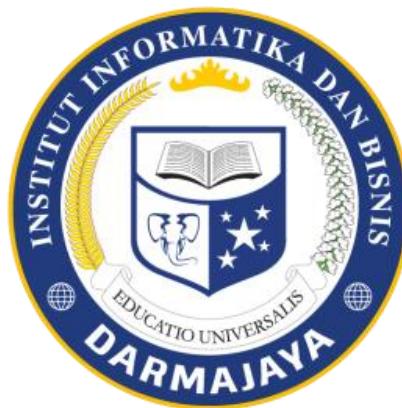


APLIKASI PENGOLAHAN DATA INVENTARIS KANTOR PADA PT PUPUK SRIWIDJAJA LAMPUNG

TUGAS AKHIR



Disusun Oleh:

Riski Ayu Septiani

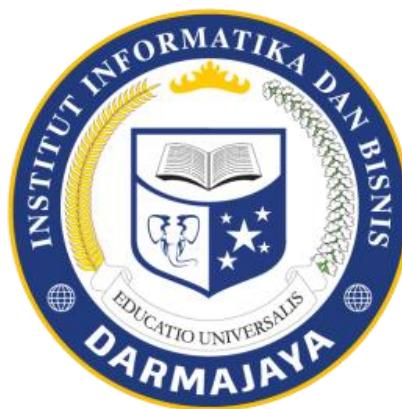
1601030012

**MANAJEMEN INFORMATIKA
FAKULTAS ILMU KOMPUTER
INSTITUT INFORMATIKA & BISNIS DARMAJAYA
BANDAR LAMPUNG
2019**

APLIKASI PENGOLAHAN DATA INVENTARIS KANTOR PADA PT PUPUK SRIWIDJAJA LAMPUNG

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat untuk Mencapai Gelar
AHLI MADYA
Pada Program Studi Manajemen Informatika
IIB Darmajaya Bandar Lampung



Disusun Oleh:

Riski Ayu Septiani

1601030012

**MANAJEMEN INFORMATIKA
FAKULTAS ILMU KOMPUTER
INSTITUT INFORMATIKA & BISNIS DARMAJAYA
BANDAR LAMPUNG
2019**



PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa tugas akhir yang saya ajukan ini adalah hasil karya saya sendiri, tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi atau karya yang pernah ditulis atau diterbitkan orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka. Karya ini adalah milik saya dan pertanggung jawaban sepenuhnya berada di pundak saya.



HALAMAN PERSETUJUAN

Judul Tugas Akhir

**: APLIKASI PENGOLAHAN DATA INVENTARIS
KANTOR PADA PT PUPUK SRIWIDJAJA
LAMPUNG**

Nama Mahasiswa

Riski Ayu Septiani

No. Pokok Mahasiswa

: 1601030012

Program Studi

: D3 Manajemen Informatika



Disetujui oleh :

Pembimbing

Ketua Program Studi

Neni Purwati, S.Kom., M.T.I
NIK. 00500802

Nurjoko, S.Kom., M.T.I
NIK. 00440702

HALAMAN PENGESAHAN

Telah diuji dan dipertahankan di depan Tim Penguji Tugas Akhir
Program Studi Manajemen Informatika IIB Darmajaya Bandar dan dinyatakan
diterima untuk memenuhi syarat guna memperoleh gelar

Ahli Madya Komputer

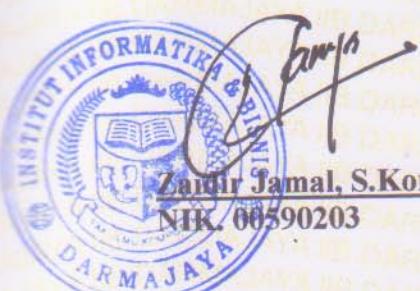
Mengesahkan

1. Tim Penguji

Anggota 1 : Arman Suryadi K, S.Kom., M.T.I

Anggota 2 : Ochi Marshella FA, S.Kom., M.T.I

2. Dekan Fakultas Ilmu Komputer



Zainir Jamal, S.Kom., M.Eng
NIK. 00590203

Tanggal Lulus Ujian Skripsi : 18 September 2019

PERSEMBERAHAN

Dengan penuh rasa syukur saya ucapkan kepada Tuhan, karena berkat Dia saya mampu meyelesaikan laporan tugas akhir ini dengan sebaik-baiknya.

Selain itu penulis mempersembahkan laporan ini untuk:

Orang tua ku terCinta (MatTubin dan Maryani), kakak tersayang (Asep saputra dan Edwin satria), ayuk ipar (Yogi chintya M dan Sundariyati) dan ponakan ku (Fellychia Risky Yochiatra). Berkat kalianlah aku bisa jadi seperti ini. Sehingga aku mampu menyelesaikan tugas akhir ini sesuai yang kita harapkan bersama. Peran kalian begitu besar buatku. Terimakasih atas dukungan dan doa nya selama ini.

Dosen pembimbing ku ibu Neni Purwati, S.Kom., M.T.I yang telah banyak meluangkan waktunya dalam memberikan bimbingan dan pengarahan kepada ku sehingga tugas akhir ini dapat diselesaikan.

Orang terdekatku khusus nya (Yogi Hardiansyah), Teman-temanku yang tiada duanya (tina, fia, chendy, faii, hary, irma,yessi, kaka han, mas pur, oppa febry, onnie edil) dan teman-teman MI 2016 serta anak-anak Hima Misi terimakasih untuk dukungannya dalam menyemangati aku dan memberikan inspirasi dalam menyusun tugas akhir ini, serta semua pihak yang telah membantu yang tidak mungkin disebutkan satu persatu Terimakasih.

Almamater tercinta IIB DARMAJAYA yang telah memberikan ilmu kepadaku.

MOTTO

*“Bertekat untuk membuktikan bahwa
hidup mu BerHARGA”*

Riski Ayu Septiani...

INTISARI

Aplikasi pengolahan data inventaris kantor pada PT pupuk sriwidjaja
lampung

Oleh

Riski Ayu Septiani

Ikiayu039@gmail.com

PT Pupuk Sriwidjaja merupakan perusahaan yang didirikan sebagai pelopor pupuk urea di indonesia dengan tujuan utamanya melaksanakan dan menunjang kebijaksanaan program pemerintah di bidang ekonomi dan pembangunan nasional khususnya di industri pupuk dan kimia. Dalam penggunaan metode pengembangan sistem baru penulis menggunakan Metode *waterfall*, dimana semua proses yang dilakukan secara berurutan sesuai dengan urutan yang ada. Model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap *maintence*. Pengumpulan data dilakukan dengan metode wawancara, observasi dan studi pustaka sedangkan implementasi dari metode ini dijelaskan melalui analisis sistem, DFD dan struktur tabel *database*. Untuk merancang dan membuat *aplikasi* ini, penulis menggunakan *Netbeans 7.1*, bahasa pemrograman *Java*, *database MySql*, dan *webserver Xampp*. Implementasi aplikasi pengolahan data Inventaris kantor pada PT Pupuk Sriwidjaja bisa menampilkan tentang data Inventaris yang ada di suatu aplikasi komputer. Ada pun kelebihan dari Sistem Informasi ini antara lain PT Pupuk Sriwidjaja dapat mengontrol data Inventaris dan informasi yang disajikan dapat dilihat dengan mudah dan cepat, sehingga hal ini dapat memudahkan pihak PT Pupuk Sriwidjaja mendapatkan informasi Inventaris yang ada dimanapun dan kapanpun sehingga membuat semuanya menjadi lebih efektif dan efisien.

Kata kunci: PT Pupuk Sriwidjaja, Java, *MySql*, *Xampp*.

ABSTRACT

Office inventory data processing application at PT Pupuk Sriwidjaja
Lampung

By

Riski Ayu Septiani

Ikiayu039@gmail.com

PT Pupuk Sriwidjaja is a company established as a pioneer of urea fertilizer in Indonesia with the main objective of implementing and supporting government policy in the economic and national development sectors, especially in the fertilizer and chemical industry. In the use of a new system development method the author uses the waterfall method, where all processes are carried out sequentially in accordance with the existing order. The waterfall model provides a sequential or sequential software life cycle approach starting from the analysis, design, coding, testing, and maintenance stages. Data collection is done by interview, observation and literature study while the implementation of this method is explained through system analysis, DFD and database table structure. To design and create this application, the author uses Netbeans 7.1, the Java programming language, MySql database, and the Xampp webserver. The Office Inventory data processing application implementation at PT Pupuk Sriwidjaja can display Inventory data in a computer application. There are also advantages of this Information System, among others, PT Pupuk Sriwidjaja can control Inventory data and the information presented can be seen easily and quickly, so this can make it easier for PT Pupuk Sriwidjaja to get Inventory information available anywhere and at any time so as to make everything more effective and efficient.

Keywords: PT Pupuk Sriwidjaja, java, MySql, Xampp.

PRAKATA

Puji syukur kita panjatkan kepada Allah SWT yang telah memberikan nikmat serta hidayah-Nya terutama nikmat kesempatan dan kesehatan sehingga saya dapat menyelesaikan penyusunan tugas akhir ini. Shalawat serta salam kita sampaikan kepada Nabi besar kita Muhammad SAW yang telah memberikan pedoman hidup yakni Al-Qur'an dan sunnah untuk keselamatan umat di dunia.

Dalam penyusunan tugas akhir ini saya menyampaikan ucapan terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam menyelesaikan tugas ini, khususnya kepada:

- Tuhan YME atas rahmat-Nya sehingga saya dapat menyelesaikan tugas akhir ini dengan baik.
- Bapak Ir. Firmansyah Y. Alfian, MBA., MSc selaku Rektor IIB Darmajaya
- Bapak Nurjoko S. Kom., M.T.I selaku ketua jurusan sistem informasi IIB Darmajaya.
- Kedua orang tua yang telah mendukung kegiatan saya di kampus.
- Ibu Neni Purwati, S.Kom., M.T.I selaku dosen pembimbing.
- Dan semua pihak yang telah membantu menyelesaikan penyusunan tugas akhir ini.

DAFTAR ISI

PERNYATAAN.....	Error! Bookmark not defined.
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN.....	Error! Bookmark not defined.
PERSEMBAHAN	Error! Bookmark not defined.
MOTTO	Error! Bookmark not defined.
INTISARI	Error! Bookmark not defined.
ABSTRACT	Error! Bookmark not defined.
PRAKATA.....	Error! Bookmark not defined.
DAFTAR ISI.....	1
DAFTAR TABEL	Error! Bookmark not defined.
DAFTAR GAMBAR.....	Error! Bookmark not defined.
BAB I PENDAHULUAN.....	Error! Bookmark not defined.
1.1. Latar Belakang	Error! Bookmark not defined.
1.2. Ruang Lingkup.....	Error! Bookmark not defined.
1.3. Rumusan Masalah.....	Error! Bookmark not defined.
1.4. Tujuan Penelitian	Error! Bookmark not defined.
1.5. Manfaat Penelitian	Error! Bookmark not defined.
1.6 Sistematika Penulisan	Error! Bookmark not defined.
BAB II LANDASAN TEORI	Error! Bookmark not defined.
2.1 Basis Data (<i>Database</i>)	Error! Bookmark not defined.
2.2 Alat Bantu Perancangan Sistem	Error! Bookmark not defined.
2.3 Aplikasi yang digunakan.....	Error! Bookmark not defined.
2.4 Metodologi pengembangan sistem.....	Error! Bookmark not defined.
BAB III METODOLOGI PENELITIAN	Error! Bookmark not defined.
3.1 Metode Pengembangan Sistem	Error! Bookmark not defined.
3.2 Metode Pengumpulan Data.....	Error! Bookmark not defined.
3.2.1 Tempat dan Waktu Penelitian	Error! Bookmark not defined.

3.3 Analisis Sistem.....	Error! Bookmark not defined.
3.3.1 Analisis Sistem yang berjalan	Error! Bookmark not defined.
3.3.2 Analisis usulan sistem.....	Error! Bookmark not defined.
3.4 Diagram Alir.....	Error! Bookmark not defined.
3.5 Desain (perancangan) Sistem Secara Terperinci.....	Error! Bookmark not defined.
3.6. Rancangan <i>Flowchart Program</i>	Error! Bookmark not defined.
BAB IV HASIL PENELITIAN DAN PEMBAHASAN....	Error! Bookmark not defined.
4.1 Hasil Penelitian	Error! Bookmark not defined.
4.1.1 Tampilan Program.....	Error! Bookmark not defined.
BAB V SIMPULAN DAN SARAN.....	Error! Bookmark not defined.
5.1 Simpulan	Error! Bookmark not defined.
5.2 Saran	Error! Bookmark not defined.

DAFTAR GAMBAR

1. Gambar 2.1 Metode waterfal	12
2. Gambar 3.1 Ilustrasi Metode Waterfall	15
3. Gambar 3.2 bentuk bagan alir dokumen sistem yang berjalan pengolahan data inventaris kantor	18
4. Gambar 3.3 <i>Context diagram</i> Sistem Diusulkan.....	20
5. Gambar 3.4 <i>DFD level 0</i> Sistem Informasi Pengolahan Data Inventaris Kantor	21
6. Gambar 3.5 Rancangan Form <i>Login</i>	22
7. Gambar 3.6 Rancangan Form Menu Utama.....	22
8. Gambar 3.7 Rancangan Laporan Data Barang	23
9. Gambar 3.8 Rancangan Laporan Pembelian Barang.....	23
10. Gambar 3.9 Rancangan Permintaan Barang.....	24
11. Gambar 3.10 Rancangan Pembelian Barang	24
12. Gambar 3.11 Rancangan Form Input Data Barang	25
13. Gambar 3.12 Rancangan Form Input Data Pegawai	25
14. Gambar 3.13 Rancangan Form Input Data Lokasi.....	26
15. Gambar 3.14 Rancangan Form Input Data Inventaris.....	26
16. Gambar 3.15 Rancangan Form Input Permintaan Barang.....	27
17. Gambar 3.16 Rancangan Form Input Pembelian Barang	27
18. Gambar 3.17 Relasi Tabel	28
19. Gambar 3.18 struktur <i>menu</i> utama	34
20. Gambar 3.19 rancangan <i>flowchart</i> program <i>menu Login</i>	35
21. Gambar 3.20 rancangan <i>flowchart</i> program <i>menu</i> utama	36
22. Gambar 3.21 rancangan <i>flowchart</i> program <i>menu Pegawai</i>	37
23. Gambar 3.22 rancangan <i>flowchart</i> program <i>menu Barang</i>	38
24. Gambar 3.23 rancangan <i>flowchart</i> program <i>menu Lokasi</i>	39
25. Gambar 3.24 rancangan <i>flowchart</i> program <i>menu Permintaan</i>	40
26. Gambar 3.25 rancangan <i>flowchart</i> program <i>menu Pembelian</i>	41

27. Gambar 3.26 rancangan <i>flowchart</i> program <i>menu Inventaris</i>	42
28. Gambar 3.27 rancangan <i>flowchart</i> program <i>menu Laporan</i>	43
29. Gambar 4.1 Tampilan form <i>Home</i>	44
30. Gambar 4.2 Tampilan Menu Utama	45
31. Gambar 4.3 Tampilan Form Barang.....	45
32. Gambar 4.4 Tampilan form Pegawai	46
33. Gambar 4.5 Tampilan form Lokasi	46
34. Gambar 4.6 Tampilan form Inventaris	47
35. Gambar 4.7 Tampilan form permintaan barang	47
36. Gambar 4.8 Tampilan form pembelian barang	48
37. Gambar 4.9 Tampilan form Laporan Inventaris Kantor.....	48
38. Gambar 4.10 Tampilan Cetak Permintaan Barang.....	49
39. Gambar 4.11 Tampilan Cetak Pembelian Barang	49
40. Gambar 4.12 Tampilan Cetak Laporan Inventaris Kantor	50
41. Gambar 4.13 Tampilan Cetak Laporan Pembelian Barang.....	50

DAFTAR TABEL

1. Tabel 2.1 Entity Relationship Diagram (ERD)	5
2. Table 2.2 Simbol-simbol dalam Kamus Data	7
3. Table 2.3 simbol diagram contex	7
4. Tabel 2.4 symbol DFD	8
5. Table 2.5 Simbol Bagan Alir Dokumen (<i>Mapping Chart</i>)	9
6. Tabel 2.6 Simbol <i>Flowchart</i> Program.....	10
7. Tabel 3.1 Tabel login	29
8. Tabel 3.2 Tabel Barang	29
9. Tabel 3.3 Tabel Pegawai	20
10. Tabel 3.4 Tabel Lokasi.....	30
11. Tabel 3.5 Tabel Inventaris.....	31
12. Tabel 3.6 Tabel Permintaan	31
13. Tabel 3.7 Tabel Detatail Permintaan.....	32
14. Tabel 3.8 Tabel Pembelian.....	32
15. Tabel 3.9 Tabel Detail Pembelian	33

BAB I

PENDAHULUAN

1.1.Latar Belakang

Semakin berkembangnya teknologi informasi sangat membantu kita dalam melakukan pekerjaan dibidang pengolahan data yang lebih akurat, dan tuntutan pekerjaan yang membutuhkan waktu yang lebih cepat dalam penggerjaannya. Menghadapi tantangan dalam era globalisasi dibutuhkan suatu teknologi modern yang menuntut kecepatan dan ketepatan dengan fasilitas yang dibutuhkan untuk mendukung semua kegiatan dalam pekerjaan.

Dalam suatu perusahaan seorang pimpinan harus selalu tanggap dan dapat mengetahui segala informasi secara akurat dan relevan, untuk mengatasi masalah yang tidak dikehendaki dalam mencapai tujuan yang ditetapkan. Bagian yang bertanggung jawab untuk mengelola data inventaris yaitu bagian staf penjualan dan umum.

PT Pupuk Sriwidjaja merupakan perusahaan yang didirikan sebagai pelopor pupuk urea di indonesia. PT Pupuk Sriwidjaja memulai usaha dengan tujuan utamanya melaksanakan dan menunjang kebijaksanaan program pemerintah di bidang ekonomi dan pembangunan nasional khususnya di industri pupuk dan kimia. Adapun pengolahan data inventaris kantor PT Pupuk Sriwidjaja masih tercatat dalam bentuk dokumen (Microsoft Eexcel). Hal ini tentunya akan menimbulkan masalah dalam pengolahan data inventaris yaitu memperlambat proses pencarian, kurang terkontrolnya barang yang sedang digunakan atau yang telah rusak. Bagian staf penjualan dan umum mempunyai kendala untuk melihat inventaris yang sedang digunakan atau yang sudah tidak layak pakai dan yang membutuhkan pemeliharaan secara rutin.

Adapun inventaris yang tercatat dikantor PT Pupuk Sriwidjaja lampung terbagi menjadi 3 kategori yaitu:

1. Kategori Kendaraan (mobil dan motor)
2. Kategori Elektronik (Computer, CPU, Printer, Proyektor, AC, Tv, Mesin photo copy, Layar proyektor, Telpon, Mesin tik, Mesin fax, Salon, Toa, Mesin absen, kipas angin, Kulkas)
3. Kategori Perlengkapan (Meja, Kursi, peralatan makan, accessoris ruangan, rak dan lemari)

Dengan begitu dibutuhkan program untuk pengolahan data inventaris agar perusahaan lebih mudah menginventarisir, mengecek dan mengontrol barang-barang inventaris tersebut.

Berdasarkan latar belakang diatas maka penulis tertarik untuk melakukan suatu perancangan sistem informasi pengolahan data inventaris pada PT Sriwidjaja untuk membantu dalam proses pengolahan data inventaris. Oleh karena itu penulis memberi judul untuk penelitian ini yaitu “**Sistem Pengolahan Data Inventaris kantor pada PT Pupuk Sriwidjaja Lampung**”.

1.2.Ruang Lingkup

Agar permasalahan yang dibahas tidak meluas dengan permasalahan lainnya maka penulis membatasi masalah yang akan dibahas pada pengolahan data inventaris barang yang terdapat di kantor PT Pupuk Sriwidjaja Lampung.

1.3.Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan diatas untuk mempermudah pengolahan data inventaris pada PT Pupuk Sriwidjaja lampung maka dapat dirumuskan masalah sebagai berikut:

“Bagaimana merancang sebuah program aplikasi data inventaris kantor pada PT Pupuk Sriwidjaja Lampung”.

1.4.Tujuan Penelitian

Untuk membuat sebuah program aplikasi pengolahan data inventaris kantor pada PT Pupuk Sriwidjaja Lampung, agar mempermudah dalam mencari data inventaris kantor dengan cepat dan tepat.

1.5.Manfaat Penelitian

Manfaat membuat program aplikasi pengolahan data inventaris kantor pad PT Pupuk Sriwidjaja Lampung adalah sebagai berikut:

1. PT Pupuk sriwidjaja diharapkan mampu menggunakan aplikasi pengolahan data inventaris kantor sehingga dapat meningkatkan pengunaan komputer secara maksimal.
2. Mempermudah dalam pekerjaan dengan sistem pengolahan data inventaris barang yang ada di Kantor PT pupuk sriwidjaja lampung sehingga kecepatan operasional menjadi lebih efektif, cepat dan akurat.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN

Berisi latar belakang masalah, rumusan masalah, batasan masalah, manfaat penelitian/kontribusi, tujuan penelitian, tinjauan pustaka dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Berisi uraian atau dasar teori teori pendukung yang berkaitan dengan topik sistem informasi pengolahan data inventaris pada PT Pupuk Sriwidjaja Lampung.

BAB III METODOLOGI PENELITIAN

Berisi pembahasan mengenai bagaimana metode yang akan digunakan dalam penelitian yang akan dibuat, menganalisis sistem, menganalisis kebutuhan sistem, perancangan sistem, perancangan basis data serta perancangan tampilan *input output*.

BAB IV HASIL dan PEMBAHASAN

Berisi hasil pembahasan dari penelitian yang meliputi implementasi rancangan aplikasi serta fungsi dan kinerja penerapan sistem informasi pengolahan data inventaris kantor pada PT Pupuk Sriwidjaja Lampung.

BAB V KESIMPULAN dan SARAN

Berisi tentang simpulan dari hasil analisis masalah dalam penelitian serta saran.

DAFTAR PUSTAKA**LAMPIRAN**

BAB II

LANDASAN TEORI

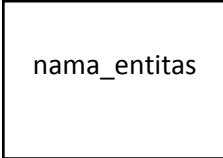
1.1 Basis Data (*Database*)

Basis data atau Database adalah kumpulan informasi yang disusun dan merupakan suatu kesatuan yang utuh yang disimpan di dalam perangkat keras (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak. Dengan sistem tersebut data yang terhimpun dalam suatu database dapat menghasilkan informasi yang berguna. Ganda Yoga Swara, M.Kom, Yunes Pebriadi (2016:30)

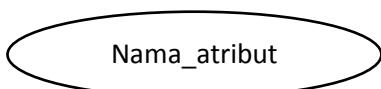
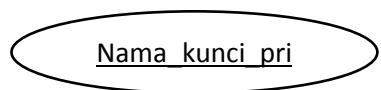
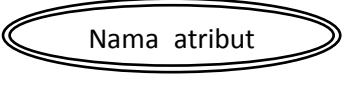
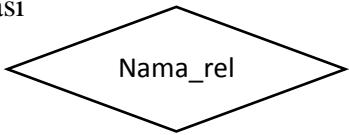
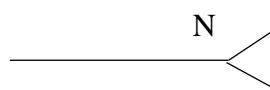
2.1.1 Entity Relationship Diagram (ERD)

Relasi antar tabel harus dirancang untuk sebaik mungkin untuk mendukung kelancaran dalam pengolahan. Tabel-tabel yang telah berelasi memudahkan programmer dalam menentukan melaksanakan kerja dalam proses pengumpulan data, pembentukan tabel dan pembentukan laporan. Entity relational data merupakan cara untuk mengorganisasi data, dimana diagram ini akan memperlihatkan hubungan entitas yang terdapat didalam sistem. Ganda Yoga Swara, M.Kom, Yunes Pebriadi (2016:33).

Tabel 2.1 Entity Relationship Diagram (ERD)

Simbol	Deskripsi
Entitas/ <i>entity</i>  nama_entitas	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama table

Tabel lanjutan 2.1 Entity Relationship Diagram (ERD)

Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
Atribut multivilai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas dimana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian .Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B

2.1.2 Kamus data

Sukamto dan Shalahuddin (2014:73), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (output) dapat dipahami secara umum (memiliki standar cara penulisan).”

Sukamto dan Shalahuddin (2014:73), menjelaskan simbol-simbol yang digunakan dalam kamus data, yaitu :

Table 2.2 Simbol-simbol dalam Kamus Data

No.	Simbol	Arti
1.	=	disusun atau terdiri atas
2.	+	Dan
3.	[]	baik ...atau...
4.	{ } ⁿ	n kali diulang/ bernilai banyak
5.	()	data operasional
6.	*...*	batas komentar

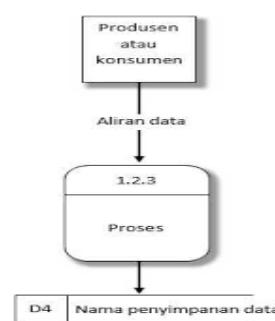
2.2 Alat Bantu Perancangan Sistem

Alat bantu perancangan sistem yaitu, Context Diagram dan DFD (Data Flow Diagram). Berikut adalah penjelasan dari alat bantu perancangan sistem.

2.2.1 Context Diagram

Context Diagram adalah diagram yang dibuat untuk menggambarkan sumber serta tujuan data yang akan diproses atau dengan kata lain diagram tersebut digunakan untuk menggambarkan sistem secara umum atau global dari keseluruhan sistem yang ada. Context Diagram dapat dilihat pada gambar di bawah ini. Indah Setia Andani, Diana Laily Fithri(2016:270).

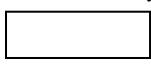
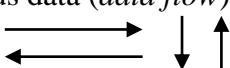
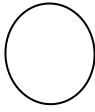
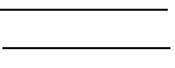
Table 2.3 simbol diagram contex



2.2.2 Data Flow Diagram (DFD)

Menurut Ganda Yoga Swara, M.Kom, Yunes Pebriadi (2016:32) Data Flow Diagram (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan harus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. DFD merupakan alat bantu dalam menggambarkan atau menjelaskan sistem yang berjalan secara logis. Beberapa simbol dari DFD dapat dilihat pada tabel di bawah ini:

Tabel 2.4 symbol DFD

Simbol	Keterangan
(<i>external entity</i>) 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem
Arus data (<i>data flow</i>) 	Menggambarkan arus data
Proses (<i>process</i>) 	Merupakan kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar
Simpanan data (<i>data store</i>) 	Merupakan komponen yang berfungsi untuk menyimpan data atau file.

2.2.3 Simbol-simbol Bagan alir Dokumen (*Mapping Chart*)

Diagram alir dokumen adalah diagram alir yang menunjukkan arus dari laporan keuangan termasuk tembusan-tembusan. Bagian ini menggambarkan dalam system dari satu bagian kebagian lain. Simbol - simbol yang digunakan dalam diagram alir dokumen dapat dilihat dibawah ini Jogiyanto (2014:84).

Table 2.5 Simbol Bagan Alir Dokumen (*Mapping Chart*)

Simbol	Keterangan
	<i>Terminator</i> Menunjukkan awal dan akhir suatu proses
	<i>Document</i> Menunjukkan dokumen <i>input</i> atau <i>output</i> , baik untuk manual, mekanik komputer atau laporan cetak
	<i>Manual Operation</i> Menunjukkan pekerjaan manual
	<i>Process</i> Menunjukkan kegiatan proses dari operasi program computer
	<i>Display</i> Menunjukkan <i>Layout</i> pada computer
	<i>Keyboard</i> Memasukkan atau <i>input</i> data melalui keyboard
	<i>Harddisk</i> Tempat penyimpanan data pada komputer
	<i>Simpan Offline</i> Penyimpanan arsip manual
	Menunjukkan penjelasan suatu proses
	<i>Connector</i> Menunjukkan penghubung ke halaman yang sama
	<i>Off-page Connentor</i> Menunjukkan penghubung ke halaman yang berbeda
	<i>Garis Alir</i> Menunjukkan simbol dari alur sistem atau proses

2.2.4 Simbol-simbol Bagan Alir Program (*Program Flowchart*)

Menggambarkan arus logika dari data yang akan diperoses kedalam suatu Bagan alir program (*Program Flowchart*) adalah bagian flowchart yang program mulai dari awal sampai akhir. Bagan alir merupakan alat yang berguna bagi program meruntuk mempersiapkan program yang rumit. (Jogiyanto, 2014:90). Simbol - simbol yang digunakan dalam diagram alir program dapat dilihat di bawah ini.

Tabel 2.6 Simbol *Flowchart* Program

Simbol	Keterangan
	<i>Terminal</i> Berfungsi untuk memulai dan mengakhiri suatu program
	<i>Proses</i> Suatu simbol yang menunjukkan setiap pengolahan yang dilakukan oleh computer
	<i>Input – Output</i> Digunakan untuk memasukan data maupun menunjukkan hasil dari proses
	<i>Decision</i> Suatu kondisi yang menghasilkan beberapa kemungkinan jawaban atau pilihan
	<i>Connector</i> Suatu prosedur akan masuk dan keluar melalui simbol ini
	<i>OFF Line Connector</i> Merupakan simbol untuk masuk dan keluarnya suatu prosedur.
	<i>Arus atau Flow</i> Prosedur yang dilakukan dari atas kebawah, bawah ke atas, dari kiri kekanan, dari kanan kekiri

2.3 Aplikasi yang digunakan

2.3.1 MySQL

MySQL merupakan server database open source yang digunakan terbanyak di dunia. Arsitekturnya yang khas menyebabkan MySQL ini sangat cepat

dalam operasinya dan mudah disesuaikan dengan kebutuhan. Penggunaan ulang kode dan pendekatan yang minimalis menghasilkan fitur yang kaya fungsi dan menghasilkan DBMS yang kecepatan, kekompakan, kestabilan dan kemudahan dalam penerapannya sangat sulit untuk ditandingi. Pemisahan kode antara server utama dan penanganan tabel memungkinkan MySQL dijalankan pada suatu kendali transaksi yang ketat, atau dengan akses disk tanpa transaksi yang sangat cepat. Sistem penanganan tabel InnoDB kini memberikan fitur tambahan server MySQL standard, termasuk dukungan transaksi secara penuh, row-level locking dan persyaratan foreign-key. MySQL akan mendukung lalu lintas yang aman antara client server, sehingga menaikkan keamanan terhadap serangan dan akses tak sah.

Aldian Hidayat,Syaafrika Deni Rizki, Dhio Saputra (2016:41).

2.3.2 Netbeans

Netbeans yang digunakan yaitu netbeans versi 7.1. Menurut Wahana Komputer (2015:20) “Netbeans adalah salah satu aplikasi IDE yang digunakan oleh developer software komputer untuk menulis, meng-compile, mencari kesalahan, dan untuk menyebarkan program”. Sedangkan menurut Nofriadi (2015:4) mengemukakan bahwa “Netbeans merupakan sebuah aplikasi Integrated Development Environment (IDE) yang berbasis Java dari Sun Microsystems yang berjalan di atas swing dan banyak digunakan sekarang sebagai editor untuk berbagai bahasa pemrograman”

2.3.3 iReport

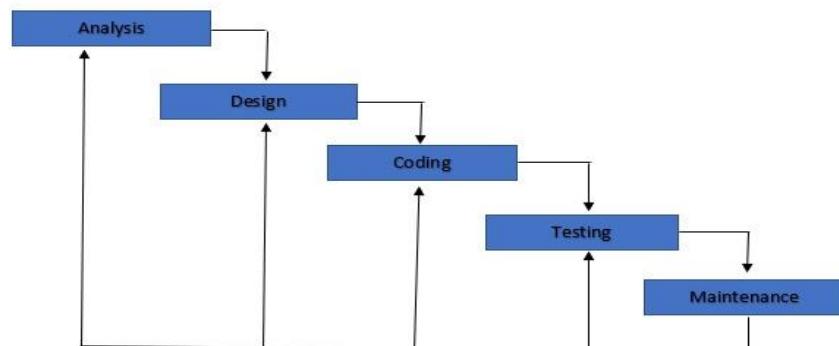
Menurut Wahana Komputer (2015:197) “iReport adalah tools yang memudahkan anda dalam pembuatan laporan. iReport biasanya digunakan dalam bahasa pemrograman Java”.

2.3.4 Xampp

Menurut Syahrul Suci Romadhon, Desmulyati (2019:24) menyatakan bahwa“ XAMPP adalah software yang membungkus Apache HTTP Server, MariaDB, PHP dan Perl. Dengan menggunakan XAMPP installasi paket software yang dibutuhkan untuk proses pengembangan web (Apache HTTP Server, MariaDB dn PHP) dapat dilakukan dengan sangat mudah, tanpa harus dilakukan secara terpisah (sendiri-sendiri)”.

2.4 Metodologi pengembangan sistem

Metode air terjun atau yang sering disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan Tahapan metode *waterfall* dapat dilihat pada gambar di bawah ini. Riyan Naufal Hay's, Ahmad Sugiyarta, Dassy Eka Winungkas (2018:23).



Gambar 2.1 Metode waterfall

2.4.1 Analisis

Analisis Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh user.

2.4.2 Design

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengkodean (Coding). Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

2.4.3 Coding (Pengkodean)

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

2.4.4 Testing (Pengujian)

Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. Pada tahap pengujian digunakan alat pengujian aplikasi black box testing. Menurut Iskandaria (2012), Pengujian blackbox (blackbox testing) adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada input dan output aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum). Tahap pengujian merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak.

2.4.5 Maintenance

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikiriman ke user. Perubahan bisa terjadi karena adanya kesalahan yang

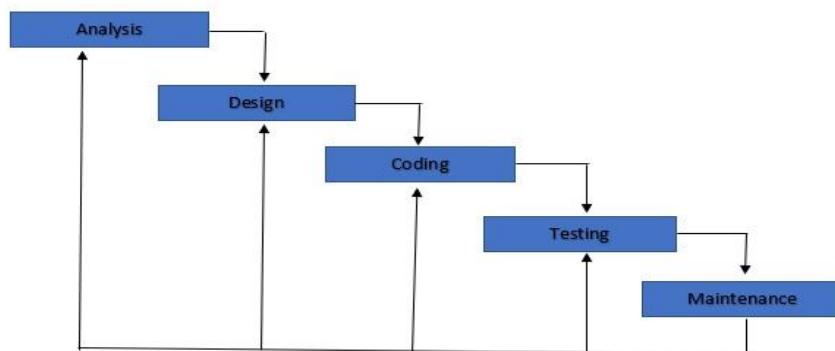
muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

BAB III

METODOLOGI PENELITIAN

3.1.Metode Pengembangan Sistem

Berikut adalah gambar model air terjun menurut . Riyan Naufal Hay's, Ahmad Sugiyarta, Dassy Eka Winungkas:



Gambar 3.1 Ilustrasi Metode Waterfall

Tahapan pengembangan pada metode *Waterfall*, terdiri dari:

1. Analisis

Sistem yang diharapkan pengguna adalah system yang bisa memberikan informasi detail mengenai “Aplikasi pengolahan data inventaris kantor pada PT. Pupuk Sriwidjaja lampung”

2. Desain

Pada tahap desain penulis membuat rancangan interface dengan cara mendesain rancangan input dan output yang akan dipakai pada system yang diusulkan.

3. Pembuatan kode program

Tahapan ini adalah tahapan dimana penulis pada system yang diusulkan dengan membuat sebuah aplikasi yaitu Neatbeans, server local XAMPP dan database PHP MyAdmin.

4. Pengujian

Tahapan ini adalah tahap dimana penulis melakukan pengujian terhadap system yang dibuat sebelum digunakan. Serta membandingkan dengan system yang sebelumnya , untuk memunculkan keunggulan pada system yang lama dan mengurangi kesalahan pada system yang baru.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan yang dilakukan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

3.2 Metode Pengumpulan Data

Pada tahapan ini dilakukan proses pengumpulan data yang digunakan dalam menyusun serta melengkapi data adalah dengan cara observasi, wawancara dan studi pustaka.

a. Observasi

Pengamatan langsung diadakan untuk memperoleh data yang dilakukan pada tempat penelitian yang terkait dengan bagian inventaris kantor di PT. Pupuk Sriwidjaja lampung.

b. Wawancara

Wawancara dilakukan dengan cara berkomunikasi langsung dengan pihak yang bertanggung jawab yaitu Rahmad agung wicaksono bagian umum di PT. Pupuk Sriwidjaja lampung.

c. Studi Pustaka

Studi pustaka dilakukan untuk memperoleh data dan informasi di tempat penelitian. Adapun data yang didapat yaitu daftar inventaris yang ada di kantor PT. Pupuk Sriwidjaja lampung.

3.2.1 Tempat dan Waktu Penelitian

Adapun tempat dan waktu penelitian yang akan dilaksanakan adalah sebagai berikut:

a. Tempat

Penelitian yang akan dilakukan yaitu pada bagian inventaris kantor di PT. Pupuk Sriwidjaja kantor pemasaran pusri daerah lampung Jl.Dr.Susilo No.57 Bandar Lampung 35214.

b. Waktu

Waktu penelitian dilaksanakan selama 1 bulan dimulai dari tanggal 26 juli 2019 sampai dengan 29 agustus 2019.

3.3 Analisis Sistem

Analisis sistem dilakukan untuk mengidentifikasi permasalahan pada sistem dan proses bisnis di PT Sriwidjaja dan tahapan ini dilakukan dengan beberapa tahapan yang meliputi :

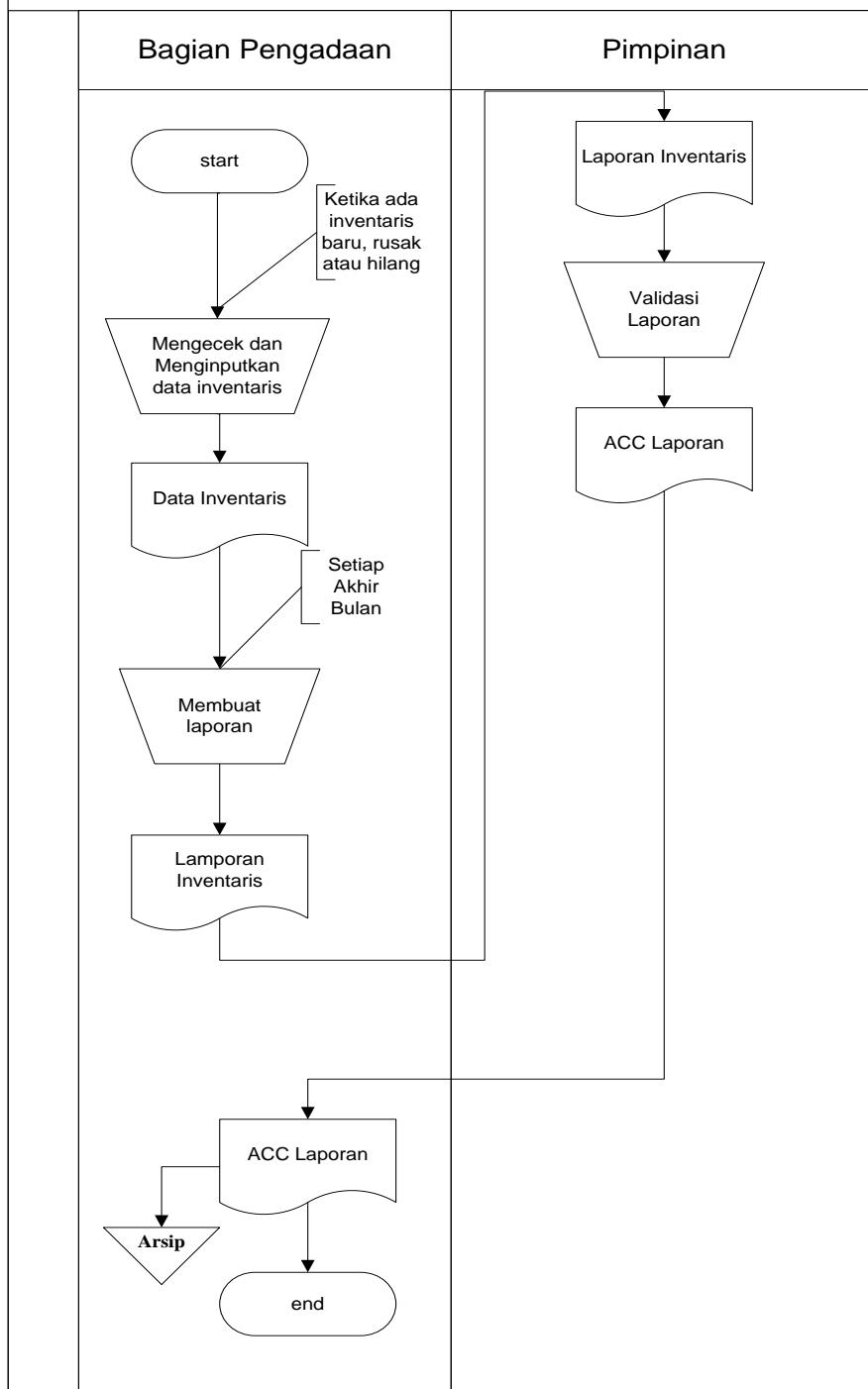
3.3.1 Analisis Sistem yang berjalan

Analisis sistem berjalan ini dilakukan dengan menggambarkan alur dari sistem pengolahan data inventaris kantor yang berjalan pada PT Sriwidjaja.adapun sistem informasi pengolahan data inventaris kantor yang berjalan di PT Sriwidjaja.

Sistem yang berjalan pengolahan data inventaris kantor :

1. Ketika ada inventaris kantor baru, rusak atau hilang bagian pengadaan mengecek inventaris.
2. Kemudian bagian pengadaan menginputkan data hasil pengecekan inventaris serta kondisi inventaris di *microsoft excel*
3. Kemudian setiap akhir bulan bagian pengadaan membuat laporan inventaris kantor kemudian diserahkan kepada pimpinan.

Sistem yang berjalan pengolahan data inventaris kantor



Gambar 3.2 bentuk bagan alir dokumen sistem yang berjalan pengolahan data inventaris kantor

3.3.1.1 Analisis kelemahan

Analisis kelemahan sistem ini dilakukan dengan menganalisis permasalahan yang terjadi berdasarkan hasil penggambaran alur sistem pengolahan data inventaris kantor yang berjalan pada PT Sriwidjaja. dimana saat file atau data inventaris hilang/rusak tidak ada backup data serta proses pencarian data inventaris membutuhkan waktu yang cukup lama.

3.3.2 Analisis usulan sistem

Setelah menemukan permasalahan yang terjadi, maka pada tahapan analisis usulan sistem ini akan di berikan usulan sistem yang baru guna mengatasi permasalahan yang telah ditemukan pada tahapan analisis kelemahan sistem. Pada tahapan analisis sistem ini.sistem yang diusulkan penulis yaitu dimana pengolahan data inventaris kantor disajikan menggunakan aplikasi pengolahan data inventaris untuk mengurangi waktu bagi bagian pengadaan melakukan proses pencarian data inventaris kantor. Adapun sistem informasi pengolahan data inventaris kantor yang diusulkan pada PT Sriwidjaja.

1. Ketika ada inventaris kantor rusak, bagian pengadaan mengecek inventaris.
2. Kemudian bagian pengadaan menginputkan data hasil pengecekan inventaris serta kondisi inventaris di aplikasi pengolahan data inventaris kantor.
3. Jika ada inventaris kantor yang baru, bagian pengadaan menginputkan data kedalam aplikasi pengolahan data inventaris kantor.
4. Jika ada inventaris kantor hilang, bagian pengadaan akan mengecek data inventaris, jika inventaris asuransi maka bagian pengadaan akan mengklaim inventaris kantor ke asuransi, tetapi jika inventaris tidak rusak maka personal yang menghilangkan barang akan mengganti inventaris tersebut melalui bagian pengadaan.

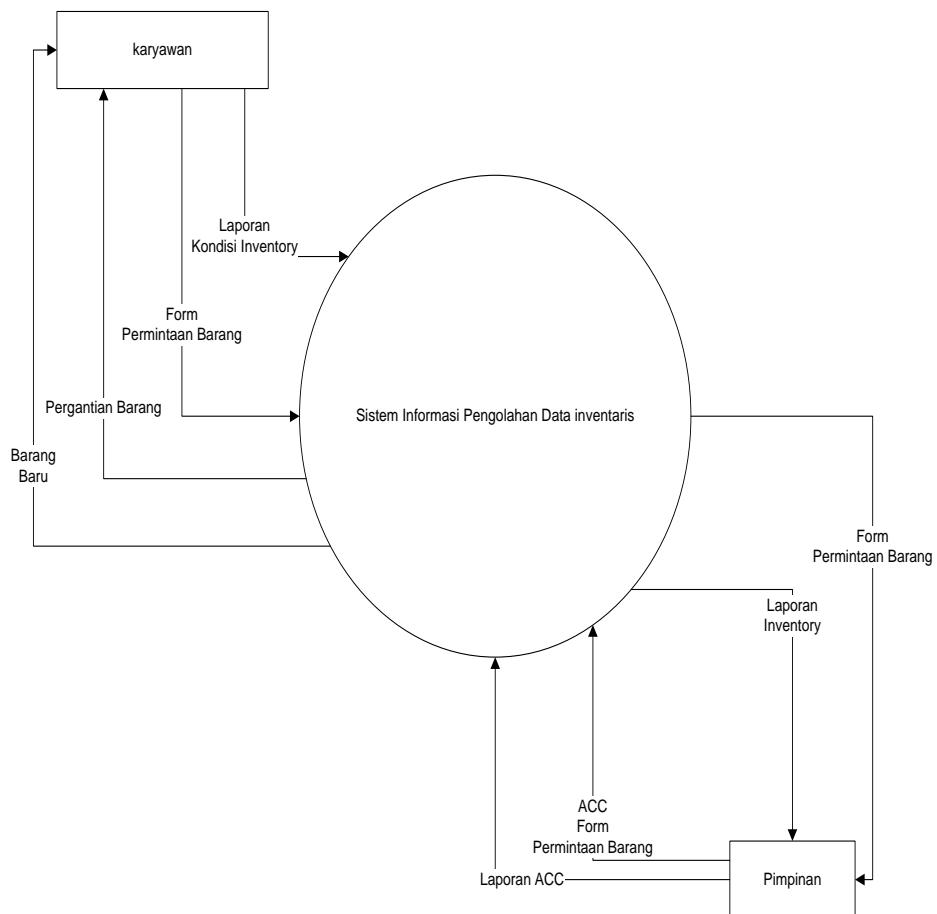
5. Bagian pengadaan akan membuat surat permintaan inventaris kemudian diserahkan kepada pimpinan
6. Pimpinan memvalidasi surat permintaan inventaris setelah itu dikembalikan kepada bagian pengadaan.
7. Kemudian setiap minggu bagian pengadaan mengupdate data barang inventaris untuk laporan diakhir bulan lalu diserahkan kepada pimpinan.

3.4 Diagram Alir

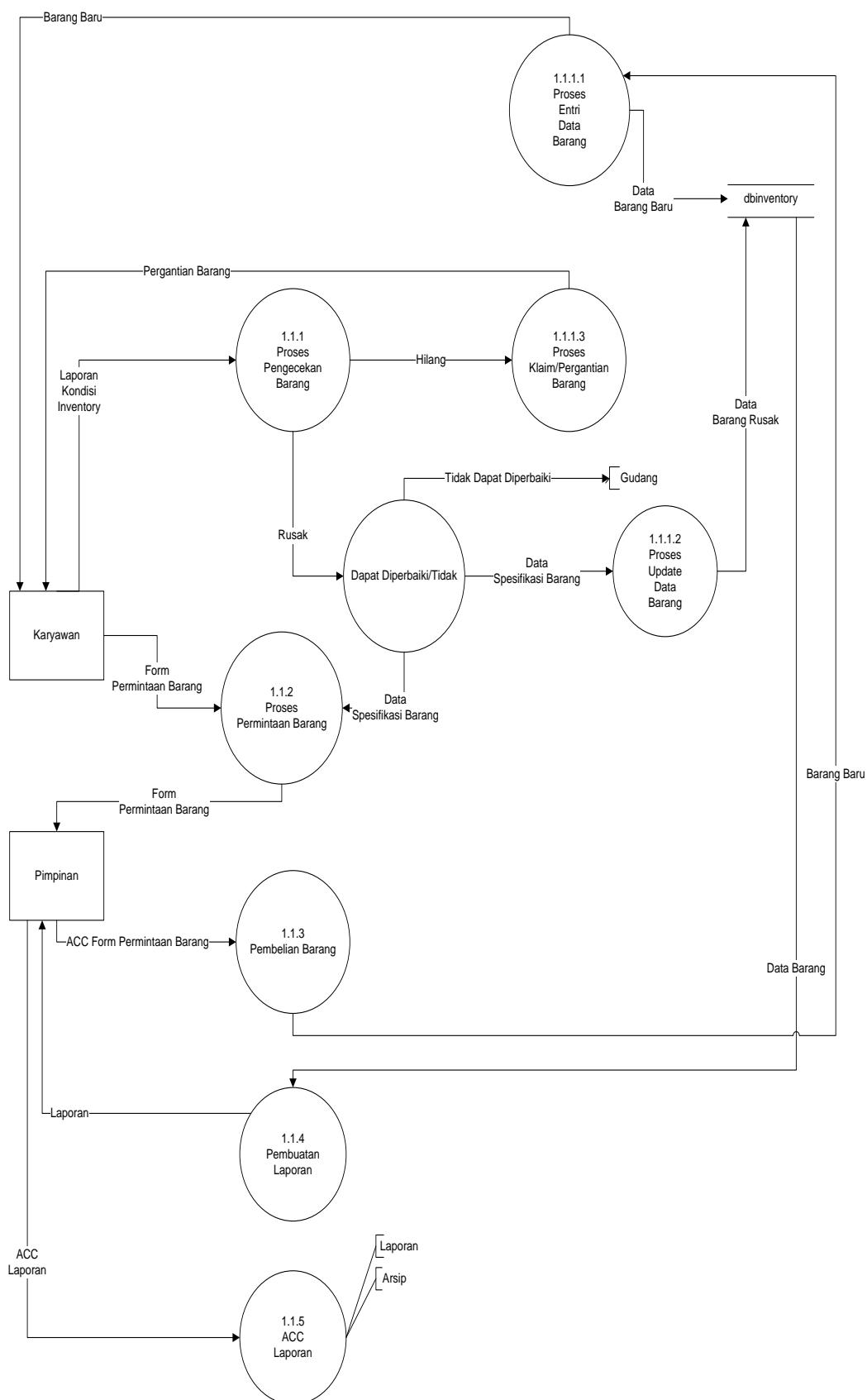
Rancangan sistem yang diusulkan meliputi beberapa komponen sistem berikut ini :

a) Model sistem

Berikut adalah model sistem yang diusulkan dalam bentuk *context diagram* dan DFD.



Gambar 3.3 Context diagram Sistem Diusulkan



Gambar 3.4 DFD level 0 Sistem Informasi Pengolahan Data Inventaris Kantor

3.5 Desain (perancangan) Sistem Secara Terperinci

Desain (perancangan) sistem secara terperinci ini dilakukan dengan menjelaskan rancangan-rancangan yang diperlukan untuk sistem yang baru secara terperinci .

3.5.1 Rancangan Form Login

Rancangan *form Login* ini dimana *user* dapat memilih berbagai pilihan yang telah disediakan untuk kemudahan dalam menjalankan sistem. Rancangan *form* ini terdiri dari *input User Name, password*:

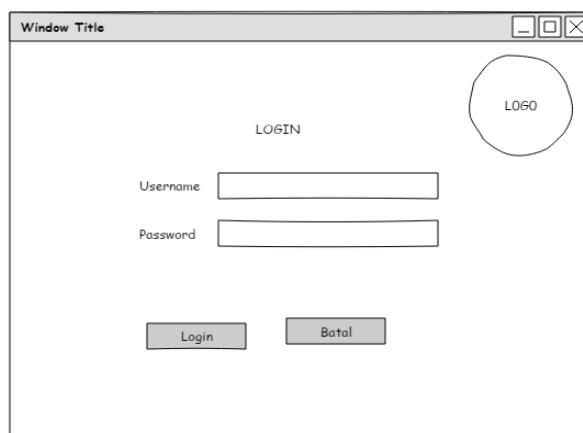


Diagram wireframe of a login window titled "Window Title". The window contains a central "LOGIN" label, a circular "LOGO" placeholder on the right, and two input fields labeled "Username" and "Password" below it. At the bottom are two buttons: "Login" and "Batal".

Gambar 3.5 Rancangan Form Login

3.5.2 Rancangan Form Menu Utama

Rancangan *form Menu Utama* ini adalah *form* yang pertama kali dijalankan oleh program setelah admin login

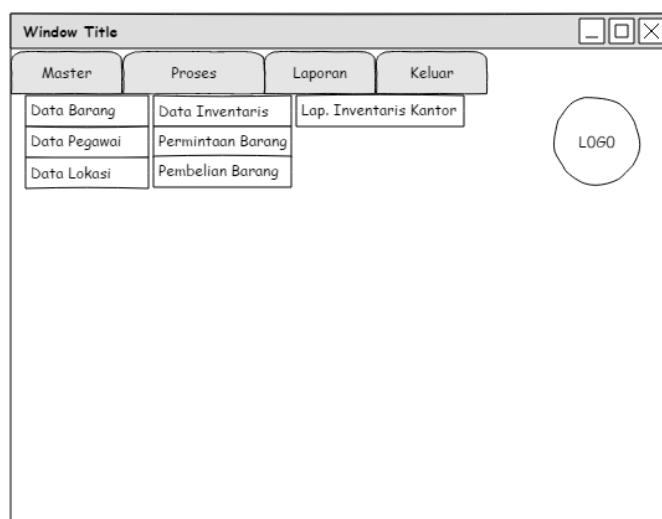


Diagram wireframe of a main menu window titled "Window Title". The window features a top navigation bar with tabs: "Master", "Proses", "Laporan", and "Keluar". Below this is a grid of six menu items: "Data Barang", "Data Inventaris", "Lap. Inventaris Kantor", "Data Pegawai", "Permintaan Barang", and "Pembelian Barang". A circular "LOGO" placeholder is located on the right side of the main area.

Gambar 3.6 Rancangan Form Menu Utama

3.5.3 Rancangan Output Sub Menu Laporan

Rancangan sub menu laporan dibuat untuk melihat laporan data barang, laporan mingguan dan laporan bulanan.

a. Rancangan Laporan Inventaris Kantor

Rancangan laporan inventaris kantor ini dimana *user* dapat melihat data barang.

LAPORAN INVENTARIS KANTOR						L060
Periode	s.d					
No	No Inventaris	Nama Inventaris	Jumlah	Lokasi	Kondisi	Penanggung Jawab

Mengetahui

Bandar Lampung, Tanggal
Pimpinan

Gambar 3.7 Rancangan Laporan Data Barang

b. Rancangan Laporan Pembelian Barang

Rancangan laporan pembelian barang ini dimana *user* dapat melihat data pembelian barang.

L060	Pupuk Sriwijaya Palembang Jl Dr. Sutomo No 57 Bandar Lampung						
LAPORAN PEMBELIAN BARANG							
Periode			s.d				
No	Kode Barang	Nama Barang	Tgl Pemintaan	Tgl Pembelian	Kondisi	Lokasi	Unit yang meminta

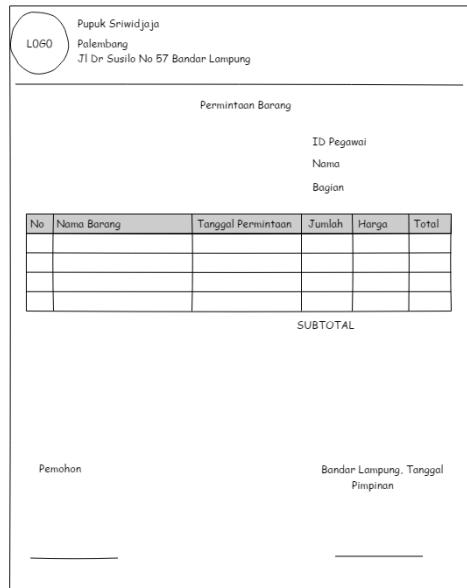
Mengetahui

Bandar Lampung, Tanggal
Pimpinan

Gambar 3.8 Rancangan Laporan Pembelian Barang

c. Rancangan Surat Permintaan Barang

Rancangan Permintaan Barang ini dimana *user* dapat mencetak permintaan barang.

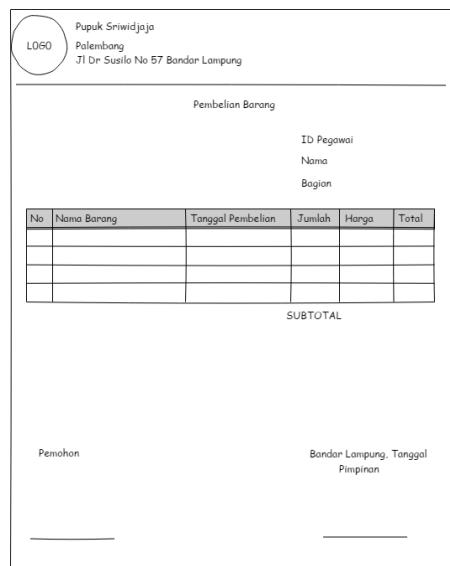


The form template for a request for goods (Permintaan Barang) is shown. It includes a header with the company logo (L060), name (Pupuk Sriwidjaja Palembang), address (Jl Dr Susilo No 57 Bandar Lampung), and a title 'Permintaan Barang'. Below this is a section for employee details: ID Pegawai, Name, and Position. A table for item requests is provided with columns for No, Name Barang, Tanggal Permintaan, Jumlah, Harga, and Total. A 'SUBTOTAL' row is at the bottom of the table. At the bottom, there are two signature lines for the requester (Penohon) and the supervisor (Pimpinan), both followed by the text 'Bandar Lampung, Tanggal'.

Gambar 3.9 Rancangan Permintaan Barang

d. Rancangan Surat Pembelin Barang

Rancangan Pembelian Barang ini dimana *user* dapat mencetak pembelian barang.



The form template for a purchase request (Pembelian Barang) is shown. It includes a header with the company logo (L060), name (Pupuk Sriwidjaja Palembang), address (Jl Dr Susilo No 57 Bandar Lampung), and a title 'Pembelian Barang'. Below this is a section for employee details: ID Pegawai, Name, and Position. A table for item purchases is provided with columns for No, Name Barang, Tanggal Pembelian, Jumlah, Harga, and Total. A 'SUBTOTAL' row is at the bottom of the table. At the bottom, there are two signature lines for the requester (Penohon) and the supervisor (Pimpinan), both followed by the text 'Bandar Lampung, Tanggal'.

Gambar 3.10 Rancangan Pembelian Barang

3.5.4 Rancangan Form-Form Input Data

Rancangan form input data dibuat untuk menginputkan data barang, data pegawai dan data lokasi.

3.5.4.1 Rancangan Form Sub Menu Master

a. Rancangan Form Input Data Barang

Rancangan *form* data barang ini dimana *user*dapat memasukan data barang.

The screenshot shows a window titled 'Window Title' with tabs for 'Master', 'Proses', 'Laporan', and 'Keluar'. A circular 'LOGO' icon is in the top right. The main area is labeled 'DATA BARANG' and contains four text input fields: 'Kode Barang', 'Nama Barang', 'Jenis', and 'Jumlah'. Below these are five buttons: 'Simpan', 'Edit', 'Hapus', 'Baru', and 'Keluar'. At the bottom is a table with columns 'No Inventaris', 'Nama', 'Jenis', and 'Jumlah', containing several rows of data.

Gambar 3.11 Rancangan Form Input Data Barang

b. Rancangan Form Input Data Pegawai

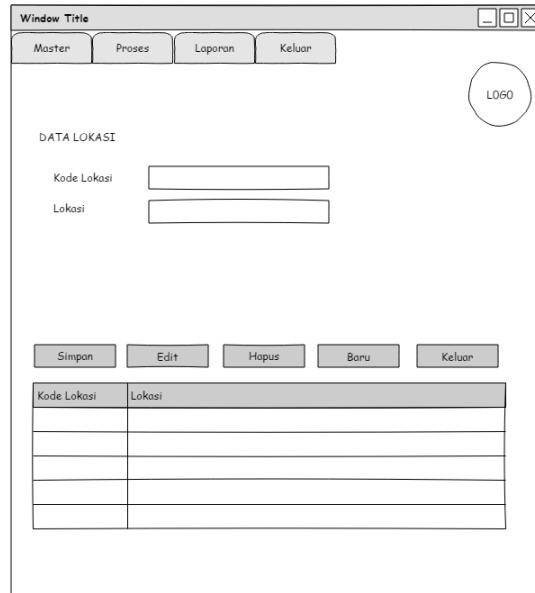
Rancangan *form* data Pegawai ini dimana *user*dapat menginputkan data pegawai.

The screenshot shows a window titled 'Window Title' with tabs for 'Master', 'Proses', 'Laporan', and 'Keluar'. A circular 'LOGO' icon is in the top right. The main area is labeled 'DATA PEGAWAI' and contains five text input fields: 'NIP', 'Nama', 'Alamat', 'Telpon', and 'Bagian'. Below these are five buttons: 'Simpan', 'Edit', 'Hapus', 'Baru', and 'Keluar'. At the bottom is a table with columns 'Nip', 'Nama', 'Alamat', 'Telpon', and 'Bagian', containing several rows of data.

Gambar 3.12 Rancangan Form Input Data Pegawai

c. Rancangan Form Input Data Lokasi

Rancangan *form* data lokasi ini dimana *user*dapat menginputkan data lokasi penyimpanan inventaris kantor.



The form window has a title bar 'Window Title' with buttons for 'Master', 'Proses' (highlighted), 'Laporan', and 'Keluar'. A circular logo 'LOGO' is in the top right. The main area is titled 'DATA LOKASI' and contains two input fields: 'Kode Lokasi' and 'Lokasi'. Below these are five buttons: 'Simpon', 'Edit', 'Hapus', 'Baru', and 'Keluar'. At the bottom is a table with columns 'Kode Lokasi' and 'Lokasi', containing six rows of data.

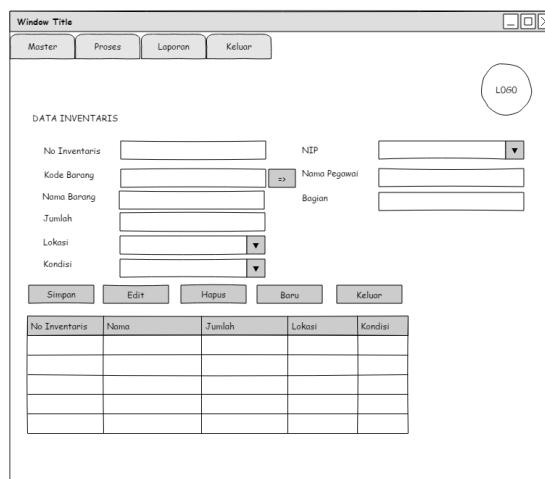
Gambar 3.13 Rancangan Form Input Data Lokasi

3.5.4.2 Rancangan Form SubMenu Proses

Rancangan sub menu master dibuat untuk menginputkan data barang, data pegawai dan data lokasi.

a. Rancangan Form Input Data Inventaris

Rancangan *form* data inventaris ini dimana *user*dapat memasukan data inventaris.

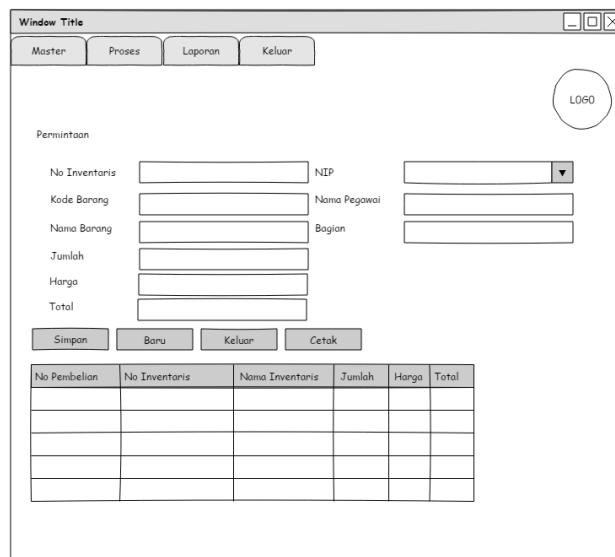


The form window has a title bar 'Window Title' with buttons for 'Master', 'Proses' (highlighted), 'Laporan', and 'Keluar'. A circular logo 'LOGO' is in the top right. The main area is titled 'DATA INVENTARIS' and contains several input fields: 'No Inventaris', 'NIP', 'Kode Barang', 'Nama Pegawai', 'Nama Barang', 'Jumlah', 'Bahan', 'Lokasi', and 'Kondisi'. Below these are five buttons: 'Simpon', 'Edit', 'Hapus', 'Baru', and 'Keluar'. At the bottom is a table with columns 'No Inventaris', 'Nama', 'Jumlah', 'Lokasi', and 'Kondisi', containing six rows of data.

Gambar 3.14 Rancangan Form Input Data Inventaris

b. Rancangan Form Input Permintaan Barang

Rancangan *form* data permintaan barang ini dimana *user*dapat menginputkan data permintaan barang.

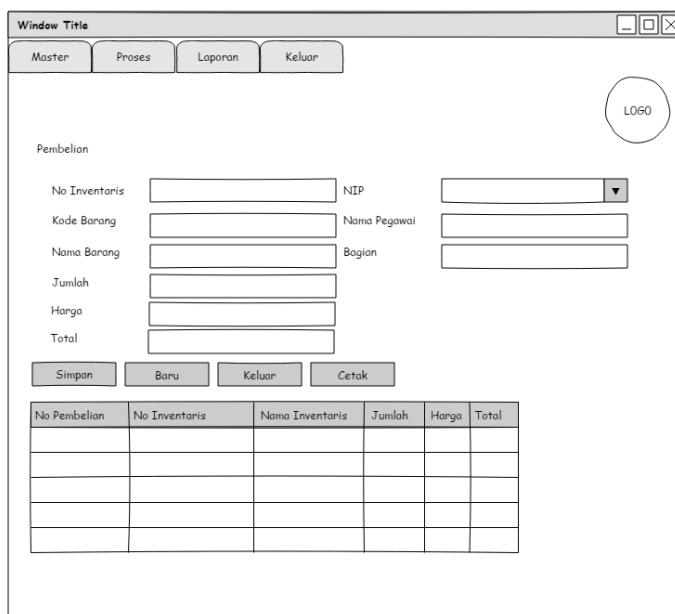


The screenshot shows a Windows-style application window titled 'Window Title'. At the top, there are four tabs: 'Master', 'Proses' (selected), 'Laporan', and 'Keluar'. In the top right corner is a circular 'LOGO' icon. The main area is labeled 'Permintaan' and contains several input fields: 'No Inventaris' (text box), 'NIP' (text box with dropdown arrow), 'Kode Barang' (text box), 'Nama Pegawai' (text box with dropdown arrow), 'Nama Barang' (text box), 'Bogian' (text box), 'Jumlah' (text box), 'Harga' (text box), and 'Total' (text box). Below these fields are four buttons: 'Simpan', 'Baru', 'Keluar', and 'Cetak'. Underneath the buttons is a table with columns: 'No Pembelian', 'No Inventaris', 'Nama Inventaris', 'Jumlah', 'Harga', and 'Total'. The table has six rows, all of which are currently empty.

Gambar 3.15 Rancangan Form Input Permintaan Barang

c. Rancangan Form Input Pembelian Barang

Rancangan *form* data pembelian barang ini dimana *user*dapat menginputkan data pembelian barang.



The screenshot shows a Windows-style application window titled 'Window Title'. At the top, there are four tabs: 'Master', 'Proses' (selected), 'Laporan', and 'Keluar'. In the top right corner is a circular 'LOGO' icon. The main area is labeled 'Pembelian' and contains several input fields: 'No Inventaris' (text box), 'NIP' (text box with dropdown arrow), 'Kode Barang' (text box), 'Nama Pegawai' (text box), 'Nama Barang' (text box), 'Bogian' (text box), 'Jumlah' (text box), 'Harga' (text box), and 'Total' (text box). Below these fields are four buttons: 'Simpan', 'Baru', 'Keluar', and 'Cetak'. Underneath the buttons is a table with columns: 'No Pembelian', 'No Inventaris', 'Nama Inventaris', 'Jumlah', 'Harga', and 'Total'. The table has six rows, all of which are currently empty.

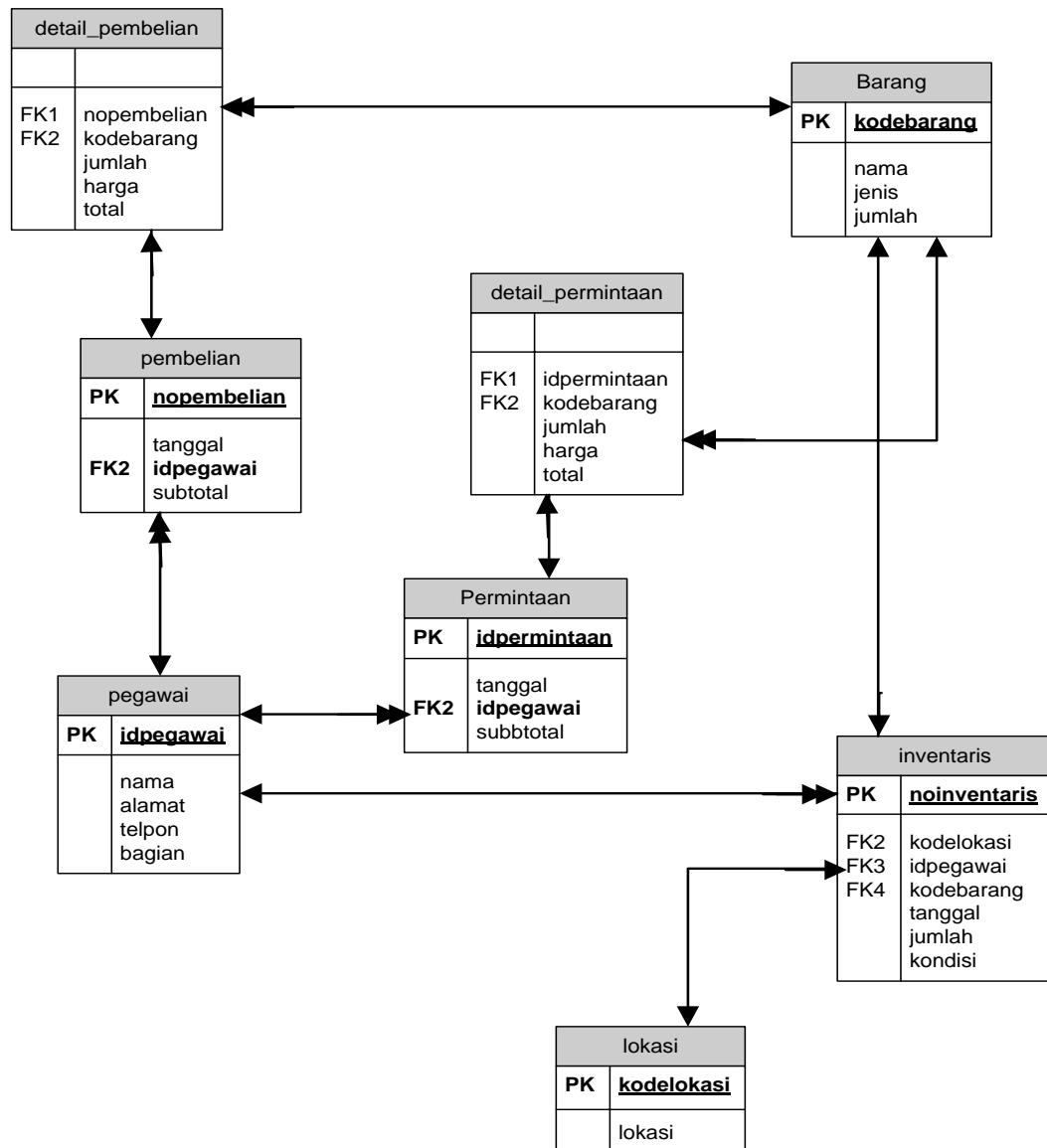
Gambar 3.16 Rancangan Form Input Pembelian Barang

3.5.5 Rancangan Database

Rancangan database ini berisikan rancangan file-file atau atribut-atribut yang dibutuhkan untuk membangun program baru dan masing-masing atribut dalam table tersebut akan berelasi antara tabel yang satu dengan yang lain. Rancangan database ini akan dibangun dengan langkah yaitu :

a. Relasi antar tabel

Relasi antar table ini menggambarkan tentang relasi dari masing-masing tabel. Adapun relasi antar tabel sistem informasi pengolahan data inventaris pada PT Sriwidjaja.



Gambar 3.17 Relasi Tabel

b. Kamus data

Kamus data ini berisikan tentang rincian masing-masing field dari setiap tabel yang dihasilkan.

1. Tabel Login

Nama *Database* : dbinventori

Nama Tabel : login

Primary key :username

Atribut : (username, password, akses).

Tabel 3.1 Tabel login

No	Nama <i>Field</i>	Tipe Data	Size	Keterangan
1	Username	Varchar	18	Nama User
2	Password	Text		Keamanan
3	Akses	Int	2	Hak Akses

2. Tabel Barang

Nama *Database* : dbinventori

Nama Tabel : barang

Primary key : kodebarang

Atribut : (kodebarang, nama, jenis, jumlah).

Tabel 3.2 Tabel Barang

No	Nama <i>Field</i>	Tipe Data	Size	Keterangan
1	Kodebarang	Varchar	15	Kode Barang
2	Nama	Varchar	25	Nama Inventaris
3	Jenis	Varchar	20	Jenis inventaris
4	Jumlah	Int	10	Jumlah

3. Tabel Pegawai

Nama *Database* : dbinventori

Nama Tabel : pegawai

Primary key : idpegawai

Atribut : (idpegawai, nama, alamat, telpon, bagian).

Tabel 3.3 Tabel Pegawai

No	Nama Field	Tipe Data	Size	Keterangan
1	Idpegawai	Varchar	6	ID Pegawai
2	Nama	Varchar	25	Nama Pegawai
3	Alamat	Text		Alamat Pegawai
4	Telpon	Varchar	13	Telpon
5	Bagian	Varchar	25	Nama Bagian

4. Tabel Lokasi

Nama Database : dbinventori

Nama Tabel : lokasi

Primary key : kodelokasi

Atribut : (kodelokasi, lokasi).

Tabel 3.4 TabelLokasi

No	Nama Field	Tipe Data	Size	Keterangan
1	Kodelokasi	Varchar	15	Kode Lokasi
2	Lokasi	Varchar	25	Lokasi Inventaris

5. Tabel Inventaris

Nama Database : dbinventori

Nama Tabel : inventaris

Primary key : noinventaris

Foreign key : kodelokasi, idpegawai

Atribut : (noinventaris, tanggal, kodebarang, kodelokasi, idpegawai, jumlah, kondisi).

Tabel 3.5 Tabel Inventaris

No	Nama Field	Tipe Data	Size	Keterangan
1	Noinventaris	Varchar	15	Nomor Inventaris
2	Tanggal	Date		Tanggal Penyimpanan
3	Kodebarang	Varchar	15	Kode Barang
4	Kodelokasi	Varchar	15	Kode Lokasi
5	Idpegawai	Varchar	6	ID Pegawai
6	Jumlah	Int	10	Jumlah
7	Kondisi	Varchar	15	Kondisi

6. Tabel Permintaan

Nama Database : dbinventori

Nama Tabel : permintaan

Primary key : idpermintaan

Foreign key : noinventaris, idpegawai

Atribut : (idpermintaan, tanggal, idpegawai, subtotal).

Tabel 3.6 Tabel Permintaan

No	Nama Field	Tipe Data	Size	Keterangan
1	Idpermintaan	Varchar	10	ID Permintaan
2	Tanggal	Date		Tanggal permintaan
3	Idpegawai	Varchar	12	ID Pegawai
4	Subtotal	Int	20	Sub Total

7. Tabel Detail Permintaan

Nama Database : dbinventori

Nama Tabel : detail_permintaan

Primary key : -

Foreign key : idpermintaan

Atribut : (idpermintaan, kodebarang, harga, jumlah, total).

Tabel 3.7 Tabel Detatal Permintaan

No	Nama Field	Tipe Data	Size	Keterangan
1	Idpermintaan	Varchar	10	ID Permintaan
2	Kodebarang	Varchar	15	Kode Barang
3	Harga	Int	20	Harga Barag
4	Jumlah	Int	5	Jumlah Barang
5	Total	Int	20	Total harga

8. Tabel Pembelian

Nama Database : dbinventori

Nama Tabel : pembelian

Primary key : nopembelian

Foreign key : noinventaris

Atribut : (nopembelian, tanggal, idpegawai,subtotal).

Tabel 3.8 Tabel Pembelian

No	Nama Field	Tipe Data	Size	Keterangan
1	Nopembelian	Varchar	10	Nomor Pembelian
2	Tanggal	Date		Tanggal Pembelian
3	Idpegawai	Varchar	12	Id Pegawai
4	Subtotal	Int	25	Subtotal

9. Tabel Detail Pembelian

Nama Database : dbinventori

Nama Tabel : detail_pembelian

Primary key : -

Foreign key : nopembelian, kodebarang

Atribut : (nopembelian, kodebarang, harga, jumlah, total).

Tabel 3.9 Tabel Detail Pembelian

No	Nama Field	Tipe Data	Size	Keterangan
1	Nopembelian	Varchar	10	Nomor Pembelian
2	Kodebarang	Varchar	15	Kode Barang
3	Harga	Int	20	Harga Barag
4	Jumlah	Int	5	Jumlah Barang
5	Total	Int	20	Total harga

c. Sistematika Pengkodean

1. Tipe kode Barang

Tipe kode Barang memakai tipe kode *urut*, terdapat 6 karakter, dengan tipe kode sebagai berikut:

Karakter 1 sampe 3 menunjukan nama barang, karakter 4-6 menunjukan nomor urut.

Contoh : MTR001

MTR = Motor

001 = menunjukkan nomor urut

2. Tipe kode pegawai

Tipe kode pegawai memakai tipe kode *urut*, terdapat 4 karakter, dengan tipe kode sebagai berikut:

Karakter 1 menunjukan jabatan, dan karakter 2-3 menunjukan nomor urut

Contoh : A001

A = Administrasi

001 = menunjukkan nomor urut

3. Tipe Nomor Pembelian

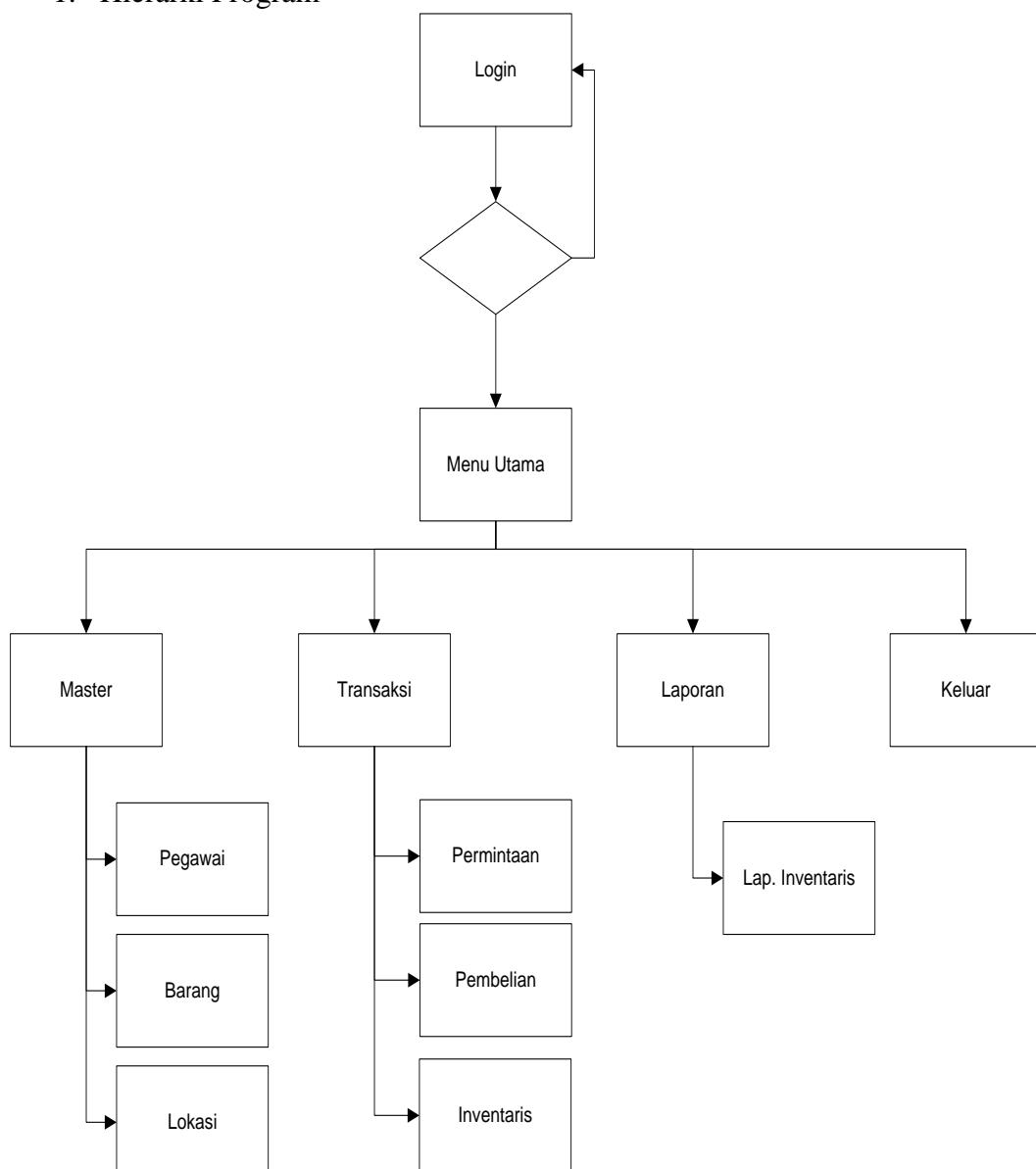
Tipe nomor pembelian memakai tipe kode urut.

4. Tipe Nomor Permintaan

Tipe nomor permintaan memakai tipe kode urut.

3.6. Rancangan Flowchart Program

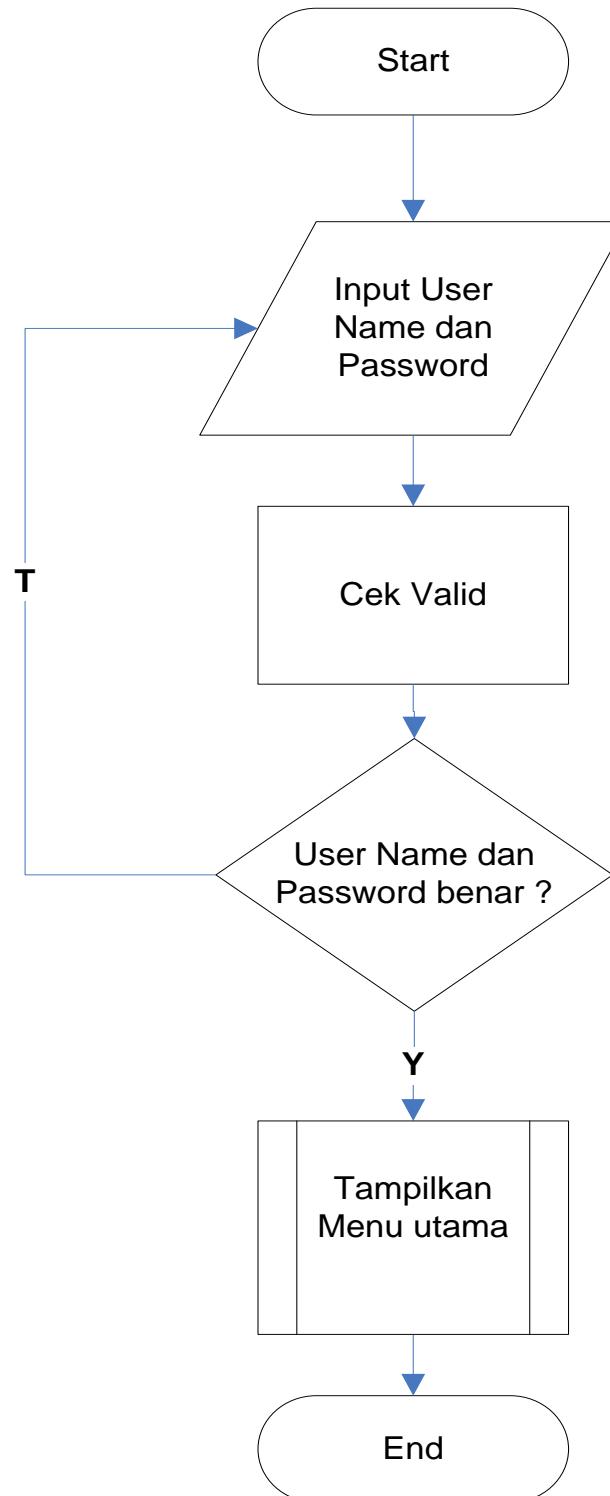
1. Hierarki Program



Gambar 3.18 struktur menu utama

2. Rancangan *flowchart* program *menu login*

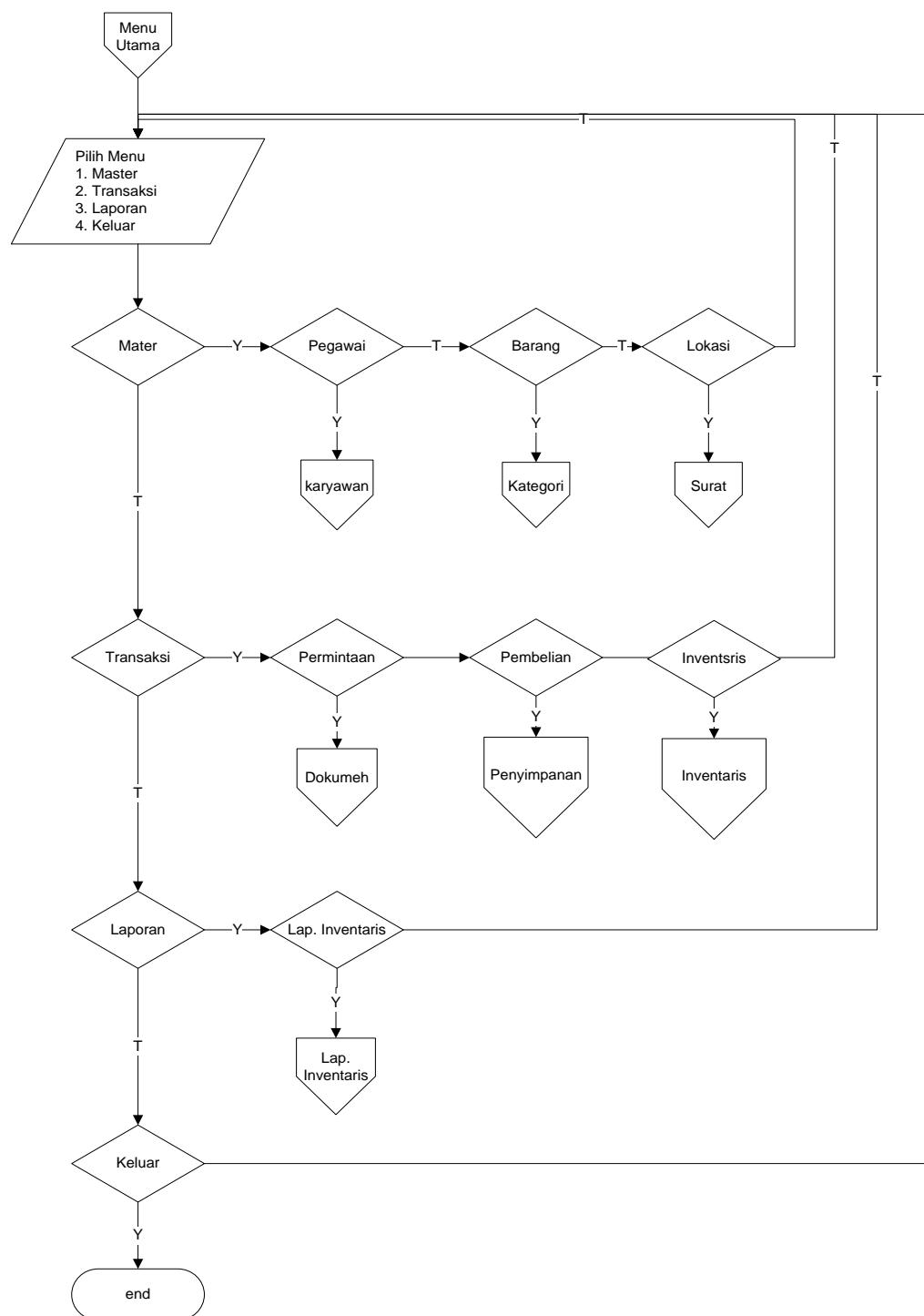
Flowchart login admin digunakan untuk menjelaskan alur *flowchart login* untuk masuk pada menu utama.



Gambar 3.19 rancangan *flowchart* program *menu Login*

3. Rancangan flowchart program menu utama

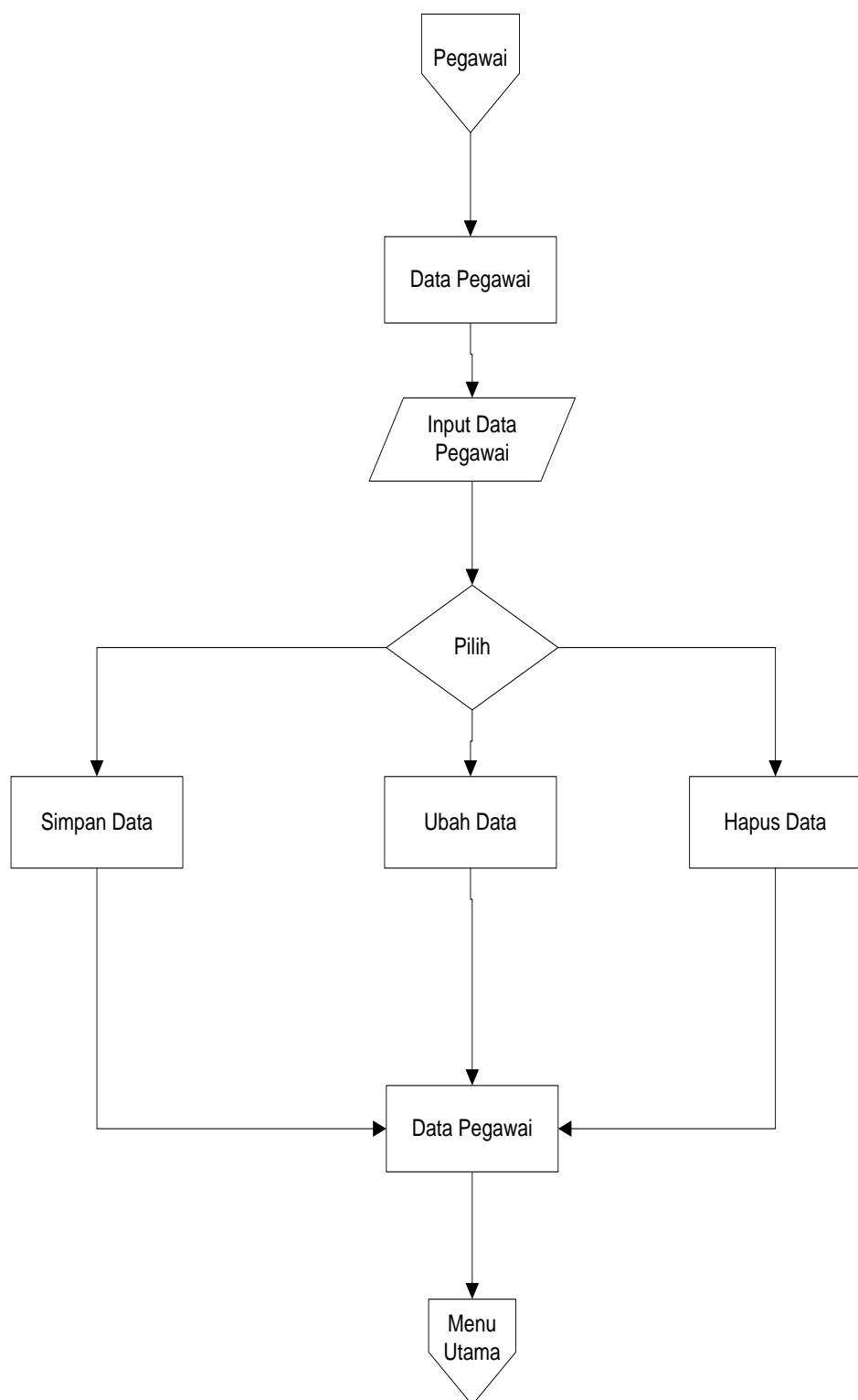
Flowchart menu utama admin digunakan untuk menjelaskan alur flowchart menu utama pada menu utama.



Gambar 3.20 rancangan flowchart program menu utama

4. Rancangan *flowchart* program *menu Pegawai*

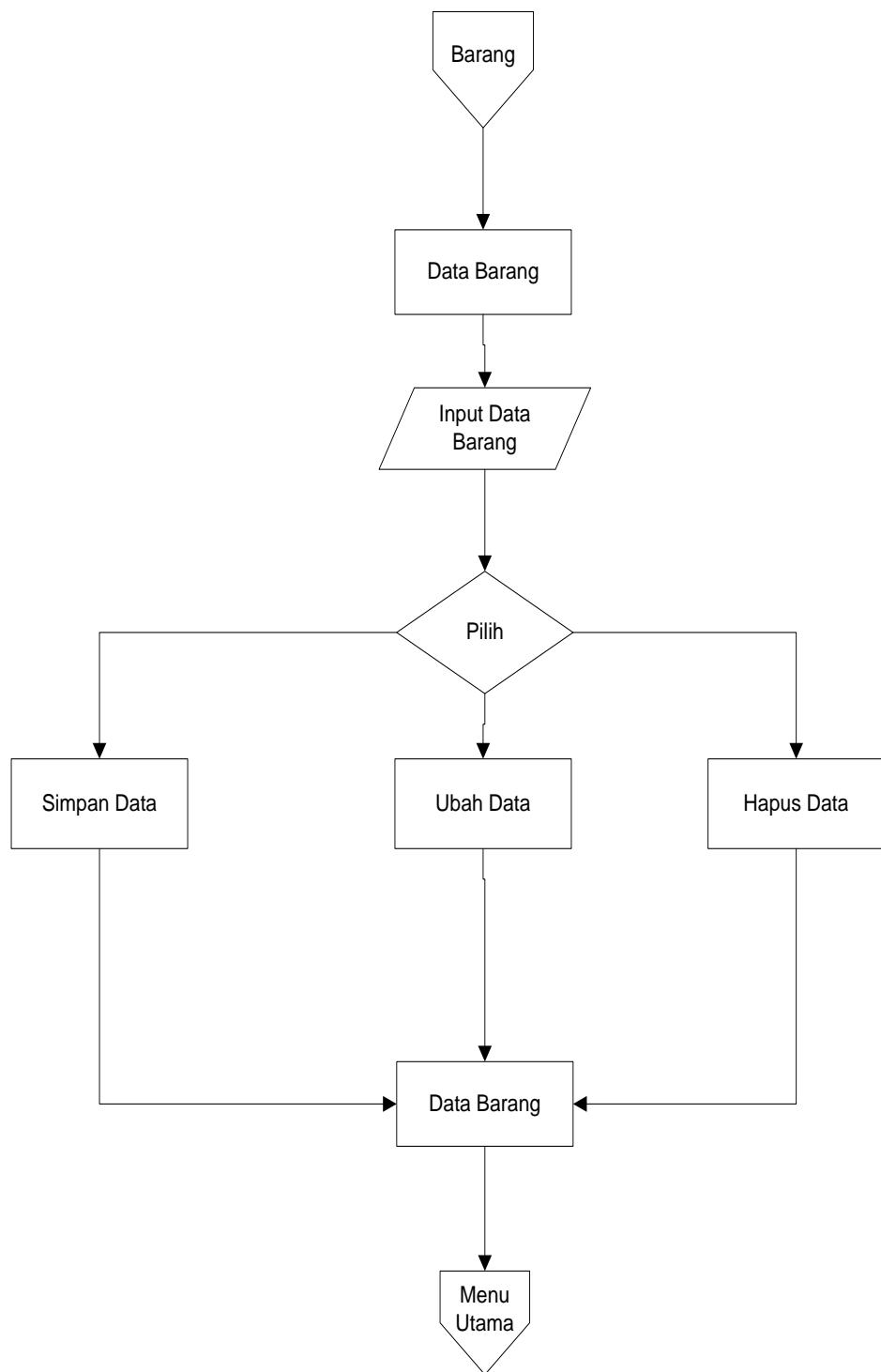
Flowchart menu pegawai admin digunakan untuk menjelaskan alur *flowchart menu pegawai* pada menu utama.



Gambar 3.21 rancangan *flowchart* program *menu Pegawai*

5. Rancangan *flowchart* program *menu* Barang

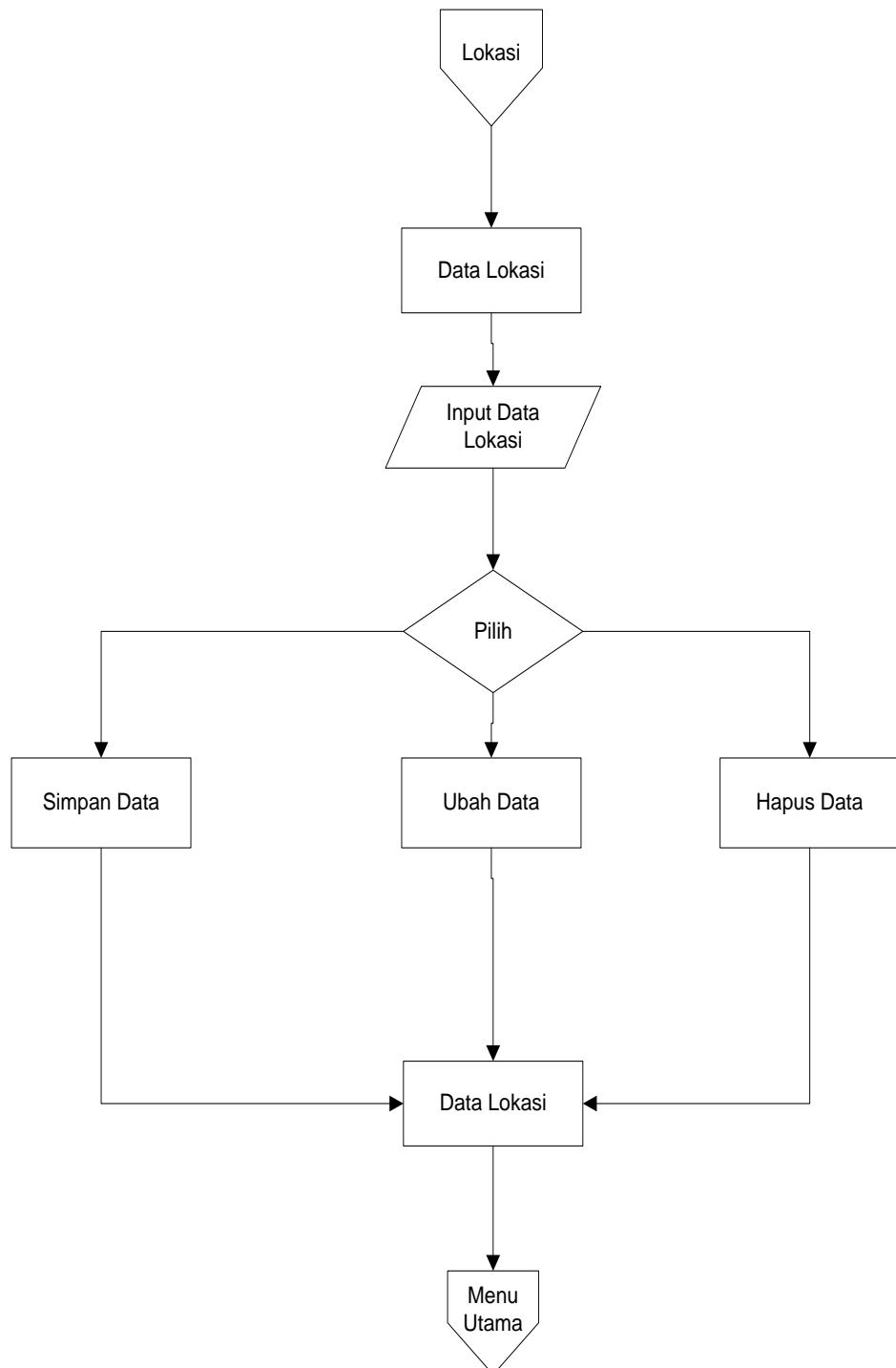
Flowchart *menu* barang admin digunakan untuk menjelaskan alur *flowchart* *menu* barang pada *menu* utama.



Gambar 3.22 rancangan *flowchart* program *menu* Barang

6. Rancangan *flowchart* program *menu Lokasi*

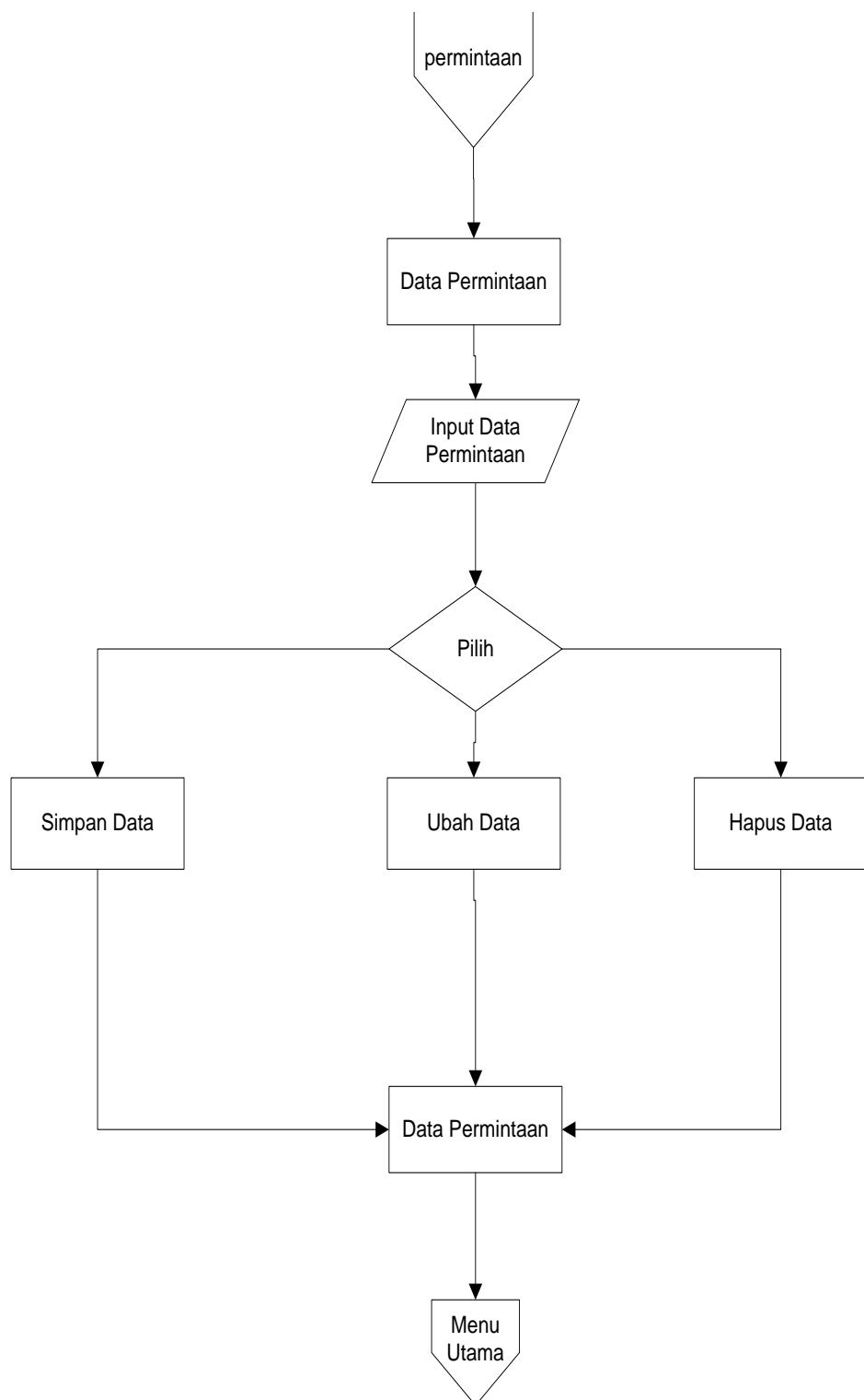
Flowchart menu lokasi admin digunakan untuk menjelaskan alur *flowchart menu* lokasi pada menu utama.



Gambar 3.23 rancangan *flowchart* program *menu Lokasi*

7. Rancangan *flowchart* program *menu* Permintaan

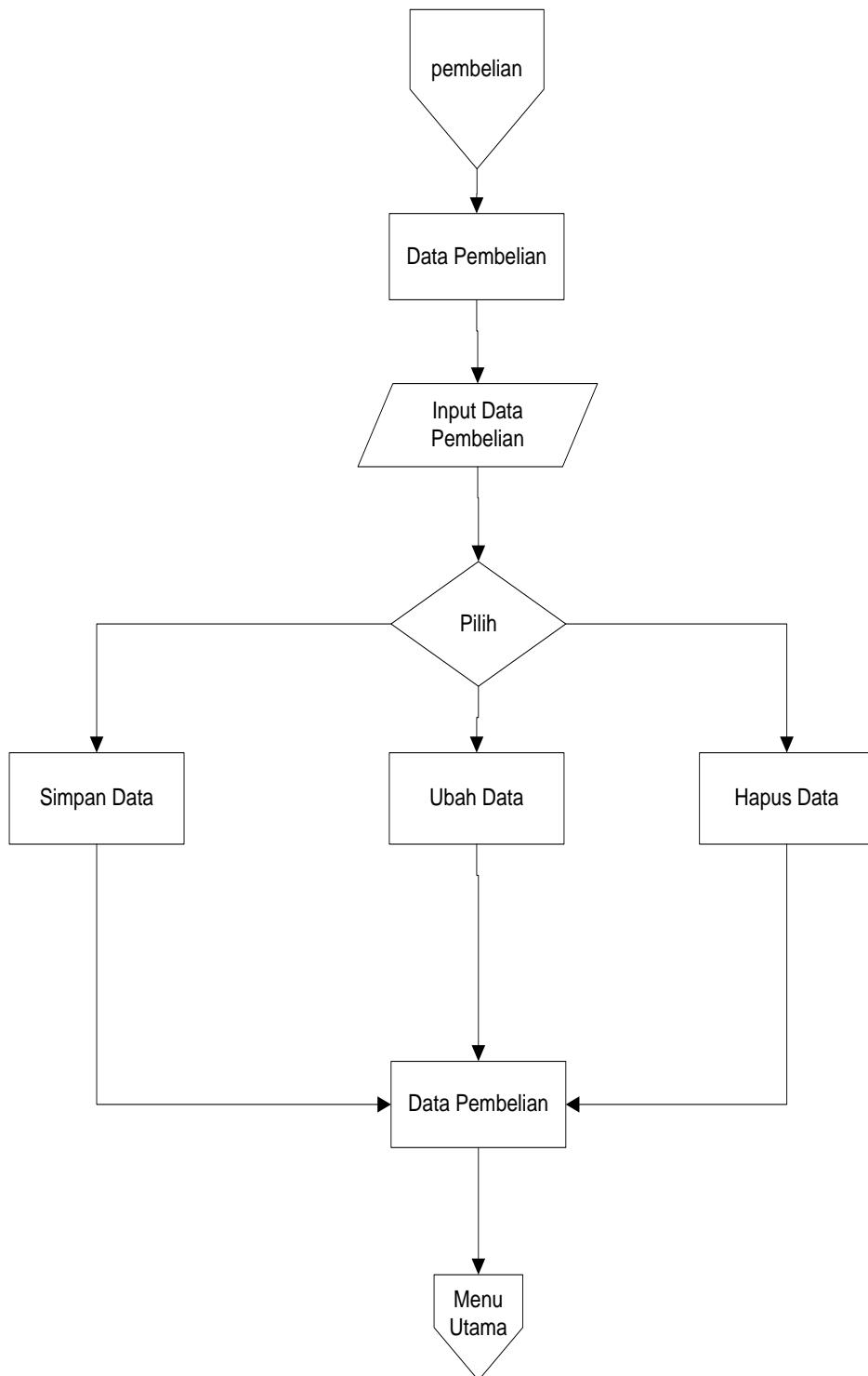
Flowchart menu permintaan admin digunakan untuk menjelaskan alur *flowchart menu* permintaan pada menu utama.



Gambar 3.24 rancangan *flowchart* program *menu* Permintaan

8. Rancangan *flowchart* program *menu* Pembelian

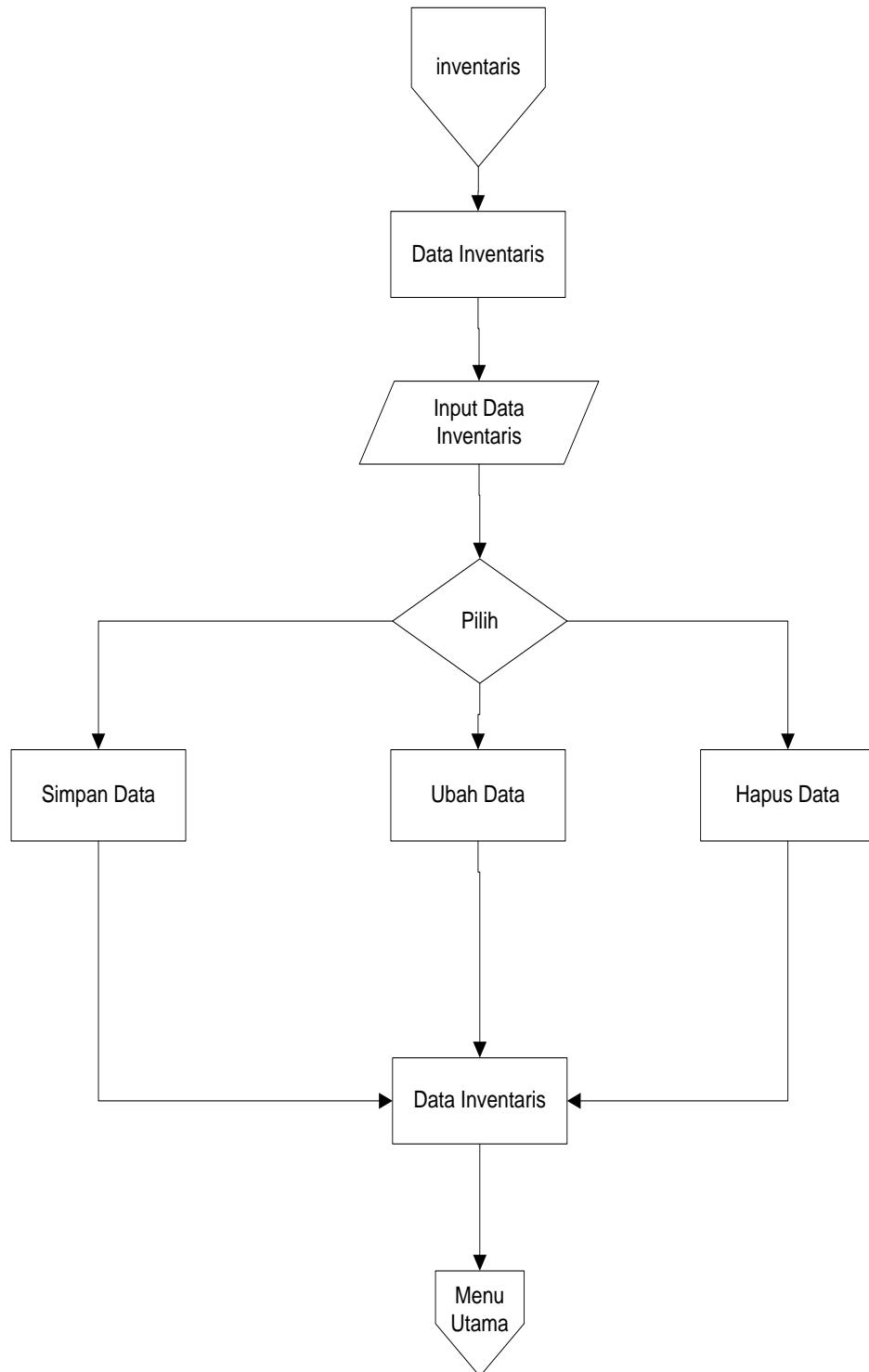
Flowchart menu pembelian admin digunakan untuk menjelaskan alur *flowchart menu pembelian* pada menu utama.



Gambar 3.25 rancangan *flowchart* program *menu* Pembelian

9. Rancangan *flowchart* program *menu Inventaris*

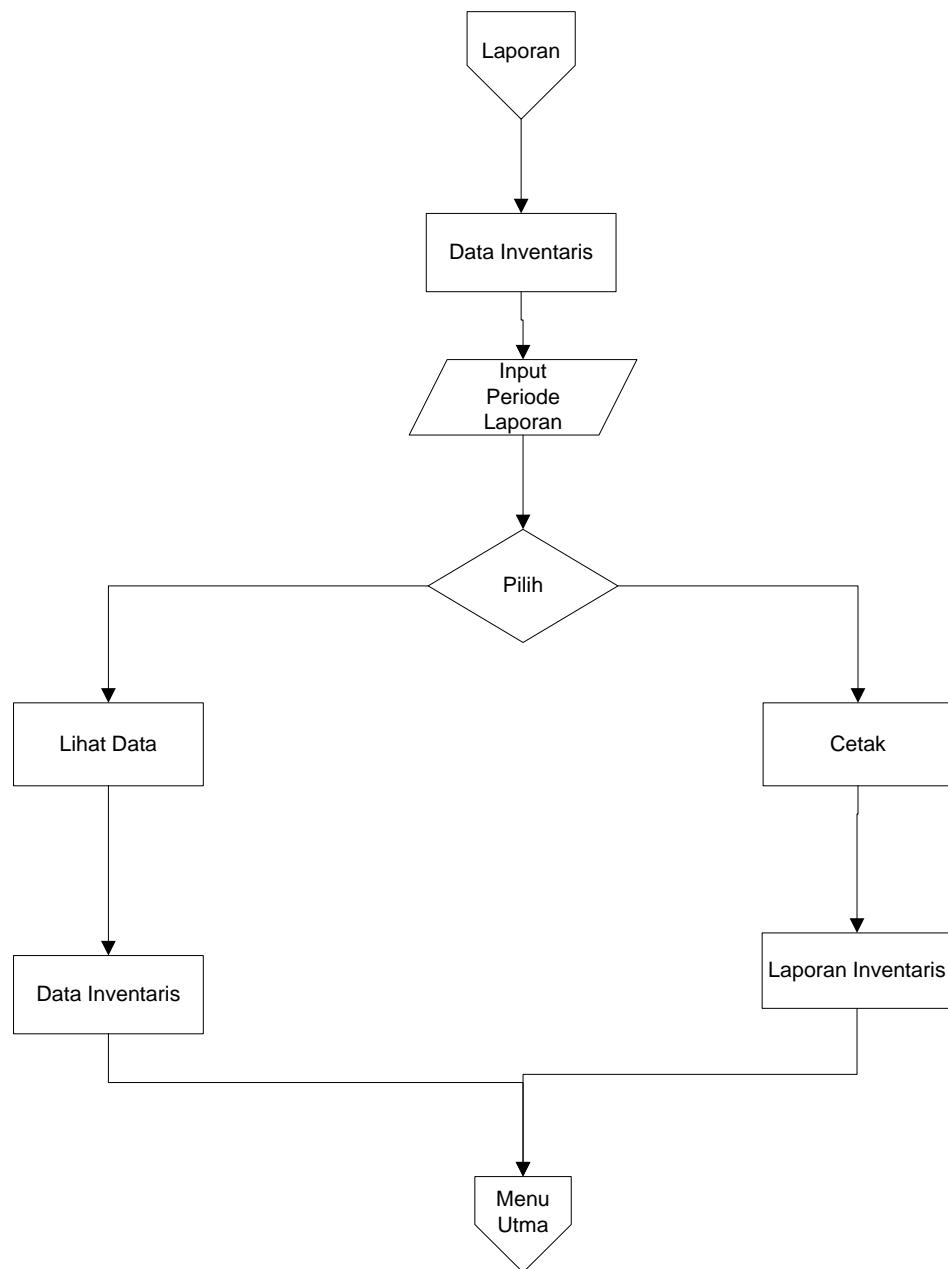
Flowchart menu inventaris admin digunakan untuk menjelaskan alur *flowchart menu inventaris* pada menu utama.



Gambar 3.26 rancangan *flowchart* program *menu Inventaris*

10. Rancangan *flowchart* program menu Laporan

Flowchart menu Laporan admin digunakan untuk menjelaskan alur *flowchart* menu Laporan pada menu utama.



Gambar 3.27 rancangan *flowchart* program menu Laporan

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Hasil Penelitian

4.1.1 Tampilan Program

Pada pembahasan ini menjelaskan mengenai isi dan fungsi dari tiap-tiap form pada aplikasi data pengolahan data inventaris dapat dilihat pada pembahasan dibawah ini.

- a) Form Login

Form *login* adalah form yang menampilkan username dan password untuk admin masuk kedalam sistem.



Gambar 4.1 Tampilan form *Home*

b) Form Menu Utama

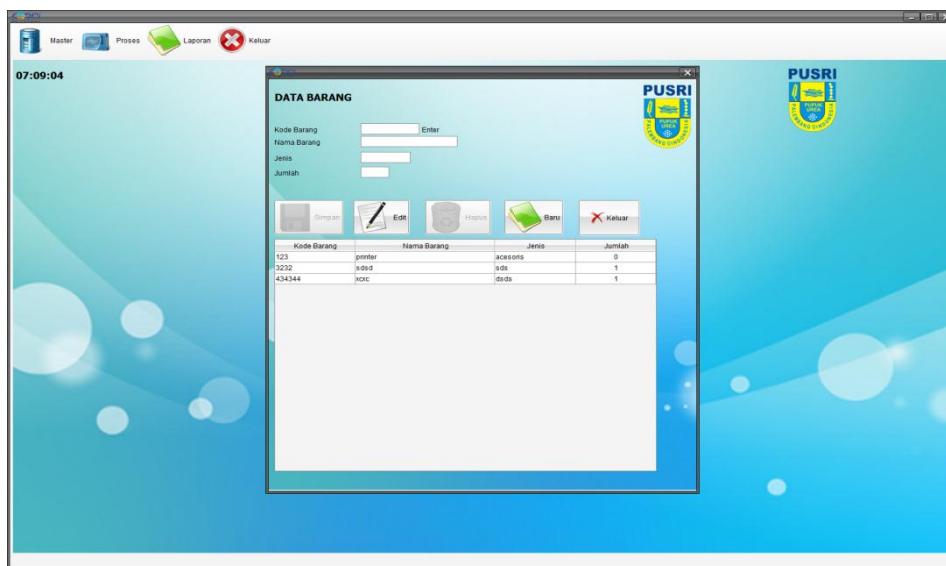
Form ini berisi menu-menu untuk memanggil form lain seperti form barang, pegawai, lokasi, inventaris, permintaan barang, pembelian barang dan laporan inventaris kantor.



Gambar 4.2 Tampilan Menu Utama

c) Form barang

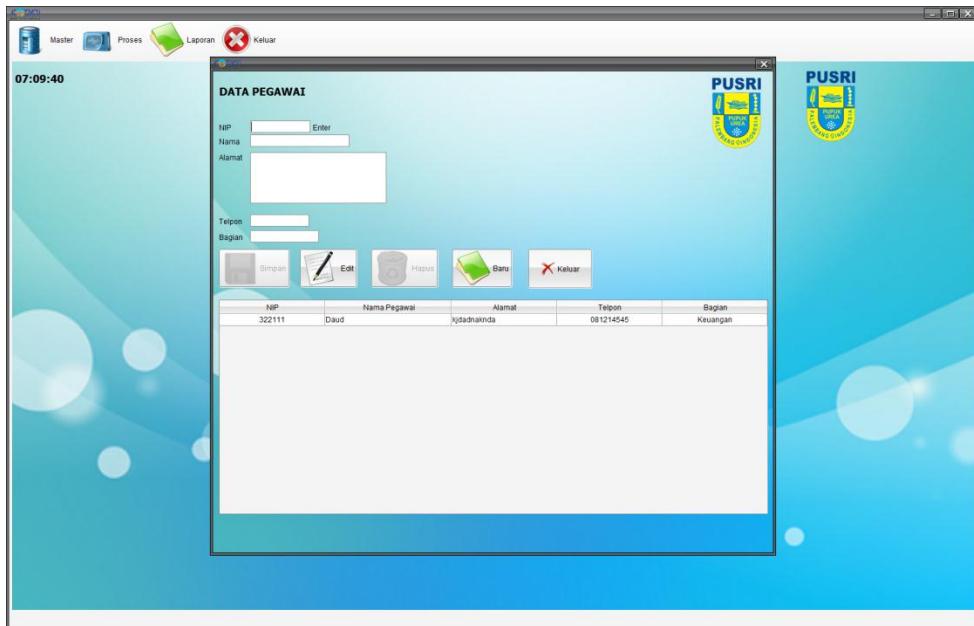
Form ini berfungsi untuk memasukan data barang.



Gambar 4.3 Tampilan Form Barang

d) Form Pegawai

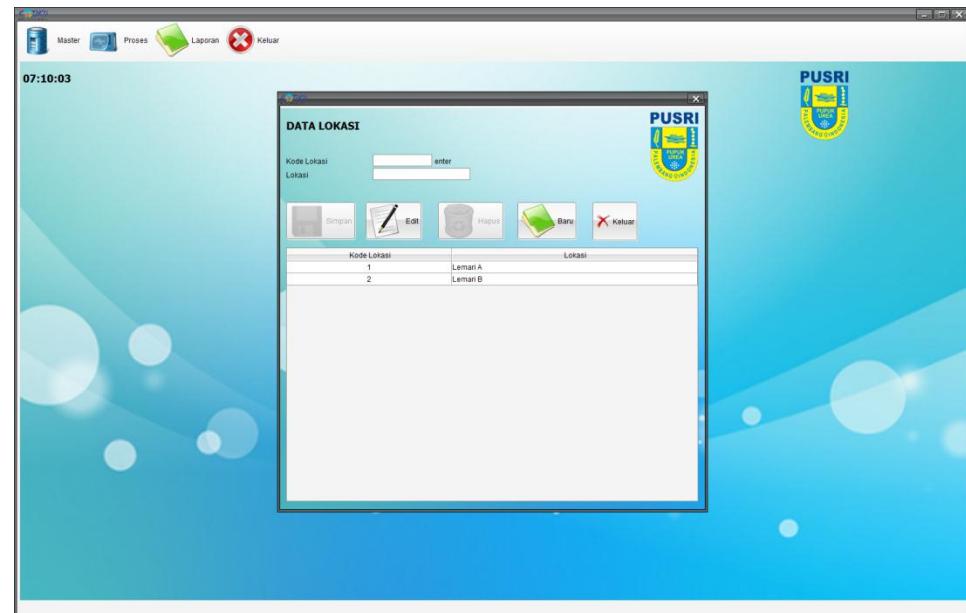
Form ini berfungsi untuk memasukan data pegawai.



Gambar 4.4 Tampilan form Pegawai

e) Form lokasi

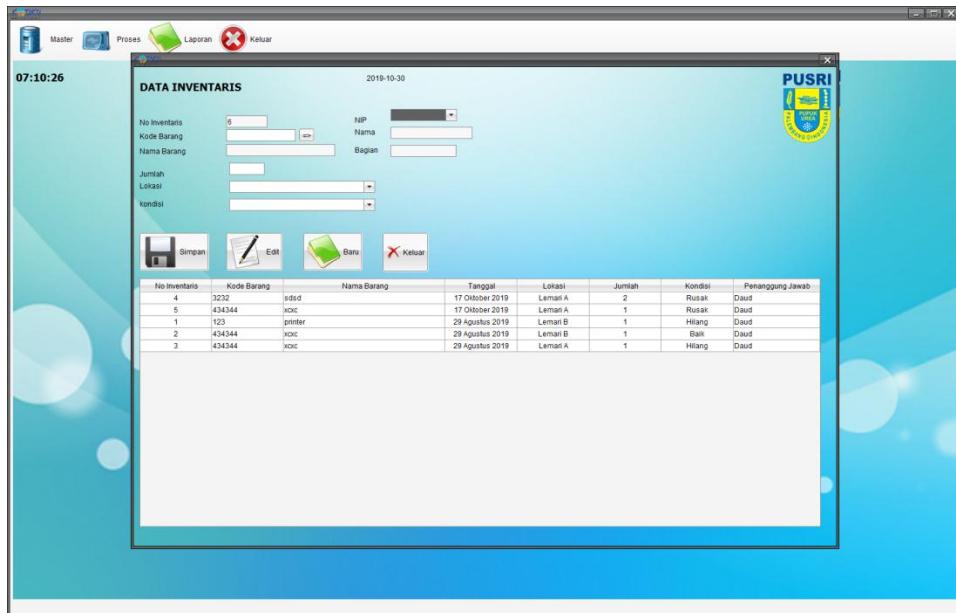
Form ini berfungsi untuk memasukan data lokasi.



Gambar 4.5 Tampilan form Lokasi

f) Form Inventaris

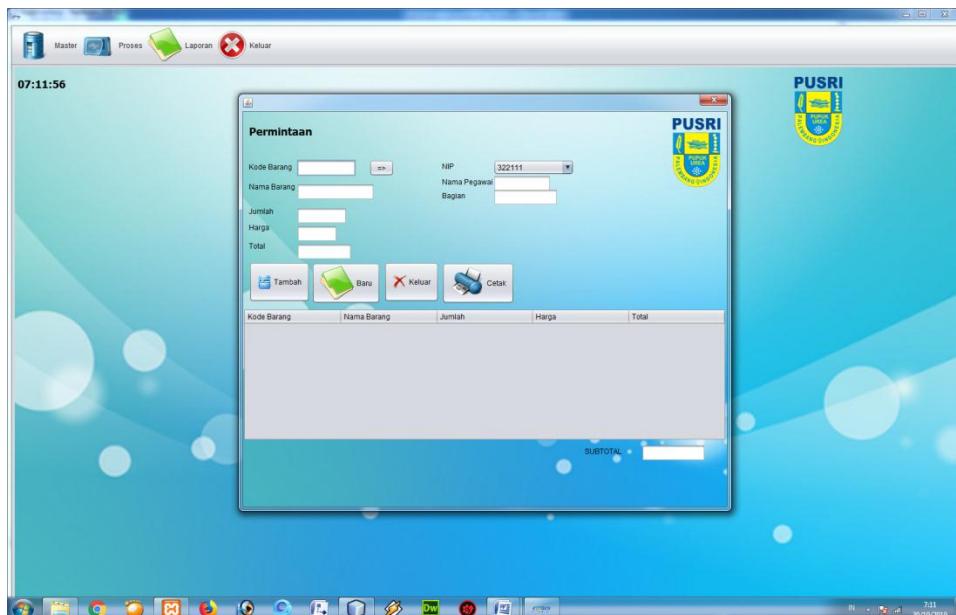
Form ini berfungsi untuk memasukan data inventaris.



Gambar 4.6 Tampilan form Inventaris

g) Form Permintaan Barang

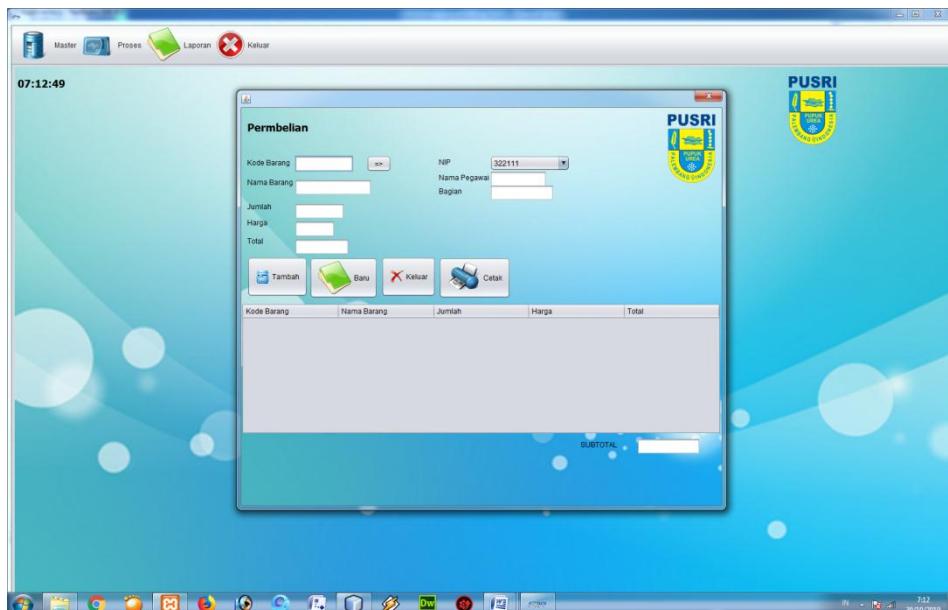
Form ini berfungsi untuk memasukan data permintaan barang.



Gambar 4.7 Tampilan form permintaan barang

h) Form Pembelian Barang

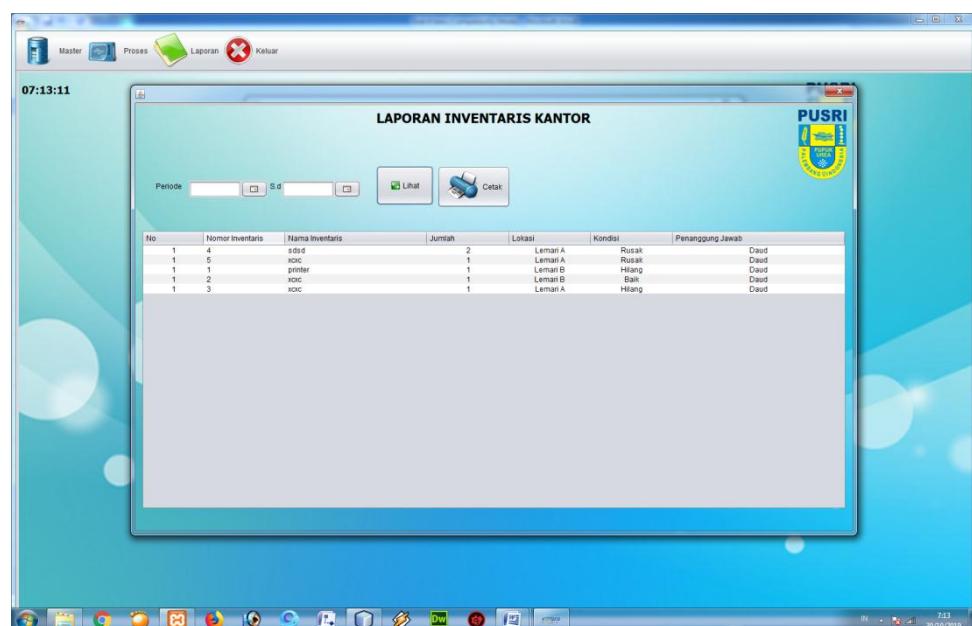
Form ini berfungsi untuk memasukan data pembelian barang.



Gambar 4.8 Tampilan form pembelian barang

i) Form Laporan Inventaris Kantor

Form ini berfungsi untuk melihat laporan inventaris kantor per periode.

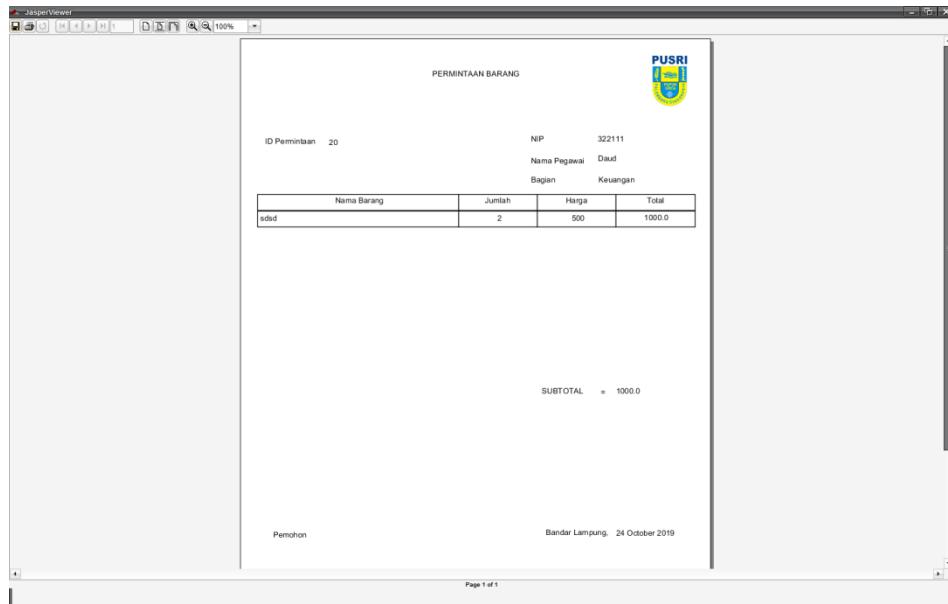


No	Nomor Inventaris	Nama Inventaris	Jumlah	Lokasi	Kondisi	Penanggung Jawab
1	4	xoc	2	Lemari A	Rusak	Daud
1	5	xoc	1	Lemari A	Rusak	Daud
1	1	printer	1	Lemari B	Hilang	Daud
1	2	xoc	1	Lemari B	Baik	Daud
1	3	xoc	1	Lemari A	Hilang	Daud

Gambar 4.9 Tampilan form Laporan Inventaris Kantor

j) Cetak Permintaan Barang

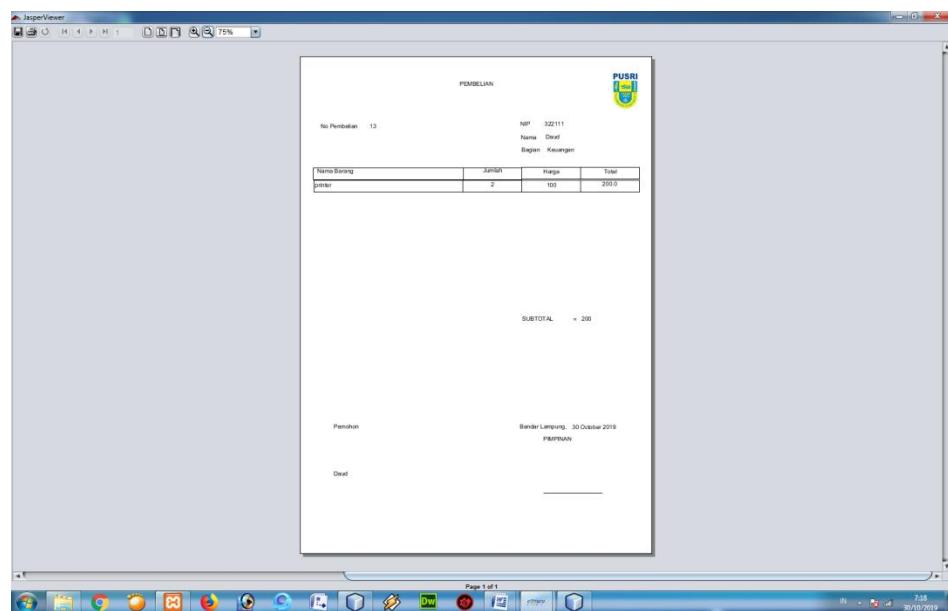
Cetak Permintaan Barang merupakan output dari form permintaan barang.



Gambar 4.10 Tampilan Cetak Permintaan Barang

k) Cetak Pembelian Barang

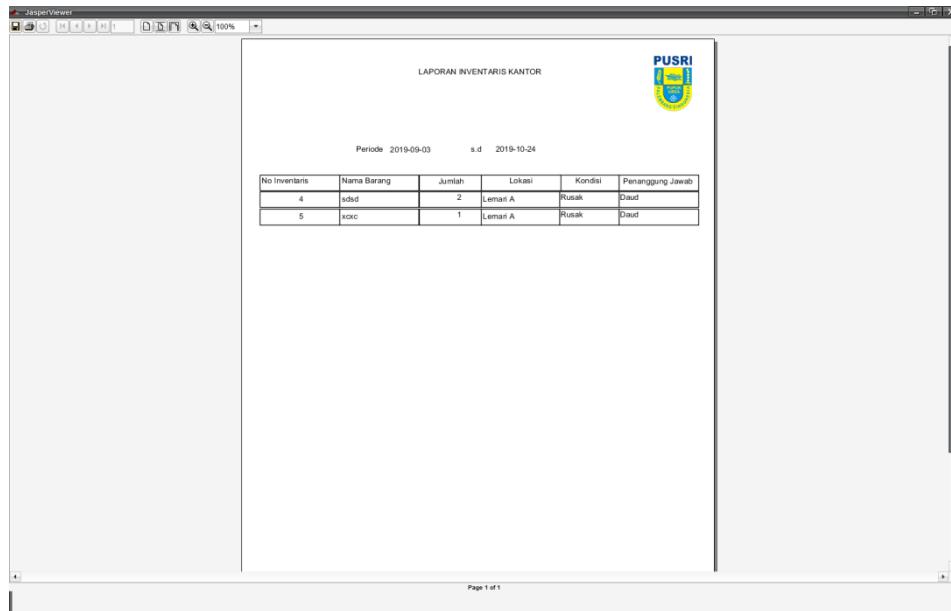
Cetak Pembelian Barang merupakan output dari form pembelian barang.



Gambar 4.11 Tampilan Cetak Pembelian Barang

1) Cetak Laporan Inventaris Kantor

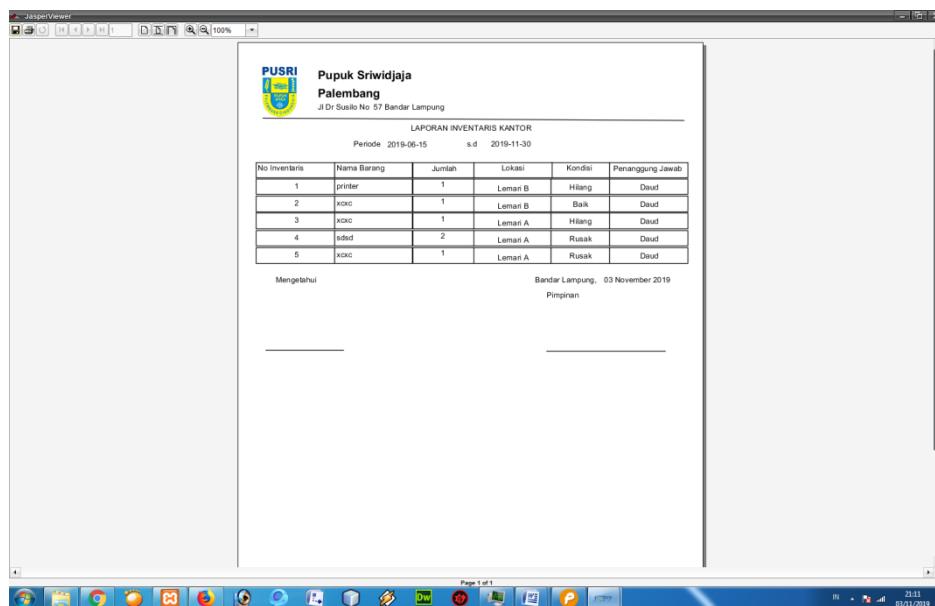
Cetak laporan inventaris kantor merupakan output dari form laporan inventaris kantor.



Gambar 4.12 Tampilan Cetak Laporan Inventaris Kantor

2) Cetak Laporan Pembelian Barang

Cetak laporan pembelian barang merupakan output dari form laporan pembelian barang.



Gambar 4.13 Tampilan Cetak Laporan Pembelian Barang

BAB V

SIMPULAN DAN SARAN

1.1 Simpulan

Berdasarkan hasil analisis dan pembahasan, kesimpulan dari penelitian ini sebagai berikut :

1. Pengolahan data inventaris kantor pada PT Sriwidjaja antara lain : membuat tabel, membuat form input, dan membuat laporan dengan memanfaatkan fasilitas yang ada.
2. Program aplikasi pengolahan data inventaris kantor pada PT Sriwidjaja dapat melakukan proses lebih cepat dan memudahkan perusahaan dalam mengontrol inventaris kantor.
3. Dengan menggunakan Data Base keamanan data dapat terjaga dengan aman dari kehilangan dan kerusakan data.

5.2 Saran

Berdasarkan kesimpulan di atas maka penulis menyarankan pada PT Sriwidjaja sebagai berikut :

1. *Upgrade* perangkat keras yang baru agar lebih optimal di dalam proses pendataan.
2. Aplikasi pengolahan data inventaris kantor ini hendaknya dapat dikembangkan kembali sesuai dengan kebutuhan PT Sriwidjaja.
3. Perlu ada pelatihan khusus untuk admin pengelola sistem.

DAFTAR PUSTAKA

Aldian Hidayat, Syafrika Deni Rizki, Dhio Saputra. 2016 Perancangan sistem informasi pengolahan nilai berbasiskan web pada sekolah menengah atas negeri 5 bukittinggi dengan menggunakan bahasa pemograan PHP. Jurnal Teknologi Vol. 6, No. 2, Hal.40-44

Syahrul Suci Romadhon, Desmulyati. 2019 Perancangan website sistem informasi simpan pinjam menggunakan framework codeiginter pada koperasi bumi sejahtera jakarta. Jurnal Teknik Informatika Vol.3, No.1, Hal.21-28

Ganda Yoga Swara, M.Kom, Yunes Pebriadi. 2016 Rekayasa perangkat lunak pemesanan tiket bioskop berbasis web. Jurnal Teknologi Informasi Vol. 4, No. 2, Hal.27-39

LAMPIRAN

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package appinventory;

import javax.swing.SwingUtilities;
import javax.swing.UIManager;

/**
 *
 * @author DoniKalong
 */
public class AppInventory {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        Splash form = new Splash();
        form.setLocationRelativeTo(null);
        try {

```

```
        UIManager.setLookAndFeel("com.jtattoo.plaf.acryl.AcrylLookAndFeel");
        SwingUtilities.updateComponentTreeUI(new MenuUtama());
    } catch (Exception e){

}

// Menampilkan Form
form.setVisible(true);

try {
    Thread.sleep(7000);
}

catch (InterruptedException e) {
}

form.dispose();

}

/*
* To change this license header, choose License Headers in Project Properties.
* To change this template file, choose Tools | Templates
* and open the template in the editor.
*/
package appinventory;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
```

```
import javax.swing.ImageIcon;
import javax.swing.Timer;
/***
 *
 * @author DoniKalong
 */
public class Splash extends javax.swing.JFrame {
    ActionListener waktu = new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            jProgressBar1.setValue(jProgressBar1.getValue()+1); //mengeset value
            dari progressbar
            if (jProgressBar1.getValue() == jProgressBar1.getMaximum()) //jika
            progressbar mencapai maksimum
            {
                jalan.stop();
                login lo = new login();
                lo.setLocationRelativeTo(null);
                lo.setVisible(true);
            }
        }
    };
    Timer jalan = new Timer(50, waktu);
    /**

```

```

* Creates new form Splash

*/
public Splash() {
    initComponents();
    jalan.start();
    this.dispose();
    icon();
}

public void icon(){
    ImageIcon ico = new
    ImageIcon("D:/Aplikasi/SysQuring/src/gambar/logo2.png");
    setIconImage(ico.getImage());
}

/**
 * This method is called from within the constructor to initialize the form.
 *
 * WARNING: Do NOT modify this code. The content of this method is
 * always
 *
 * regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jProgressBar1 = new javax.swing.JProgressBar();
    jLabel1 = new javax.swing.JLabel();
}

```

```
setDefaultCloseOperation(javax.swing.WindowConstants.DO_NOTHING_ON_CLOSE);
```

```
setUndecorated(true);
```

```
getContentPane().setLayout(new java.awt.CardLayout());
```

```
jPanel1.setBackground(new java.awt.Color(255, 255, 255));
```

```
jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/appinventory/th.jpg"))); // NOI18N
```

```
javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
```

```
jPanel1.setLayout(jPanel1Layout);
```

```
jPanel1Layout.setHorizontalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(jPanel1Layout.createSequentialGroup()
```

```
.addContainerGap()
```

```
.addComponent(jProgressBar1,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
.addContainerGap())
```

```
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())
```

```
.addContainerGap(56, Short.MAX_VALUE)
```

```
.addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 430,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addGap(52, 52, 52))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 116,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jProgressBar1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(0, 11, Short.MAX_VALUE))

);

getContentPane().add(jPanel1, "card4");

pack();

}// </editor-fold>

/**
 * @param args the command line arguments
 */

public static void main(String args[]) {
```

```
/* Set the Nimbus look and feel */

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">

/* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.

* For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

*/
try {
    for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(Splash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(Splash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(Splash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Splash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
}

//</editor-fold>

Splash form = new Splash();

// Menampilkan Form
form.setVisible(true);

}

// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JProgressBar jProgressBar1;
// End of variables declaration
}

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package appinventory;

import java.sql.DriverManager;
import java.sql.ResultSet;
```

```
import java.sql.Statement;
import javax.swing.ImageIcon;
import javax.swing.JOptionPane;
/***
 *
 * @author DoniKalong
 */
public class login extends javax.swing.JFrame {

    /**
     * Creates new form login
     */
    public login() {
        initComponents();
        icon();
        txtUser.setText("");
        txtPass.setText("");
        txtUser.requestFocus();
    }

    public void icon(){
        ImageIcon ico = new
        ImageIcon("D:/program/AppKotaku/src/appkotaku/logo.png");
        setIconImage(ico.getImage());
    }
    /**

```

```
* This method is called from within the constructor to initialize the form.

* WARNING: Do NOT modify this code. The content of this method is
always

* regenerated by the Form Editor.

*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    txtUser = new javax.swing.JTextField();
    txtPass = new javax.swing.JPasswordField();
    btnLogin = new javax.swing.JButton();
    btnBatal = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
    jLabel1.setText("LOGIN");

    jLabel2.setText("Username");

    jLabel3.setText("Password");
}
```

```
txtUser.setText("jTextField1");

txtPass.setText("jPasswordField1");

btnLogin.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
btnLogin.setText("Login");
btnLogin.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnLoginActionPerformed(evt);
    }
});

btnBatal.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
btnBatal.setText("Batal");
btnBatal.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnBatalActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 0, Short.MAX_VALUE)
);
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(40, 40, 40)
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(46, 46, 46)
                        .addComponent(btnLogin)
                    )
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(18, 18, 18)
                        .addComponent(btnBatal))
                )
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(18, 18, 18)
                        .addComponent(txtPass))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(18, 18, 18)
                        .addComponent(jLabel3)
                    )
                )
            )
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(18, 18, 18)
                .addComponent(jLabel2)
            )
        )
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(txtUser,
                javax.swing.GroupLayout.PREFERRED_SIZE, 128,
                javax.swing.GroupLayout.PREFERRED_SIZE))
    )
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(117, 117, 117)
    )
)
```

```
        .addComponent(jLabel1)))  
        .addContainerGap(106, Short.MAX_VALUE))  
    );  
  
    jPanel1Layout.setVerticalGroup(  
  
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
        .addGroup(jPanel1Layout.createSequentialGroup()  
            .addGap(35, 35, 35)  
            .addComponent(jLabel1)  
            .addGap(31, 31, 31)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
            .addComponent(jLabel2)  
            .addComponent(txtUser,  
javax.swing.GroupLayout.PREFERRED_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
            .addGap(18, 18, 18)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
            .addComponent(jLabel3)  
            .addComponent(txtPass,  
javax.swing.GroupLayout.PREFERRED_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
            .addGap(49, 49, 49)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE.BASELINE))
    .addComponent(btnLogin)
    .addComponent(btnBatal))
    .addContainerGap(97, Short.MAX_VALUE))
);
```



```
javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel1,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE))
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
    );
pack();
```

```
}// </editor-fold>

private void btnLoginActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    try{

        Class.forName(Koneksi.driver);

        java.sql.Connection konek1 = DriverManager.getConnection(
            Koneksi.database,Koneksi.user,Koneksi.pass);

        Statement st1 = konek1.createStatement();

        ResultSet rs1 = st1.executeQuery("SELECT * FROM login WHERE
username ='"+ txtUser.getText()+"' && password ='"+ txtPass.getText()+"';");

        if(rs1.next()){

            MenuUtama s = new MenuUtama();
            s.setLocationRelativeTo(null);
            s.setVisible(true);
            btnLogin.setEnabled(true);
            this.dispose();

        } else {

            JOptionPane.showMessageDialog(rootPane,"Nama dan Kata Sandi
SALAH");

            txtUser.setText("");
            txtPass.setText("");
            txtUser.requestFocus();

        }

    }catch (Exception e){

    }

}
```

```
}
```

```
private void btnBatalActionPerformed(java.awt.event.ActionEvent evt) {  
  
    // TODO add your handling code here:  
  
    this.dispose();  
  
}  
  
/**  
 * @param args the command line arguments  
 */  
  
public static void main(String args[]) {  
  
    /* Set the Nimbus look and feel */  
  
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code  
(optional) ">  
  
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default  
look and feel.  
  
     * For details see  
     http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
*/  
  
    try {  
  
        for (javax.swing.UIManager.LookAndFeelInfo info :  
        javax.swing.UIManager.getInstalledLookAndFeels()) {  
  
            if ("Nimbus".equals(info.getName())) {  
  
                javax.swing.UIManager.setLookAndFeel(info.getClassName());  
  
                break;  
            }  
        }  
    }
```

```
        } catch (ClassNotFoundException ex) {  
  
            java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
        } catch (InstantiationException ex) {  
  
            java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
        } catch (IllegalAccessException ex) {  
  
            java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {  
  
            java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
        }  
    //</editor-fold>  
  
    /* Create and display the form */  
    java.awt.EventQueue.invokeLater(new Runnable() {  
  
        public void run() {  
  
            new login().setVisible(true);  
        }  
    });  
}  
  
// Variables declaration - do not modify  
private javax.swing.JButton btnBatal;
```

```
private javax.swing.JButton btnLogin;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JPasswordField txtPass;
private javax.swing.JTextField txtUser;
// End of variables declaration
}
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package appinventory;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.Date;
import javax.swing.ImageIcon;
import javax.swing.Timer;

/**
 *
 * @author DoniKalong
 */

```

```
public class MenuUtama extends javax.swing.JFrame {  
  
    /** * Creates new form MenuUtama */  
    public MenuUtama() {  
        initComponents();  
        java.awt.Dimension d =  
        java.awt.Toolkit.getDefaultToolkit().getScreenSize();  
        this.setSize(d.width,d.height);  
        icon();  
        JCjam();  
    }  
  
    public void icon(){  
        ImageIcon ico = new  
        ImageIcon("D:/program/AppKotaku/src/appkotaku/logo.png");  
        setIconImage(ico.getImage());  
    }  
  
    void JCjam(){  
  
        ActionListener taskPerformer = new ActionListener() {  
  
            @Override  
            public void actionPerformed(ActionEvent evt) {  
                String jc_jam = "";  
                String jc_menit = "";  
                String jc_detik = "";  
            }  
        };  
    }  
}
```

```
    Date JC = new Date();

    int nilai_jam = JC.getHours();

    int nilai_menit =JC.getMinutes();

    int nilai_detik = JC.getSeconds();

    if (nilai_jam <=9) jc_jam ="0";

    if (nilai_menit <=9) jc_menit ="0";

    if (nilai_detik <=9) jc_detik ="0";

    String jam = jc_jam + Integer.toString(nilai_jam);

    String menit = jc_menit + Integer.toString(nilai_menit);

    String detik = jc_detik + Integer.toString(nilai_detik);

    lblWaktu.setText(jam + ":" + menit + ":" + detik + "");

}

};

new Timer(1000, taskPerformer).start();

}

/***
 * This method is called from within the constructor to initialize the form.
 *
 * WARNING: Do NOT modify this code. The content of this method is
 * always
 *
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jMenu1 = new javax.swing.JMenu();
    lblWaktu = new javax.swing.JLabel();
    jMenuBar1 = new javax.swing.JMenuBar();
    jMenu2 = new javax.swing.JMenu();
    jMenuItem1 = new javax.swing.JMenuItem();
    jMenuItem2 = new javax.swing.JMenuItem();
    jMenuItem3 = new javax.swing.JMenuItem();
    jMenu3 = new javax.swing.JMenu();
    jMenuItem4 = new javax.swing.JMenuItem();
    jMenuItem6 = new javax.swing.JMenuItem();
    jMenuItem7 = new javax.swing.JMenuItem();
    jMenu4 = new javax.swing.JMenu();
    jMenuItem8 = new javax.swing.JMenuItem();
    jMenu5 = new javax.swing.JMenu();

    jMenu1.setText("jMenu1");

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    lblWaktu.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
    lblWaktu.setText("jLabel1");
}
```

```
jMenu2.setText("Master");

jMenuItem1.setText("Data Barang");
jMenuItem1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem1MouseReleased(evt);
    }
});
jMenu2.add(jMenuItem1);

jMenuItem2.setText("Data Pegawai");
jMenuItem2.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem2MouseReleased(evt);
    }
});
jMenu2.add(jMenuItem2);

jMenuItem3.setText("Data Lokasi");
jMenuItem3.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem3MouseReleased(evt);
    }
});
jMenu2.add(jMenuItem3);
```

```
jMenuBar1.add(jMenu2);

jMenu3.setText("Proses");

jMenuItem4.setText("Data Inventaris");
jMenuItem4.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem4MouseReleased(evt);
    }
});
jMenu3.add(jMenuItem4);

jMenuItem6.setText("Permintaan Barang");
jMenuItem6.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem6MouseReleased(evt);
    }
});
jMenu3.add(jMenuItem6);

jMenuItem7.setText("Pembelian Barang");
jMenuItem7.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem7MouseReleased(evt);
    }
});
```

```
jMenu3.add(jMenuItem7);

jMenuBar1.add(jMenu3);

jMenu4.setText("Laporan");

jMenuItem8.setText("Laporan Inventaris Kantor");
jMenuItem8.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jMenuItem8MouseReleased(evt);
    }
});
jMenu4.add(jMenuItem8);

jMenuBar1.add(jMenu4);

jMenu5.setText("Keluar");
jMenu5.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenuItem5MouseClicked(evt);
    }
});
jMenuBar1.add(jMenu5);

setJMenuBar(jMenuBar1);
```

```
    javax.swing.GroupLayout layout = new
    javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(lblWaktu,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 209,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(10, 10, 10)
            ).addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
                    .addComponent(lblWaktu,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 43,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(10, 10, 10)
                ).addGroup(layout.createSequentialGroup()
                    .addGap(10, 10, 10)
                    .addComponent(lblWaktu,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 296,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(10, 10, 10)
                ))
            .addGap(10, 10, 10)
        ).addGap(10, 10, 10)
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(lblWaktu,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 209,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(10, 10, 10)
            ).addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(lblWaktu,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 43,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(10, 10, 10)
            ).addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(lblWaktu,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 296,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(10, 10, 10)
            ))
            .addGap(10, 10, 10)
        ).addGap(10, 10, 10)
    );
    pack();
}

// </editor-fold>

private void jMenuItem1MouseReleased(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
}
```

```
barang db = new barang(null,true);
db.setLocationRelativeTo(null);
db.setVisible(true);

}

private void jMenuItem2MouseReleased(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    pegawai db = new pegawai(null,true);
    db.setLocationRelativeTo(null);
    db.setVisible(true);
}

private void jMenuItem3MouseReleased(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    lokasi db = new lokasi(null,true);
    db.setLocationRelativeTo(null);
    db.setVisible(true);
}

private void jMenuItem4MouseReleased(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    Inventaris db = new Inventaris(null,true);
    db.setLocationRelativeTo(null);
    db.setVisible(true);
}
```

```
private void jMenuItem6MouseReleased(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
  
    permintaan db = new permintaan(null,true);  
  
    db.setLocationRelativeTo(null);  
  
    db.setVisible(true);  
}
```

```
private void jMenuItem7MouseReleased(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
  
    pembelian db = new pembelian(null,true);  
  
    db.setLocationRelativeTo(null);  
  
    db.setVisible(true);  
}
```

```
private void jMenuItem5MouseClicked(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
  
    this.dispose();  
}
```

```
private void jMenuItem8MouseReleased(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
  
    LaporanInventaris db = new LaporanInventaris(null,true);  
  
    db.setLocationRelativeTo(null);  
  
    db.setVisible(true);  
}
```

```
/**  
 * @param args the command line arguments  
 */  
  
public static void main(String args[]) {  
  
    /* Set the Nimbus look and feel */  
  
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code  
(optional) ">  
  
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default  
look and feel.  
  
     * For details see  
     http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
    */  
  
    try {  
  
        for (javax.swing.UIManager.LookAndFeelInfo info :  
            javax.swing.UIManager.getInstalledLookAndFeels()) {  
  
            if ("Nimbus".equals(info.getName())) {  
  
                javax.swing.UIManager.setLookAndFeel(info.getClassName());  
  
                break;  
            }  
        }  
    } catch (ClassNotFoundException ex) {  
  
        java.util.logging.Logger.getLogger(MenuUtama.class.getName()).log(java.util.lo  
gging.Level.SEVERE, null, ex);  
    } catch (InstantiationException ex) {  
  
        java.util.logging.Logger.getLogger(MenuUtama.class.getName()).log(java.util.lo  
gging.Level.SEVERE, null, ex);  
    } catch (IllegalAccessException ex) {  
}
```

```
java.util.logging.Logger.getLogger(MenuUtama.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(MenuUtama.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);

}

//</editor-fold>

/* Create and display the form */

java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new MenuUtama().setVisible(true);

    }

});

}

// Variables declaration - do not modify

private javax.swing.JMenu jMenu1;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu3;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenu jMenu5;

private javax.swing.JMenuBar jMenuBar1;

private javax.swing.JMenuItem jMenuItem1;

private javax.swing.JMenuItem jMenuItem2;
```

```
private javax.swing.JMenuItem jMenuItem3;
private javax.swing.JMenuItem jMenuItem4;
private javax.swing.JMenuItem jMenuItem6;
private javax.swing.JMenuItem jMenuItem7;
private javax.swing.JMenuItem jMenuItem8;
private javax.swing.JLabel lblWaktu;
// End of variables declaration
}
```