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 perangcangan menu pada *form* yang dibutuhkan.

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 memberikan gambaran secara umum kepada *user* tentang sistem baru dan menghasilkan sistem yang sesuai dengan kebutuhan dan diharapkan dapat membantu proses pengelolaan data penerimaan barang ke bagian GBB hingga pengeluaran produk jadi ke marketing di PT. Sanbe Farma Sterile Bandung.

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 data penerimaan barang ke bagian GBB hingga pengeluaran produk jadi ke marketing di PT.Sanbe Farma Sterile Bandung bisa dilakukan dengan 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.Perbedaan antara prosedur yang lama dan yang diusulkan adalah jika yang lama menggunakan cara manual dan dokumen. Sedangkan sistem informasi yang baru menggunakan alat bantu komputer dan dilakukan secara otomatis. Berikut adalah proses atau prosedur diusulkan :

A. Proses Penerimaan Barang di Gudang Bahan Baku (GBB)

1. Datang barang dari supplier.
2. Supplier memberikan tanda terima surat *Delivery Order* (DO) ke bagian GBB, kemudian dijadikan arsip.
3. Bagian GBB memeriksa surat DO dengan barang yang diterima.
4. Kemudian dibuatlah Bukti Penerimaan Barang (BPB) untuk label penandaan setiap barang yang datang, setelah itu dibuat laporan stok barang yang akan diserahkan pada manager.

B. Proses Permintaan Bahan Baku dan Bahan Kemas dari Produksi ke Gudang Bahan Baku (GBB)

1. Bag. Produksi menerima *Marketing Order* (MO) dari bagian marketing.
2. Bagian Produksi merancang jadwal produksi.
3. Bagian produksi akan membuat sekaligus menyerahkan bon permintaan bahan dan diserahkan ke bagian GBB.
4. Bagian GBB menyerahkan bahan yang diminta oleh produksi.

C. Proses Penerimaan Produk Jadi dari Bagian Produksi (*Packing)* ke Bagian Gudang Obat Jadi (GOJ)

1. Setelah proses produksi selesai hingga *packing* , bahan mentah semula berubah menjadi produk jadi. Bagian produksi menyerahkan barangnya ke bagian GOJ. Dengan menyerahkan Surat Penyerahan Obat Jadi (SPOJ).
2. Setelah barang diterima dan di periksa, bagian GOJ membuat Laporan Stok Barang yang akan diserahkan ke manager.

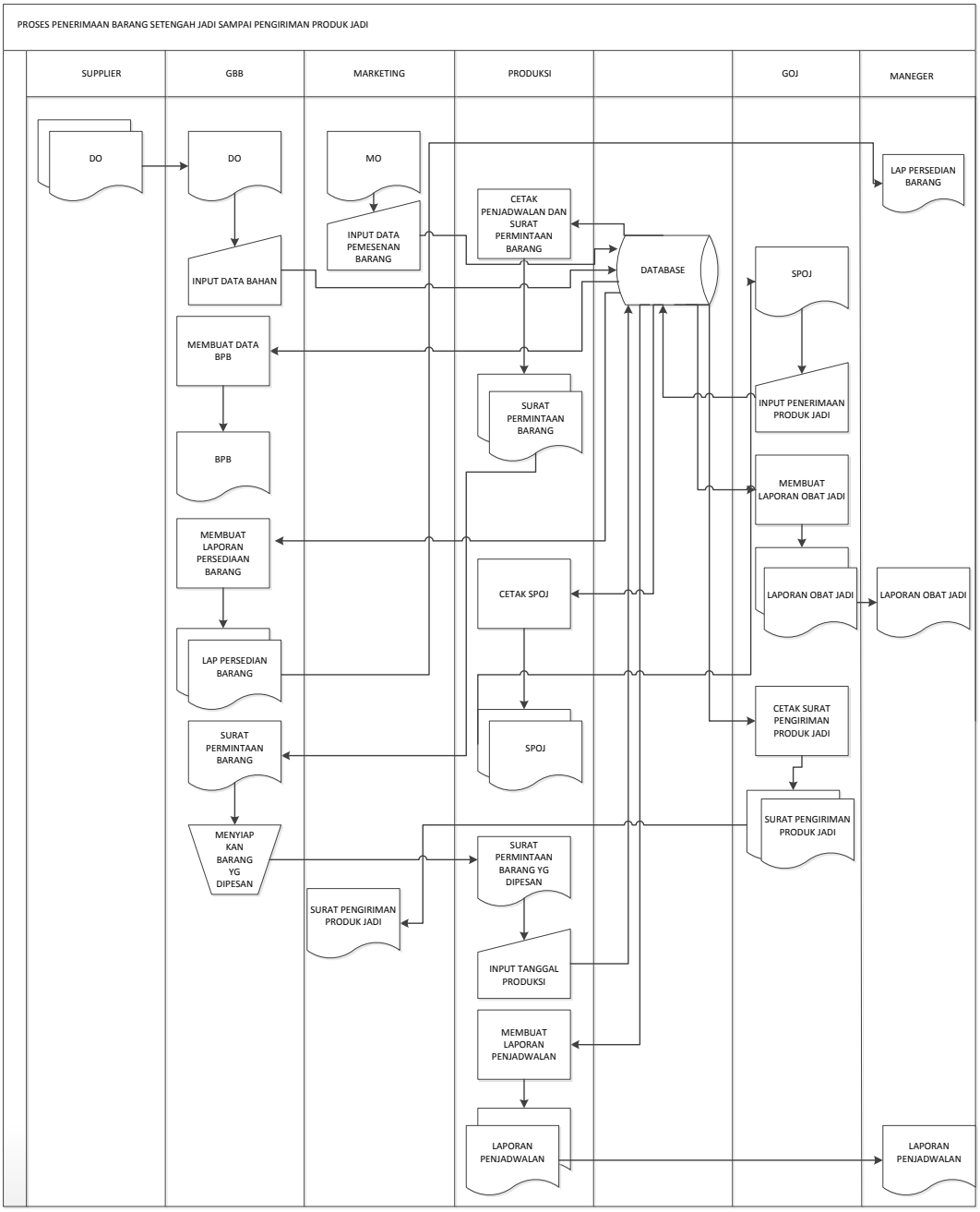
D. Proses Pengeluaran Produk Jadi dari Gudang Obat Jadi (GOJ) ke Marketing

1. Bagian produksi memberikan daftar list barang yang harus dikeluarkan bagian GOJ ke marketing.
2. Bagian GOJ membuat Bon Pengiriman Barang (*Delivery Order*) dari daftar list barang yang harus di keluarkan dan diserahkan ke bagian marketing.

Prosedur yang lama dan baru tidak jauh berbeda hanya pengolahan datanya saja yang berubah dari manual menjadisistem terkomputerisasi.

4.1.3.1.Flow Map

Flowmap menggambarkan aliran dan informasi antar area didalam sebuah organisasi.Flowmap ini menelusuri sebuah dokumen dari asalnya sampai tujuannya.Secara rinci flowmap ini menunjukkan dari mana dokumen tersebut berasal, distribusinya, dan tujuan digunakannya dokumen tersebut.Berikut ini gambar flowmap prosedur Penerimaan yang diusulkan.



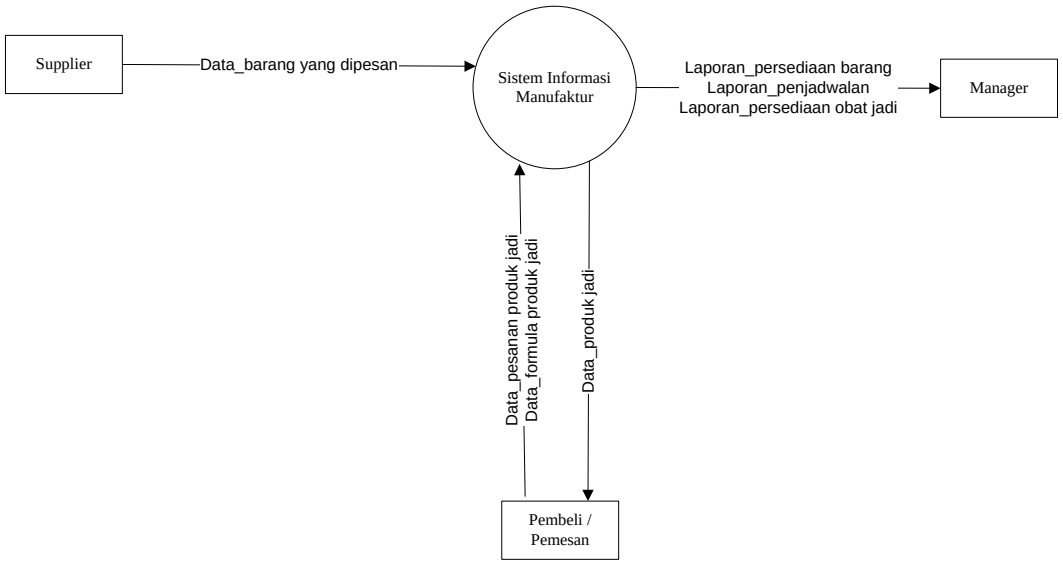
Gamabr 4.1 Flowmap prosedur penerimaan bahan baku sampai pengiriman produk jadi yang diusulkan

4.1.3.2. Diagram Konteks

Diagram konteks merupakan diagram tingkat tinggi yang menggambarkan seluruh jaringan baik masukan maupun keluaran dari suatu sistem. Tujuan

pembuatan diagram konteks yaitu untuk memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya. Pada diagram konteks akan terlihat bagaimana arus data yang masuk dan bagaimana arus data yang keluar dari entitas luar yang mempengaruhi sistem.

Dibawah ini adalah diagram konteks yang diusulkan secara garis besar dan dapat lihat sebagai berikut :



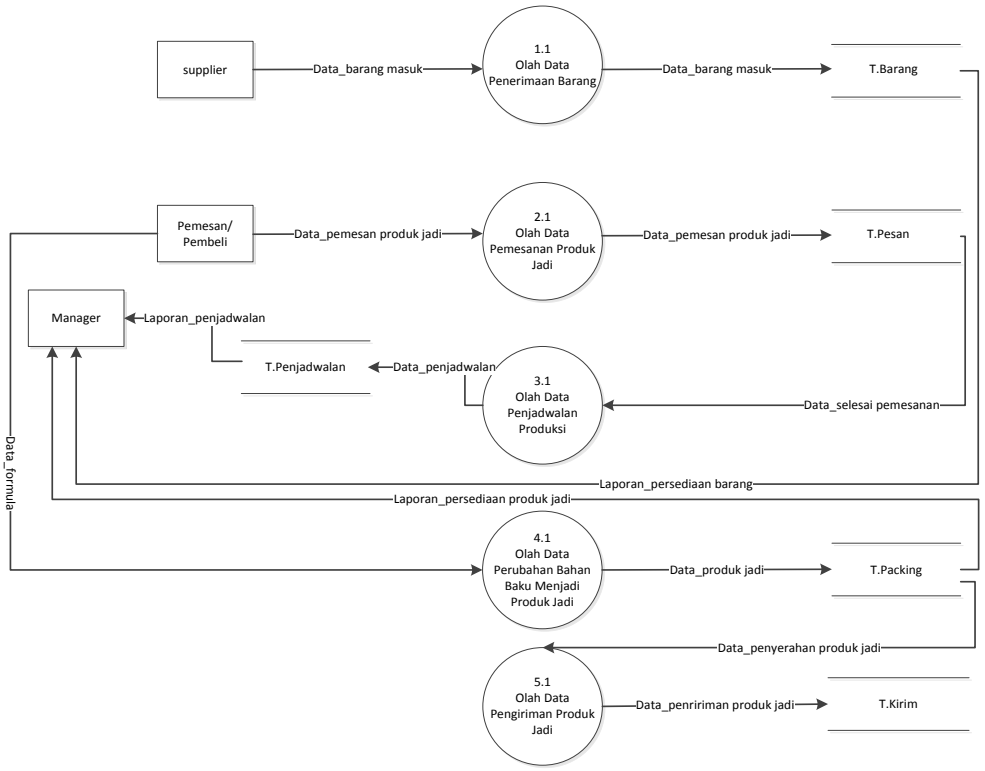
Gambar 4.2 Diagram konteks yang diusulkan

4.1.3.3. Data Flow Diagram

Data Flow Diagram (DFD) merupakan proses yang menggambarkan suatu alir informasi yang lebih detail dan terperinci yang merupakan pengembangan dari diagram konteks.

4.1.3.3.1. Data Flow Diagram (DFD) Level 1

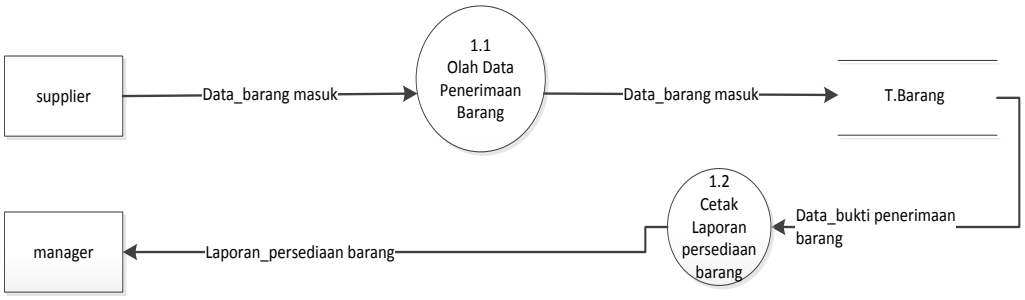
Pada DFD level 1 mengambarkan proses-proses yang ada pada pada sistem informasi penerimaan dan pengeluaran barang.Berikut ini gambar DFD Level 1 yang diusulkan :



Gambar 4.3 DFD level 1 pada sistem informasi manufaktur yang diusulkan

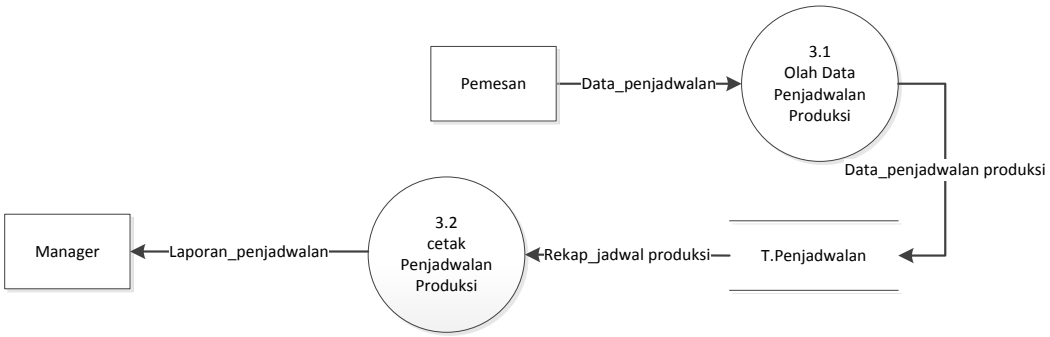
4.1.3.3.2. Data Flow Diagram (DFD) Level 2

Pada DFD level 2 mengambarkan setiap proses yang ada pada sistem informasi manufaktur. Berikut ini gambar DFD Level 2 yang diusulkan :



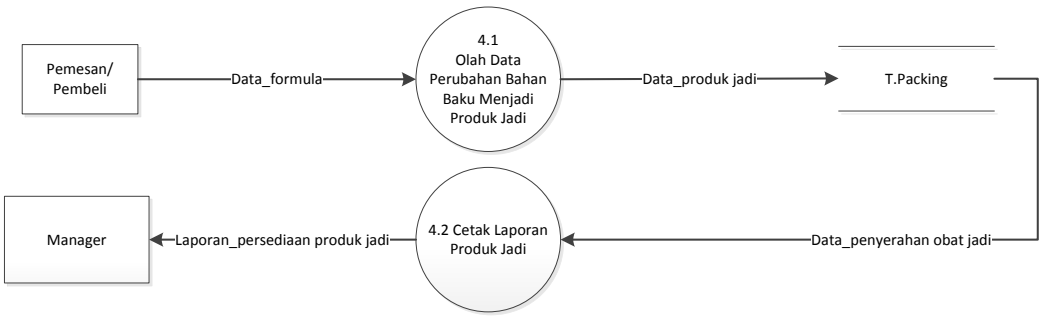
Gambar 4.4 DFD level 2 proses 1.0 penerimaan bahan yang diusulkan.

4.1.3.3.3. DFD LEVEL 2



Gambar 4.5 DFD level 2 proses 3.0 penjadwalan produksi yang diusulkan

4.1.3.3.4. DFD LEVEL 2



Gambar 4.6 DFD level 2 proses 4.0 perubahan bahan baku menjadi produk jadi yang diusulkan

4.1.3.4. Kamus Data

Kamus data adalah penjabaran dari aliran-aliran data yang ada di dalam sebuah *data flow diagram*. Berikut ini kamus data yang ada pada sistem informasi penerimaan dan pengeluaran barang :

1. Nama data : Data Bahan
   Alias : -
   Aliran data : Proses 1.1-GBB
   Bentuk data : Label
   Struktur data : no_masuk, kode_barang, qty
2. Nama data : Laporan Persediaan Bahan
   Alias : -
   Aliran data : Proses 1.2-GBB
   Bentuk data : Dokumen
   Struktur data : kode_barang, jenis_barang, nama_barang, stok stok_min, satuan
3. Nama data : Data Pemesanan Barang
   Alias : -
   Aliran data : Proses 2.1-Marketing
   Bentuk data : Dokumen
   Struktur data : no_pesan, kode_produk, qty

4. Nama data : Laporan Jadwal

   Alias : -

   Aliran data : Proses 3.2-Produksi

   Bentuk data : Dokumen

   Struktur data : kode_produksi, no_pesan, weighing, mixing, packing, status

5. Nama data : Data Formula

   Alias : -

   Aliran data : Proses 4.1-Produksi

   Bentuk data : Dokumen

   Struktur data : kode_formula, kode_barang, qty, satuan

6. Nama data : Data Laporan Produk Jadi

   Alias : -

   Aliran data : Proses 5.1-GOJ

   Bentuk data : Dokumen

   Struktur data : kode_produk, nama_produk, stok

7. Nama data : Bon pengiriman Barang

   Alias : -

   Aliran data : Proses 5.1-GOJ

   Bentuk data : Dokumen

   Struktur data : no_kirim, no_pesan, tanggal

4.1.4. Perancangan Basis Data

Sebagai penunjang sistem pengolahan data dengan bantuan komputer, maka harus ditentukan bagaimana bentuk rancangan *database* yang digunakan.Perancangna *database* ini dimaksudkan untuk mengidentifikasi kebutuhan-kebutuhan file *database* sistem yang diperlukan.

4.1.4.1.Normalisasi

Normalisasi merupakan suatu proses pengelompokan data elemen menjadi tabel-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*)

   Penerimaan&Pengiriman={~~kode_barang~~, jenis_barang, nama_barang, ~~stok~~, stok_min, ~~satuan~~, ~~kode_formula~~, ~~kode_produk~~, ~~kode_formula~~, ~~kode_barang~~, ~~qty~~, ~~satuan~~, ~~kode_kemas~~, ~~tanggal~~, ~~kode_produksi~~, ~~kode_kemas~~, ~~kode_produk~~, ~~qty~~, no_kirim, ~~no_pesan~~, ~~tanggal~~, ~~no_masuk~~, ~~tanggal~~, supplier, ~~no_masuk~~, ~~kode_barang~~, ~~qty~~, ~~kd_buyer~~, nama, no_tel, alamat, ~~no_pesan~~, ~~kd_buyer~~, ~~tanggal~~, status, ~~no_pesan~~, ~~kode_produk~~, ~~qty~~, ~~kode_produk~~, nama_produk, ~~stok~~, ~~kode_produksi~~, ~~no_pesan~~, weighing, mixing, packing, status, tgl_selesai }

2. Bentuk normal pertama

   Penerimaan&Pengiriman={kode_barang, jenis_barang, nama_barang,

stok, stok_min, satuan, kode_kemas, kode_formula, kode_produk, qty, no_kirim, no_pesan, tanggal, no_masuk, supplier, kd_buyer, nama, no_tel, alamat, status, nama_produk, kode_produksi, weighing, mixing, packing, status, tgl_selesai}

3. Bentuk normal kedua

Data Pengiriman={kode_formula, kode_produk**, kode_barang, qty, satuan, no_kirim*, no_pesan, tanggal, kd_buyer, nama, no_telp, alamat, status, nama_produk, stok, kode_produksi, weighing, mixing, packing, tgl_selesai}

Data Penerimaan={kode_barang*, jenis_barang, nama_barang, stok, stok_min, satuan, kode_kemas, kode_produksi, kode_produk, no_masuk**, tanggal, supplier, qty}

4. Bentuk normal ketiga

Data Barang ={kode_barang*, jenis_barang, nama_barang, stok, stok_min, satuan}

Data Formula ={kode_formula*, kode_produk**, kode_barang, qty, satuan}

Data Kemas ={kode_kemas*, tanggal, kode_produksi, kode_produk, qty}

Data Kirim ={no_kirim*, no_pesan**, tanggal}

Data masuk ={no_masuk*, tanggal, kode_barang, supplier, qty}

Data Pemesan ={kd_buyer*, nama, no_telp, alamat}

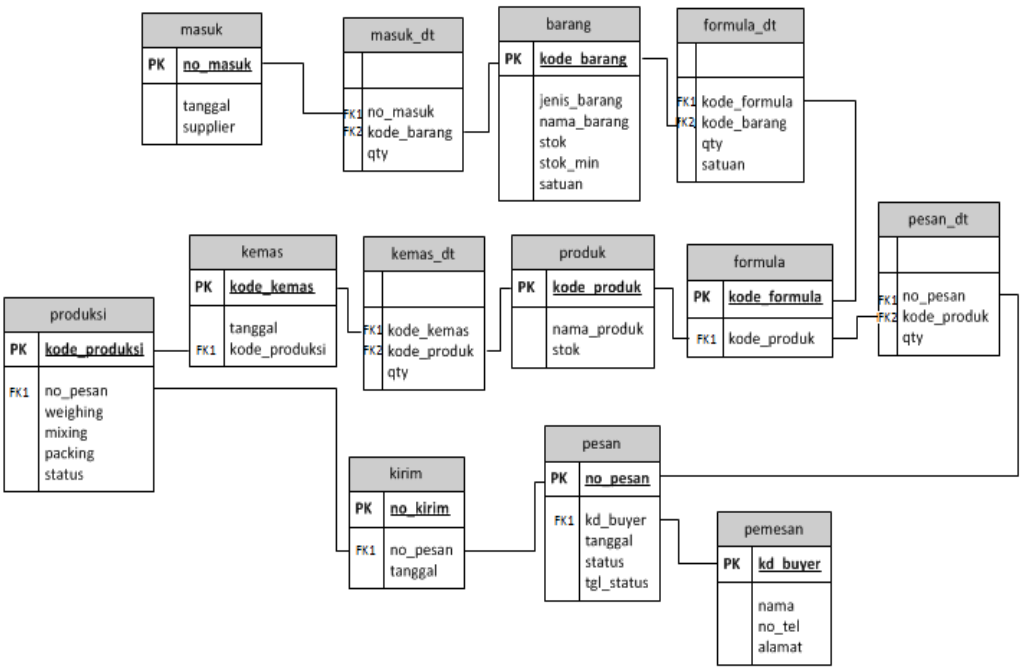
Data Pesan ={no_pesan*, kd_buyer, kode_produk, qty, tanggal, status, tgl_selesai}

Data Produk ={kode_produk*, nama_produk, stok}

Data Produksi ={kode_produksi*, no_pesan**, weighing, mixing, packing, status}

4.1.4.2. Relasi Tabel

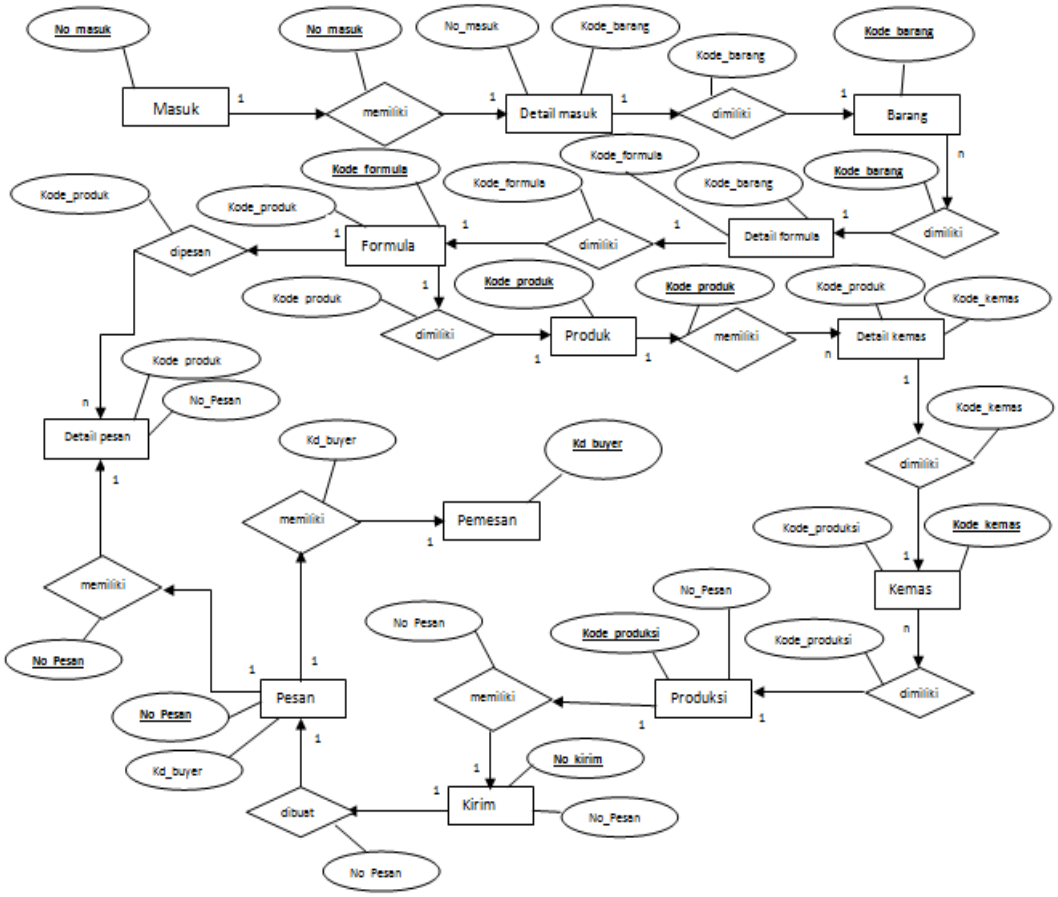
Relasi antar tabel menggambarkan hubungan antar table-tabel yang ada pada suatu sistem pengolahan data. Gambar hubungan relasi antar table pada sistem informasi pengolahan data di PT. Sanbe Farma Sterile Bandung.



Gambar 4.7 Relasi Tabel

4.1.4.3. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan gambaran hubungan antar entitas yang satu dengan yang lain yang saling berhubungan dan saling berinteraksi satu dengan yang lain. Gambar ERD sebagai berikut:



Gambar 4.8 *Entity Relationship Diagram*

4.1.4.4. 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 manufaktur di PT. Sanbe Farma Sterile Bandung.

1. File Barang

   a. Nama File : barang

   b. Primary Key : kode_barang

   c. Jumlah Field : 6

Tabel 4.1 File Barang

No	Field Name	Type	Size	Descrpition
1	Kode_barang*	Varchar	15	Kode barang
2	Jenis_barang	Varchar	100	Jenis barang
3	Nama_barang	Varchar	200	Nama barang
4	Stok	Double	-	Stok
5	Stok_min	Double	-	Stok minimal
6	Satuan	Varchar	20	Satuan

2. File Formula

   a. Nama File : formula

   b. Primary Key : kode_formula

   c. Jumlah Field : 2

Tabel 4.2 File Formula

No	Field Name	Type	Size	Descrpition
1	Kode_formula*	Varchar	15	Kode formula
2	Kode_produk	Varchar	15	Kode produk

3. File Detail Formula

   a. Nama File : formula_dt

   b. Foreign Key : kode_formula

   c. Jumlah Field : 4

Tabel 4.3 File Detail Formula

No	Field Name	Type	Size	Descrpition
1	Kode_formula**	Varchar	15	Kode formula
2	Kode_barang**	Varchar	15	Kode barang
3	Qty	Double	-	Jumlah barang
4	Satuan	Varchar	30	Satuan

4. File Kemas

   a. Nama File : kemas

   b. Primary Key : kode_kemas

   c. Jumlah Field : 3

Tabel 4.4 File Kemas

No	Field Name	Type	Size	Descrpition
1	Kode_kemas*	Varchar	15	Kode kemas
2	Tanggal	Date	-	Tanggal
3	Kode_produksi	Varchar	15	Kode produksi

5. File Detail Kemas

   a. Nama File : kemas_dt

   b. Foreign Key : Kode_kemas

   c. Jumlah Field : 3

Tabel 4.5 File Detail Kemas

No	Field Name	Type	Size	Descrpition
1	Kode_kemas**	Varchar	15	Kode kemas
2	Kode_produk	Varchar	15	Kode produk
3	Qty	Int	11	Jumlah pengemasan produk

6. File Kirim

   a. Nama File : kirim

   b. Primary Key : no_kirim

   c. Jumlah Field : 3

Tabel 4.6 File Kirim

No	Field Name	Type	Size	Descrpition
1	No_kirim*	Varchar	15	No kirim
2	No_pesan	Varchar	15	No pesan
3	Tanggal	Date	-	Tanggal

7. File Masuk

   a. Nama File : masuk

   b. Primary Key : no_masuk

   c. Jumlah Field : 3

Tabel 4.7 File Masuk

No	Field Name	Type	Size	Descrpition
1	No_masuk	Varchar	15	No masuk
2	Tanggal	Date	-	Tanggal
3	Supplier	Varchar	100	Supplier

8. File Detail Masuk

a. Nama File : masuk_dt

b. Foreign Key : no_masuk

c. Jumlah Field : 3

Tabel 4.8 File Detail masuk

No	Field Name	Type	Size	Descrpition
1	No_masuk**	Varchar	15	No masuk
2	Kode_barang	Varchar	15	Kode barang
3	Qty	Double	-	Jumlah barang masuk

9. File Pemesan

a. Nama File : pemesan

b. Primary Key : kd_buyer

c. Jumlah Field : 4

Tabel 4.9 File Pemesan

No	Field Name	Type	Size	Descrpition
1	Kd_buyer*	Varchar	15	Kode pemesan
2	Nama	Varchar	100	Nama pemesan
3	No_tel	Varchar	13	No telepon
4	Alamat	Varchar	200	Alamat

10. File Pesan

a. Nama File : pesan

b. Primary Key : no_pesan

c. Jumlah Field : 4

Tabel 4.10 File Pesan

No	Field Name	Type	Size	Descrpition
1	No_pesan*	Varchar	15	No pesan
2	Kd_buyer	Varchar	10	Kode pemesan
3	Tanggal	Date	-	Tanggal
4	Status	Int	11	Status

11. File Detail Pesan

a. Nama File : pesan_dt

b. Foreign Key : no_pesan

c. Jumlah Field : 3

Tabel 4.11 File Detail Pesan

No	Field Name	Type	Size	Descrpition
1	No_pesan**	Varchar	15	No pesan
2	Kode_produk	Varchar	15	Kode produk
3	Qty	Int	11	Jumlah pesan

12. File Produk

a. Nama File : produk

b. Primary Key : kode_produk

c. Jumlah Field : 3

Tabel 4.12 File Produk

No	Field Name	Type	Size	Descrpition
1	Kode_produk*	Varchar	15	Kode produk
2	Nama_produk	Varchar	200	Nama produk
3	Stok	Int	11	Stok produk

13. File Produksi

a. Nama File : Produksi

b. Primary Key : kode_produksi

c. Jumlah Field : 6

Tabel 4.13 File Produksi

No	Field Name	Type	Size	Descrpition
1	Kode_produksi*	Varchar	15	Kode produksi
2	No_pesan	Varchar	15	No pesan
3	Weighing	Date	-	Tanggal Penimbangan
4	Mixing	Date	-	Tanggal pengadukan
5	Packing	Date	-	Tanggal pengemasan
6	Status	Int	11	Status

4.1.4.5. 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 barang

Kode Jenis Bahan Baku

Contoh : BRG-0001

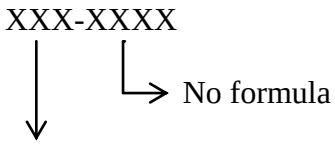
XXX-XXXX

↓ → No barang

Kode Jenis Bahan Kemas

Contoh : KMS-0001

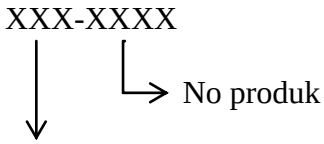
2. Kode Formula



Kode Formula

Contoh : FRM-0001

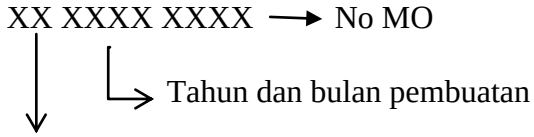
3. Kode Produk



Kode Produk

Contoh : PRD-0001

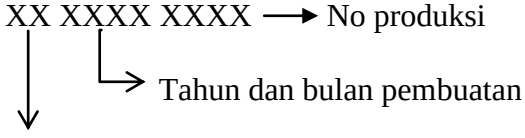
4. No Pesan



Kode pesan / *Marketing Order* (MO)

Contoh : MO14060001

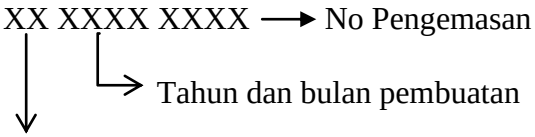
5. Kode Produksi



Kode Produksi / *Schedule* (SC)

Contoh : SC14060001

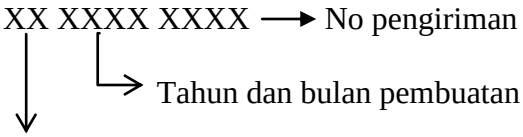
6. Kode Kemas



Kode Pengemasan / *Packing* (PC)

Contoh : PC14060001

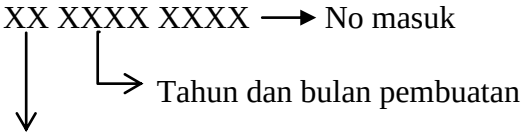
7. No Kirim



Kode Pengiriman

Contoh : KR14060001

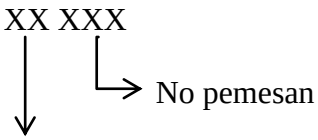
8. No Masuk



Kode Masuk

Contoh : MS14060001

9. Kode Pemesan / *Buyer*



Kode Pemesan / *buyer* (BY)

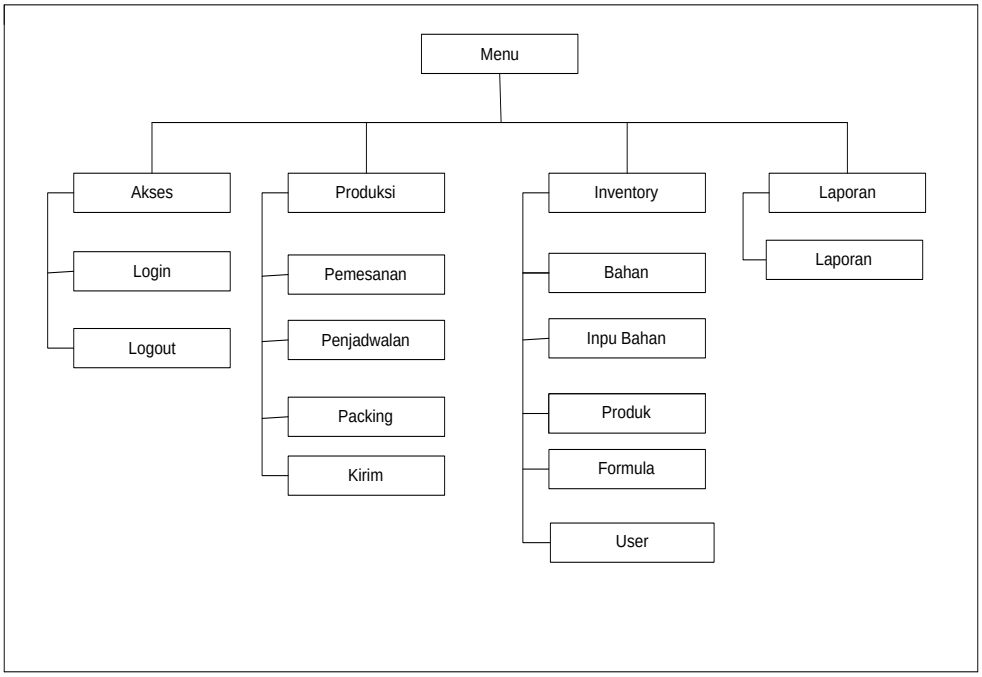
Contoh : BY001

4.2.Perancangan Antar Muka

Perancangan antar muka dirancang untuk memudahkan pemakaian dalam mengakses informasi yang dibutuhkan.

4.2.1. Struktur Menu

Rancangan struktur menu dibuat untuk memudahkan *user* dalam melakukan penggunaan fungsi-fungsi program yang ada pada sistem ini.Adapun struktur menu dari aplikasi yang dibuat dapat dilihat pada gambar berikut ini:

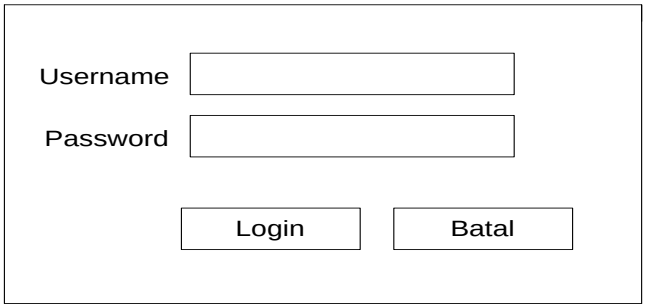


Gambar 4.9 Rancangan Struktur Menu Aplikasi.

4.2.2. Perancangan Input

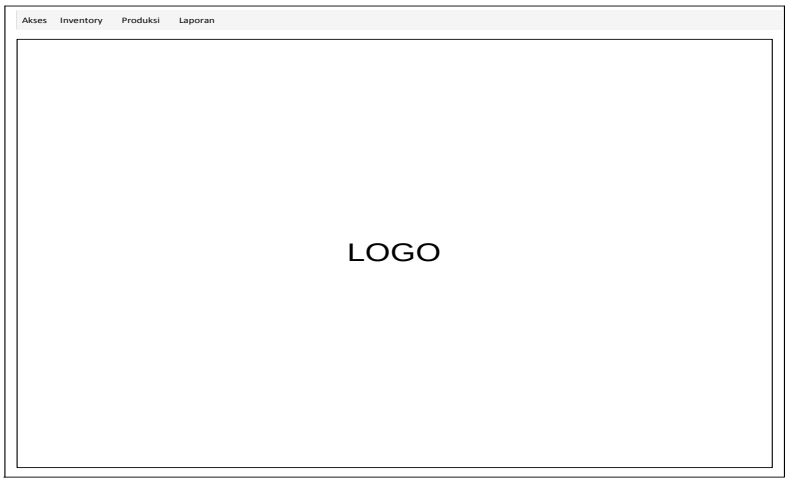
Untuk mencegah tidak akuratnya dari suatu hasil sistem informasi, maka perlu memperhitungkan data apa yang harus diinputkan kedalam sistem, sehingga hasilnya dapat berguna secara optimal dalam pengambilan keputusan khususnya penyelesaian suatu permasalahan yang dihadapi. Adapun rancangan inputnya adalah sebagai berikut :

1. Form Login



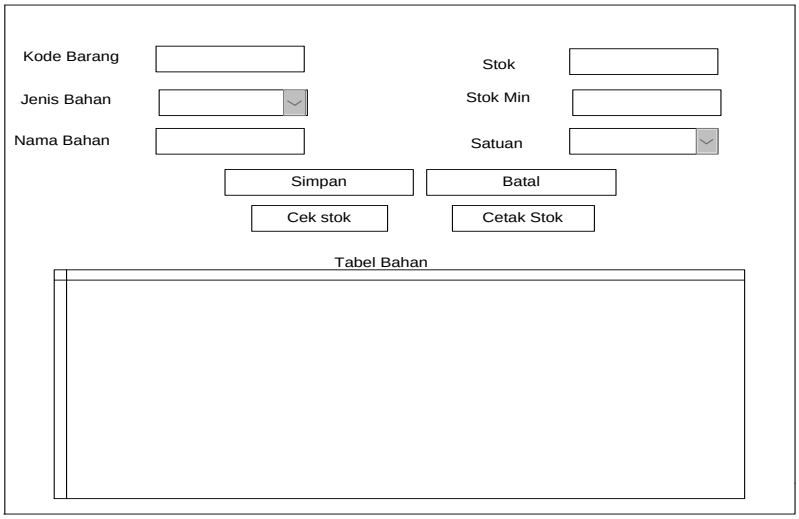
Gambar 4.10 Rancangan Form Login

2. Form utama



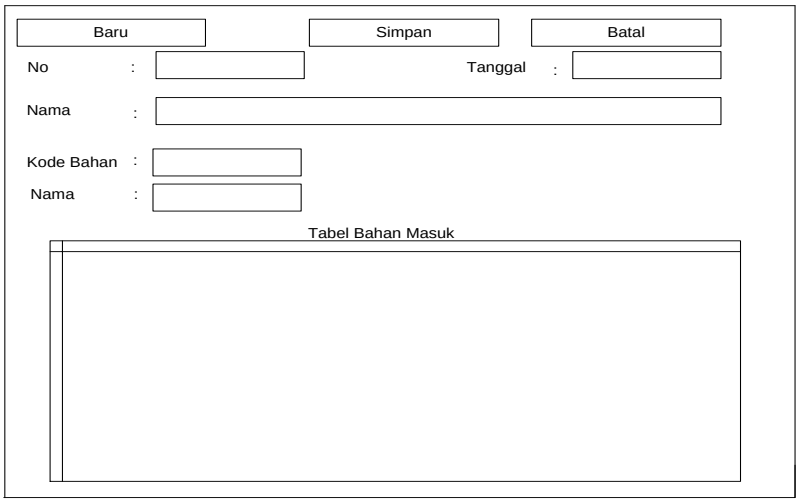
Gambar 4.11 Rancangan Form Utama

3. Form Bahan



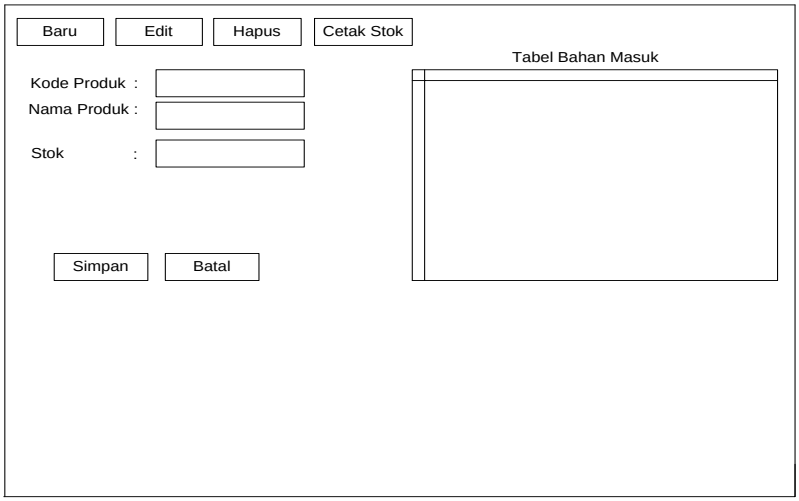
Gambar 4.12 Rancangan Form Bahan

4. Form Input Bahan



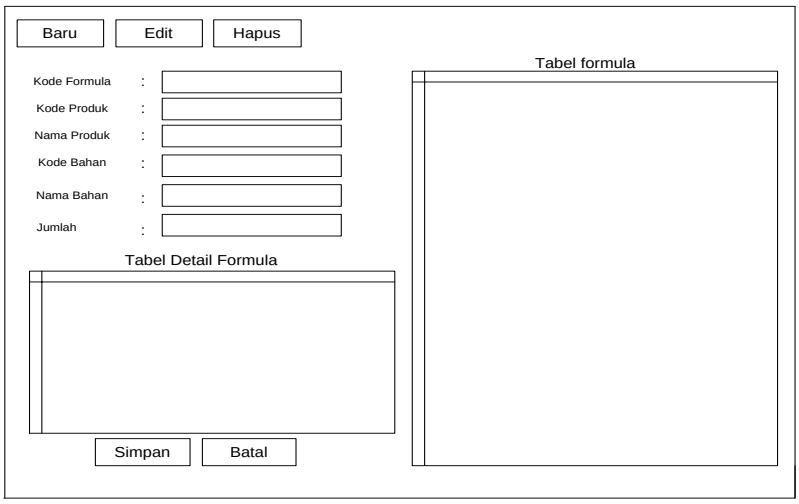
Gambar 4.13 Rancangan Form Input Bahan

5. Form Produk



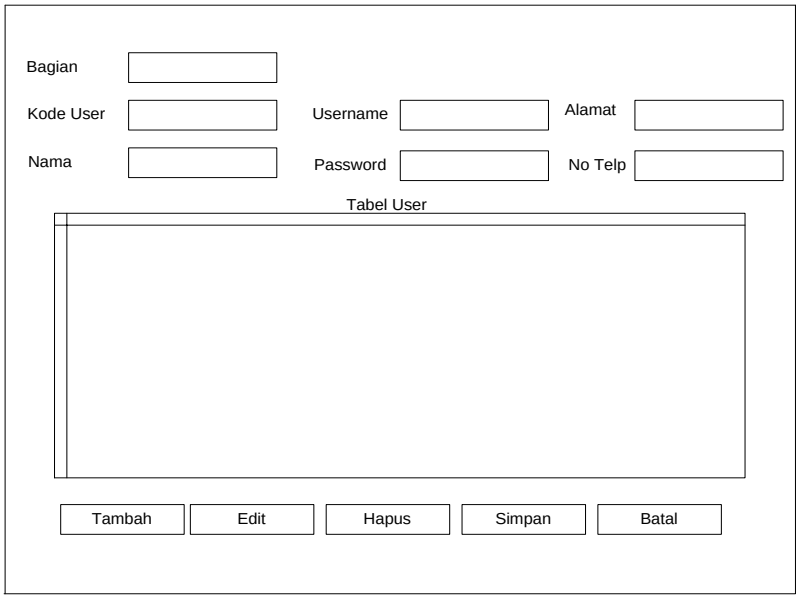
Gambar 4.14 Rancangan Form Produk

6. Form Formula



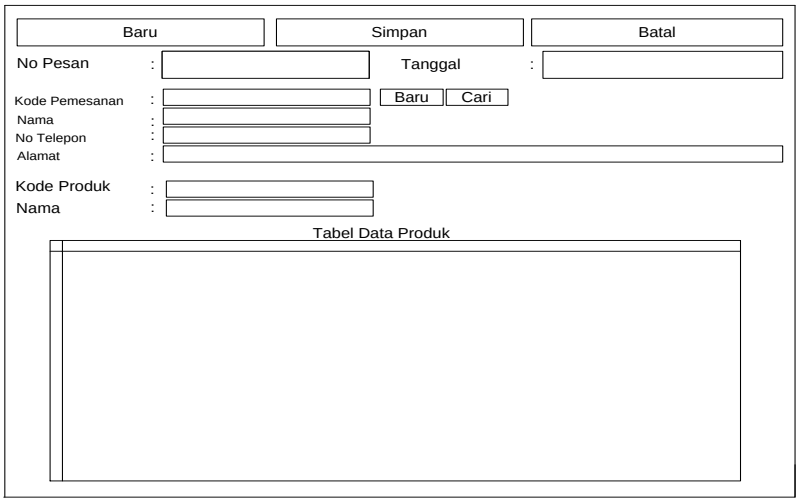
Gambar 4.15 Rancangan Form Formula

7. Form User



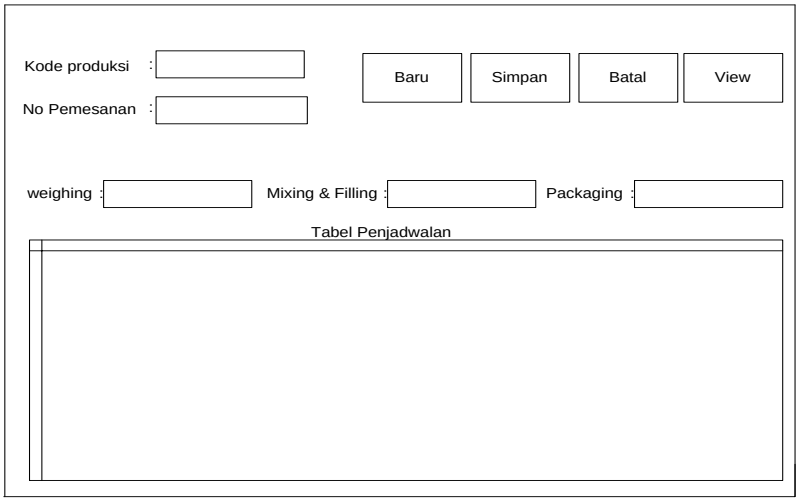
Gambar 4.16 Rancangan Form User

8. Form Pemesanan



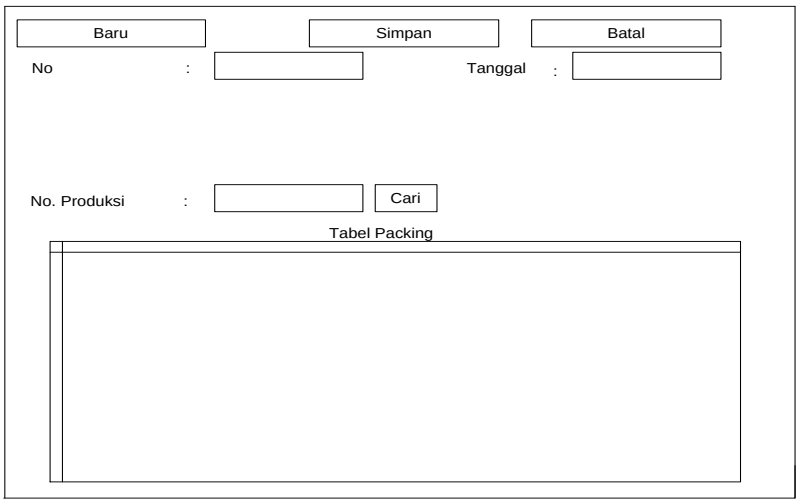
Gambar 4.17 Rancangan Form Pemesanan

9. Form Penjadwalan



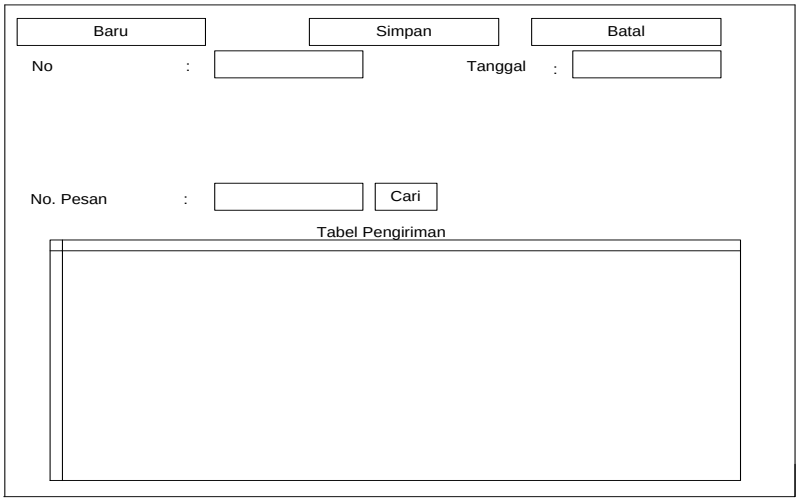
Gambar 4.18 Rancangan Form Penjadwalan

10. Form Packing



Gambar 4.19 Rancangan Form Packing

11. Form Pengiriman

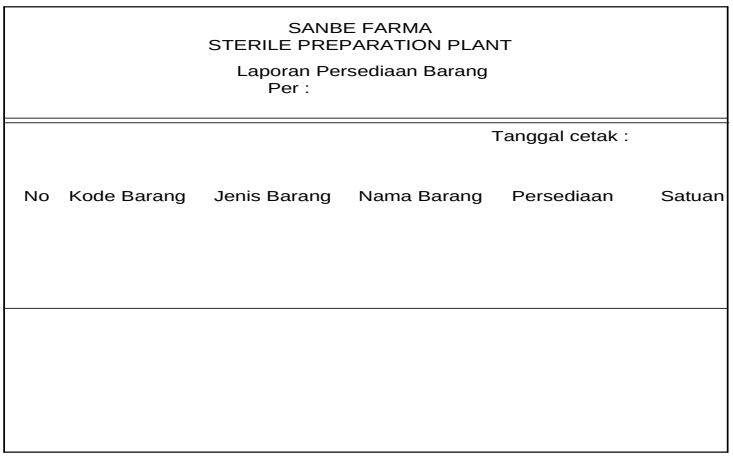


Gambar 4.20 Rancangan Form Pengiriman

4.2.3. Perancangan Output

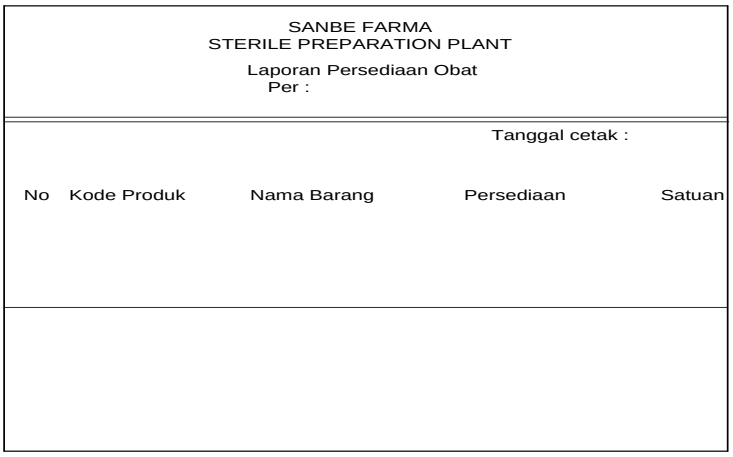
Perancangan *output* adalah informasi yang dihasilkan oleh sistem berupa laporan hasil dari proses *input*-an yang diterima oleh sistem. Adapun perancangan *output* dari sistem informasi ini adalah sebagai berikut:

1. Laporan Stok Bahan



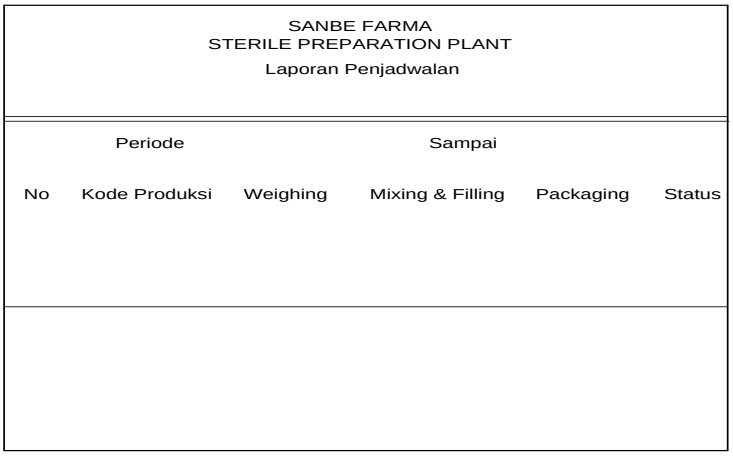
Gambar 4.21 Laporan Stok Barang

2. Laporan Persediaan Obat



Gambar 4.22 Tampilan Laporan Persedian Obat

3. Laporan Penjadwalan



Gambar 4.23 Tampilan Penjadwalan

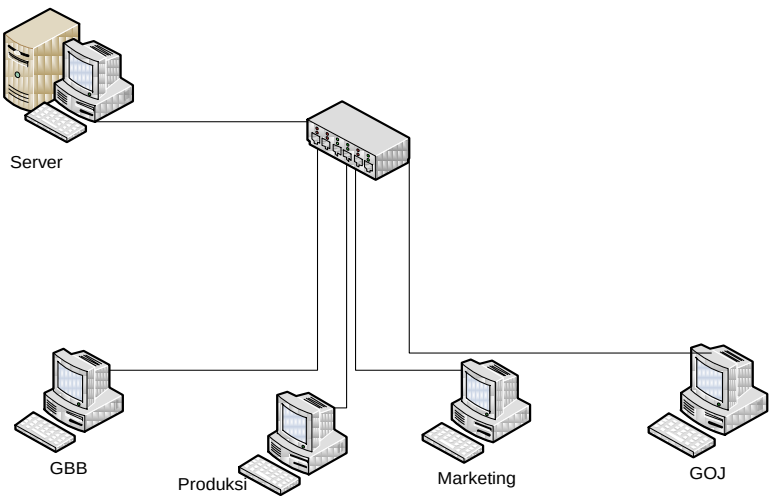
4.1 Perancangan Arsitektur Jaringan

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 PT. Sanbe Farma Sterile Bandung 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 4.24 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. Batasan Implementasi

Ada beberapa hal yang menjadi batasan implementasi Perancangan Aplikasi sistem informasi manufaktur PT. Sanbe Farma Sterile Bandung, diantaranya adalah :

a. Hanya membahas tentang seputar penerimaan bahan mentah ke GBB. Tidak membahas tentang pemesanan bahan mentahnya ke supplier.

b. Pemesanan produk dari bagian marketing ke produksi. Tidak membahas transaksi pembelian dari pemesan (*buyer*)

c. Penginputan tanggal proses produksi, perubahan bahan mentah menjadi produk jadi.

d. Penerimaan produk jadi dari bagian produksi ke GOJ. Selanjutnya pengiriman produk- produk jadi yang di pesan oleh bagian marketing.

e. Pembuatan laporan persediaan barang, Laporan persediaan obat dan laporan penjadwalan per periode. Yang diperoleh dari data yang sudah diinput dengan aplikasi tersebut.

4.4.2. 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.

4.4.3. Implementasi Perangkat Keras

Adapun perangkat 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

4.4.4. Implementasi Basis Data (Sintaks SQL)

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
```

```
--
-- Host: 127.0.0.1
-- Generation Time: Aug 10, 2014 at 03:23 AM
-- Server version: 5.5.27
-- PHP Version: 5.4.7
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";
/*!40101SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIEN
T */;
/*!40101SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RES
ULTS */;
/*!40101SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTI
ON */;
/*!40101 SET NAMES utf8 */;
--
-- Database: `sanbe`
--
-- --------------------------------------------------------
--
-- Table structure for table `barang`
```

```
--
CREATE TABLE IF NOT EXISTS `barang` (
  `kode_barang` varchar(15) NOT NULL,
  `jenis_barang` varchar(100) NOT NULL,
  `nama_barang` varchar(200) NOT NULL,
  `stok` double NOT NULL,
  `stok_min` double NOT NULL,
  `satuan` varchar(20) NOT NULL,
  PRIMARY KEY (`kode_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `barang`
--
-- --------------------------------------------------------
--
-- Table structure for table `formula`
--
CREATE TABLE IF NOT EXISTS `formula` (
  `kode_formula` varchar(15) NOT NULL,
  `kode_produk` varchar(15) NOT NULL,
  PRIMARY KEY (`kode_formula`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--

-- Dumping data for table `formula`

--

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

--

-- Table structure for table `formula_dt`

--

CREATE TABLE IF NOT EXISTS `formula_dt` (

  `kode_formula` varchar(15) NOT NULL,

  `kode_barang` varchar(15) NOT NULL,

  `qty` double NOT NULL,

  `satuan` varchar(30) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `formula_dt`

--

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

--

-- Table structure for table `kemas`

--

CREATE TABLE IF NOT EXISTS `kemas` (

  `kode_kemas` varchar(15) NOT NULL,

  `tanggal` date NOT NULL,
```

```
  `kode_produksi` varchar(15) NOT NULL,
  PRIMARY KEY (`kode_kemas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `kemas`
--
-- --------------------------------------------------------
--
-- Table structure for table `kemas_dt`
--
CREATE TABLE IF NOT EXISTS `kemas_dt` (
  `kode_kemas` varchar(15) NOT NULL,
  `kode_produk` varchar(15) NOT NULL,
  `qty` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `kemas_dt`
--
-- --------------------------------------------------------
--
-- Table structure for table `kirim`
--
```

```
CREATE TABLE IF NOT EXISTS `kirim` (
  `no_kirim` varchar(15) NOT NULL,
  `no_pesan` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`no_kirim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `kirim`
--
-- --------------------------------------------------
--
-- Table structure for table `login`
--
CREATE TABLE IF NOT EXISTS `login` (
  `kd_user` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  `log` time NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `login`
--
-- --------------------------------------------------
```

```
--
-- Table structure for table `masuk`
--
CREATE TABLE IF NOT EXISTS `masuk` (
  `no_masuk` varchar(15) NOT NULL,
  `tanggal` date NOT NULL,
  `supplier` varchar(100) NOT NULL,
  PRIMARY KEY (`no_masuk`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `masuk`
--
-- --------------------------------------------------------
--
-- Table structure for table `masuk_dt`
--
CREATE TABLE IF NOT EXISTS `masuk_dt` (
  `no_masuk` varchar(15) NOT NULL,
  `kode_barang` varchar(15) NOT NULL,
  `qty` double NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `masuk_dt`
```

```
--

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

--

-- Table structure for table `pemesan`

--

CREATE TABLE IF NOT EXISTS `pemesan` (

  `kd_buyer` varchar(15) NOT NULL,

  `nama` varchar(100) NOT NULL,

  `no_tel` varchar(13) NOT NULL,

  `alamat` varchar(200) NOT NULL,

  PRIMARY KEY (`kd_buyer`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `pemesan`

--

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

--

-- Table structure for table `pesan`

--

CREATE TABLE IF NOT EXISTS `pesan` (

  `no_pesan` varchar(15) NOT NULL,

  `kd_buyer` varchar(10) NOT NULL,

  `tanggal` date NOT NULL,
```

```
  `status` int(1) NOT NULL,
  `tgl_selesai` date NOT NULL,
  PRIMARY KEY (`no_pesan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `pesan`
--
-- --------------------------------------------------------
--
-- Table structure for table `pesan_dt`
--
CREATE TABLE IF NOT EXISTS `pesan_dt` (
  `no_pesan` varchar(15) NOT NULL,
  `kode_produk` varchar(15) NOT NULL,
  `qty` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `pesan_dt`
--
-- --------------------------------------------------------

--
-- Table structure for table `produk`
```

```
--
CREATE TABLE IF NOT EXISTS `produk` (
  `kode_produk` varchar(15) NOT NULL,
  `nama_produk` varchar(200) NOT NULL,
  `stok` int(11) NOT NULL,
  PRIMARY KEY (`kode_produk`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `produk`
--
INSERT INTO `produk` (`kode_produk`, `nama_produk`, `stok`)
VALUES
('PRD-0001', 'Infusan RL', 0),
('PRD-0002', 'Infusan Ring-As', 0),
('PRD-0003', 'Infusan D5', 0),
('PRD-0004', 'Infusan Futrolit', 0),
('PRD-0005', 'Infusan Tridex 27B', 0),
('PRD-0006', 'Infusan NS', 0),
('PRD-0007', 'Infusan D5 + 1/2 NS', 0);
-- --------------------------------------------------------
--
-- Table structure for table `produksi`
--
```

```
CREATE TABLE IF NOT EXISTS `produksi` (
  `kode_produksi` varchar(15) NOT NULL,
  `no_pesan` varchar(15) NOT NULL,
  `weighting` date NOT NULL,
  `mixing` date NOT NULL,
  `packing` date NOT NULL,
  `status` int(11) NOT NULL,
  PRIMARY KEY (`kode_produksi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `produksi`
--
-- --------------------------------------------------------
--
-- Table structure for table `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;
--
-- Dumping data for table `user`
--
/*!40101SET
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT
*/;
/*!40101SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULT
S */;
/*!40101SET
COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION
*/;
```

4.4.5. Implementasi Antar Muka

Tahap implementasi 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 data *inventory*. Fasilitas *login* ini diperuntukan bagi bagian gbb, produksi, marketing, goj.

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

Tabel 4.14 Implementasi Login

Sub Menu	Deskripsi	Nama File
Menu Login	File program yang form login	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 4.15 Halaman Utama

Sub Menu	Deskripsi	Nama File
Menu Utama	File program yang form login	FHome.java
Akses	Untuk menampilkan sub menu hak akses.	
Inventory	Untuk menampilkan sub menu	

	data persediaan.	
Produksi	Untuk menampilkan sub menu proses.	
Laporan	Menampilkan form laporan	FLaporan.java

3. Implementasi Halaman Berdasarkan Sub Menu Akses

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

Tabel 4.16 Implementasi Sub Menu Akses

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

4. Implementasi Halaman Berdasarkan Sub Menu *Inventory*

Implementasi sub-menu *inventory* dari aplikasi ini adalah sebagai berikut :

Tabel 4.17 Implementasi Sub Menu Inventory

Sub Menu	Deskripsi	Nama File
Bahan	Untuk melihat, menambah dan menghapus data barang	FBarang.java
Input Bahan	Untuk menambah barang baru yang datang	FInput.java
Formula	Untuk menambah formula	Fformula.java

	baru	
User	Untuk melihat, menambah dan menghapus data *user*	FUser.java
Produk	Untuk melihat, menambah dan menghapus data produk jadi	FProduk.java

5. Implementasi Halaman Berdasarkan Sub Menu Produksi

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

Tabel 4.18 Implementasi Sub Menu Produksi

Sub Menu	Deskripsi	Nama File
Pemesanan	Untuk input data pemesanan data penjualan	FPesan.java
Penjadwalan	Untuk input data penjadwalan produksi	FJadwal.java
Packing	Untuk input data barang dalam proses pengemasan	FKemas.java
Kirim	Untuk input data barang keluar	FKirim.java

6. Implementasi Halaman Berdasarkan Sub Menu Laporan

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

Tabel 4.19 Implementasi Sub Menu Laporan

Sub Menu	Deskripsi	Nama File
Menu Laporan	Untuk mencetak laporan penjadwalan per periode	FLaporan.java

4.4.6. 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 PT. Sanbe Farma Sterile Bandung tempat penulis melakukan penelitian.

4.4.6.1. Instalasi XAMPP

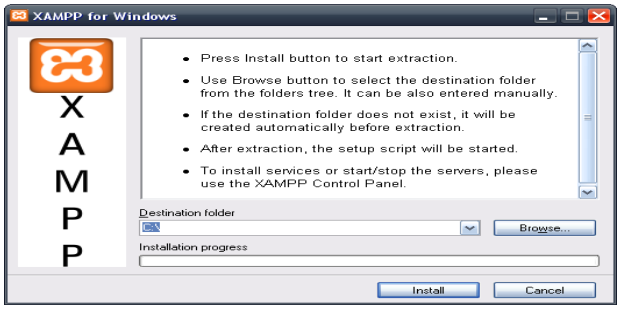
Berikut adalah langkah – langkah dalam instalasi perangkat lunak (*software*) XAMPP :

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

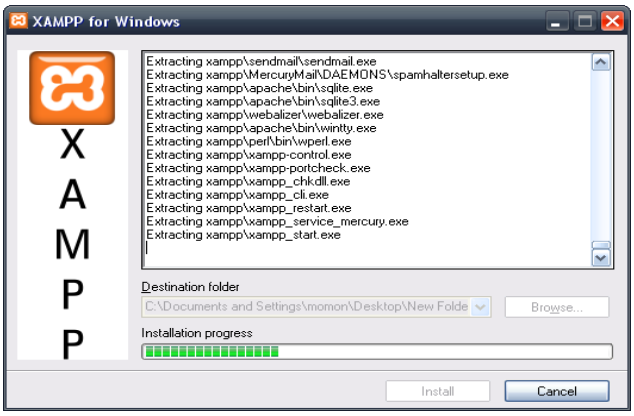


Gambar 4.25 Icon Instalasi xampp

2. Pilih *Install*

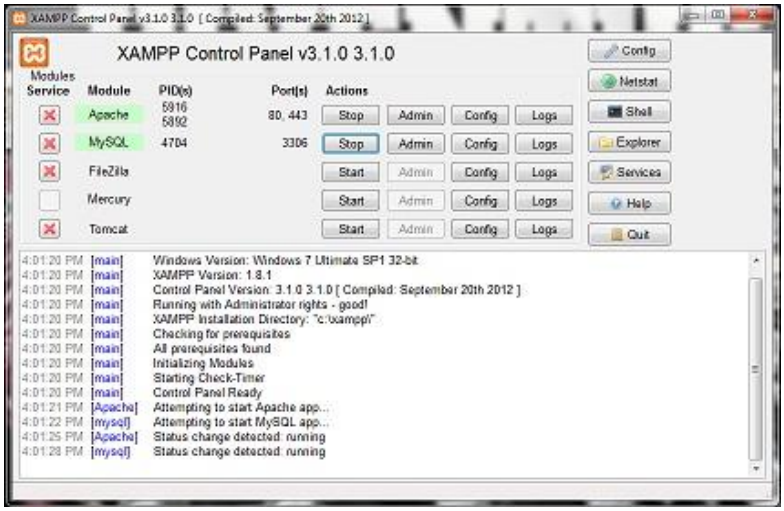


Gambar 4.26 Tampilan Awal Instalasi xampp



Gambar 4.27 Tampilan Proses Instalasi xampp

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



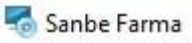
Gambar 4.28 Tampilan xampp control panel

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

4.4.6.2.Instalasi Program Aplikasi

Berikut adalah langkah – langkah dalam instalasi perangkat lunak (*software*) sistem informasi PT. Sanbe Farma Sterile Bandung :

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



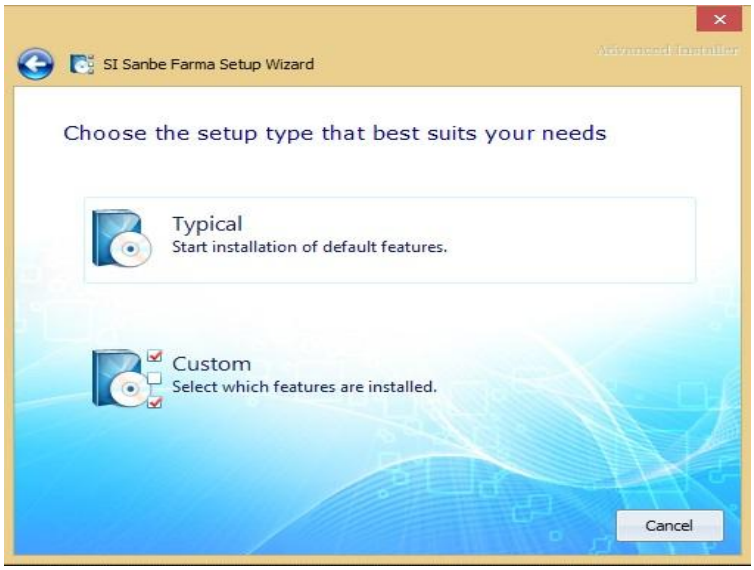
Gambar 4.29 .*Icon* Instalasi Program.

2. Muncul tampilan awal instalsi program

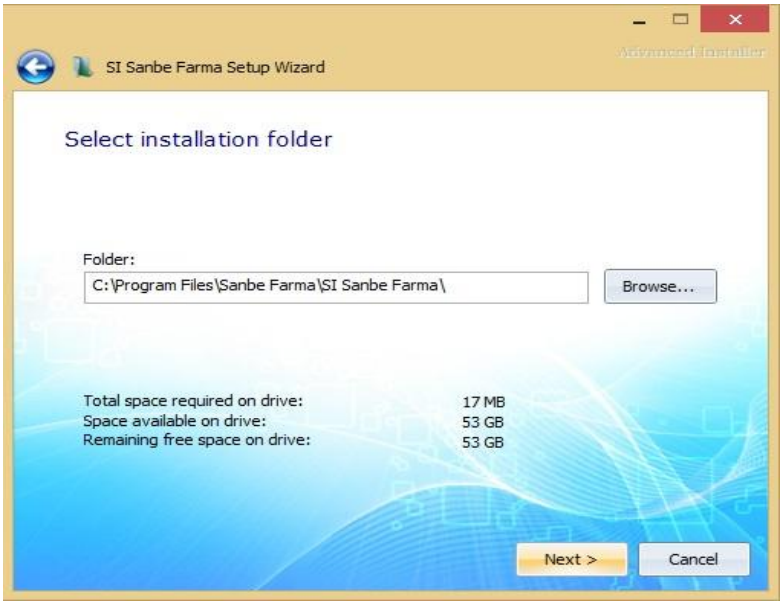


Gambar 4.30 Tampilan awal Instalasi Program

3. Klik *next*, kemudian akan muncul tampilan pemilihan folder penyimpanan, pilih direktori yang diinginkan, lalu klik *next*.



Gambar 4.31 Tampilan pemilihan direktori instalasi

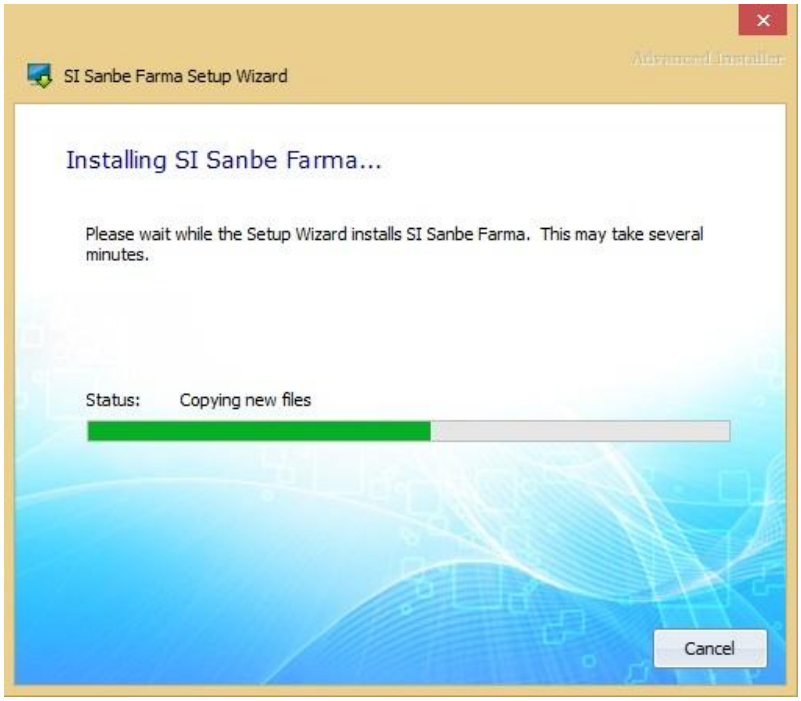


Gambar 4.32 Tampilan pemilihan folder penyimpanan

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

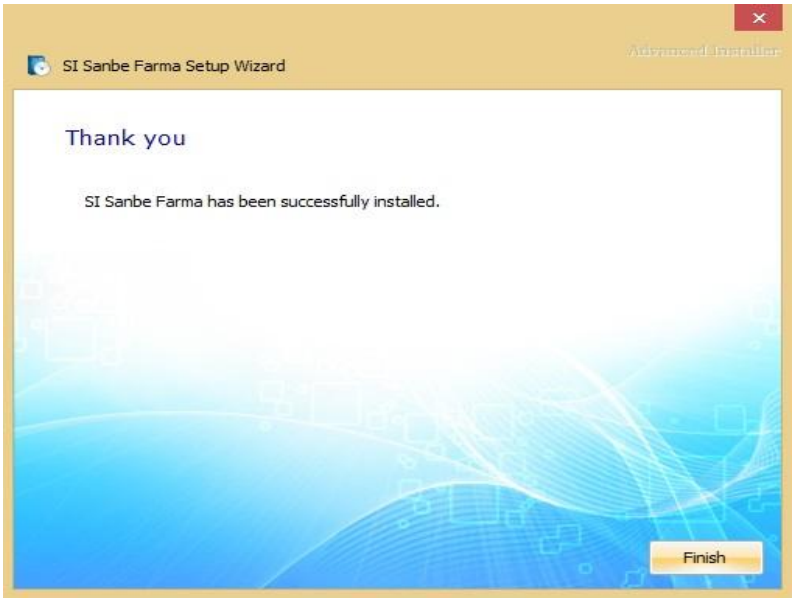


Gambar 4.33 Tampilan instalasi program



Gambar 4.34 Tampilan proses instalasi program

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



Gambar 4.35 Tampilan akhir instalasi program

4.4.7. 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 login benar, maka form utama akan bisa diakses.



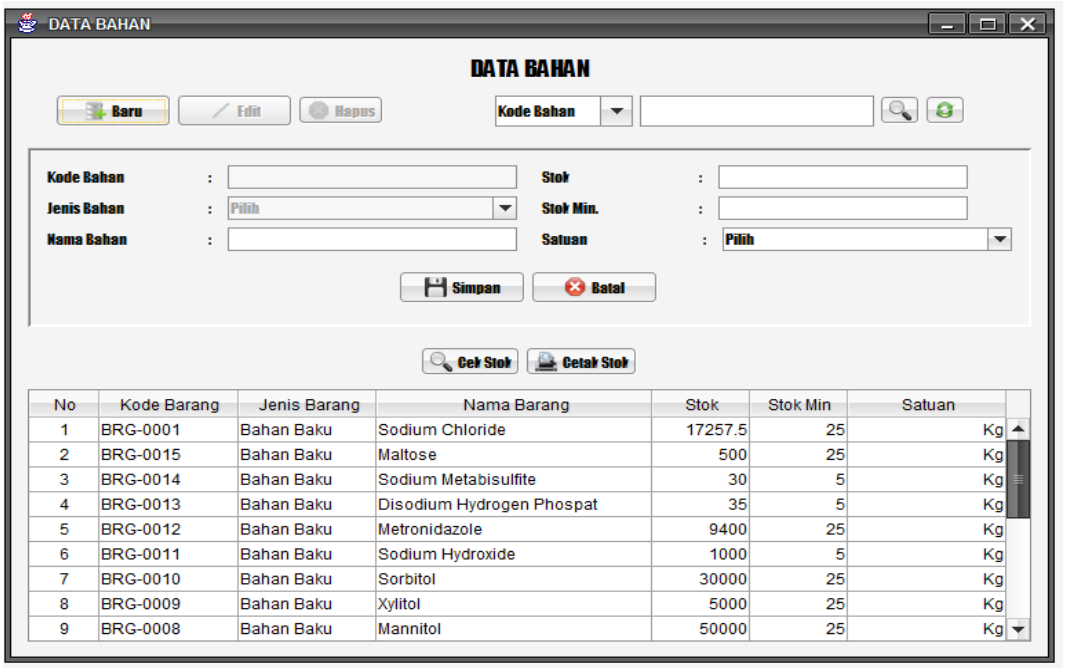
Gambar 4.36 Form Login

2. Tampilan Menu Utama



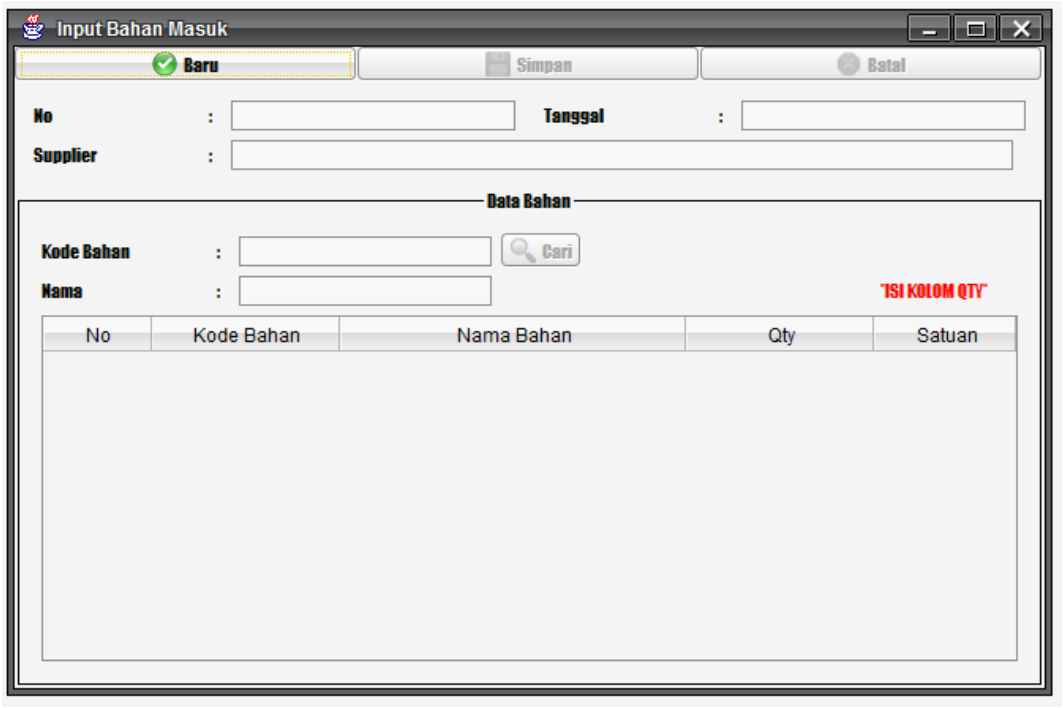
Gambar 4.37 Tampila Menu Utama

3. Tampilan Data Bahan



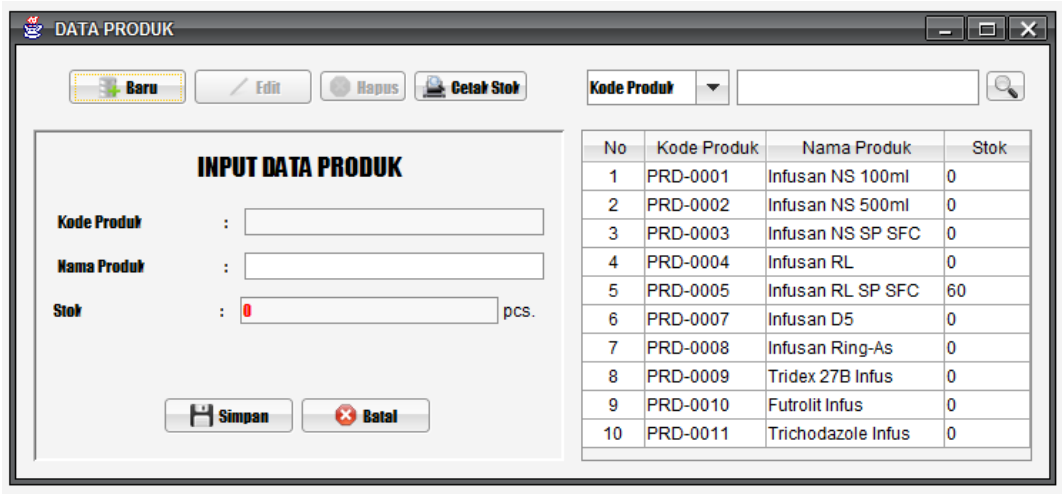
Gambar 4.38 Tampilan Data Bahan

4. Tampilan Input Bahan



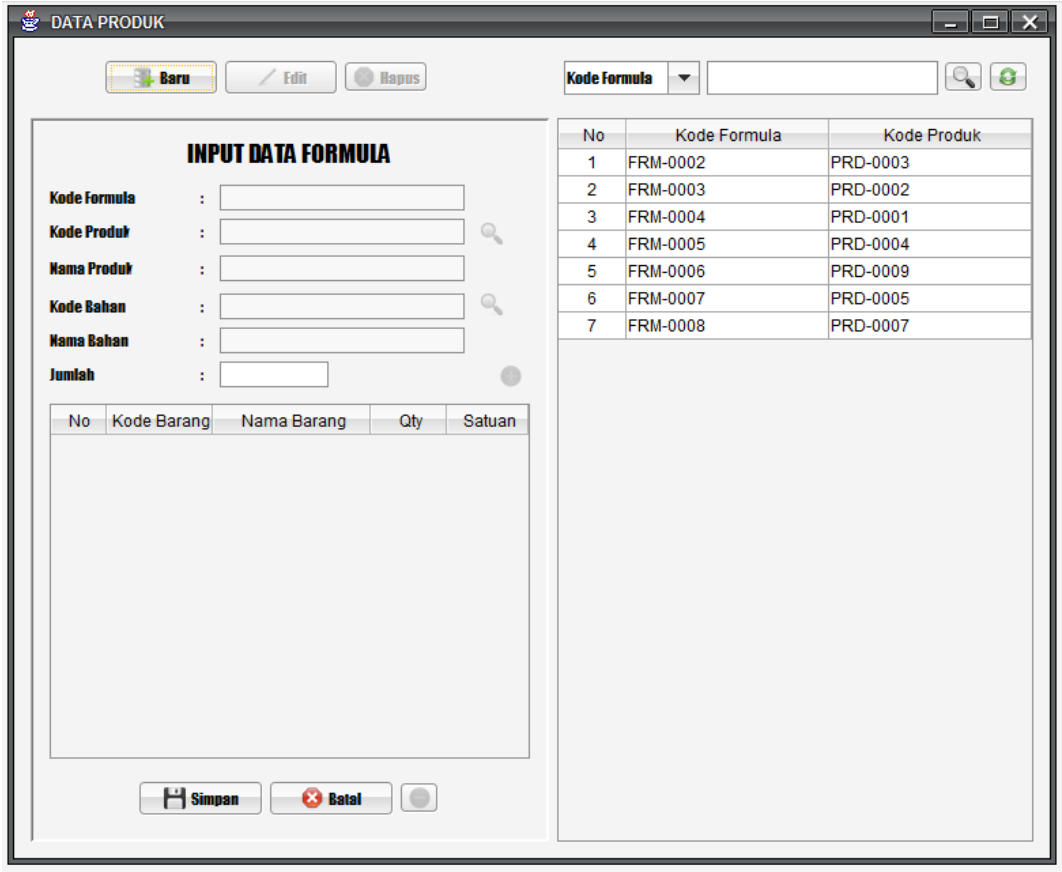
Gambar 4.39 Tampilan Input Bahan

5. Tampilan Form Produk



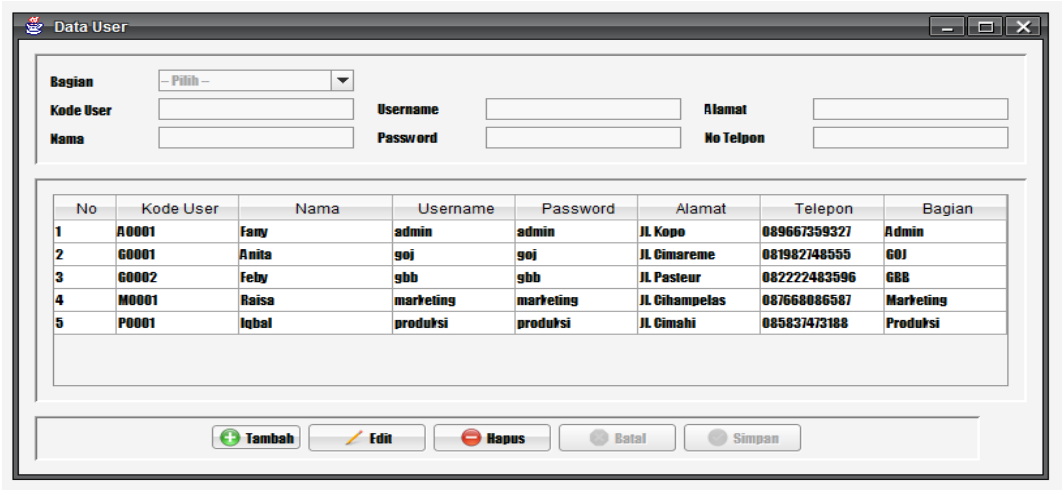
Gambar 4.40 Tampilan Form Produk

6. Tampilan Form Formula



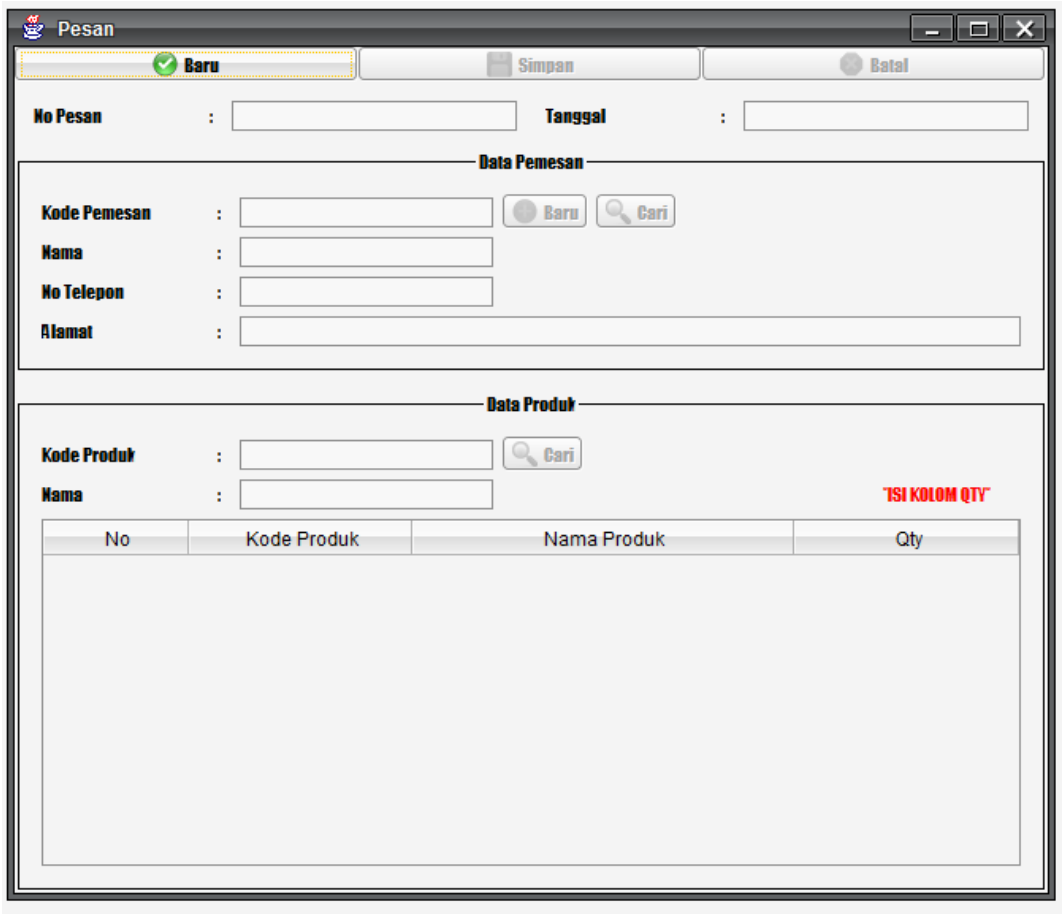
Gambar 4.41 Tampilan Form Formula

7. Tampilan Menu User



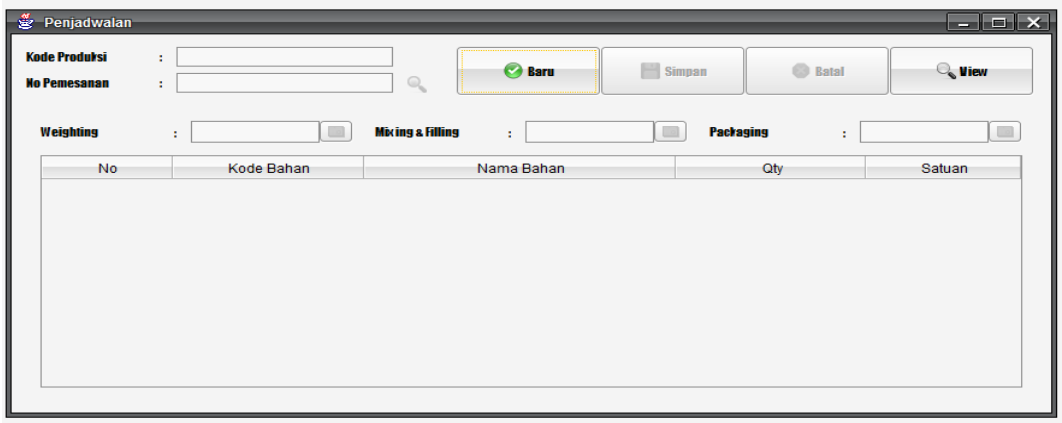
Gambar 4.42 Tampilan Menu User

8. Tampilan Form Pemesanan



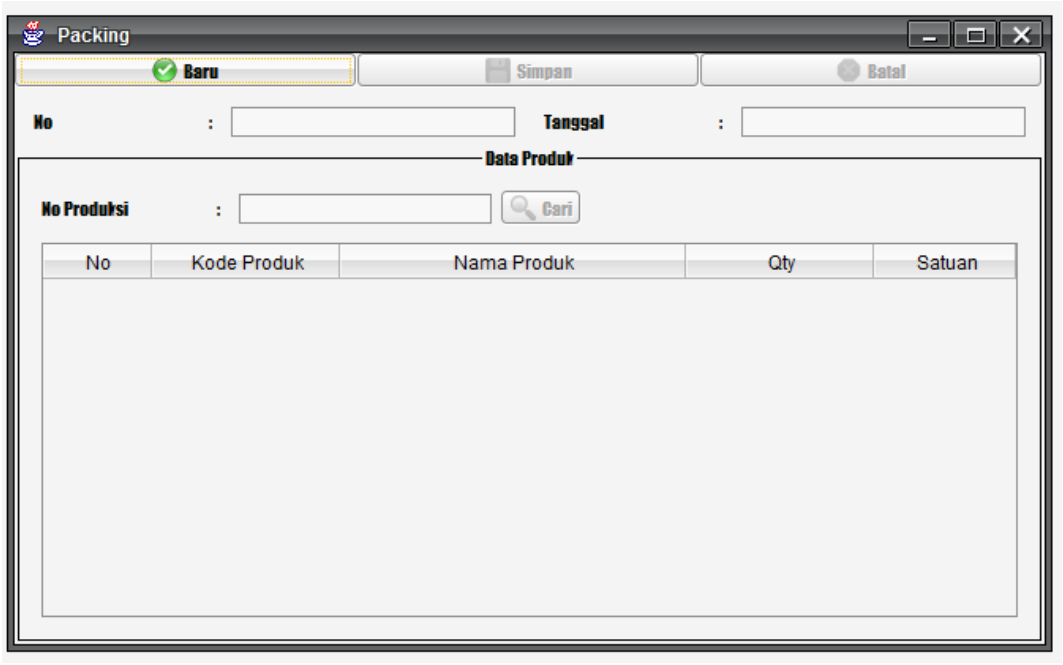
Gambar 4.43 Tampilan Form Pemesanan

9. Tampilan Form Penjadwalan



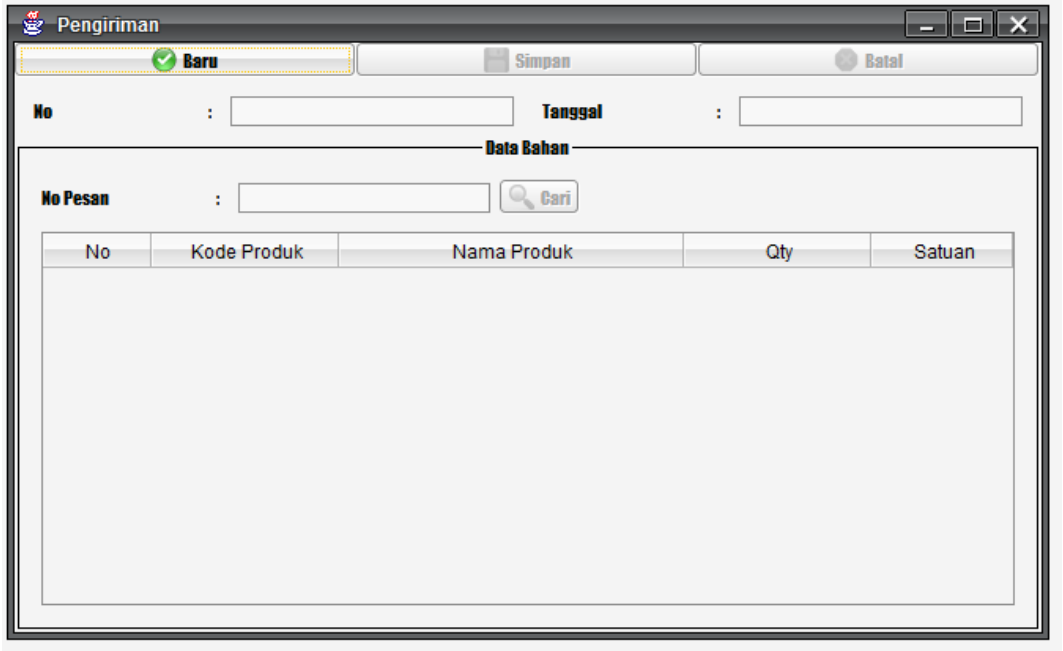
Gambar 4.44 Tampilan Form Penjadwalan

10. Tampilan Form Packing



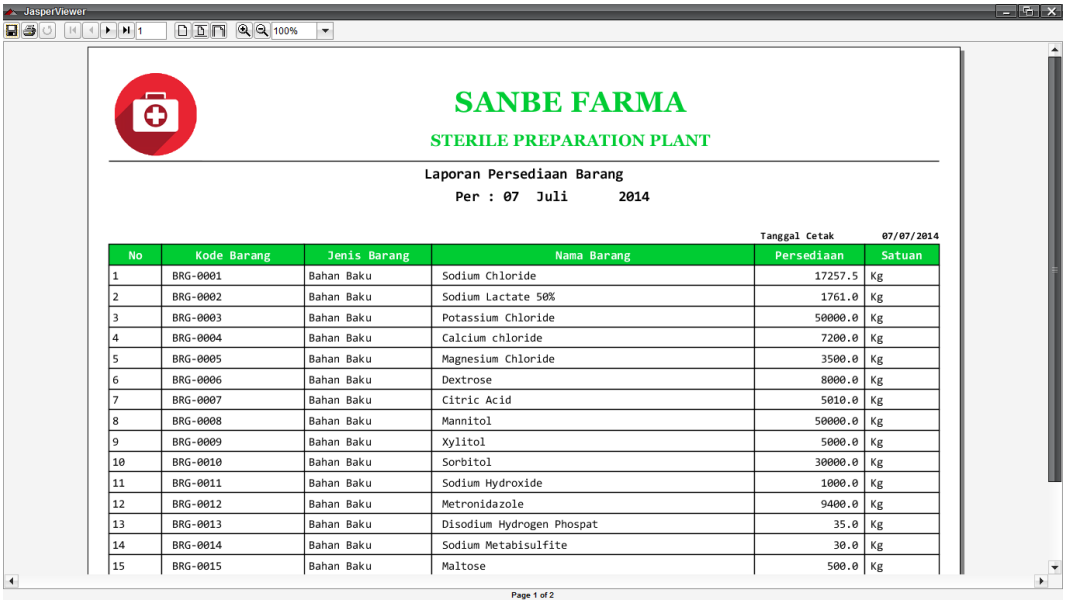
Gambar 4.45 Tampilan Form Packing

11. Tampilan Form Kirim



Gambar 4.46 Tampilan Form Kirim

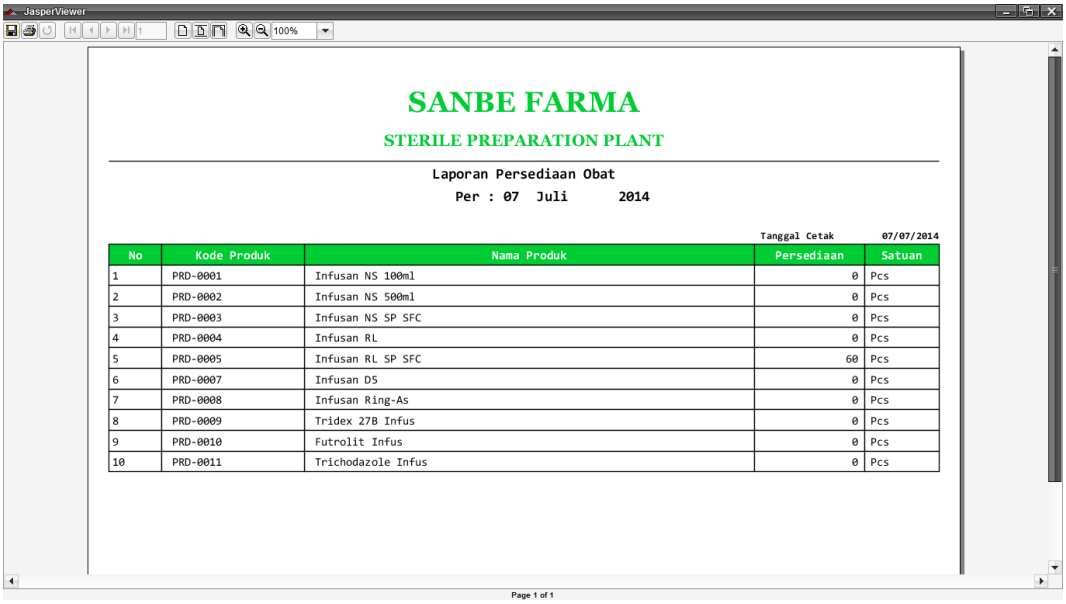
12. Tampilan Rekap Laporan Persediaan Barang



No	Kode Barang	Jenis Barang	Nama Barang	Persediaan	Satuan
1	BRG-0001	Bahan Baku	Sodium Chloride	17257.5	Kg
2	BRG-0002	Bahan Baku	Sodium Lactate 50%	1761.0	Kg
3	BRG-0003	Bahan Baku	Potassium Chloride	50000.0	Kg
4	BRG-0004	Bahan Baku	Calcium chloride	7200.0	Kg
5	BRG-0005	Bahan Baku	Magnesium Chloride	3500.0	Kg
6	BRG-0006	Bahan Baku	Dextrose	8000.0	Kg
7	BRG-0007	Bahan Baku	Citric Acid	5010.0	Kg
8	BRG-0008	Bahan Baku	Mannitol	50000.0	Kg
9	BRG-0009	Bahan Baku	Xylitol	5000.0	Kg
10	BRG-0010	Bahan Baku	Sorbitol	30000.0	Kg
11	BRG-0011	Bahan Baku	Sodium Hydroxide	1000.0	Kg
12	BRG-0012	Bahan Baku	Metronidazole	9400.0	Kg
13	BRG-0013	Bahan Baku	Disodium Hydrogen Phospat	35.0	Kg
14	BRG-0014	Bahan Baku	Sodium Metabisulfite	30.0	Kg
15	BRG-0015	Bahan Baku	Maltose	500.0	Kg

Gambar 4.47 Tampilan Rekap Laporan Peersediaan Barang

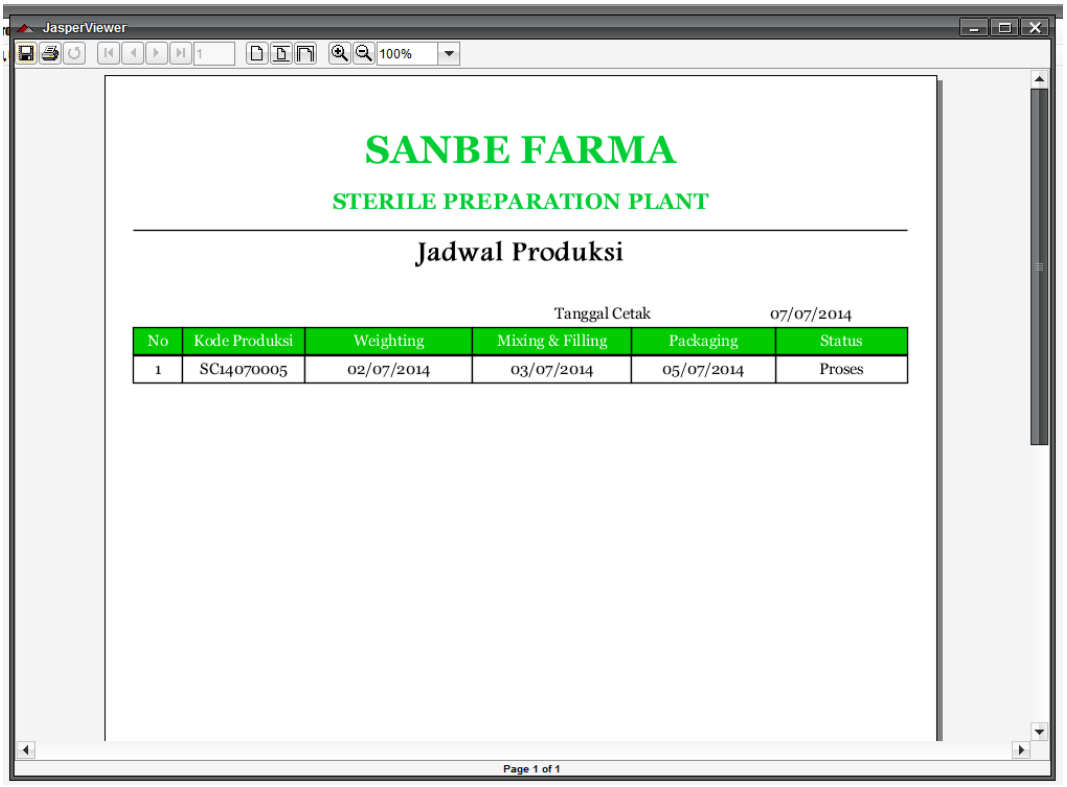
13. Tampilan Rekap Laporan Persediaan Obat



No	Kode Produk	Nama Produk	Persediaan	Satuan
1	PRD-0001	Infusan NS 100ml	0	Pcs
2	PRD-0002	Infusan NS 500ml	0	Pcs
3	PRD-0003	Infusan NS SP SFC	0	Pcs
4	PRD-0004	Infusan RL	0	Pcs
5	PRD-0005	Infusan RL SP SFC	60	Pcs
6	PRD-0007	Infusan D5	0	Pcs
7	PRD-0008	Infusan Ring-As	0	Pcs
8	PRD-0009	Tridex 27B Infus	0	Pcs
9	PRD-0010	Futrolit Infus	0	Pcs
10	PRD-0011	Trichodazole Infus	0	Pcs

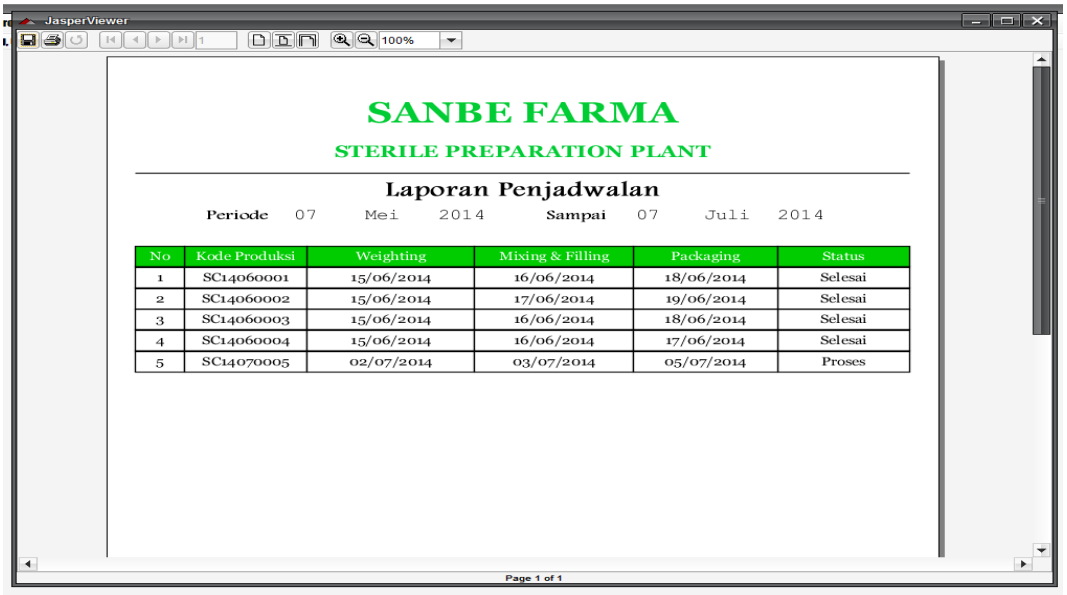
Gambar 4.48 Tampilan Rekap Laporan Persediaan Obat

14. Tampilan Rekap Jadwal Produksi yang masih dalam Proses



Gambar 4.49 Tampilan Rekap Jadwal Produksi yang masih dalam Proses

15. Tampilan Rekap Laporan Penjadwalan Per Periode



Gambar 4.50 Tampilan Rekap Laporan Produksi Per Periode

4.5 Pengujian

Adapun teknik pengujian perangkat lunak yang digunakan penulis dalam membangun sistem informasi di PT. Sanbe Farma Sterile Bandung, 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 4.20 Rencana Pengujian SI Manufaktur Sanbe Farma Sterile Bandung

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
Login	Pengecekan Login	Sistem	Blackbox
Pengisian Data	Pengisian Data Bahan	Modul	Blackbox
	Pengisian Data Input Bahan	Modul	Blackbox
	Pengisisan Data Formula	Modul	Blackbox
	Pengisisan Data Produk	Modul	Blackbox
	Pengisisan Data Penjadwalan	Modul	Blackbox
	Pengisisan Data Pemesanan	Modul	Blackbox
	Pengisisan Data Packing	Modul	Blackbox
	Pengisisan Data User	Modul	Blackbox
	Pengisisan Data Kirim	Modul	Blackbox

	Cetak Laporan Penjadwalan	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 adalah sebagai berikut :

1. Pengecekan Login Admin

   Berikut ini adalah kasus dan hasil pengujian login admin:

Tabel 4.21 Kasus dan Hasil Pengujian Pengecekan Login Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin Password : admin	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 :admin Password : admon	Tercantum pada textbox username dan textbox password.	User 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 Bahan

Berikut ini adalah kasus dan hasil pengujian pengisian data bahan:

Tabel 4.22 Kasus dan Hasil Pengujian Pengisian Data Pemesanan.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih kode bahan pada *combo box*	Tombol simpan dan batal aktif akan otomatis muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data bahan diisi secara lengkap	Penyimpanan data ke dalam file data bahan	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 bahan 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 Input Bahan

Berikut ini adalah kasus dan hasil pengujian pengisian input bahan:

Tabel 4.23 Kasus dan Hasil Pengujian Pengisian Data Input Bahan.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol "Baru"	Tabel barang akan otomatis muncul	Dapat memilih barang yang akan diambil	[X] Diterima [ ] Ditolak
Data-data bahan 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 bahan tidak diisi secara lengkap	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

4. Pengisian Data Formula

Berikut ini adalah kasus dan hasil pengujian pengisian data formula:

Tabel 4.24 Kasus dan Hasil Pengujian Pengisian Data Formula.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan Tombol 'Baru'	*Field-field* data barang akan muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data formula 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	Muncul pesan "Input berhasil", dan data dapat disimpan.	[X] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data-data formula 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

5. Pengisian Data Produk

Berikut ini adalah kasus dan hasil pengujian pengisian data produk:

Tabel 4.25 Kasus dan Hasil Pengujian Pengisian Data Produk.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan Tombol 'Baru'	*Field-field* data barang akan muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data produk 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	Muncul pesan "Input berhasil", dan data dapat disimpan.	[X]Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data-data produk 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

6. Pengisian Data Penjdwalan

Berikut ini adalah kasus dan hasil pengujian pengisian data penjadwalan:

Tabel 4.26 Kasus dan Hasil Pengujian Pengisian Data Penjadwalan.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan tombol "Tambah"	Tombol simpan dan batal aktif, serta kode supplier akan otomatis muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data penjadwalan diisi secara lengkap	Penyimpanan data ke dalam file data penjadwalan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol simpan	Data tersimpan ke dalam database	Muncul pesan "Input berhasil", dan data dapat	[X] Diterima [ ] Ditolak

		disimpan.	
Kasus dan Hasil Uji (Data Salah)			
Data-data penjadwalan 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

7. Pengisian Data Pemesanan

Berikut ini adalah kasus dan hasil pengujian pengisian data pemesanan:

Tabel 4.27 Kasus dan Hasil Pengujian Pengisian Data Pemesanan.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan tombol "Baru"	Tombol simpan,batal, serta no fakturaktif	Dapat melakukan pemilihan no faktur sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Pilih no pemesan yang akan di retur	Data barang yang telah dibeli akan otomatis muncul	Dapat melakukan pengisian data dengan benar	[X] Diterima [ ] Ditolak
Jumlah barang pemesan	Penyimpanan data ke dalam file data pemesanan	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak

diisi dengan benar			
Klik tombol simpan	Data tersimpan ke dalam database	Muncul pesan "Input berhasil", dan data dapat disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Jumlah pemesanan tidak diisi secara lengkap / benar	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. Pengisian Data Packing

Berikut ini adalah kasus dan hasil pengujian pengisian data *packing*:

Tabel 4.28 Kasus dan Hasil Pengujian Pengisian Data Packing.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih No Packing yang akan divalidasi pada *combobox*	Tombol simpan dan batal aktif, serta data barang yang akan dipesan otomatis muncul	Dapat melakukan pengisian data sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Pilih data barang dan	Penyimpanan data ke dalam file data	Dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak

isi jumlah barang yang akan dipesan	pemesanan	dengan benar	
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)			
Kolom jumlah pada barang yang dipilih tidak diisi	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

9. Pengisian Data Kirim

Berikut ini adalah kasus dan hasil pengujian pengisian datapenerimaan:

Tabel 4.29 Kasus dan Hasil Pengujian Pengisian Data Kirim.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan tombol "baru"	Tombol simpan,batal aktif, dan no MO aktif	Dapat melakukan pengisian data dan memilih no MO sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data dan no MO diisi	Data barang yang telah dipesan	Dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak

secara lengkap dan benar	sebelumnya otomatis muncul	dengan benar	
Klik tombol simpan	Data tersimpan ke dalam database data pengiriman	Muncul pesan “Input berhasil”, dan data dapat disimpan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data-data tidak diisi secara lengkap dan benar	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

10. Pengisian Data User

Berikut ini adalah kasus dan hasil pengujian pengisian datauser:

Tabel 4.30. Kasus dan Hasil Pengujian Pengisian Data Penerimaan User.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tekan tombol “baru”	Tombol simpan,batal aktif, danKode User otomatis muncul	Dapat melakukan pengisian datasesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data-data diisi secara lengkap dan benar	Penyimpanan data ke dalam file data user	Dapat melakukan penyimpanan data dengan benar	[X] Diterima [ ] Ditolak
Klik tombol	Data tersimpan ke	Muncul pesan	[X] Diterima

simpan	dalam database data user	“Input berhasil”, dan data dapat disimpan.	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data-data tidak diisi secara lengkap dan benar	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

11. Cetak Laporan Penjadwalan

Berikut ini adalah kasus dan hasil pengujian cetak laporan penjadwalan:

Tabel 4.31 Kasus dan Hasil Pengujian Cetak Laporan Penjadwalan

Kasus Dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan periode laporan yang akan dicetak	Pencarian data ke dalam database penjadwalan	Dapat melakukan pencarian data dan hasilnya dapat ditampilkan dandicetak	[X] Diterima [ ] Ditolak
Klik tombol cetak	Data pengeluaran bisa ditemukan	Muncul tampilan laporan	[X] Diterima [ ] Ditolak
Kasus Dan Hasil Uji (Data Salah)			
Periodelaporan tidak diisi / salah	Tidak dapat melakukan	Pencarian data gagal dilakukan	Tidak dapat melakukan

	pencarian		pencarian
Klik tombol cetak	Data bahan bisa ditemukan	Muncul pesan “Silahkan input periode”	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.