BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Proses analisis sistem yang telah dilakukan sebelumnya memberikan informasi-informasi mengenai sistem yang sedang berjalan, termasuk kelemahan pada sistem tersebut. Berdasarkan hasil evaluasi sistem yang sedang berjalan, maka sistem yang telah ada perlu dikembangkan. Pengembangan sistem dilakukan dengan mengubah atau memperbaiki sistem yang masih manual kedalam sistem yang terkomputerisasi.

Setelah memahami sistem yang sedang berjalan dan kriteria-kriteria sistem yang akan dibangun, maka tahap selanjutnya adalah membuat perancangan sistem informasi terlebih dahulu. Pada tahap ini akan dilakukan perancangan perangkat lunak untuk sistem informasi persediaan barang. Perancangan perangkat lunak ini akan dikembangkan berdasarkan hasil analisis sistem yang telah dilakukan.

4.1.1 Tujuan Perancangan Sistem

Tujuan perancangan sistem ini adalah untuk menganalisis perancangan sistem informasi persediaan barang di Throve Clothing. Perancangan ini diusulkan untuk langkah lebih mengefektifitaskan dan mengefisienkan waktu pengelolaan data, dan penyajian laporan.

Perancangan sistem dibuat untuk memberikan gambaran secara umum mengenai sistem yang dikembangkan atau sistem yang baru kepada

pemakai (user). Tahap perancangan sistem mempunyai dua tujuan utama, yaitu :

1. Untuk memenuhi kebutuhan sistem.
2. Untuk memberikan gambaran yang jelas kepada pembuat program.

Tujuan kedua lebih condong kepada perancangan sistem yang terinci, yaitu rancangan pembuatan, rancangan bangun yang jelas dan lengkap untuk digunakan dalam pembuatan program sistem informasi persediaan barang.

4.1.2. Gambaran Umum Sistem Yang Diusulkan

Sistem informasi persediaan barang ini dapat digambarkan sebagai sebuah fasilitas penunjang kerja bagian kasir dan bagian gudang untuk mengelola data dan memberikan informasi yang akurat mengenai kegiatan pengelolaan persediaan barang pada Throve Clothing, mulai dari informasi persediaan barang, barang kepada *supplier*, serta informasi penjualan atau transaksi atas penjualan produk kepada *customer*.

Dengan adanya perancangan sistem informasi persediaan barang berbasis *client server* ini diharapkan dapat menyelesaikan permasalahan yang terdapat pada sistem yang lama, dan permasalahan tersebut dapat diselesaikan dengan perubahan cara proses pengolahan data yang ada pada sistem yang lama, yang menyimpan data pada bentuk arsip dokumen, sedangkan sistem yang baru dilakukan dengan cara terkomputerisasi dimana penyimpanan data dilakukan pada suatu wadah yang disebut data base.

4.1.3. Perancangan Prosedur Yang Diusulkan

Adapun perancangan sequence yang diusulkan adalah sebagai berikut :

4.1.3.1 Perancangan Basis Data Definisi Aktor Yang Diusulkan

Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.

Tabel 4.1 Definisi Aktor yang Diusulkan

No	Aktor	Deskripsi
1.	General Manager	Pihak yang bertanggung jawab dalam memberikan akses login, serta menerima semua laporan.
2.	Bag. Penjualan	Pihak yang bertanggung jawab dalam mengelola data konsumen dan proses transaksi penjualan.
3.	Bag. Gudang	Pihak yang bertanggung jawab dalam mengelola data data barang.
4.	Bag. Pembelian	Pihak yang bertanggung jawab dalam mengelolah data supplier dan proses pembelian barang ke supplier.

4.1.3.2 Diagram Use Case Yang Diusulkan

Diagram use case memperlihatkan pada kita hubungan - hubungan yang terjadi antara aktor - aktor dengan use case - use case dalam sistem. Adapun use case diagram sistem informasi *inventory* yang di usulkan adalah sebagai berikut:



Gambar 4.1 Diagram Use Case Yang Diusulkan

4.1.3.3. Definisi Use Case Yang Diusulkan

Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama use case. Use case mendeskripsikan apa yang sistem (atau subsistem, kelas, atau antarmuka) kerjakan tetapi ia tidak

menspesifikasikan (dan tidak memiliki kompetensi untuk menspesifikasi) bagaimana ia melakukannya.

Tabel 4.2 Definisi Use Case dan Deskripsi Yang Diusulkan

No	Use Case	Deskripsi
1.	*Login*	Merupakan proses pengecekan hak akses siapa yang berhak mengakses, *login* wajib untuk fungsi-fungsi yang berkaitan dengan akses pengubahan ke basis data, oleh karena itu fungsi-fungsi yang melakukan perubahan basis data harus mengecek validasi *user* yang mengakses fungsi-fungsi ini.
2.	Input data konsumen	Merupakan proses pemasukan data konsumen
3.	Input data supplier	Merupakan proses pemasukan data supplier
4.	Input data barang	Merupakan proses pemasukan data barang
5.	Input data pembelian barang	Merupakan proses pembuatan daftar barang yang akan di beli ke supplier
6.	Input transaksi penjualan	Merupakan proses pemasukan data barang yang dibeli oleh konsumen.
7.	Laporan	Merupakan proses dimana General Manager bisa mengakses laporan, baik itu laporan transaksi penjualan, pembelian barang, dan lain-lain.

4.1.3.4 Skenario Use Case Yang Diusulkan

Skenario use case merupakan narasi tentang aktivitas dalam suatu use case diagram. Adapun skenario use case dalam sistem yang diusulkan adalah sebagai berikut:

1) Skenario Use Case Login Yang Diusulkan

Interaksi antara aktor pengguna, yaitu General Manager, Bagian Gudang dan Bagian Penjualan, dengan use case login yang dijelaskan dalam skenario use case sebagai berikut :

Tabel 4.3 Skenario Use Case Login Yang Diusulkan

Identifikasi	
Nama Use Case	Login
Aktor	User (General Manager, Gudang, Penjualan, Pembelian)
Tujuan	Masuk kedalam sistem sebagai pengguna (Genaral Manager, Gudang, Penjualan dan pembelian) dan dapat mengakses beberapa fasilitas pada sistem.
Skenario Utama	
Aktor	**Sistem**
1. Membuka Aplikasi.	
	2. Menampilkan halaman *form login*.
1. Mengisi form *login*, memasukkan *username* dan *password*.	
	4. Mengecek data *login* dengan data *username* dan *password* pada basis data.
	5. Menampilkan halaman menu utama untuk *user* yang bersangkutan.
Skenario Alternatif (*Username* dan *Password* Tidak Sesuai)	
	1. Mengecek data *login* dengan data *username* dan *password* pada basis data.
	2. Menampilkan pesan *username* dan *password* tidak sesuai.

2) Skenario Use Case Data Konsumen Yang Diusulkan

Tabel 4.4 Skenario Use Case Data Konsumen Yang Diusulkan

Identifikasi	
Nama Use Case	Input Data Konsumen
Aktor	User (Bagian Penjualan)
Skenario Utama	
Aktor	**Sistem**
1. Membuka *form* data konsumen	
	2. Menampilkan form data konsumen.
3. Input data konsumen	
	4. Menampilkan form data konsumen yang telah diinputkan oleh Bag. Penjualan
	5. Memvalidasi data inputan
	6. Menyimpan data konsumen
Skenario Alternatif (Input Data Pengguna Tidak Sesuai)	
	1. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	2. Menampilkan pesan bahwa data yang diinputkan belum lengkap.
3. Mengisi form data konsumen dengan lengkap.	
	4. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	5. Menyimpan data konsumen yang sudah terisi dengan persyaratan yang lengkap kedalam database.

3) Skenario Use Case Data Supplier Yang Diusulkan

Tabel 4.5 Skenario Use Case Supplier Yang Diusulkan

Identifikasi	
Nama Use Case	Input Data Supplier
Aktor	User (Bagian Pembelian)
Skenario Utama	
Aktor	**Sistem**
1. Membuka *form* data supplier.	

	2. Menampilkan form data supplier.
3. Input data supplier.	
	4. Menampilkan form supplier yang telah diinputkan.
	5. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	6. Menyimpan data supplier yang sudah terisi dengan persyaratan yang lengkap kedalam database.
Skenario Alternatif (Input Data Kategori Belum Lengkap)	
	7. Mengecek apakah formulir data yang diisikan Pembelian sudah terisi dengan lengkap.
	8. Menampilkan pesan bahwa data yang diinputkan belum lengkap.
9. Mengisi form data supplier dengan lengkap.	
	10. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	11. Menyimpan data kategori yang sudah terisi dengan persyaratan yang lengkap kedalam database.

4) Skenario Use Case Barang Yang Diusulkan

Interaksi antara actor Bagian Gudang dengan use case Form data barang yang dijelaskan dalam skenario use case sebagai berikut :

Tabel 4.6 Skenario Use Case Data Barang Yang Diusulkan

Identifikasi	
Nama Use Case	Input Data Barang
Aktor	User (Gudang)
Tujuan	Mengelola data masukan barang.
Skenario Utama	
Aktor	**Sistem**

1. Membuka *form* barang	
	2. Menampilkan form dari barang
3. Input data barang.	
.	4. Menampilkan form barang yang telah diinputkan.
	5. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	6. Menyimpan data barang yang sudah terisi dengan persyaratan yang lengkap kedalam database.
Skenario Alternatif (Input Data Kategori Belum Lengkap)	
	1. Mengecek apakah formulir data yang diisikan Pembelian sudah terisi dengan lengkap.
	2. Menampilkan pesan bahwa data yang diinputkan belum lengkap.
3. Mengisi form data barang dengan lengkap.	
	4. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	5. Menyimpan data barang yang sudah terisi dengan persyaratan yang lengkap kedalam database.

5) Skenario Use Case Data Pembelian Barang Yang Diusulkan

Interaksi antara aktor Bagian Gudang dengan use case data pembelian barang yang dijelaskan dalam skenario use case sebagai berikut :

Tabel 4.7 Skenario Use Case Data Pembelian Barang Yang Diusulkan

Identifikasi	
Nama Use Case	Input Data Pembelian Barang
Aktor	User (Bagian Pembelian)
Tujuan	Membuat daftar pembelian barang ke supplier.
Skenario Utama	
Aktor	**Sistem**

1. Membuka *form* pembelian barang	
	2. Menampilkan form dari pebelian barang.
3. Input data pembelian barang	
.	4. Menampilkan form pembelian barang yang telah diinputkan.
	5. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	6. Menyimpan form data pembelian barang yang sudah terisi dengan persyaratan yang lengkap kedalam database.
Skenario Alternatif (Input Data *Customer* Belum Lengkap)	
	1. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	2. Menampilkan pesan bahwa data yang diinputkan belum lengkap.
3. Mengisi form data pembelian barang dengan lengkap.	
	4. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	5. Menyimpan data pembelian barang yang sudah terisi dengan persyaratan yang lengkap kedalam database.

6) Skenario Use Case Transaksi Penjualan Yang Diusulkan

Interaksi antara aktor Bagian Penjualan dengan use case Transaksi Penjualan yang dijelaskan dalam skenario use case sebagai berikut :

Tabel 4.8 Skenario Use Case Transaksi Penjualan Yang Diusulkan

Identifikasi	
Nama Use Case	Transaksi Penjualan
Aktor	User (Penjualan)

Tujuan	Mengelola proses transaksi penjualan.
Skenario Utama	
Aktor	**Sistem**
1. Membuka *form* transaksi penjualan.	
	2. Menampilkan form transaksi penjualan
3. Input transaksi penjualan	
.	4. Menampilkan form transaksi penjualan yang telah diinputkan.
	5. Cetak Struk penjualan
Skenario Alternatif (Input Data Barang Masuk Belum Lengkap)	
	1. Mengecek apakah formulir data yang diisikan sudah terisi dengan lengkap.
	2. Menampilkan pesan bahwa data yang diinputkan belum lengkap.
3. Mengisi form transaksi penjualan dengan lengkap.	
	4. Mengecek apakah formdata yang diisikan Pembelian sudah terisi dengan lengkap.
	5. Cetak struk penjualan

7) Skenario Use Case Laporan Yang Diusulkan

Interaksi antara aktor General manager dengan use case Laporan yang dijelaskan dalam skenario use case sebagai berikut :

Tabel 4.9 Skenario Use Case Laporan Yang Diusulkan

Identifikasi	
Nama Use Case	Laporan
Aktor	User (General Manager)

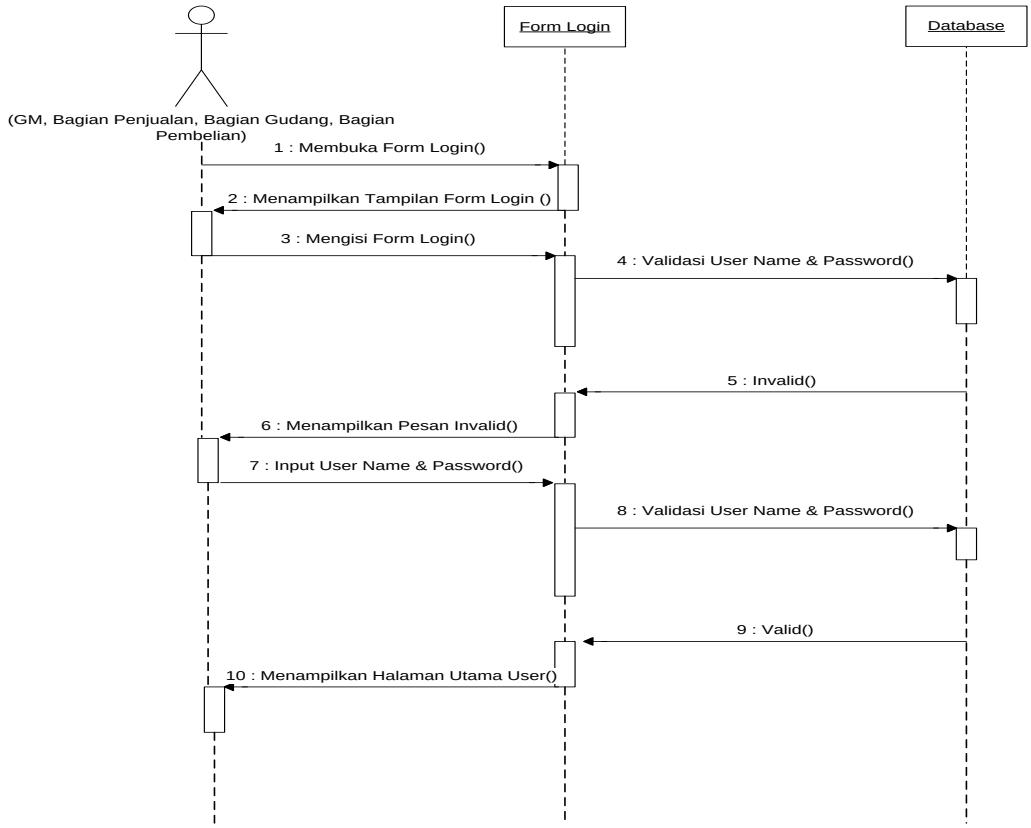
Tujuan	Mengakses Semua laporan.
Skenario Utama	
Aktor	**Sistem**
1. Membuka *form* laporan	
	2. Menampilkan form laporan
3. Memilih cetak laporan berdasarkan periode dan berdasarkan data barang.	
.	4. Menampilkan laporan yang akan dicetak

4.1.4. Sequence Diagram

Adi Nugroho (2005:92) sequence diagram adalah interaksi diagram yang memperlihatkan event-event yang berurutan sepanjang berjalannya waktu. Selain itu Sequence Diagram merupakan salah satu diagram interaksi yang menjelaskan bagaimana suatu operasi itu dilakukan. Masing-masing sequence diagram akan menggambarkan aliran-aliran pada suatu use case.

1. Sequence Diagram Login

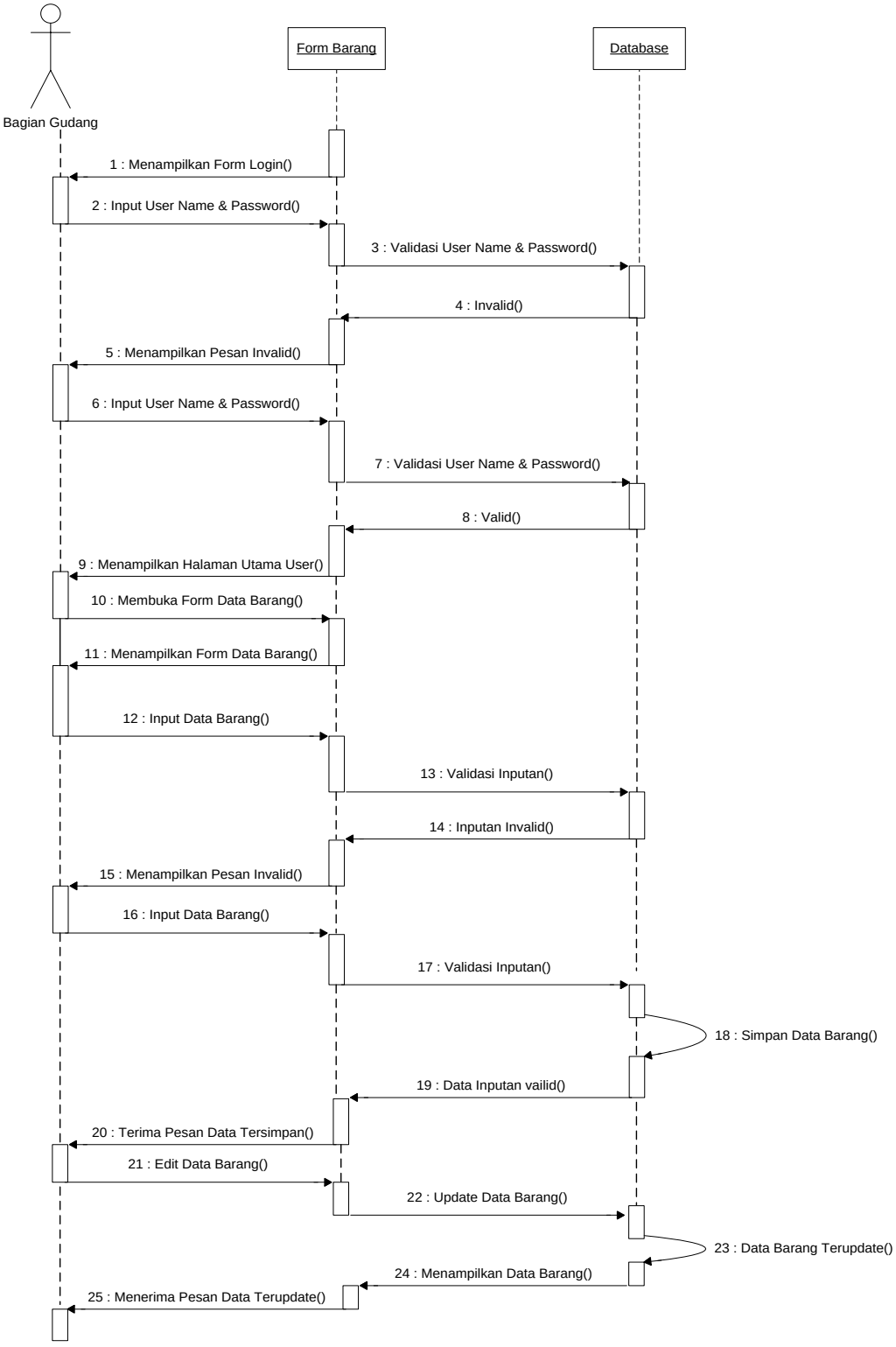
Berikut merupakan sequence diagram untuk login user (general manager, bagian pembelian, bagian penjualan) :



Gambar 4.8 Sequence Diagram Login

2. Sequence Diagram Data Barang

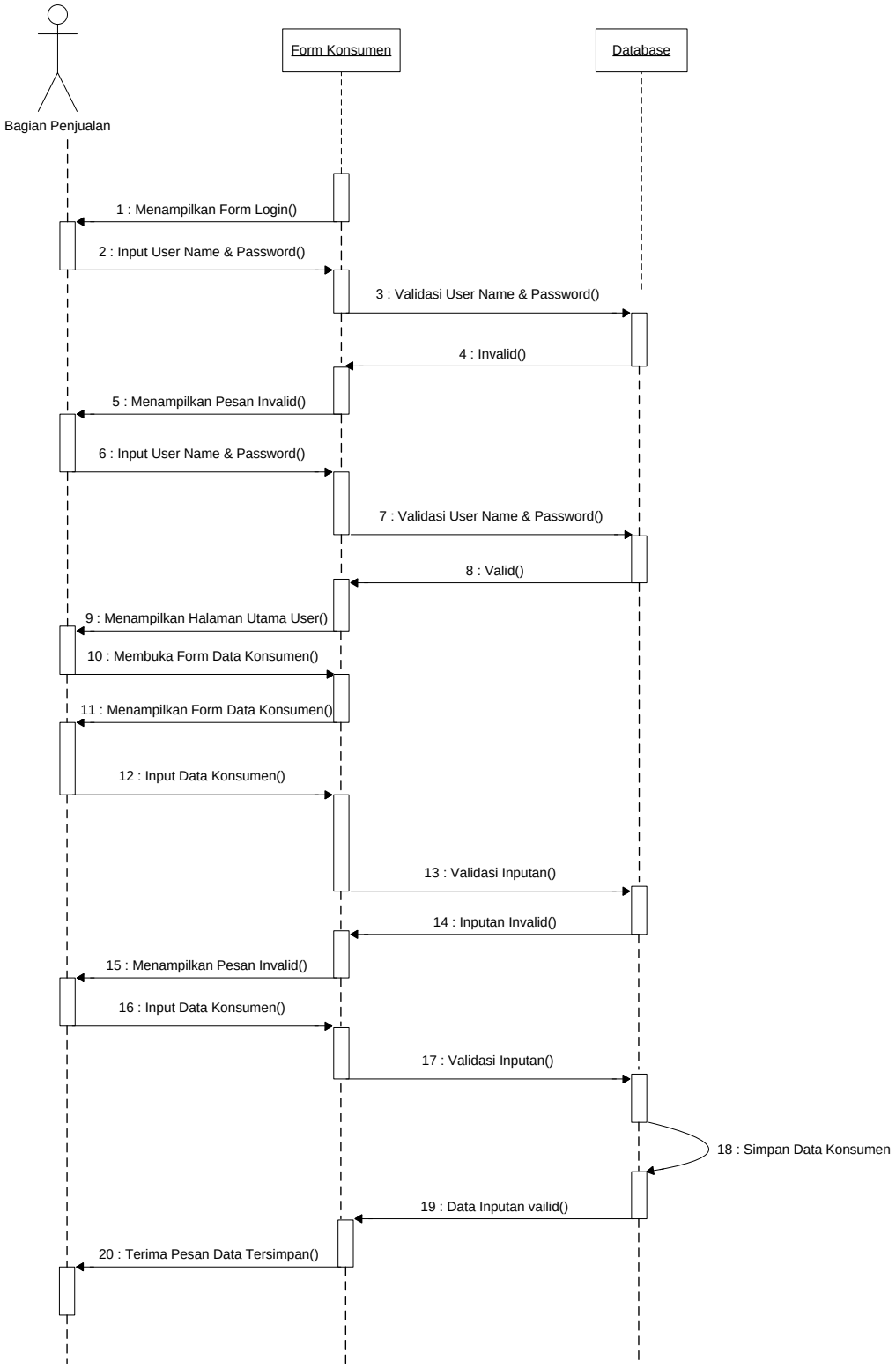
Berikut merupakan sequence diagram untuk Data Barang, yaitu :



Gambar 4.9 Sequence Diagram Data Barang

3. Sequence Diagram Data Konsumen

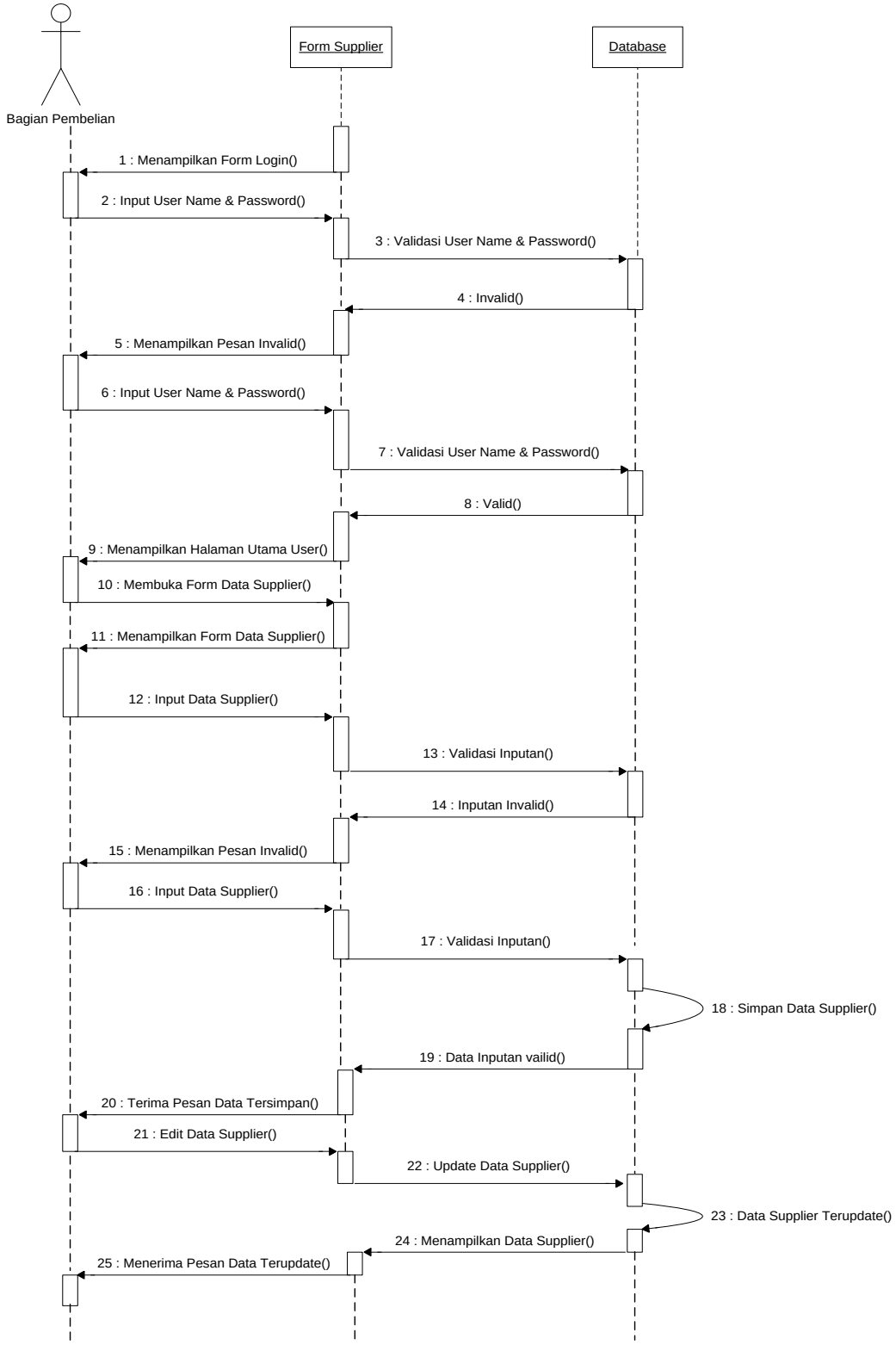
Berikut merupakan sequence diagram untuk Data Konsumen, yaitu :



Gambar 4.10 Sequence Diagram Data Konsumen

4. Sequence Diagram *Supplier*

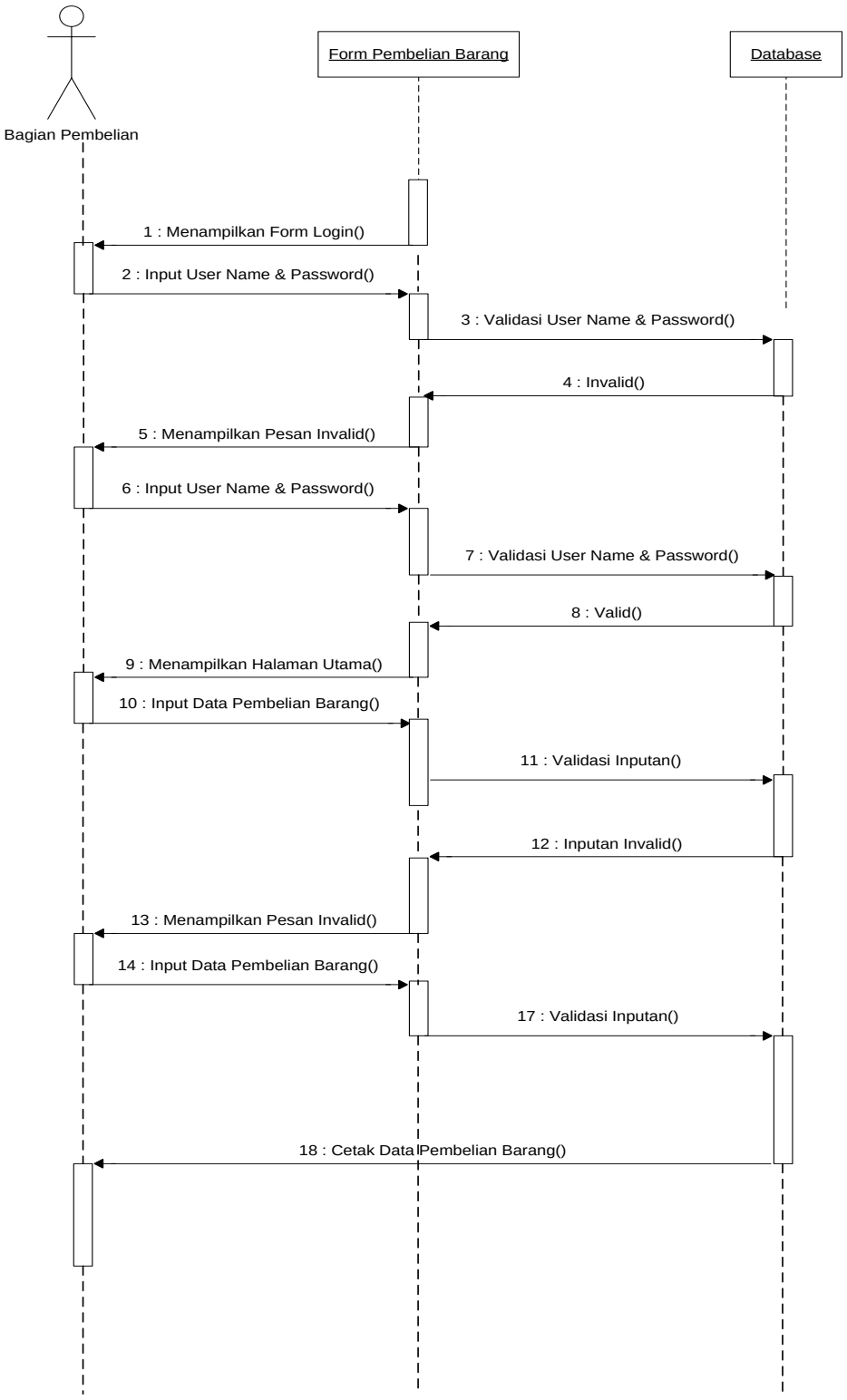
Berikut merupakan sequence diagram untuk Data Konsumen, yaitu :



Gambar 4.11 Sequence Diagram *Supplier*

5. Sequence Diagram Data Pembelian Barang

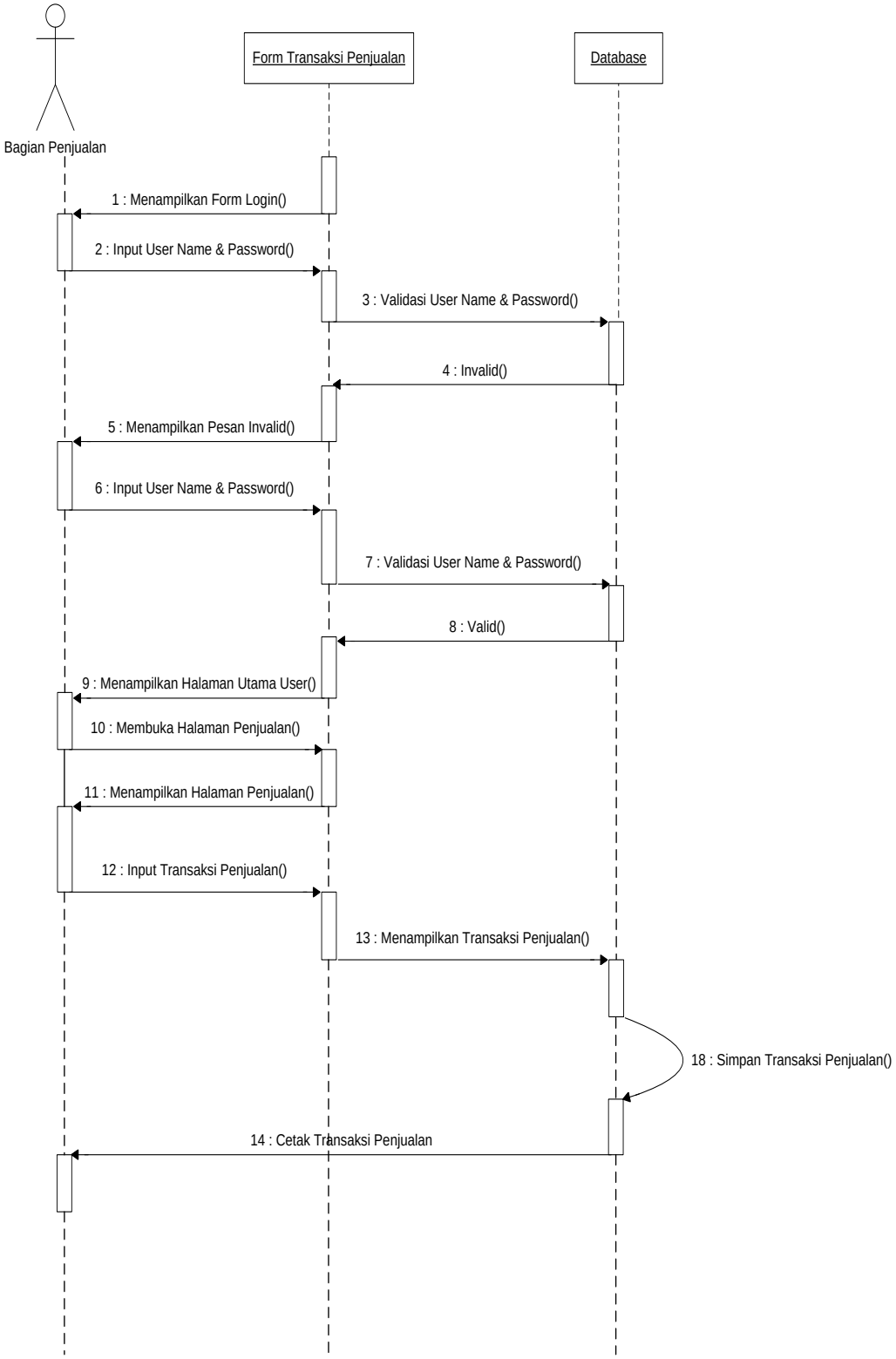
Berikut merupakan sequence diagram untuk Data Pembelian Barang, yaitu :



Gambar 4.12 Sequence Diagram Data Pembelian Barang

6. Sequence Diagram Data Transaksi Penjualan

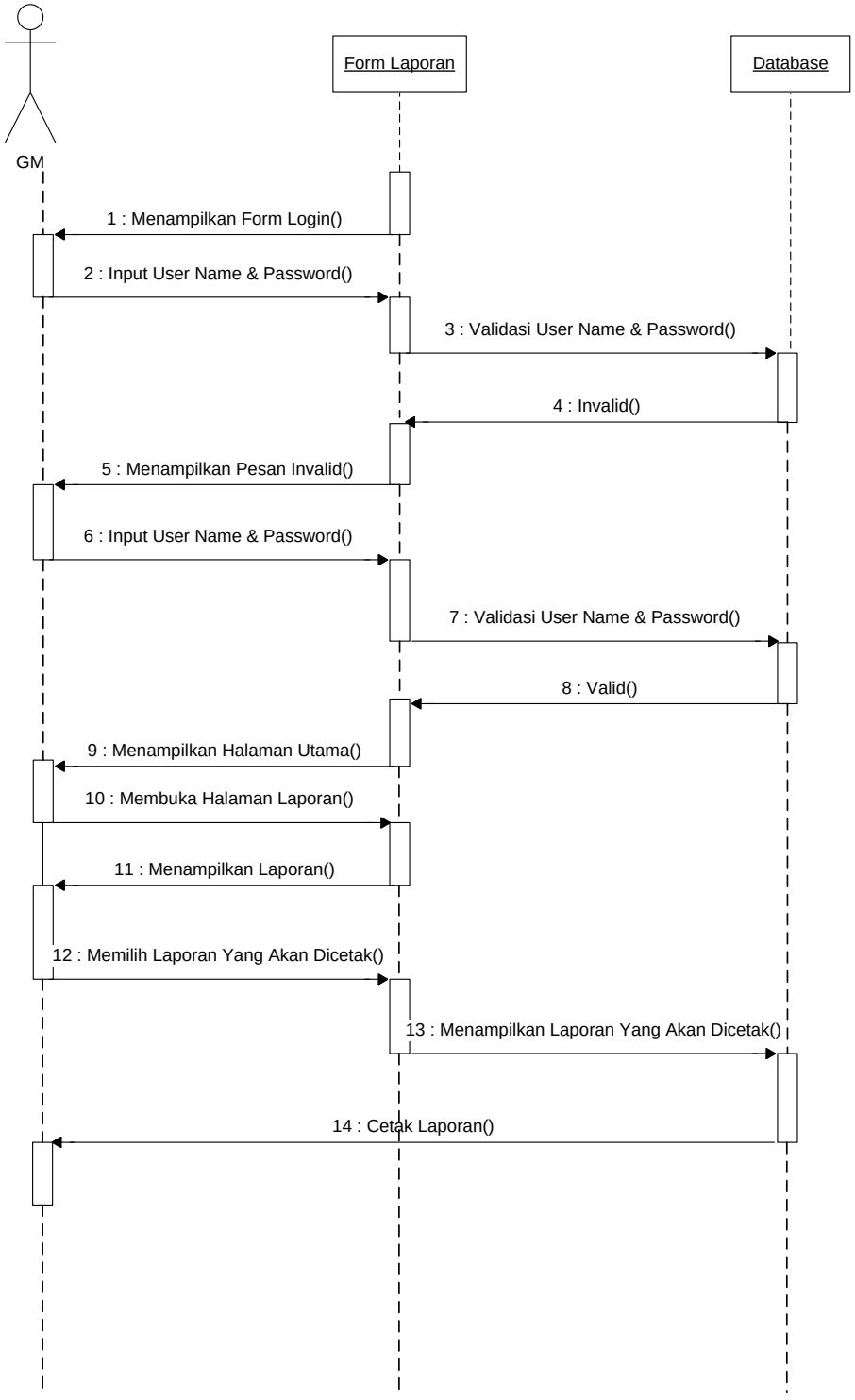
Berikut merupakan sequence diagram untuk Data Transaksi Penjualan, yaitu :



Gambar 4.13 Sequence Diagram Data Transaksi Penjualan

7. Sequence Diagram Laporan

Berikut merupakan sequence diagram untuk Data Laporan, yaitu :



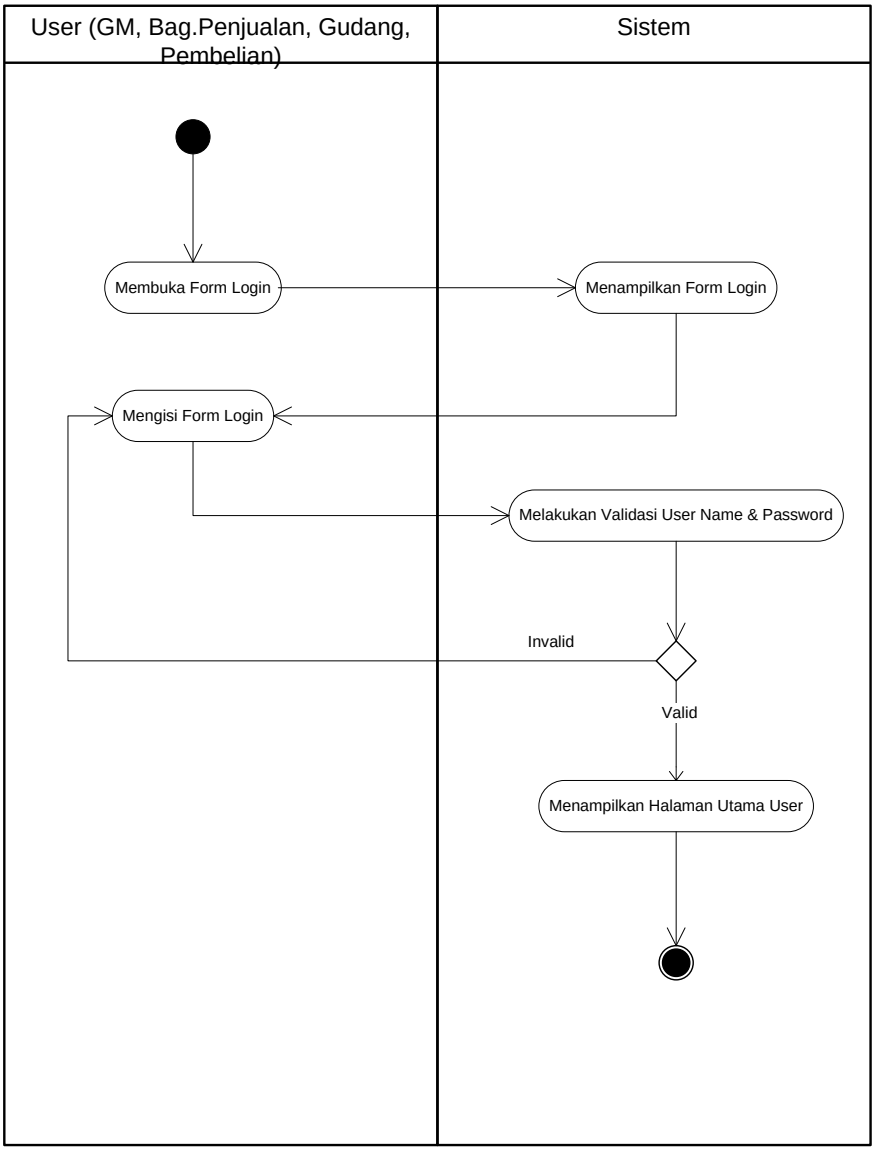
Gambar 4.14 Sequence Diagram Laporan

4.1.4.1.Diagram Activity Yang Diusulkan

Adi Nugroho (2005:61) Activity Diagram adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu use case, memperlihatkan aliran kendali dari suatu aktifitas ke aktifitas lainnya.

1. Activity Diagram Login Yang Diusulkan

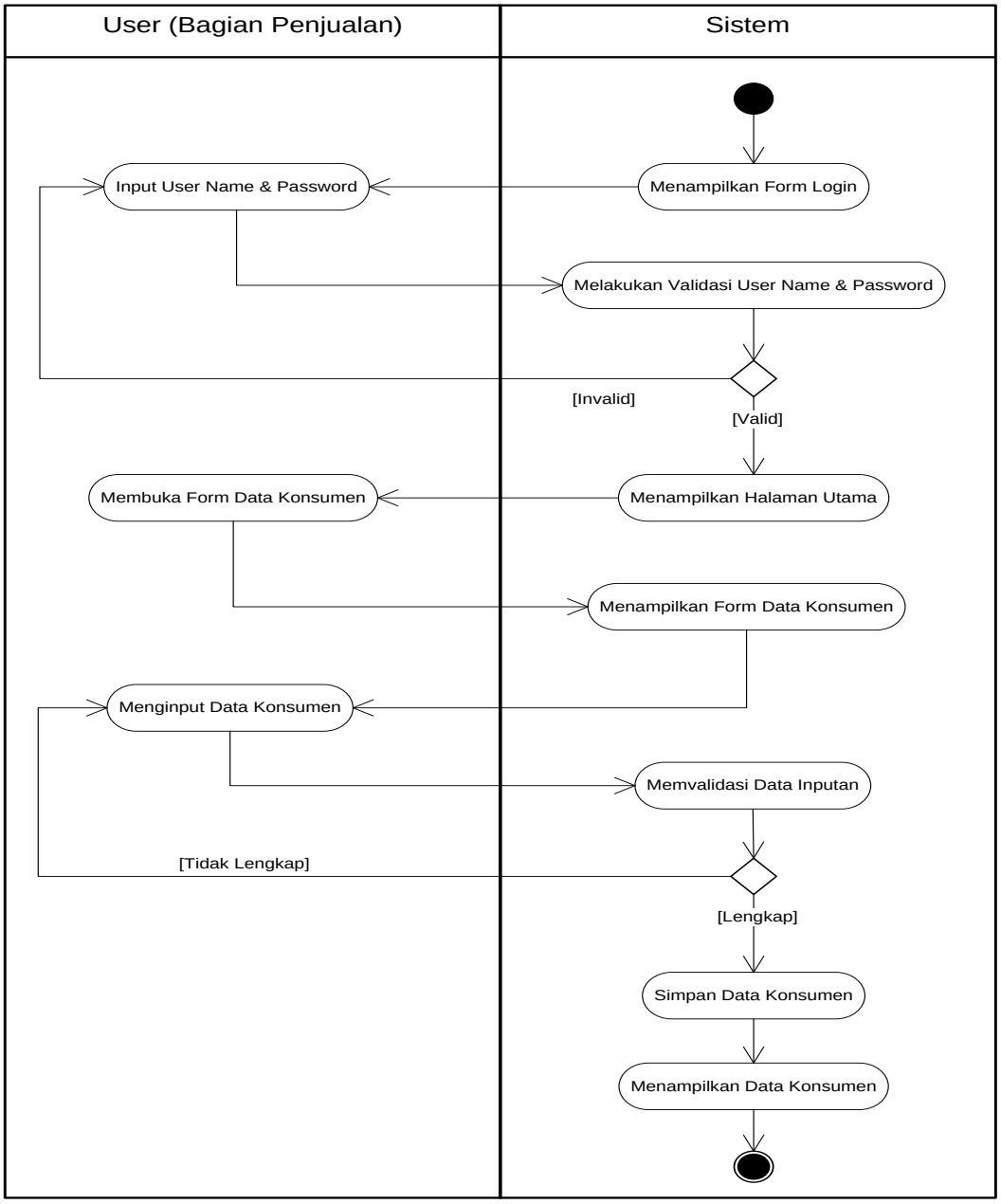
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram login yang diusulkan adalah sebagai berikut:



Gambar 4.6 Activity Diagram Login Yang Diusulkan

2. Activity Diagram Data Konsumen Yang Diusulkan

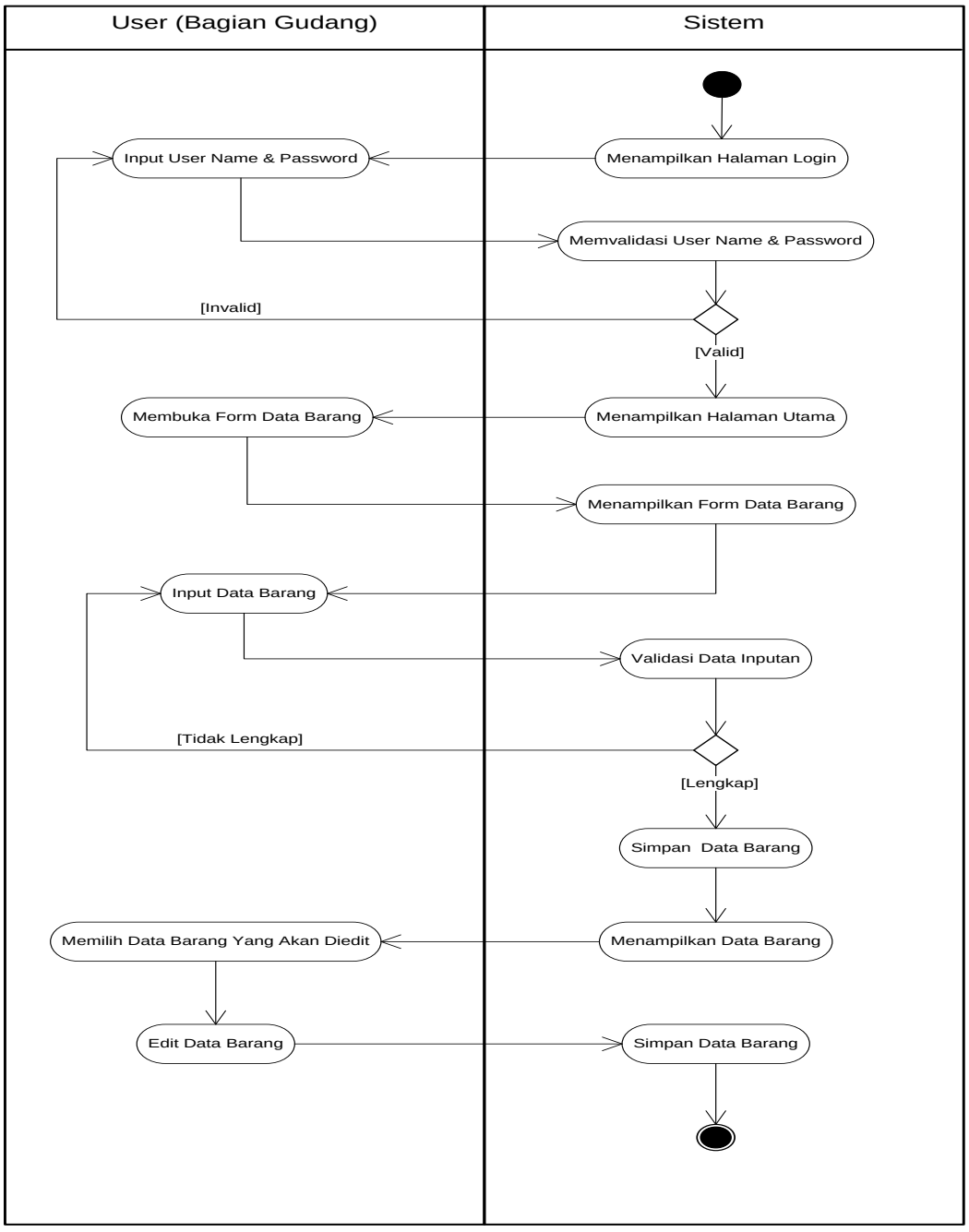
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram data pengguna yang diusulkan adalah sebagai berikut:



Gambar 4.2 Activity Diagram Data Konsumen Yang Diusulkan

3. Activity Diagram Data Barang Yang Diusulkan

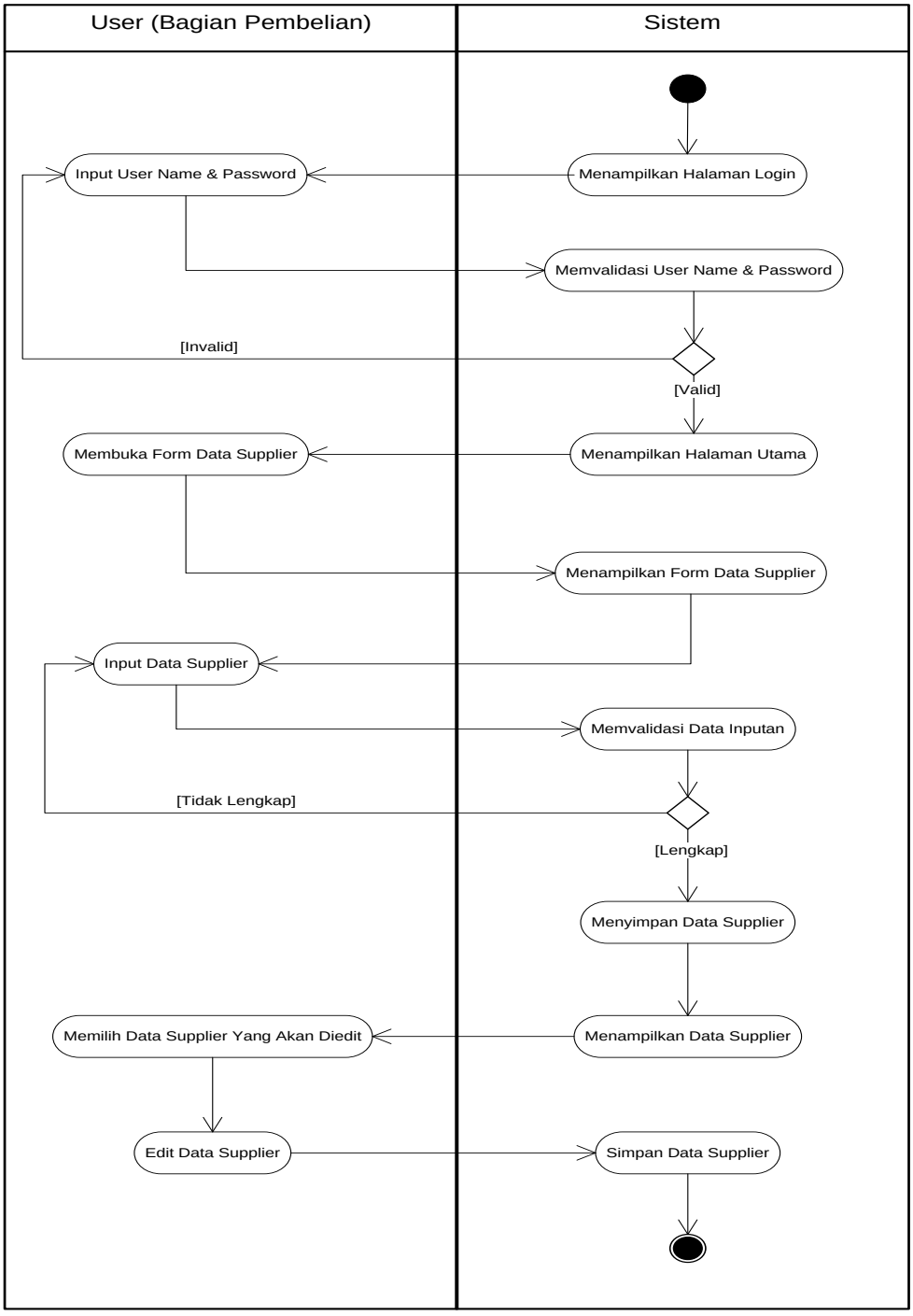
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram data barang yang diusulkan adalah sebagai berikut :



Gambar 4.3 Activity Diagram Data Barang Yang Diusulkan

4. Activity Diagram *Supplier* Barang Yang Diusulkan

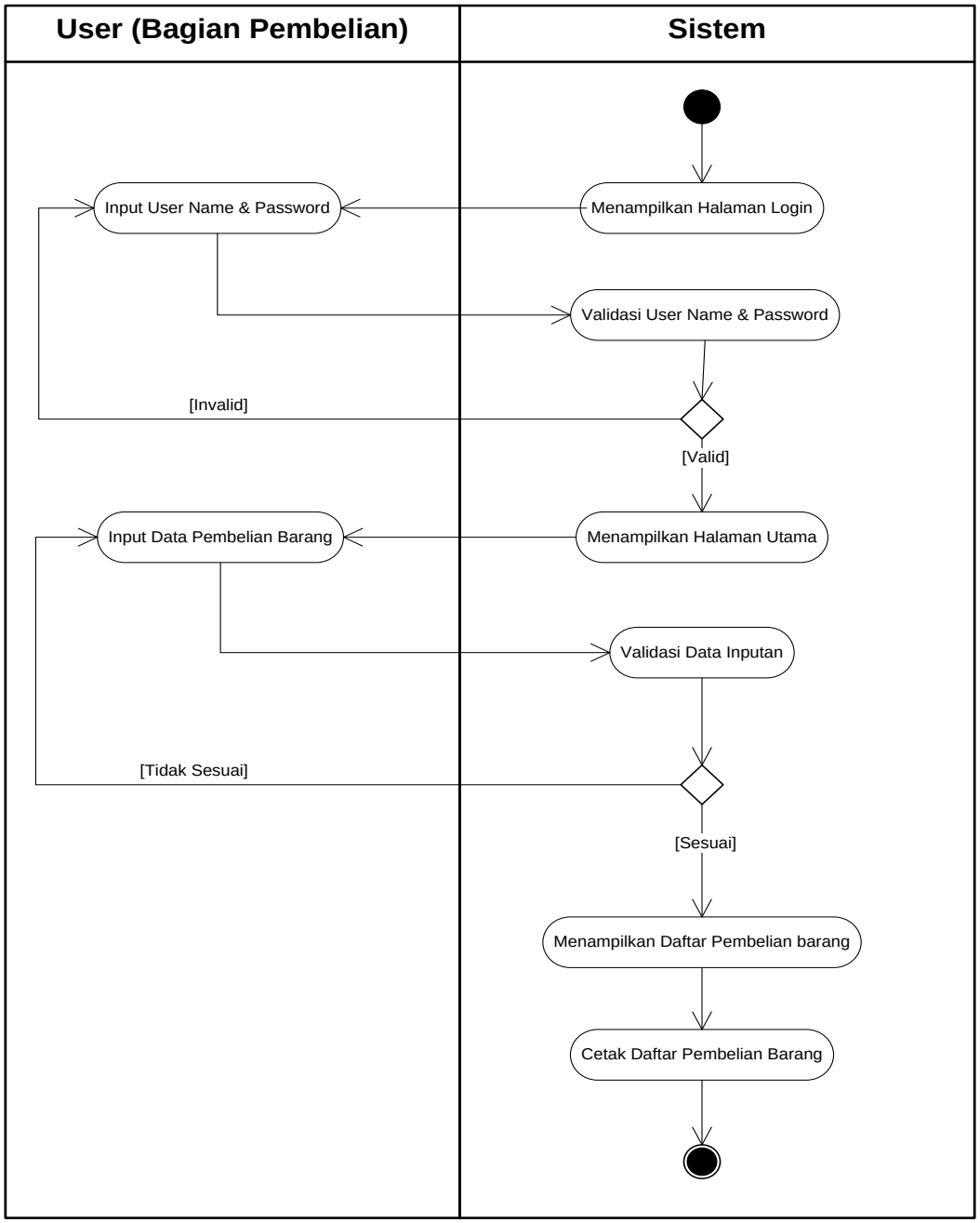
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram permintaan barang yang diusulkan adalah sebagai berikut :



Gambar 4.4 Activity Diagram *Supplier* Yang Diusulkan

5. Activity Diagram Data Pembelian Barang Yang Diusulkan

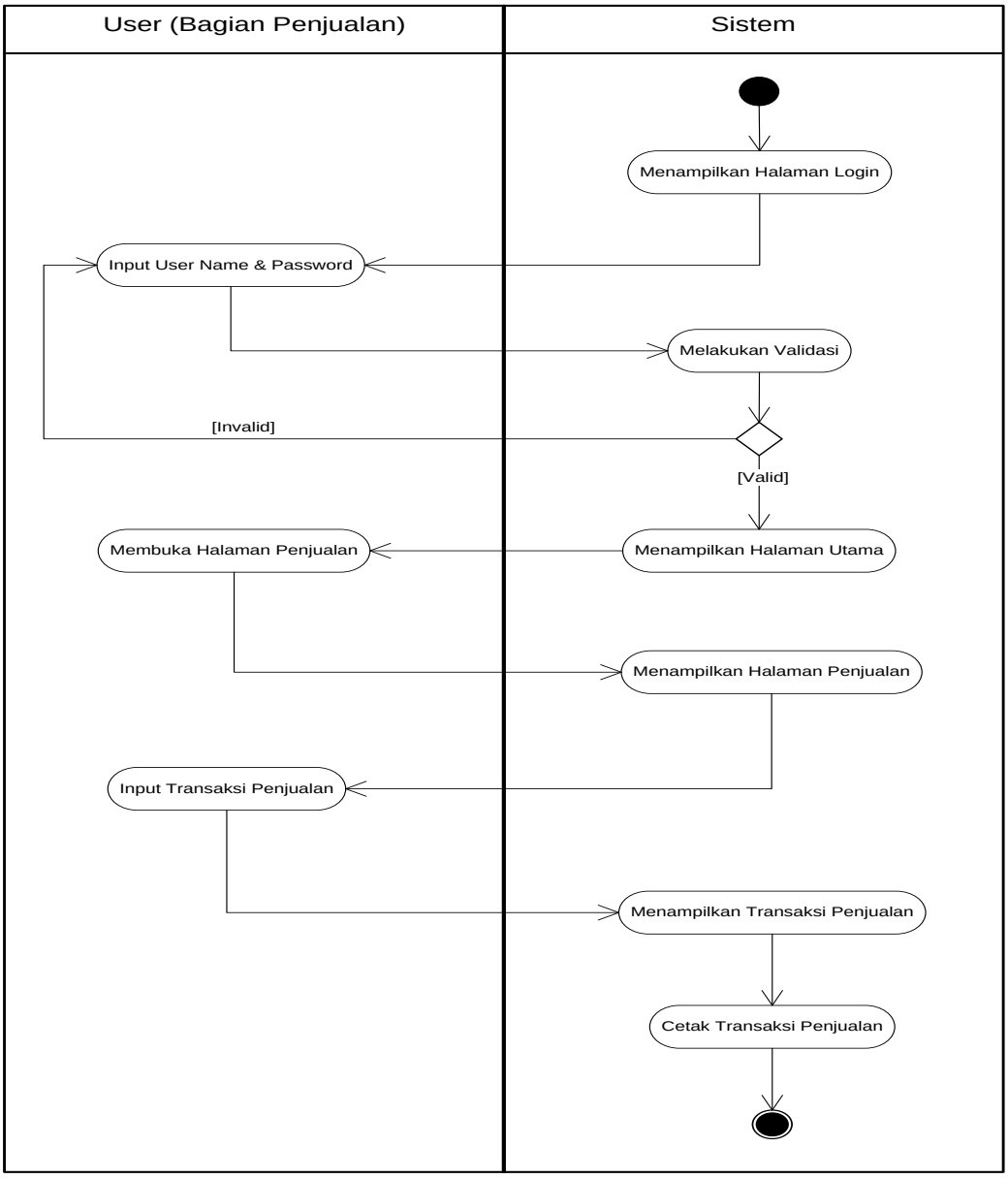
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram pengajuan barang yang diusulkan adalah sebagai berikut :



Gambar 4.15 Activity Diagram Pembelian Barang Yang Diusulkan

6. Activity Diagram Transaksi Penjualan Yang Diususlkan

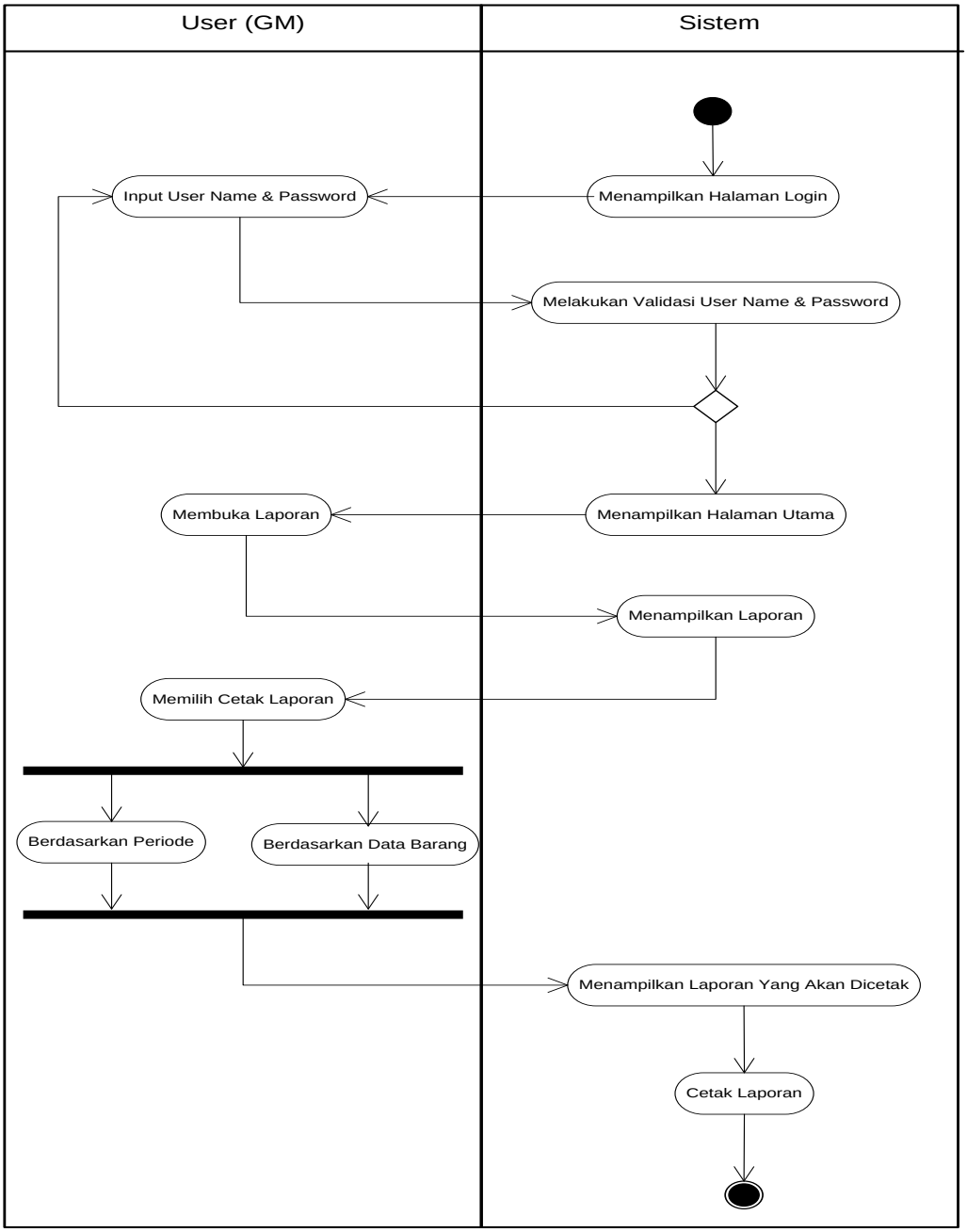
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram pengajuan barang yang diusulkan adalah sebagai berikut :



Gambar 4.6 Activity Diagram Transaksi Penjualan Yang Diusulkan

7. Activity Diagram Laporan Yang Diusulkan

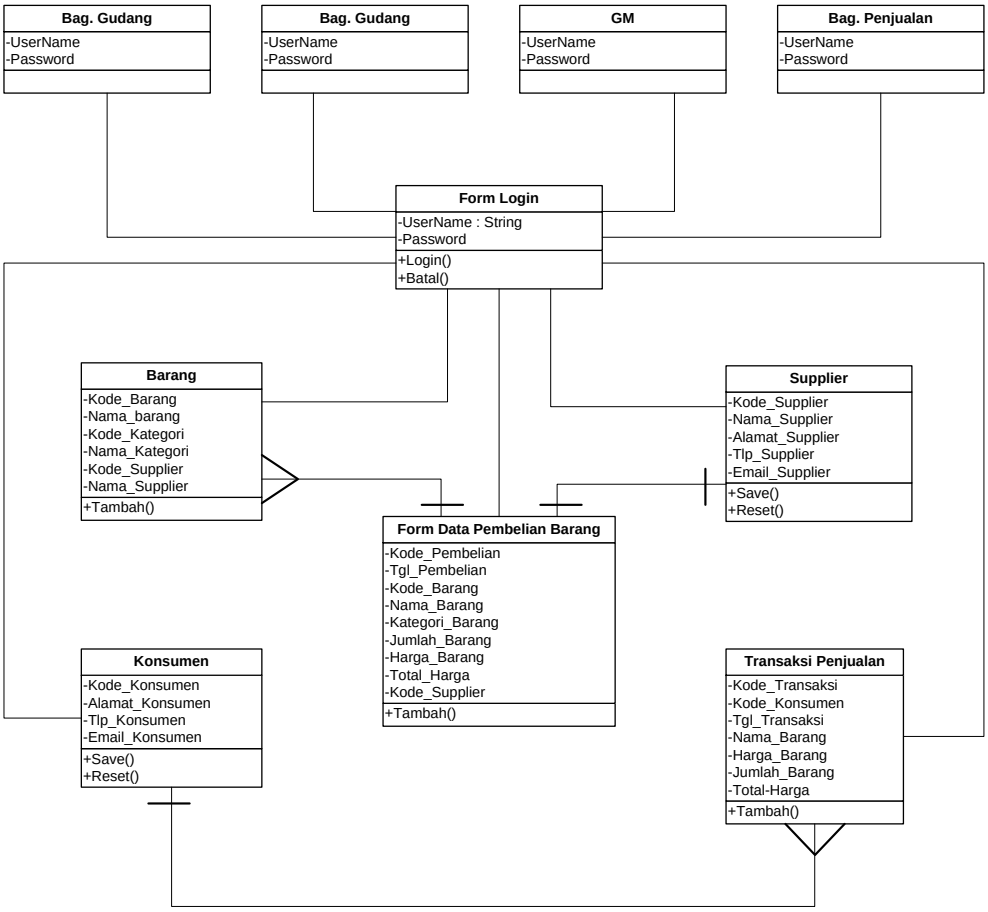
Aktivitas Diagram adalah suatu gambaran tentang aktifitas-aktifitas yang terjadi dan terkait dalam suatu proses tunggal. Adapun aktifitas diagram pengajuan barang yang diusulkan adalah sebagai berikut :



Gambar 4.7 Activity Diagram Laporan Yang Diusulkan

4.1.4.2 Class Diagram

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan *object* beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dll. Berikut adalah *Class Diagram* yang menggambarkan skema yang terdapat dalam *database*:

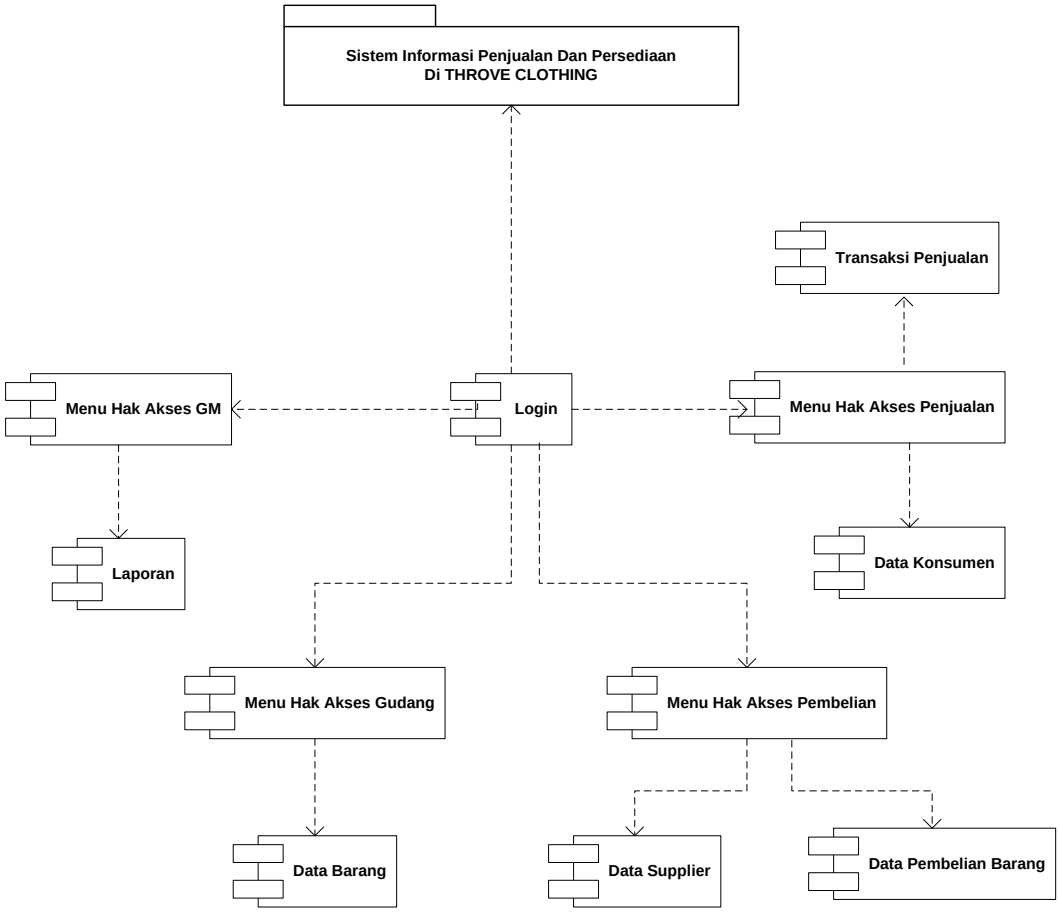


Gambar 4.15 Class Diagram

4.1.4.3 Component Diagram

Komponen perangkat lunak adalah bagian fisik dari sebuah sistem yang menetap di komputer. komponen merupakan implementasi software dari sebuah class. Komponen bisa berupa tabel, *file* data, *file* exe, file DLL, dokumen dan lain-

lain. Berikut adalah *component diagram* yang menggambarkan struktur proses yang terdapat dalam program aplikasi sistem informasi Penjualan dan Persediaan di Throve Clothing.

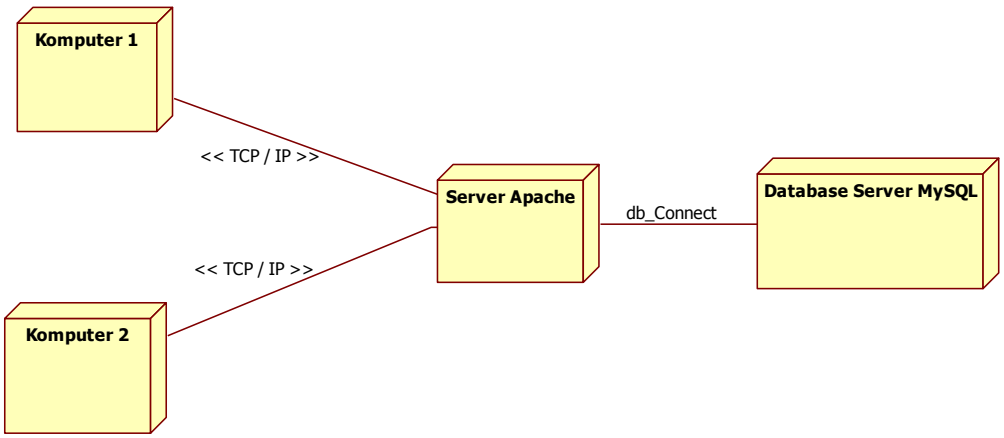


Gambar 4.16 Component Diagram

4.1.4.4. Deployment Diagram

Deployment diagram menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware* yang digunakan untuk mengimplementasikan sebuah sistem dan keterhubungan antara komponen-komponen *hardware* tersebut. *Deployment diagram* dapat digunakan pada bagian-bagian awal proses perancangan sistem

untuk mendokumentasikan arsitektur fisik sebuah sistem. Berikut Deployment diagram yang menggambarkan susunan fisik dan perangkat lunak dalam sistem:



Gambar 4.17 Deployment Diagram

4.1.4.5. Kodifikasi

Pengkodean adalah pembuatan kode untuk tujuan mengklasifikasikan data, memasukkan data ke komputer dan mengambil berbagai informasi yang dibutuhkan. Pengkodean digunakan untuk menjabarkan item - item data yang bersifat unik. Dalam perancangan, penulis melakukan pengkodean sebagai berikut:

1. **Kode Barang**

Bentuk Umum : XXXXYYY

1 **2**

Keterangan :

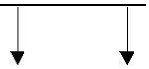
1 : Menunjukkan 4 huruf inisial dari nama jenis barang yang ada

2 : Menunjukkan nomor urut barang

Contoh : TJMN010

2. Kode Kategori

Bentuk Umum : XXXXXXYY



1 2

Keterangan :

1 : Menunjukkan 6 huruf dari nama throve

2 : Menunjukan 2 angka dari nomer keterangan barang

Contoh : THROVE01, THROVE02

3. Kode *Supplier*

Bentuk Umum : XXYY



1 2

Keterangan :

1 : Menunjukkan 2 huruf dari singkatan *supplier*

2 : Menunjukan 2 angkat dari nomor urut *supplier*

Contoh : SP01, SP12

4.2. Perancangan Antar Muka

Perancangan antar muka sangat penting dalam pembuatan suatu program, karena merupakan dasar untuk membuat antar muka yang dapat memberikan kemudahan dan tidak membingungkan bagi user dalam melakukan aktivitasnya. Berikut adalah rancangan inputan yang dibuat:

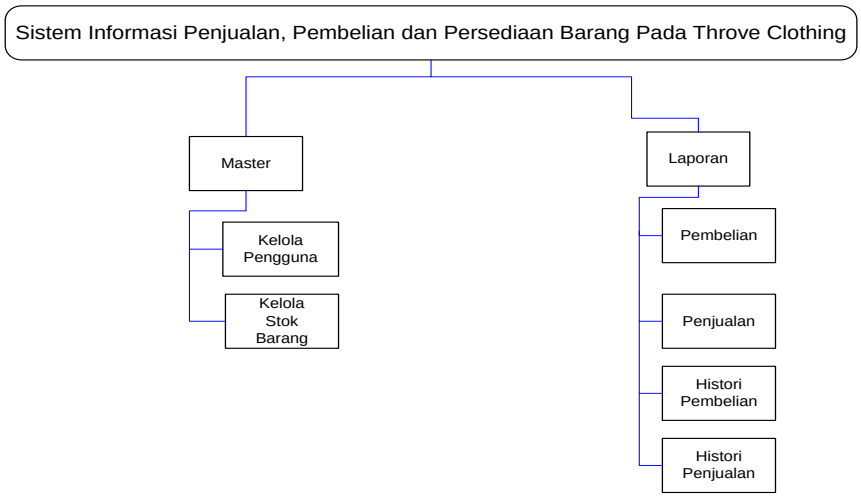
4.2.1. Struktur Menu

Perancangan menu di gunakan untuk memudahkan dalam penelusuran

program yang di buat. Struktur menu di bagi menjadi 3 yaitu struktur menu general manager, bagian pembelian dan bagian penjualan, berikut adalah gambar perancangannya :

1. **Struktur Menu General Manager**

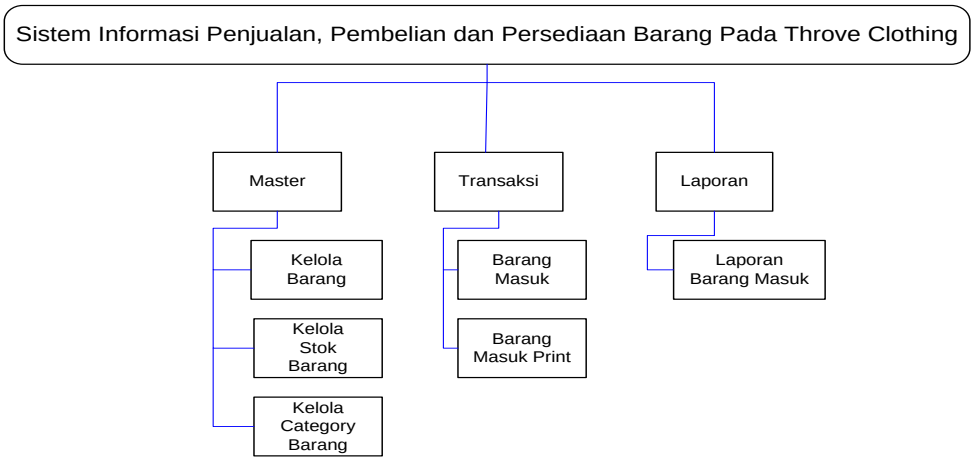
Berikut adalah tampilan struktur menu untuk admin :



Gambar 4.18 Struktur Menu General Manager

2. **Struktur Menu Bagian Pembelian**

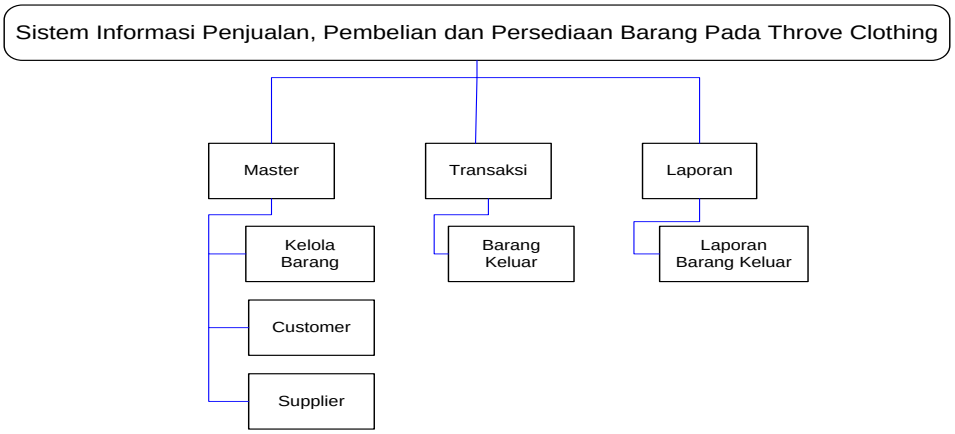
Berikut adalah rancangan tampilan untu pimpinan.



Gambar 4.19 Struktur Menu Bagian Pembelian

3. **Struktur Menu Bagian Penjualan**

Berikut adalah rancangan tampilan untu pimpinan.



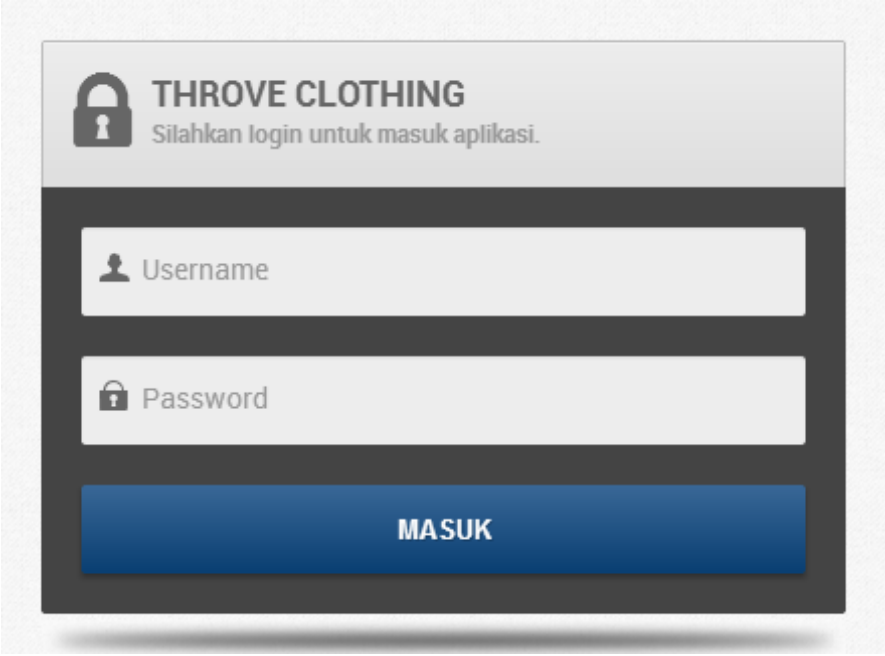
Gambar 4.20 Struktur Menu Bagian Penjualan

4.2.2. Perancangan Input

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

1. Rancangan Tampilan Login

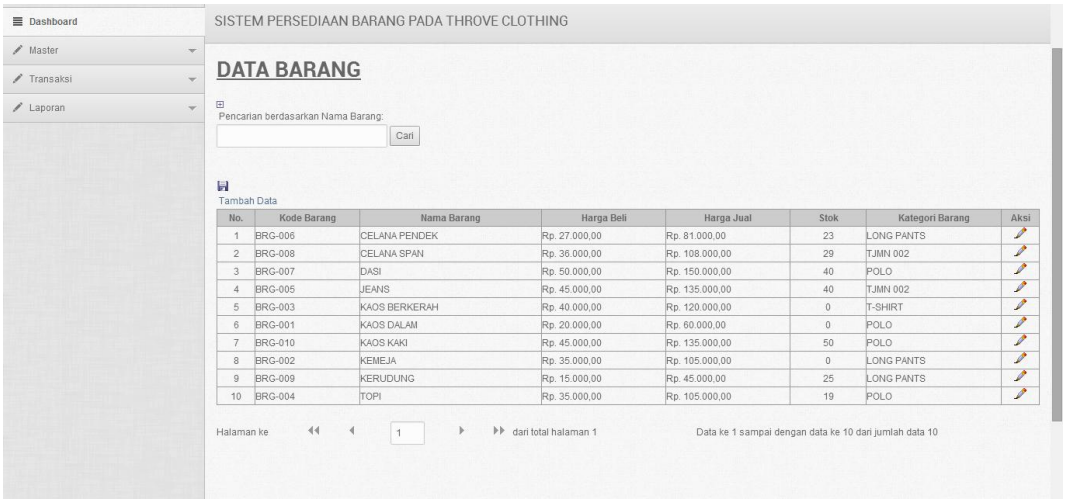
Rancangan tampilan ini dipergunakan bagi pengguna yang berkepentingan untuk menggunakan program aplikasi. Bagi yang berkepentingan menggunakan program aplikasi ini baik admin maupun pemimpin maka terlebih dahulu harus memasukkan username dan password.



Gambar 4.21 Rancangan Tampilan Login Pengguna

2. Rancangan Tampilan Data Barang

Form yang digunakan untuk melakukan manipulasi data barang di Throve Clothing.



Gambar 4.22 Rancangan Tampilan Data Barang

3. Rancangan Tampilan Form Kategori

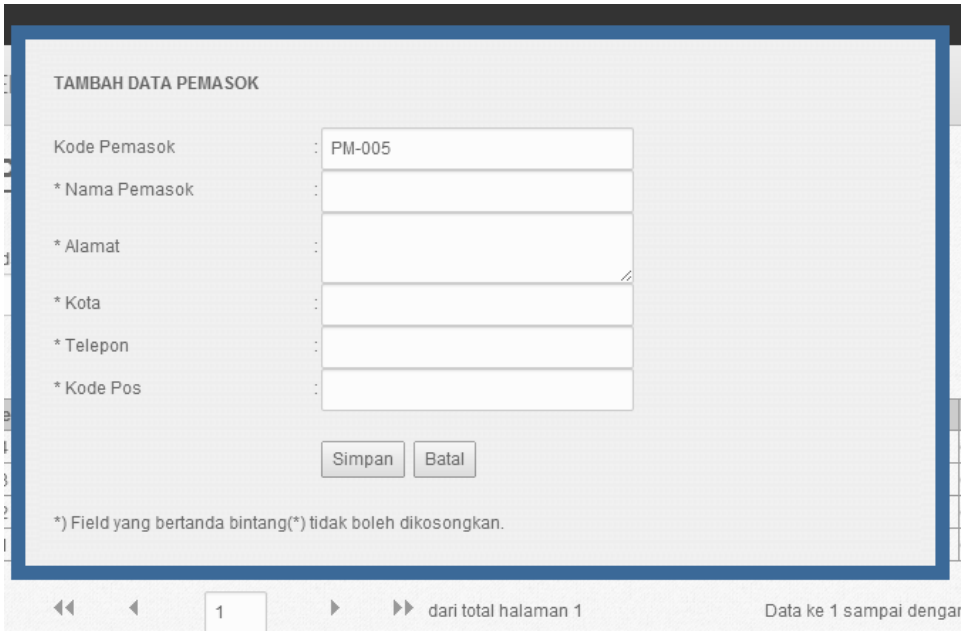
Form yang digunakan untuk melihat dan memanipulasi data kategori barang muali dari kode barang, nama barang dan lainnya pada Throve Clothing.



Gambar 4.23 Rancangan Tampilan Form Kategori

4. Rancangan Tampilan Form *Supplier*

Form yang digunakan untuk melihat dan memanipulasi data *supplier* :



Gambar 4.24 Rancangan Tampilan Form *Supplier*

5. Rancangan Tampilan Form *Customer*

Form yang digunakan untuk melihat dan memanipulasi data *customer* :

- 

Gambar 4.25 Rancangan Tampilan Form *Customer*

6. Rancangan Tampilan Barang Masuk

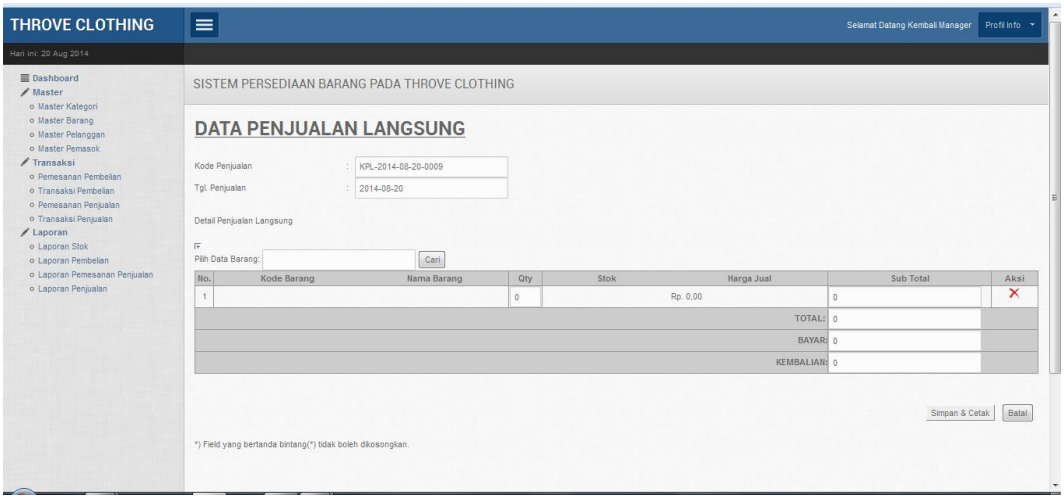
Form yang digunakan untuk melihat dan memanipulasi barang masuk :



Gambar 4.26 Rancangan Tampilan Form Barang Masuk

7. Rancangan Tampilan Penjualan

Form yang digunakan untuk melihat dan memanipulasi barang masuk :



Gambar 4.27 Rancangan Tampilan Form Barang Keluar

8. Rancangan Tampilan Laporan

Form yang digunakan untuk melihat laporan baik berupa penjualan maupun data pembelian



LAPORAN DATA SELURUH STOK

No.	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Stok	Kategori	Pemasok
1	BRG-001	KAOS DALAM	Rp. 20.000,00	Rp. 60.000,00	0	POLO	PT. Anugrah Jaya Indah
2	BRG-002	KEMEJA	Rp. 35.000,00	Rp. 105.000,00	0	LONG PANTS	PT. Eka Prima Abadi Setia
3	BRG-003	KAOS BERKERAH	Rp. 40.000,00	Rp. 120.000,00	0	T-SHIRT	PT. Anugrah Jaya Indah
4	BRG-004	TOPI	Rp. 35.000,00	Rp. 105.000,00	19	POLO	PT. Anugrah Jaya Indah
5	BRG-005	JEANS	Rp. 45.000,00	Rp. 135.000,00	40	TJMN 002	Cv. Chloting BDG
6	BRG-006	CELANA PENDEK	Rp. 27.000,00	Rp. 81.000,00	23	LONG PANTS	PT. Kurnia Jaya
7	BRG-007	DASI	Rp. 50.000,00	Rp. 150.000,00	40	POLO	PT. Eka Prima Abadi Setia
8	BRG-008	CELANA SPAN	Rp. 36.000,00	Rp. 108.000,00	29	TJMN 002	PT. Anugrah Jaya Indah
9	BRG-009	KERUDUNG	Rp. 15.000,00	Rp. 45.000,00	25	LONG PANTS	PT. Eka Prima Abadi Setia
10	BRG-010	KAOS KAKI	Rp. 45.000,00	Rp. 135.000,00	50	POLO	PT. Kurnia Jaya

Bandung, ...
Kepala Bagian Gudang Throve Clothing

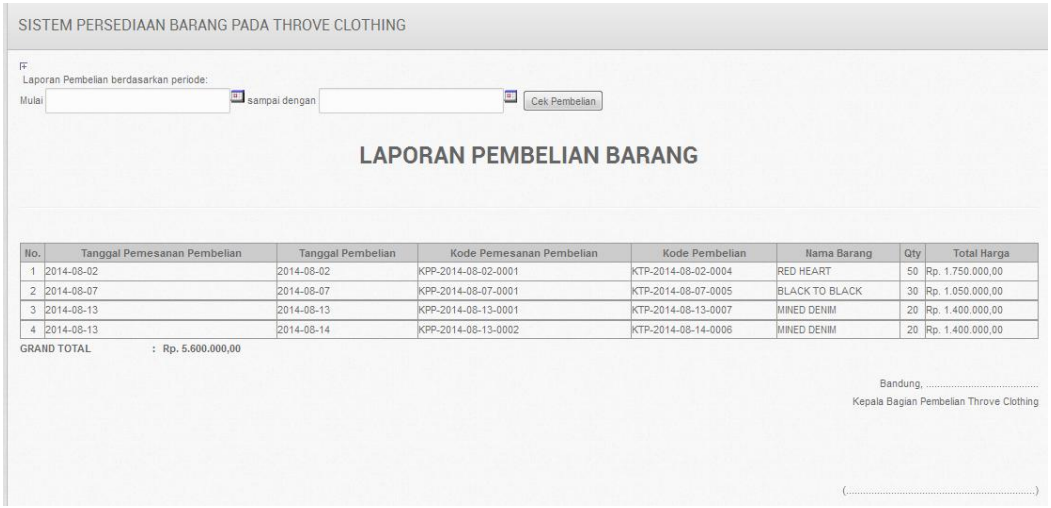
Gambar 4.28 Rancangan Tampilan Form Laporan

4.2.3. Perancangan Output

Tampilan *output* pada rancangan ini berupa kumpulan data-data yang telah dimasukan melalui inputan data menjadi laporan. Laporan yang ada adalah laporan data barang produksi, laporan pengajuan barang serta laporan data transaksi atau penjualan. Dengan barang berupa hasil print dari Ms Word. Tujuan *output* data ini adalah untuk dijadikan arsip oleh admin atau sebagai data dalam bentuk cetak oleh *user*. Berikut ini adalah tampilan *form*at laporannya yaitu sebagai berikut :

a. Laporan Data pembelian

Laporan Data Barang Masuk berisi tentang informasi data barang, mulai dari kode barang, nama barang, hingga jumlah barang yang tersedia, yang diinputkan oleh bagian pembelian berdasarkan informasi dari barang yang tersedia dan barang yang masuk dari *supplier*.


SISTEM PERSEDIAAN BARANG PADA THROVE CLOTHING

Laporan Pembelian berdasarkan periode:

Mulai sampai dengan Cek Pembelian

LAPORAN PEMBELIAN BARANG

No.	Tanggal Pemesanan Pembelian	Tanggal Pembelian	Kode Pemesanan Pembelian	Kode Pembelian	Nama Barang	Qty	Total Harga
1	2014-08-02	2014-08-02	KPP-2014-08-02-0001	KTP-2014-08-02-0004	RED HEART	50	Rp. 1.750.000,00
2	2014-08-07	2014-08-07	KPP-2014-08-07-0001	KTP-2014-08-07-0005	BLACK TO BLACK	30	Rp. 1.050.000,00
3	2014-08-13	2014-08-13	KPP-2014-08-13-0001	KTP-2014-08-13-0007	MINED DENIM	20	Rp. 1.400.000,00
4	2014-08-13	2014-08-14	KPP-2014-08-13-0002	KTP-2014-08-14-0006	MINED DENIM	20	Rp. 1.400.000,00

GRAND TOTAL : Rp. 5.600.000,00

Bandung, ...

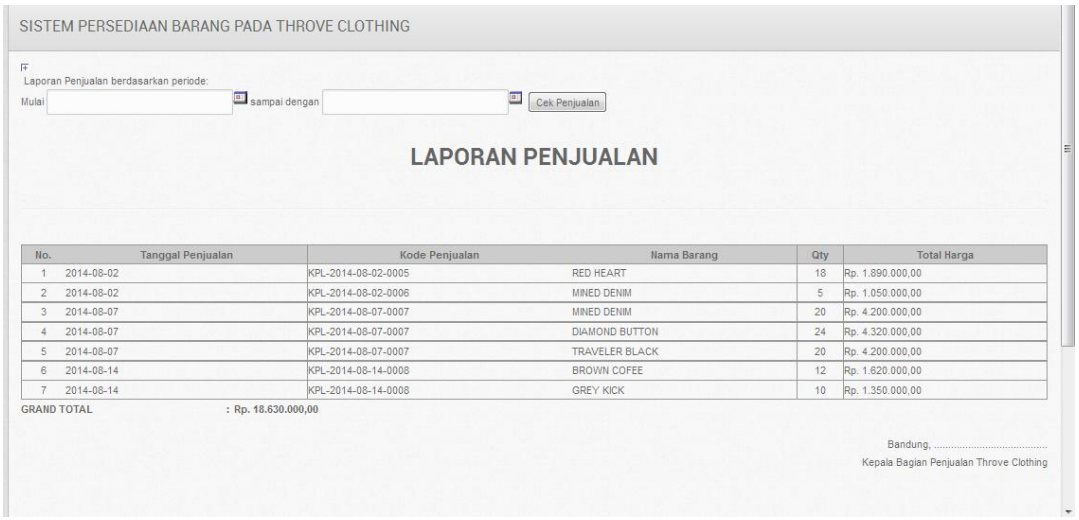
Kepala Bagian Pembelian Throve Clothing

(.....)

Gambar 4.29 Laporan Data Pembelian

b. Laporan Data Barang Penjualan

Laporan Data Barang berisi tentang informasi data barang, mulai dari kode barang, nama barang, hingga jumlah barang yang tersedia, yang diinputkan oleh bagian pembelian berdasarkan informasi dari barang yang tersedia dan barang yang masuk dari *supplier*.


SISTEM PERSEDIAAN BARANG PADA THROVE CLOTHING

Laporan Penjualan berdasarkan periode:

Mulai sampai dengan Cek Penjualan

LAPORAN PENJUALAN

No.	Tanggal Penjualan	Kode Penjualan	Nama Barang	Qty	Total Harga
1	2014-08-02	KPL-2014-08-02-0005	RED HEART	18	Rp. 1.890.000,00
2	2014-08-02	KPL-2014-08-02-0006	MINED DENIM	5	Rp. 1.050.000,00
3	2014-08-07	KPL-2014-08-07-0007	MINED DENIM	20	Rp. 4.200.000,00
4	2014-08-07	KPL-2014-08-07-0007	DIAMOND BUTTON	24	Rp. 4.320.000,00
5	2014-08-07	KPL-2014-08-07-0007	TRAVELER BLACK	20	Rp. 4.200.000,00
6	2014-08-14	KPL-2014-08-14-0008	BROWN COFEE	12	Rp. 1.620.000,00
7	2014-08-14	KPL-2014-08-14-0008	GREY KICK	10	Rp. 1.350.000,00

GRAND TOTAL : Rp. 18.630.000,00

Bandung, ...

Kepala Bagian Penjualan Throve Clothing

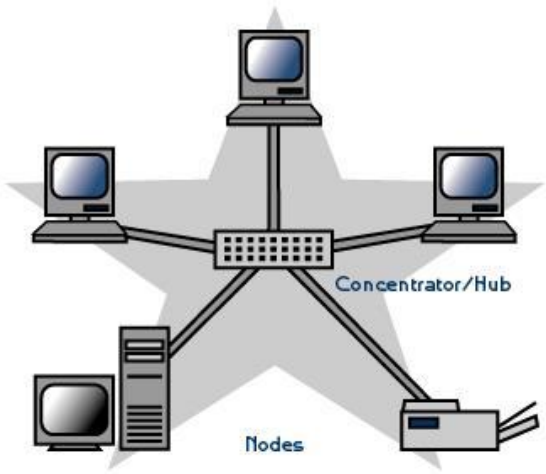
Gambar 4.30 Laporan Data Penjualan

4.3. Perancangan Arsitektur Jaringan

Arsitektur jaringan yang akan digunakan adalah Sistem client/server mempunyai dua komponen utama yaitu komputer client dan komputer server. Server merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer client dan bertindak sebagai server database yang menyimpan data. Client yaitu komputer atau workstasion yang melakukan pengiriman permintaan-permintaan data pada server kemudian menampilkan data tersebut pada interface aplikasi yang dimilikinya. Selain itu client juga mempunyai kemampuan untuk mengubah atau menghapus data itu.

Topologi jaringan yang akan digunakan adalah topologi star (bintang). Topologi star merupakan bentuk topologi jaringan yang berupa konvergensi dari

node tengah ke setiap node atau pengguna. Topologi jaringan bintang termasuk topologi jaringan dengan biaya menengah.



Gambar 4.31 Contoh gambar topologi jaringan star

Kelebihan Topologi Bintang

a. Kerusakan pada satu saluran hanya akan memengaruhi jaringan pada saluran tersebut dan station yang terpaut.
b. Tingkat keamanan termasuk tinggi.
c. Tahan terhadap lalu lintas jaringan yang sibuk. Penambahan dan pengurangan station dapat dilakukan dengan mudah.

Kekurangan

a. Jika node tengah mengalami kerusakan, maka seluruh rangkaian akan berhenti.

4.4 Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan aplikasi sistem informasi penjualan, pembelian dan persediaan barang yang sesuai dengan tujuan yang diharapkan.

4.4.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak pemantauan dan pengendalian aplikasi sistem informasi penjualan, pembelian dan persediaan barang terdapat beberapa hal yang menjadi batasan implementasi, yaitu:

Dalam pengimplementasian aplikasi informasi penjualan, pembelian dan persediaan barang ini, penulis menetapkan beberapa hal yang menjadi batasan implementasi :

1. Aplikasi hanya bias dijalankan pada komputer dengan OS (sistem operasi)Windows.
2. Pengelolaan Data User hanya bisa di akses oleh General manager.
3. Bagian pembelian hanya mengakses data persediaan barang dan pembelian barang kepada supplier.
4. Bagian penjualan hanya mengakses penjualan barang kepada pelanggan.

4.4.2 Implementasi Perangkat Lunak

Perancangan Aplikasi Sistem Informasi penjualan, pembelian, dan persediaan barang yang dibangun ini, dibangun dan diimplementasikan dengan menggunakan bahasa pemrograman PHP yang dinamis dan dapat mudah untuk dikembangkan. Penyimpanan data yang digunakan adalah MySQL Server.

4.4.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan khususnya oleh pihak-pihak yang terkait pengaksesan aplikasi berdasarkan kebutuhan minimal yang harus dipenuhi adalah sebagai berikut:

1. Rekomendasi minimum adalah dengan menggunakan processor minimal intel Pentium IV 2.0 Ghz.
2. Harddisk terpasang 80 GB.
3. Memori (RAM) rekomendasi minimal 512 MB.
4. VGA card terpasang rekomendasi minimal 32 MB.
5. Mouse dan keyboard dan monitor sebagai peralatan antarmuka.

4.4.4. Implementasi Basis Data (Sintaks SQL)

Sebuah basis data atau database, sangat dibutuhkan dalam sebuah sistem informasi yang terkomputerisasi. Pada sistem informasi penjualan, pembelian dan persediaan barang ini dibangun database yang saling berelasi dan berkesinambungan demi terciptanya sistem informasi yang terintegrasi. Berikut adalah basis data dalam bahasa SQL dalam sistem informasi penjualan, pembelian dan persediaan barang :

```
-- phpMyAdmin SQL Dump
```

```
-- version 3.3.9
-- http://www.phpmyadmin.net
--
-- Host: localhost
-- Generation Time: Aug 20, 2014 at 09:58 PM
-- Server version: 5.5.8
-- PHP Version: 5.3.5

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";


/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `db_throve`
--

-- --------------------------------------------------------

--
-- Table structure for table `t_barang`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 15, 2014 at 09:52 AM
--

DROP TABLE IF EXISTS `t_barang`;
CREATE TABLE IF NOT EXISTS `t_barang` (
  `kode_brg` varchar(7) NOT NULL,
  `nama_brg` varchar(40) NOT NULL,
  `harga_beli` double NOT NULL,
  `harga_jual` double NOT NULL,
  `stok` int(2) NOT NULL,
  `nama_kat` varchar(30) NOT NULL,
  `kode_pm` varchar(6) NOT NULL,
  PRIMARY KEY (`kode_brg`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_barang`
--

INSERT DELAYED IGNORE INTO `t_barang` (`kode_brg`,
`nama_brg`, `harga_beli`, `harga_jual`, `stok`, `nama_kat`,
`kode_pm`) VALUES
('BRG-008', 'BLACK BUTTON', 60000, 180000, 30, 'SHIRT', 'PM-
001'),
```

```
('BRG-006', 'MINED DENIM', 70000, 210000, 18, 'LONG PANTS',
'PM-003'),
('BRG-005', 'TRAVELER BLACK', 70000, 210000, 5, 'LONG
PANTS', 'PM-003'),
('BRG-004', 'RED HEART', 45000, 135000, 30, 'POLO', 'PM-
003'),
('BRG-003', 'GREY KICK', 45000, 135000, 10, 'POLO', 'PM-
003'),
('BRG-002', 'LOGOS SPLASH', 35000, 105000, 30, 'T-SHIRT',
'PM-001'),
('BRG-007', 'DIAMOND BUTTON', 60000, 180000, 6, 'SHIRT',
'PM-001'),
('BRG-001', 'BLACK TO BLACK', 35000, 105000, 30, 'T-SHIRT',
'PM-001'),
('BRG-009', 'BRUCE ', 60000, 180000, 30, 'SHIRT', 'PM-001'),
('BRG-010', 'ABSOLUTE RELOAD', 35000, 105000, 30, 'T-SHIRT',
'PM-001'),
('BRG-011', 'BROWN COFEE', 45000, 135000, 18, 'POLO', 'PM-
003'),
('BRG-012', 'susu', 3000, 9000, 100, 'POLO', 'PM-001');

-- --------------------------------------------------------

--
-- Table structure for table `t_detail_pembelian`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 13, 2014 at 03:46 PM
--

DROP TABLE IF EXISTS `t_detail_pembelian`;
CREATE TABLE IF NOT EXISTS `t_detail_pembelian` (
  `kode_pems_pemb` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `qty` int(2) NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_detail_pembelian`
--

INSERT DELAYED IGNORE INTO `t_detail_pembelian`
(`kode_pems_pemb`, `kode_brg`, `qty`, `sub_total`) VALUES
('KPP-2014-07-29-0001', 'BRG-003', 20, 800000),
('KPP-2014-07-29-0001', 'BRG-001', 28, 560000),
('KPP-2014-07-31-0002', 'BRG-001', 20, 400000),
('KPP-2014-07-31-0002', 'BRG-003', 30, 1200000),
('KPP-2014-07-30-0001', 'BRG-002', 34, 1190000),
('KPP-2014-08-02-0001', 'BRG-004', 50, 1750000),
('KPP-2014-08-07-0001', 'BRG-001', 30, 1050000),
('KPP-2014-08-13-0002', 'BRG-006', 20, 1400000),
('KPP-2014-08-13-0001', 'BRG-006', 20, 1400000);
```

```
-- --------------------------------------------------------

--
-- Table structure for table `t_detail_pemesanan_pembelian`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 15, 2014 at 10:02 AM
--

DROP TABLE IF EXISTS `t_detail_pemesanan_pembelian`;
CREATE TABLE IF NOT EXISTS `t_detail_pemesanan_pembelian` (
  `kode_pems_pemb` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `kode_pm` varchar(6) NOT NULL,
  `qty` int(2) NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_detail_pemesanan_pembelian`
--

INSERT DELAYED IGNORE INTO `t_detail_pemesanan_pembelian`
(`kode_pems_pemb`, `kode_brg`, `kode_pm`, `qty`,
`sub_total`) VALUES
('KPP-2014-08-15-0001', 'BRG-005', 'PM-003', 20, 0),
('KPP-2014-08-15-0002', 'BRG-005', 'PM-003', 21, 0);

-- --------------------------------------------------------

--
-- Table structure for table `t_detail_pemesanan_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:22 AM
--

DROP TABLE IF EXISTS `t_detail_pemesanan_penjualan`;
CREATE TABLE IF NOT EXISTS `t_detail_pemesanan_penjualan` (
  `kode_pems_penj` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `qty` int(2) NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_detail_pemesanan_penjualan`
--

INSERT DELAYED IGNORE INTO `t_detail_pemesanan_penjualan`
(`kode_pems_penj`, `kode_brg`, `qty`, `sub_total`) VALUES
('KPJ-2014-07-30-0001', 'BRG-002', 1, 105000),
```

```
('KPJ-2014-08-07-0003', 'BRG-006', 5, 1050000),
('KPJ-2014-08-02-0002', 'BRG-005', 5, 1050000),
('KPJ-2014-08-07-0003', 'BRG-001', 30, 3150000),
('KPJ-2014-08-14-0004', 'BRG-006', 12, 2520000),
('KPJ-2014-08-14-0005', 'BRG-006', 10, 2100000),
('KPJ-2014-08-14-0005', 'BRG-003', 10, 1350000);

-- --------------------------------------------------------

--
-- Table structure for table `t_detail_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:21 AM
--

DROP TABLE IF EXISTS `t_detail_penjualan`;
CREATE TABLE IF NOT EXISTS `t_detail_penjualan` (
  `kode_penj` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `qty` int(2) NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_detail_penjualan`
--

INSERT DELAYED IGNORE INTO `t_detail_penjualan`
(`kode_penj`, `kode_brg`, `qty`, `sub_total`) VALUES
('KPL-2014-07-30-0001', 'BRG-003', 1, 120000),
('KPL-2014-07-30-0002', 'BRG-004', 1, 105000),
('KPL-2014-07-30-0002', 'BRG-008', 1, 108000),
('KPL-2014-07-30-0003', 'BRG-009', 3, 135000),
('KPL-2014-07-31-0004', 'BRG-005', 20, 2700000),
('KPL-2014-07-31-0004', 'BRG-002', 2, 210000),
('KPL-2014-08-02-0005', 'BRG-004', 18, 1890000),
('KPL-2014-08-02-0006', 'BRG-006', 5, 1050000),
('KPL-2014-08-07-0007', 'BRG-005', 20, 4200000),
('KPL-2014-08-07-0007', 'BRG-007', 24, 4320000),
('KPL-2014-08-07-0007', 'BRG-006', 20, 4200000),
('KPL-2014-08-14-0008', 'BRG-011', 12, 1620000),
('KPL-2014-08-14-0008', 'BRG-003', 10, 1350000);

-- --------------------------------------------------------

--
-- Table structure for table `t_kategori`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 13, 2014 at 03:46 PM
--
```

```
DROP TABLE IF EXISTS `t_kategori`;
CREATE TABLE IF NOT EXISTS `t_kategori` (
  `kode_kat` int(2) NOT NULL AUTO_INCREMENT,
  `nama_kat` varchar(30) NOT NULL,
  PRIMARY KEY (`kode_kat`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

--
-- Dumping data for table `t_kategori`
--

INSERT DELAYED IGNORE INTO `t_kategori` (`kode_kat`,
`nama_kat`) VALUES
(14, 'T-SHIRT'),
(17, 'LONG PANTS'),
(16, 'SHIRT'),
(15, 'POLO');

-- --------------------------------------------------------

--
-- Table structure for table `t_pelanggan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 13, 2014 at 03:46 PM
--

DROP TABLE IF EXISTS `t_pelanggan`;
CREATE TABLE IF NOT EXISTS `t_pelanggan` (
  `kode_pl` varchar(6) NOT NULL,
  `nama_pl` varchar(50) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `kota` varchar(30) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `kodepos` varchar(10) NOT NULL,
  PRIMARY KEY (`kode_pl`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pelanggan`
--

INSERT DELAYED IGNORE INTO `t_pelanggan` (`kode_pl`,
`nama_pl`, `alamat`, `kota`, `telepon`, `kodepos`) VALUES
('PL-003', 'Diana Mulyawati', 'Jl. Dago atas no. 90',
'Bandung', '022-567788', '678921'),
('PL-004', 'Jaka Munir', 'Jl. Gagak Selatan Sekeloa No. 19',
'Bandung', '022786541', '40000'),
('PL-005', 'Endun Kasinah', 'Jl. Cileunyi no. 11',
'Bandung', '089767788', '40123'),
('PL-006', 'ona tayane', 'jl. siliwangi no 9', 'bandung',
'085297876655', '56121');
```

```
-- --------------------------------------------------------

--
-- Table structure for table `t_pemasok`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 13, 2014 at 03:46 PM
--

DROP TABLE IF EXISTS `t_pemasok`;
CREATE TABLE IF NOT EXISTS `t_pemasok` (
  `kode_pm` varchar(6) NOT NULL,
  `nama_pm` varchar(50) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `kota` varchar(30) NOT NULL,
  `telepon` varchar(20) NOT NULL,
  `kodepos` varchar(10) NOT NULL,
  PRIMARY KEY (`kode_pm`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pemasok`
--

INSERT DELAYED IGNORE INTO `t_pemasok` (`kode_pm`,
`nama_pm`, `alamat`, `kota`, `telepon`, `kodepos`) VALUES
('PM-001', 'CV. Brother Fancy', 'Jl. Borobudur No. 15',
'Jakarta', '021-3456798', '548799'),
('PM-002', 'CV. Putra Mekar', 'Jl. Jati Padang No. 60',
'Jakarta', '021-2878967', '345678'),
('PM-003', 'CV. Global Fashion', 'Jl. Kepatihan  no. 201
Jakarta Barat', 'DKI Jakarta', '0219899923', '567834'),
('PM-004', 'CV. Jaya Winata', 'Jl, Surapati no. 125',
'Bandung', '02277848499', '40012');

-- --------------------------------------------------------

--
-- Table structure for table `t_pembelian`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 13, 2014 at 03:46 PM
--

DROP TABLE IF EXISTS `t_pembelian`;
CREATE TABLE IF NOT EXISTS `t_pembelian` (
  `kode_pemb` varchar(19) NOT NULL,
  `tgl_pemb` date NOT NULL,
  `kode_pems_pemb` varchar(19) NOT NULL,
  `tgl_pems_pemb` date NOT NULL,
  `kode_pm` varchar(6) NOT NULL,
  `total_item` int(2) NOT NULL,
  `total` double NOT NULL,
```

```
  PRIMARY KEY (`kode_pemb`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pembelian`
--

INSERT DELAYED IGNORE INTO `t_pembelian` (`kode_pemb`,
`tgl_pemb`, `kode_pems_pemb`, `tgl_pems_pemb`, `kode_pm`,
`total_item`, `total`) VALUES
('KTP-2014-07-30-0001', '2014-07-30', 'KPP-2014-07-29-0001',
'2014-07-29', 'PM-003', 2, 1360000),
('KTP-2014-07-31-0002', '2014-07-31', 'KPP-2014-07-31-0002',
'2014-07-31', 'PM-003', 2, 1600000),
('KTP-2014-07-31-0003', '2014-07-31', 'KPP-2014-07-30-0001',
'2014-07-30', 'PM-002', 1, 1190000),
('KTP-2014-08-02-0004', '2014-08-02', 'KPP-2014-08-02-0001',
'2014-08-02', 'PM-003', 1, 1750000),
('KTP-2014-08-07-0005', '2014-08-07', 'KPP-2014-08-07-0001',
'2014-08-07', 'PM-001', 1, 1050000),
('KTP-2014-08-14-0006', '2014-08-14', 'KPP-2014-08-13-0002',
'2014-08-13', 'PM-003', 1, 1400000),
('KTP-2014-08-13-0007', '2014-08-13', 'KPP-2014-08-13-0001',
'2014-08-13', 'PM-003', 1, 1400000);

-- --------------------------------------------------------

--
-- Table structure for table `t_pemesanan_pembelian`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 15, 2014 at 10:02 AM
--

DROP TABLE IF EXISTS `t_pemesanan_pembelian`;
CREATE TABLE IF NOT EXISTS `t_pemesanan_pembelian` (
  `kode_pems_pemb` varchar(19) NOT NULL,
  `tgl_pems_pemb` date NOT NULL,
  `kode_pm` varchar(6) NOT NULL,
  `total_item` int(2) NOT NULL,
  `total` double NOT NULL,
  PRIMARY KEY (`kode_pems_pemb`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pemesanan_pembelian`
--

INSERT DELAYED IGNORE INTO `t_pemesanan_pembelian`
(`kode_pems_pemb`, `tgl_pems_pemb`, `kode_pm`, `total_item`,
`total`) VALUES
('KPP-2014-08-15-0001', '2014-08-15', 'PM-003', 1, 0),
('KPP-2014-08-15-0002', '2014-08-15', 'PM-003', 1, 0);
```

```
-- --------------------------------------------------------

--
-- Table structure for table `t_pemesanan_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:22 AM
--

DROP TABLE IF EXISTS `t_pemesanan_penjualan`;
CREATE TABLE IF NOT EXISTS `t_pemesanan_penjualan` (
  `kode_pems_penj` varchar(19) NOT NULL,
  `tgl_pems_penj` date NOT NULL,
  `kode_pl` varchar(6) NOT NULL,
  `total_item` int(2) NOT NULL,
  `total` double NOT NULL,
  `status` varchar(20) NOT NULL,
  PRIMARY KEY (`kode_pems_penj`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pemesanan_penjualan`
--

INSERT DELAYED IGNORE INTO `t_pemesanan_penjualan`
(`kode_pems_penj`, `tgl_pems_penj`, `kode_pl`, `total_item`,
`total`, `status`) VALUES
('KPJ-2014-07-31-0001', '2014-07-31', 'PL-004', 1, 1863000,
'1'),
('KPJ-2014-08-02-0002', '2014-08-02', 'PL-006', 1, 1050000,
'1'),
('KPJ-2014-08-07-0003', '2014-08-07', 'PL-004', 2, 4200000,
'1'),
('KPJ-2014-08-14-0004', '2014-08-14', 'PL-006', 1, 2520000,
'0'),
('KPJ-2014-08-14-0005', '2014-08-14', 'PL-006', 2, 3450000,
'0');

-- --------------------------------------------------------

--
-- Table structure for table `t_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:21 AM
--

DROP TABLE IF EXISTS `t_penjualan`;
CREATE TABLE IF NOT EXISTS `t_penjualan` (
  `kode_penj` varchar(19) NOT NULL,
  `tgl_penj` date NOT NULL,
  `total` double NOT NULL,
```

```
  PRIMARY KEY (`kode_penj`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_penjualan`
--

INSERT DELAYED IGNORE INTO `t_penjualan` (`kode_penj`,
`tgl_penj`, `total`) VALUES
('KPL-2014-07-30-0001', '2014-07-30', 120000),
('KPL-2014-07-30-0002', '2014-07-30', 213000),
('KPL-2014-07-30-0003', '2014-07-30', 135000),
('KPL-2014-07-31-0004', '2014-07-31', 2910000),
('KPL-2014-08-02-0005', '2014-08-02', 1890000),
('KPL-2014-08-02-0006', '2014-08-02', 1050000),
('KPL-2014-08-07-0007', '2014-08-07', 12720000),
('KPL-2014-08-14-0008', '2014-08-14', 2970000);

-- --------------------------------------------------------

--
-- Table structure for table
`t_tmp_detail_pemesanan_pembelian`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 15, 2014 at 10:04 AM
--

DROP TABLE IF EXISTS `t_tmp_detail_pemesanan_pembelian`;
CREATE TABLE IF NOT EXISTS
`t_tmp_detail_pemesanan_pembelian` (
  `kode_pems_pemb` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `kode_pm` varchar(6) NOT NULL,
  `qty` int(2) NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_detail_pemesanan_pembelian`
--

INSERT DELAYED IGNORE INTO
`t_tmp_detail_pemesanan_pembelian` (`kode_pems_pemb`,
`kode_brg`, `kode_pm`, `qty`, `sub_total`) VALUES
('null', '', '', 0, 0);

-- --------------------------------------------------------

--
-- Table structure for table
`t_tmp_detail_pemesanan_penjualan`
--
```

```
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:22 AM
--

DROP TABLE IF EXISTS `t_tmp_detail_pemesanan_penjualan`;
CREATE TABLE IF NOT EXISTS
`t_tmp_detail_pemesanan_penjualan` (
  `kode_pems_penj` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `nama_brg` varchar(30) NOT NULL,
  `qty` int(2) NOT NULL,
  `stok` int(2) NOT NULL,
  `harga_jual` double NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_detail_pemesanan_penjualan`
--

INSERT DELAYED IGNORE INTO
`t_tmp_detail_pemesanan_penjualan` (`kode_pems_penj`,
`kode_brg`, `nama_brg`, `qty`, `stok`, `harga_jual`,
`sub_total`) VALUES
('KPJ-2014-08-11-0004', 'BRG-002', 'LOGOS SPLASH', 6, 30,
105000, 630000),
('KPJ-2014-08-13-0004', 'BRG-011', 'BROWN COFEE', 21, 30,
135000, 2835000),
('KPJ-2014-08-13-0004', 'BRG-004', 'RED HEART', 26, 30,
135000, 3510000);

-- --------------------------------------------------------

--
-- Table structure for table `t_tmp_detail_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 11:04 AM
--

DROP TABLE IF EXISTS `t_tmp_detail_penjualan`;
CREATE TABLE IF NOT EXISTS `t_tmp_detail_penjualan` (
  `kode_penj` varchar(19) NOT NULL,
  `kode_brg` varchar(7) NOT NULL,
  `nama_brg` varchar(30) NOT NULL,
  `qty` int(2) NOT NULL,
  `stok` int(2) NOT NULL,
  `harga_jual` double NOT NULL,
  `sub_total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_detail_penjualan`
```

```
--

INSERT DELAYED IGNORE INTO `t_tmp_detail_penjualan`
(`kode_penj`, `kode_brg`, `nama_brg`, `qty`, `stok`,
`harga_jual`, `sub_total`) VALUES
('KPL-2014-08-14-0009', 'BRG-003', 'GREY KICK', 0, 10,
135000, 0);

-- --------------------------------------------------------

--
-- Table structure for table `t_tmp_pelanggan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 08:34 AM
--

DROP TABLE IF EXISTS `t_tmp_pelanggan`;
CREATE TABLE IF NOT EXISTS `t_tmp_pelanggan` (
  `kode_pl` varchar(6) NOT NULL,
  `nama_pl` varchar(50) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_pelanggan`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_tmp_pemasok`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 15, 2014 at 10:04 AM
--

DROP TABLE IF EXISTS `t_tmp_pemasok`;
CREATE TABLE IF NOT EXISTS `t_tmp_pemasok` (
  `kode_pm` varchar(6) NOT NULL,
  `nama_pm` varchar(50) NOT NULL,
  `total` double NOT NULL,
  PRIMARY KEY (`kode_pm`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_pemasok`
--
```

```
-- --------------------------------------------------------

--
-- Table structure for table `t_tmp_penjualan`
--
-- Creation: Aug 13, 2014 at 03:46 PM
-- Last update: Aug 14, 2014 at 11:04 AM
--

DROP TABLE IF EXISTS `t_tmp_penjualan`;
CREATE TABLE IF NOT EXISTS `t_tmp_penjualan` (
  `kode_penj` varchar(19) NOT NULL,
  `total` double NOT NULL,
  PRIMARY KEY (`kode_penj`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_tmp_penjualan`
--

INSERT DELAYED IGNORE INTO `t_tmp_penjualan` (`kode_penj`,
`total`) VALUES
('KPL-2014-08-14-0009', 0);

-- --------------------------------------------------------

--
-- Table structure for table `t_user`
--
-- Creation: Aug 13, 2014 at 03:46 PM
--

DROP TABLE IF EXISTS `t_user`;
CREATE TABLE IF NOT EXISTS `t_user` (
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `login` varchar(50) NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_user`
--

INSERT DELAYED IGNORE INTO `t_user` (`username`, `password`,
`login`) VALUES
('gm', '92073d2fe26e543ce222cc0fb0b7d7a0', 'Manager'),
('gudang', '202446dd1d6028084426867365b0c7a1', 'Gudang'),
('pembelian', '025b94c9e65037bb7317c8e25389155d',
'Pembelian'),
('penjualan', '13bf2c8effae21d17a277520aa9b9277',
'Penjualan');
```

4.4.5. Implementasi Antar Muka

Antar muka sebuah perangkat lunak adalah hal yang sangat penting, karena program yang baik adalah program yang dapat menciptakan media yang baik pula antara general manager, bagian pembelian dan bagian penjualan. Oleh karena itu berikut merupakan gambaran implementasi antar muka pada aplikasi sistem informasi penjualan, pembelian dan persediaan barang yang digambarkan sesuai dengan klasifikasi hak akses anggota dan hak akses petugas.

4.4.6. **Implementasi Instalasi Program**

Langkah terakhir dari perancangan aplikasi sistem informasi penjualan, pembelian dan persediaan barang adalah dilakukannya proses uploading pada internet agar dapat di akses tanpa batas ruang dan waktu. Proses uploading ini dapat berupa fasilitas gratis (free hosting) atau paid hosting. Ada pun cara-cara meng-upload website ke internet adalah sebagai berikut :

1. Signup ke salah satu web penyedia webhosting untuk meng-upload isi web, misal : www.phpnet.us
2. Setelah berhasil mendaftar maka akan mendapatkan accout pada phpnet.us seperti :

   Welcome to phpnet.us

Your account details are :

FTP

Server/Host : ftp.phpnet.us

Username : pn_624448

Password : xxxxx

Short flash movie : How to FTP connect to your web site:

http://www.phpnet.us/videos/FTP.swf

MySQL

Username : pn_624448

Password : xxxxx

Server : sql1.phpnet.us

Please Create your database in the cpanel

Cpanel URL : http://cpanel.phpnet.us

Cpanel User : pn_624448

Cpanel Password : xxxxx

3. Setelah itu, melakukan login pada phpnet.us melalui http://cpanel.phpnet.us
4. ZIP seluruh file Website menggunakan Winzip/winrar menjadi satu file . untuk mempercepat proses upload, bila menggunakan xampp sebagai webserver dan mysql server maka letak website pada C:\Program Files\xampp\htdocs\.
5. Upload file zip web dengan mengklik file manager.
6. Klik link www.

7. Klik tombol upload.
8. Upload file zip anda pada bagian archieve dengan mengklik tombol browser.
9. Pilih file zip anda dan klik open.
10. Setelah di pilih klik tombol centang pada pojok kiri atas.
11. Phpnet.us akan melakukan file checking pada file anda, jika sukses maka akan muncul tampilan konformasi.
12. Selanjutnya klik gambar centang.
13. Cek pada file manager apakah website telah terupload pada folder www.
14. Lakukan upload databases, dengan cara :

    Apabila anda telah membuat website dari rumah, dan akan mengupload web tersebut, maka anda perlu membackup databases web yang telah anda buat dengan menggunakan phpmyadmin, dengan cara :

    a. Buka browser baru
    b. Pada Address isikan http://localhost/phpmyadmin
    c. Pada Pilihan databases klik database auracms (sesuai dengan nama databases yang anda buat)
    d. Klik bar Export untuk membackup
    e. Pada web browser paling bawah centang Save As dan beri nama file backup (misal : auracms) dan klik go.
    f. Download file tersebut letakkan pada flashdisk/media penyimpanan lain agar dapat diupload pada warnet atau kampus.
    g. Buat databases melalui cpanel phpnet.us dengan mengklik link MySQ manager.
    h. Buat databases dengan nama auracms lalu klik klik create.
    i. Databases anda telah berhasil di buat.

j. Kembali ke menu cpanel phpnet.us, lalu lakukan upload databases dengan mengklik link phpmyadmin.
k. Lakukan koneksi ke databases yang telah anda buat, dengan mengklik databases yang telah anda create tadi.
l. Lakukan import databases dengan mengklik bar Import.
m. Browse databases yang telah anda backup (auracms) tadi dan klik open.
n. Lalu klik Go.
o. Kembali ke cpanel phpnet.us.

15. Lakukan editing script pada config.php aura cms dengan masuk ke file manager phpnet.us, klik link www, edit file config.php yang terletak pada include
16. Isikan mysql user dengan user,password anda dengan username dan password anda pada phpnet.us, sedangkan mysql_databases diisikan dengan databases yang telah anda ciptakan pada phpnet.us , sedangkan mysql_host diisi dengan sql1.phpnet.us
17. Klik icon disket telah berhasil mengupload web ke internet.

Langkah – langkah memforward domain menjadi alamat lain yaitu www.langgamdimensi.com

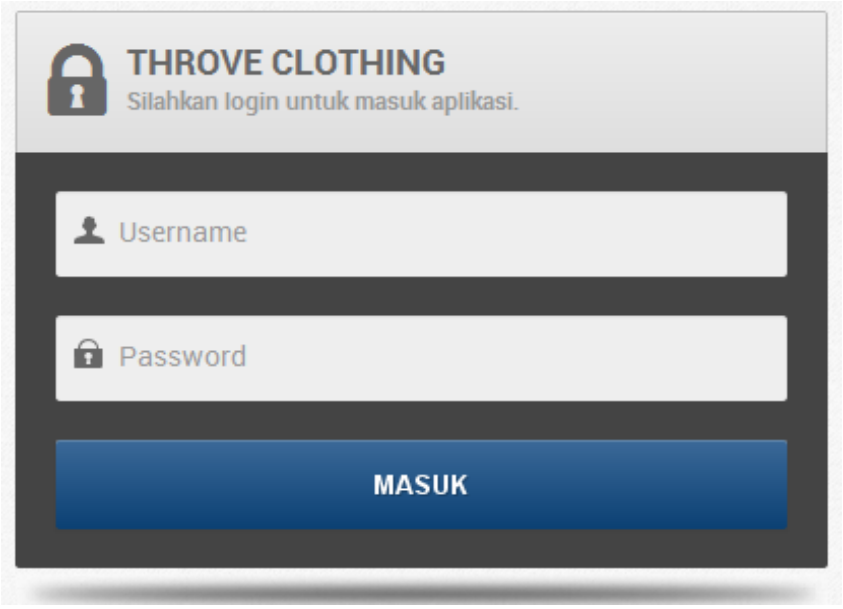
1. Buka www.co.nr pada address bar web browser.
2. Isikan alamat yang anda inginkan misal e-crm (tapi akhiran tetep co.nr).
3. Jika alamat tsb belum dipakai orang maka dapat anda signup.
4. Isi pertanyaan untuk menghindari spam, lalu go to next step.
5. Baca peraturan dan klik I have read and I agree.
6. Isi formulir seperti anda membuat website, yang paling penting adalah target url di arahkan pada alamat web anda pada phpnet.us.

7. Klik Next Step.
8. Isikan informasi mengenai isi web.
9. Klik Next Step.
10. Anda telah mempunyai alamat domain baru.
11. Klik Complete untuk mengakhiri dan anda akan dikirimi email oleh co.nr yang berisi aktivasi code buka email anda dan klik link untuk mengaktifkan lalu isi dengan benar.

4.4.7 Penggunaan Program

1. Form Login

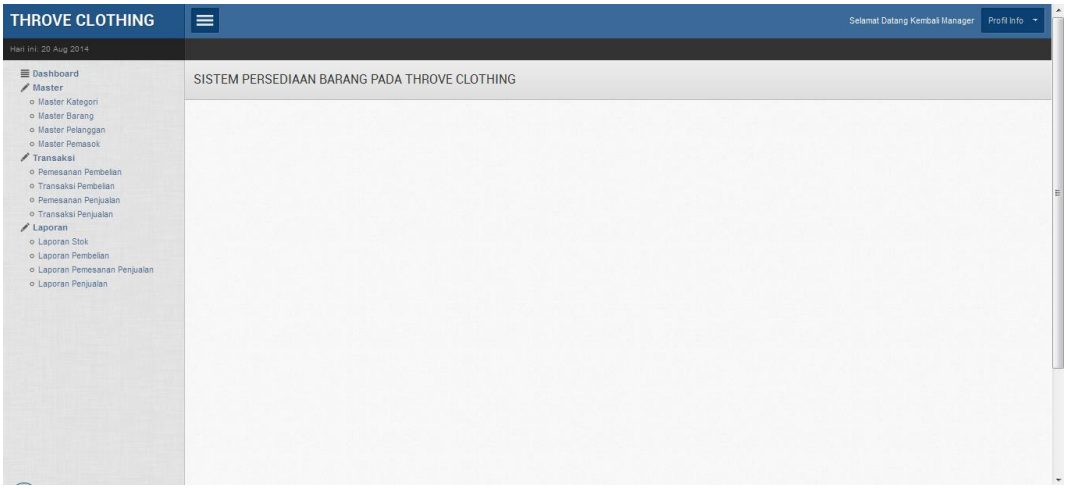
Form login ini untuk menentukan hakakses penggunaan sistem.



Gambar 4.32 Form Login.

2. Form Utama General Manager

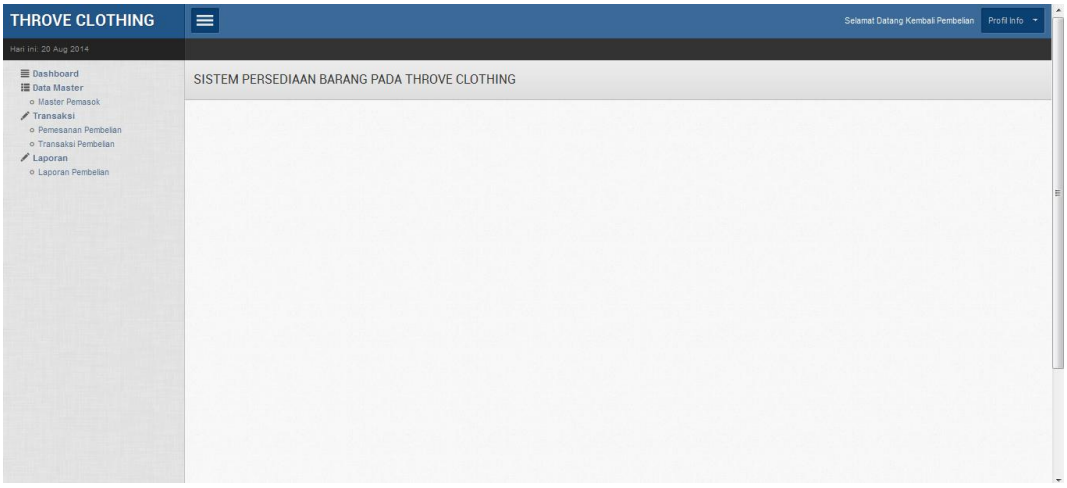
Halaman ini merupakan halaman utama ketika gm masuk setelah melakukan login.



Gambar 4.33 Halaman Utama General Manager.

3. Halaman Utama Pembelian

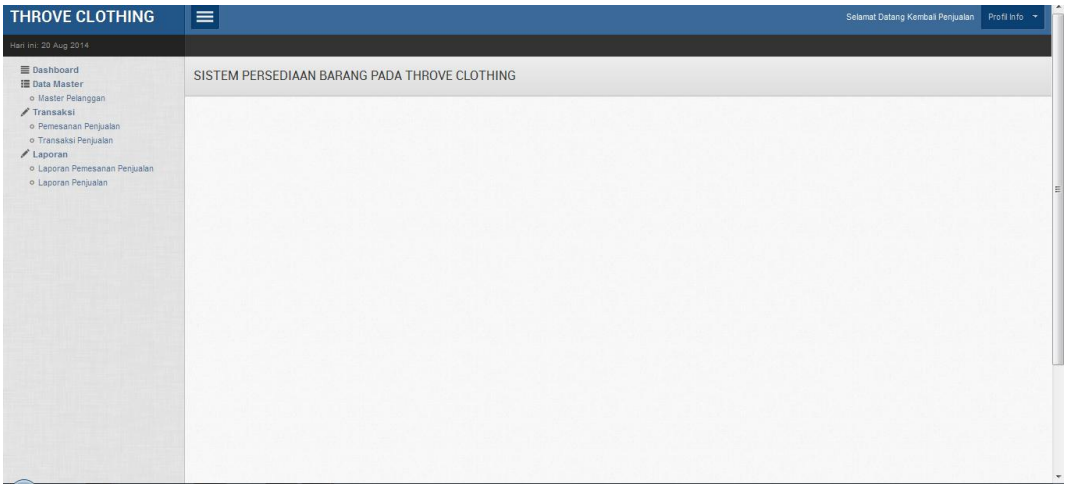
Halaman ini merupakan halaman utama ketika bagian pembelian masuk setelah melakukan login



Gambar 4.34 Halaman Utama Pembelian

2. Halaman Utama Penjualan

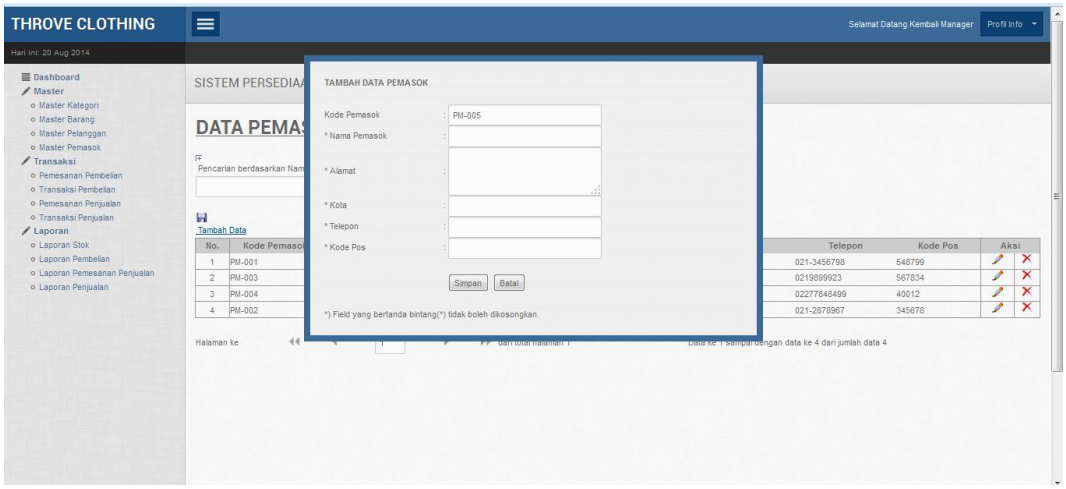
Halaman ini merupakan halaman utama ketika bagian penjualan masuk setelah melakukan login.



Gambar 4.35 Halaman Utama Penjualan

3. Form Data Supplier

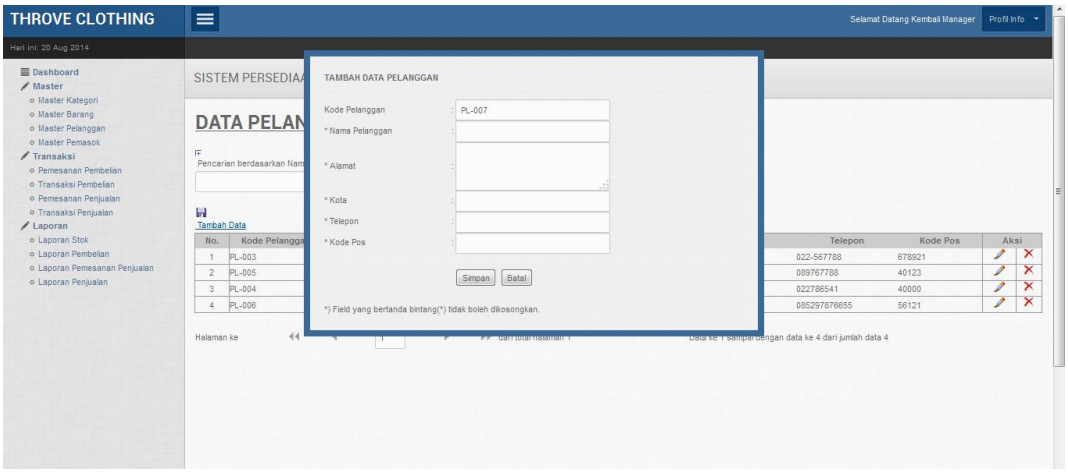
Form data supplier hanya dapat di akses bila login sebagai pembelian, fungsi form ini untuk menambah,merubah dan menghapus data supplier.



Gambar 4.36 Form Supplier

4. Form Data Customer

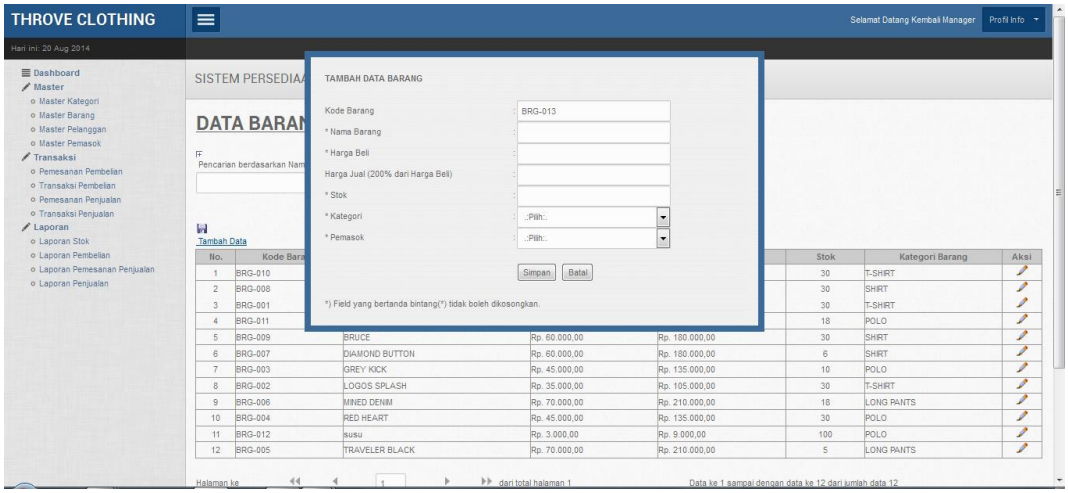
Form data customer hanya dapat di akses bila login sebagai penjualan, fungsi form ini untuk menambah,merubah dan menghapus data customer.



Gambar 4.37 Form Customer

4. Form Data Barang

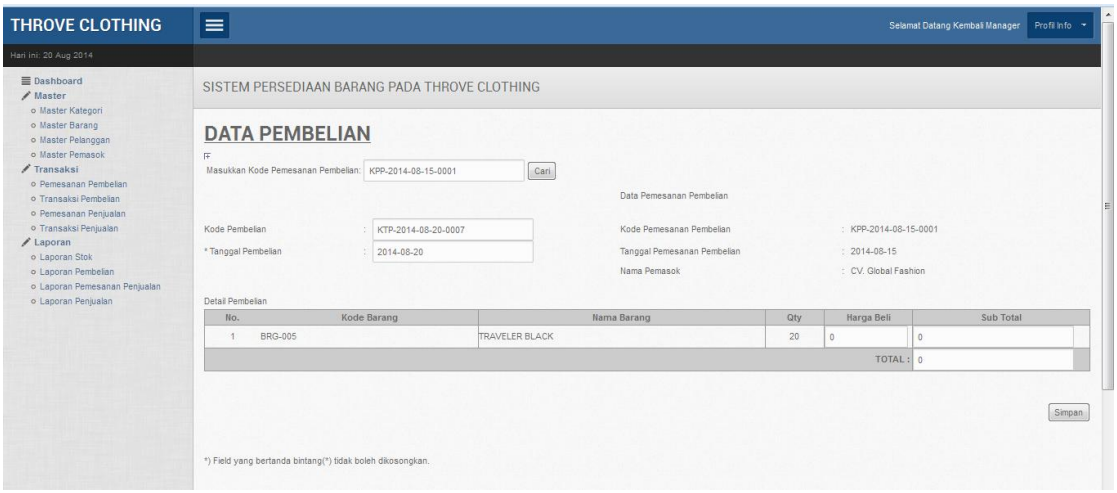
Dalam form ini user bisa melihat list barang yang ada serta user bisa menambah,edit serta mnghapus data barang.



Gambar 4.38 Form Barang

5. Form Data Pembelian

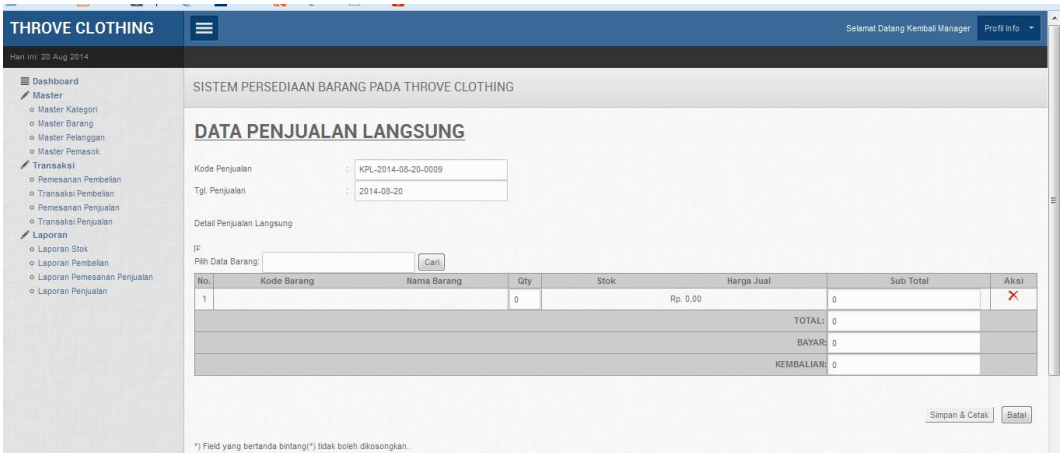
Dalam form ini user bisa melihat data barang yang ada serta menginput kuantiti barang



Gambar 4.39 Form Data Pembelian

7. Form Barang Keluar

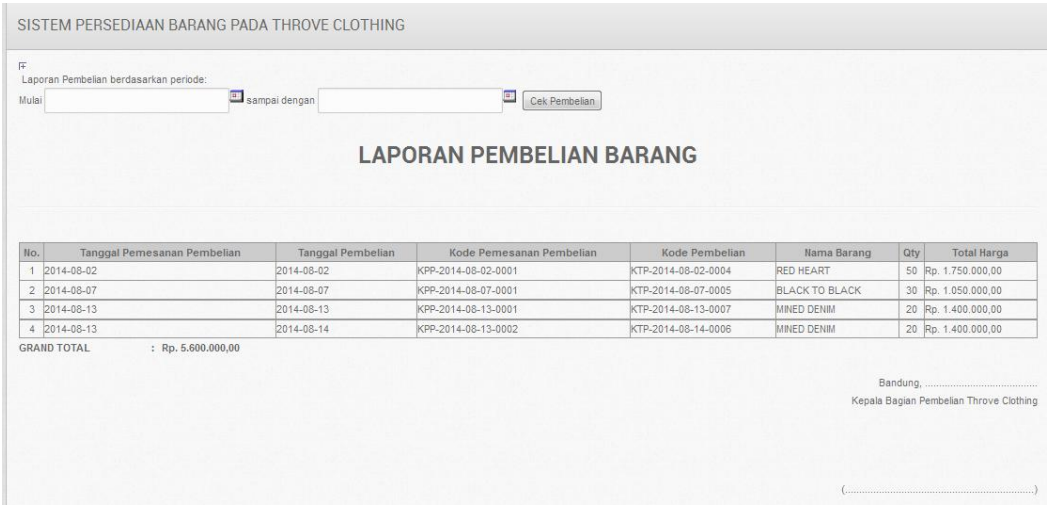
Dalam form ini user bisa melihat data barang yang ada serta user bisa menambah,edit serta menghapus data barang keluar.



Gambar 4.40 Form Data Penjualan

10. Form Laporan Pembelian

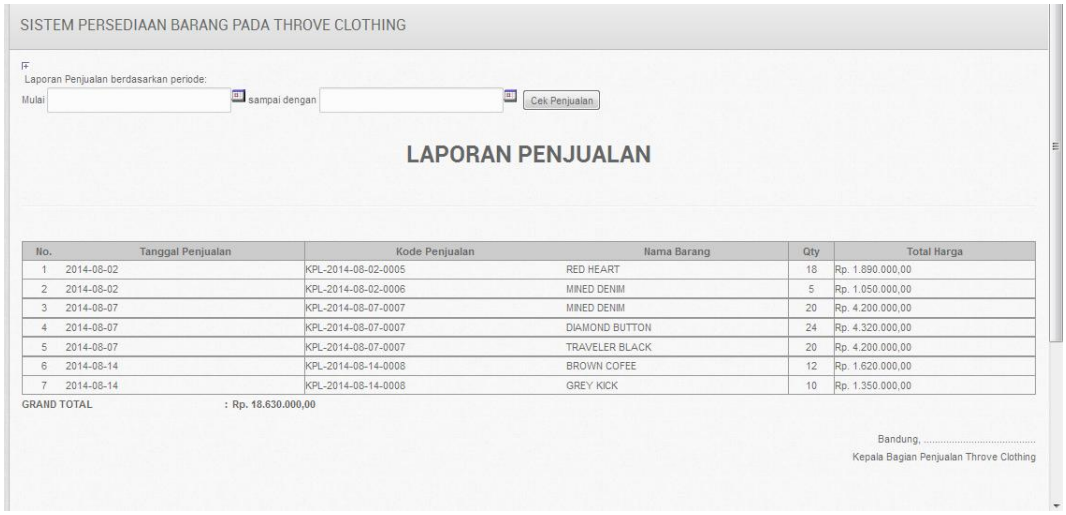
Form laporan pembelian ini berfungsi untuk membuat laporan harian maupun perperiode.



Gambar 4.41 Form Laporan Pembelian

11. From Laporan Penjualan

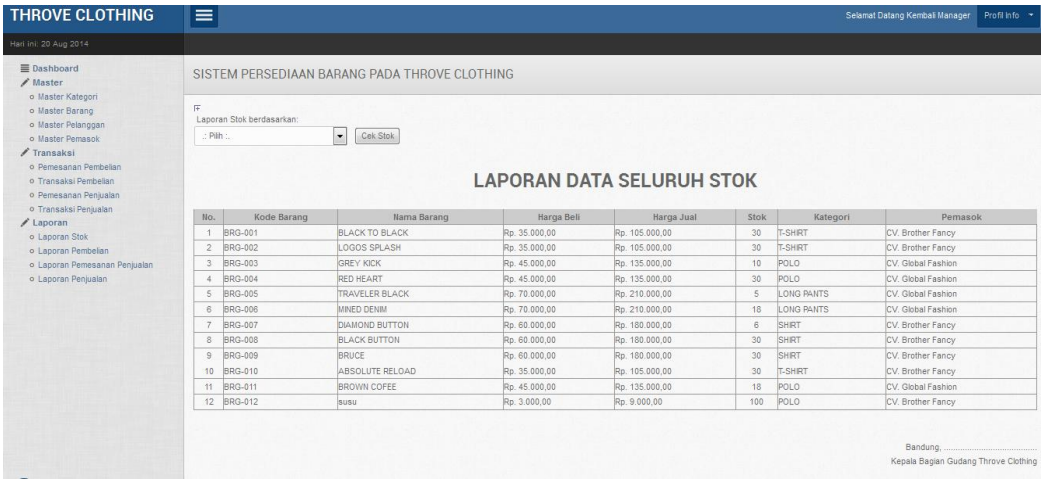
Form laporan penjualan ini berfungsi untuk membuat laporan harian maupun perperiode.



Gambar 4.42 Form Laporan Penjualan

12. From Laporan Semua Stok

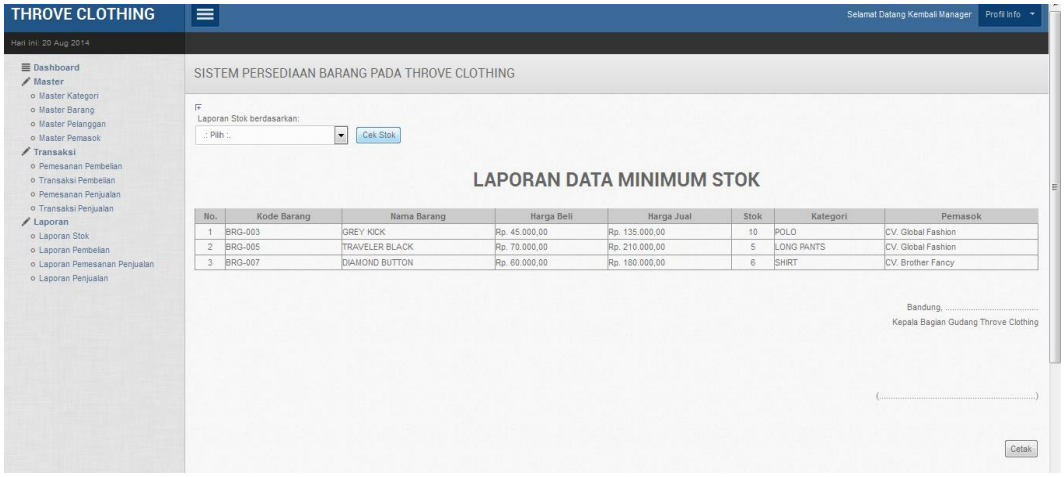
Form laporan Semua Stok berfungsi untuk mengetahui laporan keseluruhan stok barang



Gambar 4.43 Laporan Semua Stok

13. Laporan Minimum Stok

Form laporan Minimum Stok untuk mengetahui stok barang apa saja yang telah minimum



Gambar 4.44 Laporan Minimum Stok

4.5. Pengujian

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal. Pengujian terhadap program itu sendiri yang bertujuan agar program dapat berjalan dengan baik tanpa mengalami gangguan atau error, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut. Pengujian perangkat lunak ini menggunakan metode pengujian black box. Pengujian black box ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Berikut ini adalah rencana pengujian input/output aplikasi sistem informasi penjualan, pembelian dan pengelolaan barang pada Throve Clothing.

4.5.1. Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Pengujian aplikasi sistem informasi penjualan, pembelian dan pengelolaan barang pada Throve Clothing berikut menggunakan data uji berupa masukan dari gm, pembelian dan penjualan. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini :

Tabel 5.4 Rencana Pengujian

Kelas Uji	Detail Pengujian	Jenis Pengujian
Login gm	Verifikasi data login gm dengan memasukkan *username* dan kata sandi oleh gm.	*Black Box*
Login pembelian	Verifikasi data login pembelian dengan memasukkan *username* dan kata sandi oleh pembelian.	*Black Box*
Login penjualan	Verifikasi data login penjualan dengan memasukkan *username* dan kata sandi oleh penjualan.	*Black Box*
Pengolahan data pengguna	Proses pengisian data untuk pemberian hak akses kepada bagian yang terkait.	*Black Box*
	Proses batal untuk mengisi data pengguna	
Pengolahan data anggota	Proses pengisian data anggota pada form anggota serta pengisian aktivasi untuk akses login anggota beserta proses simpan.	*Black Box*
	Proses batal untuk mengisi data pengguna.	
Pengolahn data Supplier	Proses pengisian data supplier pada form supplier beserta proses simpan.	
	Proses batal untuk mengisi data supplier.	
Pengolahn data Customer	Proses pengisian data customer pada form customer beserta proses simpan.	
	Proses batal untuk mengisi data customer.	
Pengolahan data kategori	Proses pengisian data kategori pada form kategori beserta proses simpan.	
	Proses batal untuk mengisi data kategori.	
Pengolahan data Barang masuk	Proses pengisian data barang masuk yang didalamnya ada pengisian data barang, jumlah barang dan supplier pada form barang masuk beserta proses simpan.	*Black Box*

	Proses batal untuk mengisi data barang masuk.	*Black Box*
	Cetak data barang masuk yang dilakukan oleh gm, dan pembelian.	
Pengolahan data Barang keluar	Proses pengisian data barang keluar yang didalamnya ada pengisian data barang, jumlah barang dan customer pada form barang keluar beserta proses simpan.	
	Proses batal untuk mengisi data barang keluar.	
	Cetak data barang masuk yang dilakukan oleh gm, dan penjualan.	

4.5.2. Kasus dan Hasil Pengujian

Pengujian perangkat lunak ini dilakukan pada modul-modul yang ada pada aplikasi sistem informasi penjualan, pembelian dan persedian barang, maka dapat dilakukan pengujian sebagai berikut:

1. Pengujian *Login* Gm

Pengujian login gm yang mempunyai hak akses penuh untuk melakukan pengelolaan data pengguna, pengolahan laporan penjualan, pembelian, histori penjualan dan histori pembelian, dalam aplikasi sistem informasi penjualan, pembelian dan persedian barang pada Throve Clothing.

Tabel 5.5 Pengujian *Login* Genera Manager

Kasus dan Hasil Uji *Login* Gm (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan

Nama Pengguna : Gm Kata Sandi : gm	Dapat masuk ke form Gm	Nama pengguna dan kata sandi sesuai dengan hak akses.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji *Login* Gm (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pengguna : Gm Kata Sandi : salah	Tidak dapat login sebagai Gm dan menampilkan pesan "Nama Pengguna dan Kata Sandi yang dimasukan salah".	Nama pengguna dan kata sandi tidak sesuai dengan hak akses.	[X] Diterima [ ] Ditolak

2. Pengujian *Login* Pembelian

Pengujian login pembelian jika berhasil melakukan login maka pembelian dapat mengakses pengelolaan data barang masuk.

Tabel 5.6 Pengujian *Login* Pembelian

Kasus dan Hasil Uji *Login* Pembelian (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nis Pengguna : Pembelian Kata Sandi : Pembelian	Dapat masuk ke form halaman utama pembelian.	Nama pengguna dan kata sandi sesuai dengan hak akses.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji *Login* Pembelian (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pengguna : Pembelian Kata Sandi : salah	Tidak dapat login sebagai pembelian dan menampilkan pesan “Login gagal, periksa kembali *username* dan password”.	Nama pengguna dan kata sandi tidak sesuai dengan hak akses.	[X] Diterima [ ] Ditolak

3. Pengujian *Login* Penjualan

Pengujian login penjualan jika berhasil melakukan login maka penjualan dapat mengakses pengelolaan data barang keluar.

Tabel 5.7 Pengujian *Login* Penjualan

Kasus dan Hasil Uji *Login* Penjualan (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nis Pengguna : Penjualan Kata Sandi : Penjualan	Dapat masuk ke form halaman utama penjualan.	Nama pengguna dan kata sandi sesuai dengan hak akses.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji *Login* Penjualan (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pengguna : Penjualan Kata Sandi : salah	Tidak dapat login sebagai penjualan dan menampilkan pesan “Login gagal, periksa kembali	Nama pengguna dan kata sandi tidak sesuai dengan hak akses.	[X] Diterima [ ] Ditolak

	username dan password".		

4. Pengujian Pengolahan Data barang

Pengujian ini adalah pengujian dengan memasukan pengisian data barang dengan form data barang ke dalam database.

Tabel 5.8 Pengujian Pengolahan Data Barang.

Kasus dan Hasil Uji Pengisian Data Barang (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data barang	Dapat masuk ke form isian barang.	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah tersimpan.	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol	Data yang telah diinputkan	Tombol reset sesuai yang	[X] Diterima

batal	pada *filed* menjadi kosong.	diharapkan.	[ ] Ditolak

5. Pengujian Pengolahan Data Barang Masuk

Pengujian ini adalah pengujian dengan memasukan pengisian data barang masuk dengan form data barang masuk ke dalam database.

Tabel 5.9 Pengujian Pengolahan Data Barang Masuk

Kasus dan Hasil Uji Pengisian Data Buku (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data barang	Dapat masuk ke form isian barang masuk	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah tersimpan.	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol batal	Data yang telah diinputkan pada *filed* menjadi kosong.	Tombol reset sesuai yang diharapkan.	[X] Diterima [ ] Ditolak

Klik tombol Cetak	Data baranag masuk dapat langsung dicetak dan akan menjadi sebuah laporan	Tombol print sesuai dengan yang diharapkan.	[X] Diterima [ ] Ditolak

6. Pengujian Pengolahan Data Barang Keluar

Pengujian ini adalah pengujian dengan memasukan pengisian data barang keluar dengan form data barang keluar ke dalam database.

Tabel 5.10 Pengujian Pengolahan Data Barang Keluar

Kasus dan Hasil Uji Pengisian Data Buku (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data barang	Dapat masuk ke form isian barang keluar.	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah tersimpan.	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik	Data yang telah	Tombol reset	[X]

tombol batal	diinputkan pada *filed* menjadi kosong.	sesuai yang diharapkan.	Diterima [ ] Ditolak
Klik tombol Cetak	Data baranag keluar dapat langsung dicetak dan akan menjadi sebuah laporan	Tombol print sesuai dengan yang diharapkan.	[X] Diterima [ ] Ditolak

7. Pengujian Pengolahan Data Supplier

Pengujian ini adalah pengujian dengan memasukan pengisian data supplier dengan form data supplier ke dalam database.

Tabel 5.11 Pengujian Pengolahan Data Supplier

Kasus dan Hasil Uji Pengisian Data Supplier (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data supplier	Dapat masuk ke form isian supplier.	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak

	tersimpan.		
Klik tombol batal	Data yang telah diinputkan pada *filed* menjadi kosong.	Tombol reset sesuai yang diharapkan.	[X] Diterima [ ] Ditolak

8. Pengujian Pengolahan Data Customer

Pengujian ini adalah pengujian dengan memasukan pengisian data *customer* dengan form data *customer* ke dalam database.

Tabel 5.12 Pengujian Pengolahan Data *Customer*

Kasus dan Hasil Uji Pengisian Data *Customer* (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data *customer*	Dapat masuk ke form isian *customer*.	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah tersimpan.	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol batal	Data yang telah diinputkan pada *filed* menjadi	Tombol reset sesuai yang diharapkan.	[X] Diterima [ ] Ditolak

	kosong.		

9. Pengujian Pengolahan Data Kategori

Pengujian ini adalah pengujian dengan memasukan pengisian data kategori dengan form data kategori ke dalam database.

Tabel 5.13 Pengujian Pengolahan Data Kategori

Kasus dan Hasil Uji Pengisian Data Supplier (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Isi data kategori	Dapat masuk ke form isian kategori.	Pengisian sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke database dan akan memunculkan pesan data telah tersimpan.	Tombol simpan sesuai yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol batal	Data yang telah diinputkan pada *filed* menjadi kosong.	Tombol reset sesuai yang diharapkan.	[X] Diterima [ ] Ditolak

4.5.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus sample uji yang telah dilakukan memberikan kesimpulan bahwa perangkat lunak ini dapat

digunakan dengan baik, namun pengujian tersebut dapat dikatakan belum sempurna, karena hanya dilakukan pada satu sisi pengujian. Dari semua yang telah dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lain dalam aplikasi sistem informasi penjualan, pembelian dan persediaan barang.