Bab 6
Implementasi

Pada tahap implementasi ini merupakan penerapan aplikasi dari hasil perancangan sistem yang ada untuk mencapai suatu tujuan yang diinginkan. Implementasi melaksanakan perintah-perintah yang secara terstruktur dari awal sampai akhir.

Pada implementasi ini juga akan mengimplementasikan sebuah perangkat lunak menggunakan program Java development kit (JDK), editor pemograman eclipse, dan My SQL sebagai media penyimpan data (*database*).

6.1 Implementasi Identitas

Pada perancangan untuk identitas data telah dipilih menggunakan *existence-based identity* karena memiliki beberapa keunggulan. tombol utama yang dihasilkan adalah atribut tunggal dan keseragaman urutan. Mengidentifikasi objek juga dapat mengkodekan informasi untuk mempromosikan pencarian yang efisien, terutama model yang besar dan kompleks. Sebagai berikut adalah proses implementasi dari identitas:

Sql Query pengurutan otomatis :

SELECT * FROM `barang` ORDER BY 'id_barang' ASC

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	id_barang	varchar(15)	latin1_general_ci	UNSIGNED	No		auto_increment	
☐	nama_barang	varchar(25)	latin1_general_ci		No			
☐	jenis_barang	varchar(15)	latin1_general_ci		No			
☐	harga_barang	int(11)			No			
☐	id_supplier	varchar(15)	latin1_general_ci		No			

Gambar 6.1 Pengurutan Data Otomatis

6.2 Implementasi Domain

Domain adalah seperangkat nilai-nilai, yang dapat diwakili dalam sebuah *field*, bahasa tipe data kelas domain. Domain untuk "byte" jenis data dapat dinyatakan sebagai [1.255], yang berarti bahwa byte dapat menyimpan nilai antara dan termasuk 1 dan 255. Karena tipe data RDBMS seringkali kurang kaya daripada jenis bahasa, metode untuk representasi domain yang diperlukan.

Table 6.1 Penetapan Domain

Atribut	Domain	Nilai
Id	Varchar	15
Nama	Varchar	25
Alamat	Varchar	50
Tanggal	Date	-
Uraian	Varchar	50
Rupiah	Int	11
No telepon	Varchar	15
Paswoord	Varchar	20
Jabatan	Varchar	15

6.3 Implementasi Kelas dan Atribut

Database MySQL merupakan database open source yang begitu hemat resource memory. Membuat database MySQL tidak perlu menggunakan cara manual lagi dengan mengetikkan perintah-perintah SQL, karena sekarang sudah banyak tool-tool visual yang membantu kita untuk membuat database dengan mudah dengan cepat, salah satunya adalah tool phpMyAdmin, tool ini merupakan salah satu paket dari XAMPP. Sebagai berikut Tabel-tabel *database* penyimpanan dari data sistem informasi pengadaan barang.

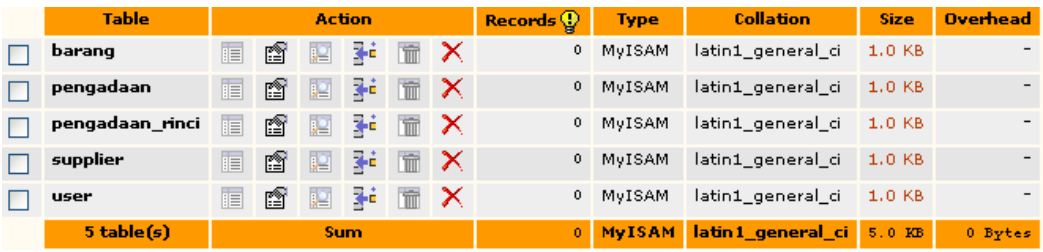


	Table	Action	Records	Type	Collation	Size	Overhead
☐	barang		0	MyISAM	latin1_general_ci	1.0 KB	-
☐	pengadaan		0	MyISAM	latin1_general_ci	1.0 KB	-
☐	pengadaan_rinci		0	MyISAM	latin1_general_ci	1.0 KB	-
☐	supplier		0	MyISAM	latin1_general_ci	1.0 KB	-
☐	user		0	MyISAM	latin1_general_ci	1.0 KB	-
	5 table(s)	Sum	0	MyISAM	latin1_general_ci	5.0 KB	0 Bytes

Gambar 6.2 Implementasi Tabel Kelas

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	user	varchar(15)	latin1_general_ci		No			
☐	nama	varchar(25)	latin1_general_ci		No			
☐	alamat	varchar(50)	latin1_general_ci		No			
☐	no_telepon	varchar(15)	latin1_general_ci		No			
☐	jabatan	varchar(15)	latin1_general_ci		No			
☐	password	varchar(20)	latin1_general_ci		No			

Gambar 6.3 Implementasi Atribut Kelas User

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	id_barang	varchar(15)	latin1_general_ci	UNSIGNED	No		auto_increment	
☐	nama_barang	varchar(25)	latin1_general_ci		No			
☐	jenis_barang	varchar(15)	latin1_general_ci		No			
☐	harga_barang	int(11)			No			
☐	id_supplier	varchar(15)	latin1_general_ci		No			
☐	stok_barang	varchar(15)	latin1_general_ci		No			

Gambar 6.4 Implementasi Atribut Kelas Barang

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	id_supplier	varchar(15)	latin1_general_ci	UNSIGNED	No		auto_increment	
☐	nama_supplier	varchar(25)	latin1_general_ci		No			
☐	no_telepon	varchar(15)	latin1_general_ci		No			
☐	alamat_supplier	varchar(50)	latin1_general_ci		No			

Gambar 6.5 Implementasi Atribut Kelas Supplier

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	no_faktur	varchar(15)	latin1_general_ci	UNSIGNED	No		auto_increment	
☐	uraian	varchar(50)	latin1_general_ci		No			
☐	tgl_pengadaan	date			No			
☐	user	varchar(15)	latin1_general_ci		No			
☐	id_supplier	varchar(15)	latin1_general_ci		No			
☐	tot_bayar	int(11)			No			

Gambar 6.6 Implementasi Atribut Kelas Pengadaan

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
☐	id_pengadaan_rinci	varchar(15)	latin1_general_ci	UNSIGNED	No		auto_increment	
☐	no_faktur	varchar(15)	latin1_general_ci		No			
☐	id_barang	varchar(15)	latin1_general_ci		No			
☐	jumlah	int(11)			No			
☐	harga_barang	int(11)			No			

Gambar 6.7 Implementasi Atribut Kelas Pengadaan Rinci

QUERY SQL PEMBUATAN TABEL:

```
-- Table structure for table `user`
CREATE TABLE `user` (
  `user` varchar(15) collate latin1_general_ci NOT NULL,
  `nama` varchar(25) collate latin1_general_ci NOT NULL,
  `alamat` varchar(50) collate latin1_general_ci NOT NULL,
  `no_telepon` varchar(15) collate latin1_general_ci NOT NULL,
  `jabatan` varchar(15) collate latin1_general_ci NOT NULL,
  `password` varchar(20) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`user`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci;

-- Table structure for table `barang`
CREATE TABLE `barang` (
  `id_barang` varchar(15) unsigned NOT NULL auto_increment,
  `nama_barang` varchar(25) collate latin1_general_ci NOT NULL,
  `jenis_barang` varchar(15) collate latin1_general_ci NOT NULL,
  `harga_barang` int(11) NOT NULL,
  `id_supplier` varchar(15) collate latin1_general_ci NOT NULL,
  `stok_barang` varchar(15) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_barang`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=1

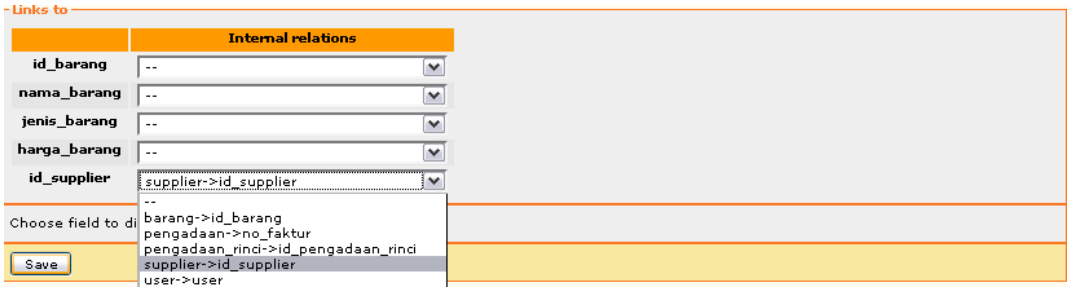
-- Table structure for table `supplier`
CREATE TABLE `supplier` (
  `id_supplier` varchar(15) unsigned NOT NULL auto_increment,
  `nama_supplier` varchar(25) collate latin1_general_ci NOT NULL,
  `no_telepon` varchar(15) collate latin1_general_ci NOT NULL,
  `alamat_supplier` varchar(50) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_supplier`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=1


-- Table structure for table `pengadaan`
CREATE TABLE `pengadaan` (
  `no_faktur` varchar( (15) unsigned NOT NULL auto_increment,
  `uraian` varchar(50) collate latin1_general_ci NOT NULL,
  `tgl_pengadaan` date NOT NULL,
  `user` varchar(15) collate latin1_general_ci NOT NULL,
  `id_supplier` varchar(15) collate latin1_general_ci NOT NULL,
  `tot_bayar` int(11) NOT NULL,
  PRIMARY KEY  (`no_faktur`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=1

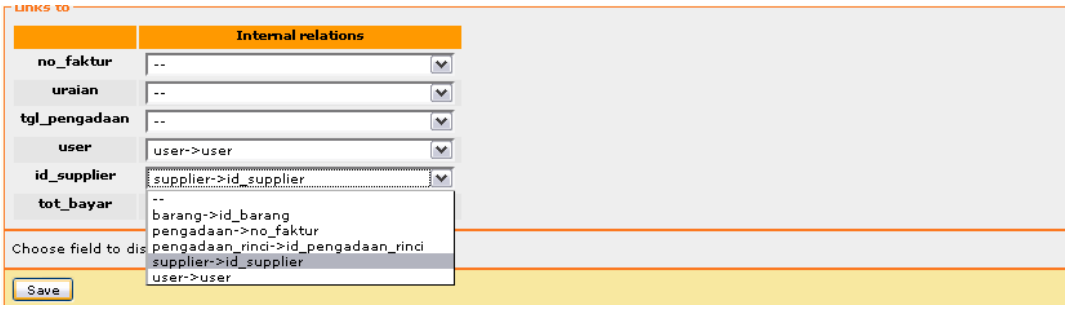
-- Table structure for table `pengadaan_rinci`
CREATE TABLE `pengadaan_rinci` (
  `id_pengadaan_rinci` varchar(15) unsigned NOT NULL auto_increment,
  `no_faktur` varchar(15) collate latin1_general_ci NOT NULL,
  `id_barang` varchar(15) collate latin1_general_ci NOT NULL,
  `jumlah` int(11) NOT NULL,
  `harga_barang` int(11) NOT NULL,
  PRIMARY KEY  (`id_pengadaan_rinci`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=1
```

6.4 Implementasi Asosiasi

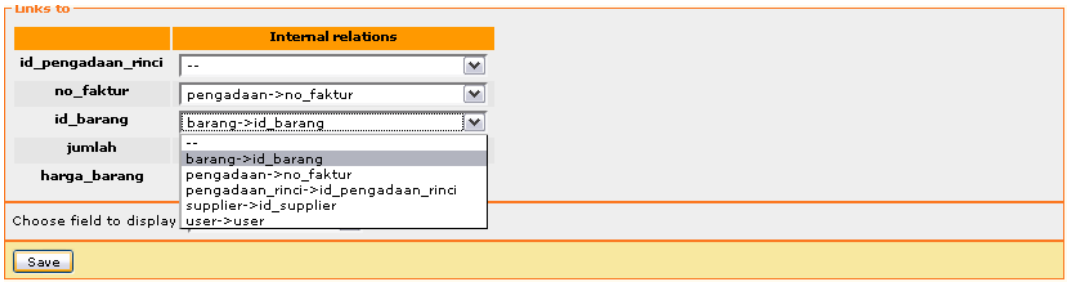
Sebuah asosiasi satu arah diimplementasikan sebagai pointer - atribut yang berisi referensi obyek. Jika multiplisitas adalah satu, maka diimplementasikan sebagai pointer sederhana. Jika keseragaman tersebut banyak, maka itu adalah satu set pointer. Jika ujung banyak diperintahkan, kemudian daftar dapat digunakan bukan menetapkan. Sebuah asosiasi dua arah diimplementasikan sebagai atribut di kedua arah, dengan menggunakan teknik untuk asosiasi satu arah. Sebagai berikut implementasi cara pembuatan asosiasi pada *database* My SQL:



Gambar 6.8 Implementasi Asosiasi Pada Kelas barang



Gambar 6.9 Implementasi Asosiasi Pada Kelas Pengadaan



Gambar 6.10 Implementasi Asosiasi Pada Kelas Pengadaan Rinci

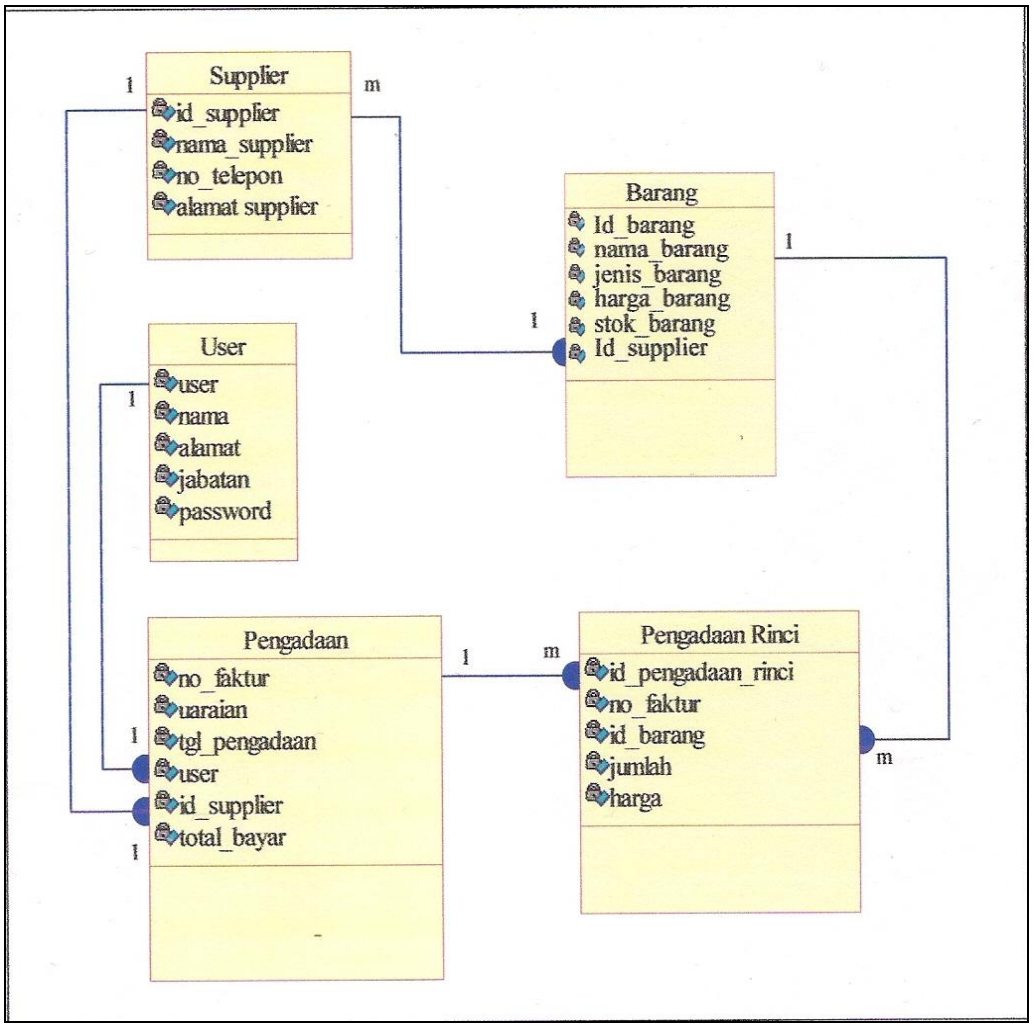
QUERY SQL ASOSIASI:

RELATIONS FOR TABLE barang:
-- id_supplier
-- supplier -> id_supplier

-- RELATIONS FOR TABLE pengadaan:
-- id_supplier
-- supplier -> id_supplier
-- user
-- user -> user

-- RELATIONS FOR TABLE pengadaan_rinci:
-- id_barang
-- barang -> id_barang
-- no_faktur
-- pengadaan -> no_faktur

6.5 Implementasi Asosiasi Maju



Gambar 6.11 Implementasi Asosiasi Maju

6.6 Implementasi Interface

Interface merupakan salah satu media yang digunakan sistem informasi untuk berkomunikasi dengan manusia. Interface diterapkan dalam bentuk aplikasi komputer.

Pembuatan aplikasi tidak secara asal, perlu melihat faktor ergonomi yang harus diperhatikan dalam pembuatannya. ergonomi adalah faktor yang akan menjadi acuan pembuatan aplikasi agar aplikasi dipakai dengan nyaman.

6.6.1 Form Aplikasi

Form adalah komponen utama dalam pengembangan aplikasi. Form Designer adalah tempat melekatnya komponen-komponen yang lain.

1. **Login**

   Merupakan menu awal interface aktor dengan sebuah sistem informasi dapat disebut juga sebagai kunci untuk membuka aplikasi. Ada dua aktor yang berperan sebagai user yaitu manajer dan pekerja keduanya memiliki masing-masing proporsi penggunaan aplikasi.

   Tinjauan desain login dari faktor ergonomi:

   a. Kesederhanan tampilan

      Form login dibentuk persegi dengan ukuran font 12 merupakan ukuran standar yang sering dipakai. Logo dan nama perusahan ditamplkan untuk membuat ciri khas aplikasi.

   b. Desain warna

      Form login didesain dengan dominan silver agar terlihat elegan dan simple, teks nama perusahan berwarna biru dan merah merupakan kombinasi yang tidak terlalu mencolok akan menyamankan dalam interface awal.



Gambar 6.12 Form Login

2. Menu Manajerial

Menu menejerial adalah menu yang menampilkan peranan yang dapat dilakukan oleh menejer seperti edit user, data barang, data supplier, dan data pengadaan. Semua peranan tersebut secara otomatis tidak dapat diperankan oleh user pekerja.

Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form menu menejerial dibuat persegi berisikan tombol-tombol peranan yang dapat dilakukan menejer ditampilkan samping kiri agar memudahkan dalam mengklik tombol. Logo perusahan ditampilkan ditengah agar pandangan tertuju pada logo tersebut menambah motifasi dan loyalitas terhadap perusahaan.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Tombol exit berwarna merah tampak mencolok dan menarik perhatian. Teks nama perusahan berwarna biru merupakan kombinasi yang tidak terlalu mencolok akan menyamankan dalam interface awal.



Gambar 6.13 Form Menu Menejerial

3. Form Edit User

Form edit user merupakan proses yang dilakukan menejer untuk menambah dan merubah user sebagai id user masing-masing aktor. Menejer merupakan admin dari sitem informasi pengadaan.

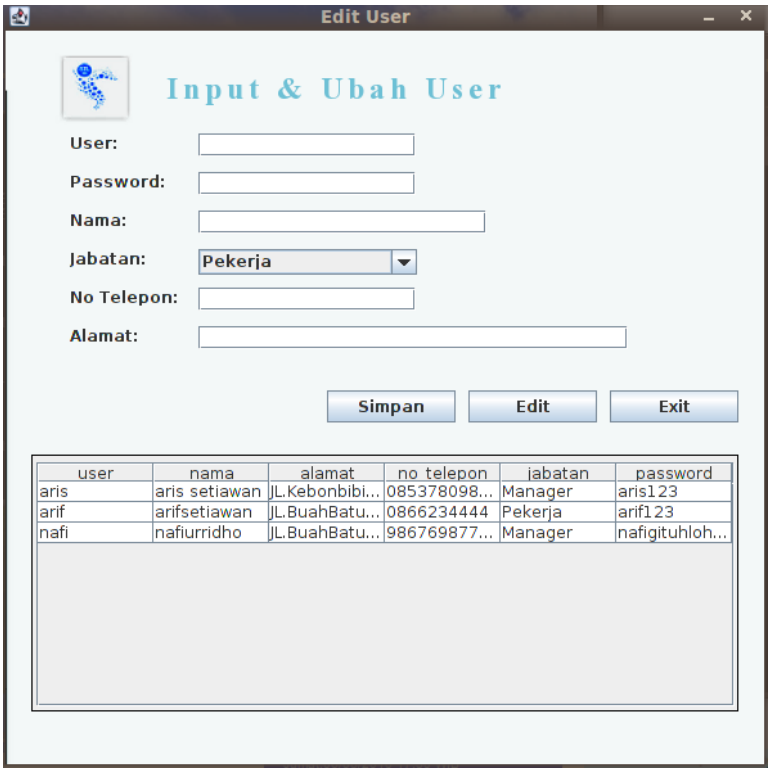
Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form edit user dibuat sederhana dalam penamilannya agar tidak terlalu membingungkan menejer dalam melakukan proses tersebut.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Teks input ukuran font 24 berwarna biru agar menejer dapat dengan jelas membaca form proses untuk menghindari kesalahan input yang sering terjadi pada (*human error).*



Gambar 6.14 Form Edit user

4. Menu Pekerja

Menu pekerja adalah menu yang menampilkan peranan yang dapat dilakukan oleh pekerja seperti input supplier, input barang, dan proses pengadaan. Semua peranan tersebut secara otomatis tidak dapat diperankan oleh user menejer.

Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form menu pekerja dibuat persegi berisikan tombol-tombol peranan yang dapat dilakukan pekerja ditampilkan samping kanan agar memudahkan dalam mengklik tombol. Logo perusahan ditampilkan ditengah agar pandangan tertuju pada logo tersebut menambah motifasi dan loyalitas terhadap perusahaan.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Tombol exit berwarna merah tampak mencolok dan menarik

perhatian. Teks nama perusahan berwarna biru merupakan kombinasi yang tidak terlalu mencolok akan menyamankan dalam interface awal.



Gambar 6.15 Form Menu Pekerja

5. Form Input Supplier

Form Input supplier merupakan proses yang dilakukan pekerja untuk menginput dan mangubah data supplier.

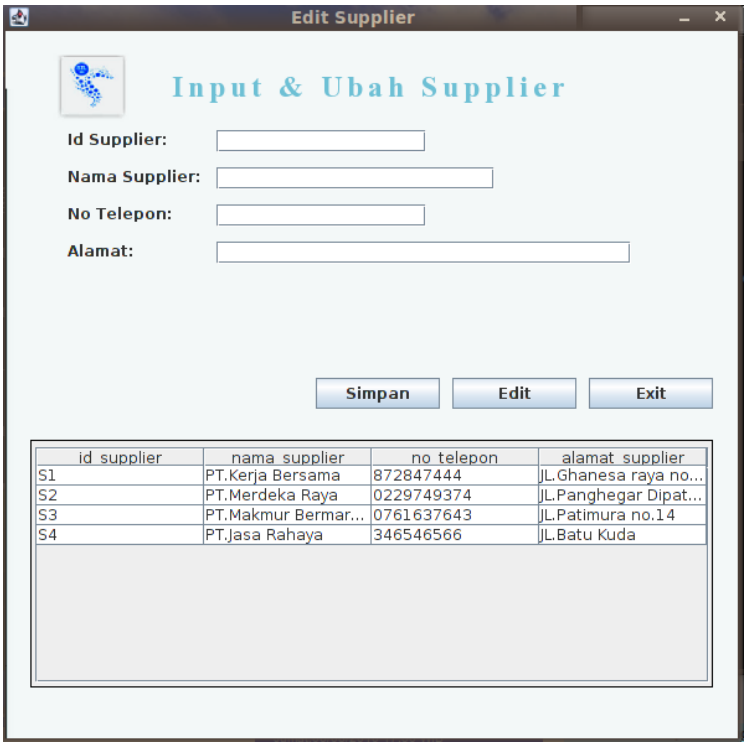
Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form input supplier dibuat sederhana dalam penamilannya agar tidak terlalu membebani pekerja dalam melakukan proses tersebut.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Teks input ukuran font 24 berwarna biru agar menejer dapat dengan jelas membaca form proses untuk menghindari kesalahan input yang sering terjadi pada (*human error).*



Gambar 6.16 Form Input Supplier

6. Form Input Barang

Form Input barang merupakan proses yang dilakukan pekerja untuk menginput dan mangubah data barang.

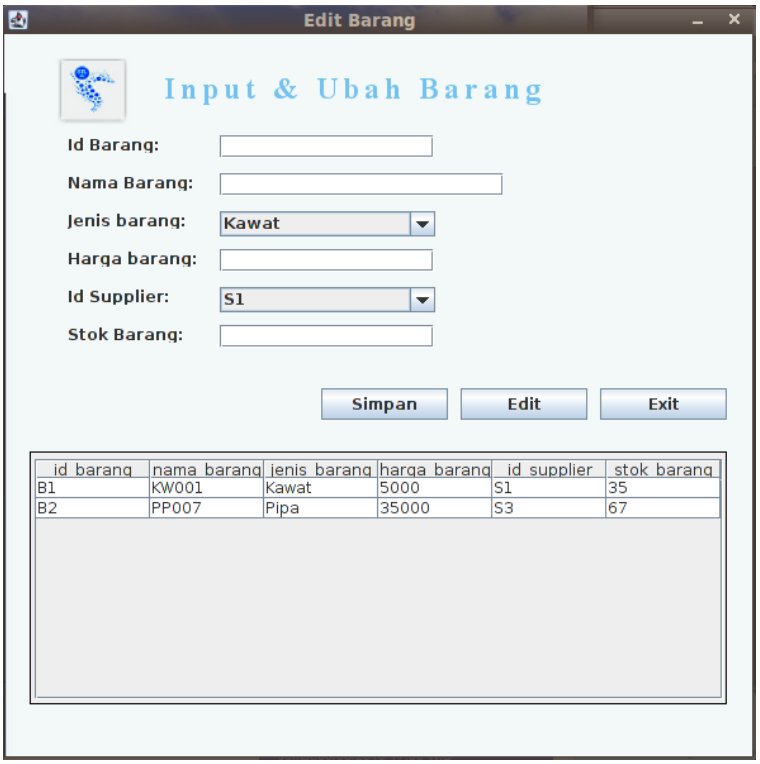
Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form input barang dibuat sederhana dalam penamilannya agar tidak terlalu membebani pekerja dalam melakukan proses tersebut.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Teks input ukuran font 24 berwarna biru agar menejer dapat dengan jelas membaca form proses untuk menghindari kesalahan input yang sering terjadi pada (*human error).*



Gambar 6.17 Form Input Supplier

7. Form Input Pengadaan

Form input pengadaan merupakan proses yang dilakukan pekerja untuk menginput dan mangubah data pengadaan. Dalam form pengadaan ada beberapa tombol yang memiliki proses yang berbeda seperti proses merupakan tombol untuk melakukan pengadaan apabila kuantitas barang lebih dari satu, buat Purchase Order (PO) merupakan proses cetak untuk melakukan pemesanan, buat Kartu Raw Material (KRM) merupakan proses cetak untuk pengecekan barang oleh bagian kualitas, dan buat bukti penerimaan barang (BPB) sebaga bukti apabila barang telah diterima.

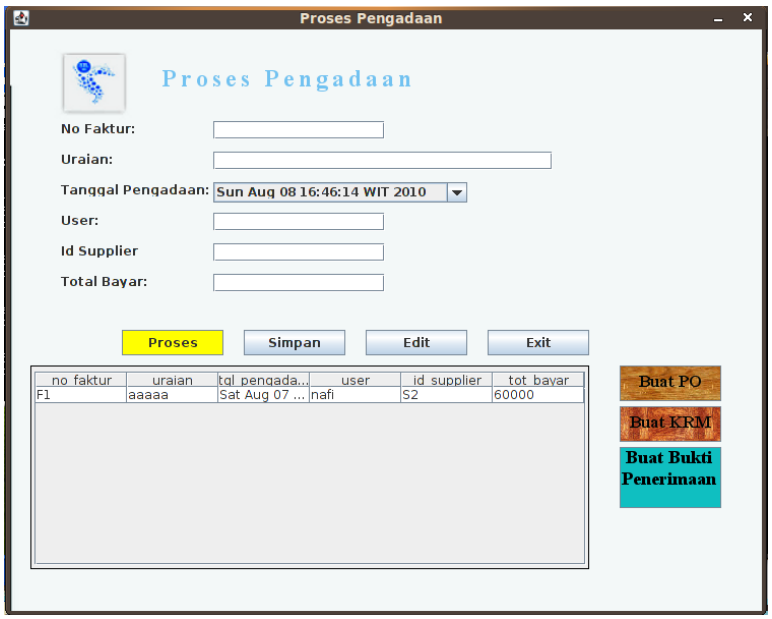
Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

   Form input pengadaan dibuat sederhana dalam penamilannya agar tidak terlalu membebani pekerja dalam melakukan proses tersebut.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Teks input ukuran font 24 berwarna biru agar menejer dapat dengan jelas membaca form proses untuk menghindari kesalahan input yang sering terjadi pada (*human error)*. Pada tombol-tombol ada beberapa memiliki warna berbeda yang memiliki fungsi berbeda, agar memiliki daya tarik mata untuk melihat dan tidak keliru dalam melakukan proses masing-masing.



Gambar 6.18 Form Input Pengadaan

8. Form Input Pengadaan Rinci

Form input pengadaan rinci merupakan proses yang dilakukan pekerja untuk menginput dan mangubah data pengadaan rinci. Form ini ditampilkan apabila pekerja melakukan prose pengadaan dengan kuantitas barang yang banya.

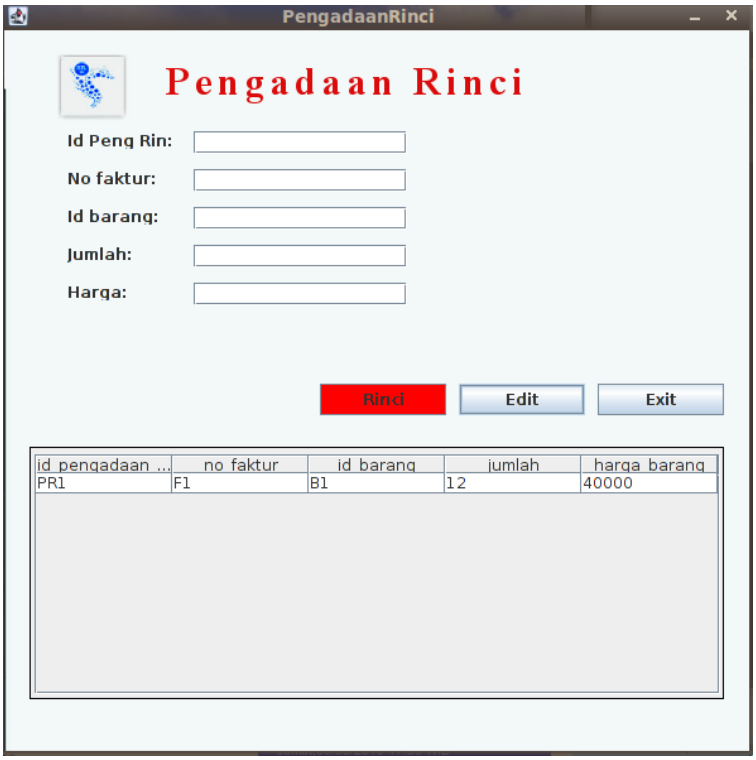
Tinjauan desain login dari faktor ergonomi:

a. Kesederhanan tampilan

Form input pengadaan rinci dibuat sederhana dalam penamilannya agar tidak terlalu membebani pekerja dalam melakukan proses tersebut.

b. Desain warna

Form login didesain dengan dominan silver agar terlihat elegan dan simple. Teks input ukuran font 24 berwarna merah berbeda dengan form lainnya agar mencirikan form pecahan dari form pengadaan.



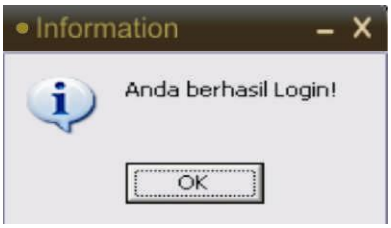
Gambar 6.19 Form Input Pengadaan

6.6.2 Event Aplikasi

Event aplikasi yaitu kejadian yang menimbulkan pesan pada aplikasi dikarenakan adanya pesan yang harus diketahui oleh user terhadap proses tertentu. Ada beberapa event pada aplikasi pengadaan barang seperti informasi, kesalahan, dan konfirmasi merupakan pesan penting sebelum melakukan proses selanjutnya.

1. Event Informasi Login Berhasil

Event informasi login berhasil terjadi karena input user dan password pada form login adalah benar. Event secara otomatis tampil dan aplikasi mengizinkan user melanjutkan ke proses selanjutnya. Gambar event sebagai berikut:



Gambar 6.20 Event Informasi Login Berhasil

2. Event Informasi Login Gagal

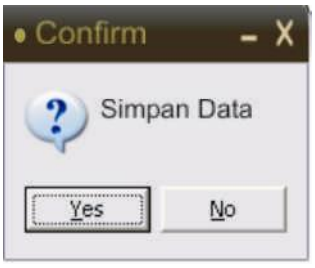
Event informasi login gagal terjadi karena user melakukan kesalahan ketika input user dan password. Event secara otomatis tampil dan aplikasi tidak mengizinkan user untuk melanjutkan ke proses selanjutnya, user dapat mengklik *button* ok untuk melakukan proses input ulang. Gambar event sebagai berikut:



Gambar 6.21 Event Informasi Login Gagal

3. Event Konformasi Simpan

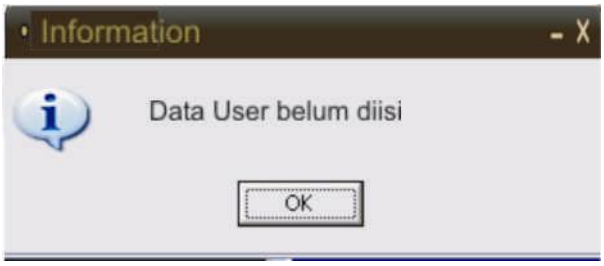
Event konfirmasi simpan terjadi ketika user akan melakukan proses simpan data, dihadapkan dua pertanyaan antara ya melakukan proses simpan dan tidak melakukan proses simpan. Gambar event sebagai berikut:



Gambar 6.22 Event Konfirmasi Simpan

4. Event Informasi Data User Belum Diisi

Event informasi data user belum diisi terjadi ketika melakukan proses simpan data user, user belum mengisi data yang akan disimpan. Klik *button* ok untuk melakukan indup data ulang . Gambar event sebagai berikut:



Gambar 6.23 Event Informasi Data User Belum Diisi

5. Event Informasi Data Supplier Belum Diisi

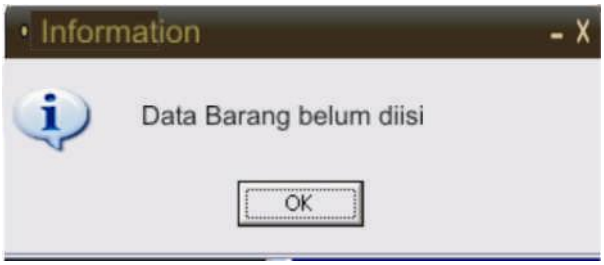
Event informasi data supplier belum diisi terjadi ketika melakukan proses simpan data supplier, user belum mengisi data yang akan disimpan. Klik *button* ok untuk melakukan indup data ulang . Gambar event sebagai berikut:



Gambar 6.24 Event Informasi Data Supplier Belum Diisi

6. Event Informasi Data Barang Belum Diisi

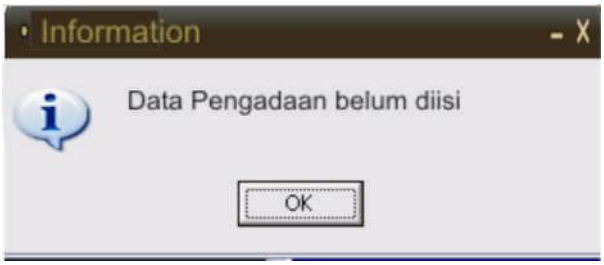
Event informasi data barang belum diisi terjadi ketika melakukan proses simpan data barang, user belum mengisi data yang akan disimpan. Klik *button* ok untuk melakukan indup data ulang . Gambar event sebagai berikut:



Gambar 6.25 Event Informasi Data barang Belum Diisi

7. Event Informasi Data Pengadaan Belum Diisi

Event informasi data pengadaan belum diisi terjadi ketika melakukan proses simpan data pengadaan, user belum mengisi data yang akan disimpan. Klik *button* ok untuk melakukan indup data ulang . Gambar event sebagai berikut:



Gambar 6.26 Event Informasi Data pengadaan Belum Diisi

8. Event Informasi Data Pengadaan Rinci Belum Diisi

Event informasi data pengadaan rinci belum diisi terjadi ketika melakukan proses simpan data pengadaan rinci, user belum mengisi data yang akan disimpan. Klik *button* ok untuk melakukan indup data ulang . Gambar event sebagai berikut:



Gambar 6.27 Event Informasi Data pengadaan Belum Diisi

9. Event Konformasi Edit Data

Event konfirmasi edit data terjadi ketika user akan melakukan proses edit data, dihadapkan dua pertanyaan antara ya melakukan proses edit data dan tidak melakukan proses edit data. Gambar event sebagai berikut:



Gambar 6.27 Event Konfirmasi Edit Data

10. Event Konformasi Keluar Dari Menu

Event konfirmasi keluar dari menu tampil ketika user akan keluar dari form menu. Terdapat button ya untuk proses keluar dari menu dan tidak untuk membatalkan proses keluar dari menu. Gambar event sebagai berikut:



Gambar 6.29 Event Konfirmasi Keluar Dari Menu

11. Event Konformasi Keluar Dari Aplikasi

Event konfirmasi keluar dari aplikasi tampil ketika user akan keluar dari aplikasi pengadaan barang. Terdapat button ya untuk proses keluar dari aplikasi dan tidak untuk membatalkan proses keluar dari aplikasi. Gambar event sebagai berikut:



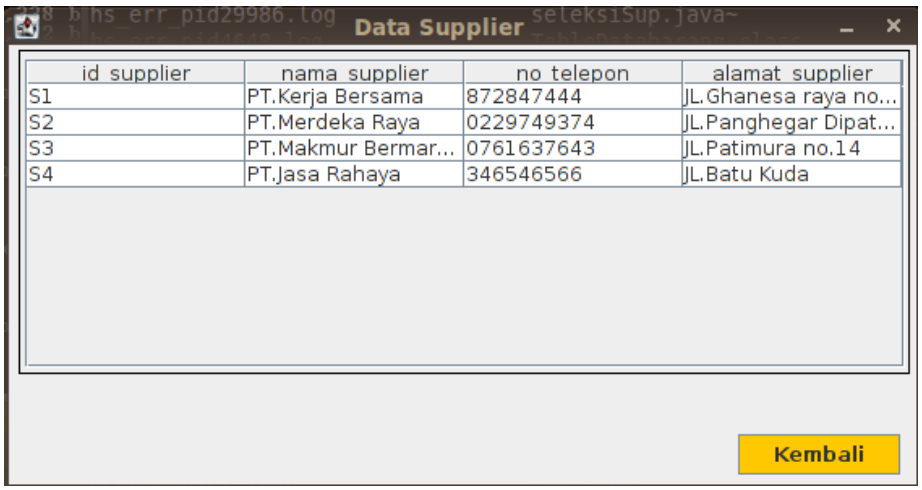
Gambar 6.30 Event Konfirmasi Keluar Dari Aplikasi

6.6.3 Laporan

laporan merupakan output dari proses input data yang akan dicetak dari aplikasi pengadaan barang.

1. Laporan Data Supplier

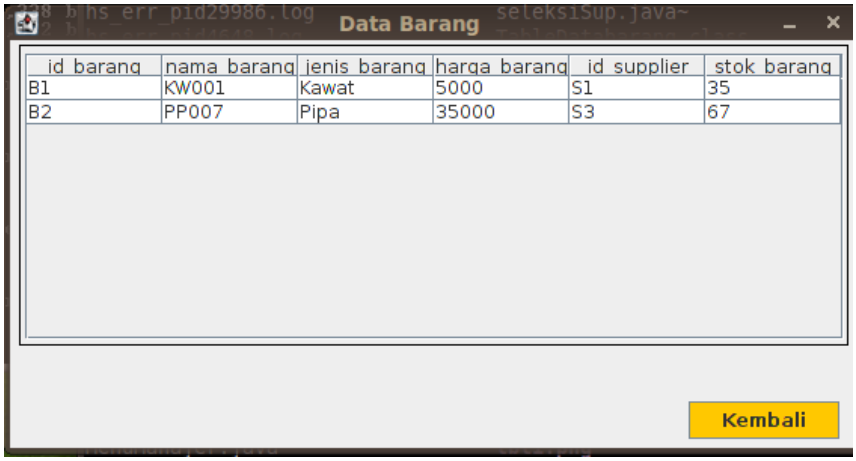
Laporan data supplier hasil keluaran yang diproses oleh manajer keluaran tersebut dihasilkan dari proses masukan data supplier yang diperankan pekerja. Didesain sederhana dengan data perusahaan sebagai identitas di bagian atas laporan dan laporan berbentuk table untuk mermudahkan dalam pembacaan.



Gambar 6.31 Laporan Data Supplier

2. Laporan Data Barang

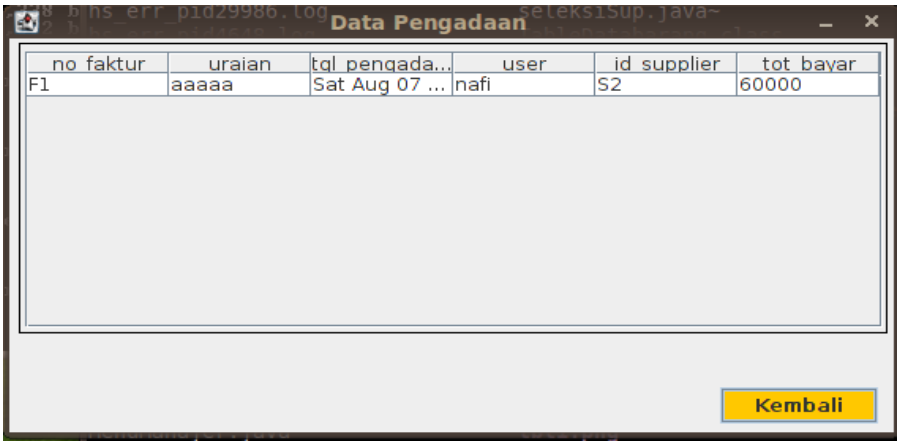
Laporan data barang hasil keluaran yang diproses oleh manajer keluaran tersebut dihasilkan dari proses masukan data barang yang diperankan pekerja. Didesain sederhana dengan data perusahaan sebagai identitas di bagian atas laporan dan laporan berbentuk table untuk memudahkan dalam pembacaan.



Gambar 6.32 Laporan Data Barang

3. Laporan Data Pengadaan

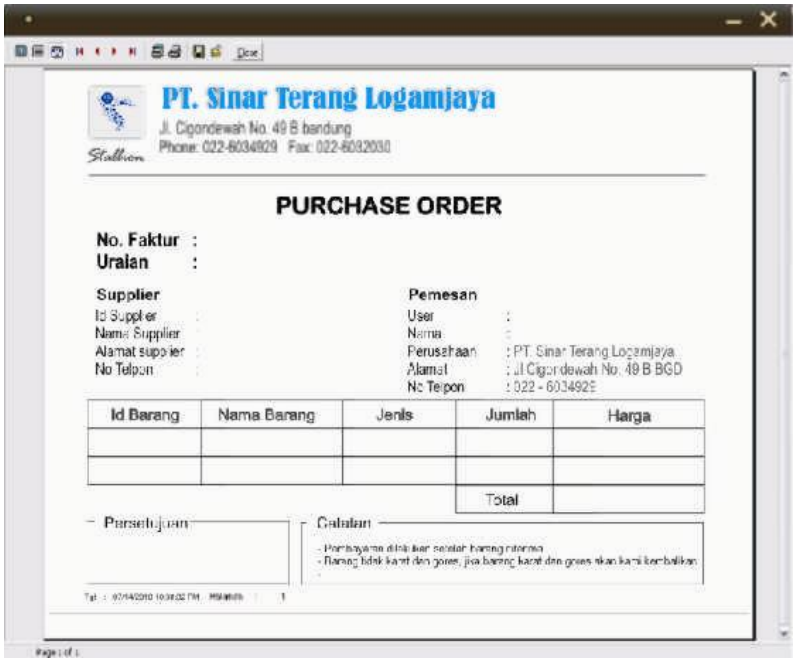
Laporan data barang hasil keluaran yang diproses oleh manajer keluaran tersebut dihasilkan dari proses masukan data pengadaan yang diperankan pekerja. Didesain sederhana dengan data perusahaan sebagai identitas di bagian atas laporan dan laporan berbentuk table untuk memudahkan dalam pembacaan.



Gambar 6.33 Laporan Data Pengadaan

4. Purchase Order (PO)

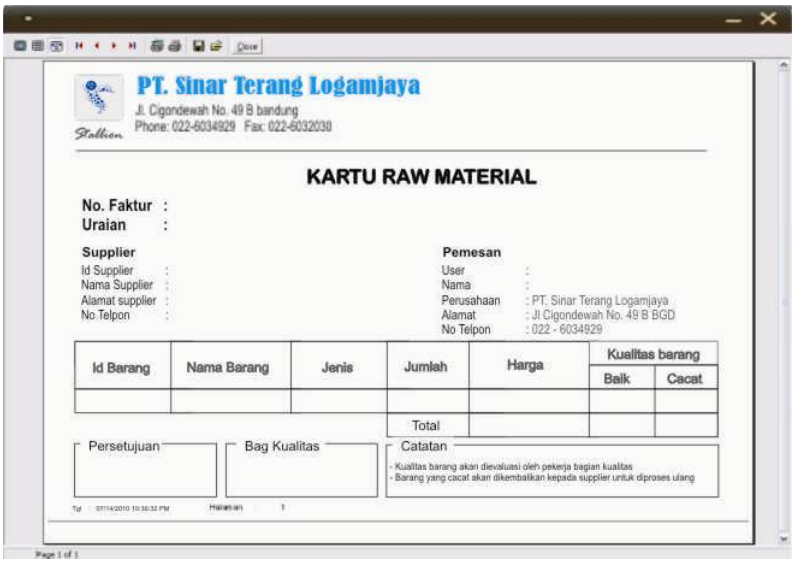
Purchase Order (PO) adalah keluaran yang diproses oleh pekerja, keluaran dicetak dalam format nota pemesanan dengan data perusahan sebagai identitas di atasnya. Purchase Order (PO) merupakan rincian data pemesanan barang, yang ditunjukan kepada supplier.



Gambar 6.34 Purchase Order (PO)

5. Kartu Raw Material (KRM)

Kartu raw material (KRM) adalah keluaran yang diproses oleh pekerja, keluaran dicetak dalam format pengecekan barang dengan data perusahan sebagai identitas di atasnya. Kartu raw material (KRM) merupakan kartu pengecekan barang yang ditunjukan kepada bagian kualitas dan di dalam format tersebut ada kolom persetujuan langsung dari bagian kualitas.



Gambar 6.35 Kartu Raw Material (KRM)

6. Bukti Penerimaan Barang

Bukti penerimaan barang (BPB) adalah keluaran yang diproses oleh pekerja, keluaran dicetak dalam format surat bukti penerimaan dengan data perusahan sebagai identitas di atasnya. Bukti penerimaan barang (BPB) merupakan surat yang dicetak setelah barang yang dipesan semuanya telah diterima dan ditunjukan kepada bagian keuangan untuk proses pembayaran . Dengan itu proses pengadaan barang selesai dilakukan.



Gambar 6.36 Bukti penerimaan barang (BPB)

6.6.4 Implementasi Fungsi

Fungsi merupakan kode pemrograman yang akan dipakai dalam tiap-tiap proses yang dilakukan oleh aplikasi pengadaan barang. Sebagai berikut adalah fungsi-fungsi proses sistem informasi pengadaan barang:

```
//Fungsi Login
try
        {
                String uname = JTFusername.getText();
                String pass = JPFpassword.getText();
                String jab = JCBJabatan.getSelectedItem().toString();
                Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
                con1 = DriverManager.getConnection("jdbc:odbc:aris","root","");
                stat = con1.prepareStatement("select * from user where Username =?");
                stat.setString(1,uname);
                rs = stat.executeQuery();
                while (rs.next())
        {
                String buname = rs.getString("Username");
                String bupass = rs.getString("Password");
                String bujab = rs.getString("Jabatan");

                if (pass.equals(bupass))
        {
                if (jab.equals(bujab))
        {
                if (bujab.equals("Pekerja"))
        {

        JOptionPane.showMessageDialog(null,"masuk ke menu pekerja");
                }
                else if (bujab.equals("Manager"))
                {

        JOptionPane.showMessageDialog(null,"masuk ke menu managerial");
                        }
                }
                else if (jab!=bujab)
                {
        JOptionPane.showMessageDialog(null,"ID atau PASSWORD anda salah!");
                        }
                }
                else if (pass!=bupass)
                {

        JOptionPane.showMessageDialog(null,bupass+"HARUS SAMA
DENGAN"+pass+"Jabatan Harus samaa antara : "+jab+" DENGAN "+bujab);
        JOptionPane.showMessageDialog(null,"Password yang anda masukkan salah,priksa
kembali");
                }
                else if(rs.getRow()==0)
                        {
                JOptionPane.showMessageDialog(null,"Anda tidak mempunyai hak");
```

```
                    }
                    else
                    {
          JOptionPane.showMessageDialog(null,"Program bermasalah hubungi Administrator");
                    }
                    }
                    con1.close();
                    }
                    catch(Exception cn)
                              {
                    JOptionPane.showMessageDialog(null,"Coneksi gagal"+cn+"");
                              }

                    }
                    else if (e.getSource() ==JBExit)
                    {
                    JOptionPane.showMessageDialog(null,"Yakin keluar aplikasi.?.C U.");
                    System.exit(0);
                    }


try
                    {
 //Membangun Koneksi
                     Class.forName( "org.gjt.mm.mysql.Driver" ).newInstance();
                     con                    =                    DriverManager.getConnection
("jdbc:mysql://localhost/test","aris","arista");
                     stmt = con.createStatement();
                     String query = "SELECT * FROM Barang";
                     ResultSet rs = stmt.executeQuery(query);

 //Genred otomatis ID dan Insert
                     if(rs.getRow()==0)
                               {
                     String IDB = "B1";
                     String NB = JTFNamaB.getText();
                     String JB = JCBJenisB.getItemAt();
                     int HB = JTFHargaB.getInteger();
                     String IS = JTFIDS.getString();
                     String          str          =          "insert          into          Barang
(id_Barang,Nama_barang,Jenis_barang,Haraga_barang,id_Supplierer) values (?,?,?,?,?)";
                     stmt.setString(1,IDB);
                     stmt.setString(2,NB);
                     stmt.setString(3,JB);
                     stmt.setInt(4,HB);
                     stmt.setString(5,IS);
                     stmt.executeUpdate();
                               }
                               else if(rs.getRow()!=0)
                               {
                     int aid = rs.getRow();
                     int bid = aid+1;
                     String IDB = "B"+bid;
                     String NB = JTFNamaB.getText();
                     String JB = JCBJenisB.getItemAt();
                     int HB = JTFHargaB.getInteger();
                     String IS = JTFIDS.getString();
                     String          str          =          "insert          into          Barang
```

```
(id_Barang,Nama_barang,Jenis_barang,Haraga_barang,id_Supplierer) values (?,?,?,?,?)";
                    stmt.setString(1,IDB);
                    stmt.setString(2,NB);
                    stmt.setString(3,JB);
                    stmt.setInt(4,HB);
                    stmt.setString(5,IS);
                    stmt.executeUpdate();

//Event informasi dan Kesalahan
                            }
                            else(SQLException eslsa)
                            {
                    JOptionPane.showMessageDialog(null,"data barang belum diisi "+e+" );
                            }
                    }
                    catch(SQLException eSQL)
                    {
JOptionPane.showMessageDialog(null,"Ada Kesalahan Pada Koneksi Database, Hubungi
Administrator");
                    }
                    return;
```

6.6.5 Analisis Kebutuhan Aplikasi

Aplikasi sistem informasi pengadaan barang diimplementasikan dengan menggunakan software pemograman Java development kit (JDK), editor pemograman eclipse, dan My SQL sebagai media penyimpanan datanya. Ada beberapa klasifikasi perangkat keras yang dibutuhkan untuk menjalankan software tersebut, dengan tujuan dapat menunjang proses implementasi aplikasi sistem informasi pengadaan barang. Dibawah ini merupakan klasifikasi perangkat keras yang dibutukan untuk menunjang aplikasi sistem informasi pengadaan barang yaitu:

Table 6.2 Analisa Klasifikasi Kebutuhan Aplikasi

Hadware	**Spesifikasi**
Matherboard	Pentium 4
Processor	P4 2.2 Ghz
Random access memory (RAM)	256 Mb
Monitor	14 “
Hardisk	20 Gb
CD-Rom	*Speed* 16X
Video graphics array (VGA)	32 Mb
Printer	Standart
Keyboard	Standart
Mouse	Standart