Produksi



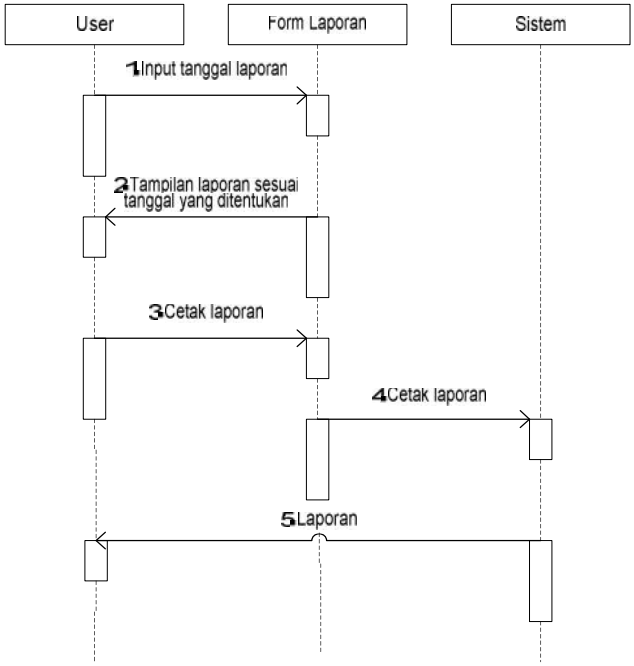
Gambar 4.13. *Sequence Diagram* Pengeolaan produksi pada Boutique JASMINE Yang Diusulkan

4. *Sequence Diagram* Pengelolaan barang



Gambar 4.14. *Sequence Diagram* Pengeolaan barang pada Boutique JASMINE Yang Diusulkan

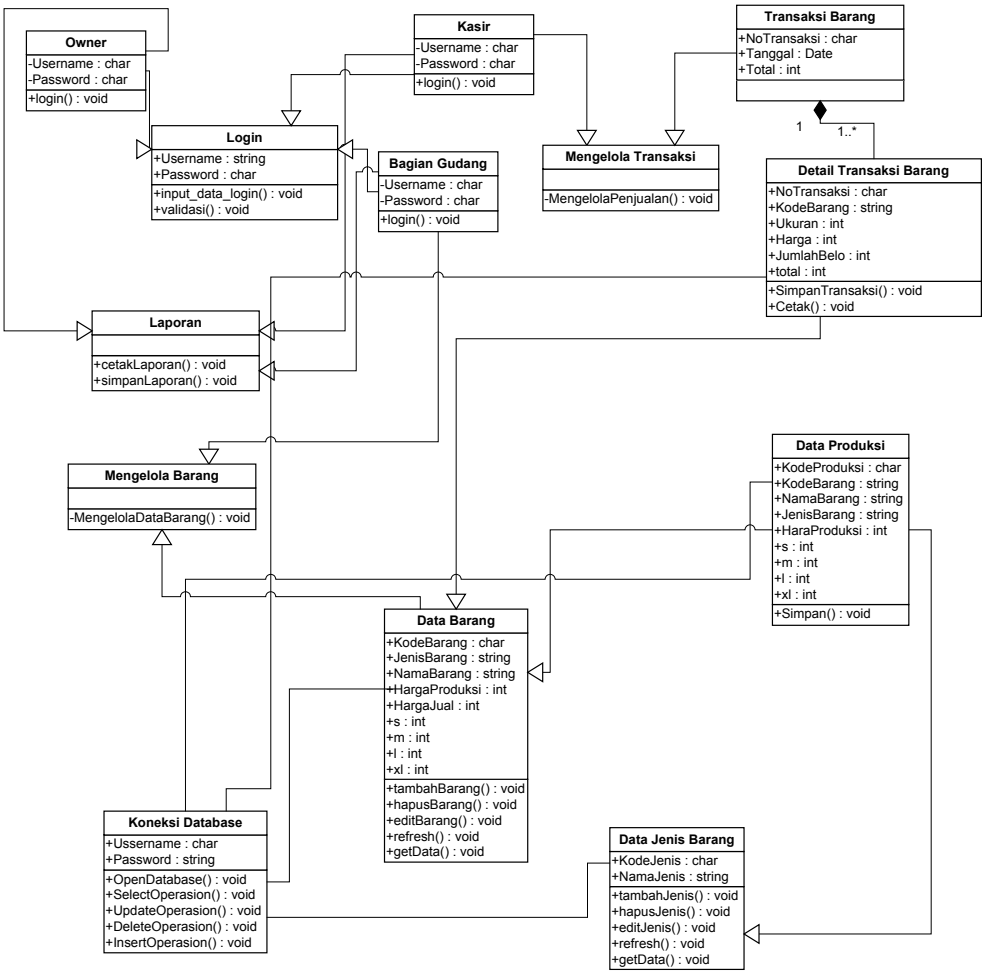
5. *Sequence Diagram* Pengelolaan Laporan



Gambar 4.15. *Sequence Diagram* Pengelolaan Laporan pada Boutique JASMINE Yang Diusulkan

4.3.3.4 Class Diagram

Menggambarkan struktur statis *class* di dalam sistem, *class* merepresentasikan sesuatu yang ditangani oleh sistem *class* dapat berhubungan dengan cara lain melalui berbagai cara *sociatea* (terhubung satu sama lain), *dependen* (satu *class* tergantung/menggunakan *class* yang lain), atau *package* (group bersama sebagai satu unit).sebuah sistem biasanya mempunyai beberapa *class diagram.*

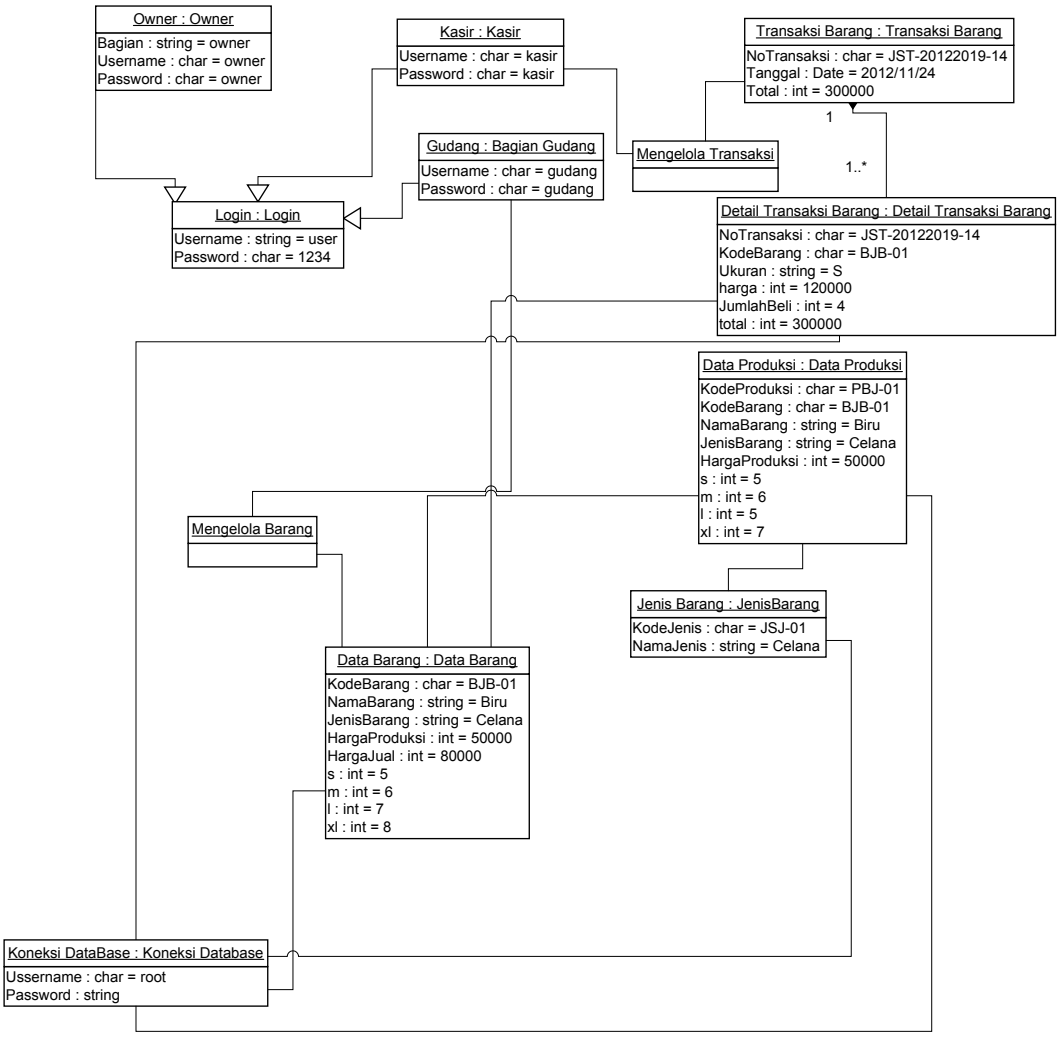


Gambar 4.16. *Class Diagram* pada Boutique JASMINE yang Diusulkan

4.3.3.5 Object Diagram

Diagram objek ialah merupakan suatu diagram yang mengambarkan objek-objek dalamsuatu system. Bila pada kelas diagram terdapat banyak relasi antar kelas, namun pada diagram objek, hanya terdapat satu relasi antar kelas, yaitu asosiasi. Tidak hanya itu, pada diagram objek yang di definisikan hanya

nama objek dalam suatu kelas beserta atributnya saja, tanpa operasi yang dilakukan oleh objek tersebut

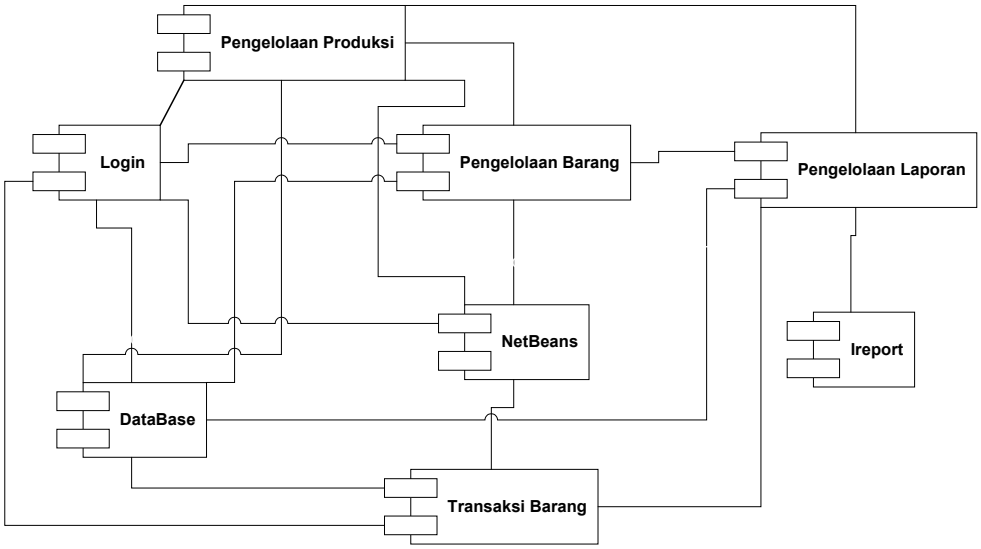


Gambar 4.17. *Object Diagram* pada Boutique JASMINE yang diusulkan

4.3.3.6 Diagram Component

Diagram ini menggambarkan struktur dan hubungan antara komponen piranti lunak, termasuk ketergantungan (*depedency*)

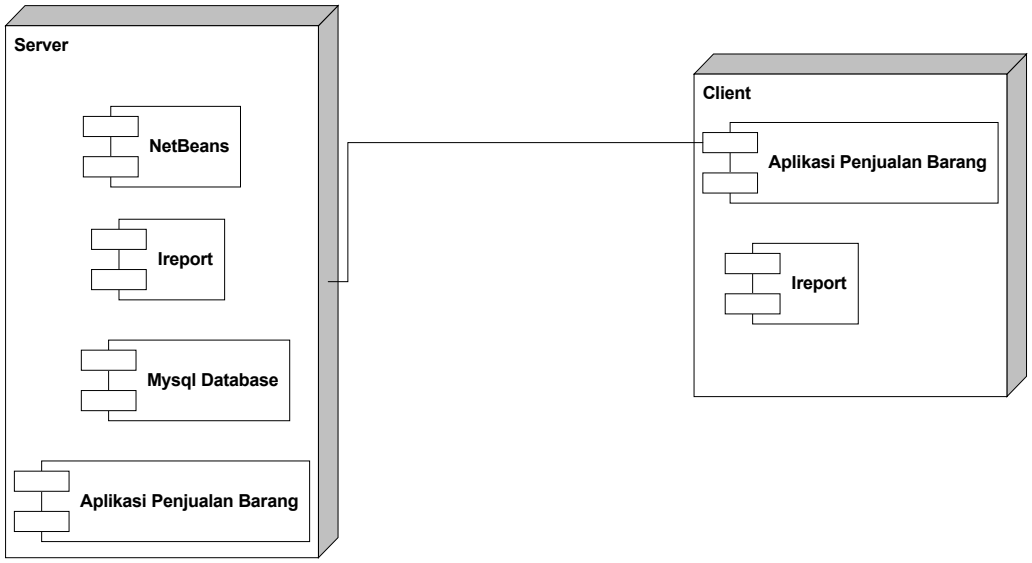
1. *Componen Diagram* Pengelolaan Barang



Gambar 4.18. *Component Diagram* pada Boutique JASMINE yang diusulkan

4.3.3.7 Diagram Deployment

Diagram Deployment menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian software yang berjalan pada bagian-bagian hardware



Gambar 4.19 *Deployment Diagram* pada Boutique JASMINE yang diusulkan

4.4 Perancangan Antarmuka

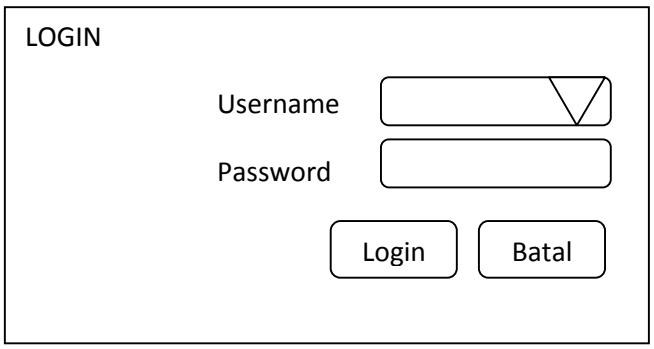
Agar sistem berinteraksi dengan para pengguna secara baik, maka perlu dirancang sebuah *interface* yang dapat memudahkan pengguna untuk mengoperasikannya. Sistem informasi yang baik bukan hanya dinilai dari segi tampilannya semata, namun akan dinilai juga bagaimana pola aliran informasi yang dibangun dalam bentuk sistem tersebut. Untuk mendukung proses pembentukan tersebut, secara umum perancangan antar muka suatu program meliputi : **Struktur Menu, Perancangan Input, Perancangan Output**.

4.4.1. Perancangan Input

Perancangan *input* merupakan desain yang dirancang untuk menerima masukan dari pengguna sistem. Rancangan *input* ini harus dapat memberikan penjelasan bagi pemakainya, baik dari bentuk maupun dari masukan-masukan yang akan diisi.

1. Tampilan Login

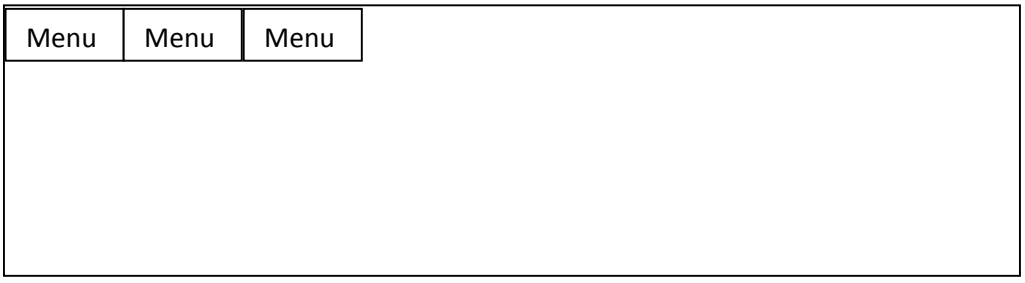
Pada tampilan login user memasukan username dan password



Gambar 4.20. Tampilan Menu Login

2. Tampilan Menu Utama Bagian Gudang

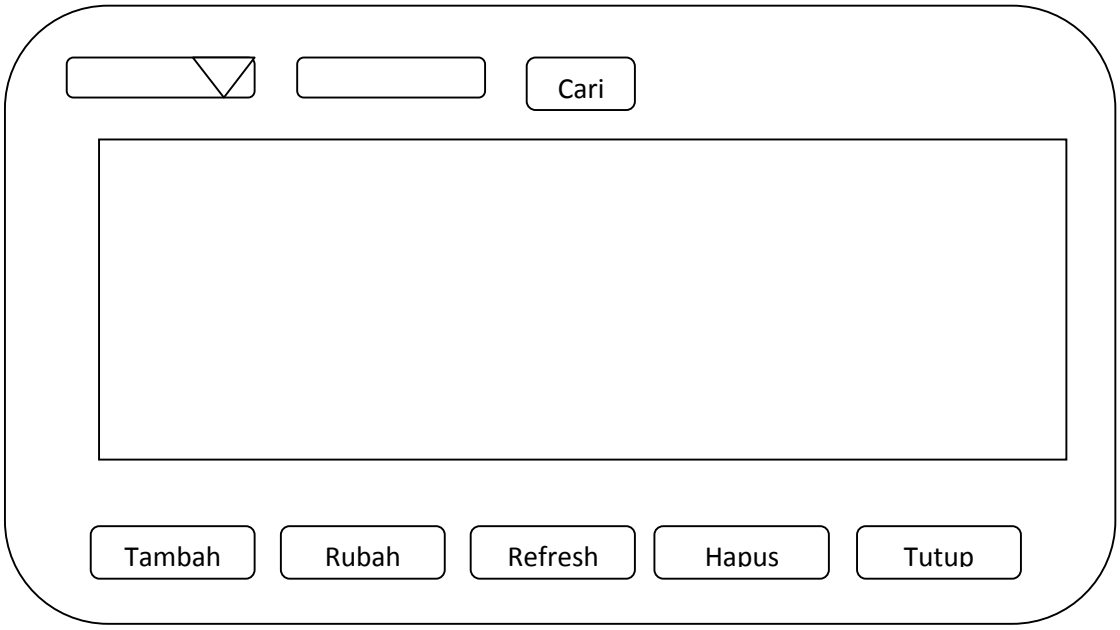
Ini adalah tampilan menu utama bagian gudang



Gambar 4.21 Tampilan Menu Utama Bagian Gudang

3. Tampilan Data Jenis Barang

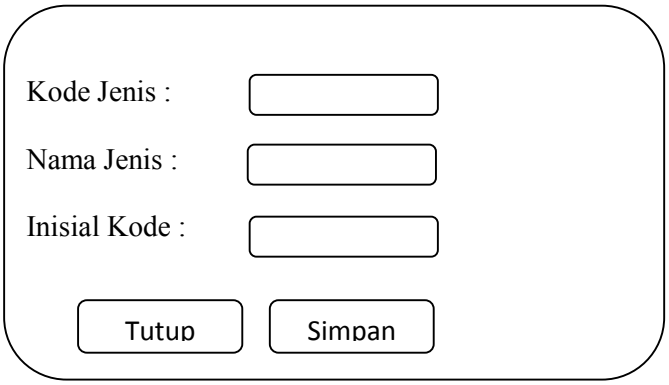
Ini adalah tampilan data jenis barang pada bagian gudang



Gambar 4.22. Tampilan Menu Data Jenis Barang

4. Tampilan Tambah Jenis Barang

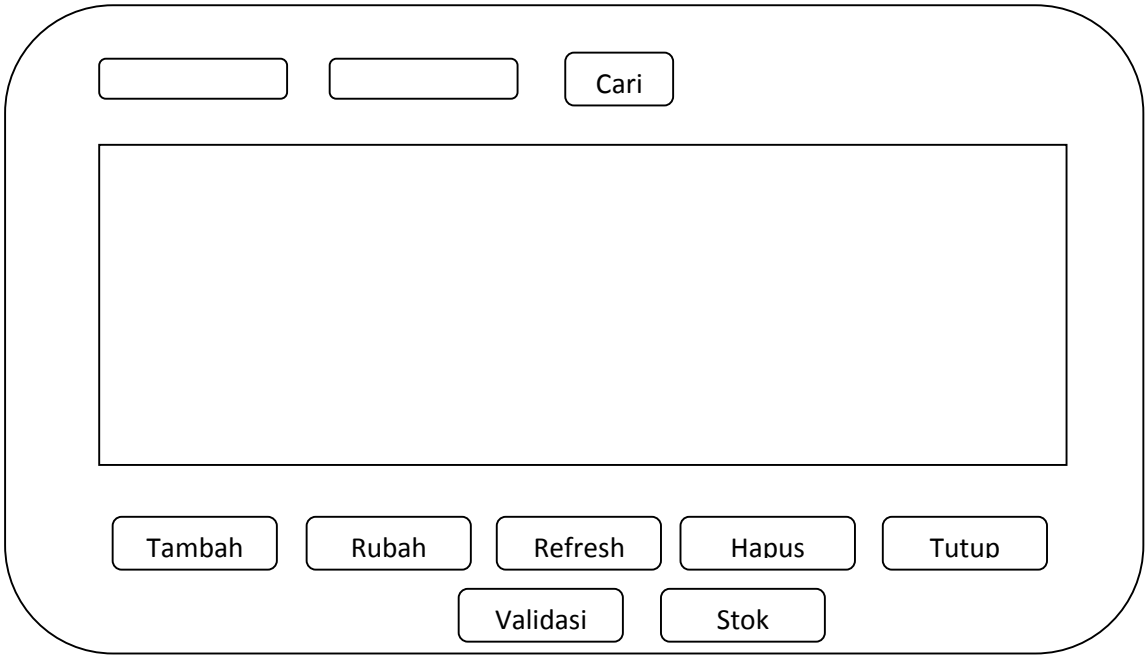
Pada tampilan ini bagian gudang menginputkan data jenis barang



Gambar 4.23. Tampilan Tambah Jenis Barang

5. Tampilan Data Barang

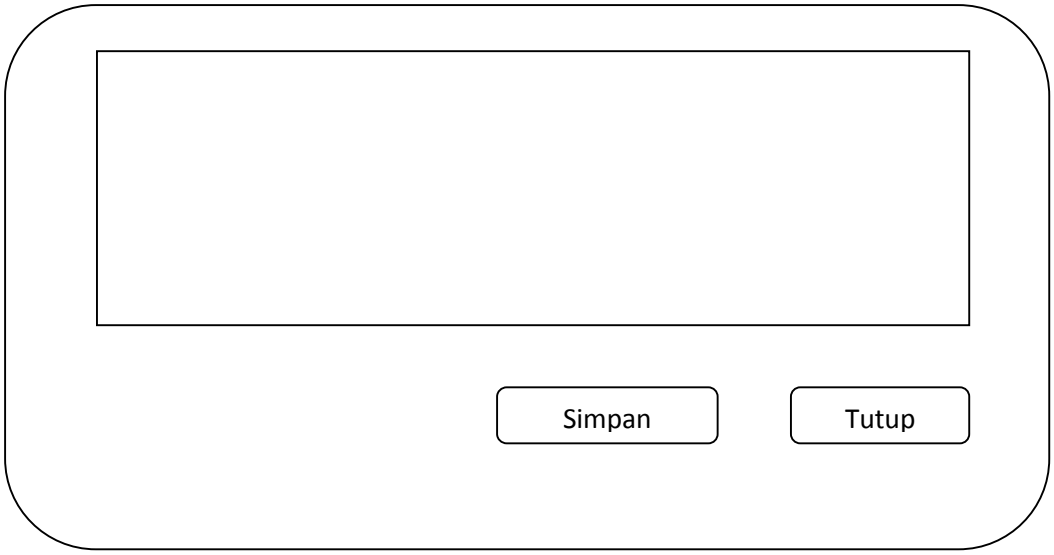
Ini adalah tampilan data barang pada bagian gudang



Gambar 4.24. Tampilan Data Barang

6. Tampilan Tamabah barang baru

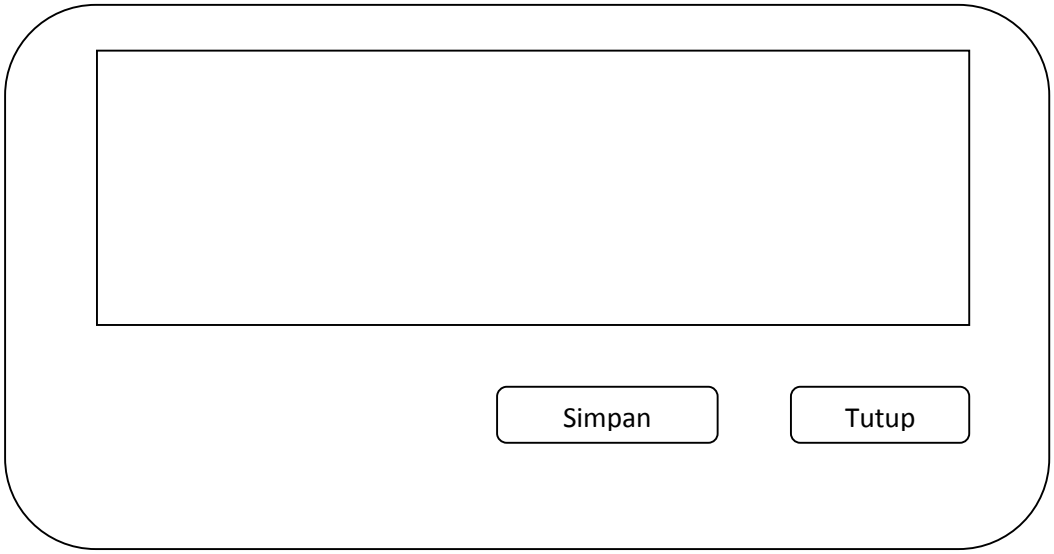
Ini adalah tampilan tambah barang baru



Gambar 4.25 Tampilan Tmabah barang baru

7. Tampilan Barang Lama

Pada bagian ini bagian gudang menambahkan barang lama



Gambar 4.26 Tampilan Tambah Barang Lama

6. Tampilan Stok barang

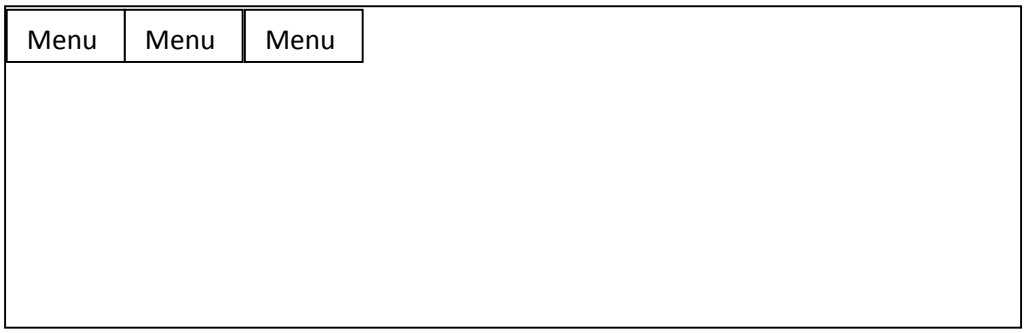
Ini adalah tampilan stok barang yang ada



Gambar 4.27 Tambilan Stok barang

7. Tampilan Menu Utama Bagian Produksi

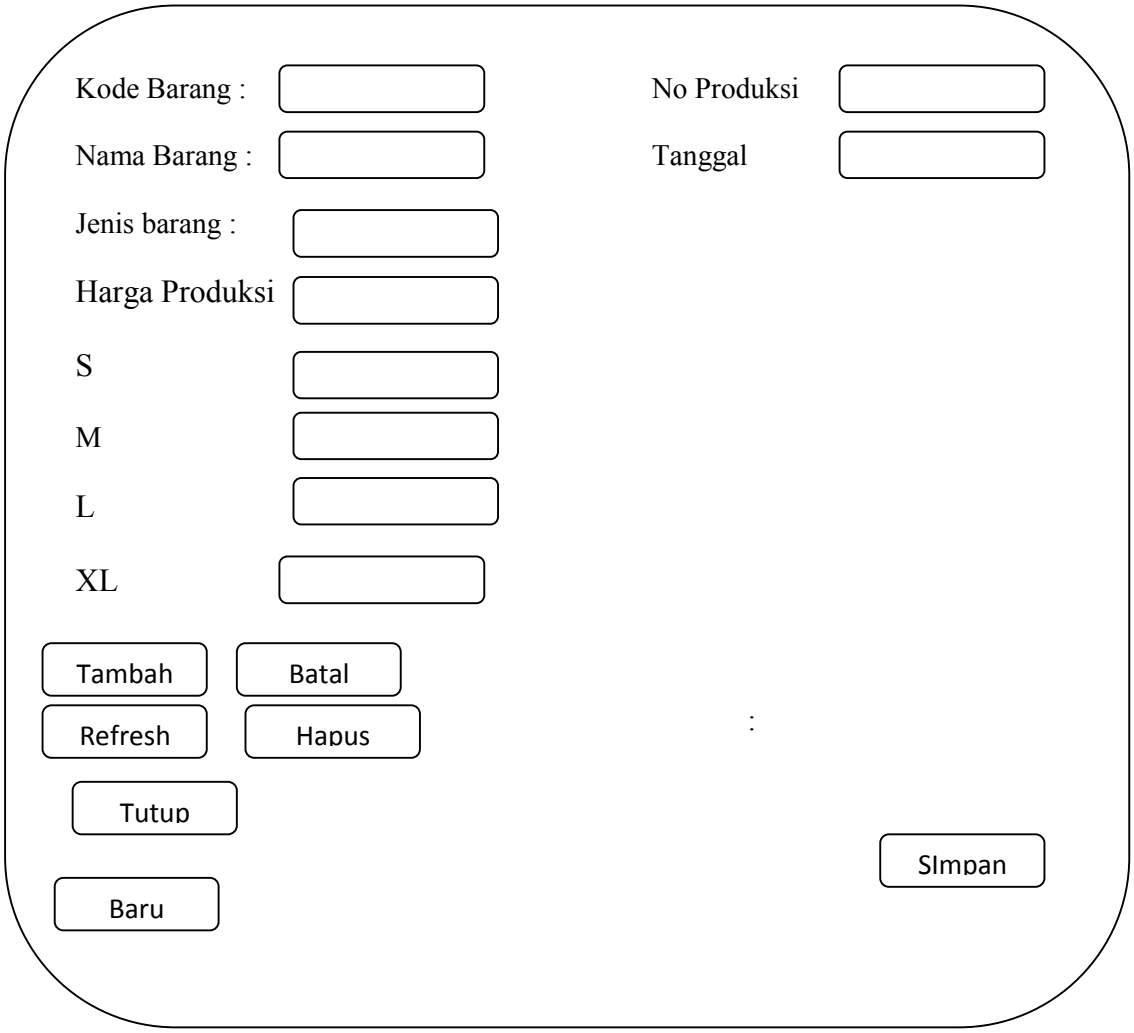
Ini adalag tampilan menu utama bagian produksi



Gambar 4.28 Tampilan utama bagian produksi

8. Tambah Barang Baru

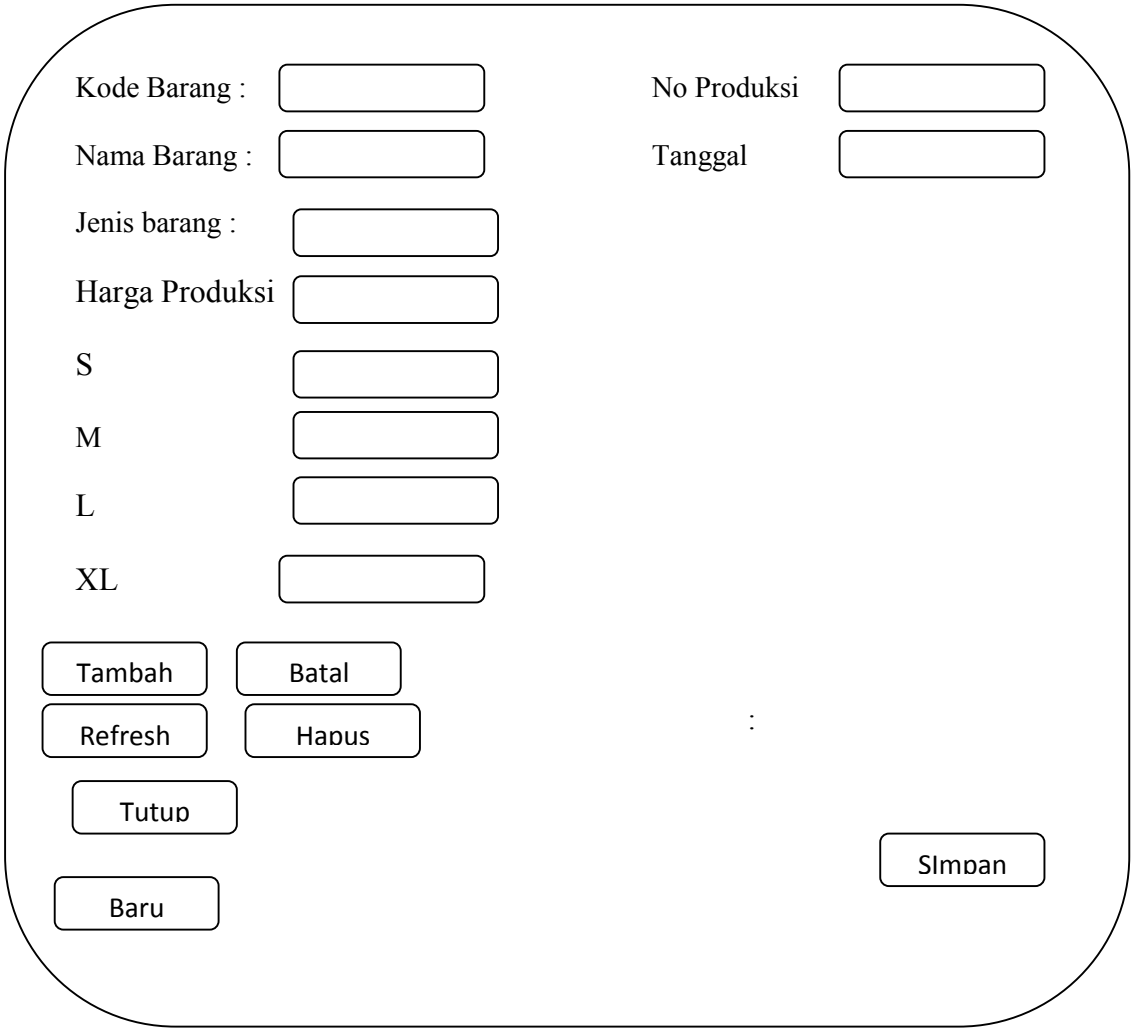
Pada bagian ini bagian produksi menambahkan barang baru



Gambar 4.29 Tampilan Tambah Barang Baru

9. Tampilan Tambah Barang lama

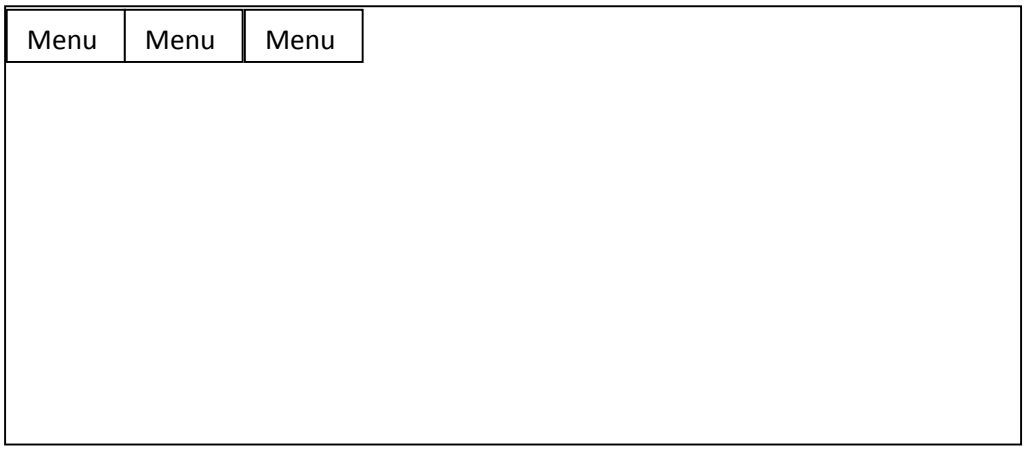
Pada bagian ini bagian produksi menambahkan barang lama



Gambar 4.30 Tampilan Tambah Barang Lama

10. Tampilan Menu Utama Bagian Kasir

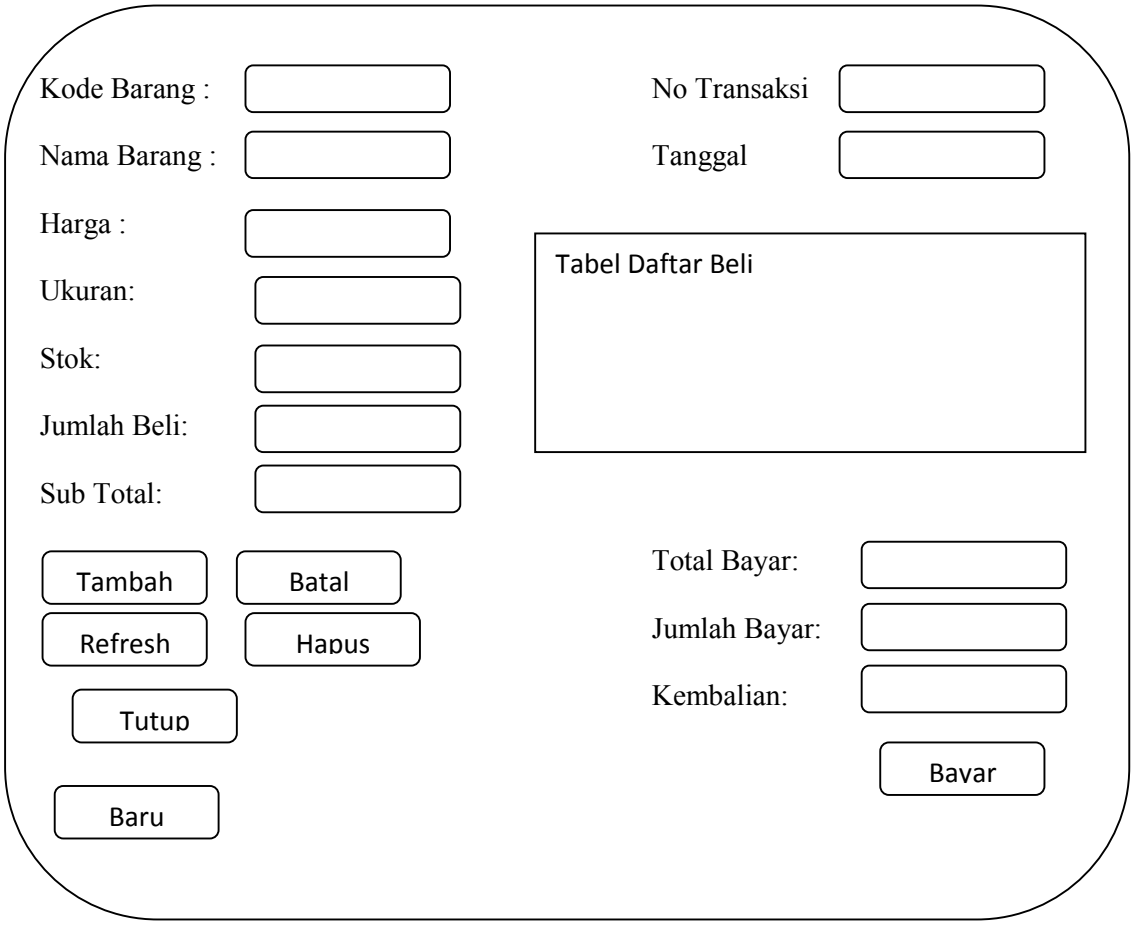
Ini adalah tampilan menu utama bagian kasir



Gambar 4.31 Tampilan Menu utama Bagian Kasir

9. Tampilan Transaksi Barang

Pada bagian ini kasir melakukan proses transaksi barang



Gambar 4.32 Tampilan Transaksi Barang

4.4.2 Perancangan Output

Perancangan output merupakan bentuk tampilan keluaran berupa laporan-laporan hasil transaksi.

1. Laporan Penjualan Barang

Ini adalah tampilan laporan penjualan barang



Gambar 4.33. Laporan Penjualan Barang

2. Laporan Stok Barang

Ini adalah tampilan laporan stok barang



Gambar 4.34. Laporan Stok Barang

3. Faktur Transaksi Barang

Ini adalah tampilan faktur dari transaksi barang

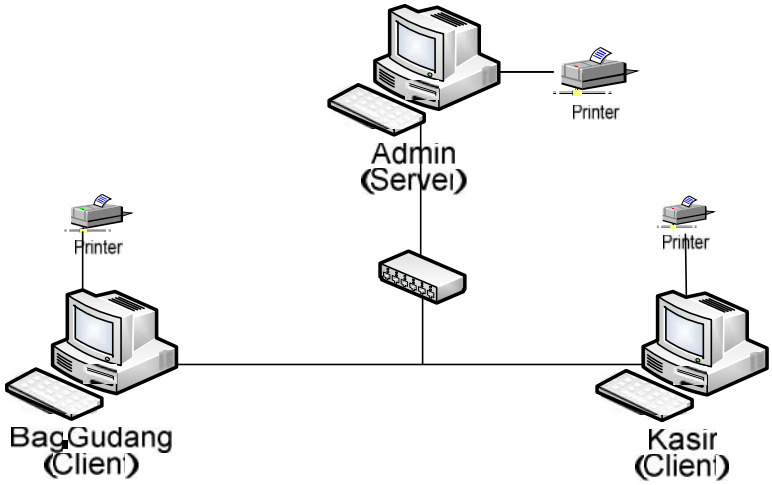


Gambar 4.35. Faktur Transaksi Barang

4.5 Perancangan Jaringan

Topologi jaringan adalah bentuk hubungan dari suatu jaringan. Jenis topologi jaringan yang dipakai adalah *Point To Point* (titik ke titik).

Jaringan kerja titik ke titik merupakan jaringan kerja yang paling sederhana tetapi dapat digunakan secara luas. Begitu sederhananya jaringan ini, sehingga seringkali tidak dianggap sebagai suatu jaringan tetapi hanya merupakan komunikasi biasa. Dalam hal ini, kedua simpul mempunyai kedudukan yang setingkat, sehingga simpul manapun dapat memulai dan mengendalikan hubungan dalam jaringan tersebut. Data dikirim dari satu simpul langsung kesimpul lainnya sebagai penerima, misalnya antara terminal dengan CPU.



Gambar 4.36. Arsitektur Jaringan