BAB IV

HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Perancangan sistem merupakan suatu kegiatan pengembangan prosedur dan proses yang sedang berjalan untuk menghasilkan suatu sistem yang baru, atau memperbaharui sistem yang ada untuk meningkatkan efektifitas kerja agar dapat memenuhi hasil yang digunakan dengan tujuan memanfaatkan teknologi dan fasilitas yang tersedia. Pada bagian ini penyusun akan memberikan usulan yang merupakan sistem informasi secara komputerisasi yang diharapkan akan membantu dan mempermudah pekerjaan.

4.1.1 Tujuan Perancangan Sistem

Perancangan sistem bertujuan untuk memberikan gambaran yang jelas dan rancang bangun yang sesuai dengan kebutuhan pemakai sistem itu sendiri. Perancangan sistem dilakukan apabila tahap analisis sistem telah dilakukan. Maka untuk selanjutnya seorang analis sistem merancang bagaimana membentuk sistem yang baru ataupun memperbaharui sistem yang lama. Tahap inilah yang dinamakan dengan istilah dari perancangan sistem.

Adapun tujuan perancangan sistem yang diusulkan yaitu untuk menghasilkan suatu rancangan sistem yang dapat membantu proses pengolahan data dan transaksi dalam menangani proses penjualan barang dan saja yang semuanya terintegrasi dengan *database* dan aplikasi tersebut dapat mencapai sasaran kebutuhan dan tujuan dari perusahaan yaitu kepuasan pelanggan.

4.1.2 Gambaran Umum Sistem yang Diusulkan

Gambaran umum tentang sistem yang diusulkan pada proses perancangan ini adalah suatu sistem yang merubah seluruh proses bisnis yang manual menjadi terkomputerisasi. Proses bisnis yang dimaksud adalah pengolahan data barang, jasa, pegawai, hingga proses penggajian ditangani oleh sistem informasi ini. Sistem ini diharapkan dapat membantu efektifitas dan optimalisasi waktu pada seluruh proses bisnis yang berjalan serta dapat membantu mengatasi permasalahan yang ada dan yang paling penting dapat menghasilkan informasi yang cepat, tepat dan akurat.

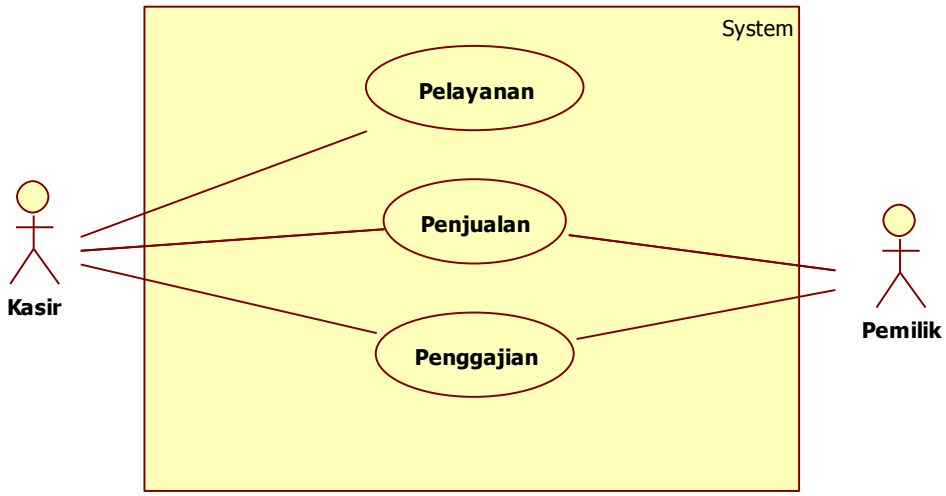
4.1.3 Perancangan Prosedur yang Diusulkan

Perancangan prosedur merupakan awal dari pembuatan sistem informasi yang akan dibuat, dimana dapat dilihat proses-proses apa saja yang nantinya diperlukan dalam pembuatan suatu sistem informasi. Sedangkan perancangan prosedur yang diusulkan merupakan tahap untuk memperbaiki atau meningkatkan

efisiensi kerja. Tahap perancangan sistem yang digambarkan merupakan tahap perancangan untuk membangun suatu sistem dan mengkonfigurasikan komponen-komponen perangkat lunak dan perangkat keras sehingga menghasilkan sistem informasi yang baik. Sistem informasi yang dirancang tersebut menjadi satu komponen. Tahapan perancangan prosedur ini akan dijelaskan dengan menggunakan pemodelan sistem informasi berorientasi objek dengan *Unified Modeling Language* (UML).

4.1.3.1 Use case Diagram

Diagram *use case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, *use case* bisnis, dan pekerja-pekerja bisnis untuk suatu organisasi atau perusahaan. Diagram ini memberikan model lengkap tentang apa yang dilakukan perusahaan, siapa yang ada di dalam perusahaan, serta siapa yang berada di luar perusahaan. Berikut adalah gambar model diagram *use case* bisnis yang diusulkan :



Gambar 4. 1 *Use case* diagram yang diusulkan

1. Identifikasi *Use case*

Identifikasi *use case* yang diusulkan adalah sebagai berikut :

Tabel 4. 1 Identifikasi *use case* sistem yang diusulkan

Kode	*Use case*	Deskripsi
	Pelayanan	Fungsionalitas untuk mengolah data transaksi jasa.
	Penjualan	Fungsionalitas untuk melakukan pengolahan data dan transaksi barang
	Penggajian	Fungsionalitas untuk melakukan pengolah data dan pencetakan laporan gaji.

2. Identifikasi Aktor

Aktor yang berperan dalam menjalankan sistem adalah sebagai berikut :

Tabel 4. 2 Identifikasi aktor sistem diusulkan

No	Aktor	Deskripsi
1	Kasir	Aktor yang berperan menjalankan proses transaksi di salon
2	Pemilik	Aktor yang mempunyai hak untuk mengolah data master, pengecekan dan pembuatan laporan.

4.2.3.1 Skenario *Use case*

Skenario *Use case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan dari skenario *use case* pada sistem yang diusulkan adalah sebagai berikut :

4.2.3.2.1 Skenario *use case* pelayanan

Berikut skenario *use case* pelayanan *user* yang diusulkan :

Tabel 4. 3 Skenario *use case* pelayanan

Identifikasi	
Judul	Pelayanan
Kode *Use case*	A
Deskripsi *use case*	Aktor Melakukan Pelayanan
Aktor	Kasir
Trigger	Aktor mengklik menu transaksi jasa
Kondisi awal	Aplikasi menampilkan form transaksi jasa
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengklik menu transaksi jasa.	
	2. Sistem membuka form transaksi jasa sesuai permintaan aktor.
	3. Aktor menginputkan permintaan jasa sesuai permintaan pelanggan.
	4. Sistem menyimpan data transaksi jasa sesuai dengan pelayanan.
Kondisi akhir skenario utama	Data transaksi jasa telah tersimpan.

4.2.3.2.2 Skenario *use case* penjualan

Berikut skenario *use case* memproses transaksi penjualan yang diusulkan :

Tabel 4. 4 Skenario *use case* memproses transaksi penjualan

Identifikasi	
Judul	Penjualan
Kode *Use case*	B
Deskripsi *use case*	Aktor memproses penjualan
Aktor	Kasir
Trigger	Aktor mengklik menu transaksi barang
Kondisi awal	Aplikasi menampilkan form transaksi barang
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengklik menu transaksi barang.	
	2. Sistem form transaksi barang sesuai dengan permintaan aktor
	3. Sistem mencatat transaksi barang sesuai dengan permintaan aktor
Kondisi akhir skenario utama	Data transaksi barang telah tersimpan.

4.2.3.2.3 Skenario *use case* penggajian

Berikut skenario *use case* memproses penggajian yang diusulkan :

Tabel 4. 5 Skenario *use case* memproses transaksi penggajian

Identifikasi	
Judul	Memproses Penggajian
Kode *Use case*	C
Deskripsi *use case*	Aktor Penggajian
Aktor	Pemilik
Trigger	Aktor mengklik laporan penggajian dari menu utama
Kondisi awal	Aplikasi menampilkan form laporan penggajian
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengklik menu laporan penggajian.	
	2. Sistem membuka form laporan penggajian sesuai dengan permintaan aktor
	3. Aktor memasukan permintaan laporan penggajian.

	4. Sistem memproses laporan penggajian sesuai dengan permintaan aktor.
	5. Sistem menampilkan laporan penggajian sesuai dengan permintaan aktor.
Kondisi akhir skenario utama	Laporan penggajian tampil.

4.2.3.2 *Activity Diagram*

Activity diagram (diagram aktivitas) adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Berikut *activity diagram* pada sistem yang diusulkan :

4.2.3.2.1 Activity Diagram Pelayanan

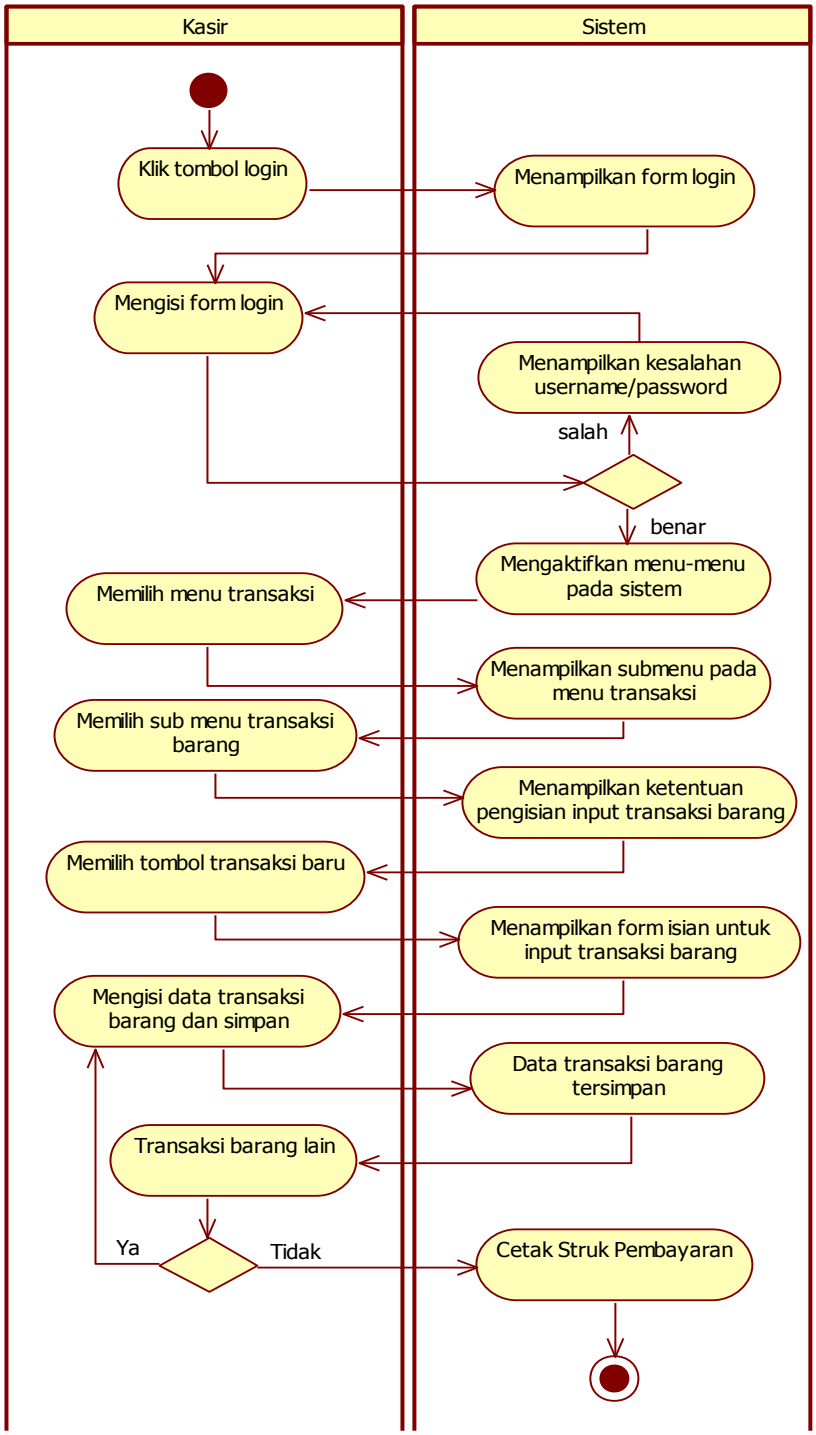
Berikut *activity diagram* transaksi pelayanan yang diusulkan :



Gambar 4. 2 *Activity diagram* pelayanan

4.2.3.2.2 Activity Diagram penjualan

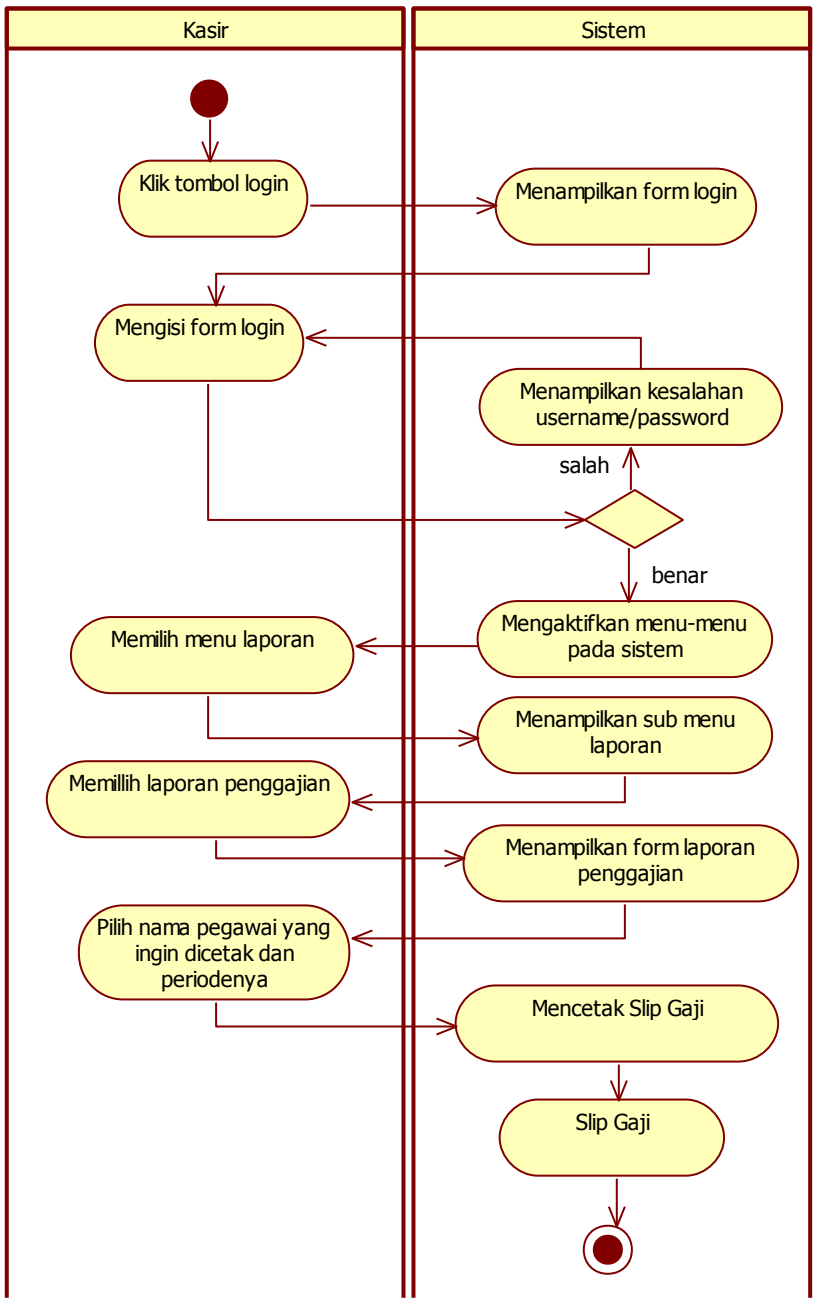
Berikut *activity diagram* transaksi penjualan yang diusulkan :



Gambar 4. 3 *Activity diagram* transaksi penjualan

4.2.3.2.3 Activity Diagram Penggajian

Berikut *activity diagram* penggajian yang diusulkan :



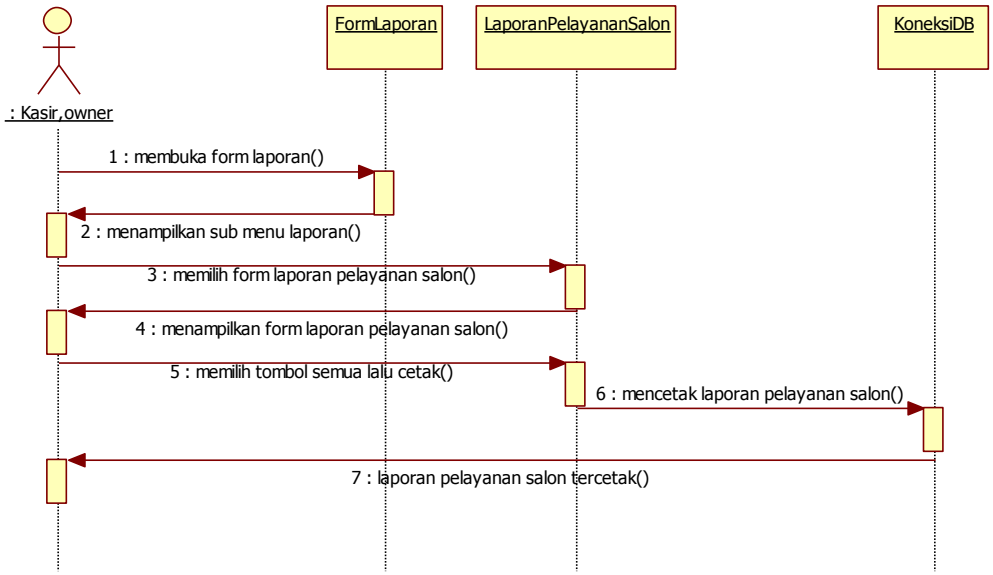
Gambar 4. 4 *Activity diagram* laporan penggajian

4.2.3.3 *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya *sequence diagram* digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antar object, juga interaksi antar objek, dan menunjukkan sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat bernama, pesan diwakili oleh garis dengan tanda panah, dan waktu yang ditunjukkan dengan proses vertikal. Berikut adalah *sequence diagram* yang diusulkan :

4.2.3.4.1 *Sequence Diagram* pelayanan

Berikut *sequence diagram* pelayanan yang diusulkan :



Gambar 4. 5 *Sequence diagram* pelayanan

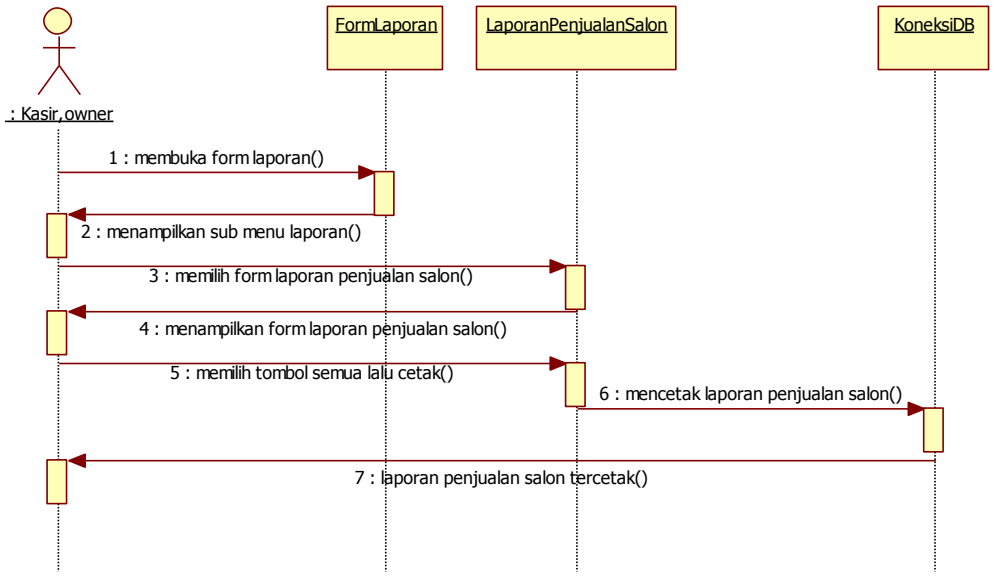
Keterangan:

Pada gambar 4.5 dapat dilihat bahwa ada 4 kelas yang saling berinteraksi, yakni:

1. Login
2. Menu Utama
3. Transaksi Saja
4. DBManager

4.2.3.4.2 *Sequence Diagram* Penjualan

Berikut *sequence diagram* penjualan yang diusulkan :



Gambar 4. 6 *Sequence diagram* penjualan

Keterangan:

Pada gambar 4.6 dapat dilihat bahwa ada 4 kelas yang saling berinteraksi, yakni:

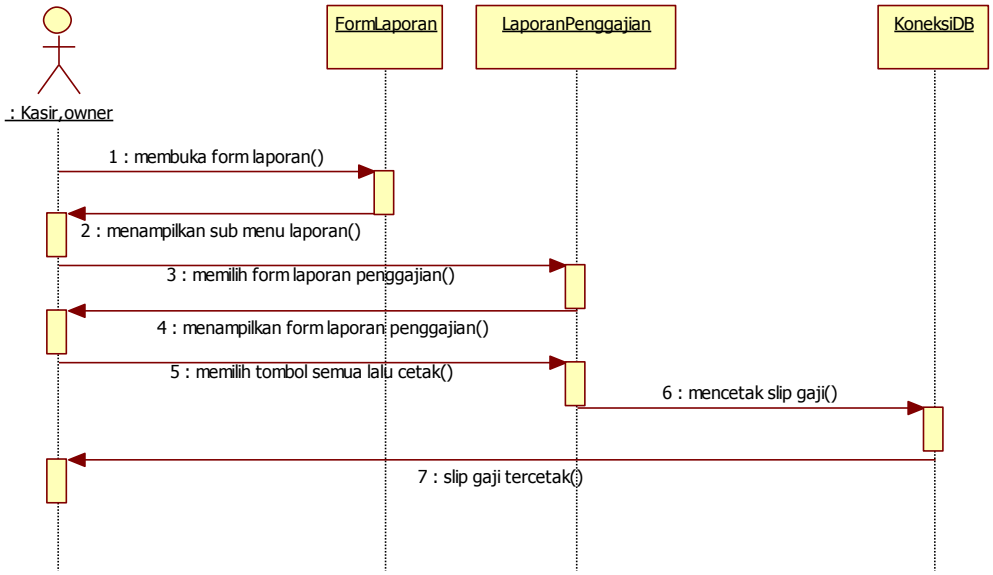
1. Login
2. Menu Utama

3. Transaksi Barang

4. DBManager

4.2.3.4.3 *Sequence Diagram* penggajian

Berikut *sequence diagram* penggajian yang diusulkan :



Gambar 4. 7 *Sequence diagram* penggajian

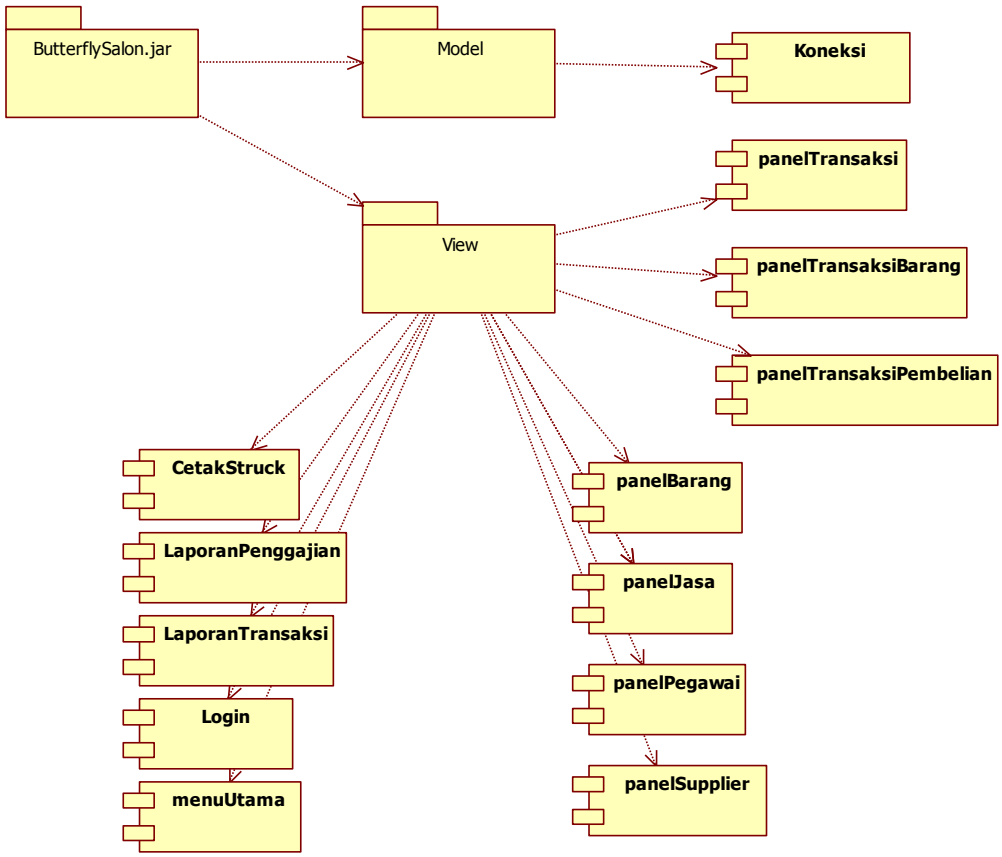
Keterangan:

Pada gambar 4.7 dapat dilihat bahwa ada 4 kelas yang saling berinteraksi, yakni:

1. Login
2. Menu Utama
3. Form Penggajian
4. DBManager

4.2.3.5 *Component Diagram*

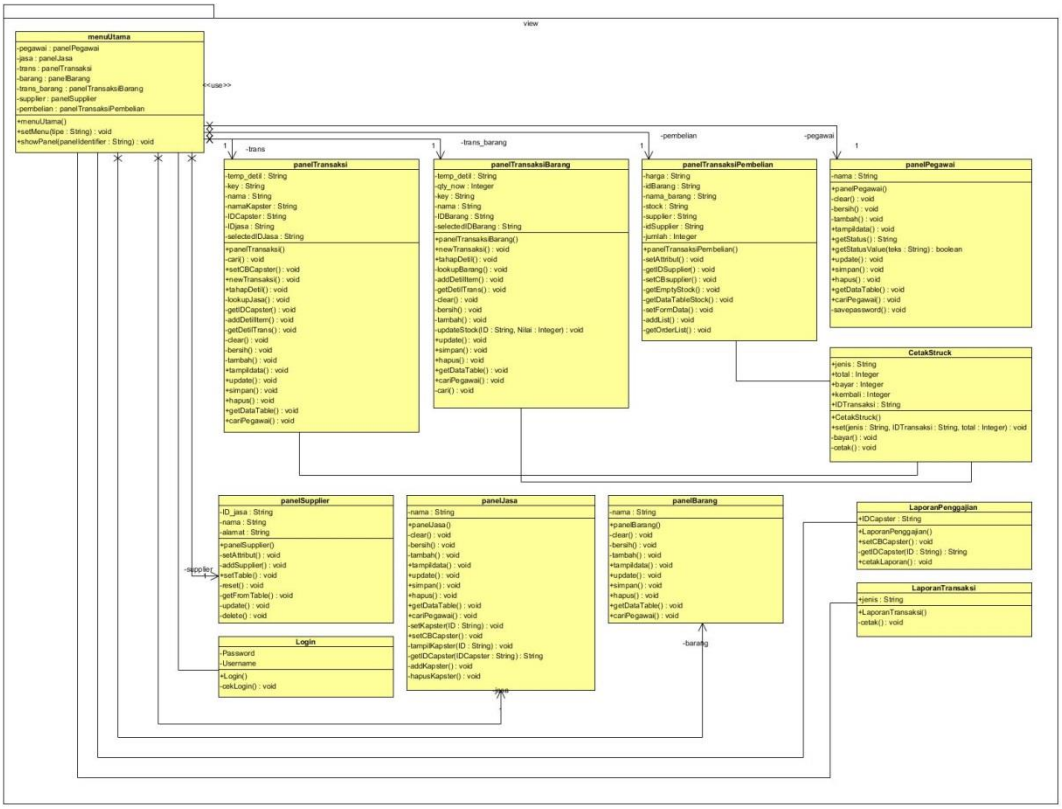
Component diagram menggambarkan struktur dan hubungan antar komponen perangkat lunak, termasuk ketergantungan (*dependency*) di antaranya. *Component* piranti lunak adalah modul berisi *code*, baik *berisi source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Berikut ini adalah *component diagram* yang dibutuhkan :



Gambar 4. 8 *Component diagram*

4.2.3.4 *Class Diagram*

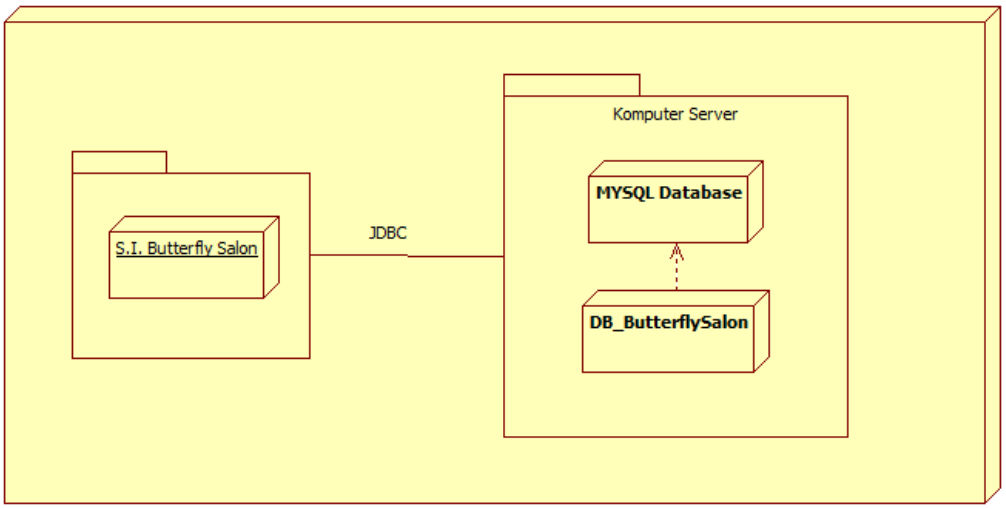
Menggambarkan struktur statis *class* di dalam sistem. *Class* merepresentasikan sesuatu yang ditangani oleh sistem. Dengan melihat karakteristik sistem informasi salon dari penjualan dan penggajian, maka dapat dibuat *class diagram* Berikut *class diagram* yang diusulkan pada butterfly salon:



Gambar 4. 9 *Class diagram*

4.2.3.5 *Deployment Diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. *Deployment diagram* berhubungan dengan diagram komponen dimana *deployment diagram* memuat satu atau lebih komponen-komponen.



Gambar 4. 10 *Deployment diagram* sistem yang diusulkan

4.1.4 Perancangan Antar Muka

Perancangan antar muka dirancang untuk memudahkan pemakai (*user*) dalam mengakses informasi-informasi yang dibutuhkan. Perancangan antar muka meliputi perancangan tampilan input dan output. Adapun perancangan antar muka adalah sebagai berikut :

4.1.4.1 Struktur Menu

Halaman ini merupakan halaman menu utama, sebelum masuk dan mengakses ke sistem terlebih dahulu *user* melakukan login. Disini terdapat 2 *user* yang dapat menggunakan sistem informasi ini yaitu pemilik dan kasir. Kedua *user* tersebut memiliki hak akses yang berbeda-beda sesuai dengan proses bisnis yang dirancang.



Gambar 4. 11 Desain tampilan untuk *menu utama*

Keterangan :

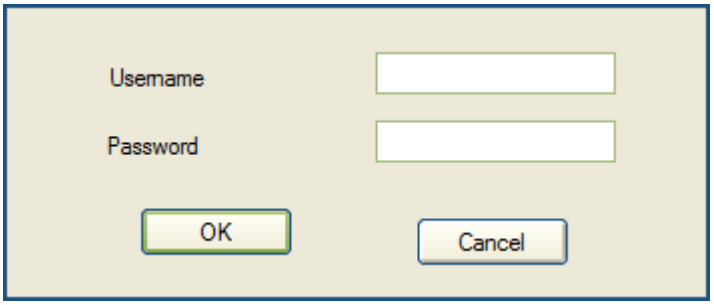
a. Menu file berisi logout untuk berganti user dan exit untuk keluar dari aplikasi
b. Menu Data berisi Pegawai, Barang dan Jasa untuk mengatur data master
c. Menu Transaksi berisi barang dan jasa untuk melakukan transaksi

d. Menu Laporan berisi Laporan barang, Laporan jasa, Laporan Transaksi Barang, Laporan Transaksi Jasa, Penggajian untuk menampilkan laporan sesuai dengan permintaan.
e. Menu About Berisi Aplikasi dan Pembuat untuk menampilkan informasi Aplikasi maupun pembuat.

4.1.4.2 Perancangan Input

1. Rancangan *login user*

Halaman ini merupakan halaman login *user*, sebelum masuk dan mengakses ke sistem terlebih dahulu *user* melakukan login. Disini terdapat 2*user* yang dapat menggunakan sistem informasi ini yaitu pemilik dan kasir. Kedua *user* tersebut memiliki hak akses yang berbeda-beda sesuai dengan proses bisnis yang dirancang.



Gambar 4. 12 Desain tampilan untuk *user* login

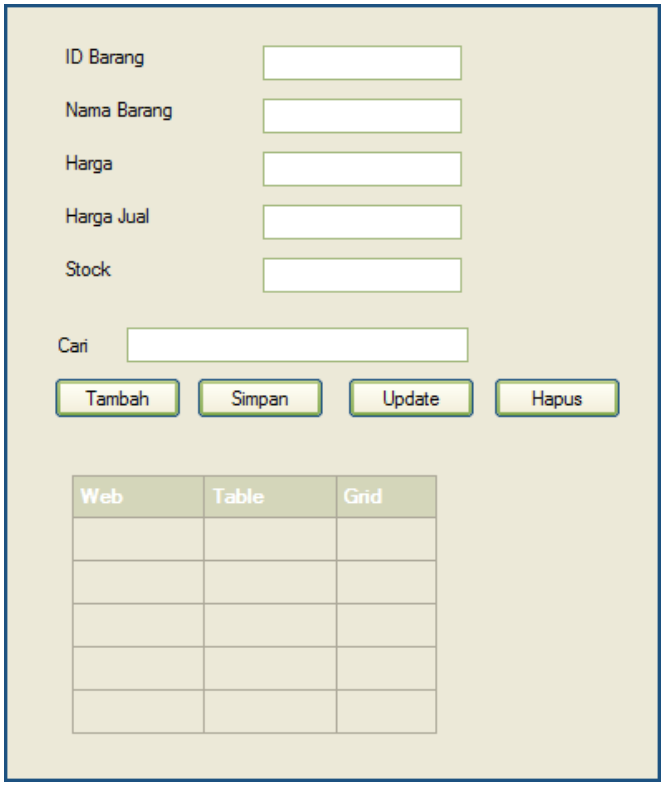
Keterangan :

a. *Textbox Username* dan *password* digunakan untuk mengisi data dari *user*.

b. *Button* Ok digunakan untuk melakukan proses masuk ke dalam sistem sesuai dengan hak akses masing-masing *user*.

2. Rancangan data master barang

Halaman ini merupakan halaman data master barang, pada halaman ini user dapat menambahkan, mengupdate dan menhapus data barang.



Gambar 4. 13 Desain tampilan untuk data master barang

Keterangan :

a. Textbox ID barang digunakan untuk mengisi kode indentitas barang.
b. Textbox nama barang digunakan untuk mengisi nama barang.
c. Textbox harga digunakan untuk mengisi harga dasar barang
d. Textbox harga jual digunakan untuk mengisi harga jual dari barang
e. Textbox stok digunakan untuk mengisi jumlah stok dari barang
f. Tombol tambah digunakan untuk memulai menambahkan barang
g. Tombol simpan digunakan untuk menyimpan data barang baru yang ditambahkan
h. Tombol update digunakan untuk menyimpan perubahan data
i. Tombol hapus digunakan untuk menghapus data

3. Rancangan data master jasa

Halaman ini merupakan halaman data master jasa, pada halaman ini user dapat menambahkan, mengupdate dan menhapus data jasa.



Gambar 4. 14 Desain tampilan untuk data master jasa

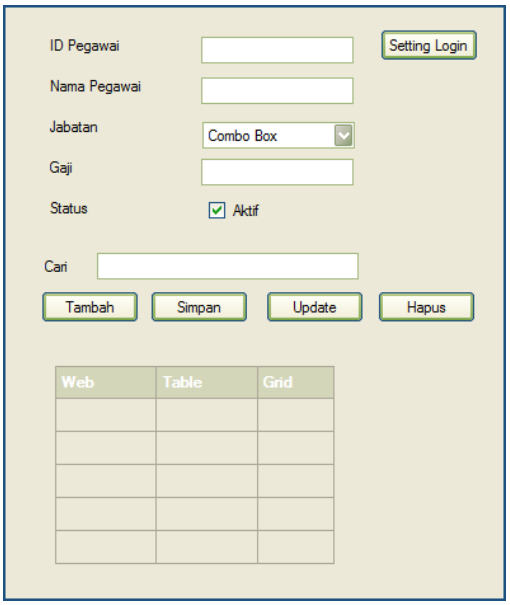
Keterangan :

a. Textbox ID jasa digunakan untuk mengisi kode indentitas jasa.
b. Textbox nama digunakan untuk mengisi nama jasa.
c. Textbox harga digunakan untuk mengisi harga jasa
d. Textbox komisi digunakan untuk mengisi nilai komisi dari jasa
e. Tombol tambah digunakan untuk memulai menambahkan jasa
f. Tombol simpan digunakan untuk menyimpan data jasa baru yang ditambahkan
g. Tombol update digunakan untuk menyimpan perubahan data

h. Tombol hapus digunakan untuk menghapus data

4. Rancangan data master pegawai

Halaman ini merupakan halaman data master pegawai, pada halaman ini user dapat menambahkan, mengupdate dan menhapus data pegawai.



Gambar 4. 15 Desain tampilan untuk data master pegawai

Keterangan :

a. Textbox ID pegawai digunakan untuk mengisi kode indentitas pegawai.
b. Textbox nama pegawai digunakan untuk mengisi nama pegawai.
c. ComboBox jabatan digunakan untuk mengisi jabatan pegawai
d. Textbox harga jual digunakan untuk mengisi harga jual dari barang
e. Textbox gaji digunakan untuk mengisi gaji perbulan pergawai
f. Cekbox status digunakan untuk status keaktifan pergawai

g. Tombol tambah digunakan untuk memulai menambahkan pegawai

h. Tombol simpan digunakan untuk menyimpan data pegawai baru

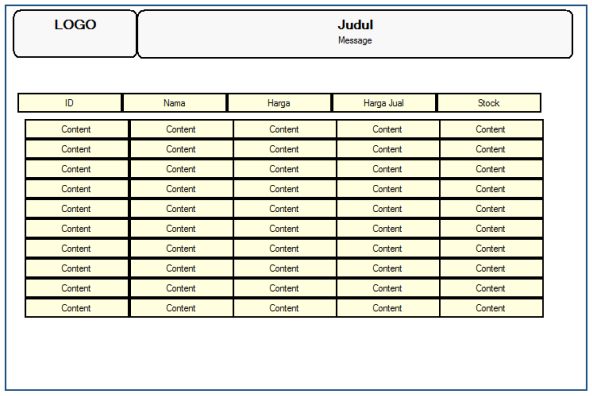
i. Tombol update digunakan untuk menyimpan perubahan data

j. Tombol hapus digunakan untuk menghapus data

4.1.4.3 Perancangan Output

1. Rancangan laporan

Halaman ini merupakan halaman untuk mencetak laporan transaksi maupun transaksi barang. Pada halaman ini user terlebih dahulu harus menentukan tanggal dari data yang akan ditampilkan.



Gambar 4. 16 Desain tampilan untuk laporan transaksi

Keterangan :

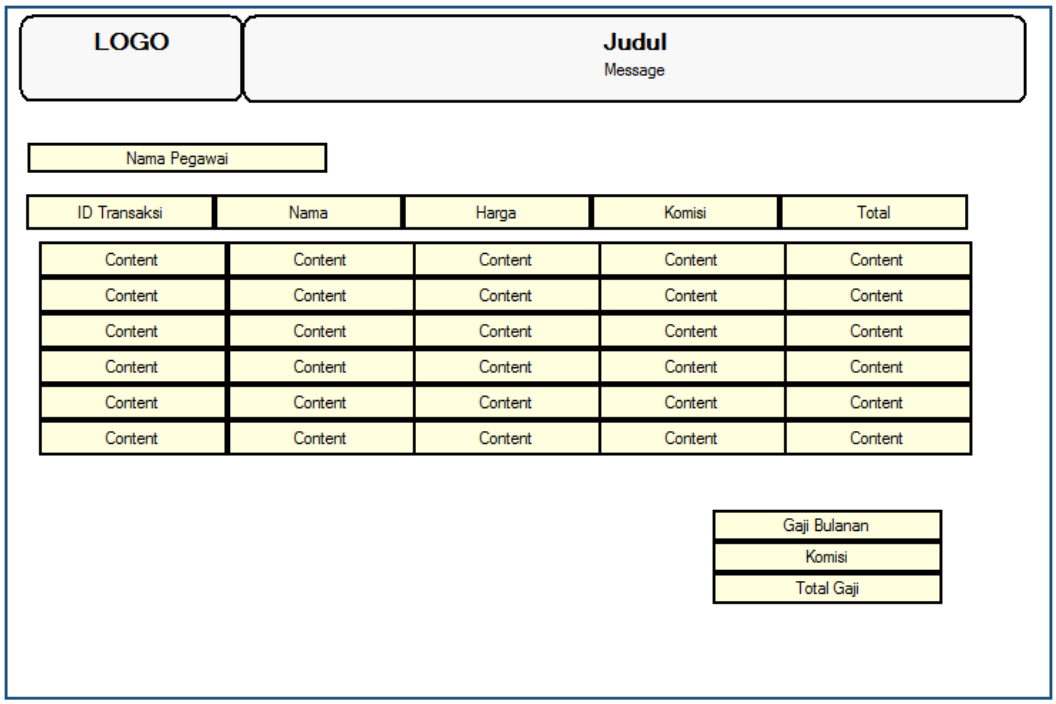
a. Kotak logo akan digunakan untuk menampilkan logo.

b. Kotak Judul akan digunakan untuk menampilkan judul.

c. Kolom dan baris akan digunakan untuk menampilkan data.

2. Rancangan laporan penggajian

Halaman ini merupakan halaman untuk mencetak laporan penggajian dari pegawai. Pada halaman ini user dapat memilih ingin menampilkan seluruh gaji dari pegawai atau satu-persatu. Gaji pegawai yang akan ditampilkan terdiri dari gaji bulanan dan komisi.



Gambar 4. 17 Desain tampilan untuk laporan penggajian

Keterangan :

a. Kotak logo akan digunakan untuk menampilkan logo.

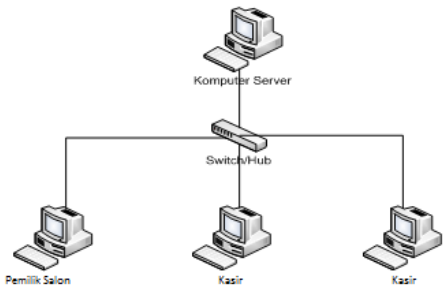
b. Kotak Judul akan digunakan untuk menampilkan judul.

c. Kolom dan baris akan digunakan untuk menampilkan data.

d. Kotak gaji bulanan akan digunakan untuk menampilkan.
e. Kotak total komisi akan digunakan untuk menampilkan total komisi.
f. Kotak total gaji akan digunakan untuk menampilkan total gaji.

4.1.5 Perancangan Arsitektur Jaringan

Arsitektur jaringan yang digunakan adalah sistem klien server mempunyai dua komponen utama yaitu komputer server dan komputer klien. Server merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer client dan bertindak sebagai server *database* yang menyimpan data. Klien yaitu komputer yang melakukan pengiriman permintaan-permintaan data pada server kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya.



Gambar 4. 18 Arsitektur jaringan yang diusulkan

Gambar 4.18 menunjukkan arsitektur jaringan. Disini penulis menggunakan topologi star dengan alasan jika salah satu komputer klien rusak,

komputer yang lain masih dapat diakses dan kontrol menagement lebih mudah karena semuanya terpusat ke server.

4.2 Implementasi

Setelah sistem dirancang dan didesain secara rinci, maka selanjutnya menuju tahap implementasi. Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga *user* tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukkan kepada pembuat sistem.

4.2.1 Implementasi Perangkat Lunak

Spesifikasi minimum perangkat lunak untuk menjalankan sistem in*form*asi ini adalah sebagai berikut:

1. Sistem Operasi Windows XP
2. *Java Development Kit* (JDK) 1.6 dan *Java Runtime Enviroment* (JRE) 1.6
3. XAMPP versi 1.7.1 yang terdiri dari :
   a. PHP Versi 5.2.9
   b. MySQL 5.0.51
   c. Apache

4.2.2 Implementasi Perangkat Keras

Kebutuhan minimum perangkat keras (*hardware*) yang diperlukan untuk mengimplementasikan program aplikasi yang dibuat adalah perangkat keras komputer PC kompatibel dengan spesifikasi yang disebutkan dibawah ini. Semakin tinggi spesifikasi komputer yang digunakan untuk menjalankan aplikasi, maka akan semakin baik. Kebutuhan minimumnya yaitu :

1. Server
    a. Processor 1,2 GHz
    b. Memory 2GB
    c. Harddisk 500GB
2. Klien
    a. Processor 1,2 Ghz
    b. Memory 512 MB
    c. Harddisk 40 GB
3. Jaringan Komputer
    a. Switch 4 port
    b. Kabel Ethernet Cat 5e

4.2.3 Implementasi Basis Data

Implementasi basis data diambil berdasarkan kebutuhan data untuk aplikasi, secara fisik, implementasi basis data diimplementasikan menggunakan perangkat lunak MySQL server, tabel-tabel berikut menggambarkan struktur tabel yang diimplementasikan pada basis data.

1. Tabel Pegawai

```
CREATE TABLE IF NOT EXISTS `pegawai` (
  `id_pegawai` varchar(50) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `jabatan` varchar(50) NOT NULL,
  `gaji_bulanan` int(11) NOT NULL,
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `status` varchar(50) NOT NULL,
  PRIMARY KEY (`id_pegawai`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Produk Barang

```
CREATE TABLE IF NOT EXISTS `produk_barang` (
  `id_barang` varchar(255) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `harga` int(11) NOT NULL,
  `harga_jual` int(11) NOT NULL,
  `stock` int(11) NOT NULL,
  PRIMARY KEY (`id_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel Produk Jasa

```
CREATE TABLE IF NOT EXISTS `produk_jasa` (
  `id_jasa` varchar(255) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `harga_jual` int(11) NOT NULL,
  `komisi` int(11) NOT NULL,
  PRIMARY KEY (`id_jasa`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel trans

```
CREATE TABLE IF NOT EXISTS `trans` (
  `id_trans` varchar(255) NOT NULL,
  `tanggal` date NOT NULL,
  `id_kasir` varchar(20) NOT NULL,
  PRIMARY KEY (`id_trans`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel trans_barang

```
CREATE TABLE IF NOT EXISTS `trans_barang` (
  `id_trans` varchar(255) NOT NULL,
  `tanggal` date NOT NULL,
  `id_kasir` varchar(20) NOT NULL,
  PRIMARY KEY (`id_trans`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel trans_detil

```
CREATE TABLE IF NOT EXISTS `trans_detil` (
  `id_detil` int(11) NOT NULL AUTO_INCREMENT,
  `id_trans` varchar(255) NOT NULL,
  `id_jasa` varchar(20) NOT NULL,
  `id_capster` varchar(20) NOT NULL,
  PRIMARY KEY (`id_detil`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=106 ;
```

7. Tabel trans_detil_barang

```
CREATE TABLE IF NOT EXISTS `trans_detil_barang` (
  `id_detil` int(11) NOT NULL AUTO_INCREMENT,
  `id_trans` varchar(255) NOT NULL,
  `id_barang` varchar(20) NOT NULL,
  `qty` varchar(20) NOT NULL,
  PRIMARY KEY (`id_detil`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=86 ;
```

8. Tabel list_belanja

```
CREATE TABLE IF NOT EXISTS `list_belanja` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `id_barang` varchar(50) NOT NULL,
  `id_supplier` varchar(50) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `status` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=28 ;
```

9. Tabel produk_jasa_kapster

```
CREATE TABLE IF NOT EXISTS `produk_jasa_kapster` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `id_jasa` varchar(50) NOT NULL,
  `id_kapster` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=229 ;
```

10. Tabel supplier

```
CREATE TABLE IF NOT EXISTS `supplier` (
  `id_supplier` varchar(50) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  PRIMARY KEY (`id_supplier`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.2.4 Implementasi Antar Muka

Dalam *Netbeans IDE* , implementasi antarmuka dilakukan dengan sebuah *form* yang berekstensi *form*. Setiap halaman dalam perangkat lunak dibuka *file program* yang ditulis dengan berektensi *form*. Berikut ini adalah implementasi dari setiap halaman yang dibuat :

Tabel 4.6 Implementasi Halaman Utama

Menu	Deskripsi	Nama File
Master Barang	Menu untuk melihat dan melakukan penambahan atau perubahan data barang	panelBarang.java
Master Jasa	Menu untuk melihat dan melakukan penambahan atau perubahan data jasa	panelJasa.java
Master Pegawai	Menu untuk melihat dan melakukan penambahan atau perubahan data pegawai	panelPegawai.java
Transaksi Jasa	Menu untuk melakukan transaksi jasa	panelTransaksi.java
Transaksi Barang	Menu untuk melakukan transaksi jasa	panelTransaksiBarang.java
Laporan Barang	Menu untuk menampilkan data barang	lapBarang.java
Laporan Jasa	Menu untuk menampilkan jasa barang	lapJasa.java
Laporan	Menu untuk menampilkan laporan	laporanTrans.java

Transaksi Barang	transaksi barang	
Laporan Transaksi Jasa	Menu untuk menampilkan laporan transaksi barang	laporanTrans.java
Laporan Penggajian	Menu untuk menampilkan penggajian pegawai	lapPenggajian.java

4.2.5 Implementasi Instalasi Program

Berikut adalah implementasi instalasi program untuk menggunakan program sistem informasi penjualan dan penggajian butterfly salon. Langkah-langkahnya sebagai berikut :

1. Menjalankan Xampp

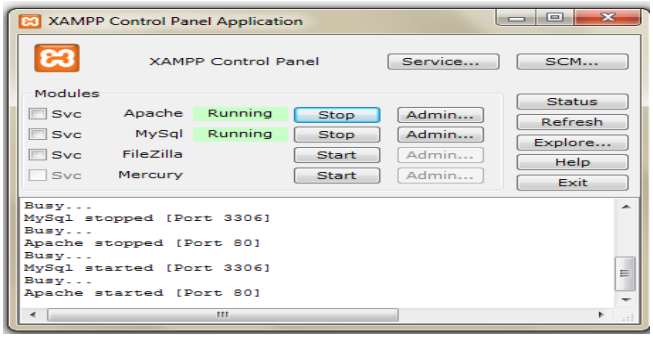
   Pertama buka xampp pada *dekstop* dengan cara lakukan klik 2 kali pada *icon* Xampp



Gambar 4.19 Menjalankan Xampp

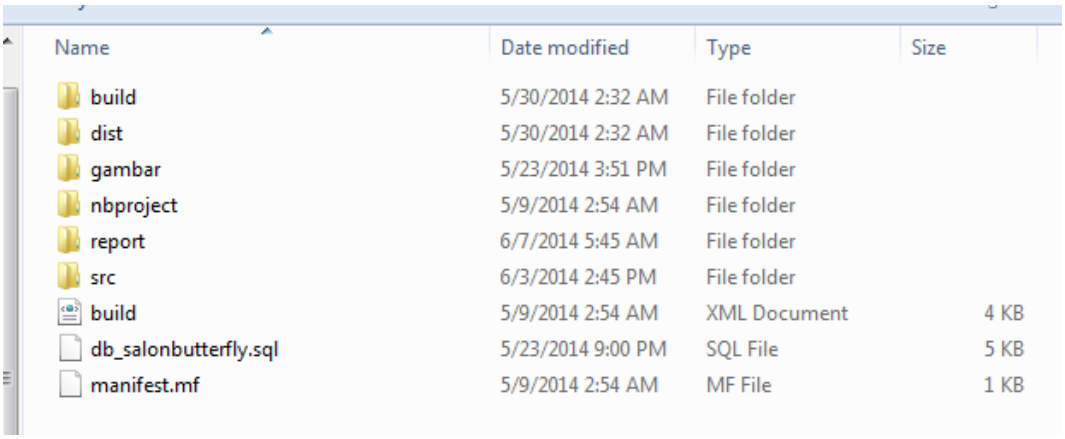
2. Pengaktifan MySql

Setelah itu maka akan muncul *Xampp Control Panel Aplication* seperti gambar berikut. Kemudian klik tombol Start pada baris Mysql, ini adalah proses pengaktifan fitur MySql pada komputer kita.



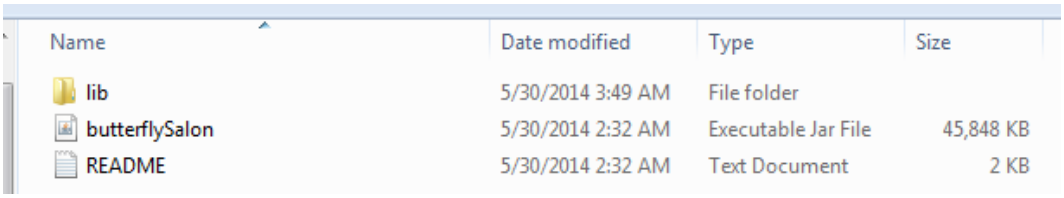
Gambar 4.20 Xampp Control Panel Aplication

3. Membuka *folder* penyimpanan program.



Gambar 4.21 Membuka folder penyimpanan program

4. Membuka *folder dist* pada folder penyimpanan program.



Gambar 4.22 Membuka folder dist

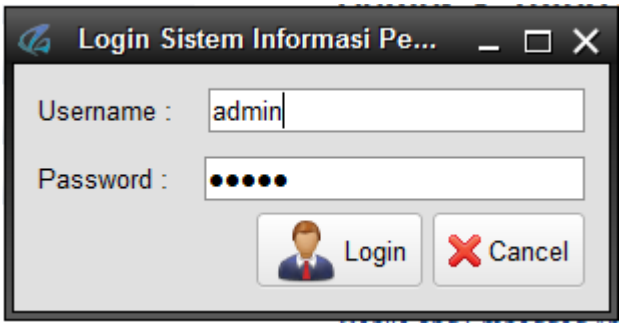
5. Langkah berikutnya yaitu klik kanan *file* butterflySalon.jar kemudian pilih create shortcut. Pindahkan *file* shortcut yang sudah dibuat tadi ke dekstop untuk mempermudah apabila ingin menjalankan program.

4.2.6 Penggunaan Program

Penggunaan program bertujuan untuk memberikan informasi tentang bagaimana tata cara dalam menjalankan dan menggunakan program sistem informasi penjualan dan penggajian bagi pengguna atau *user*.

1. Login

   Halaman ini merupakan halaman login *user*, sebelum masuk dan mengakses ke sistem terlebih dahulu *user* melakukan login. Disini terdapat 2 *user* yang dapat menggunakan sistem informasi ini yaitu pemilik dan kasir. Kedua *user* tersebut memiliki hak akses yang berbeda-beda sesuai dengan proses bisnis yang dirancang.



Gambar 4.23 *Form* login

Keterangan :

a. *Textbox Username* dan password digunakan untuk mengisi data dari *user*.
b. *Button* Login digunakan untuk melakukan proses masuk ke dalam sistem sesuai dengan hak akses masing-masing *user*.

2. Menu Utama

Halaman ini merupakan *form* utama yang mana terdapat pilihan tombol menu yang dapat dipilih oleh pengguna sesuai dengan hak akses masing-masing *user*.

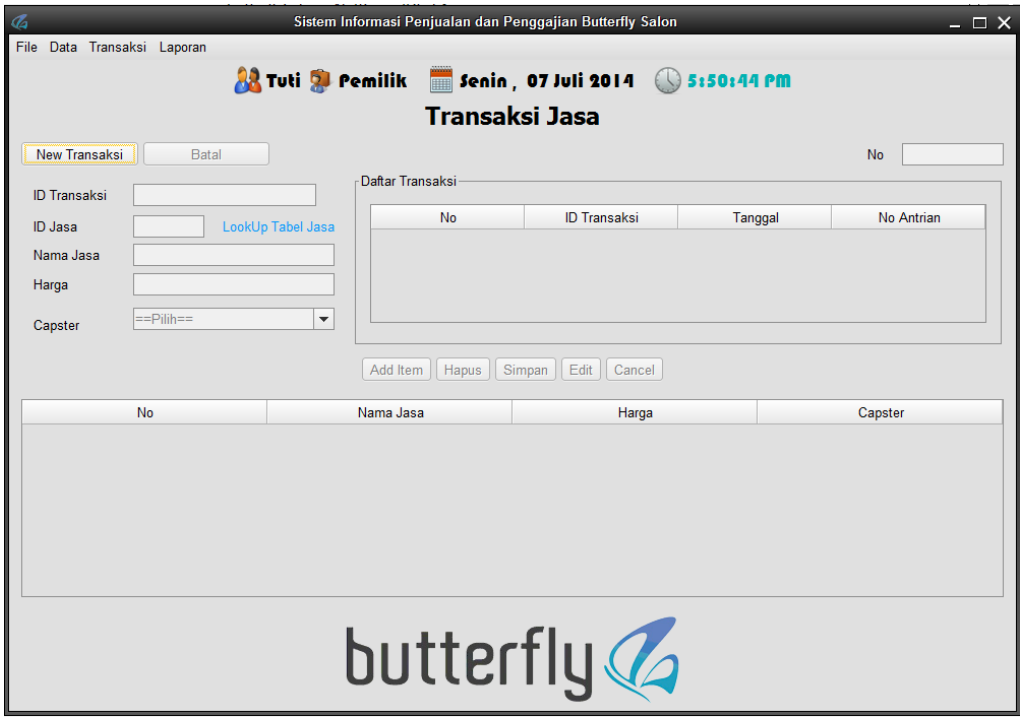
Berikut hak akses masing-masing dari *user* :

a. Pemilik

Pemilik salon dapat mengakses seluruh tombol menu yang terdapat pada *form* utama.

b. Kasir

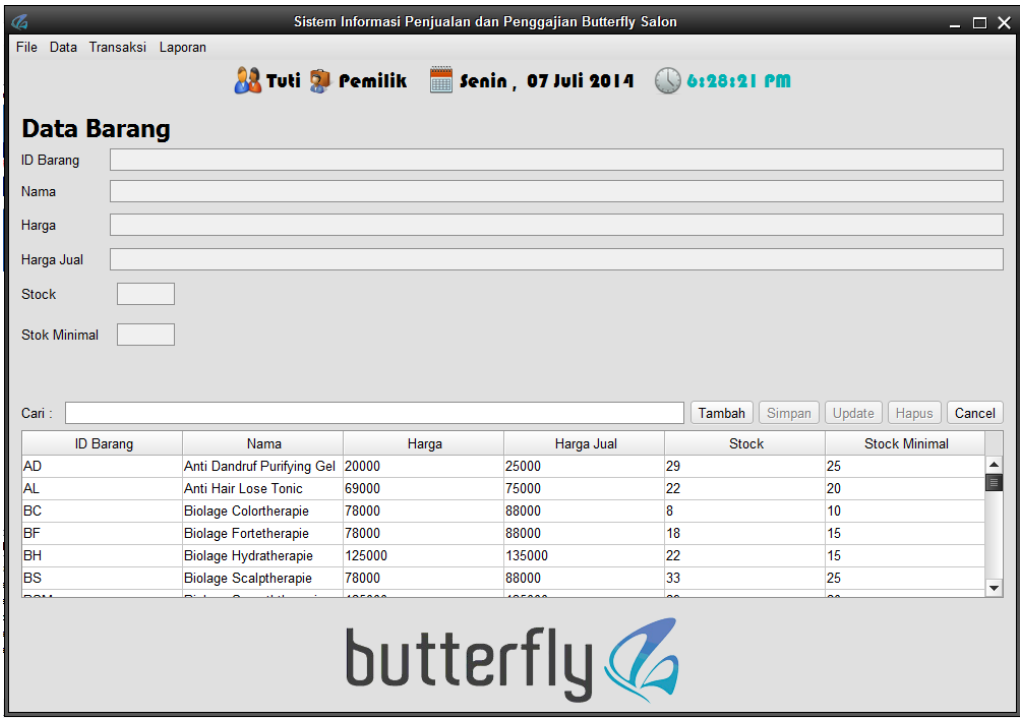
Kasir dapat mengakses Transaksi untuk melakukan transaksi.



Gambar 4.24 Menu Utama

3. Data Master Barang

Halaman ini merupakan *form* master barang. Hanya pemilik salon yang memiliki akses terhadap *form* ini dan digunakan apabila akan menambahkan barang.



Gambar 4.25 *Form* master *barang*

Keterangan :

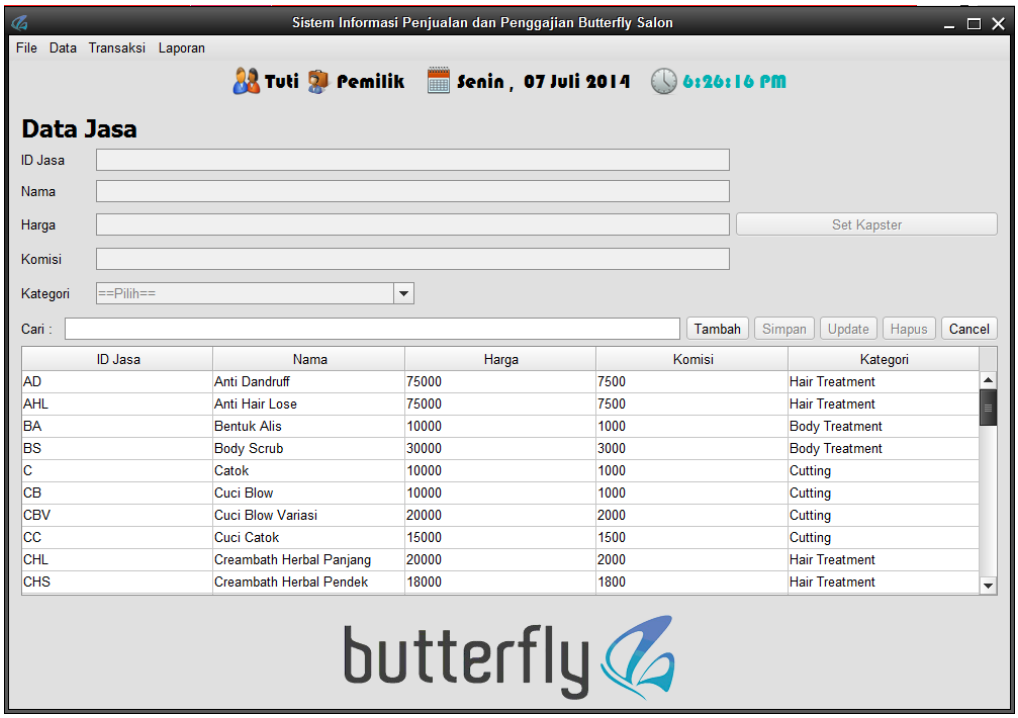
a. Teksbox ID Barang digunakan untuk mengisi identias barang.
b. Teksbox nama barang digunakan untuk mengisi nama barang.
c. Teksbox harga digunakan untuk mengisi harga dasar barang.
d. Teksbox harga jual digunakan untuk mengisi harga jual barang.
e. Teksbox stok digunakan untuk mengisi stok dari barang
f. *Button* Tambah digunakan untuk menambah data barang.
g. *Button* Simpan digunakan untuk menyimpan data barang setelah dilakukan input data dengan benar dan telah divalidasi oleh sistem.
h. *Button* Update digunakan untuk mengupdate data jasa yang dipilih dari tabel jasa.

i. *Button* Delete digunakan untuk menghapus data barang.

j. *Button* Cancel digunakan untuk membatalkan proses yang sedang berlangsung.

4. Data Master Jasa

Halaman ini merupakan *form* master jasa. Hanya pemilik salon yang memiliki akses terhadap *form* ini dan digunakan apabila akan menambahkan data jasa.



Gambar 4.26 *Form* master *jasa*

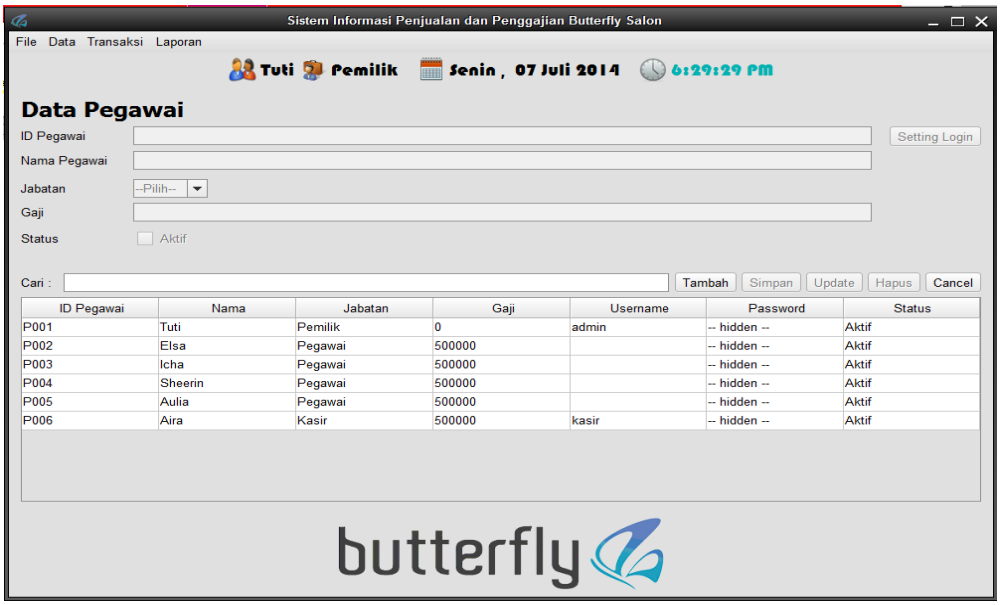
Keterangan :

a. Teksbox ID Jasa digunakan untuk mengisi identias jasa.

b. Teksbox nama jasa digunakan untuk mengisi nama jasa.
c. Teksbox harga digunakan untuk mengisi harga jasa.
d. Teksbox komisi digunakan untuk mengisi komisi jasa.
e. *Button* Tambah digunakan untuk menambah data jasa.
f. *Button* Simpan digunakan untuk menyimpan data jasa setelah dilakukan input data dengan benar dan telah divalidasi oleh sistem.
g. *Button* Update digunakan untuk mengupdate data jasa yang dipilih dari tabel jasa.
h. *Button* Delete digunakan untuk menghapus data jasa.
i. *Button* Cancel digunakan untuk membatalkan proses yang sedang berlangsung.

5. Data Master Pegawai

Halaman ini merupakan *form* master pegawai. Hanya pemilik salon yang memiliki akses terhadap *form* ini dan digunakan apabila akan menambahkan data pegawai.



Gambar 4.27 *Form* master *pegawai*

Keterangan :

a. Teksbox ID pegawai digunakan untuk mengisi kode pegawai.

b. Teksbox nama pegawai digunakan untuk mengisi nama pegawai.

c. Combo jabatan digunakan untuk memilih jabatan pegawai.

d. Cekbox status digunakan untuk memilih status keaktifan pegawai

e. *Button* Tambah digunakan untuk menambah data pegawai.

f. *Button* Simpan digunakan untuk menyimpan data pegawo setelah dilakukan input data dengan benar dan telah divalidasi oleh sistem.

g. *Button* Update digunakan untuk mengupdate data pegawai yang dipilih dari tabel pegawai.

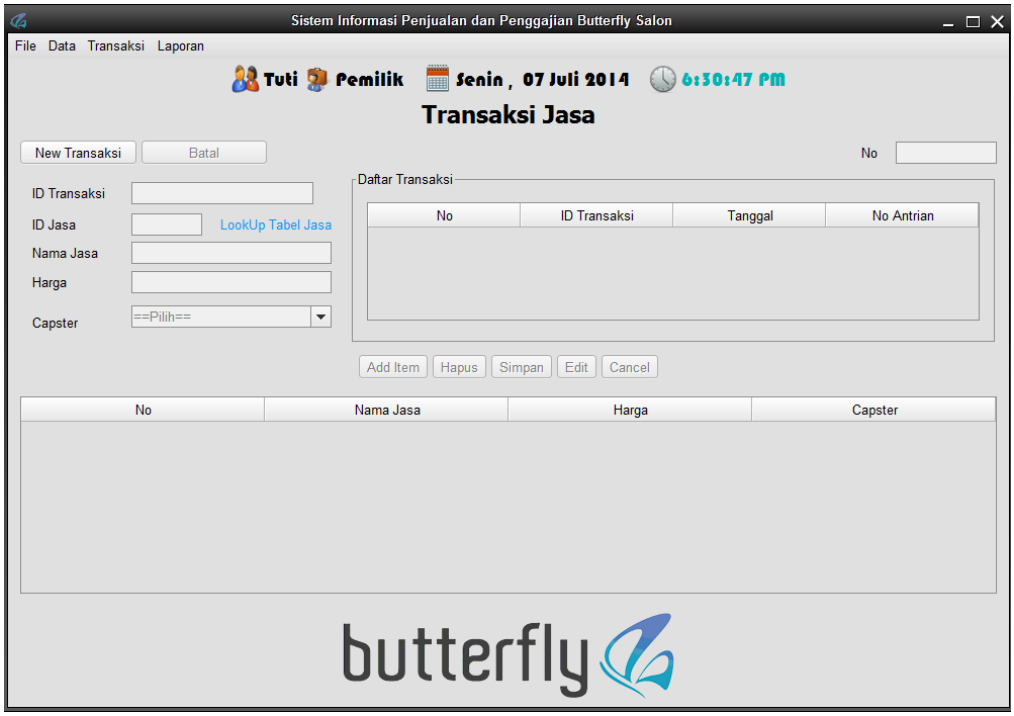
h. *Button* Delete digunakan untuk menghapus data pegawai.

i. *Button* Cancel digunakan untuk membatalkan proses yang sedang berlangsung.

j. *Button* setting login digunakan untuk mengatur login user pada aplikasi.

6. Transaki Jasa

Halaman ini merupakan *form* transaksi jasa. Hanya pemilik salon dan kasir yang memiliki akses terhadap *form* ini dan digunakan apabila akan melakukan transaksi jasa.



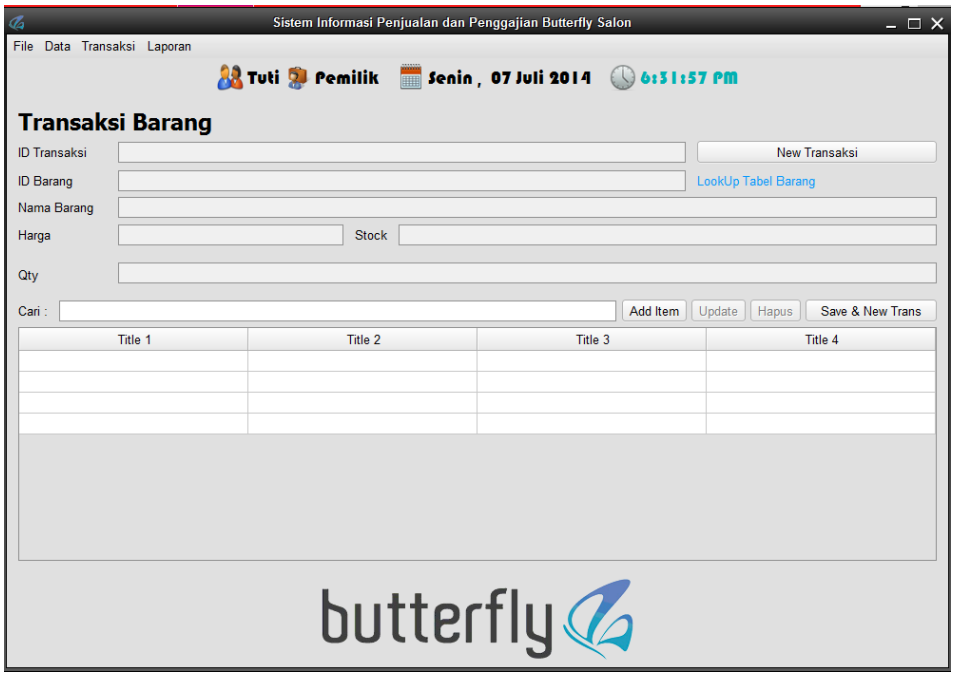
Gambar 4.28 *Form* transaksi jasa

Keterangan :

a. Teksbox ID Transaksi digunakan untuk mengisi kode transaksi.
b. Teksbox ID Jasa digunakan untuk mengisi id jasa.
c. Teksbox nama jasa digunakan untuk menampilkan nama jasa
d. Teksbox harga digunakan untuk menampilkan harga jasa
e. Combo Kapster digunakan untuk memilih cappaster
f. Button New Transaksi digunakan untuk membuat trasaksi baru.
g. *Button* add item digunakan untuk menambah item transaksi.
h. *Button* Update digunakan untuk mengupdate data trans yang dipilih dari tabel trans.
i. *Button* Hapus digunakan untuk menghapus detil item.
j. *Button* Save & New Trans digunakan untuk menutup trasaksi, mencetak truk dan membuat transaksi baru.

7. Transaki Barang

Halaman ini merupakan *form* transaksi barang. Hanya pemilik salon dan kasir yang memiliki akses terhadap *form* ini dan digunakan apabila akan melakukan transaksi barang.



Gambar 4.29 *Form* transaksi barang

Keterangan :

a. Teksbox ID Transaksi digunakan untuk mengisi kode transaksi.

b. Teksbox ID barang digunakan untuk mengisi id barang.

c. Teksbox nama barang digunakan untuk menampilkan nama barang

d. Teksbox harga digunakan untuk menampilkan harga barang

e. Teksbox Qty digunakan untuk mengisi jumlah barang

f. Button New Transaksi digunakan untuk membuat trasaksi baru.

g. *Button* add item digunakan untuk menambah item transaksi.

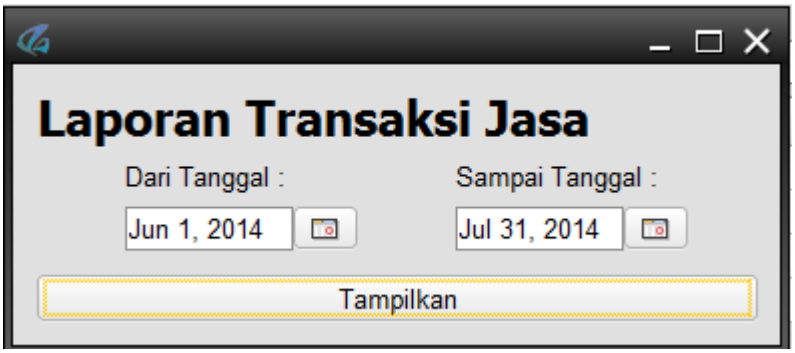
h. *Button* Update digunakan untuk mengupdate data trans yang dipilih dari tabel trans.

i. *Button* Hapus digunakan untuk menghapus detil item.

j. *Button* Save & New Trans digunakan untuk menutup trasaksi, mencetak truk dan membuat transaksi baru.

8. Laporan Transaksi

Halaman ini merupakan *form* untuk menampilkan laporan transaksi baik jasa maupun barang. Hanya pemilik salon yang memiliki akses terhadap *form* ini dan digunakan apabila ingin melihat data transaksi yang terjadi.



Gambar 4.30 *Form* laporan transaksi

Keterangan :

a. Teksbox Dari tanggal digunakan untuk memilih tanggal awal data transaksi yang akan dimunculkan pada laporan.

b. Teksbox Dari tanggal digunakan untuk memilih tanggal awal data transaksi yang akan dimunculkan pada laporan.

c. Tombol Tampilkan digunakan untuk menampilkan laporan.

9. Laporan Penggajian

Halaman ini merupakan *form* untuk menampilkan laporan penggajian dalam periode waktu tertentu. Penggajian terdiri dari gaji bulanan dan komisi. Hanya pemilik salon yang memiliki akses terhadap *form* ini dan digunakan apabila ingin melihat data penggajian.



Gambar 4.32 *Form* laporan penggajian

Keterangan :

a. Combo Pegawai digunakan untuk memilih data pegawai yang akan ditampilkan.
b. Teksbox Dari tanggal digunakan untuk memilih tanggal awal data transaksi yang akan dimunculkan pada laporan.
c. Teksbox Dari tanggal digunakan untuk memilih tanggal awal data transaksi yang akan dimunculkan pada laporan.
d. Tombol Tampilkan digunakan untuk menampilkan laporan.

4.3 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak.

4.3.1 Rencana Pengujian

Pengujian perangkat lunak Sistem Informasi Penjualan dan Penggajian Butterfly Salon ini menggunakan metode *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Tabel 4.7 Rencana Pengujian

Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis Pengujian**	**Penguji**
Login	Pengujian login (*username* dan password)	Sistem	*Black box*	Pemilik salon, kasir
Input Data	Pengisian data master	Modul	*Black box*	Pemilik Salon
Transaksi	Transaksi Barang	Modul	*Black box*	Kasir
	Transaksi Jasa	Modul	*Black box*	Kasir
Laporan	Laporan Data Master	Sistem	*Black box*	Pemilik Salon
	Laporan Transaksi	Sistem	*Black box*	Pemilik Salon
	Laporan Penggajian	Sistem	*Black box*	Pemilik Salon

4.3.2 Kasus dan Hasil Pengujian

Setelah adanya rencana pengujian maka dilanjutkan dengan kasus dan hasil pengujian. Berikut kasus dan hasil pengujian Sistem Informasi Penjualan dan Penggajian pada Butterfly Salon:

4.3.2.1 Pengujian Login

1. Pengujian login *user*

Berikut merupakan pengujian login *user* yaitu dilakukan oleh pemilik salon dan kasir dengan menggunakan *username* dan password masing-masing.

Tabel 4.8 Pengujian Login

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan *username* dan password yang benar	Dapat login dan masuk ke halaman utama	Masuk ke halaman menu utama	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan *username* dan password yang salah	Tidak dapat login dan masuk ke menu utama	Tidak dapat login dan menampilkan pesan kesalahan pengisian *username* dan password.	[✓] Diterima [ ] Ditolak

4.3.2.2 Pengujian Input

1. Pengujian data master

   Di bawah ini merupakan pengujian pengisian data master

Tabel 4.9 Pengujian data master

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua data *master* diisi dengan lengkap	Menampilkan data *master* dengan lengkap pada *database*	Dapat mengisi data *master* sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data dengan tidak lengkap	Tidak dapat tersimpan di dalam database	Data tidak dapat tersimpan di database dan memunculkan pesan data harus lengkap	[✓] Diterima [ ] Ditolak

2. Pengujian transaksi

   Di bawah ini merupakan pengujian transaksi

Tabel 4.10 Pengujian transaksi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Input data transaksi secara benar dan lengkap	Input transaksi dihitung oleh sistem dan disimpan ke *database*	Input transaksi berhasil menghitung seluruh inputan dan tersimpan di *database*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Jumlah barang pada saat transaksi tidak diinputkan	Menampilkan kotak pesan kesalahan	Transaksi tidak bisa disimpan dan muncul pesan "Jumlah barang harus diisi"	[✓] Diterima [ ] Ditolak

4.3.2.3 Pengujian Laporan

1. Pengujian laporan

   Di bawah ini merupakan pengujian laporan penggajian

Tabel 4.10 Pengujian laporan penggajian

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal periode yang sesuai dengan data yang diinputkan	Dapat mencetak laporan jasa sesuai dengan tanggal periode yang dipilih	Laporan dapat dicetak	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal periode yang tidak sesuai dengan data yang diinputkan	Tampilan output laporan perbaikan kosong tidak ada daftar perbaikan.	Hanya menampilkan output laporan kosong	[✓] Diterima [ ] Ditolak

4.3.3 Kesimpulan Hasil Pengujian

Kesimpulan pengujian yang dapat di ambil dari kasus uji dan hasil pengujian yaitu bahwa suatu perangkat lunak harus di uji terlebih dahulu agar dalam suatu sistem yang dibuat akan mendapatkan suatu komponen sistem yang baik dan fungsional, sehingga jika kemungkinan ada kesalahan pada sistem akan jauh lebih kecil atau sedikit dan terlihat secara detail dimana letak kesalahan pada sistem yang di rancang tersebut.