BAB IV

HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Pada tahap perancangan sistem ini akan dijelaskan mengenai perancangan pada objek yang digunakan, perancangan arsitektur program yang akan dibuat, perancangan tampilan dan perancangan menu pada *form* yang dibutuhkan.

Tujuan dari perancangan sistem ini adalah untuk memberikan gambaran secara umum kepada *user* tentang sistem yang baru.

4.1.1. Tujuan Perancangan Sistem

Perancangan sistem merupakan suatu proses perencanaan dalam pembuatan sistem baru dan mengacu pada kebutuhan yang sudah dianalisis sebelumnya dengan tujuan untuk menghasilkan sistem baru yang sesuai dengan kebutuhan dan diharapkan dapat membantu proses pengelolaan rekam medis di CV. Panji Putra menjadi lebih baik, efektif dan efisien.

4.1.2. Gambaran Umum Sistem Yang Diusulkan

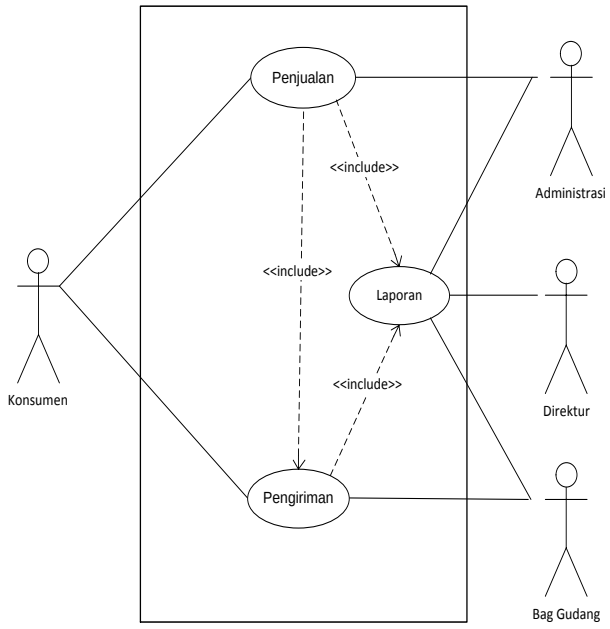
Adapun gambaran umum dari sistem yang diusulkan adalah penulis akan merancang sebuah sistem terkomputerisasi dan otomatisasi yang sebelumnya tidak ada, sehingga proses pengelolaan absensi dan penggajian bisa dilakukan lebih mudah.

4.1.3. Perancangan Prosedur Yang Diusulkan

Perancangan prosedur merupakan awal dari pembuatan sistem yang akan dibuat. Sedangkan perancangan prosedur yang diusulkan merupakan tahap untuk memperbaiki atau meningkatkan efisiensi kerja. Tahapan perancangan prosedur ini akan dijelaskan dengan menggunakan pemodelan sistem informasi berorientasi objek dengan UML *(Unified Modelling Language).*

4.1.3.1. Use Case Diagram

Berikut adalah gambar diagram *use case* yang diusulkan :



Gambar 4.1*Use Case* Sistem Yang Diusulkan.

4.1.3.2. Skenario Use Case

Berikut adalah skenario*use case* yang diusulkan :

1. Nama Use Case :Penjualan

   Aktor :Administrasi.

   Tujuan : Menyimpan transaksi penjualan

Tabel 4.1 Tabel Skenario *Use Case* Pemesanan

No	Aktor	Sistem
1	Input data penjualan	
2		Validasi data penjualan apabila data tidak lengkap akan menampilkan pesan "Silahkan Lengkapi Data"
3	Input kembali data penjualan	
4		Menyimpan data penjualan
5		Memproses dan menampilkan notapenjualan
6	Mencetak notapenjualan	

2. Nama Use Case : Pengiriman

   Aktor : B. Gudang.

   Tujuan : Melakukan Pengiriman Barang.

Tabel 4.2 Tabel Skenario *Use Case* Pengiriman.

No	Aktor	Sistem
1	Input data pengiriman berdasarkan nota penjualan	
2		Menampilkan data penjualan dan data barang yang harus dikirim
3		Menyimpan data pengiriman
4		Memproses dan menampilkan surat jalan pengiriman barang
5	Mencetak surat jalan pengiriman barang	
6	Mengirim barang	

3. Nama Use Case : Laporan

   Aktor :B. Gudang, Administrasi, Direktur.

   Tujuan : Mencetak laporan.

Tabel 4.3 Tabel Skenario *Use Case* Laporan

No	Aktor	Sistem
1	Memilih jenis laporan yang akan dibuat	
2		Menampilkan form dari

		jenis yang dipilih
3	Mengisi periode laporan	
4		Menampilkan Laporan
5	Mencetak Laporan	

4.1.3.3. Activity Diagram

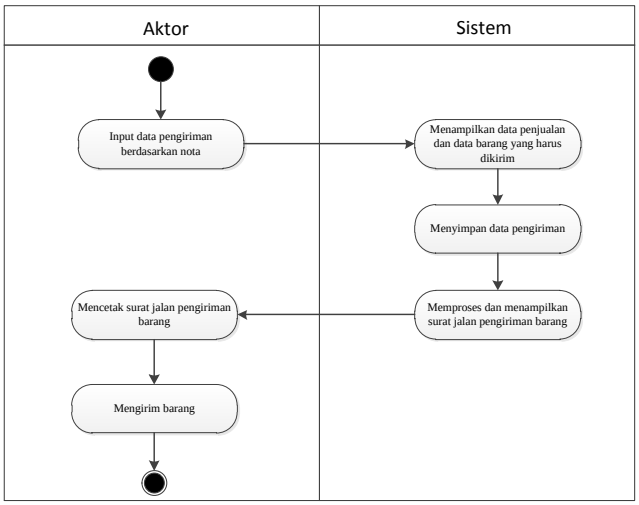
Berikut adalah *activity diagram* yang diusulkan :

1. Activity Diagram Penjualan



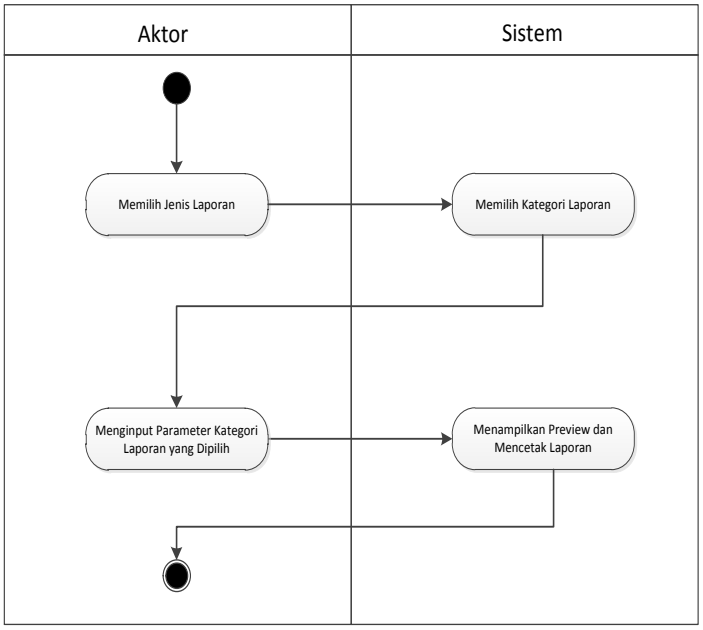
Gambar 4.2*Activity Diagram* Penjualan.

2. Activity Diagram Pengiriman



Gambar 4.3 *Activity Diagram* Pengiriman.

3. Activity Diagram Laporan

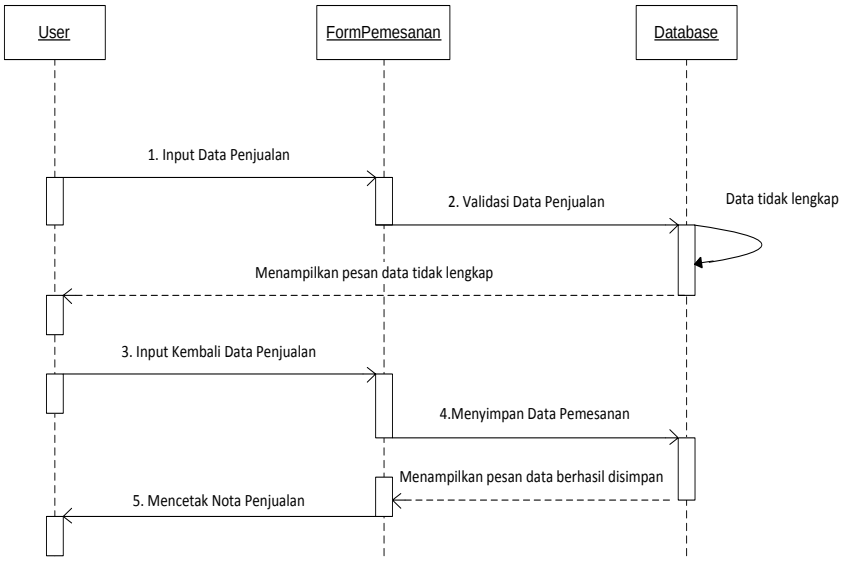


Gambar 4.4 *Activity Diagram* Laporan.

4.1.3.4. Sequence Diagram

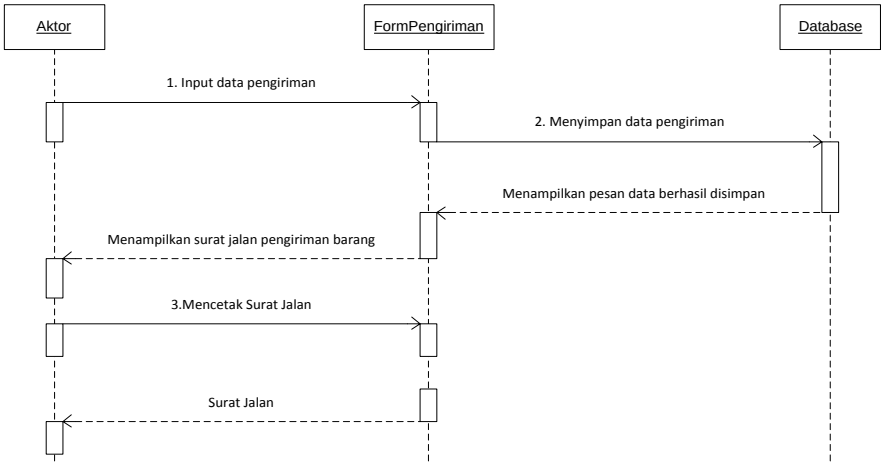
Berikut adalah *sequence diagram* yang diusulkan :

1. Sequence Diagram Penjualan



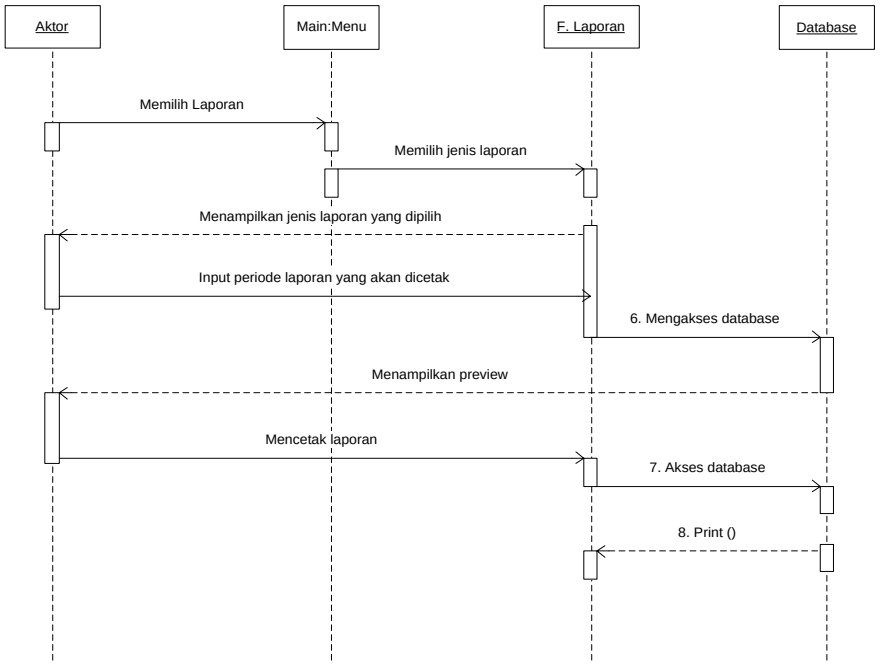
Gambar 4.5.*Sequence Diagram* Penjualan.

2. Sequence Diagram Pengiriman



Gambar 4.6*Sequence Diagram* Pengiriman.

3. Sequence DiagramLaporan

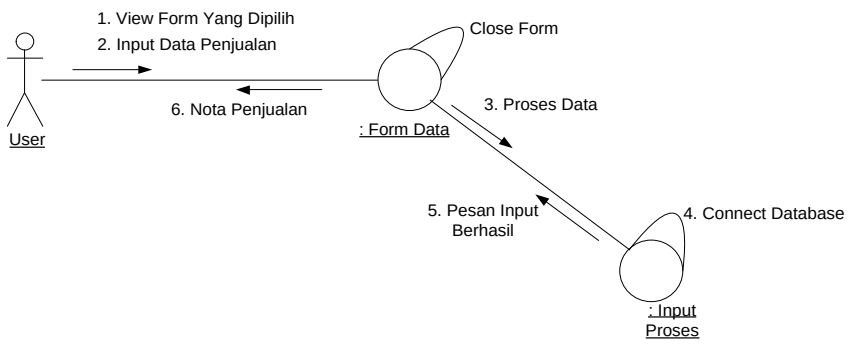


Gambar 4.7 *Sequence Diagram* Laporan.

4.1.3.5. Collaboration Diagram

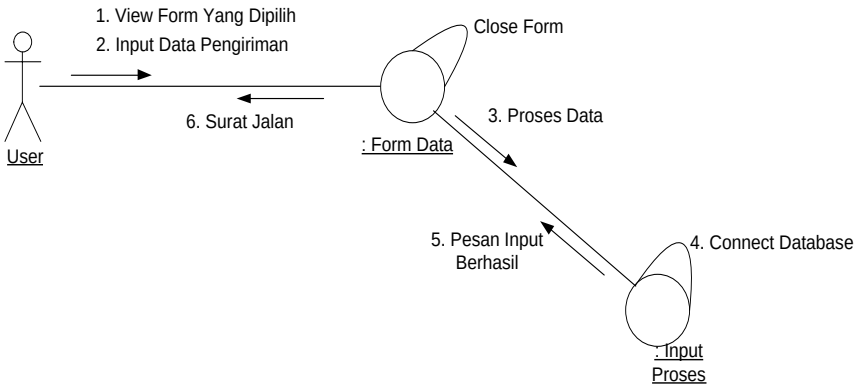
Berikut adalah *collaboration diagram*dari sistem yang diusulkan :

1. *Collaboration Diagram*Penjualan



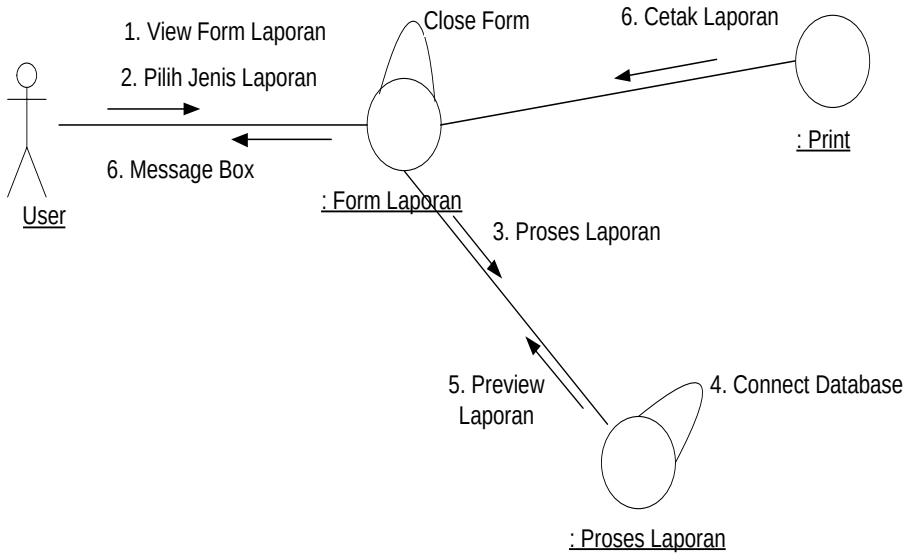
Gambar 4.8 *Collaboration Diagram*Penjualan.

2. *Collaboration Diagram*Pengiriman



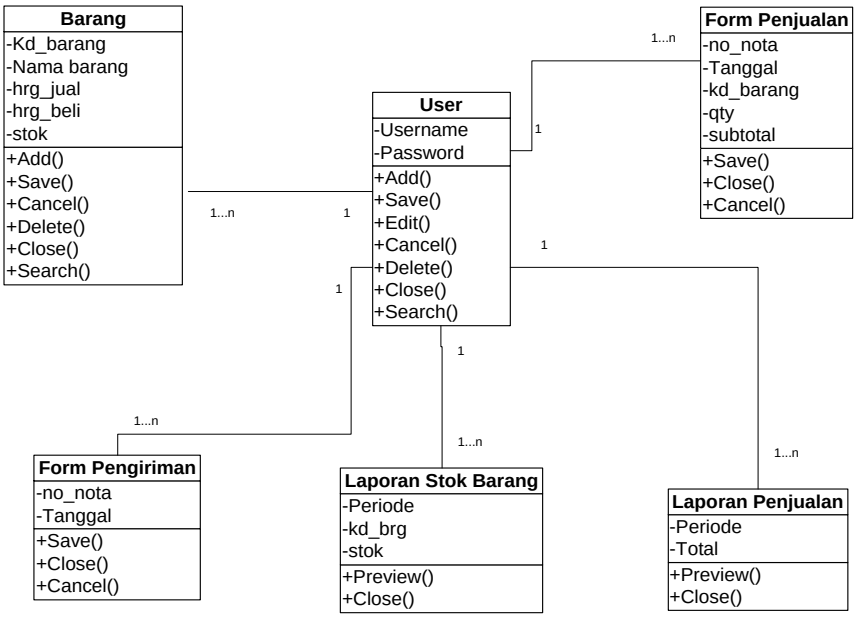
Gambar 4.9 *Collaboration Diagram*Pengiriman.

3. *Collaboration Diagram*Laporan



Gambar 4.10 *Collaboration Diagram* Laporan.

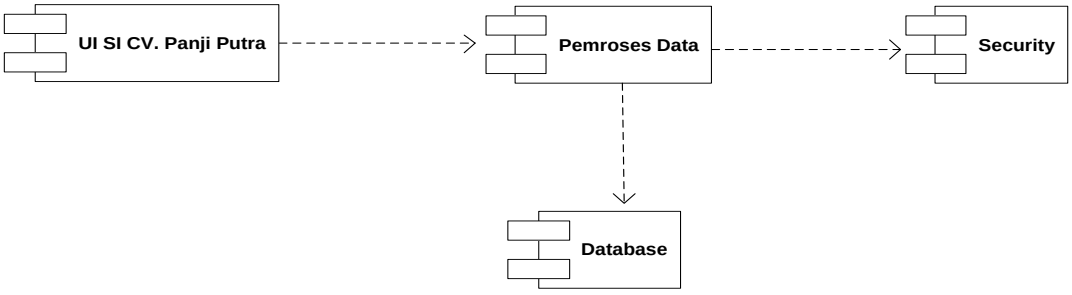
4.1.3.6. Class Diagram



Gambar 4.11*ClassDiagram*.

4.1.3.7. Component Diagram

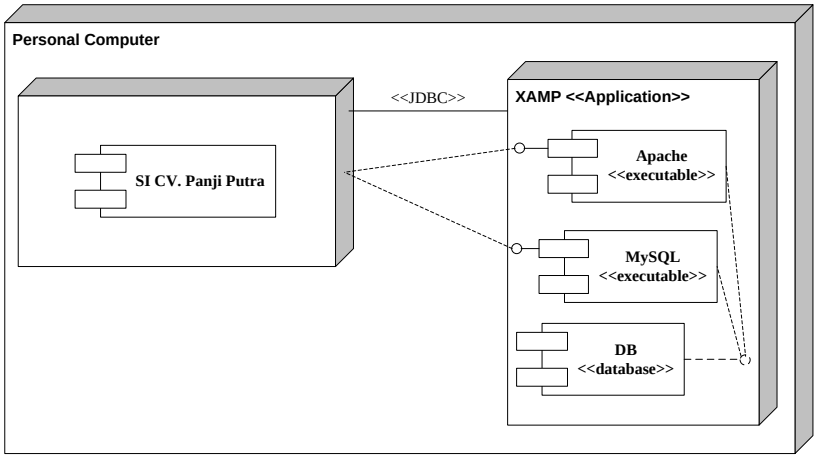
Component diagram menggambarkan struktur dan hubungan antar komponen peranti lunak, termasuk ketergantungan (*dependency*).Pada umunya komponen terbentuk dari beberapa *class* dan *package*, tapi dapat juga dari komponen-komponen yang lebih kecil.



Gambar 4.12*ComponentDiagram*.

4.1.3.8. Deployment Diagram

Adalah diagram untuk mengembangkan dari suatu sistem yang telah diciptakan sehingga dapat ditelusuri apabila terjadi kesalahan baik dalam perancangan maupun implementasinya.



Gambar 4.13*DeploymentDiagram*.

4.2 Perancangan Antar Muka

Perancangan antar muka (*interface*) merupakan suatu tahapan agar aplikasi yang dibuat lebih menarik dan tentunya lebih mudah digunakan.Tujuan dari perancangan antar muka pada adalah untuk mendapatkan suatu kriteria yang sangat penting dalam pengoprasian sebuah program aplikasi, yaitu aspek ramah dengan pengguna (*user friendly*).

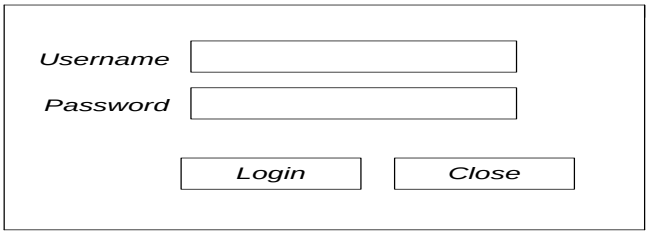
Perancangan antar muka yang ada dalam sistem berdasarkan diagram-diagram UML yang telah dibuat sebelumnya. Adapun perancangan antar muka ini terdiri dari struktur menu, perancangan *input* dan perancangan *output*.

4.2.1. Perancangan Input

Perancangan *input* merupakan *form interface* tentang sistem masukan suatu proses dari pengguna sistem yang nantinya akan dijadikan sebagai laporan untuk admin. Adapun perancangan *input* sistem informasi ini adalah sebagai berikut:

1. Form Login

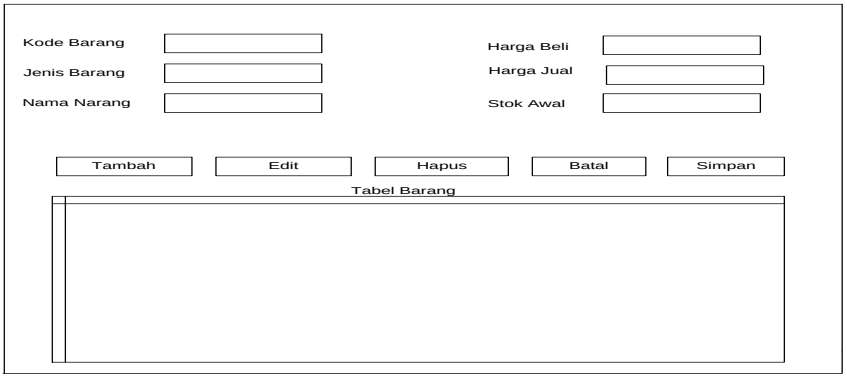
   Form login ini berfungsi sebagai keamanan sistem dimana user harus melakukan login terlebih dahulu untuk mengakses menu utama program.



Gambar 4.14 Perancangan*Form Login*.

2. Form Barang

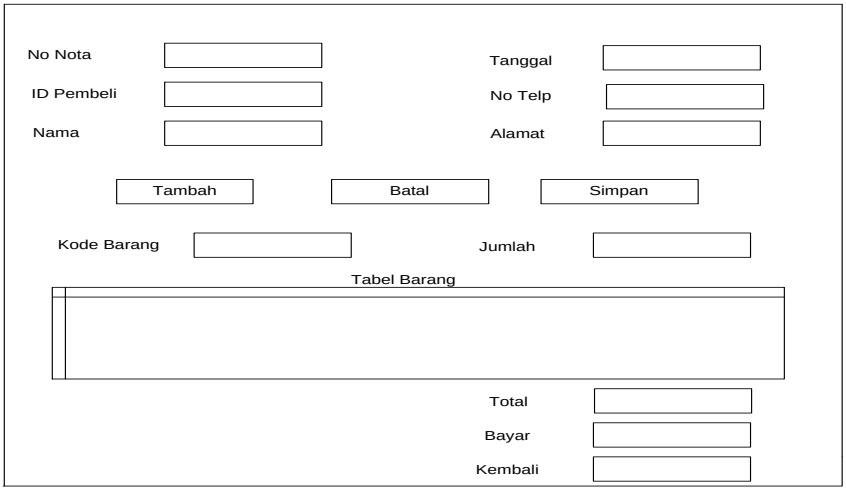
   Form ini berfungsi sebagai melihat dan mengolah data barang yang ada di gudang.



Gambar 4.15 Perancangan*Form* Barang.

3. Form Penjualan

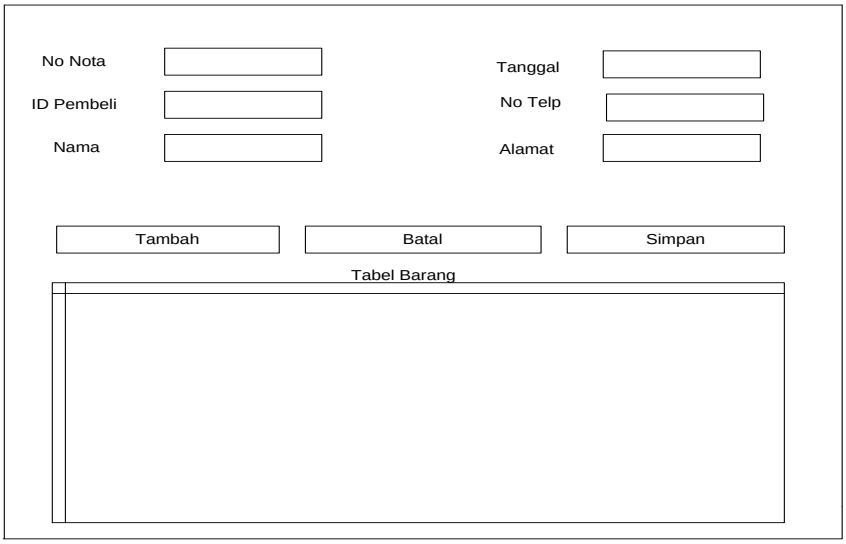
Form ini berfungsi sebagai pengelolaan pemesanan barang apa saja yang di pesan oleh konsumen.



Gambar 4.16 Perancangan*Form* Penjualan.

4. Form Pengiriman

Form ini berfungsi dalam pengelolaan data pengiriman barang.

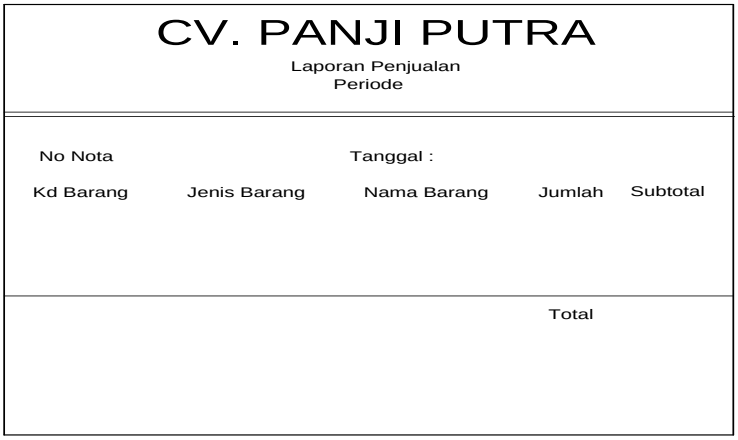


Gambar 4.17 Perancangan*Form* Pembayaran.

4.2.2. Perancangan Output

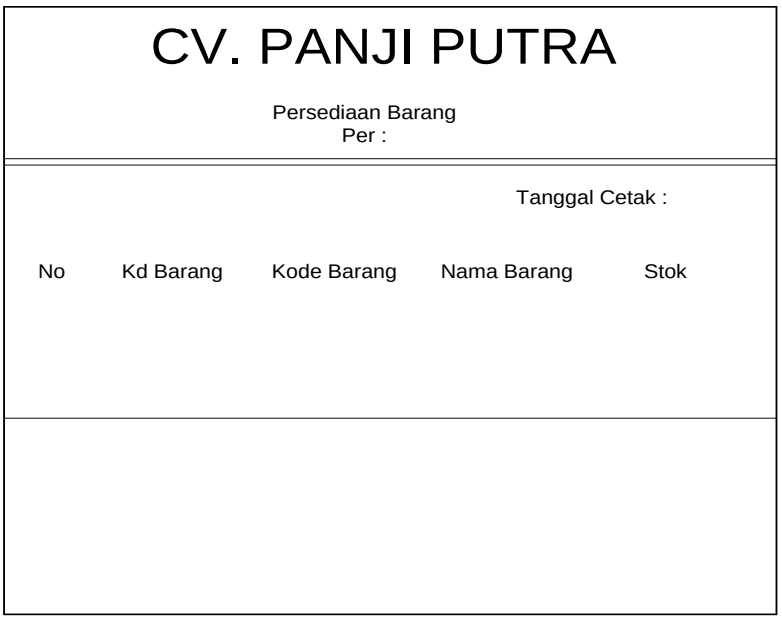
Perancangan *output* adalah informasi yang dihasilkan oleh sistem berupa laporan hasil dari proses *input*-an yang diterima oleh sistem. Adapun perancangan *output* dari sistem informasi ini adalah sebagai berikut:

1. Laporan Penjualan



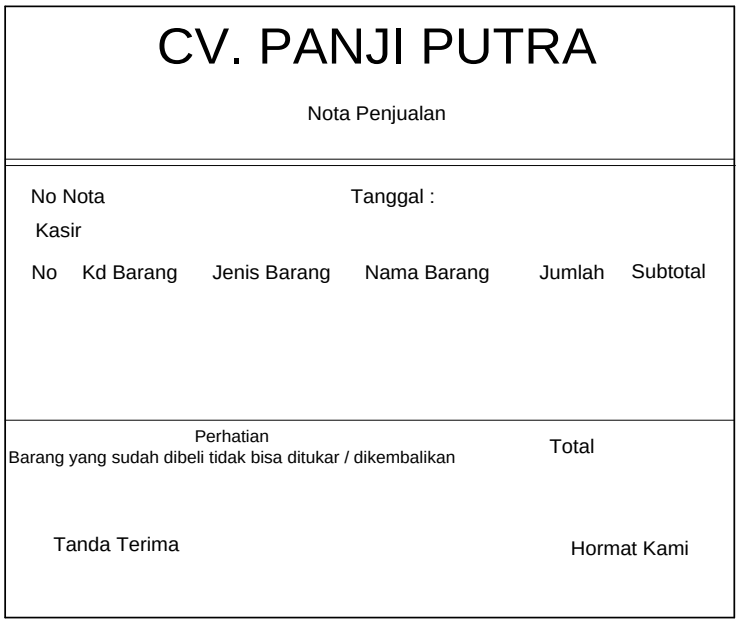
Gambar 4.18 Tampilan Laporan Penerimaan.

2. Laporan Stok Barang



Gambar 4.19 Tampilan Laporan Stok Barang.

3. Nota Penjualan


CV. PANJI PUTRA

Nota Penjualan

No Nota Tanggal :

Kasir

No Kd Barang Jenis Barang Nama Barang Jumlah Subtotal

Perhatian
Barang yang sudah dibeli tidak bisa ditukar / dikembalikan

Total

Tanda Terima

Hormat Kami

Gambar 4.20 Tampilan Nota.

4.2.3. Struktur File

Struktur *file* adalah penggambaran tentang *file-file* dalam table sehingga dapat dilihat bentuk *file-file* tersebut baik *field*, tipe data serta ukuran dari data tersebut. Berikut ini adalah struktur *file* adalah struktur *file* pada sistem informasi penerimaan dan pengeluaran barang di CV. Panji Putra.

1. File Barang
   a. Nama File : barang
   b. Primary Key : kd_brg
   c. Jumlah Field : 7

Tabel 4.4 File Barang

No	Field Name	Type	Size	Descrpition
1	Kd_brg*	Varchar	8	Kode barang
2	Jns_brg	Varchar	30	Jenis barang
3	Nm_brg	Varchar	100	Nama barang

4	Hrg_beli	Double		Harga beli barang
5	Hrg_jual	Double		Harga jual barang
6	Stok	Integer	11	Stok barang
7	Satuan	Varchar	10	Satuan barang

2. File Orders

   a. Nama File : orders

   b. Primary Key : noPO

   c. Jumlah Field : 4

Tabel 4.5 File Orders

No	Field Name	Type	Size	Descrpition
1	NoPO*	Varchar	11	No purchase order
2	Tanggal	Date	-	Tanggal PO
3	kd_pelanggan	Varchar	5	Nama supplier
4	Status	Integer	1	Status PO

3. File Detail Orders

   a. Nama File : dt_orders

   b. Foreign Key : noPO

   c. Jumlah Field : 3

Tabel 4.6 File Detail Orders

No	Field Name	Type	Size	Descrpition
1	NoPO**	Varchar	11	No purchase order
3	kd_brg**	Varchar	8	Kode Barang
4	Qty	Int	11	Jmlah Barang

4. File Pesan

   a. Nama File : pesan

   b. Primary Key : no_pesan

   c. Jumlah Field : 4

Tabel 4.7 File Pesan

No	Field Name	Type	Size	Descrpition
1	No_pesan*	Varchar	11	No pemesanan
2	Tanggal	Date	-	Tanggal pesan
3	Supplier	Varchar	50	Nama Supplier
4	Total	Double	-	Total pemesanan
5	Status	Integer	1	Status pemesanan

5. File Detail Pesan

   a. Nama File : dt_ pesan

   b. Foreign Key : No_pesan

   c. Jumlah Field : 3

Tabel 4.8 File Detail Pesan

No	Field Name	Type	Size	Descrpition
1	No_pesan**	Varchar	11	No purchase order
3	kd_brg**	Varchar	8	Kode Barang
4	Qty	Int	11	Jmlah Barang

6. File Pengiriman

   a. Nama File : pengiriman

   b. Foreign Key : no_kirim

   c. Jumlah Field : 3

Tabel 4.9 File pengiriman

No	Field Name	Type	Size	Descrpition
1	No_kirim*	Varchar	11	No pengiriman
2	noPO**	Varchar	10	No PO
3	Tanggal	Date	-	Tanggal pengiriman

7. File Jual

   a. Nama File : jual

   b. Primary Key : no_nota

   c. Jumlah Field : 7

Tabel 4.10 File Jual

No	Field Name	Type	Size	Descrpition
1	No_nota*	Varchar	11	No nota
2	Tanggal	Date	-	Tanggal transaksi
3	NoPO**	Varchar	11	No PO
4	Kasir	Varchar	50	Nama Kasir
5	Total	Double		Total penjualan
6	Bayar	Double		Total bayar
7	Cek	Varchar	10	No cek/giro

8. File Detail Jual

   a. Nama File : dt_jual

   b. Foreign Key : no_nota

   c. Jumlah Field : 3

Tabel 4.11 File Detail Jual

No	Field Name	Type	Size	Descrpition
1	No_nota**	Varchar	11	No nota
3	kd_brg**	Varchar	8	Kode Barang
4	Qty	Int	11	Jmlah Barang

9. File Supplier

   a. Nama File : supplier

   b. Primary Key : Kd_supplier

   c. Jumlah Field : 4

Tabel 4.12 File Supplier

No	Field Name	Type	Size	Descrpition
1	Kd_supplier	Varchar	15	Kode supplier
2	Nm_supplier	Varchar	250	Namasupplier
3	No_telp	Varchar	13	No telepon
4	Alamat	Varchar	300	Alamat supplier

4.2.4. Kodifikasi

Kodifikasi atau pengkodean digunakan untuk menjabarkan item-item data yang bersifat unik. Dalam perancangan ini penulis melakukan pengkodean sebagai berikut :

1. Kode Barang

   XXX-XXXX

   ↓ ↳ No Urut barang

   Kode Jenis Barang

   Contoh :CAT-0001

   CAT : Menunjukan kode jenis barang.

   0001 : Menunjukan no. urut barang.

2. No Pemesanan

   XXXX XXXX ⟶ No Urut Pemesanan

   ↓ ↳ Tahun dan bulan pembuatan

   Kode Pemesanan

   Contoh : PS-14040001

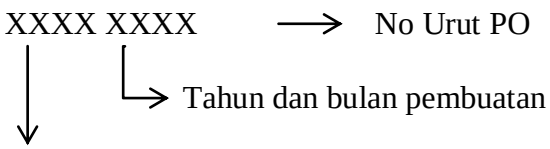
   PS : Menunjukan kode pemesanan.

   14 : Menunjukan tahun pemesanan

   04 : Menunjukan bulan pemesanan.

0001 : Menunjukan no. urut pemesanan.

3. No PO (*Purchase Order*)



Contoh : PO-14040001

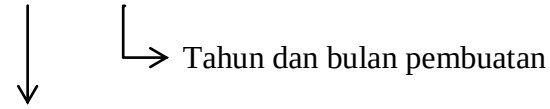
PO : Menunjukan kode PO.

14 : Menunjukan tahun PO

04 : Menunjukan bulan PO.

0001 : Menunjukan no. urut PO.

4. No Nota



Contoh :NJ-14040001

NJ : Menunjukan kode nota.

14 : Menunjukan tahun nota

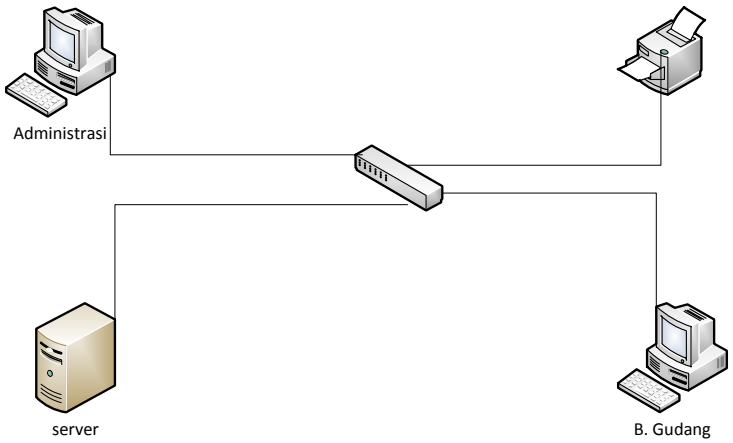
04 : Menunjukan bulan nota.

0001 : Menunjukan no. urut nota.

4.3 Perancangan Arsitektur Jaringan

Perancangan arsitektur jaringan adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer secara terintegrasi.Pada perancangan arsitektur jaringan harus disesuaikan dengan fungsi bagi penggunanya agar tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan.

Adapun tipe jaringan yang digunakan adalah tipe jaringan LAN *(local area network*), yakni tipe jaringan untuk area kecil yang cocok digunakan dalam perancangan sistem informasi ini. Dan model hubungan yang digunakan adalah *client server*, dimana model *client server* ini memungkinkan jaringan untuk mensentralisasi fungsi dan aplikasi kepada satu atau dua *dedicated file server*. Sebuah *file server* menjadi jantung dari keseluruhan sistem yang bisa mengakses sumber daya dan menyediakan keamanan. Model hubungan ini menyediakan mekanisme untuk mengintegrasikan seluruh komponen yang ada di jaringan dan memungkinkan banyak pengguna secara bersama-sama memakai sumber daya pada *file server*.



Gambar 4.21. Model Hubungan *Client Server*

4.4 Implementasi

Tahap Selanjutnya dalam membangun sebuah perangkat lunak adalah implementasi.Implementasi merupakan kumpulan dari elemen-elemen yang telah didesain ke dalam bentuk pemrograman untuk menghasilkan suatu tujuan yang dibuat berdasarkan kebutuhan yang telah dianalisis sebelumnya.Langkah-langkah dalam tahap implementasi ini adalah urutan kegiatan awal sampai akhir yang harus dilakukan untuk mewujudkan sistem yang telah dirancang.

4.4.1. Implementasi Perangkat Lunak

Perangkat lunak yang dibutuhkan dan digunakan dalam pembuatan aplikasi sistem informasi ini adalah sebagai berikut:

1. Windows 7 (*seven)*, sebagai sistem operasi yang dipakai oleh penulis.
2. Netbeans IDE 7.4, sebagai *software* yang penulis gunakan dalam membangun sistem informasi pengelolaan data tamu dan pengunjung ini.
3. iReport 4.7.1, sebagai aplikasi untuk pembuatan laporan.
4. MySQL, sebagai pengembang basis data dan perangkat lunak lainnya menggunakan XAMPP.

4.4.2. Implementasi Perangkat Keras

Adapunperangkat keras yang digunakan dalam pembuatan aplikasi sistem informasi ini adalah sebagai berikut :

1. Proccesor : Intel(R) Core(TM) i3 CPU 540 @ 3.10GHz.

2. Memory : 4096MB RAM.

3. Printer : Standar.

4. Hard Disk : 500GB

5. Mouse dan Keyboard.

4.4.3. Implementasi Basis Data

```
-- phpMyAdmin SQL Dump

-- version 3.5.2.2

-- http://www.phpmyadmin.net

--

-- Inang: 127.0.0.1

-- Waktu pembuatan: 04 Jun 2014 pada 00.02

-- Versi Server: 5.5.27

-- Versi PHP: 5.4.7

--

-- Basis data: `panji`

--


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


--

-- Struktur dari tabel `barang`

--
```

```
CREATE TABLE IF NOT EXISTS `barang` (
  `kd_brg` varchar(15) NOT NULL,
  `jenis_brg` varchar(50) NOT NULL,
  `nm_brg` varchar(200) NOT NULL,
  `hrg_beli` int(11) NOT NULL,
  `hrg_jual` int(11) NOT NULL,
  `stok` int(11) NOT NULL,
  `satuan` varchar(15) NOT NULL,
  PRIMARY KEY (`kd_brg`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `beli`
--

CREATE TABLE IF NOT EXISTS `beli` (
  `no_faktur` varchar(15) NOT NULL,
  `no_po` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  `supplier` varchar(200) NOT NULL,
  `total` int(11) NOT NULL,
```

```
  PRIMARY KEY (`no_faktur`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `buyer`
--

CREATE TABLE IF NOT EXISTS `buyer` (
  `kd_pelanggan` varchar(8) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `NoTelp` varchar(15) NOT NULL,
  `Alamat` varchar(300) NOT NULL,
  PRIMARY KEY (`kd_pelanggan`),
  UNIQUE KEY `NoTelp` (`Alamat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `dt_jual`
--
```

```
CREATE TABLE IF NOT EXISTS `dt_jual` (
  `no_nota` varchar(15) NOT NULL,
  `kd_brg` varchar(15) NOT NULL,
  `jenis_brg` varchar(50) NOT NULL,
  `nm_brg` varchar(200) NOT NULL,
  `jumlah` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `dt_pesan`
--

CREATE TABLE IF NOT EXISTS `dt_pesan` (
  `no_po` varchar(15) NOT NULL,
  `kd_brg` varchar(15) NOT NULL,
  `jenis_brg` varchar(50) NOT NULL,
  `nm_brg` varchar(200) NOT NULL,
  `jumlah` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

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

--
-- Struktur dari tabel `dt_po`
--

CREATE TABLE IF NOT EXISTS `dt_po` (
  `noPO` varchar(10) NOT NULL,
  `kd_brg` varchar(10) NOT NULL,
  `jenis_brg` varchar(50) NOT NULL,
  `nm_brg` varchar(100) NOT NULL,
  `qty` int(11) NOT NULL,
  `satuan` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `jual`
--

CREATE TABLE IF NOT EXISTS `jual` (
  `no_nota` varchar(15) NOT NULL,
```

```
  `tanggal` date NOT NULL,
  `noPO` varchar(15) NOT NULL,
  `kasir` varchar(200) NOT NULL,
  `total` double NOT NULL,
  `bayar` double NOT NULL,
  `cek` varchar(20) NOT NULL,
  PRIMARY KEY (`no_nota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `login`
--

CREATE TABLE IF NOT EXISTS `login` (
  `kd_user` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  `log` time NOT NULL,
  `bagian` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

```
--
-- Struktur dari tabel `pengiriman`
--

CREATE TABLE IF NOT EXISTS `pengiriman` (
  `no_kirim` varchar(15) NOT NULL,
  `noPO` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`no_kirim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `pesan`
--

CREATE TABLE IF NOT EXISTS `pesan` (
  `no_po` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  `supplier` varchar(200) NOT NULL,
  `total` int(11) NOT NULL,
```

```
  `keterangan` int(1) NOT NULL COMMENT '0=Belum, 1=Sudah',
  PRIMARY KEY (`no_po`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `po`
--

CREATE TABLE IF NOT EXISTS `po` (
  `noPO` varchar(10) NOT NULL,
  `Tanggal` date NOT NULL,
  `kd_pelanggan` varchar(10) NOT NULL,
  `keterangan` int(1) NOT NULL,
  PRIMARY KEY (`noPO`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `supplier`
--
```

```
CREATE TABLE IF NOT EXISTS `supplier` (
  `KodeSupplier` varchar(8) NOT NULL,
  `NamaSupplier` varchar(250) NOT NULL,
  `NoTelp` varchar(15) NOT NULL,
  `Alamat` varchar(300) NOT NULL,
  PRIMARY KEY (`KodeSupplier`),
  UNIQUE KEY `NoTelp` (`Alamat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

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

--
-- Struktur dari tabel `user`
--

CREATE TABLE IF NOT EXISTS `user` (
  `kd_user` varchar(15) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `username` varchar(50) NOT NULL,
  `password` varchar(15) NOT NULL,
  `alamat` varchar(200) NOT NULL,
  `telpon` varchar(15) NOT NULL,
```

```
  `Posisi` varchar(50) NOT NULL,
  PRIMARY KEY (`kd_user`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.4.4. Implementasi Antar Muka

Tahap implemantasi antar muka yang dilakukan adalah dengan menggunakan sebuah *file project* yang berekstensi .java yang ada dalam program Netbeans, yang di dalamnya terdapat *form-form* untuk pembuatan aplikasi.

1. Implementasi Login

   Login merupakan hal penting agar dapat masuk ke dalam menu utama aplikasi pengelolaan datapasien dan data transaksi. Fasilitas *login* ini diperuntukan bagi*user*(bag. gudang dan admin).

   Dalam melakukan *login*, *input*-an*password* dan *username* harus benar, apabila salah, maka *user* tidak bisa mengakses aplikasi ini.

Tabel 4.13 Implementasi Login

Sub Menu	Deskripsi	Nama File
Menu Login	File menu utama program	FLogin.java
Login	Untuk login admin (petugas)	
Batal	Untuk batal melakukan login	
Keluar	Untuk keluar dari form login	

2. Imlpementasi Halaman Utama

Implementasi halaman utama dari aplikasi ini adalah sebagai berikut :

Tabel 4.14 Halaman Utama

Sub Menu	Deskripsi	Nama File
Menu Utamam	File program yang form login	FHome.java
Akses	Untuk menampilkan sub menu hak akses.	
Master	Untuk menampilkan sub menu data master.	
Transaksi	Untuk menampilkan sub menu transaksi.	
Laporan	Menampilkan form laporan	FLaporan.java

3. Implementasi Halaman Berdasarkan Sub Menu Akses

Implementasi sub-menu akses dari aplikasi ini adalah sebagai berikut :

Tabel 4.15 Implementasi Sub Menu Akses

Sub Menu	Deskripsi	Nama File
Logout	Untuk logout *user*	
Keluar	Untuk keluar dari aplikasi	

4. Implementasi Halaman Berdasarkan Sub Menu Master

Implementasi sub-menu master dari aplikasi ini adalah sebagai berikut :

Tabel 4.16 Implementasi Sub Menu Master

Sub Menu	Deskripsi	Nama File
Barang	Untuk mengolah data barang	FBarang.java
Pelanggan	Untuk mengolah data pelanggan	FPelanggan.java
User	Untuk mengolah data*user*	Fuser.java
Supplier	Untuk mengolah data*supplier*	FSupplier.java

5. Implementasi Halaman Berdasarkan Sub Menu Proses

Implementasi sub-menu proses dari aplikasi ini adalah sebagai berikut :

Tabel 4.17 Implementasi Sub Menu Proses

Sub Menu	Deskripsi	Nama File
Pemesanan	Untuk input data pemesanan barang dari pelanggan	FPesan.java
Pembayaran	Untuk input data penjualan	FJual.java
Kirim	Untuk input data pengiriman	FKirim.java
Order	Untuk input data *order* barang ke *supplier*	FOrder.java
Penerimaan	Untuk input data penerimaan	FTerima.java

6. Implementasi Halaman Berdasarkan Sub Menu Laporan

Implementasi sub-menu laporan dari aplikasi ini adalah sebagai berikut :

Tabel 4.18 Implementasi Sub Menu Laporan

Sub Menu	Deskripsi	Nama File
Menu Laporan	Untuk mencetak laporan pembelian dan penjualan	FLaporan.java

4.4.5. Implementasi Instalasi Program

Dalam proses implementasi instalasi program ini, akan dijelaskan bagaimana cara menginstal XAMPP, sebagai pengembang pengelolaan basis data, dan juga cara menginstal program yang diusulkan oleh penulis kepada CV. Panji Putratempat penulis melakukan penelitian.

4.4.5.1. Instalasi XAMPP

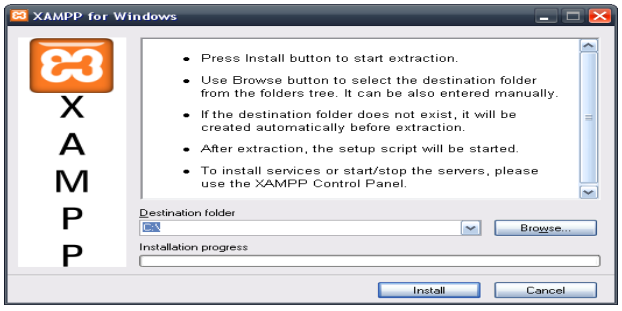
Berikut adalah langkah – langkah dalam instalasi perangkat lunak (*software*)XAMPP :

1. Pertama yang harus dilakukan adalah membuka *file* instalasi xampp win32-1.8.1-VC9-installer yang berekstensi.exe lalu *double click* untuk menginstalnya.

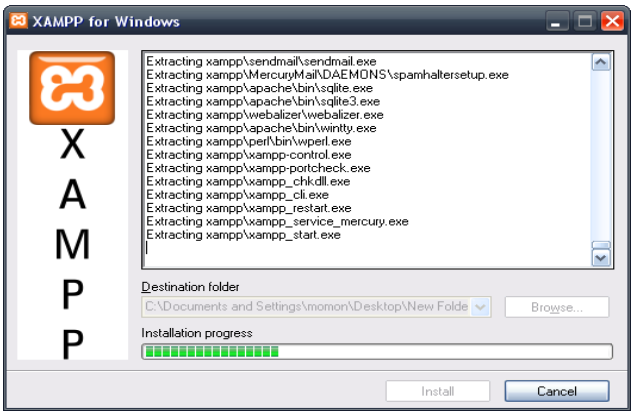


Gambar 4.22 Icon Instalasi xampp

2. Pilih *Install*

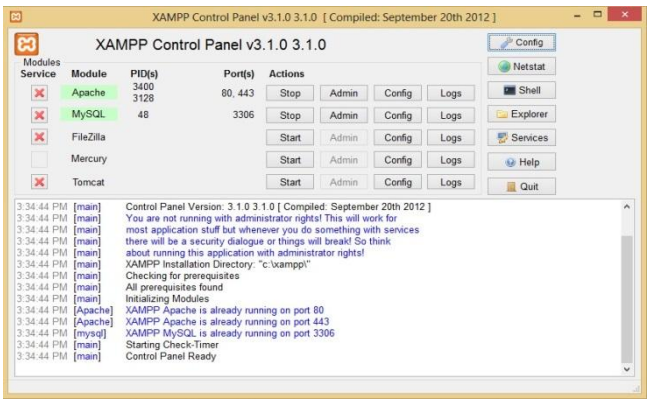


Gambar 4.23 Tampilan Awal Instalasi xampp



Gambar 4.24 Tampilan Proses Instalasi xampp

3. Setelah selesai instalasi, *double click* xampp-control untuk menjalankannya. Maka akan muncul tampilan seperti pada Gambar berikut :



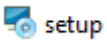
Gambar 4.25 Tampilan xampp control panel

4. Jalankan Apache dan MySql nya dengan menekan tombol start di sisi kanannya sehingga pada Apache dan MySql muncul progressbar “Running”.
5. Selesai

4.4.5.2. Instalasi Program Aplikasi

Berikut adalah langkah – langkah dalam instalasi perangkat lunak (*software*)sistem informasi CV. Panji Putra :

1. Pertama yang harus dilakukan adalah *double* klik pada *icon file setup* program.



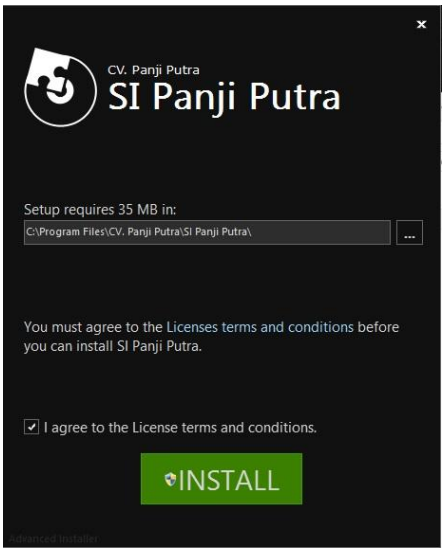
Gambar 4.26.*Icon* Instalasi Program.

2. Muncul tampilan awal instalsi program



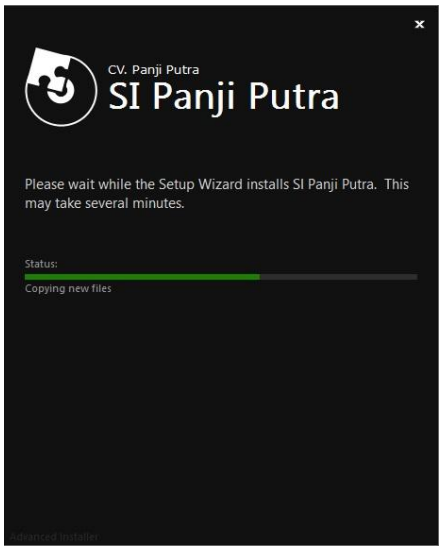
Gambar 4.27 Tampilan awal Instalasi Program

3. Pilih direktori yang diinginkan, kemudian centang *I agree to the license terms and conditions*, lalu akan muncul tombol*install*.



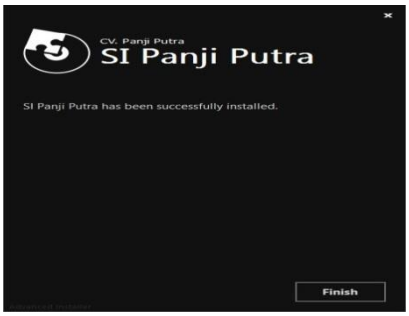
Gambar 4.28 Tampilan pemilihan direktori instalasi

4. Klik *install*, lalu tunggu hingga proses instalasi selesai.



Gambar 4.29 Tampilan proses instalasi program

5. Setelah proses instalasi selesai, klik *finish* dan program sudah berhasil terinstal



Gambar 4.30 Tampilan akhir instalasi program

4.4.6. Penggunaan Program

Kegiatan ini bertujuan untuk menerangkan secara singkat mengenai penggunaan program sistem informasi ini. Adapun tampilan dari program tersebut adalah sebagai berikut:

1. Tampilan Form Login

Form login ini berfungsi sebagai keamanan sistem. Untuk melakukan *login*, isi *password* dan *username* dan klik tombol *login*.Jika *username* dan *password* benar, maka *form* utama muncul. Klik tombol batal untuk membatalkan *login* dan keluar dari program, tombol *reset* untuk mengosongkan *fieldusername* dan *password*.



Gambar 4.31. Form Login.

2. Tampilan Menu Utama

Form menu utama ini berfungsi untukmeng-akses sub-sub menu yang ada pada aplikasi ini. Untuk memilihsub-sub menu yang ada pada aplikasi ini, klik pada nama menu yang diinginkan.



Gambar 4.32.Tampilan Menu Utama.

3. Tampilan Data Barang

Form ini berfungsi untukmelihat, mengolah, dan mencari data barang.Untuk menambah data barang, *user* harus menekan tombol tambah kemudian harus mengisi *field-field* data barang dengan lengkap.Setelah mengisi *field-field* data barang kemudian *user* harus menekan tombol simpan untuk melakukan penyimpanan data barang ke database, klik tombol batal untuk membatalkan penambahan data barang.

Apabila ada kesalahan dalam pengisian data barang, *user* dapat merubah data barang tersebut dengan cara memilih data barang yang

ada pada tabel, kemudian tekan tombol edit untuk melakukan koreksi data barang, untuk menyimpan perubahan data barang tekan update setelah melakukan koreksi data barang.

Untuk menghapus data barang, user terlebih dahulu memilih data barang yang ada pada tabel, kemudian tekan tombol hapus.Klik tombol cetak untuk mencetak laporan persediaan barang.



Gambar 4.33.Tampilan Data Barang.

4. Tampilan Data Pelanggan

Form ini berfungsi untukmelihat dan mengolah data pelanggan.Untuk menambah data pelanggan, user harus menekan tombol tambah kemudian isi *field-field* data pelanggan dengan lengkap, kemudian klik tombol simpan untuk melakukan penyimpanan data ke dalam database, klik tombol batal untuk membatalkan penambahan data.

Klik tombol edit untuk melakukan koreksi data pelanggan apabila ada kesalahan data pelanggan, untuk menyimpan perubahan tekan tombol update.

Untuk menghapus data pelanggan, *user* terlebih dahulu memilih data pelanggan yang ada pada tabel, kemudian tekan tombol hapus.



Gambar 4.34.Tampilan Data Pelanggan.

5. Tampilan Form Penjualan

Form ini berfungsi untuk*input* data penjualan barang dari pelanggan. Untuk melakukan transaksi penjualan klik tombol baru, kemudian input data pelanggan. Apabila data pelanggan sudah ada dalam database, klik tombol cari kemudian pilih data pelanggan, klik tombol baru apabila data pelanggan belum ada pada database.

Kemudian *user* menginputkan data-data barang yang dipesan, setelah data terisi klik tombol simpan untuk menyimpan data penjualan.



Gambar 4.35.Tampilan Form Penjualan.

6. Tampilan Form Order Barang

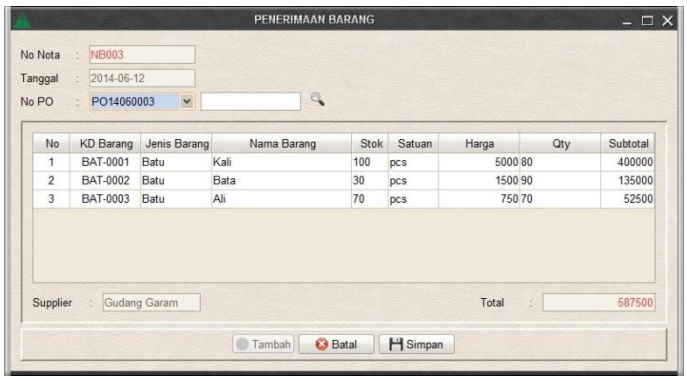
Form ini berfungsi untuk*input* data *order* barang ke *supplier*. Untuk melakukan pemesanan barang ke *supplier* pertama-tama pilih *supplier* kemudian pilih data barang yang akan di pesan. Klik tombol simpan untuk menyimpan data pemesanan barang ke supplier, Klik tombol batal untuk membatalakan pemesanan.



Gambar 4.36.Form Order Barang.

7. Tampilan Form Penerimaan Barang

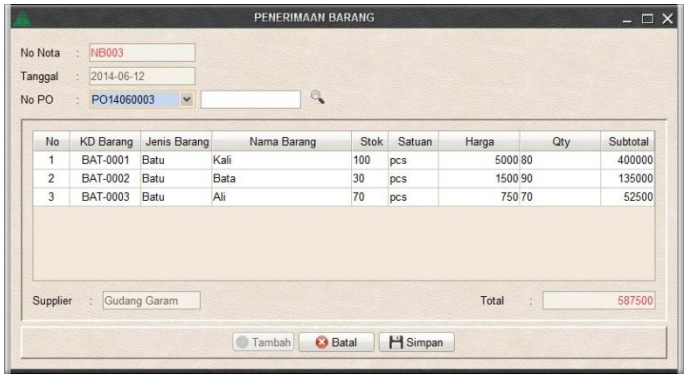
Form ini berfungsi untuk*input* data penerimaan barang.Pilih no pesanan yang tertera pada *combobox* kemudian data pemesanan akan muncul pada tabel. Klik tombol simpan untuk melakukan penerimaan barang kemudian klik tombol batal untuk membatalkan penerimaan barang.



Gambar 4.37.Form Penerimaan Barang.

8. Tampilan Form Pengiriman Barang

Form ini berfungsi untuk*input* data pengiriman barang.Pilih no pesanan yang tertera pada *combobox* kemudian data pengiriman barangakan muncul pada tabel.Klik tombol simpan untuk melakukan pengiriman barang kemudian klik tombol batal untuk membatalkan penerimaan barang.

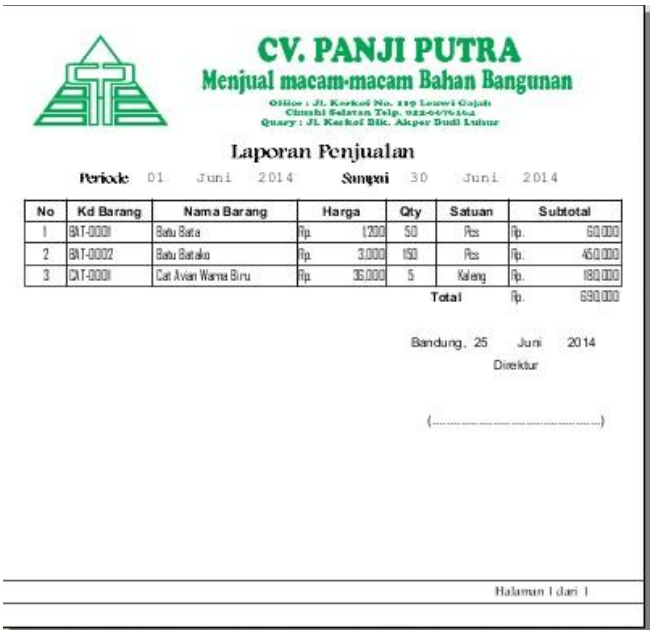


Gambar 4.38.Form Pengiriman Barang.

9. Tampilan Laporan Penjualan

Untuk menampilkan laporan penjualan terlebih dahulu pilih menu laporan pada form utama, kemudian pilih menu laporan penjualan. Masukan periode laporan kemudian klik tombol cetak untuk melihat laporan penjualan.

Berikut adalah tampilan dari laporan penjualan.

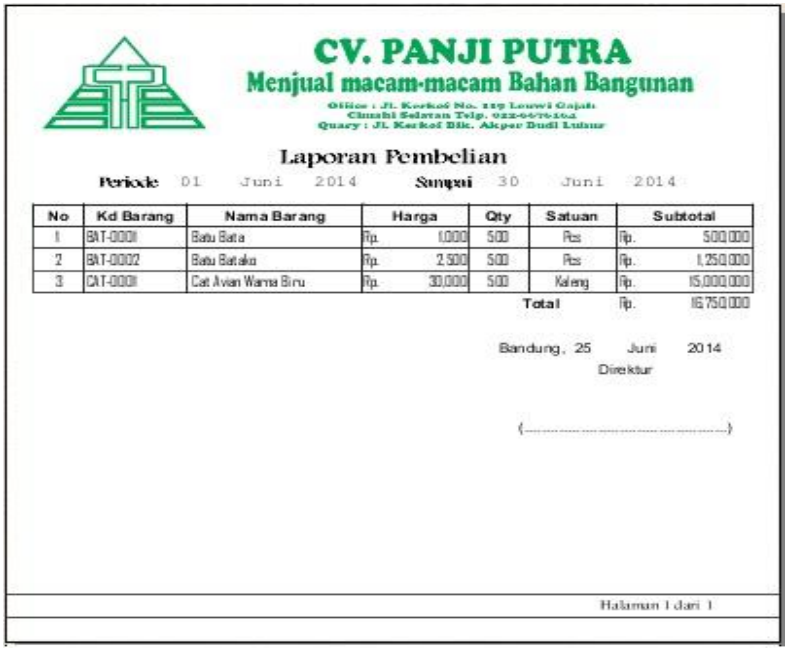


Gambar 4.39.Tampilan Laporan Penjualan.

10. Tampilan Laporan Pembelian

Untuk menampilkan laporan pembelian terlebih dahulu pilih menu laporan pada form utama, kemudian pilih menu laporan pembelian. Masukan periode laporan kemudian klik tombol cetak untuk melihat laporan pembelian.

Berikut adalah tampilan dari laporan pembelian.



CV. PANJI PUTRA

Menjual macam-macam Bahan Bangunan

Office : Jl. Kerkof No. 119 Leuwi Gajah
Cimahi Selatan Telp. 022-6670164
Quary : Jl. Kerkof Blk. Akper Dudi Lubur

Laporan Pembelian

Periode 01 Juni 2014 Sampai 30 Juni 2014

No	Kd Barang	Nama Barang	Harga	Qty	Satuan	Subtotal
1	BAT-0001	Batu Bata	Rp. 1.000	500	Pcs	Rp. 500.000
2	BAT-0002	Batu Bataka	Rp. 2.500	500	Pcs	Rp. 1.250.000
3	CAT-0001	Cat Avian Warna Biru	Rp. 30.000	500	Kaleng	Rp. 15.000.000
					Total	Rp. 16.750.000

Bandung, 25 Juni 2014
Direktur

(.....)

Halaman 1 dari 1

Gambar 4.40.Tampilan Laporan Pembelian.

4.5 Pengujian

Adapun teknik pengujian perangkat lunak yang digunakan penulis dalam membangun sistem informasi di CV. Panji Putraini, adalah teknik pengujian *blackbox testing*. Menurut Al-Bahra bin Ladjamudin (2006:379), metode *blackbox testing* ini berfokus pada keperluan fungsional dari *software*, yang memungkinkan untuk membuat himpunan kondisi *input* yang akan melatih seluruh syarat-syarat

fungsional suatu program. Uji coba *blackbox* merupakan pendekatan yang melengkapi untuk menemukan kesalahan yang terjadi.

4.5.1. Rencana Pengujian

Berikut ini adalah rencana pengujian sistem informasi ini:

Tabel 4.19. Rencana Pengujian CV. Panji Putra

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
Login	Pengecekan Login	Sistem	Blackbox
Pengisian Data	Pengisian Data Penjualan Barang	Modul	Blackbox
	Pengisian Data Penerimaan Barang	Modul	Blackbox
	Pengisian Data Barang	Modul	Blackbox
	Pengisian Data Pelanggan	Modul	Blackbox
	Pengisian Data Pengiriman	Modul	Blackbox
	Pengisian Data Order Barang	Modul	Blackbox
	Cetak Laporan Penjualan	Modul	Blackbox
	Cetak LaporanPembelian	Modul	Blackbox

4.5.2. Kasus dan Hasil Pengujian

Setelah melakukan rencana pengujian, maka hal selanjutnya yang perlu dilakukan adalah melakukan studi kasus dan bagaimana hasil dari pengujian studi kasus tersebut. Adapun, kasus dan hasil pengujian yang dilakukan oleh bagian kasir adalah sebagai berikut :

1. Pengecekan Login

Berikut ini adalah kasus dan hasil pengujian *login*:

Tabel 4.20. Kasus dan Hasil Pengujian Pengecekan *LoginUser*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : kasir Password : kasir	Tercantum pada textbox username dan textbox password.	Mengisi login sesuai dengan yang diharapkan.	[X] Diterima [ ] Ditolak
Klik tombol login	Menampilkan pesan "login berhasil" dan dapat mengakses sub menu yang ada di menu utama	Tombol dapat berfungsi sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Username:kasur Password : kasur	Tercantum pada textbox username dan textbox password.	Kasir gagal melakukan login	[X] Diterima [ ] Ditolak
Klik Tombol Login	Menampilkan pesan "ada kesalahan pada username dan password" dan user tidak dapat masuk dan mengakses ke dalam menu utama	Tombol dapat berfungsi sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak

2. Pengisian Data Penjualan Barang

Berikut ini adalah kasus dan hasil pengujian pengisian data penjualan:

Tabel 4.21. Kasus dan Hasil Pengujian Pengisian Data Penjualan.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol tambah	Tombol simpan dan batal aktif, serta no pesan akan otomatis muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data pemesanandan pelanggan diisi secara lengkap	Penyimpanan data ke dalam file data pemesanan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database data pemesanan	Muncul pesan “Input berhasil”, dan data dapat disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data-data pemesanan tidak diisi secara lengkap dan ada field yang masih kosong	Tidak dapat melakukan penyimpanan	Tidak dapat melakukan penyimpan data	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpanan	Muncul pesan “Silahkan Lengkapi Data”	[X] Diterima [ ] Ditolak

3. Pengisian Data Penerimaan Barang

Berikut ini adalah kasus dan hasil pengujian pengisian datapenerimaan barang:

Tabel 4.22. Kasus dan Hasil Pengujian Pengisian Data Penerimaan Barang.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol tambah	No penerimaan dan data order barang otomatis muncul	Dapat melakukan pengisian no pemesanan	[X] Diterima [ ] Ditolak
Field No pemesanan diisi dengan benar	Muncul data barang sesuai pemesanan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database.	Muncul pesan "Data telah Disimpan", dan stok barang bertambah.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Field No pemesanan tidak diisi / salah	data barang sesuai pemesanan tidak muncul	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpan data	Muncul pesan "Silahkan Lengkapi Data"	[X] Diterima [ ] Ditolak

4. Pengisian Data Barang

Berikut ini adalah kasus dan hasil pengujian pengisian databarang:

Tabel 4.23. Kasus dan Hasil Pengujian Pengisian Data Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol tambah	Kode barang otomatis muncul dan field-field data barang akan dapat diisi	Dapat melakukan pengisian data barang	[X] Diterima [ ] Ditolak
Isisemua field data barang	Dapat melakukan penyimpanan data barang	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database.	Muncul pesan “Data telah Disimpan”, dan data berhasil disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
field data barang tidak disi dengan lengkap	Tidak dapat melakukan penyimpanan data barang	Tidak dapat melakukan penyimpanan data barang	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpan data	Muncul pesan “Silahkan Lengkapi Data”	[X] Diterima [ ] Ditolak

5. Pengisian Data Pelanggan

Berikut ini adalah kasus dan hasil pengujian pengisian data pelanggan:

Tabel 4.24. Kasus dan Hasil Pengujian Pengisian DataPelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol tambah	Kode pelanggan otomatis muncul dan field-field data pelanggan akan dapat diisi	Dapat melakukan pengisian data pelanggan	[X] Diterima [ ] Ditolak
Isi semua field data pelanggan	Dapat melakukan penyimpanan data pelanggan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database.	Muncul pesan “Data telah Disimpan”, dan data berhasil disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
field data pelanggan tidak disi dengan lengkap	Tidak dapat melakukan penyimpanan data	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpan data	Muncul pesan “Silahkan Lengkapi Data”	[X] Diterima [ ] Ditolak

6. Pengisian DataPengiriman

Berikut ini adalah kasus dan hasil pengujian pengisian datapengiriman:

Tabel 4.25. Kasus dan Hasil Pengujian Pengisian DataPengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol tambah	No kirim otomatis muncul dan dapat melakukan pencarian data pengiriman	Dapat melakukan pengisian data pengiriman	[X] Diterima [ ] Ditolak
Field No pemesanan diisi dengan benar	Muncul data barang sesuai pemesanan yang akan dikirm	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database.	Muncul pesan “Data telah Disimpan”, dan stok barang berkurang.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Field No pemesanan tidak diisi / salah	data barang sesuai pemesanan yang akan dikirm tidak muncul	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpan data	Muncul pesan “Silahkan Lengkapi Data”	[X] Diterima [ ] Ditolak

7. Pengisian DataOrder Barang

Berikut ini adalah kasus dan hasil pengujian pengisian dataorder barang:

Tabel 4.26. Kasus dan Hasil Pengujian Pengisian DataOrder Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih *Supplier* pada *combobox*	Tombol simpan dan batal aktif, serta no pesan akan otomatis muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data pemesanandiisi secara lengkap	Penyimpanan data ke dalam file data pemesanan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database data pemesanan	Muncul pesan "Input berhasil", dan data dapat disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data-data pemesanan tidak diisi secara lengkap dan ada field yang masih kosong	Tidak dapat melakukan penyimpanan	Tidak dapat melakukan penyimpan data	[X] Diterima [ ] Ditolak
Klik tombol Simpan	Tidak dapat melakukan penyimpanan	Muncul pesan "Silahkan Lengkapi Data"	[X] Diterima [ ] Ditolak

8. CetakLaporan Penjualan

Berikut ini adalah kasus dan hasil pengujian cetak laporan penjualan:

Tabel 4.27. Kasus dan Hasil Pengujian Cetak Laporan Penjualan

Kasus Dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan periode laporanyang akan dicetak	Pencarian data ke dalam file penjualan sesuai periode	Dapat melakukan pencarian data Sesuai periode	[X] Diterima [ ] Ditolak
Klik tombol cetak	Data penjualan bisa ditemukan	Muncul laporan penjualan	[X] Diterima [ ] Ditolak
Kasus Dan Hasil Uji (Data Salah)			
Periode tidak diisi	Tidak dapat melakukan pencarian	Pencarian data gagal dilakukan	Tidak dapat melakukan pencarian
Klik tombol cetak	Data penjualantidak ditemukan	Laporan penjualan tidak muncul	Tidak dapat melakukan pencarian

9. CetakLaporan Pembelian

Berikut ini adalah kasus dan hasil pengujian cetak laporanpembelian:

Tabel 4.28. Kasus dan Hasil Pengujian Cetak LaporanPembelian

Kasus Dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan periode laporan yang akan dicetak	Pencarian data ke dalam file pembelian sesuai periode	Dapat melakukan pencarian data	[X] Diterima [ ] Ditolak
Klik tombol cari	Data bisa ditemukan	Muncul Laporan Pembelian	[X] Diterima [ ] Ditolak
Kasus Dan Hasil Uji (Data Salah)			
Periode tidak diisi	Tidak dapat melakukan pencarian	Pencarian data gagal dilakukan	Tidak dapat melakukan pencarian
Klik tombol cari	Data tidakditemukan	Laporan pembelian tidak muncul	Tidak dapat melakukan pencarian

4.5.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian yang sudah dilakukan dengan beberapa kondisi yang memungkinkan pada saat program dijalankan, maka dapat ditarik kesimpulan bahwa perangkat lunak yang dibuat sudah berjalan dengan cukup baik sebagaimana yang diharapkan oleh perusahaan.

Normalisasi

Normalisasi merupakan suatu proses pengelompokan data elemen menjadi table-tabel yang menunjukan *entity* dan relasinya yang berfungsi untuk menghilangkan redudansi data, menentukan *key* yang unik untuk mengakses data atau merupakan pembentukan *relation* sedemikian rupa sehingga *database* tersebut mudah di modifikasi.

1. Bentuk tidak normal (*unnormal*)

   Pembelian&Penjualan={no_po, kd_brg, jns_brg, nm_brg, qty, kd_brg, jns_brg, nm,_brg, qty, kd_brg, jns_brg, nm_brg, kd_supplier ,kd_brg, qty, no_nota, no_po, tgl, kd_brg, nm_brg, qty, total, no_pesan, tgl, kd_brg, nm_brg, qty, kd_ brg, jns_brg, nm,_brg, qty, no_po, tgl,kd_pelanggan, hrg_jual, hrg_beli, no_kirim, no_pesan, tgl, total}

2. Bentuk normal pertama

   Pembelian&Penjualan={kd_brg, nm_brg, jns_brg, hrg_jual, hrg_beli, stok, no_pesan, tgl, harga, qty, total, no_po, tgl,kd_supplier, no_faktur, kd_pelanggan,no_kirim,total}

3. Bentuk normal kedua

   Data Pembelian = { no_po*,tgl, kd_brg**,nm_brg,jns_brg, qty, hrg_beli, total, kd_supplier**}

   Data Penjualan= {no_pesan*, tgl, kd_brg**,nm_brg,jns_brg, qty, hrg_jual, total, kd_pelanggan**, no_nota}

4. Bentuk normal ketiga

DataBarang={kd_brg*, nm_brg, jns_brg, hrg_jual, hrg_beli,stok}

Data Pembelian= { no_faktur*, tgl, no_po**, kd_supplier**,total}

Data Penjualan= {no_nota*, tgl, no_pesan**, kasir, total, bayar, cek}

DataSupplier = {kd_supplier*,nama_supplier,alamat,no_tlp}

DataPelanggan = {kd_pelanggan*, nama, alamat,no_telp}

DataPengiriman = {no_kirim*, no_pesan**, tgl}

Data Pemesanan= { no_pesan*, tgl, kd_pelanggan**,total,

kd_brg**,qty}

Data Order= {no_po*, tgl, kd_supplier**, kd_brg**, total, qty}