BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1.Analisis Sistem Yang Berjalan

Kegiatan analisis sistem yang berjalan merupakan kegiatan penguraian suatu sistem informasi yang utuh dan nyata ke dalam bagian-bagian atau komponen komputer dengan tujuan untuk mengidentifikasi serta mengevaluasi masalah yang muncul dari sistem tersebut. Sehingga mengarah kepada suatu solusi untuk perbaikan maupun pengembangan ke arah yang lebih baik serta sesuai dengan kebutuhan perkembangan teknologi.

4.1.1.Analisis Prosedur Yang Sedang Berjalan

Untuk mengetahui sistem yang sedang berjalan dan untuk mempelajari sistem yang ada, diperlukan suatu gambaran aliran-aliran informasi dari bagian-bagian yang terkait baik dari dalam maupun dari luar sistem.Hal ini memudahkan kita untuk memahami informasi-informasi yang didapat dan dikeluarkan oleh sistem itu sendiri. Adapun aliran informasi yang sedang berjalan di Toko Grosir Sunny Gemilang adalah sebagai berikut :

1. Dalam transaksi penjualan konsumen menyerahkan barang yang akan di beli kepada pegawai.

2. Pegawai mencatat data-data barang yang dibeli konsumen, jika pegawai lupa dengan harga barang yang dibeli, pegawai harus mencari dulu harga barang tersebut di buku catatan data barang.
3. Pegawai menghitung jumlah yang harus dibayar konsumen.
4. Konsumen memberikan pembayaran tunai kepada pegawai.
5. Pegawai memberikan barang yang dibeli konsumen.
6. Konsumen menerima barang yang dibeli dan uang kembalian jika memang ada kembalian.
7. Dalam transaksi pembelian pegawai akan mengetahui persediaan barang.jika ada barang yang kosong, pegawai akan mencatat data barang yang kosong tersebut.

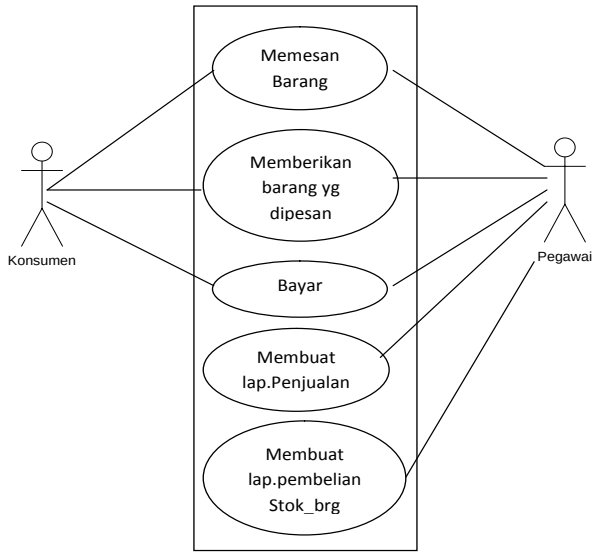
4.1.1.1.Aktor

Dalam menganalisis kebutuhan menggunakan pemograman berorientasi objek, pertama kita harus mengetahui aktor-aktor yang terlibat dalam sistem yang akan kita bangun. Adapun aktor-aktor yang berhubungan dengan sistem ini adalah sebagai berikut :

1. Konsumen
2. Pegawai

4.1.1.2.*Use Case Diagram*

Pemodelan ini dimaksudkan untuk menggambarkan kegiatan-kegiatan dan hubungan yang terjadi antara para aktor dan *use case* di dalam sistem yang sedang berjalan.Adapun *use case* sistem yang sedang berjalan di Toko Sunny Gemilang adalah sebagai berikut :



Gambar 4.1.*Use Case Diagram* yang sedang berjalan

4.1.1.3.Skenario*Use Case*

1. Nama Use Case : Penjualan

   Actor :Konsumen, dan Pegawai

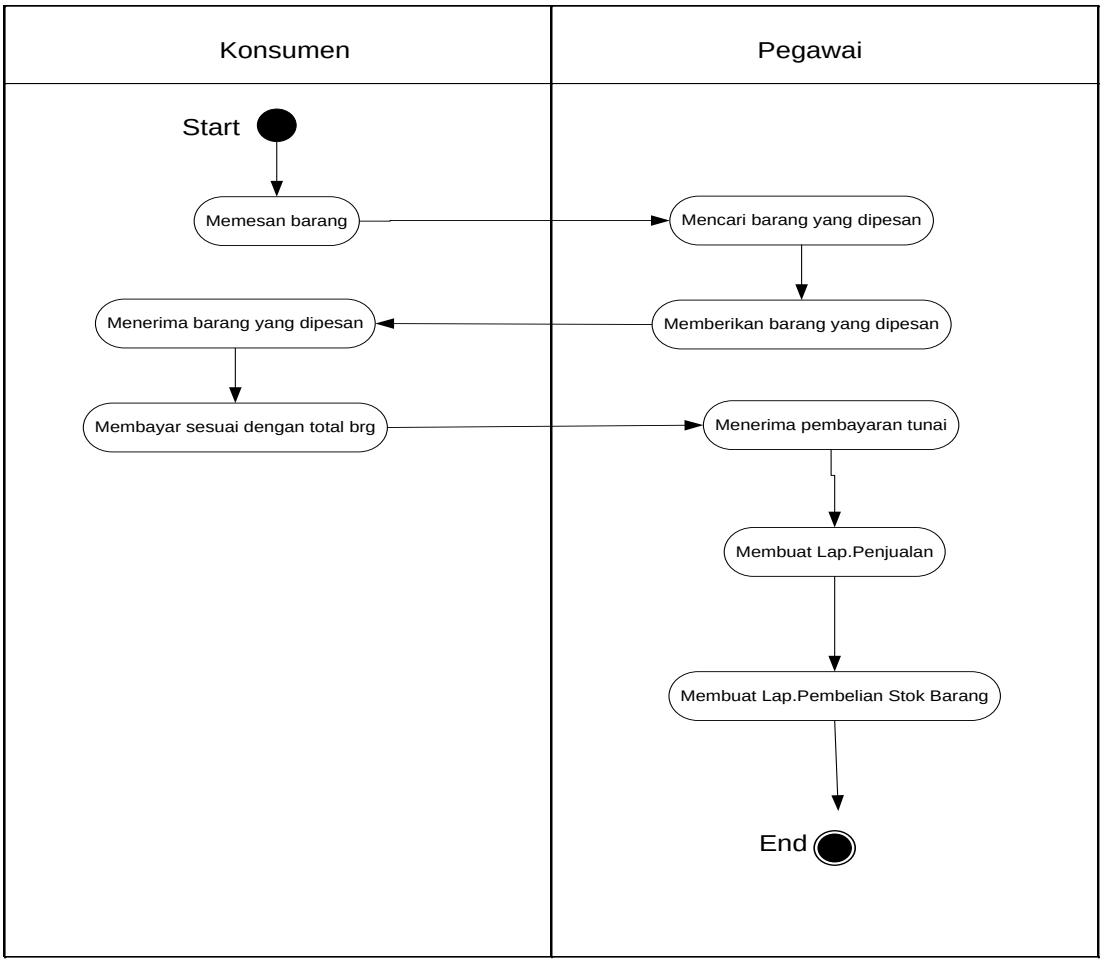
   Tujuan :Transaksi Penjualan Produk

Table 4.1.Skenario *Use Case* Menyerahkan Barang

Aksi Aktor	**Reaksi Sistem**
1.Konsumen memesan Barang	
	2.Pegawai Mencari barang yang dipesan konsumen
	3.Pegawai memberikan barang yang dipesan konsumen
4.Konsumen menerima barang yang dipesan	
5.Konsumen membayar sesuai dengan harga barang	
	6.Pegawai Menerima pembayaran tunai dari konsumen
	7.Pegawai membuat laporan penjualan
	8.Pegawai membuat laporan pembelian stok barang

4.1.1.4.*Activity Diagram*

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada dalam suatu sistem. Agar dapat lebih memahami tentang sistem yang akan dibuat, maka perlu dibuatkan *activity*diagram tentang sistem yang sedang berjalan. Berikut merupakan *activity* diagram sistem yang sedang berjalan di Toko Grosir Sunny Gemilang :



Gambar 4.2.*Activity Diagram* yang sedang berjalan

4.1.2.Evaluasi Sistem Yang Sedang Berjalan

Berdasarkan hasil pengamatan dan penelitian dari sistem pejualan di Toko Grosir Sunny Gemilang yang sedang berjalan, maka penulis mengevaluasi sistem tersebut sebagai berikut :

1. Kegiatan dalam pencatatan data penjualan masih ditulis tangan yang rentan salah perhitungan jumlah total bayar.

   Solusi : membuat Sistem Informasi Penjualan dan Pembelian dimana sistem ini lebih efektif dan akurat dalam perhitungan jumlah total bayar.
2. Pengecekan stok barang yang langsung mencari satu per satu barang kosong dimana membutuhkan waktu yang cukup lama.

   Solusi : membuat Sistem Informasi Penjualan dan Pembelian dimana sistem ini lebih efektif dan efisien dalam pengecekan stok barang.
3. Pencatatan daftar harga barang masih kedalam buku yang butuh waktu untuk mencari harga barang yang diinginkan.

   Solusi : membuat Sistem Informasi Penjualan dan Pembelian dimana sistem ini lebih efektif dan efisien dalam mencari harga barang yang diinginkan.
4. Tidak tersedianya laporan Penjualan

   Solusi : membuat Sistem Informasi Penjualan dan Pembelian dimana sistem ini lebih efektif dan efisien dalam pembuatan laporan penjualan.

4.2.Perancangan Sistem

Perancangan sistem adalah gambaran, perancangan dan pembuatan skema atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan mempunyai fungsi dan tujuan. Elemen-elemen sistem dirancang dengan tujuan untuk dikomunikasikan kepada *user*. Dalam pengembangan sistem dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang ada, baik secara keseluruhan maupun memperbaiki sistem yang telah ada.

Proses pengembangan sistem melewati beberapa tahapan, mulai dari sistem yang diterapkan, dioperasikan dan dipelihara. Bila sistem yang sudah dikembangkan masih timbul kembali permasalahan-permasalahan yang kritis serta dapat diatasi dalam tahap pemeliharaan sistem, maka perlu dikembangkan kembali suatu sistem untuk mengatasi dan proses ini kembali ketahap yang pertama yaitu perancangan sistem. Siklus ini disebut juga dengan siklus hidup suatu sistem.

4.2.1.Tujuan Perancangan Sistem

1. Memperbaiki pengolahan data penjualan, pembelian, barang, dan supplier pada Toko Grosir Sunny Gemilang.
2. Dapat menyimpan dan mengolah data transaksi penjualan barang dengan hasil laporannya.
3. Dapat mengetahui persediaan barang secara cepat dan akurat.

4.2.2.Gambaran Umum Sistem Yang Diusulkan

Sistem yang dirancang adalah Sistem yang mempunyai kelebihan dalam kecepatan dan keakuratan ,serta dapat mempermudah dalam memasukan data barang, data pembelian, dan proses transaksi penjualan barang. Transaksi tersebut yang secara keseluruhan tersimpan dalam sebuah *database*, sehingga dari sistem aplikasi yang dirancang dapat meminimalisir adanya kesalahan data. Sistem yang dirancang memiliki fasilitas untuk melakukan proses pembuatan laporan penjualan.

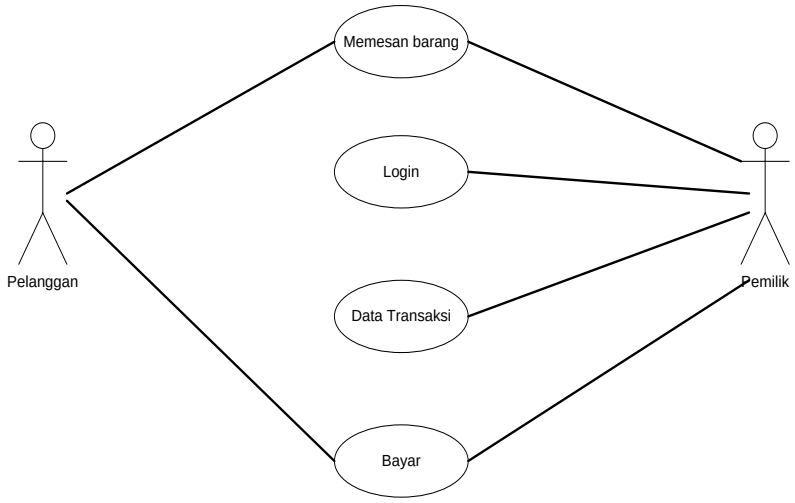
Sistem ini menggunakan bahasa pemograman Java yang berbasis objek dengan program aplikasi yang digunakan adalah Netbeans IDE 7.1.2 dan MySQL sebagai program aplikasi untuk membangun databasenya.

4.2.3.Perancangan UML Yang Diusulkan

Perancangan UML yang diusulkan meliputi *Use Case* Diagram, Skenario *Use Case*, *Activity* Diagram, *Sequence* Diagram, *Class* Diagram, dan *Deployment* Diagram.

4.2.3.1.Use Case Diagram

Use Case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" diperbuat sistem, dan bukan "bagaimana". Gambaran *Use Case* diagram Sistem Aplikasi Bisnis



Gambar 4.3.*Use case Diagram* Sistem Penjualan dan Pembelian

4.2.3.2.Skenario *Use Case*

Aliran proses sebelumnya, dapat dideskripsikan atau diuraikan lebih lengkap lagi dengan menggunakan suatu skenario *use case*. Yaitu dengan menggunakan table-tabel yang berisikan Nama, Tujuan, Deskripsi singkat dan Aktor yang terlibat di dalamnya. Dan pada bagian skenario diuraikan aksi dari aktor yang terlibat dan bagaimana reaksi dari sistem.

1. Nama Use Case :Login

   Actor :Pemilik

   Tujuan :Proses Validasi bagian penjualan

Table 4.2.Skenario *Use Case Login User*

Aktor	Sistem
1.Masukan User Name dan Password	
	2.Validasi User Name dan Passsword
	3.Menampilkan pesan Login tidak valid
4.Memasukan kembali user name dan password	
	5.Validasi user name dan password
	6.Masuk ke menu utama

2. Nama Use Case :Transaksi Penjualan

   Actor :Pemilik dan Konsumen

   Tujuan :Mengolah data penjualan

Table 4.3.Skenario *Use Case* Transaksi Penjualan

Aktor	Sistem
1.Input Kode Barang	
	2.Menampilkan Deskripsi data barang
3.Input Jumlah barang	
	4.Menampilkan total bayar

5.Input jumlah bayar	
6.Cetak struk penjualan	
	7.Menampilkan cetak struk

3. Nama Use Case :Transaksi Beli Barang

   Actor :Pemilik

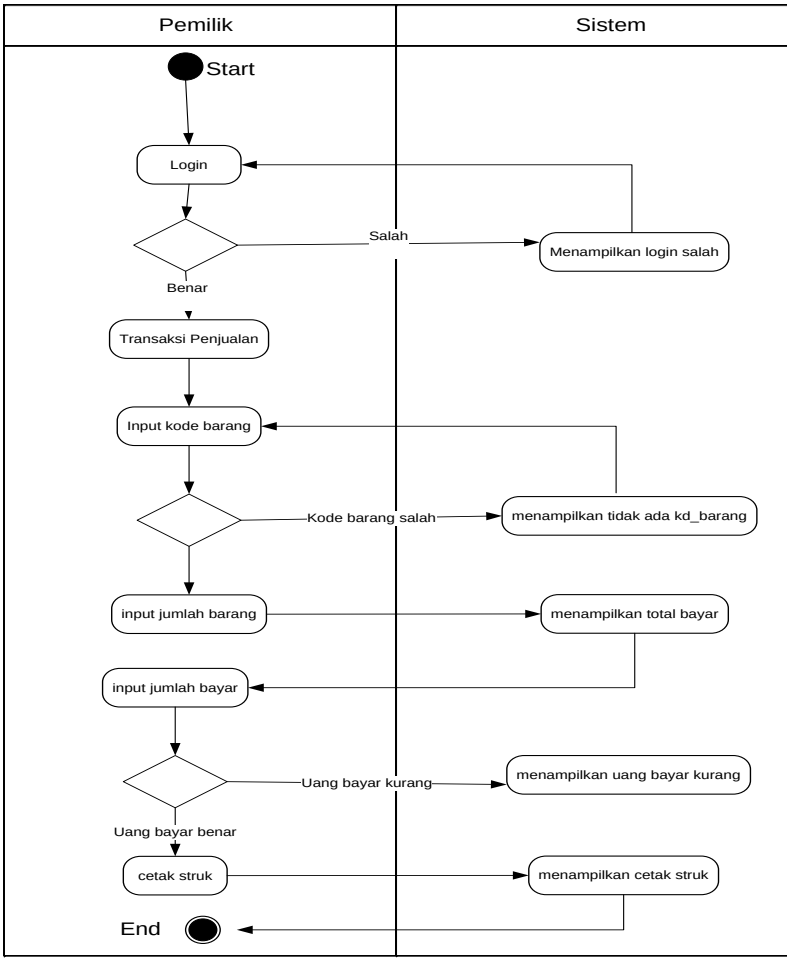
   Tujuan :Mengolah data pembelian barang

Table 4.4.Skenario *Use Case* Transaksi Beli Barang

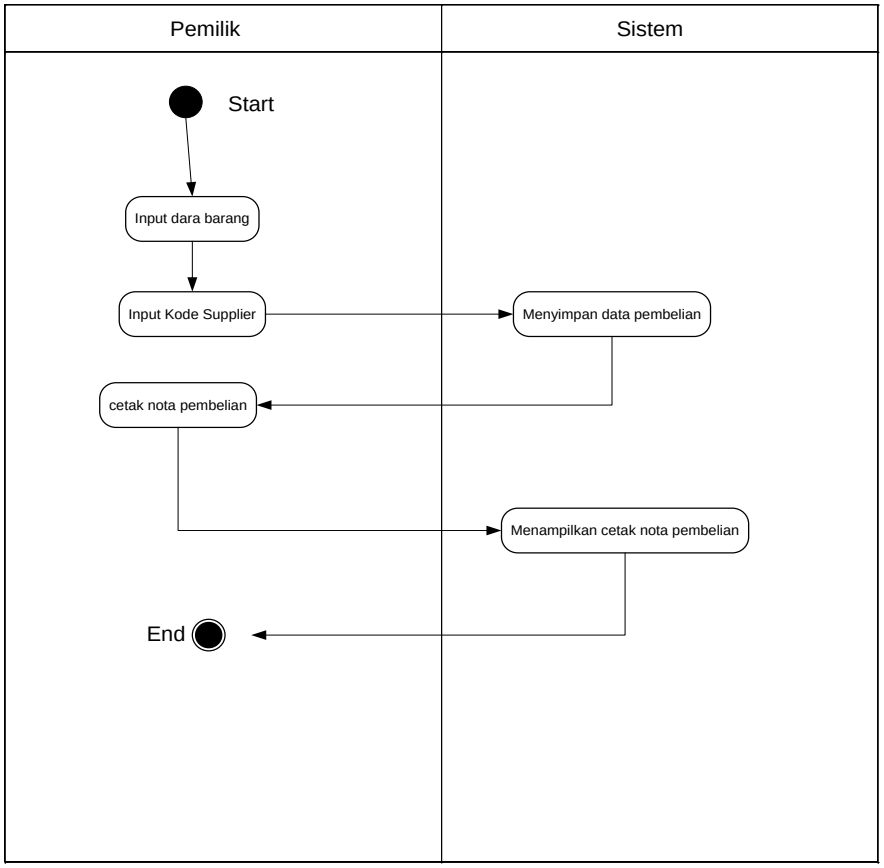
Aktor	Sistem
1.Input Kode Barang	
	2.Menampilkan Deskripsi data barang
3.Input Jumlah barang	
	4.Menampilkan total bayar
5.Input jumlah bayar	
6.Cetak struk penjualan	
	7.Menampilkan cetak struk

4.2.3.3.Activity Diagram

Aliran-aliran kegiatan atau kerja yang terjadi di dalam sistem aplikasi bisnis ini, akan digambarkan dalam *Activity Diagram*. *Activity diagram* memodelkan *workflow* proses bisnis dan urutan aktivitas dalam sebuah proses.



Gambar 4.4.*Activity Diagram* Sistem penjualan pada Toko Grosir Sunny Gemilang yang diusulkan

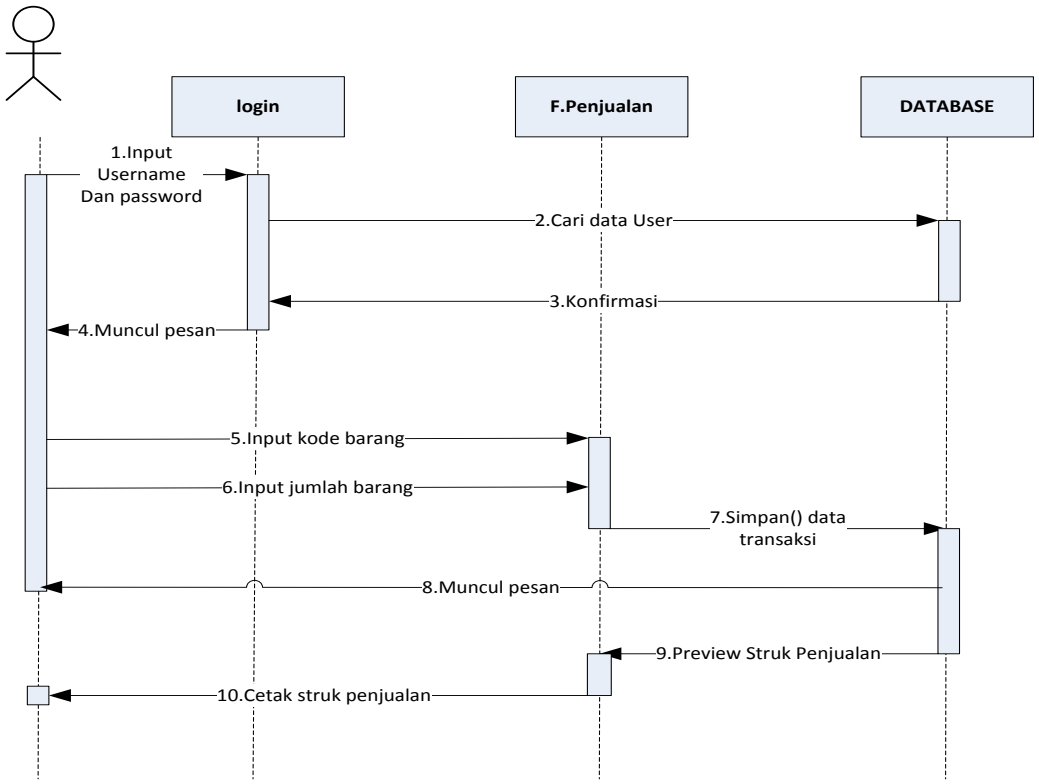


Gambar 4.5.*Activity Diagram* Data Pembelian

4.2.3.4.*Sequence* Diagram

Menggambarkan kolaborasi dinamis antara sejumlah *object.* Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim atara *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama sequence diagram terdiri atas objek yang dituliskan dengan segiempat bernama pesan diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan proses vertikal.

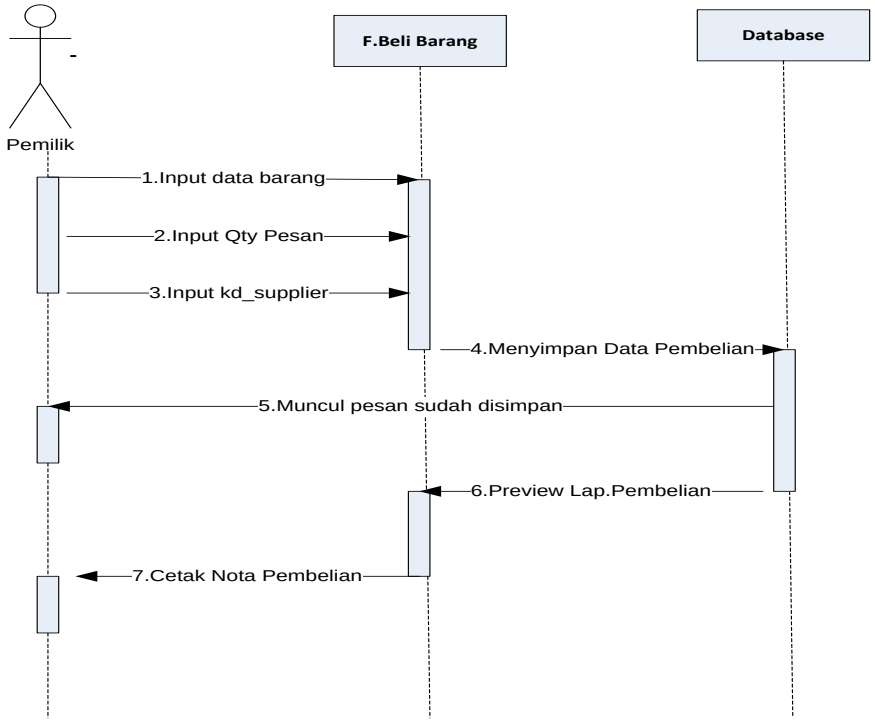
1. *Sequence Diagram* Sistem Penjualan



Gambar 4.6.*Sequence Diagram* Sistem Penjualan

-

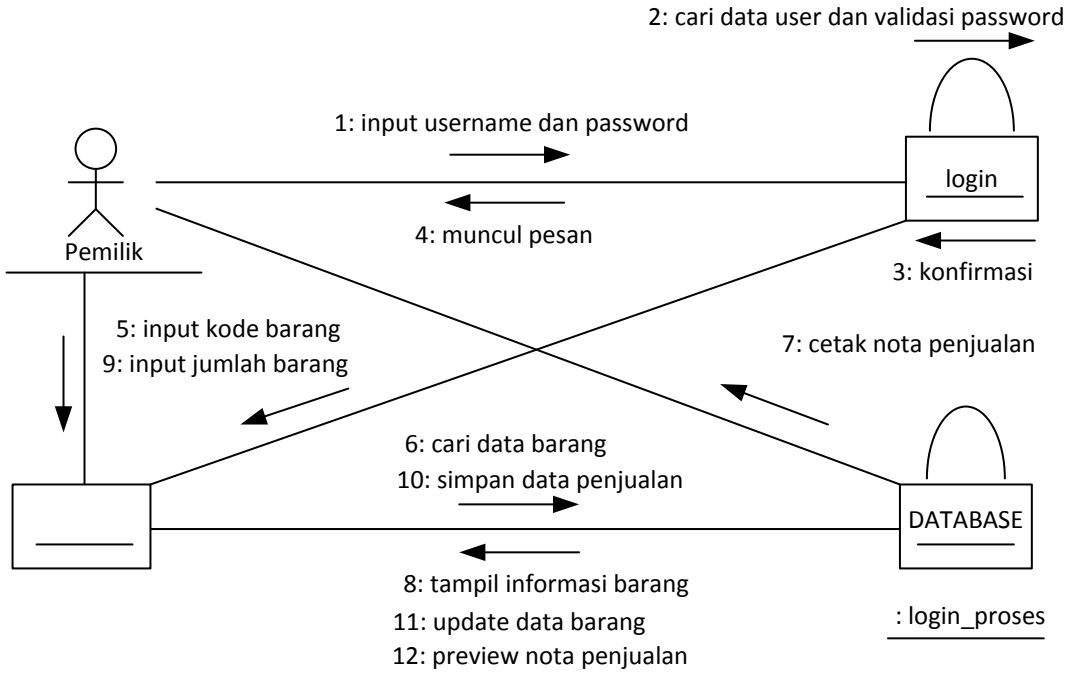
2. *Sequence Diagram* Pembelian



Gambar 4.7.*Sequence Diagram* Pembelian

4.2.3.5. Collaboration Diagram

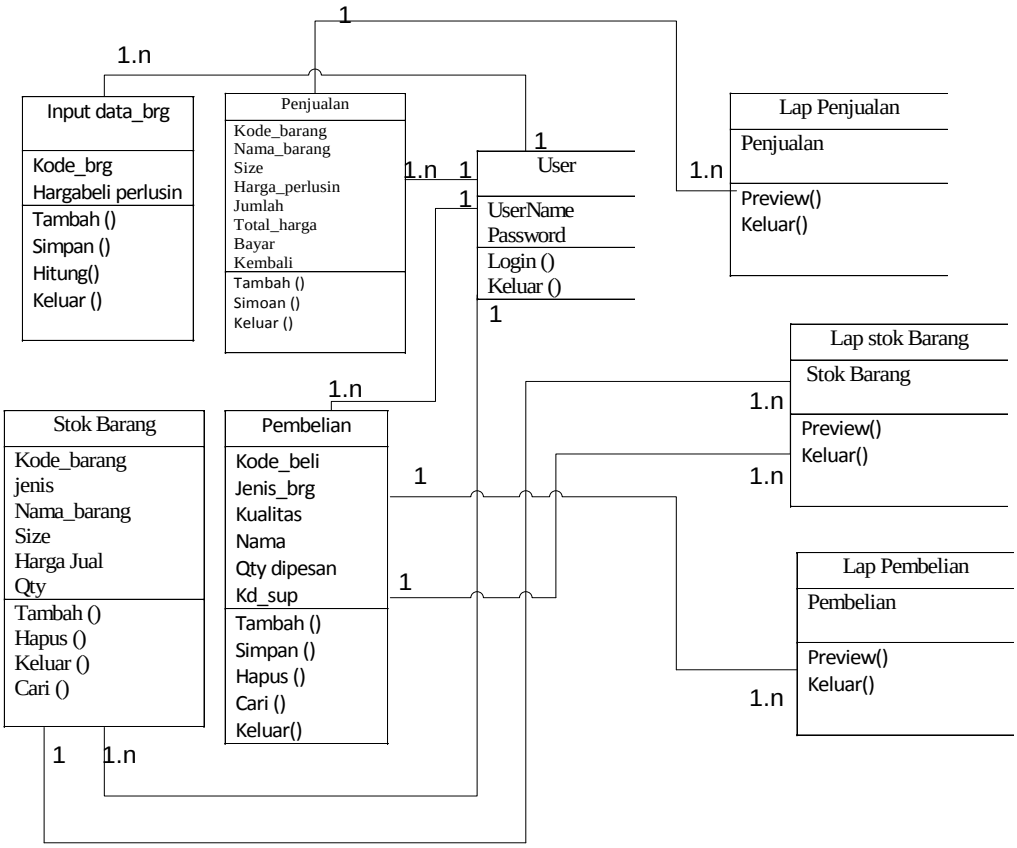
Collaboration diagram memberi sebuah cara mengelompokkan potongan-potongan behavior interaksi saat peran-peran dimainkan oleh class yang berbeda. Interaksi penjual dengan sistem pada proses penjualan tanpa penggambaran orientasi waktu, digambarkan oleh gambar collaboration diagram berikut :



Gambar 4.8. *Collaboration Diagram* Penjualan

4.2.3.6.*Class* Diagram

Menggambarkan struktur statis *class* di dalam sistem.*Class* merepresentasikan sesuatu yang ditangani oleh sistem .

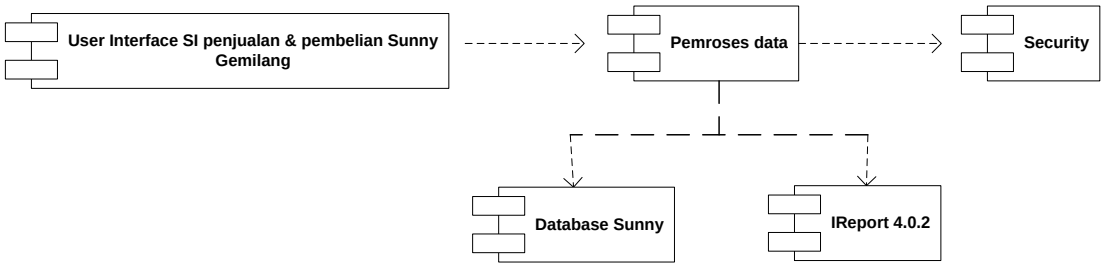


Gambar 4.9.*Class* Diagram

4.2.3.7.Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen perangkat lunak, termasuk ketergantungan di antaranya.

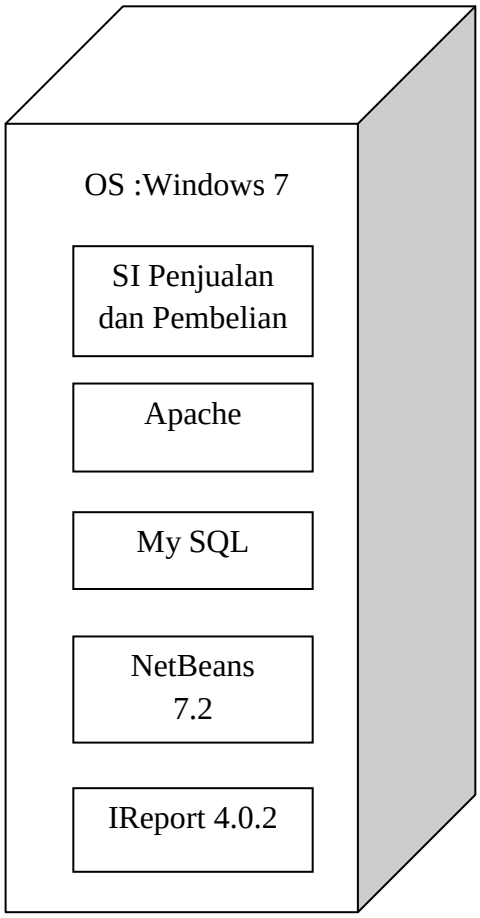
Component perangkat lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link time, maupun run time. Umumnya komponen terbentuk dari beberapa class dan/atau package, tapi dapat juga dari komponen yang lebih kecil. Komponen dapat juga berupa interface, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain. Berikut ini adalah Component Diagram yang dibutuhkan :



Gambar 4.9.*Component* Diagram

4.2.3.8.*Deployment* Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-deploy dalam *infrastruktur* sistem, dimana 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.



Gambar 4.10.*Deployment* Diagram

4.2.4.Perancangan Antar Muka

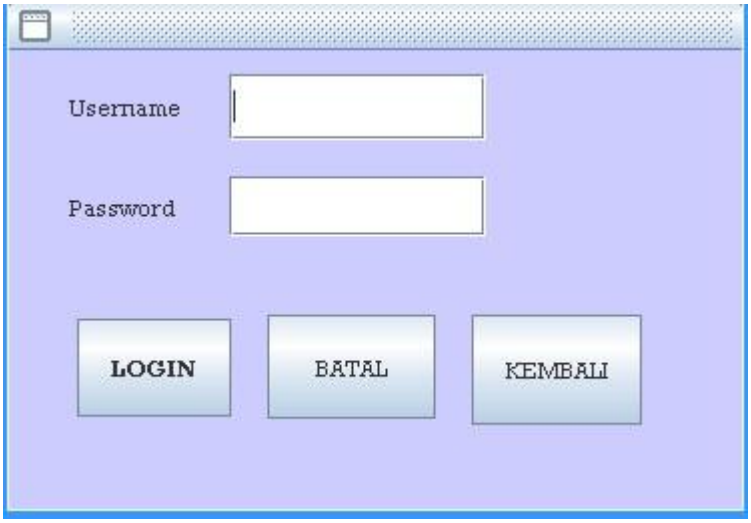
Antar muka adalah suatu media yang digunakan untuk komunikasi antar manusia (*user*) dengan komputer, oleh sebab itu aplikasi ini didesain agar *user* dapat dengan mudah menggunakannya atau mengoperasikannya .

4.2.4.1. Perancangan *Input*

Desain *input* merupakan awal di mulainya suatu proses informasi, dimana bahan bahan mentah dari informasi atau data, yang terdiri dari transaksi, entri, angka-angka, dan grafik atau tabel yang dilakukan oleh suatu organisasi. Akurat tindaknya suatu data dari sistem informasi tidak lepas dari data yang dimasukan. Adapun perancangan *input* dari sistem aplikasi bisnis ini adalah sebagai berikut :

1. *Login User*

   Digunakan oleh *user* dalam melakukan *Login* agar dapat mengakses menu utama, dimana *user* harus mempunyai *password* sehingga dapat menjaga kerahasiaan data yang ada.



Gambar 4.11.*Login User*

2. Menu Utama

Menu ini digunakan untuk meberikan kemudahan bagi *user* dalam mengelola data yang ada, selain itu untuk memudahkan *user* dalam membuat laporan sehingga kecepatan dan ketepatan pembuatan laporan bisa dilakukan dengan mudah.



Gambar 4.12.Menu Utama

3. Transaksi Penjualan

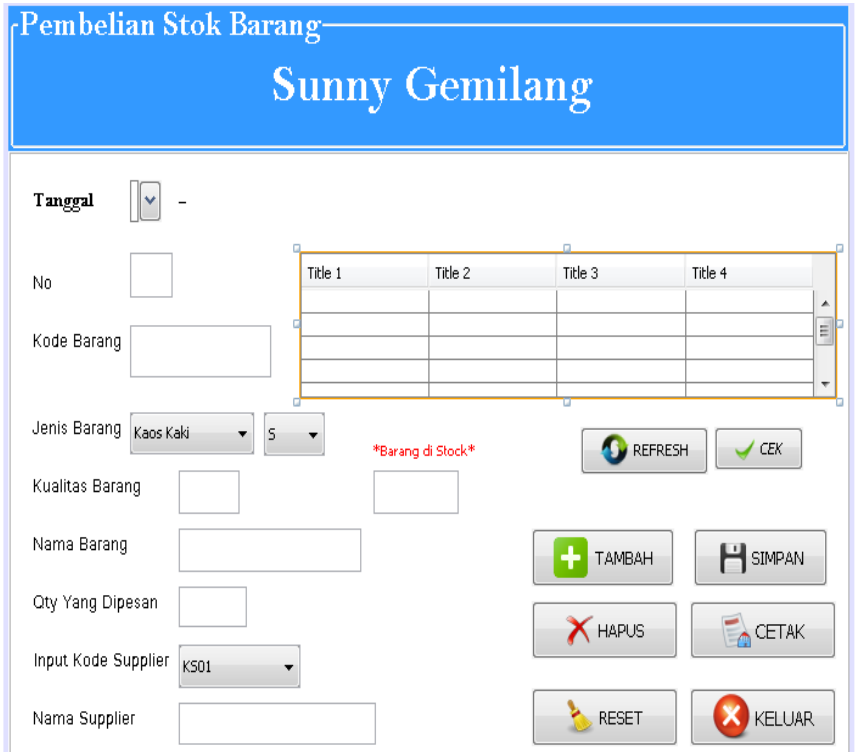
Berikut rancangan tampilan data transaksi :



Gambar 4.13.Transaksi Penjualan

4. Transaksi Pembelian Barang

Berikut rancangan tampilan data pembelian barang:



Gambar 4.14.Pembelian Barang

5. Data Stok Barang

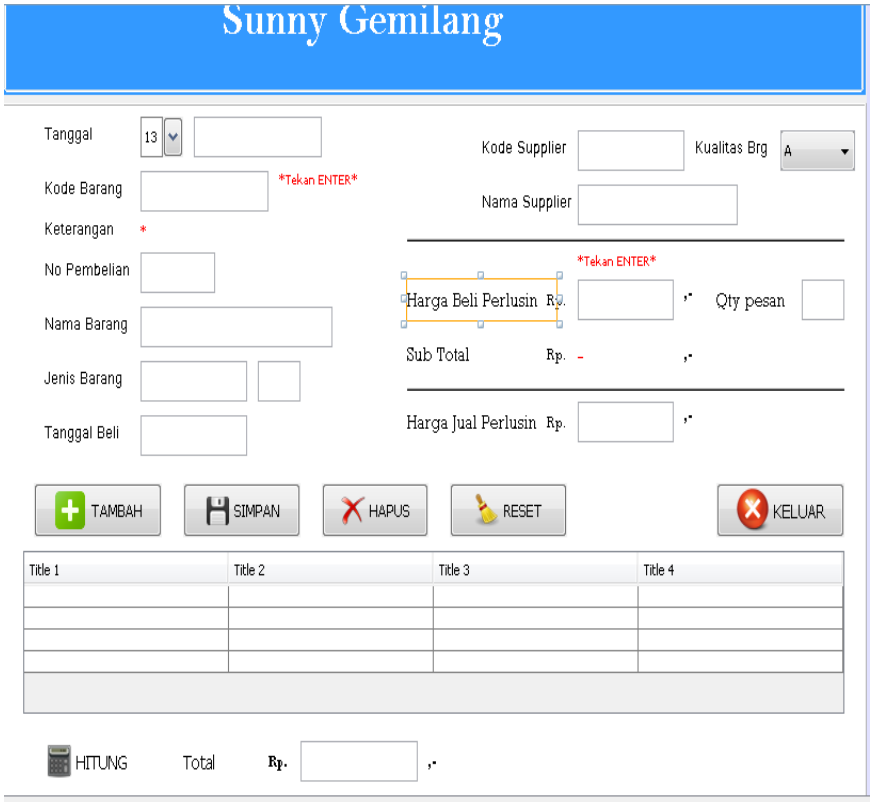
Berikut rancangan tampilan data stok barang:



Gambar 4.15.Stok Barang

6. Input Data Pembelian

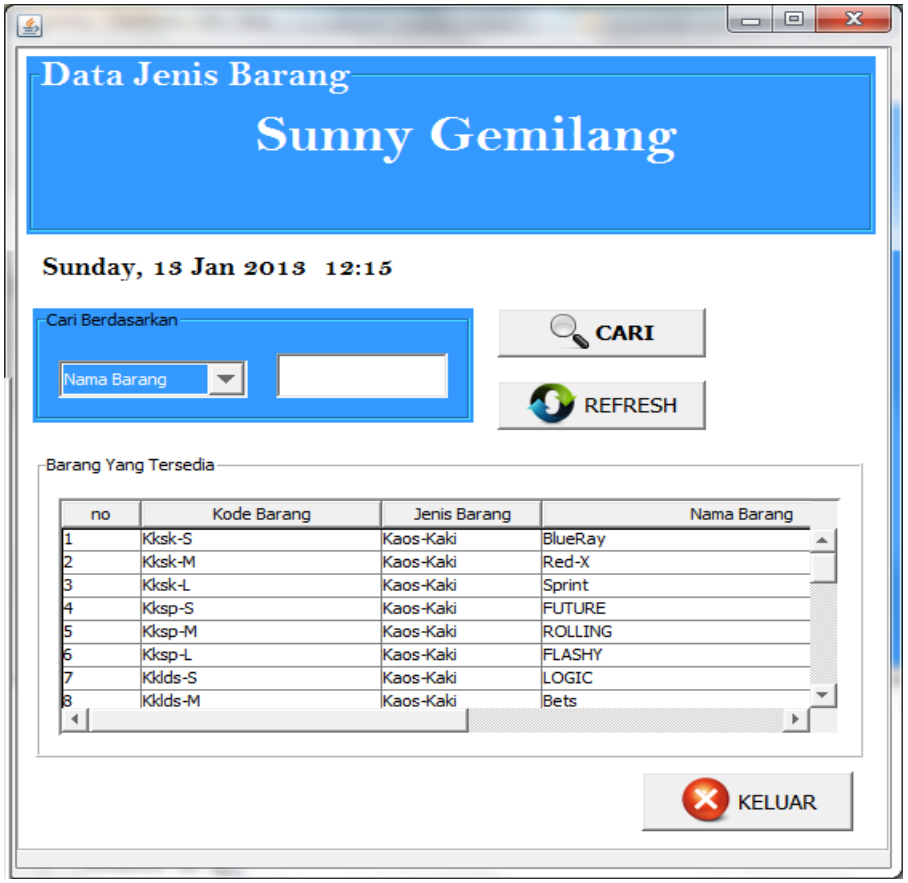
Berikut rancangan tampilan Input Data Pembelian:



Gambar 4.16.Data Pembelian

7. Data Jenis Barang

Berikut rancangan tampilan Input Data Jenis Barang:



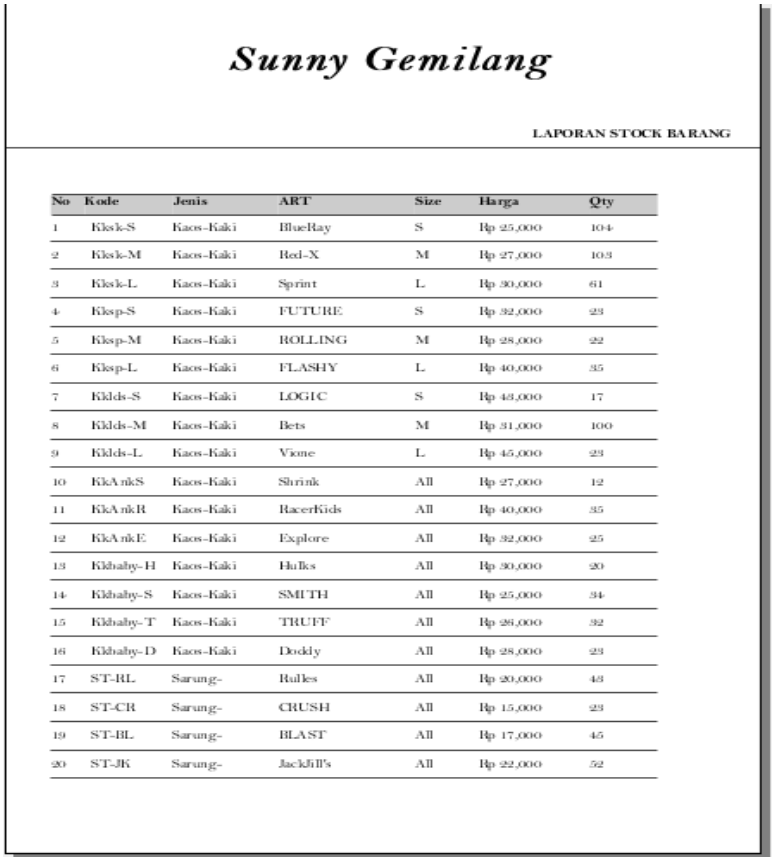
Gambar 4.17.Jenis Barang

4.2.4.2. Perancangan *Output*

Perancangan *output* merupakan rancangan informasi yang dihasilkan dari pengolahan data penjualan. Adapun perancangan *output* dari sistem aplikasi bisnis ini adalah sebagai berikut :

1. Laporan Data Stok Barang

   Berikut rancangan laporan data stok barang :



Sunny Gemilang

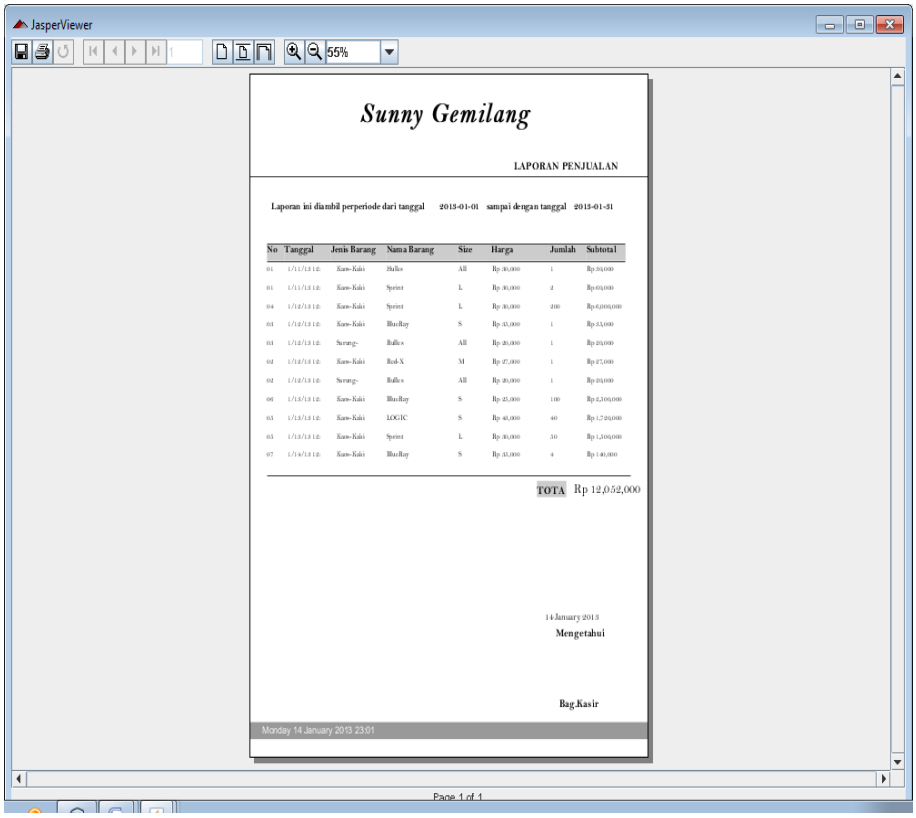
LAPORAN STOCK BARANG

No	Kode	Jenis	ART	Size	Harga	Qty
1	Klsk-S	Kaos-Kaki	BlueRay	S	Rp 25,000	104
2	Klsk-M	Kaos-Kaki	Red-X	M	Rp 27,000	103
3	Klsk-L	Kaos-Kaki	Sprint	L	Rp 30,000	61
4	Klsp-S	Kaos-Kaki	FUTURE	S	Rp 32,000	23
5	Klsp-M	Kaos-Kaki	ROLLING	M	Rp 28,000	22
6	Klsp-L	Kaos-Kaki	FLASHY	L	Rp 40,000	35
7	Klds-S	Kaos-Kaki	LOGIC	S	Rp 43,000	17
8	Klds-M	Kaos-Kaki	Bets	M	Rp 31,000	100
9	Klds-L	Kaos-Kaki	Vione	L	Rp 45,000	23
10	KkAnkS	Kaos-Kaki	Shrink	All	Rp 27,000	12
11	KkAnkR	Kaos-Kaki	RacerKids	All	Rp 40,000	35
12	KkAnkE	Kaos-Kaki	Explore	All	Rp 32,000	25
13	Klbaby-H	Kaos-Kaki	HuIks	All	Rp 30,000	20
14	Klbaby-S	Kaos-Kaki	SMITH	All	Rp 25,000	34
15	Klbaby-T	Kaos-Kaki	TRUFF	All	Rp 26,000	32
16	Klbaby-D	Kaos-Kaki	Dockly	All	Rp 28,000	23
17	ST-RL	Sarung-	Rulles	All	Rp 20,000	43
18	ST-CR	Sarung-	CRUSH	All	Rp 15,000	23
19	ST-BL	Sarung-	BLAST	All	Rp 17,000	45
20	ST-JK	Sarung-	JackJill's	All	Rp 22,000	52

Gambar 4.18.Laporan Stok Barang

2. Laporan Laporan Penjualan

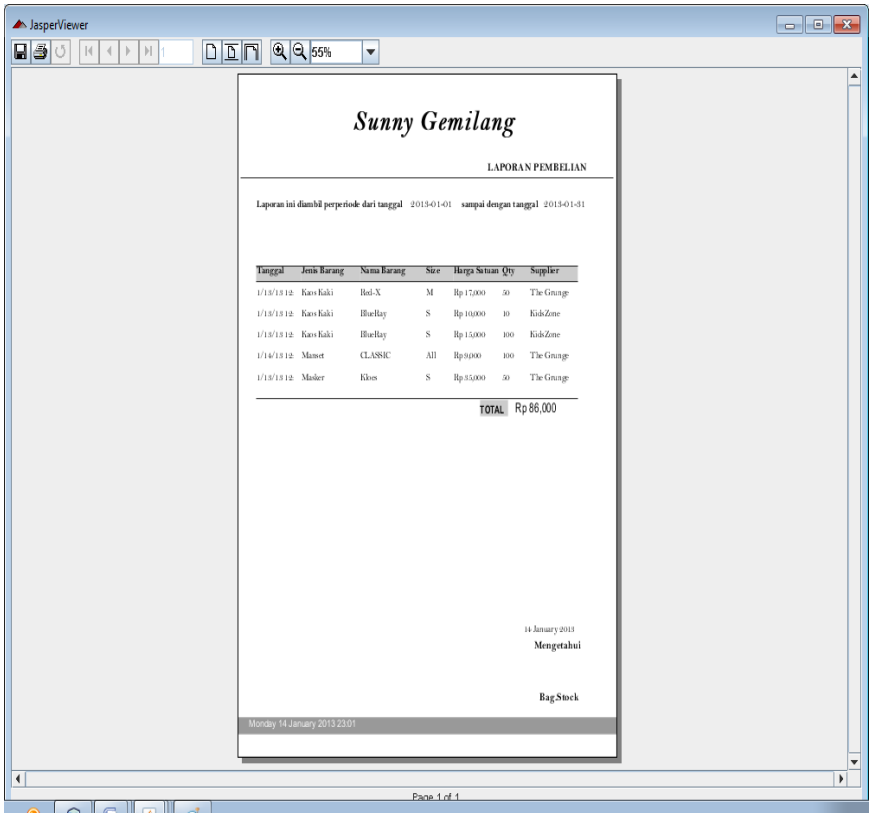
Berikut rancangan laporan penjualan :



Gambar 4.19.Laporan Penjualan

3. Laporan Pembelian

Berikut rancangan laporan pembelian :



Gambar 4.20.Laporan Pembelian