BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Tujuan analisis yaitu memahami permasalahan secara menyeluruh, mengungkapkan apa yang harus dikerjakan oleh system perangkat lunak untuk memengelola hi kebutuhan pemakai dan menengetahui ruang lingkup produk dan pemakai yang akan menggunakan produk perangkat lunak tersebut.

Pada bagian ini akan dilakukan analisis terhadap sistem yang sedang berjalan. Tujuannya adalah untuk mengidentifikasi dan mengevaluasi berbagai kelemahan serta kelebihan yang terdapat pada sistem yang sudah dimiliki oleh PT. Garuda Mas Semesta yang nantinya akan digunakan untuk membuat rancangan sistem yang akan diusulkan.

3.1.1 Analisis Masalah

Dari hasil pengamatan, sistem lama yang dimiliki PT. Garuda Mas Semesta pada saat ini belum bisa memengelola hi kebutuhan PT. Garuda Mas Semesta. Kendala yang dihadapi adalah :

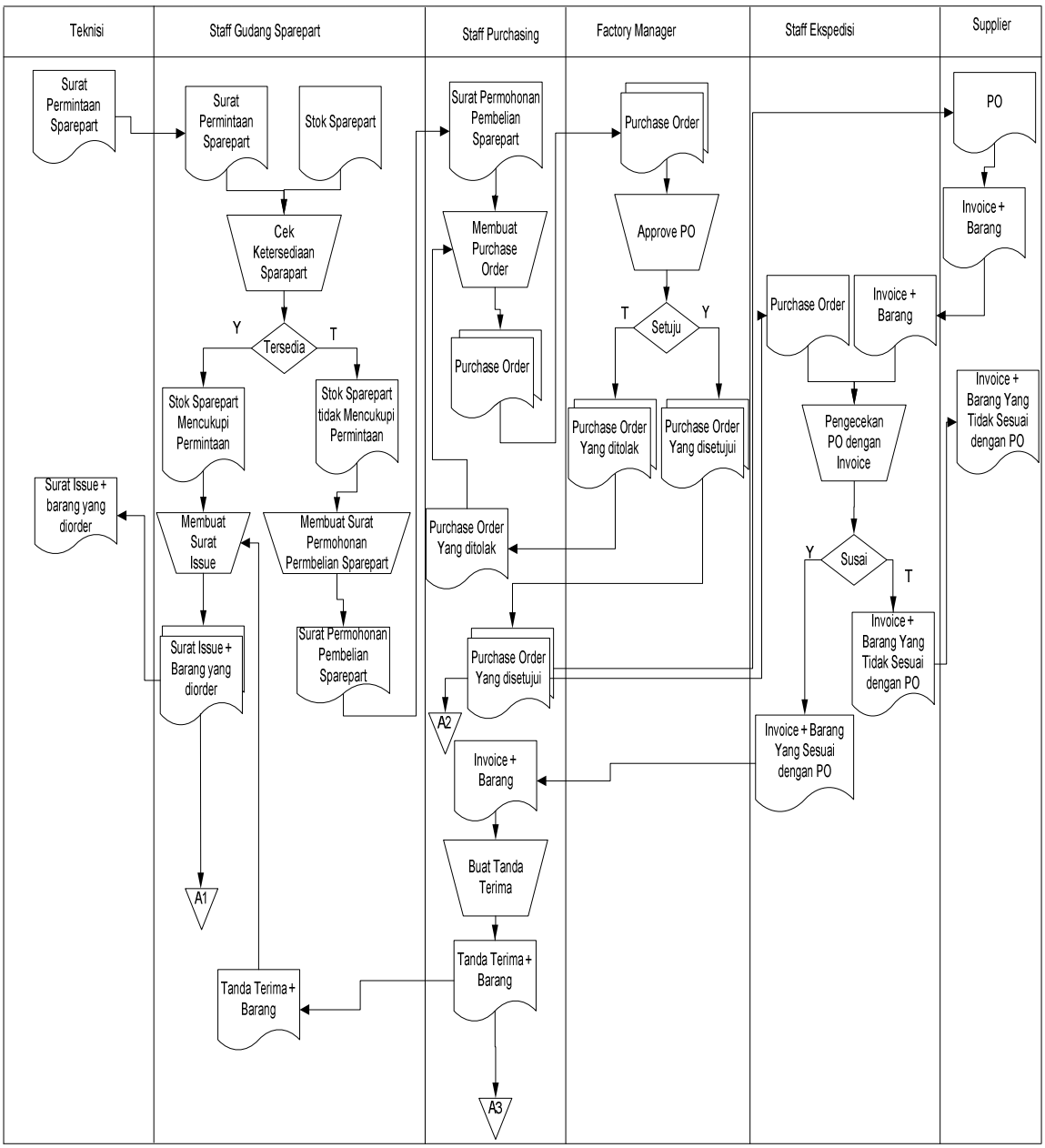
1. Sulitnya para staff gudang *sparepart* melakukan proses pemasukan data ke dalam sistem yang mengakibatkan ketidakakuratan data *sparepart.*
2. Sulitnya para staff *purchasing* mendapatkan data pembelian terakhir.
3. Sulitnya *factory manager* dalam menentukan jumlah *sparepart* yang akan dibeli.
4. Sulitnya staff ekspedisi mendapatkan data *purchase order.*

Berdasarkan analisis masalah, maka dibuatkan sebuah sistem yang memberikan kemudahan dalam proses pengolahan data *sparepart* serta memberikan peramalan kebutuhan *sparepart* untuk periode selanjutnya.

3.1.2 Analisis Prosedur yang Sedang Berjalan

Analisis prosedur atau analisis proses adalah tahapan yang memberi gambaran tentang sistem yang sedang berjalan. Analisis ini bertujuan untuk memberi gambaran yang lebih detail bagaimana cara kerja dari sistem yang sedang berjalan. Analisa prosedur pada bisnis proses pengendalian stock adalah sebagai berikut :

1. Bagian Teknisi mengelola lis surat permintaan *sparepart* dan memberikan surat permintaan *sparepart* ke bagian gudang.
2. Kemudian staff gudang *sparepart* mengecek stock barang yang dipesan, apabila *siaprepart* yang dipinta tersedia maka staff gudang akan membuat surat *issue* barang kemudian memberikannya kepada bagian Teknisi. apabila stock habis maka diteruskan kepada membuat surat permohonan pembelian barang yang kemudian diberikan kepada bagian *purchase*. Data *issue* barang disimpan (A1) di bagian gudang sebagai arsip.
3. Bagian *purchase* membuat *purchase order* sesuai dengan surat permohonan permohonan pembelian *sparepart*, lalu *purchase order* yang telah dibuat diberikan kepada *factory manager* untuk dilakukan proses persetujuan (*approval*).
4. *Purchase order* yang telah disetujui akan dilakukan pembelian kepada supplier.
5. Supplier akan memberikan barang dan invoice pembelian kepada staff ekspedisi, apabila sesuai maka diberikan kepada staff *purchase*. Apabila tidak sesuai maka dikembalikan kepada supplier.
6. Staff *purchase* akan membuat tanda terima, kemudian barang dan tanda terima akan diberikan kepada staff gudang *sparepart*.



Gambar 3.1 Flowmap Proses Order Barang

Keterangan:

A1 : arsip laporan *sparepart* keluar (*issue*)

A2 : arsip *purchase order*

A3: arsip tanda terima barang (*reveive*)

3.1.2.1 Aturan Bisnis

1. Pengelolaan Stok

   Untuk pengelolaan stok yaitu stok akan berkurang ketika terjadinya proses transaksi permintaan *sparepart* oleh teknisi melalui staff gudang *sparepart*. Stok akan betambah jika staff *purchase* telah melakukan pembelian dan staff gudang *sparepart* telah menerima *sparepart* yang telah dibeli oleh staff *purchase* setelah dilakukan pengecekan oleh staff ekpedisi.

2. Permintaan Pembelian

   Untuk permintaan pembelian yaitu proses permintaan yang dilakukan antara staff gudang *sparepart* dan staff *purchase*. Permintaan pembelian dilakukan jika stok *sparepart* yang ada di gudang *sparepart* tidak dapat memengelola hi permintaan teknisi.

3. *Purchase Order*

   Untuk *Purchase Order* yaitu kelanjutan dari proses permintaan pembelian yang dilakukan antara staff *purchase* dan *Factory manager*. Pada proses ini *factory manager* melakukan *approval* pada *purchase order order* yang telah dibuat oleh staff *purchase.*

4. *Sparepart* Masuk (*Receive*)

   Untuk *sparepart* masuk yaitu proses yang dilakukan oleh *staff* ekspedisi dan *staff* guddang *sparepart* dimana pada proses ini akan mengakibatkan penambahan stok *sparepart*. Proses ini dilakukan setiap kali ada pengiriman *sparepart* dari *supplier.*

5. *Sparepart* keluar (*issue*)

   Untuk *sparepart* keluar yaitu proses transaksi/permintaan yang dilakukan antara teknisi dengan staff gudang *sparepart.* Dimana proses ini akan mengakibatkan pengurangan stok *sparepart.*

6. Peramalan

   Dalam sistem ini sistem peramalan menggunakan metode *single moving average* dengan periode 3 harian. Untuk mendapatkan hasil peramalan yang paling mendekati untuk periode selanjutnya dapat butuhditentukan

dari perbandingan penyimpangan nilai yang lebih kecil dari *Mean Absolute Error*.

3.1.2.2 Evaluasi Sistem yang Sedang Berjalan

Berdasarkan analisis terhadap apa yang dibutuhkan dalam membangun aplikasi ini maka dapat dievaluasi hal-hal seperti pada tabel 3.1

Tabel 3.1 Evaluasi Sistem yang sedang berjalan

No	Permasalahan	Bagian/ Pihak
1	Dalam membuat laporan *sparepart* sebagian masih menggunakan cara manual dan membutuhkan waktu yang lama	Staff Gudang
2	Pengarsipan yang dilakukan secara manual sering kali tercecer atau hilang	Staff Gudang, Staff *Purchase*, Staff Ekspedisi
3	Selisih antara jumlah *sparepart* yang ada di gudang *sparepart* dengan data di komputer	Staff Gudang
4	Dalam melakukan penentuan jumlah *saparepart* yang akan dibeli menggunakan intuisi	*Factory Manager*

3.1.2.3 Solusi yang ditawarkan

Berdasarkan hasil evaluasi sistem yang sedang berjalan, solusi yang ditawarkan adalah pembuatan aplikasi pengolahan data *sparepart* dengan ditambahkan model peramalan untuk membantu *factory manager* menentukan jumlah *sparepart* yang akan dibeli. Dalam sistem ini staff gudang *sparepart* tidak perlu lagi membuat permintaan pembelian *spaperpart* secara manual, cukup mengisi form yang sudah ada dalam sistem saja. Staff *purchase* dan *Staff* ekpedisi sudah dapat melihat laporan *sparepart* masuk, *sparepart* keluar dan permintaan pembelian *sparepart* dengan lebih cepat dan akurat. Sistem ini juga dapat meminimalkan selisih antara jumlah *sparepart* yang ada di gudang *sparepart* dengan data di komputer dengan cara membuat standar pengkodean nama *sparepart*.

3.1.3 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibituhkan untuk sistem yang akan dibangun sampai dengan sistem di implementasikan.

3.1.3.1 Analisis Pengguna

Analisis pengguna aplikasi dilakukan untuk mengetahui kemampuan pengguna dalam mengoperasikan perangkat aplikasi komputer dalam melakukan pekerjaannya. Pengguna yang akan terlibat dalam penggunaan aplikasi yang dibuat dijelaskan pada tabel 3.2

Tabel 3.2 Karakteristik Pengguna yang Sedang Berjalan

Bagian	Tanggung Jawab	Tingkat Keterampilan
IT	Merawat perangkat komputer, membuat program aplikasi, dan keamanan data.	Bisa menggunakan komputer, mengerti semua alur bisnis, mengerti php dan mysql
Staff Gudang Sparepart	Membuat laporan keluar masuk *sparepart.*	Bisa menggunakan komputer dan mengerti alur bisnis gudang
Staff *Purchase*	Membuat laporan pembelian	Bisa menggunakan komputer dan mengerti alur bisnis *purchasing*
Staff Ekspedisi	Memeriksa seluruh barang yang masuk dan keluar dari lingkungan pabrik baik hasil pembelian atau pun penjualan.	Bisa menggunakan komputer dan mengerti alur bisnis ekspedisi
Factory Manager	Bertanggung jawab atas seluruh kegiatan produksi	Mengeti alur proses seluruh produksi

Karakteristik pengguna yang dibutuhkan untuk menjalankan aplikasi dapat dilihat pada tabel 3.3

Tabel 3.3 Karakteristik Pengguna yang Dibutuhkan

Pengguna	**Hak Akses**	**Tanggung Jawab**	**Tingkat Keterampilan**
Administrator	Mengelola Data Jenis *Warehouse*, data Jenis *Sparepart*, data *Warehouse*, data Sparepart, data *Vendor*, data *store request*, data *purchase request*, data *purchase order*, data *issue*, data *receive*, data peramalan, laporan *request*, laporan PO, laporan *Receive*, laporan stok, laporan issue	Menjaga aplikasi agar dapat berfungsi dengan baik	Bisa menggunakan komputer, mengerti semua alur bisnis, mengerti php dan mysql
Staff Gudang Sparepart	Mengelola data *store request*, data *purchase request*, data *issue*, data *receive*, laporan *request*, laporan *Receive*, laporan	Mengelola *sparepart* masuk dan keluar, mengelola permintaan pembelian *sparepart*	Bisa menggunakan komputer dan mengerti alur bisnis gudang

	stok, laporan issue		
Staff *Purchase*	Mengelola data *Purchase order*, Laporan request, laporan PO, Laporan *receive*	Mengelola *Purchase Order*	Bisa menggunakan komputer dan mengerti alur bisnis *purchasing*
Staff Ekspedisi	Laporan PO	Memeriksa keseuaian antara *sparepart* yang datang dengan *PO*	Bisa menggunakan komputer dan mengerti alur bisnis ekspedisi
Factory Manager	Mengelola data *PO*, data Peramalan, laporan *request*, laporan PO, laporan *Receive*, laporan stok, laporan issue	*Approval* pembelian *sparepart*	Bisa menggunakan komputer

Berdasarkan tabel 3.2 dan 3.3 maka dapat disimpulkann bahwa karakteristik pengguna yang ada sudah memengelola hi syarat untuk menjalankan aplikasi yang akan dibuat.

3.1.3.2 Analisis Perangkat Keras

Perangkat keras akan digunakan untuk memperlancar proses pengaplikasian sistem informasi pengendalian stok *spare part* ini adalah sebagai berikut :

1. Spesifikasi perangkat keras yang digunakan untuk menjalankan aplikasi, dapat dilihat pada tabel 3.4

Tabel 3.4 Spesifikasi Perangkat Keras untuk Menjalankan Aplikasi

Perangkat Keras	Spesifikasi
Processor	Intel Pentium 4
Memory	512 Mb
Harddisk	40 Gb
Vga	32 Mb

2. Perangkat keras yang dimiliki PT. Garuda Mas Semesta

   PT. Garuda Mas Semesta memiliki dua spesifikasi komputer standar yang biasa digunakan untuk membantu proses dokumentasi, kedua spesifikasi itu dapat dilihat pada tabel 3.5

Tabel 3.5 Spesifikasi Perangkat Keras yang Dimiliki PT. Garuda Mas

Perangkat Keras	Spesifikasi	
	Spesifikasi Standar 1	Spesifikasi Standar 2
Processor	Intel Core i3	Intel Dual Core
Memory	2 Gb	1 Gb
Harddisk	250 Gb	160 Gb
Vga	128 Mb	128 Mb

Berdasarkan tabel 3.4 dan 3.5 diketahui tidak ada masalah/kekurangan pada perangkat keras untuk menjalankan aplikasi dengan perangkat keras yang dimiliki PT. Garuda Mas Semesta maka spesifikasi standar 1 akan dijadikan sebagai *server* dan spesifikasi standar 2 akan dijadikan sebagai *client.*

3.1.3.3 Analisis Perangkat Lunak

Dalam analisis kebutuhan perangkat lunak, dibutuhkan beberapa *software* yang mendukung untuk digunakan dalam menjalankan aplikasi yang dibuat. *Software* yang digunakan untuk menjalankan aplikasi antara lain :

1. Sistem Operasi *Windows Xp*
2. Browser misalnya Mozilla Firefox atau google chrome

3. Xampp

Sedangkan *software* standar yang dimiliki PT. Garuda Mas Semesta adalah sebagai berikut :

1. Sistem Operasi Windows *Xp Profesional/Home Edition.*
2. Browser *Mozilla Firefox* dan *Internet Explorer.*

Terdapat kekurangan pada *software* standar yang dimiliki PT. Garuda Mas Semesta maka akan dilakukan penambahan *software* Xampp pada komputer *server* agar aplikasi dapat berjalan dengan lancar

3.1.4 Analisis Metode Peramalan

Peramalan merupakan proses memprediksi keadaan yang tidak diketahui. PT. Garuda Mas Semesta biasanya menggunakan data stok terakir untuk menentukan pengadaan *sparepart.* Jika stok *sparepart* kurang maka staff *purchase* akan menghubungi *supplier* untuk melakukan pemesanan ulang dan biasanya jumlah *sparepart* yang dipesan hanya berdasarkan intuisi dari *factory manager* sehingga sering terjadinya pembelian *sparepart* berlebih atau bahkan kurang.

3.1.4.1 Perhitungan Peramalan

Data yang digunakan untuk peramalan hanyalah data dari *sparepart* keluar (*issue*). Untuk meramalkan jumlah pembelian *sparepart* pada periode mendatang, digunakan metode *single moving averarage.* Peramalan dengan metode *single moving average* dilakukan dengan menggunakan hasil rata-rata pada periode sebelumnya.

$$S_{t+1} = \frac{X_t + X_{t-1} + \cdots + X_{t-n+1}}{n}$$

$$= \frac{1}{n} \sum \sum_{j=t-n+1}^{t} X_j$$

Dimana :

St = Peramalan untuk 75eriod eke t

X1= Data untuk periode t

n = Jangka waktu *moving average*

Misalkan membuat peramalan untuk *elco* dengan metode 3 harian *moving average* dari tanggal 1 Januari 2013 sampai 1 Januari 2013 adalah sebagai berikut:

1 Januari 2013 = 9

2 Januari 2013 = 9

3 Januari 2013 = 8

Peramalan kebutuhan untuk tanggal 4 Januari adalah

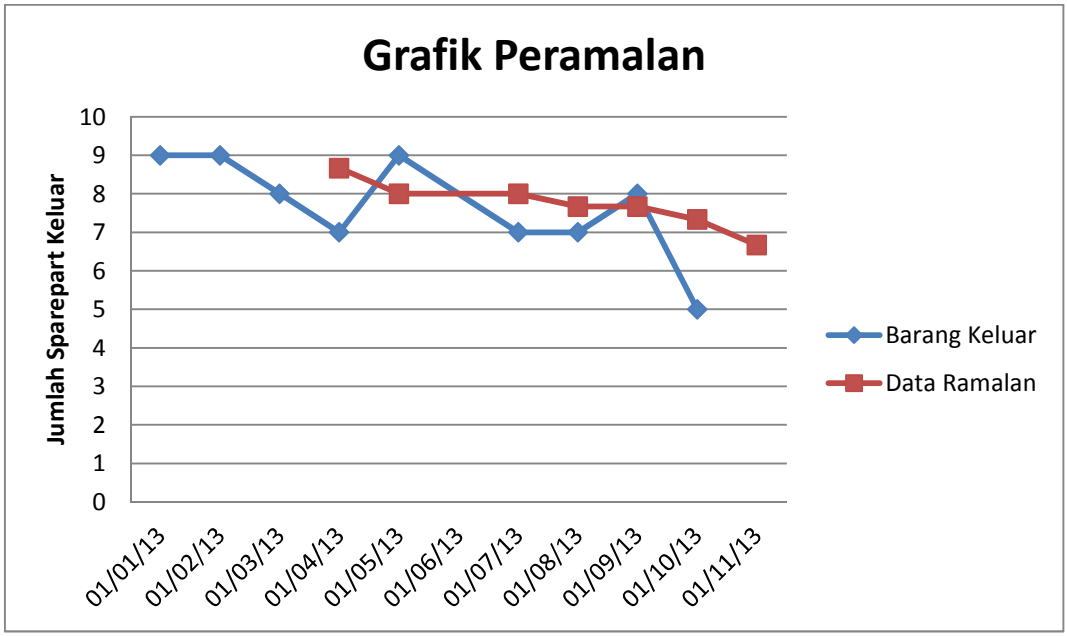
$$\frac{9+9+8}{3} = 8.7$$

Jadi kebutuhan *elco* pada periode selanjutnya adalah 8.7 atau 9

Untuk lebih jelasnya dapat dilihat pada tabel 3.6

Tabel 3.6 Perhitungan Peramalan

No	Tgl	Barang Keluar (kebutuhan Sparepart)	Peramalan
1	1/1/2013	9	
2	2/1/2013	9	
3	3/1/2013	8	
4	4/1/2013	7	8.7
5	5/1/2013	9	8.0
6	7/1/2013	7	8.0
7	8/1/2013	7	7.7
8	9/1/2013	8	7.7
9	10/1/2013	5	7.3
10	11/1/2013		6.7



Gambar 3.2 Grafik Peramalan Kebutuhan *Sparepart*

3.1.4.2 Perhitungan Kesalahan Peramalan

Untuk menghitung kesalahan (*error*) peramalan biasanya digunakan *mean absolute error*. *Mean absolute error* adalah rata-rata nilai *absolute* dari kesalahan meramal (tidak menghiraukan positif dan negative)

$$MAE = \sum \frac{|X_t - S_t|}{n}$$

Tabel 3.7 Perhitungan Kesalahan Peramalan

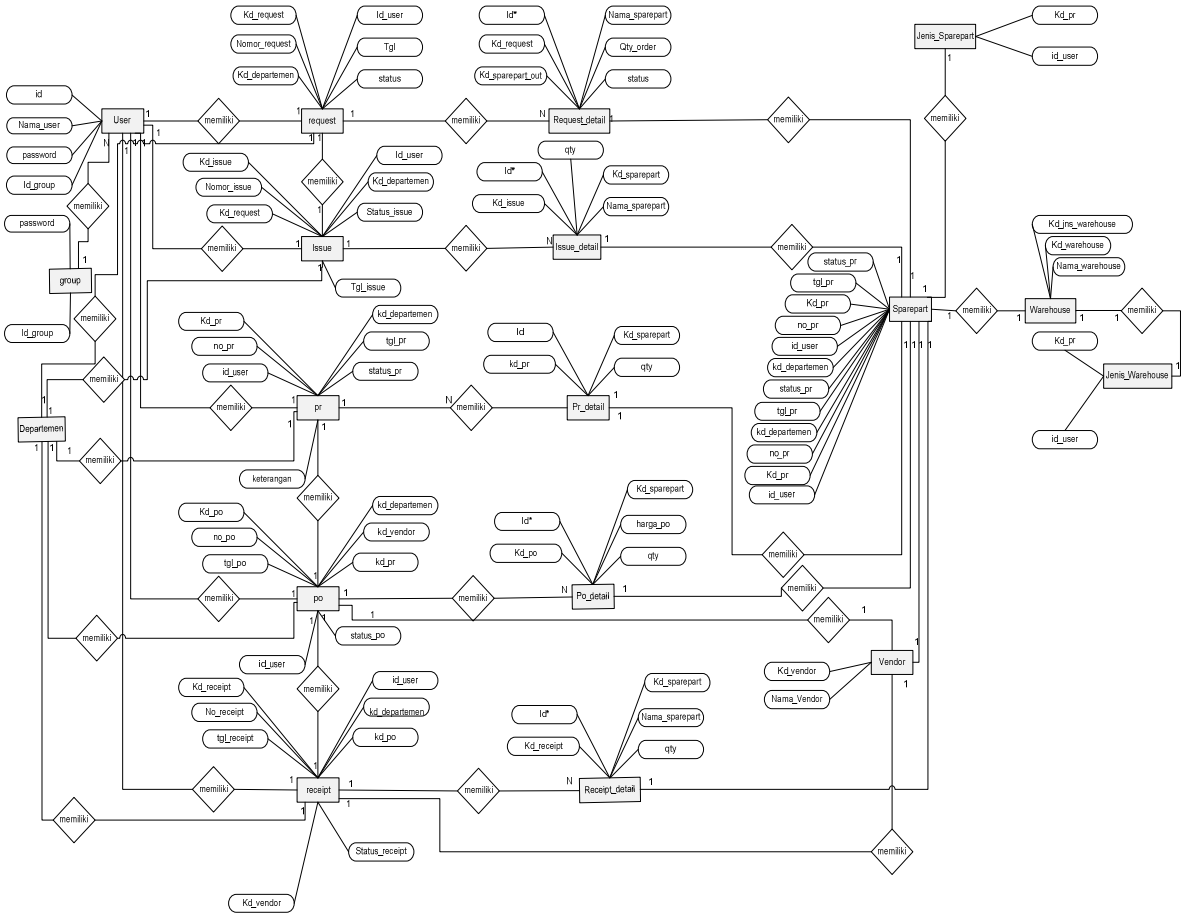
No	Tgl	*Sparepart* Keluar	Peramalan	Kesalahan Peramalan		
				Galat	Absolute Error	Absolute^2 Error
1	1/1/2013	9				
2	2/1/2013	9				
3	3/1/2013	8				
4	4/1/2013	7	8.7	-1.7	1.7	2.8
5	5/1/2013	9	8.0	1.0	1.0	1.0

6	7/1/2013	7	8.0	-1.0	1.0	1.0
7	8/1/2013	7	7.7	-0.7	0.7	0.4
8	9/1/2013	8	7.7	0.3	0.3	0.1
9	10/1/2013	5	7.3	-2.3	2.3	5.4
10	11/1/2013		6.7			
Jumlah				-4.3	7.0	10.8
Rata-rata				-0.7	1.2	1.8

3.1.4.3 Analisis ERD (*Entity Relation Diagram*)

Dalam menggambarkan hubungan antara data yang terdapat dalam sistem digunakan alat bantu yaitu ERD (*Entity Relation Diagram*). ERD merupakan salah satu cara untuk mengolah basisdata sehingga data resebut dapat diketahui hubungan atau relasi antar entitas.

Agar lebih jelas mengenai ERD pada sistem yang akan dibuat dapat dilihat pada gambar 3.3



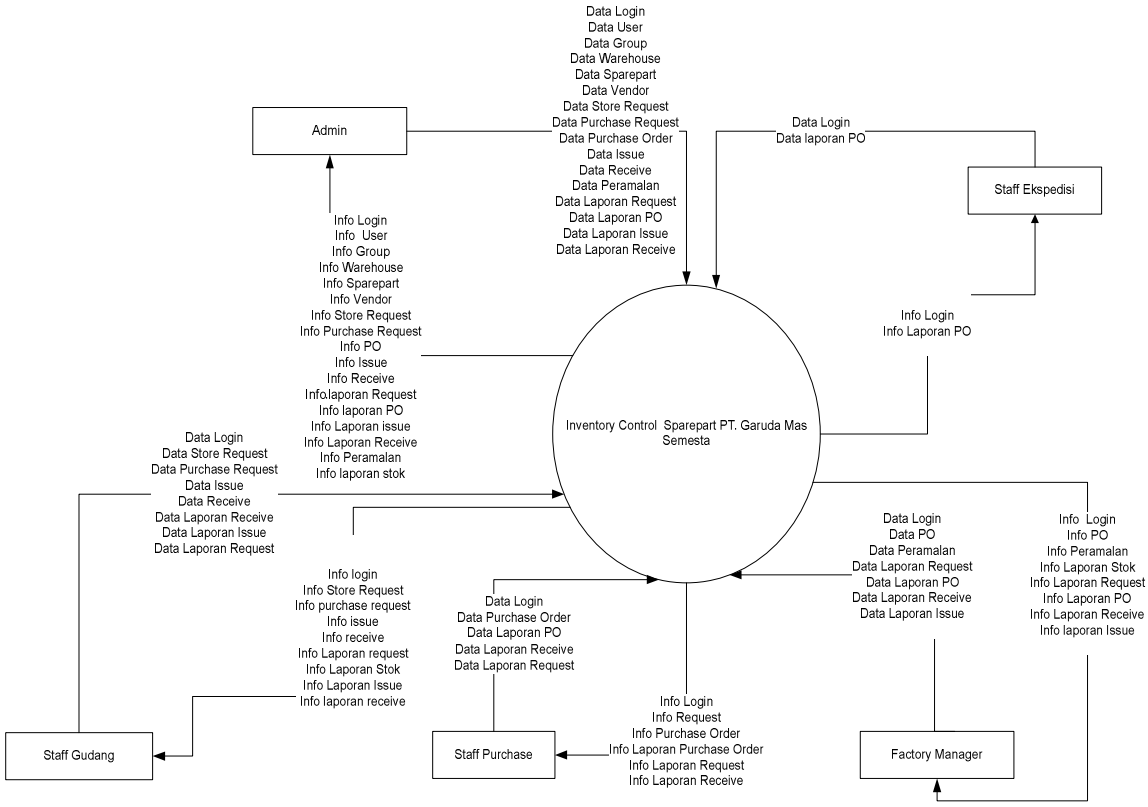
Gambar 3.3 ER Diagram

3.1.5 Analisis Kebutuhan Fungsional

Kebutukan fungsional berhubungan dengan fitur software yang akan dibuat atau dikembangkan. Berikut ini adalah tahapan analisis kebutuhan fungsional sistem informasi pengendalian stok *spare part*.

3.1.5.1 Diagram Konteks

Diagram konteks adalah diagram yang menggambarkan secara umum yang menjadi masukan, proses, dan keluaran yang terjadi pada sebuah sistem. Diagram konteks untuk perangkat lunak yang akan dibagun sebagai berukut :



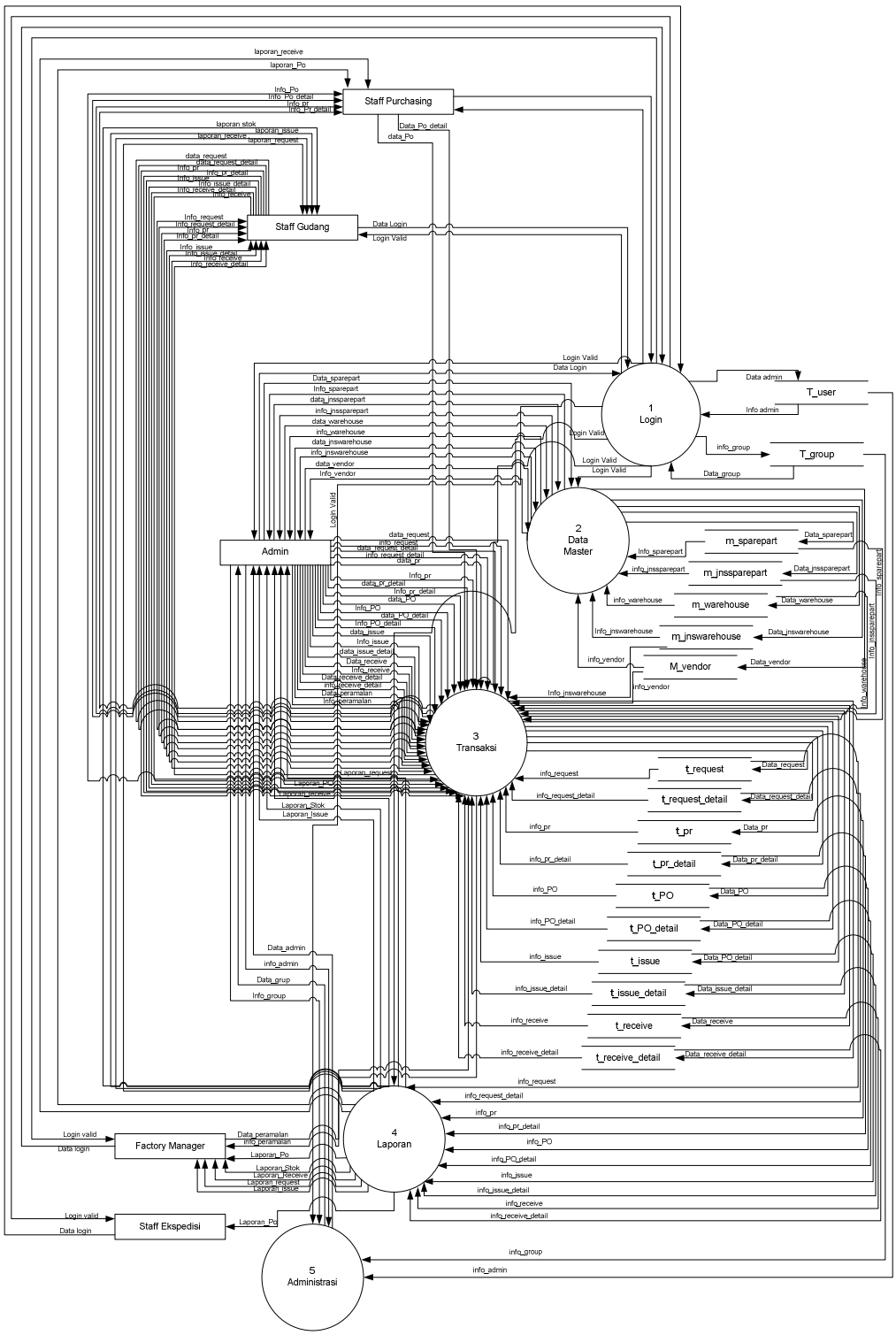
Gambar 3.4 Diagram Konteks

3.1.5.2 Data Flow Diagram

DFD (*Data Flow Diagram*) merupakan suatu media yang digunakan untuk menggambarkan aliran data yang mengalir pada suatu sistem informasi. DFD aplikasi yang diusulkan terdiri dari beberapa bagian, berikut DFD selemgkapnya :

3.1.5.2.1 DFD Level 1

Proses DFD level 1 adalah proses yang menggambarkan keseluruhan dari aplikasi yang dibuat, DFD level 1 dapat dilihat pada gambar 3.5.



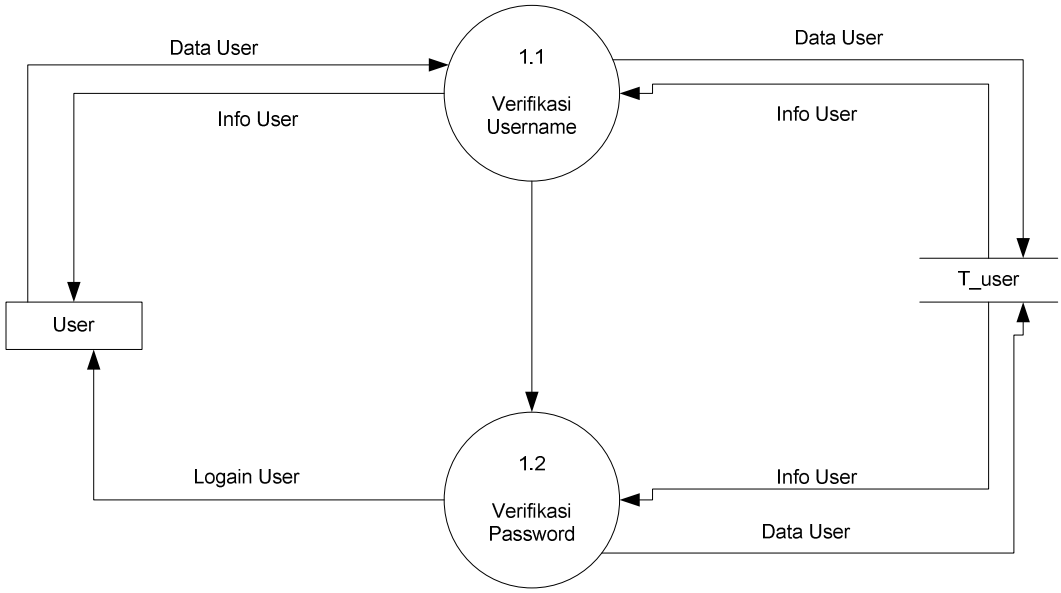
Gambar 3.5 DFD Level 1

3.1.5.2.2 DFD Level 2

Untuk menjelaskan lebih terperinci tentang proses-proses yang ada pada DFD level 1 maka akan digambarkan proses-prosesnya.

3.1.5.2.2.1 DFD Level 2 Proses *Login*

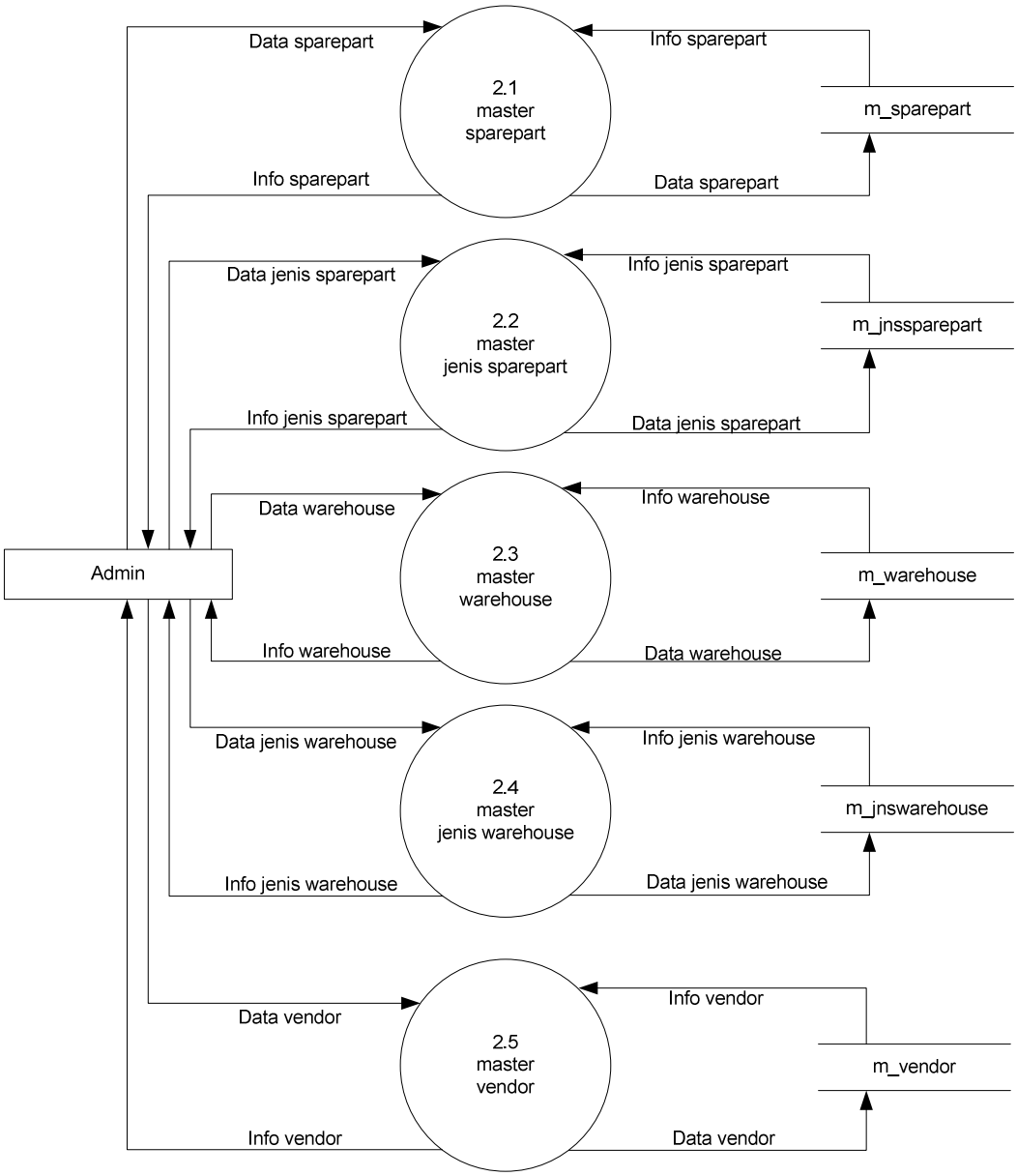
Diagram alir data proses *login* adalah proses untuk *login.* Seperti terlihat pada gambar 3.6



Gambar 3.6 DFD Level 2 Proses *Login*

3.1.5.2.2.2 DFD Level 2 Proses *Master*

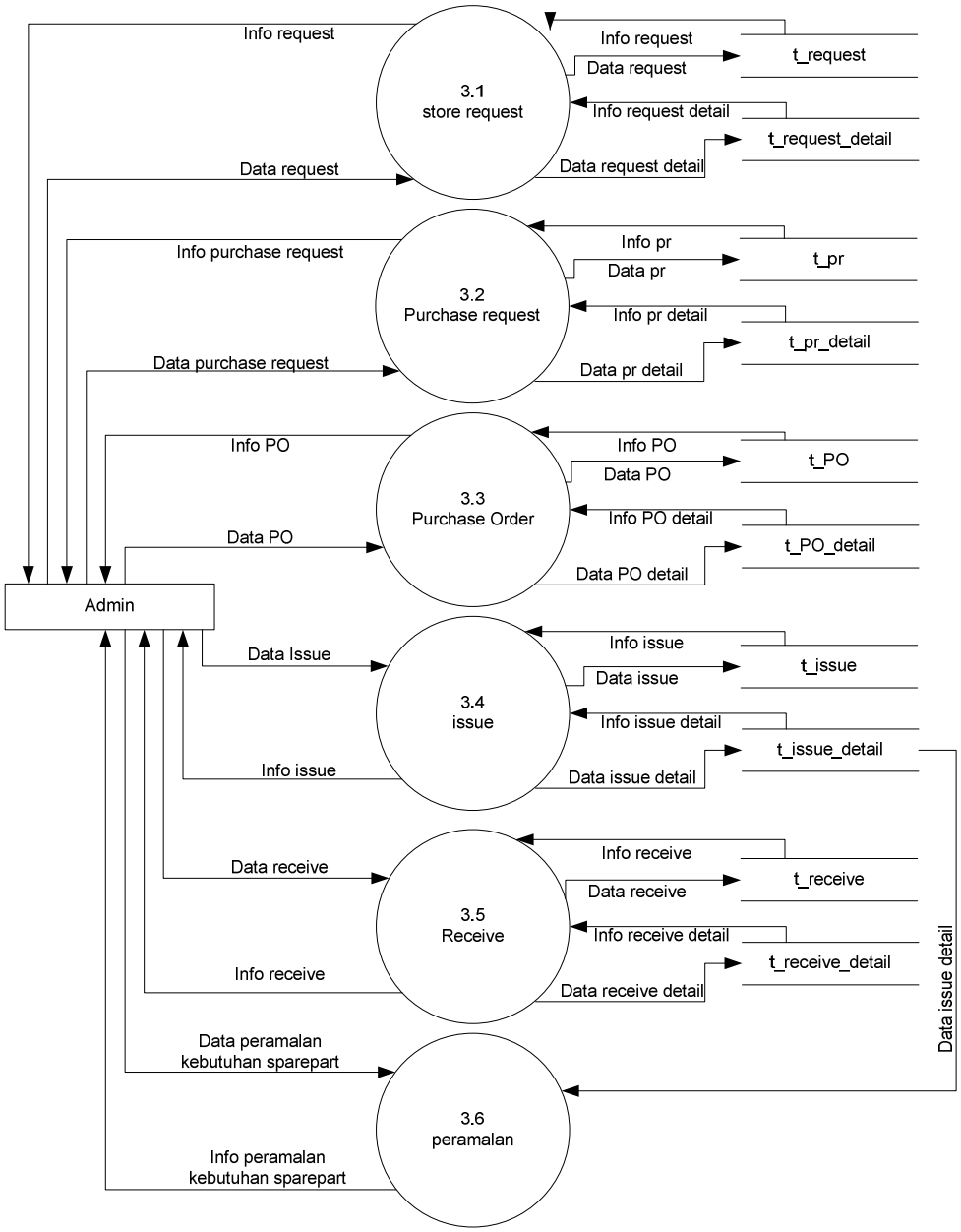
Diagram alir data proses data *master* adalah proses untuk data *warehouse,* data jenis *warehouse*, data *sparepart*, data jenis *sparepart*, dan data *vendor.* Seperti terlihat pada gambar 3.7



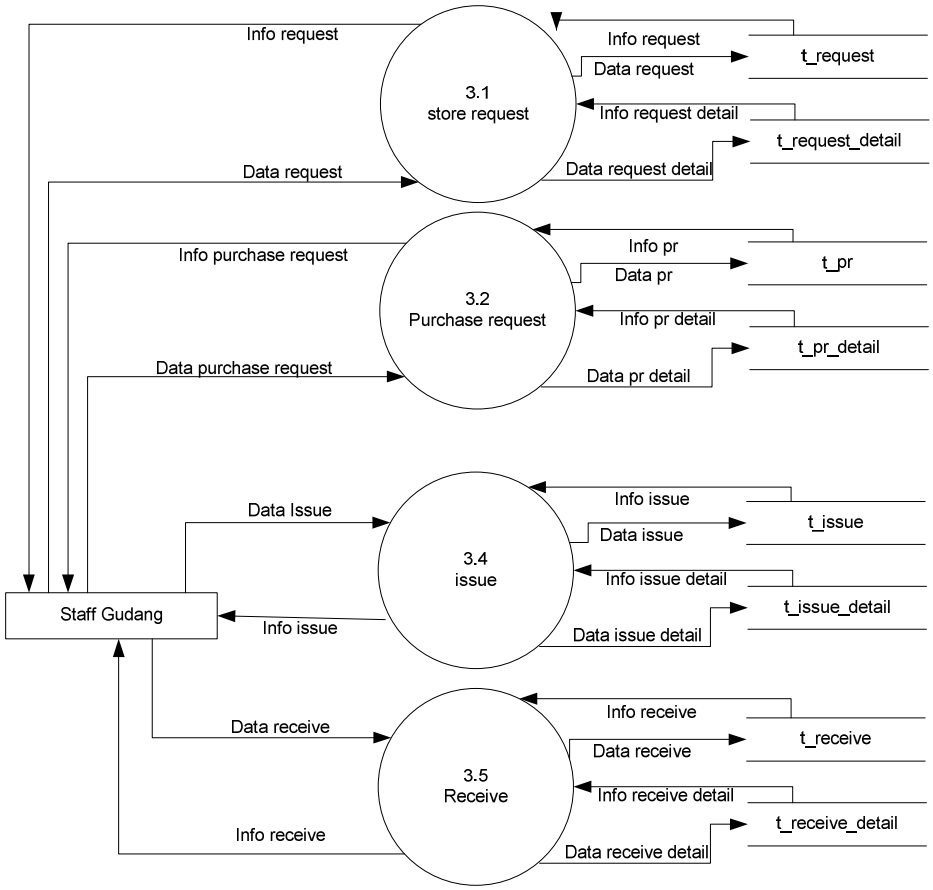
Gambar 3.7 DFD Level 2 Proses Data *Master*

3.1.5.2.2.3 DFD Level 2 Proses *Transaksi*

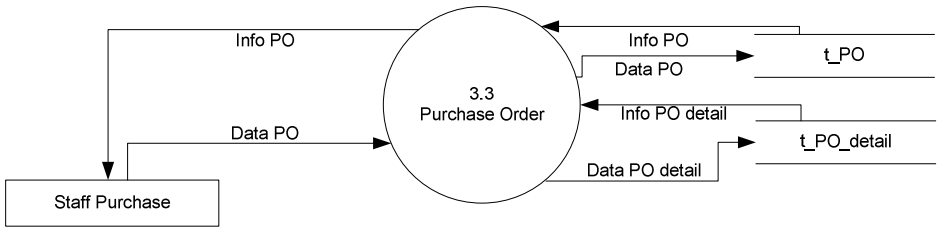
Diagram alir data proses *transaksi* adalah proses untuk data *request*, data *requeat detail*, data *purchase request*, data *purchase request detail*, data *purchase order*, data *purchase order detail*, data *issue*, data *issue detail*, data *receive*, data *receive detail*, dan peramalan kebutuhan *sparepart.* Seperti terlihat pada gambar 3.8



Gambar 3.8 DFD Level 2 Proses Transaksi (Admin)



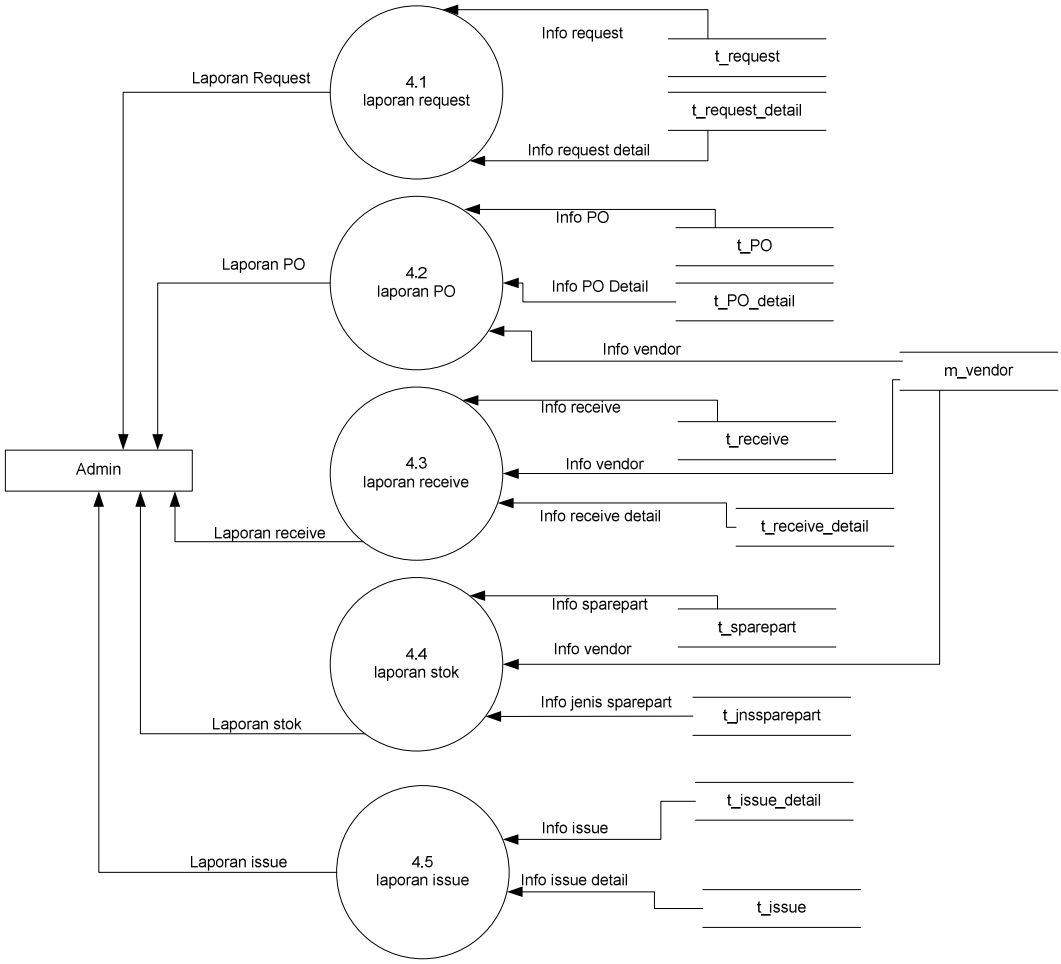
Gambar 3.9 DFD Level 2 Proses Transaksi (Staff Gudang *Sparepart*)



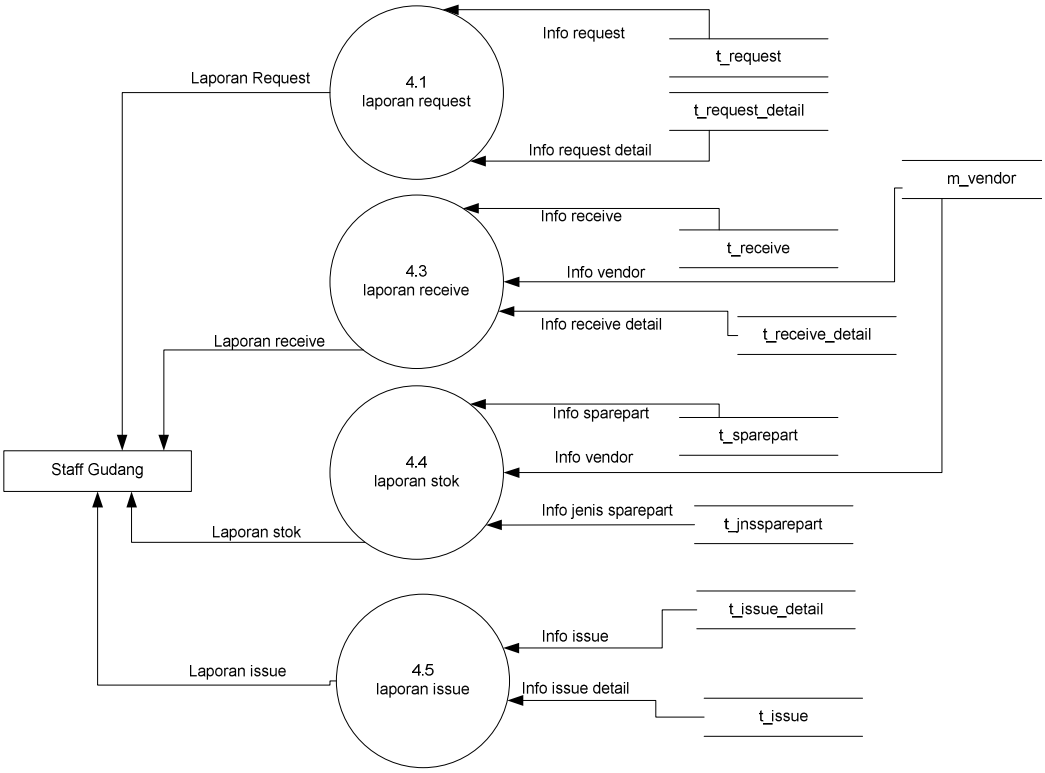
Gambar 3.10 DFD Level 2 Proses Transaksi (Staff Purchasing)

3.1.5.2.2.4 DFD Level 2 Proses Laporan

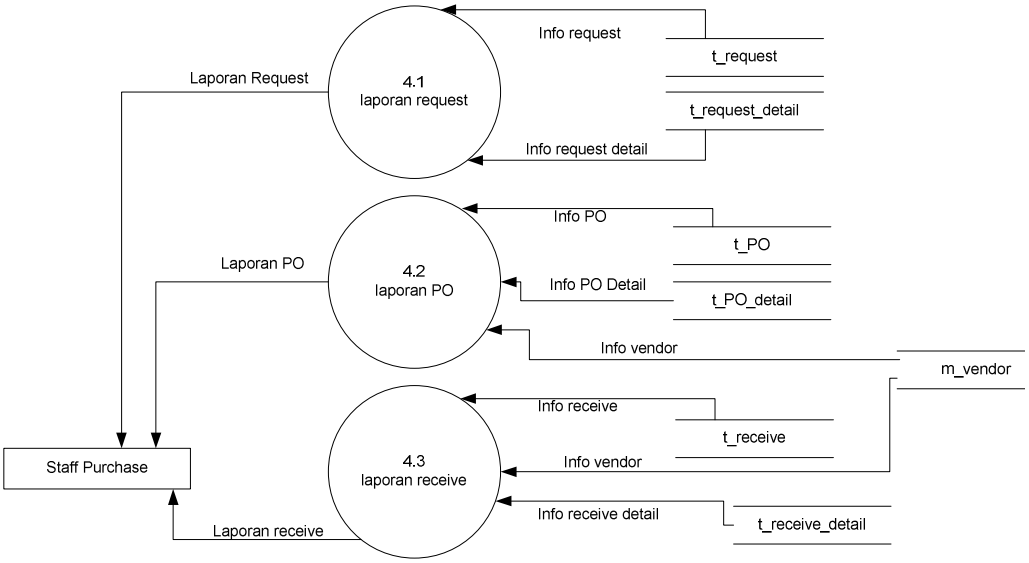
Diagram alir data proses laporan adalah proses keluaran dari data yang telah dimasukan. Data laporan tersebut terdiri dari laporan *request*, laporan PO, laporan *receive*, laporan stok, dan laporan *issue* .Seperti terlihat pada gambar 3.9



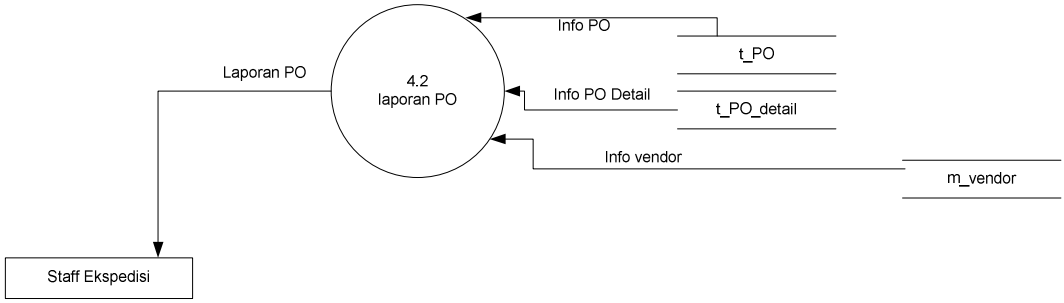
Gambar 3.11 DFD Level 2 Laporan (Admin dan *Factory Manager*)



Gambar 3.12 DFD Level 2 Laporan (Staff Gudang)



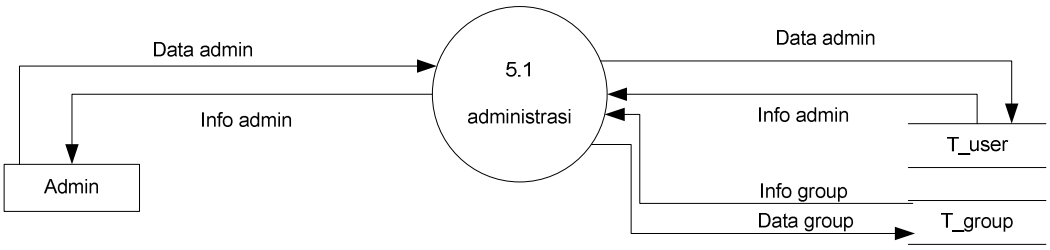
Gambar 3.13 DFD Level 2 Laporan (*Staff Purchase*)



Gambar 3.14 DFD Level 2 Laporan (Staff Ekspedisi)

3.1.5.2.2.5 DFD Level 2 Proses Administrasi

Diagram alir data proses administrasi adalah proses untuk pengelolaan pengguna/*user* .Seperti terlihat pada gambar 3.15



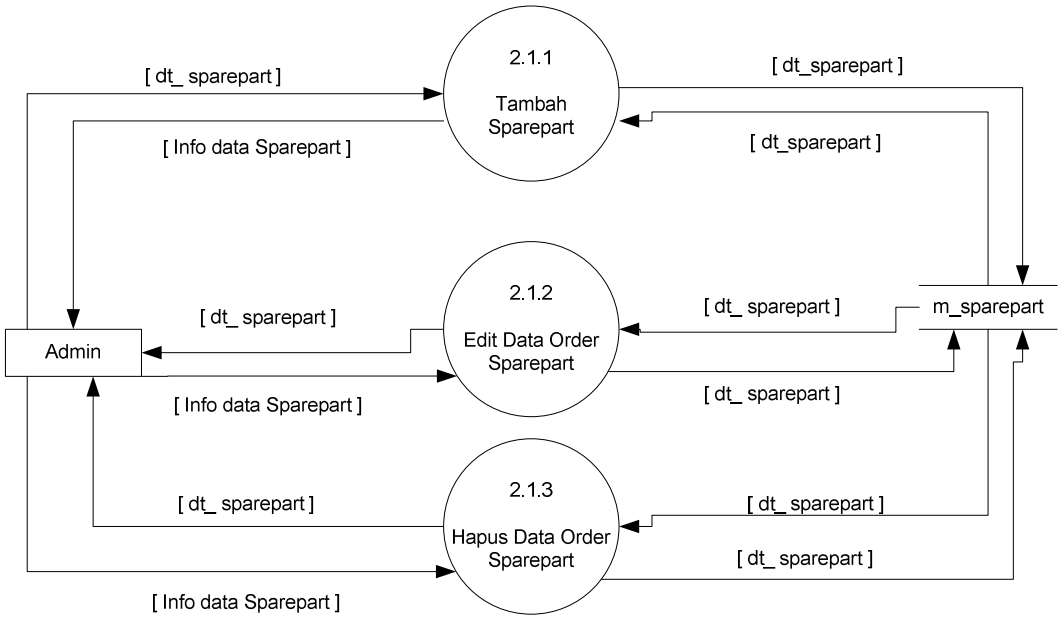
Gambar 3.15 DFD Level 2 Proses Administrasi

3.1.5.2.3 DFD Level 3

Dari DFD level 2 diatas terdapat proses yang harus diturunkan k elevel 3, berikut akan dipaparkan turunan proses dari DFD level 2.

3.1.5.2.3.1 DFD Level 3 Proses 2.1 Master *Sparepart*

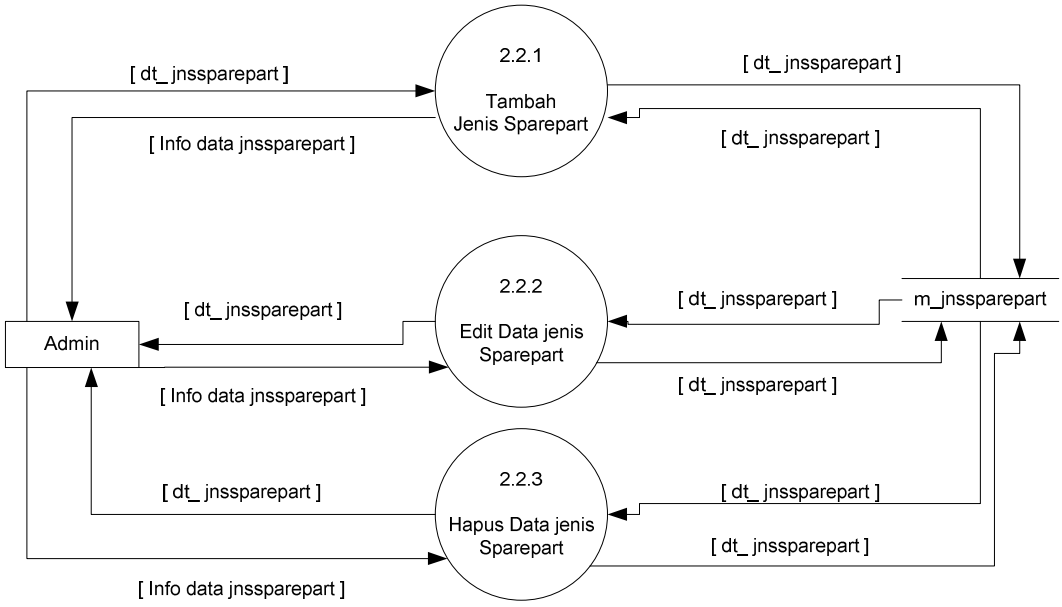
Diagram alir data pada proses data master *sparepart* adalah proses untuk tambah, hapus, dan ubah. Seperti pada gambar 3.16



Gambar 3.16 DFD Level 3 Proses 2.1 Master *Sparepart*

3.1.5.2.3.2 DFD Level 3 Proses 2.2 Master Jenis *Sparepart*

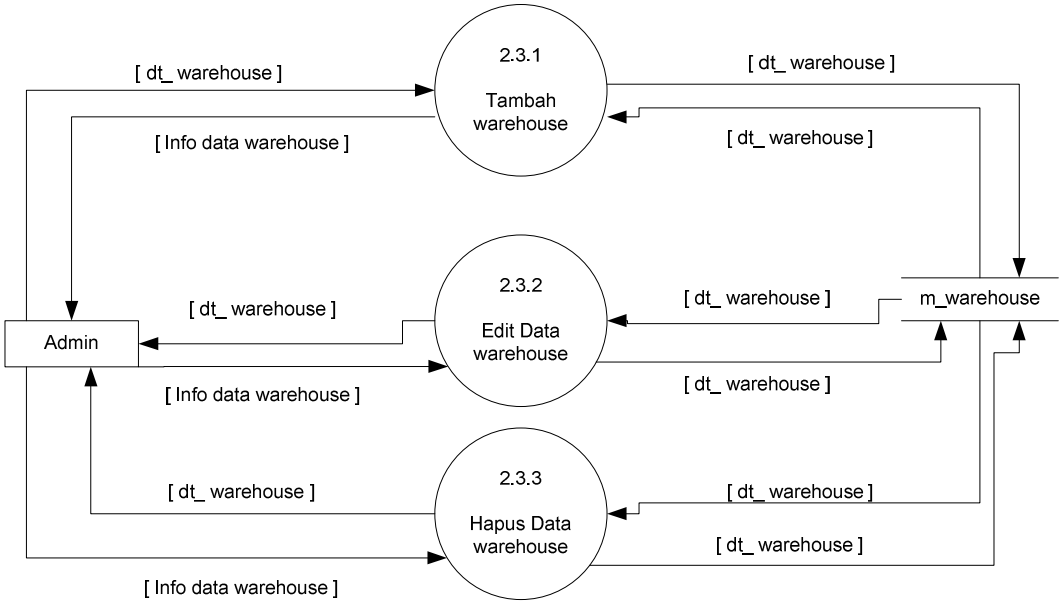
Diagram alir data pada proses data master jenis *sparepart* adalah proses untuk tambah, hapus, dan ubah. Seperti pada gambar 3.17



Gambar 3.17 DFD Level 3 Proses 2.2 Master Jenis *Sparepart*

3.1.5.2.3.3 DFD Level 3 Proses 2.3 Master *Warehouse*

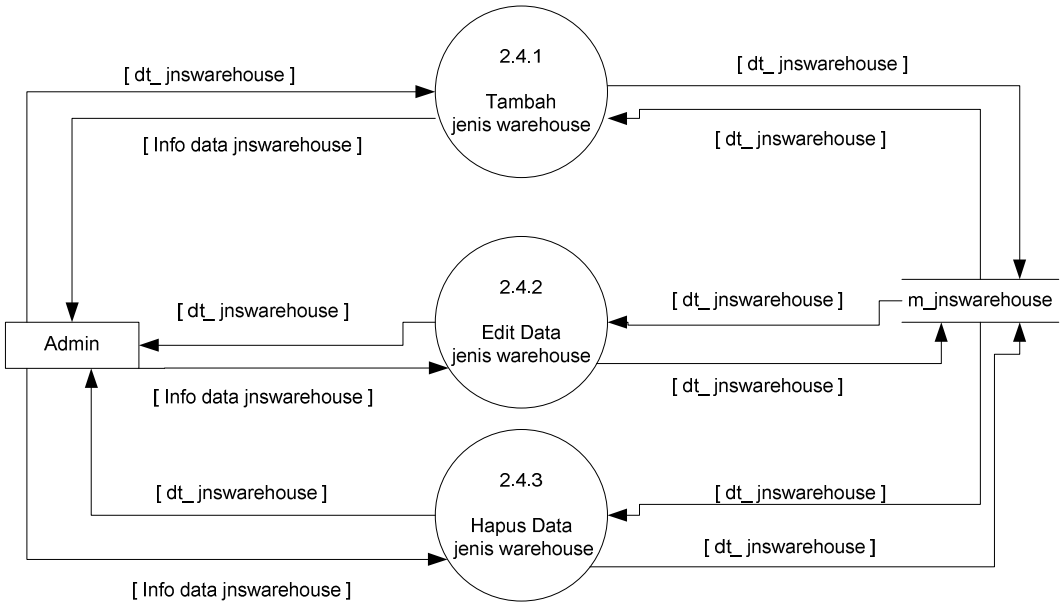
Diagram alir data pada proses data master *warehouse* adalah proses untuk tambah, hapus, dan ubah. Seperti pada gambar 3.18



Gambar 3.18 DFD Level 3 Proses 2.3 Master *Warehouse*

3.1.5.2.3.4 DFD Level 3 Proses 2.4 Master Jenis *Warehouse*

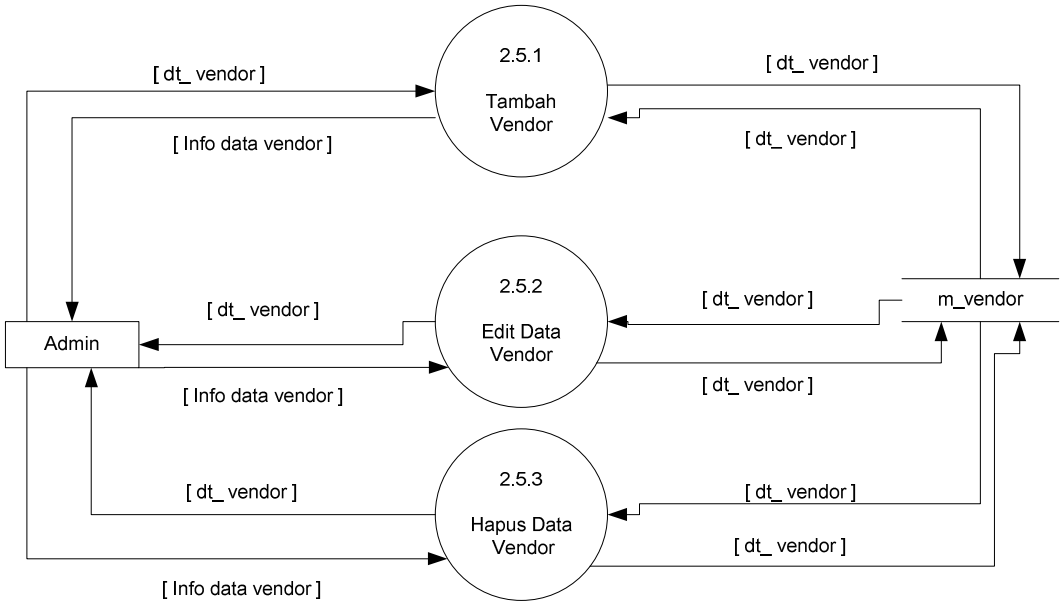
Diagram alir data pada proses data master *warehouse* adalah proses untuk tambah, hapus, dan ubah. Seperti pada gambar 3.19



Gambar 3.19 DFD Level 3 Proses 2.4 Master Jenis *Warehouse*

3.1.5.2.3.5 DFD Level 3 Proses 2.5 Master *Vendor*

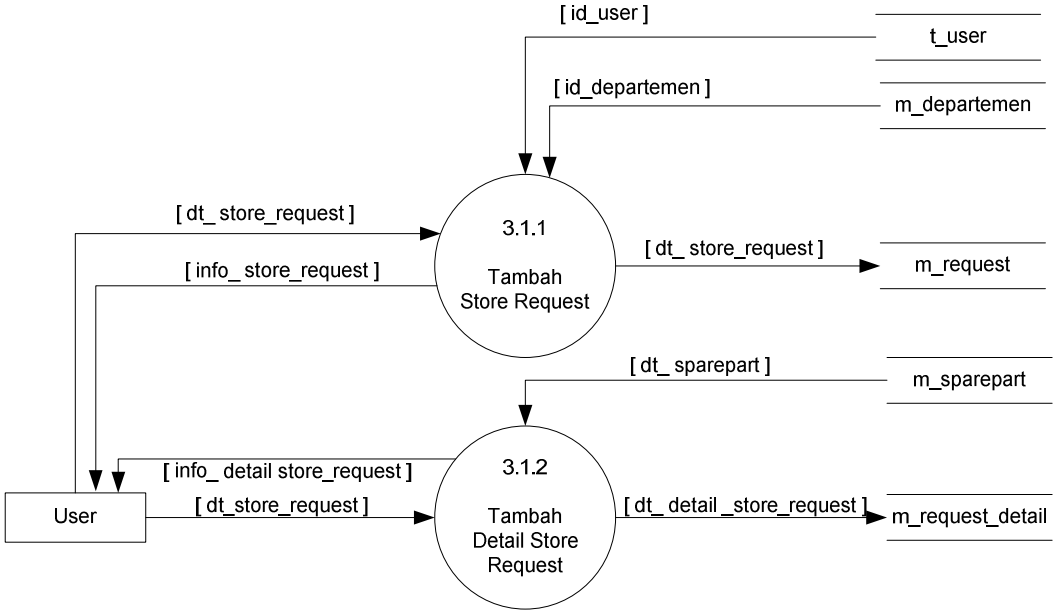
Diagram alir data pada proses data master *vendor* adalah proses untuk tambah, hapus, dan ubah. Seperti pada gambar 3.20



Gambar 3.20 DFD Level 3 Proses 2.5 Master *Vendor*

3.1.5.2.3.6 DFD Level 3 Proses 3.1 *Store Request*

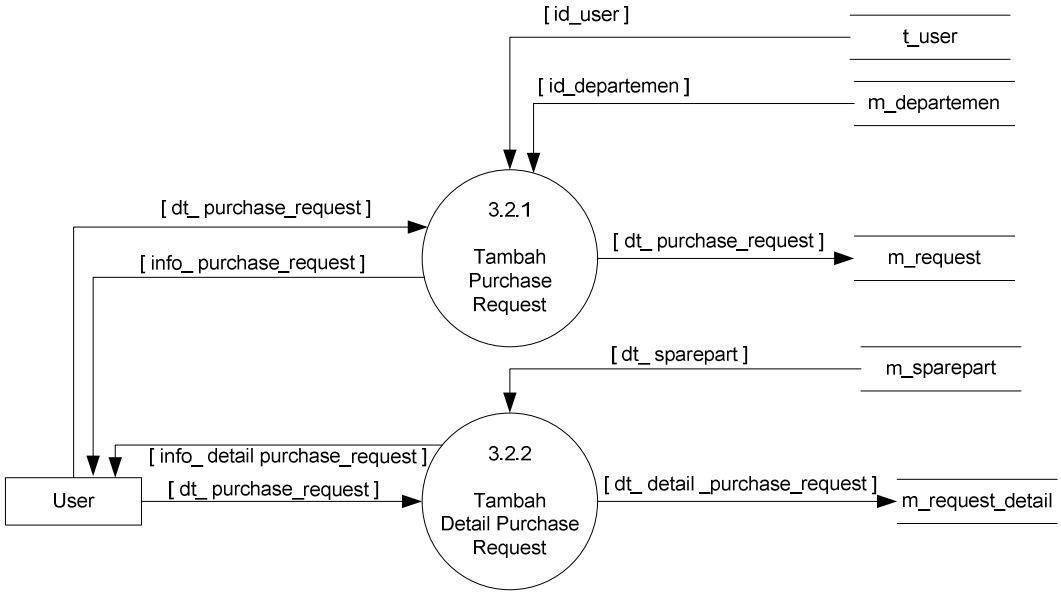
Diagram alir data pada proses data *store request* adalah proses untuk mengolah data *store request* seperti tambah data *store request* dan tambah data *detail store request*. Seperti pada gambar 3.21



Gambar 3.21 DFD Level 3 Proses 3.1 *Store Request*

3.1.5.2.3.7 DFD Level 3 Proses 3.2 *Purchase Request*

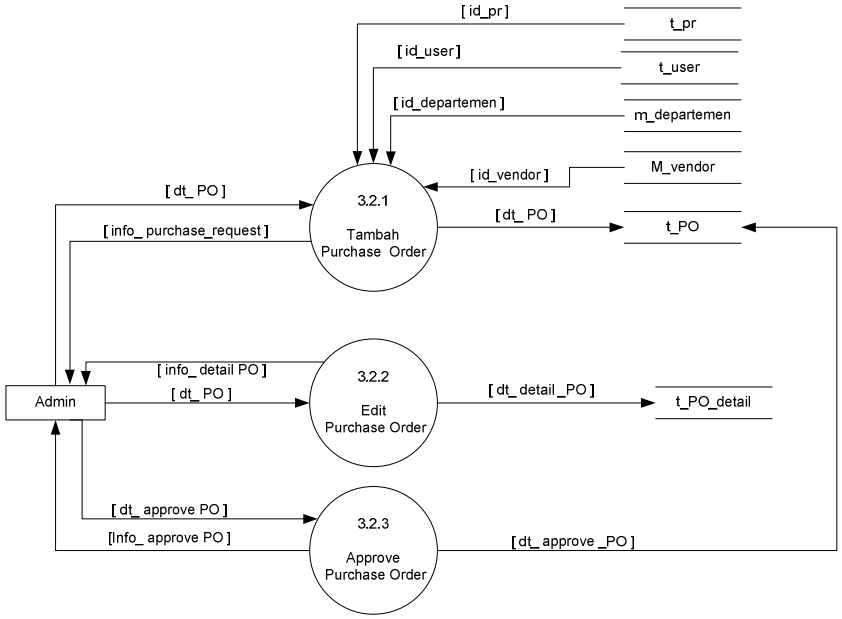
Diagram alir data pada proses data *purchase request* adalah proses untuk mengolah data *store request* seperti tambah data *purchase request* dan tambah data *detail purchase request*. Seperti pada gambar 3.22



Gambar 3.22 DFD Level 3 Proses 3.2 *Purchase Request*

3.1.5.2.3.8 DFD Level 3 Proses 3.3 *Purchase Order*

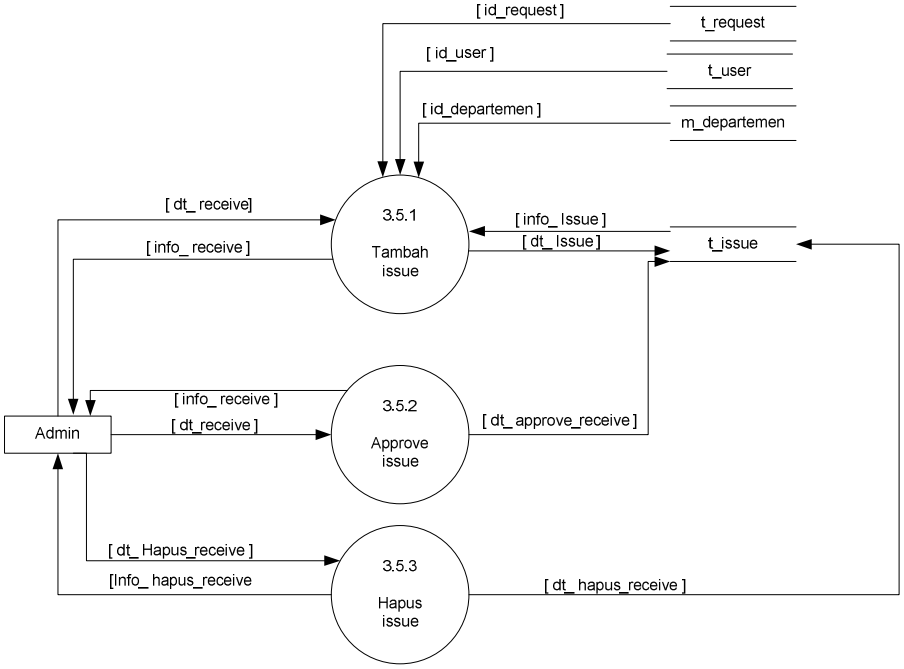
Diagram alir data pada proses data *purchase order* adalah proses untuk mengolah data *purchase order* seperti tambah data *purchase order* dan tambah data *detail purchase order*. Seperti pada gambar 3.23



Gambar 3.23 DFD Level 3 Proses 3.3 *Purchase Order*

3.1.5.2.3.9 DFD Level 3 Proses 3.4 *Issue*

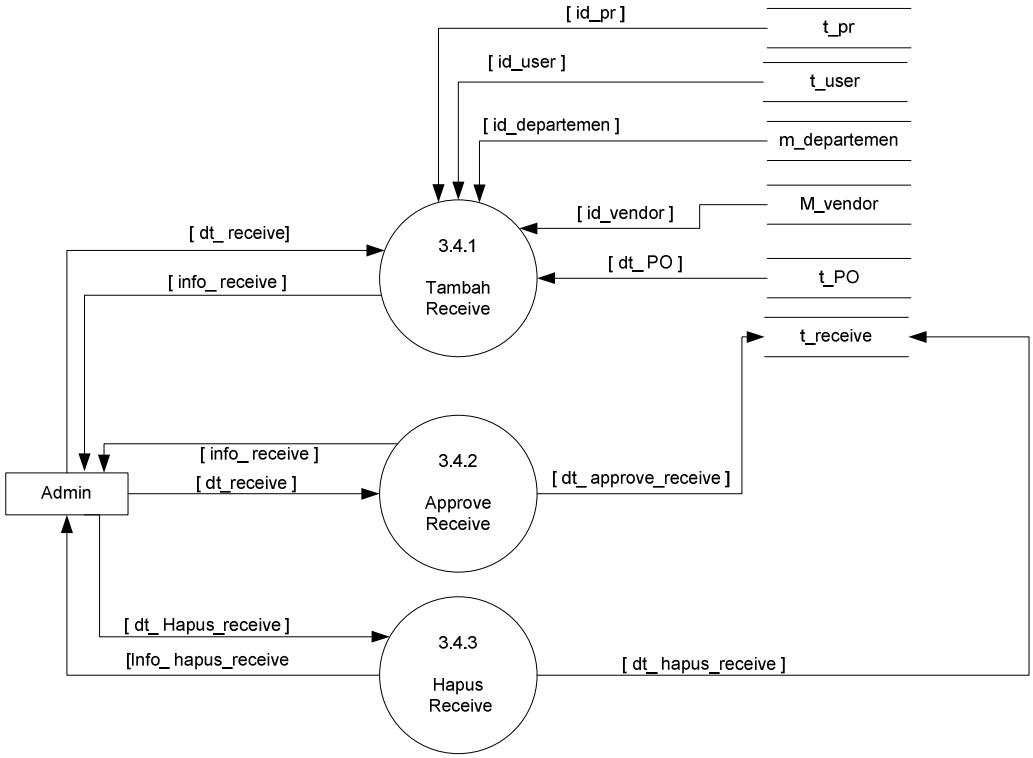
Diagram alir data pada proses data *issue* adalah proses untuk mengolah data *issue* atau pengeluaran *sparepert* seperti tambah data *issue* dan ubah status *issue*. Seperti pada gambar 3.24



Gambar 3.24 DFD Level 3 Proses 3.4 *Issue*

3.1.5.2.3.10 DFD Level 3 Proses 3.5 *Receive*

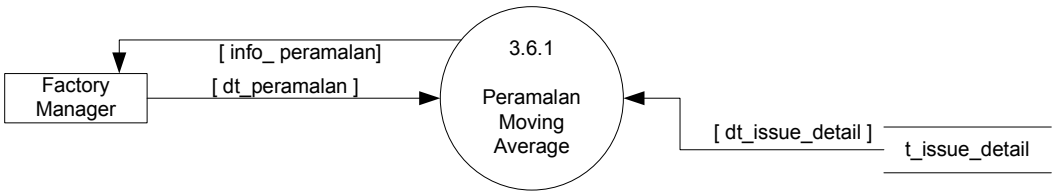
Diagram alir data pada proses data *Receive* adalah proses untuk mengolah data *receive* atau penerimaan *sparepart* masuk seperti tambah data *receive* dan ubah status *receive*. Seperti pada gambar 3.25



Gambar 3.25 DFD Level 3 Proses 3.5 *Receive*

3.1.5.2.3.11 DFD Level 3 Proses 3.6 Peramalan Kebutuhan *Sparepart*

Diagram alir data pada proses peramalan kebutuhan *sparepart* adalah proses untuk mengolah data peramalan kebutuhan *sparepart*. Seperti pada gambar 3.26



Gambar 3.26 DFD Level 3 Proses 3.6 Peramalan Kebutuhan *Sparepart*

3.1.5.3 Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD. Spesifikasi proses dari gambaran DFD akan dijelaskan pada tabel 3.8

Tabel 3.8 Spesifikasi Proses

No	Proses	Keterangan
1	No. Proses	1
	Nama Proses	*Login*
	Sumber	Admin, Staff Ekspedisi, Staff *Purchase*, Staff Gudang *sparepart*, *Factory Manager*
	Input	Data admin, data staff ekspedisi, data staff *Purchase*, Data Staff Gudang *Sparepart*, dan data *Factory Manager*
	Output	Info *valid*
	Tujuan	Admin, Staff Ekspedisi, Staff *Purchase*, Staff Gudang *sparepart*, *Factory Manager*
	Logika	1. *User* mengisi *username* dan *password*. 2. Jika benar *user* dapat masuk ke proses yang lain. 3. Jika salah muncul peringatan
2	No. Proses	2.1.1
	Nama Proses	Tambah *sparepart*
	Sumber	Admin, staff gudang *spareparat*, staff *purchase*, staff ekspedisi, *factory manager*
	Input	Data *sparepert*
	Output	Data admin, data staff ekspedisi, data staff *Purchase*, Data Staff Gudang *Sparepart*, dan data *Factory Manager*
	Tujuan	Admin, staff gudang *spareparat*, staff *purchase*, staff ekspedisi, *factory manager*
	Logika	1. Admin mengisikan data *sparepart* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel m_sparapart. 3. Jika data salah maka akan muncul peringatan
2	No. Proses	2.1.2
	Nama Proses	Edit *sparepart*
	Sumber	Admin
	Input	Data ubah *sparepart*
	Output	Info *sparepart* ubah
	Tujuan	admin

	Logika	1. Admin memilih data *sparepart* yang akan dirubah 2. Admin meng-*input*-kan data yang baru 3. Jika data benar maka akan disimoan ke tabel m_sparepart 4. Jika data salah maka akan muncul peringatan
3	No. Proses	2.1.3
	Nama Proses	Hapus data *sparepart*
	Sumber	Admin
	Input	Data *sparepart* hapus
	Output	Info *sparepart* hapus
	Tujuan	Admin
	Logika	1. Admin memilih data *sparepart* yang akan di hapus 2. Admin menghapus data *sparepart* yang ada di tabel m_sparepart.
4	No. Proses	2.2.1
	Nama Proses	Tambah jenis *sparepart*
	Sumber	Admin
	Input	Data jenis *sparepart* baru
	Output	Info jenis *sparepart* baru
	Tujuan	Admin
	Logika	1. Admin mengisikan data jenis *sparepart* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel m_jnssparapart. 3. Jika data salah maka akan muncul peringatan
5	No. Proses	2.2.2
	Nama Proses	Edit jenis *sparepart*
	Sumber	Admin
	Input	Data ubah jenis *sparepart*
	Output	Info jenis *sparepart* ubah
	Tujuan	Admin
	Logika	1. Admin memilih data jeni *sparepart* yang akan dirubah 2. Admin meng-*input*-kan data yang baru 3. Jika data benar maka akan disimoan ke tabel m_jnssparepart
6	No. Proses	2.2.3
	Nama Proses	Hapus data jenis *sparepart*

	Sumber	Admin
	Input	Data jenis *sparepart* hapus
	Output	Info jenis *sparepart* hapus
	Tujuan	Admin
	Logika	1. Admin memilih data jenis *sparepart* yang akan di hapus 2. Admin menghapus data jenis *sparepart* yang ada di tabel m_jnssparepart.
7	No. Proses	2.3.1
	Nama Proses	Tambah data *warehouse*
	Sumber	Admin
	Input	Data *warehouse* baru
	Output	Info data *warehouse* baru
	Tujuan	Admin
	Logika	1. Admin mengisikan data *warehouse* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel m_warehouse. 3. Jika data salah maka akan muncul peringatan
8	No. Proses	2.3.2
	Nama Proses	Edit data *warehouse*
	Sumber	Admin
	Input	Data ubah *warehouse*
	Output	Info ubah data *warehouse*
	Tujuan	Admin
	Logika	1. Admin memilih data *warehouse* yang akan dirubah 2. Admin meng-*input*-kan data yang baru 3. Jika data benar maka akan disimoan ke tabel m_warehouse
9	No. Proses	2.3.3
	Nama Proses	Hapus data *warehouse*
	Sumber	Admin
	Input	Data *warehouse* hapus
	Output	Info *warehouse* hapus
	Tujuan	Admin
	Logika	1. Admin memilih data *warehouse* yang akan di hapus

		2. Admin menghapus data *warehouse* yang ada di tabel m_warehouse.
10	No. Proses	2.4.1
	Nama Proses	Tambah jenis *warehouse*
	Sumber	Admin
	Input	Data jenis *warehouse* baru
	Output	Info jenis *warehouse* baru
	Tujuan	Admin
	Logika	1. Admin mengisikan data jenis *warehouse* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel m_jnswarehouse. 3. Jika data salah maka akan muncul peringatan
11	No. Proses	2.4.2
	Nama Proses	Edit jenis *warehouse*
	Sumber	Admin
	Input	Data ubah jenis *warehouse*
	Output	Info jenis *warehouse* ubah
	Tujuan	Admin
	Logika	1. Admin memilih data jeni *warehouse* yang akan dirubah 2. Admin meng-*input*-kan data yang baru 3. Jika data benar maka akan disimoan ke tabel m_jnswarehouse
12	No. Proses	2.4.3
	Nama Proses	Hapus data jenis *warehouse*
	Sumber	Admin
	Input	Data jenis *warehouse* hapus
	Output	Info jenis *warehouse* hapus
	Tujuan	Admin
	Logika	1. Admin memilih data jenis *warehouse* yang akan di hapus 2. Admin menghapus data jenis *warehouse* yang ada di tabel m_jnswarehouse
13	No. Proses	2.5.1
	Nama Proses	Tambah data *vendor*
	Sumber	Admin
	Input	Data *Vendor* baru

	Output	Info data *Vendor* baru
	Tujuan	Admin
	Logika	1. Admin mengisikan data *vendo* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel m_vendor. 3. Jika data salah maka akan muncul peringatan
14	No. Proses	2.5.2
	Nama Proses	Edit data *vendor*
	Sumber	Admin
	Input	Data edit data *vendor*
	Output	Info data *vendor* ubah
	Tujuan	Admin
	Logika	1. Admin memilih data *vendor* yang akan dirubah 2. Admin meng-*input*-kan data yang baru 3. Jika data benar maka akan disimoan ke tabel m_vendor
15	No. Proses	2.5.3
	Nama Proses	Hapus data *vendor*
	Sumber	Admin
	Input	Data *vendor* hapus
	Output	Info data *vendor* hapus
	Tujuan	Admin
	Logika	1. Admin memilih data *vendor* yang akan di hapus 2. Admin menghapus data *vendor* yang ada di tabel m_vendor.
16	No. Proses	3.1.1
	Nama Proses	Tambah *Store request*
	Sumber	Admin / Staff Gudang *sparepart*
	Input	Data *store request* baru
	Output	Info *store request* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *sparepart* mengisikan data *store request* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_request. 3. Jika data salah maka akan muncul peringatan

17	No. Proses	3.1.2
	Nama Proses	Tambah detail *Store request*
	Sumber	Admin / Staff Gudang *sparepart*
	Input	Data detail *store request* baru
	Output	Info detail *store request* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *sparepart* mengisikan data detail *store request* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_detail request. 3. Jika data salah maka akan muncul peringatan
18	No. Proses	3.2.1
	Nama Proses	Tambah *Purchase request*
	Sumber	Admin / Staff Gudang *sparepart*
	Input	Data *purchase request* baru
	Output	Info *purchase request* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *sparepart* mengisikan data *purchase request* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_pr. 3. Jika data salah maka akan muncul peringatan
19	No. Proses	3.2.2
	Nama Proses	Tambah detail *purchase request*
	Sumber	Admin / Staff Gudang *sparepart*
	Input	Data detail *purchase request* baru
	Output	Info detail *purchase request* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *sparepart* mengisikan data detail *purchase request* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_pr_detail. 3. Jika data salah maka akan muncul peringatan

20	No. Proses	3.3.1
	Nama Proses	Tambah *Purchase Order*
	Sumber	Admin / Staff *Purchase*
	Input	Data *Purchase Order* baru
	Output	Info data *Purchase Order* baru
	Tujuan	Admin, staff gudang *purchase*
	Logika	1. Admin / Staff *Purchase* mengisikan data *Purchase Order* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_PO. 3. Jika data salah maka akan muncul peringatan 4. Admin / Staff *Purchase* memilih kode *purchase request*, jika data yang di masukan benar maka akan disimpan di tabel t_PO_detail
21	No. Proses	3.3.2
	Nama Proses	*Approve Purchase Order*
	Sumber	*Factory Manager*
	Input	Status *Purchase Order*
	Output	Info status *Purchase Order*
	Tujuan	*Factory manager*
	Logika	1. *Factory Manager* akan memilih status *Purchase Order* 2. Jika status *purchase order* sudah di-*approve* maka akan muncul di proses *receive*.
22	No. Proses	3.3.3
	Nama Proses	Hapus *Purchase Order*
	Sumber	Admin / Staff *Purchase*
	Input	Data *Purchase Order* hapus
	Output	Info data *Purchase Order* hapus
	Tujuan	Admin, staff gudang *purchase*
	Logika	1. Admin / Staff *Purchase* akan memilih data *Purchase Order* yang akan di hapus 2. Admin / Staff *Purchase* menghapus data *purchase order* yang ada di tabel t_po dan t_po_detail
23	No. Proses	3.4.1
	Nama Proses	Tambah data *receive*

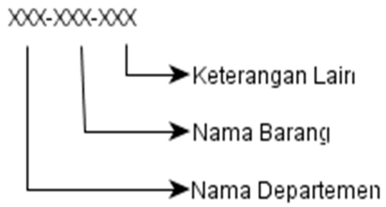
	Sumber	Admin / Staff Gudang *Sparepart*
	Input	Data *receive* baru
	Output	Info data *receive* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *Sparepart* mengisikan data *receive* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_receipt. 3. Jika data salah maka akan muncul peringatan 4. Admin / Admin, staff gudang *sparepart* memilih kode *purchase order,* jika data yang di masukan benar maka akan disimpan di tabel t_ *receipt* _detail
24	No. Proses	3.4.2
	Nama Proses	*Approve receive*
	Sumber	Admin / Staff Gudang *Sparepart*
	Input	Status *receive*
	Output	Info status *receive*
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff *Purchase* akan memilih status *receive* 2. Jika status *purchase order* sudah di-*approve* maka stok *sparepart* akan bertambah
25	No. Proses	3.4.3
	Nama Proses	Hapus *receive*
	Sumber	Admin / Staff Gudang *Sparepart*
	Input	Data *receive* hapus
	Output	Info data *receive* hapus
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff *Purchase* akan memilih data *receive* yang akan di hapus 2. Admin / Staff *Purchase* menghapus data *purchase order* yang ada di tabel t_receipt dan t_ receipt_detail
26	No. Proses	3.5.1
	Nama Proses	Tambah data *issue*
	Sumber	Admin / Staff Gudang *Sparepart*

	Input	Data *issue* baru
	Output	Info data *issue* baru
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *Sparepart* mengisikan data *issue* baru 2. Jika data yang dimasukan benar maka data akan disimpan dalam tabel t_issue. 3. Jika data salah maka akan muncul peringatan 4. Admin, staff gudang *sparepart* memilih kode *store request*, jika data yang di masukan benar maka akan disimpan di tabel t_issue_detail
21	No. Proses	3.5.2
	Nama Proses	*Approve issue*
	Sumber	Admin / Staff Gudang *Sparepart*
	Input	Status *issue*
	Output	Info status *issue*
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *Sparepart* akan memilih status *issue* 2. Jika status *issue* sudah di-*approve* maka stok *sparepart* akan langsung berkurang
22	No. Proses	3.5.3
	Nama Proses	Hapus *issue*
	Sumber	Admin / Staff Gudang *Sparepart*
	Input	Data *issue* hapus
	Output	Info data *issue* hapus
	Tujuan	Admin, staff gudang *sparepart*
	Logika	1. Admin / Staff Gudang *Sparepart* akan memilih data *issue* yang akan di hapus 2. Admin / Staff Gudang *Sparepart* menghapus data *issue* yang ada di tabel t_issue dan t_issue_detail
23	No. Proses	3.6.1
	Nama Proses	Perhitungan Peramalan
	Sumber	sistem
	Input	Info data *issue_detail*

	Output	Data hasil perhitungan peramalan
	Tujuan	*Factory manager*
	Logika	1. Dari *issue* dan *issue* detail di dapat info *sparepart* keluar yang akan dimasukan ke dalam rumus peramalan kebutuhan 2. *Factory manager* memilih *sparepart* yang akan diramalkan 3. Didapatkan hasil peramalan kebutuhan *sparepart*

3.1.5.4 Analisis Pengkodean

Pengkodean ini berfungsi untuk mendefinisikan suatu objek secara singkat, mengklarifikasikan data dan masukan data, selain itu kode dapat dibentuk dari kumpulan huruf, angka, dan karakter khusus. Pada program sistem informasi ini terdapat no urut pengkodeanyaitu sebagai berikut :



Contoh : LISELC035

LISELC35 : Listrik Elco 35Ω

3.2 Perancangan Sistem

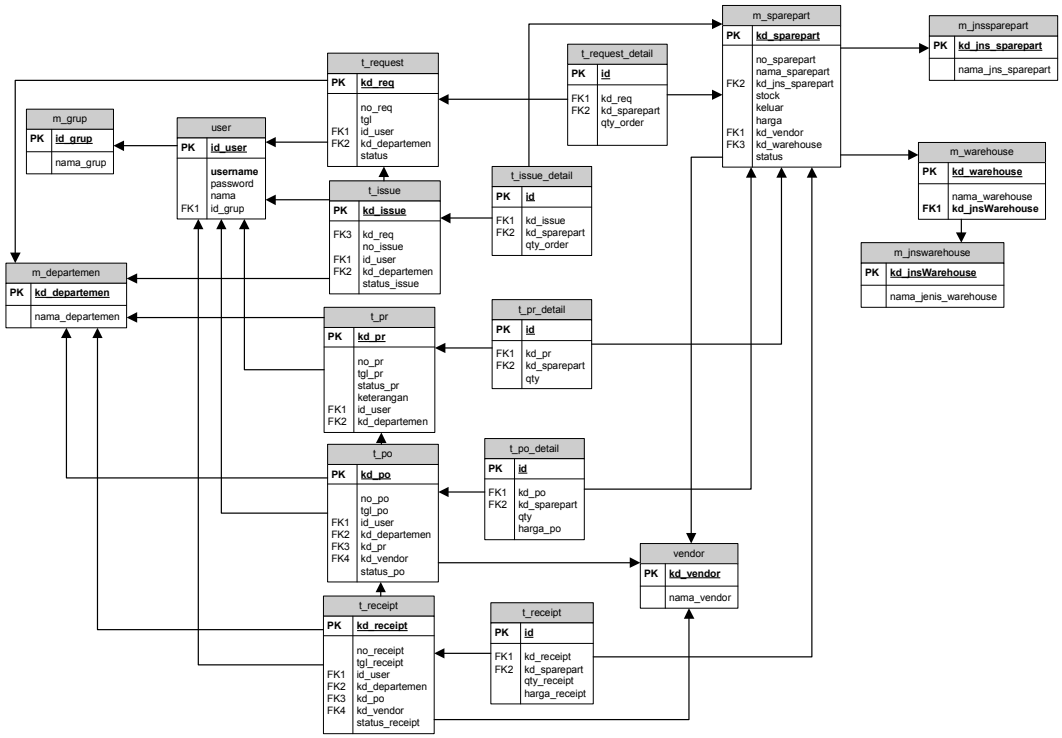
3.2.1 Perancangan Basis Data

Pada bagian ini dijelaskan perancangan suatu basis data (Database) yang akan dirancang pada sistem yang akan dibuat. Basis data yang akan dirancangan berupa tabel relasi dan struktur file

3.2.1.1 Skema Relasi

Untuk lebih mudah dalam menganalisis dan merancangan sistem maka dibuat suatu diagram yang menggambarkan hubungan antara file yang saling berhubungan. Adapun tabel relasi atau relasi file dari rancangan sistem ini dapat digambarkan sebagai berikut :

.



Gambar 3.27 Skema Relasi

3.2.1.2 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam sistem ini adalah sebagai berikut :

1. Tabel User

Tabel user digunakan untuk menyimpan data id, nama_user dan password. Sebagai acuan dari validasi untuk enkripsi pada form administrator. Rancangan tabel admin dapat di lihat pada tabel 3.9

Tabel 3.9 Tabel User

No	Nama File	Tipe Data	Keterangan
1.	id_user	Int (3)	Primary key (autoincrement)
2.	username	Varchar(12)	
3.	password	varchar (32)	

4	id_grup	Int(11)	foreign key
5	nama	Varchar (30)	

2. Tabel Grup

Tabel grup digunakan untuk menyimpan data grup. Rancangan tabel grup dapat di lihat pada tabel 3.10

Tabel 3.10 Tabel Grup

No	Nama File	Tipe Data	Keterangan
1	id_grup	Int(11)	Primary key (autoincrement)
2	nama_grup	Varchar(30)	

3. Tabel Jenis *Spare Part*

Tabel jenis *spare part* digunakan untuk menyimpan data jenis *spare part*. Rancangan tabel jenis *spare part* dapat di lihat pada tabel 3.11

Tabel 3.11 Tabel Jenis *Sparepart*

No	Nama File	Tipe Data	Keterangan
1	kd_jns_sparepart	Int(11)	Primary key (autoincrement)
2	nama_jns_sparepart	Varchar(20)	

4. Tabel Jenis Warehouse

Tabel jenis *warehaouse* di gunakan untuk menyimpan data *warehouse*. Rancangan tabel jenis *warehouse* di gambarkan pada tabel 3.12

Tabel 3.12 Tabel Jenis Warehouse

No	Nama File	Data Type	Index
1	kd_jns_warehouse	Int(11)	Primary key (autoincrement)
2	nama_jns_warehouse	Varchar(30)	

5. Tabel Warehouse

Tabel *warehouse* digunakan untuk menyimpan data *warehouse*. Rancangan tabel *warehouse* dapat di lihat pada tabel 3.13

Tabel 3.13 Tabel Warehouse

No	Nama File	Tipe Data	Keterangan
1	kd_warehouse	Int(11)	Primary key (autoincrement)
2	nama_warehouse	Varchar(20)	
3	kd_jns_warehouse	Int(20)	Foreign Key

6. Tabel Vendor

Tabel vendor di gunakan untuk menyimpan data vendor. Rancangan tabel vendor di gambarkan pada tabel 3.14

Tabel 3.14 Tabel Vendor

No	Nama File	Tipe Data	Keterangan
1	kd_vendor	Int(11)	Primary key (autoincrement)
2	nama_vendor	Varchar(30)	

7. Tabel Departemen

Tabel Departemen di gunakan untuk menyimpan data departemen. Rancangan tabel departmen di gambarkan pada tabel 3.15

Tabel 3.15 Tabel Departemen

No	Nama File	Tipe Data	Keterangan
1	kd_departemen	Int(11)	Primary key (autoincrement)
2	nama_departemen	Varchar(30)	

8. Tabel *Purchase Request*

Tabel Mesin di gunakan untuk menyimpan data mesin Rancangan tabel mesin di gambarkan pada tabel 3.16

Tabel 3.16 Tabel *Purchase Request*

No	Nama File	Tipe Data	Keterangan
1	kd_pr	Int(11)	Primary key (autoincrement)
2	no_pr	Varchar(12)	
3	id_user	Varchar(3)	foreign key
4	kd_departemen	Varchar(11)	foreign key
5	tgl_pr	datetime	
6	status	Varchar(30)	
7	keterangan	Varchar(500)	

9. Tabel Sparepart

Tabel Sparepart digunakan untuk menyimpan data sparepart . Rancangan tabel *sparepart* dapat di lihat pada tabel 3.17

Tabel 3.17 Tabel Sparepart

No	Nama File	Tipe Data	Keterangan
1	kd_sparepart	Int(11)	Primary key (autoincrement)
2	no_sparepart	Varchar(30)	
3	nama_sparepart	Varchar(30)	
4	kd_jns_sparepart	Int(11)	foreign key
5	stock	Int(11)	
6	harga	Double	
7	kd_vendor	Int(11)	foreign key
8	kd_warehouse	Int(11)	foreign key
9	status	Int(11)	
10	keluar	Int(11)	

10. Tabel Request

Tabel Request atau pemohonan digunakan untuk menyimpan data request. Rancangan data permohonan dapat di lihat pada tabel 3.18

Tabel 3.18 Tabel Request

No	Nama File	Tipe Data	Keterangan
1	kd_req	Int(11)	Primary key (autoincrement)
2	no_req	Varchar(30)	
3	tgl	Date	
4	id_user	Int(30)	foreign key
5	kd_departemen	Int(11)	foreign key
6	status	Varchar(15)	

11. Tabel Request Detail

Tabel Request Detail digunakan untuk menyimpan request detail. Perancangan tabel Request Detail digambarkan pada Tabel 3.19

Tabel 3.19 Tabel Request Detail

No	Nama File	Tipe Data	Keterangan
1	id	Int(11)	Primary key (autoincrement)
2	kd_req	Int(11)	foreign key
3	kd_sparepart_out	Int(11)	foreign key
4	qty_order	Int	

12. Tabel Purchase

Tabel Purchase digunakan untuk menyimpan data-data purchase. Perancangan tabel penyerahan digambarkan pada Tabel 3.20

Tabel 3.20 Tabel Purchase

No	Nama File	Tipe Data	Keterangan
1	kd_po	Int(11)	Primary key (autoincrement)
2	no_po	Varchar(30)	
3	tgl_po	Datetime	
4	id_user	Int(11)	foreign key
5	kd_departemen	Int(11)	foreign key
6	kd_pr	Int(11)	foreign key
7	kd_vendor	Int(11)	
8	status_po	Varchar(12)	

13. Tabel Purchase Detail

Tabel Purchase Detail digunakan untuk menyimpan data-data purchase detail. Perancangan tabel penyerahan digambarkan pada Tabel 3.21

Tabel 3.21 Tabel Purchase Detail

No	Nama File	Tipe Data	Keterangan
1	id	Int(11)	Primary key (autoincrement)
2	kd_po	Varchar(20)	foreign key
3	kd_sparepart	Int(11)	foreign key
4	qty	Int(4)	
5	harga_po	Int(11)	

14. Tabel Receipt

Tabel Receipt digunakan untuk menyimpan data-data receipt. Perancangan tabel penyerahan digambarkan pada Tabel 3.22

Tabel 3.22 Tabel Receipt

No	Nama File	Tipe Data	Keterangan
1	kd_receipt	Int(11)	Primary key (autoincrement)
2	no_receipt	Varchar(30)	
3	tgl_receipt	Datetime	
4	id_user	Int(11)	foreign key
5	kd_departemen	Int(11)	foreign key
6	kd_po	Int(11)	foreign key
7	kd_vendor	Int(11)	foreign key
8	status_receipt	Varchar(30)	

15. Tabel Receipt Detail

Tabel Receipt Detail digunakan untuk menyimpan data-data receipt detail. Perancangan tabel penyerahan digambarkan pada Tabel 3.23

Tabel 3.23 Tabel Receipt Detail

No	Nama File	Tipe Data	Keterangan
1	id	Int(11)	Primary key (autoincrement)
2	kd_receipt	Int(11)	foreign key
3	kd_sparepart	Int(11)	foreign key
4	qty	Int(11)	
5	harga_receipt	Int(11)	

16. Tabel Issue

Tabel Issue digunakan untuk menyimpan data-data issue. Perancangan tabel penyerahan digambarkan pada Tabel 3.24

Tabel 3.24 Tabel Issue

No	Nama File	Tipe Data	Keterangan
1	kd_issue	Int(11)	Primary key (autoincrement)
2	kd_req	Int(11)	foreign key
3	nomor_issue	Varchar(30)	
4	tgl_issue	date	
5	id_user	Int(11)	foreign key
6	kd_departemen	Int(11)	foreign key
7	status_issue	Varchar(13)	

17. Tabel Issue Detail

Tabel Issue Detail digunakan untuk menyimpan data-data issue detail. Perancangan tabel penyerahan digambarkan pada Tabel 3.25

Tabel 3.25 Tabel Issue Detail

No	Nama File	Tipe Data	Keterangan
1	id	Int(11)	Primary key (autoincrement)
2	kd_issue	Int(11)	foreign key
3	kd_sparepart	Int(11)	foreign key
4	qty	Int(11)	

18. Tabel Pr_detail

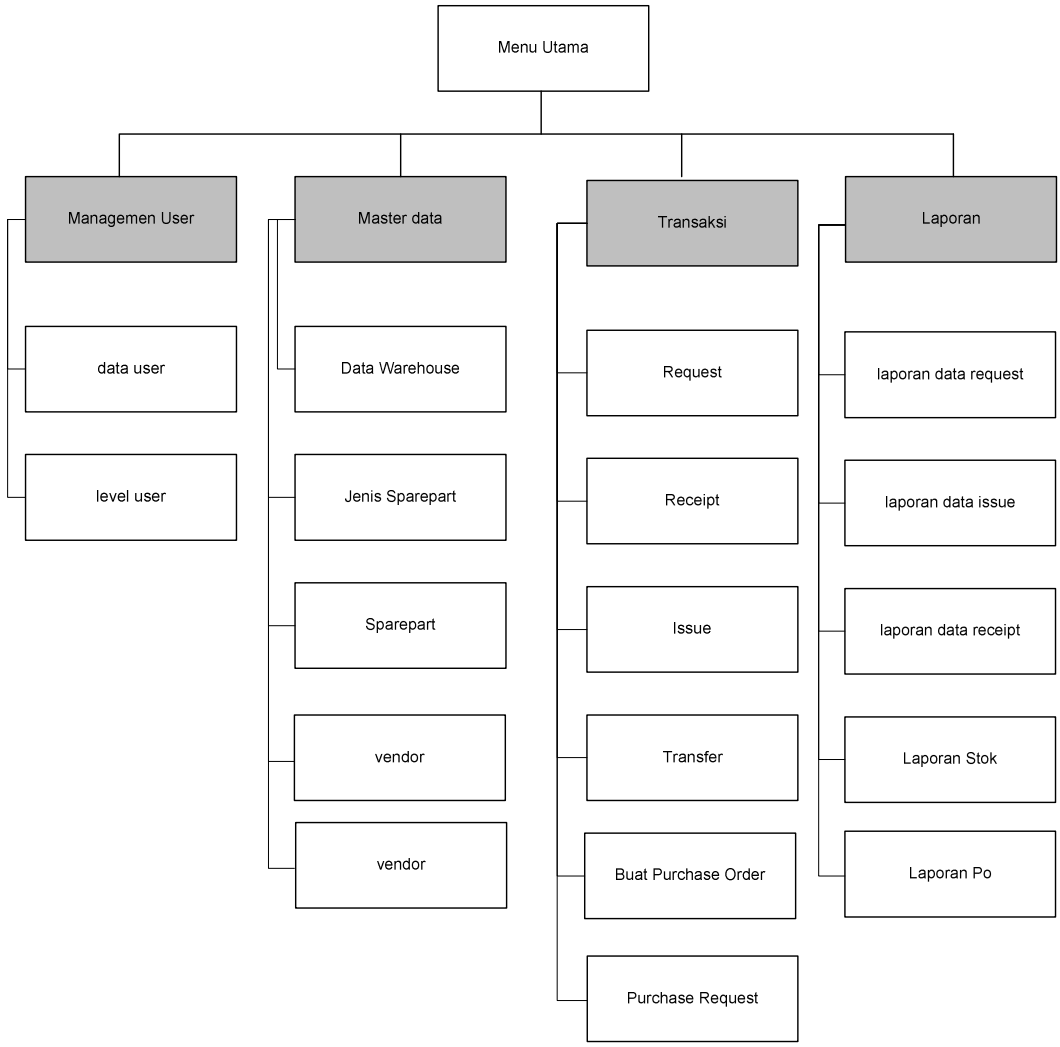
Tabel pr_ Detail digunakan untuk menyimpan data-data *purchase request* detail. Perancangan tabel penyerahan digambarkan pada Tabel 3.27

Tabel 3.26 Tabel Pr Detail

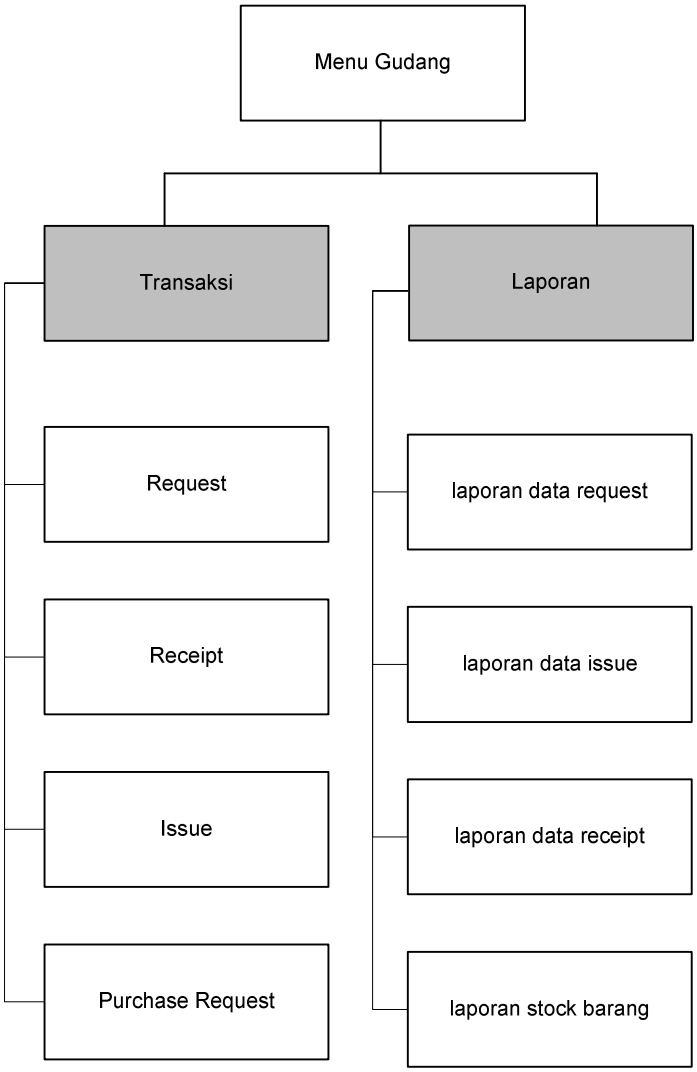
No	Nama File	Tipe Data	Keterangan
1	id	Int(11)	Primary key (autoincrement)
2	kd_pr	Int(11)	foreign key
3	kd_sparepart	Int(11)	foreign key
4	qty	Int(4)	

3.2.2 Perancangan Struktur Mengelola

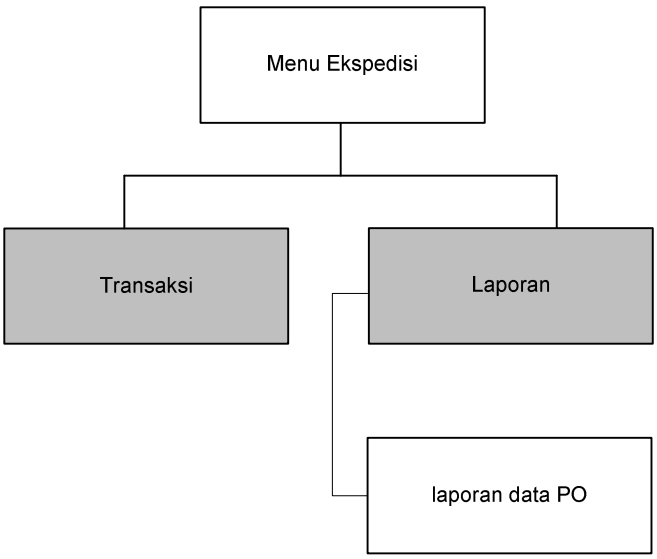
Perancangan Arsitektur Struktur Mengelola



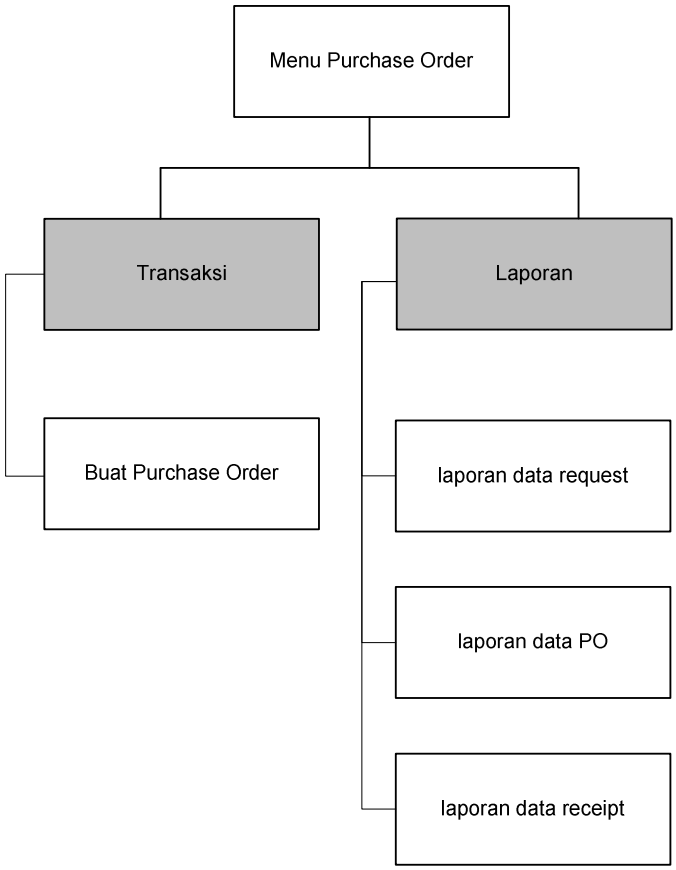
Gambar 3.28 Sturktur Mengelola Level Admin



Gambar 3.29 Sturktur Mengelola Level Gudang



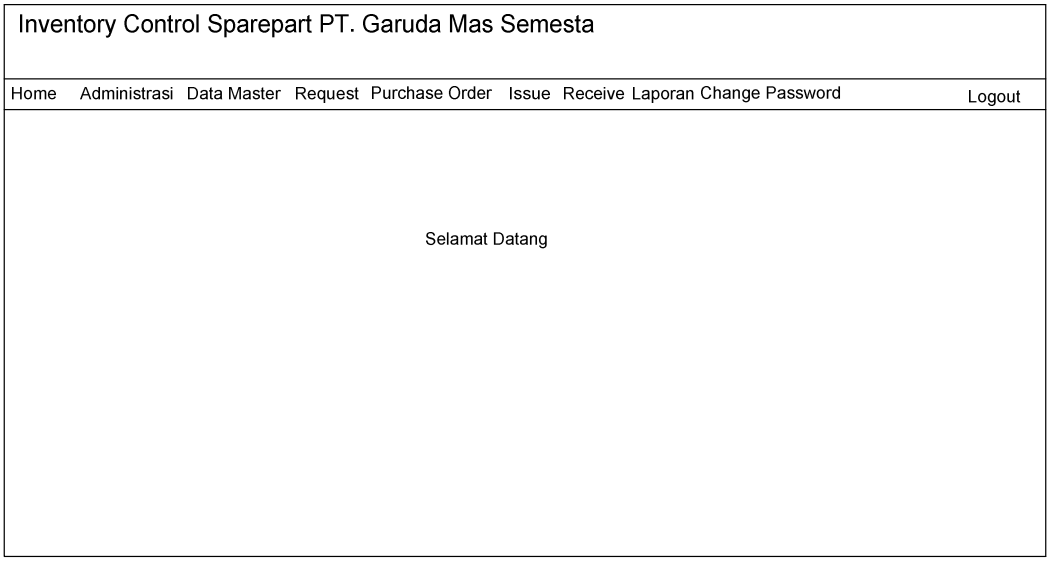
Gambar 3.30 Sturktur Mengelola Level Ekspedisi



Gambar 3.31 Sturktur Mengelola Level Purchase

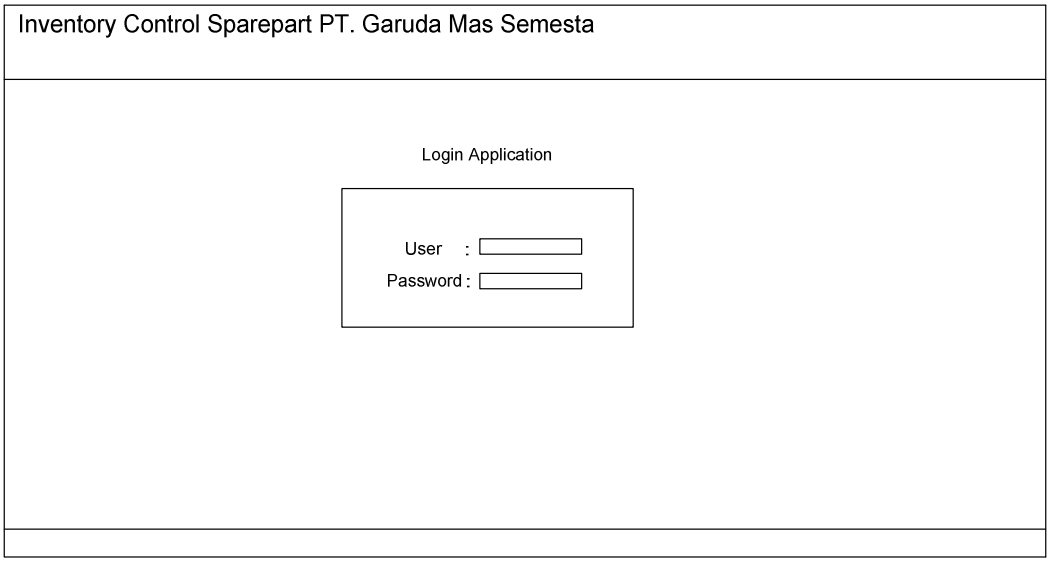
3.2.3 Perancangan Antar Muka

Perancangan antarmuka adalah perancangan tampilan yang dibuat sebelum aplikasi dibuat, perancangan antarmuka sistem informasi pengendalian stok *spare part* adalah sebagai berikut :



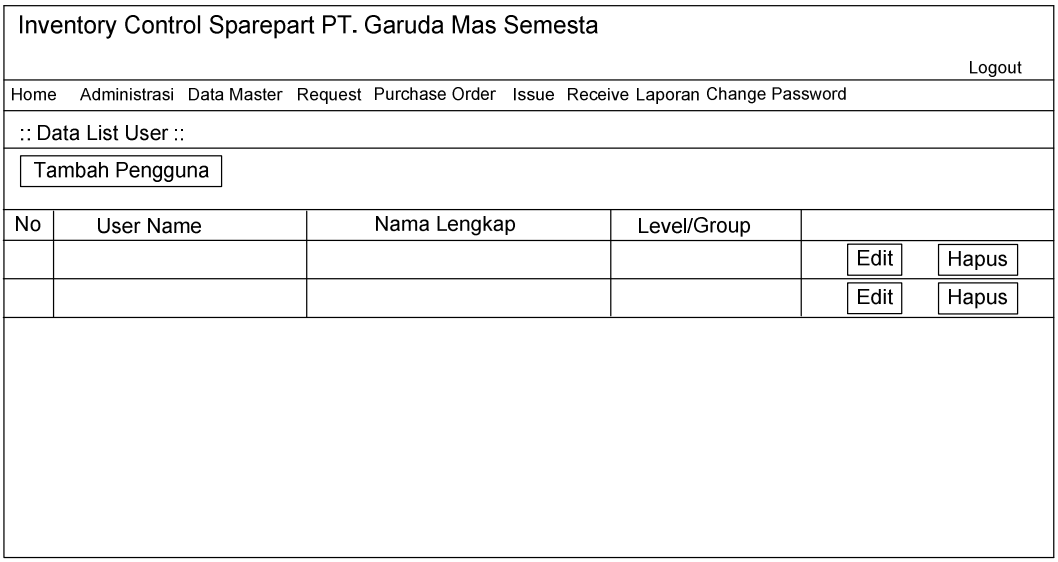
Gambar 3.32 Mengelola Utama

3.2.3.1 Tampilan Login



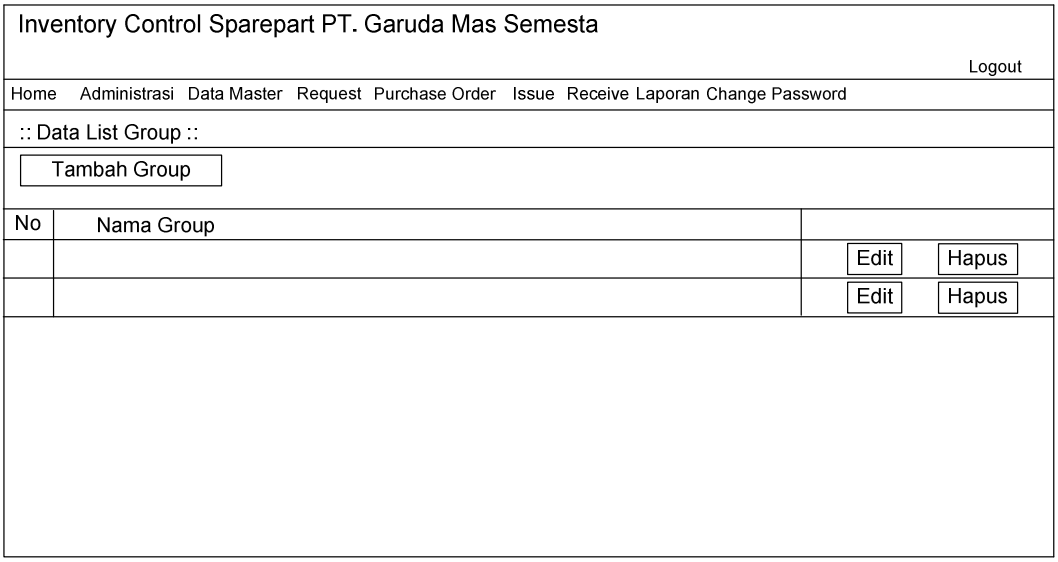
Gambar 3.33 Login

3.2.3.2 Tampilan Data User



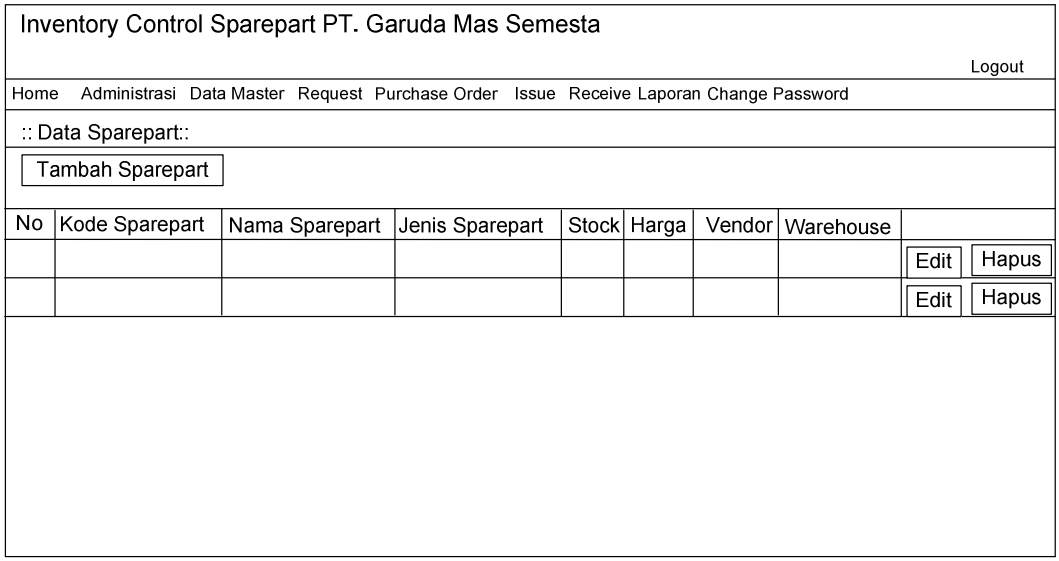
Gambar 3.34 Data User

3.2.3.3 Tampilan Data Level User / Group



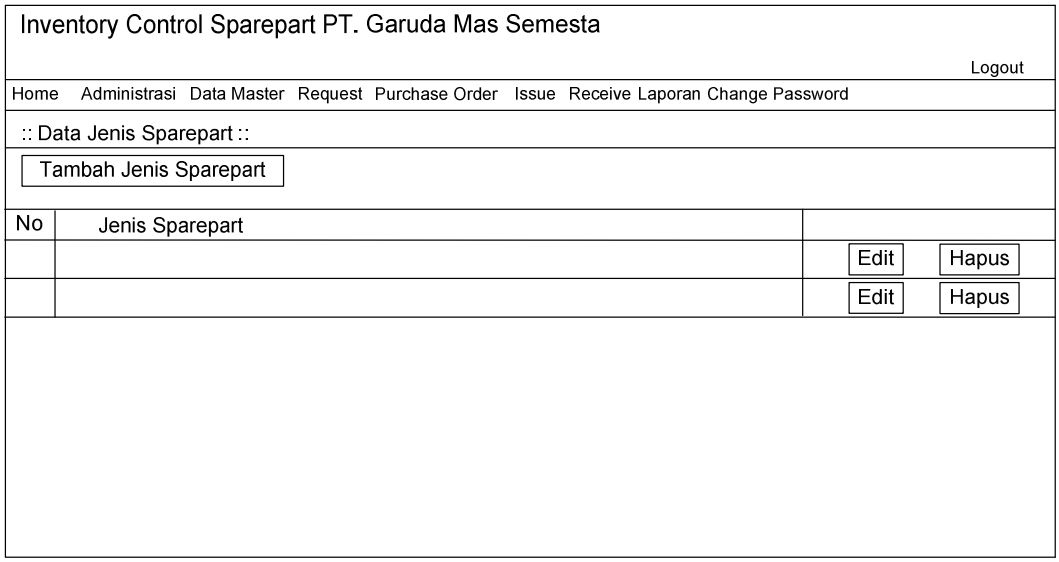
Gambar 3.35 Data Level/Group

3.2.3.4 Tampilan Data Barang / Sparepart



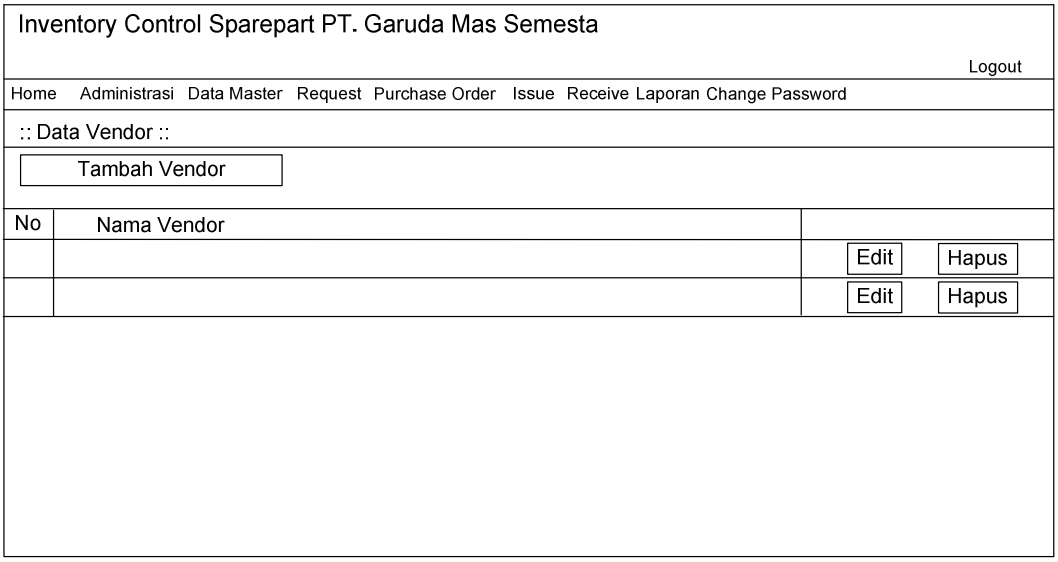
Gambar 3.36 Data Barang/Sparepart

3.2.3.5 Tampilan Data Jenis Barang / Sparepart



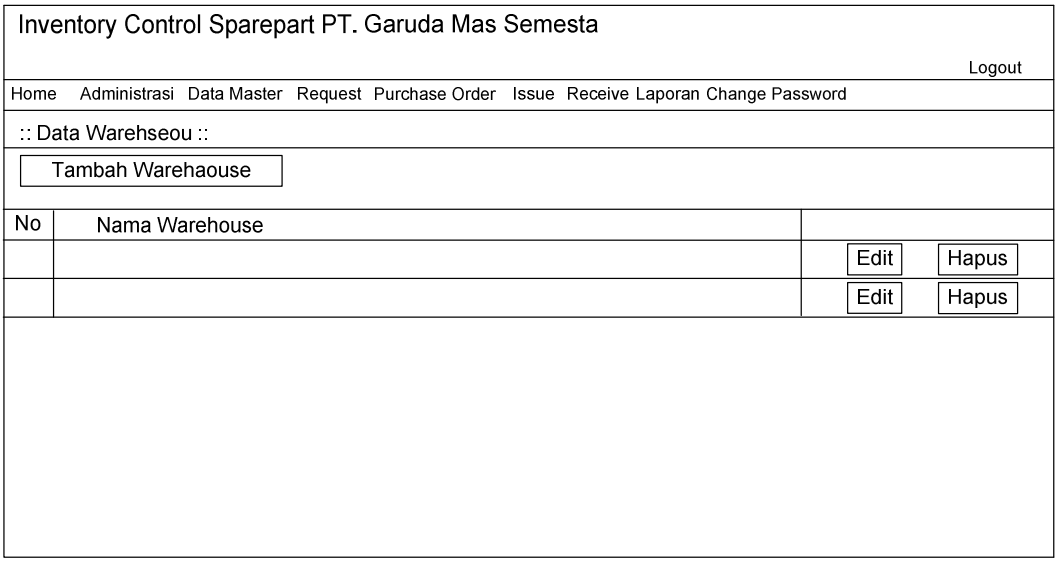
Gambar 3.37 Data Jenis Barang/Sparepart

3.2.3.6 Tampilan Data Vendor



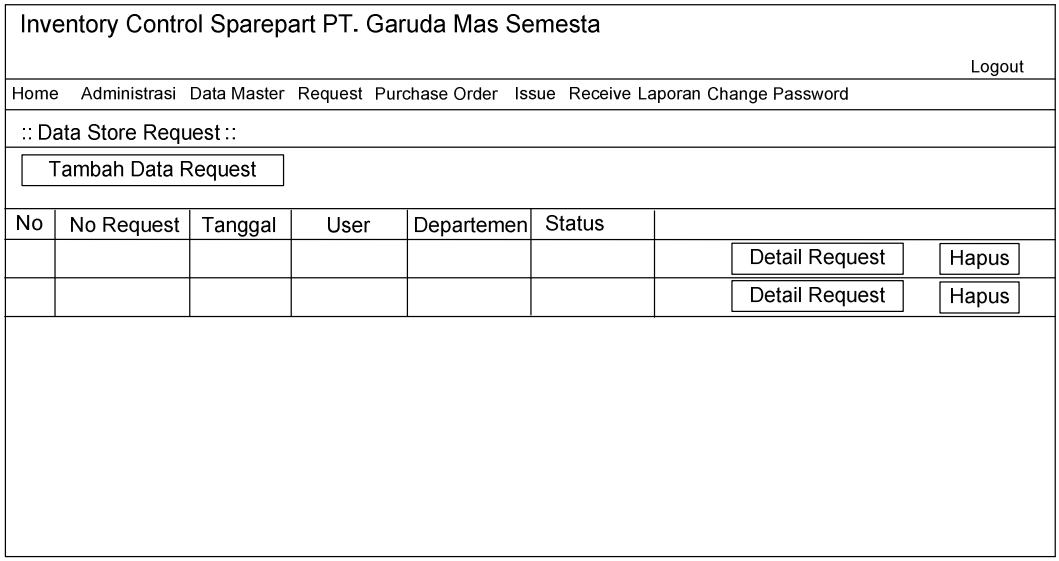
Gambar 3.38 Data Vendor

3.2.3.7 Tampilan Data Warehouse



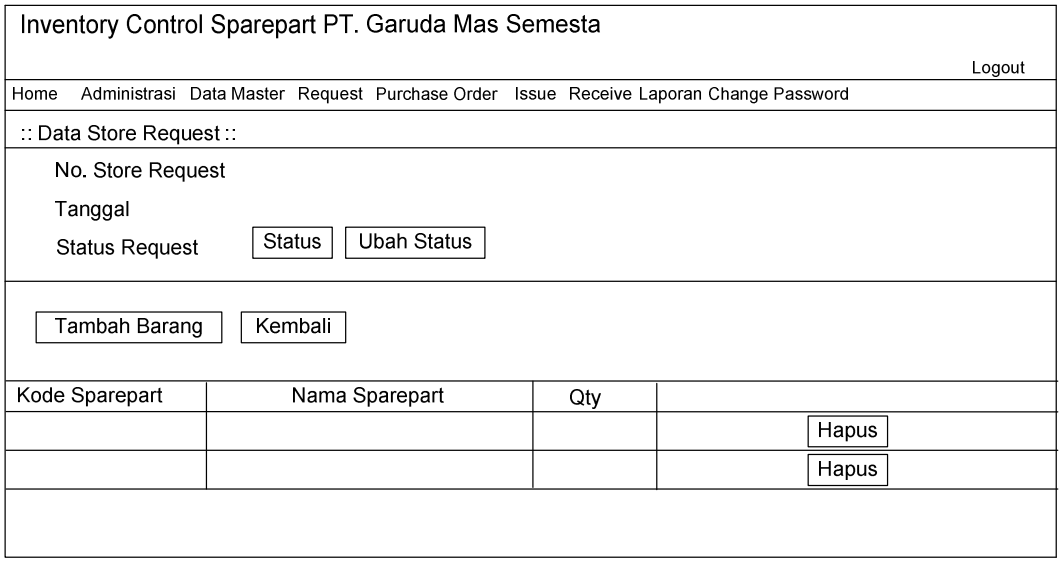
Gambar 3.39 Data Warehouse

3.2.3.8 Tampilan Data Store Request



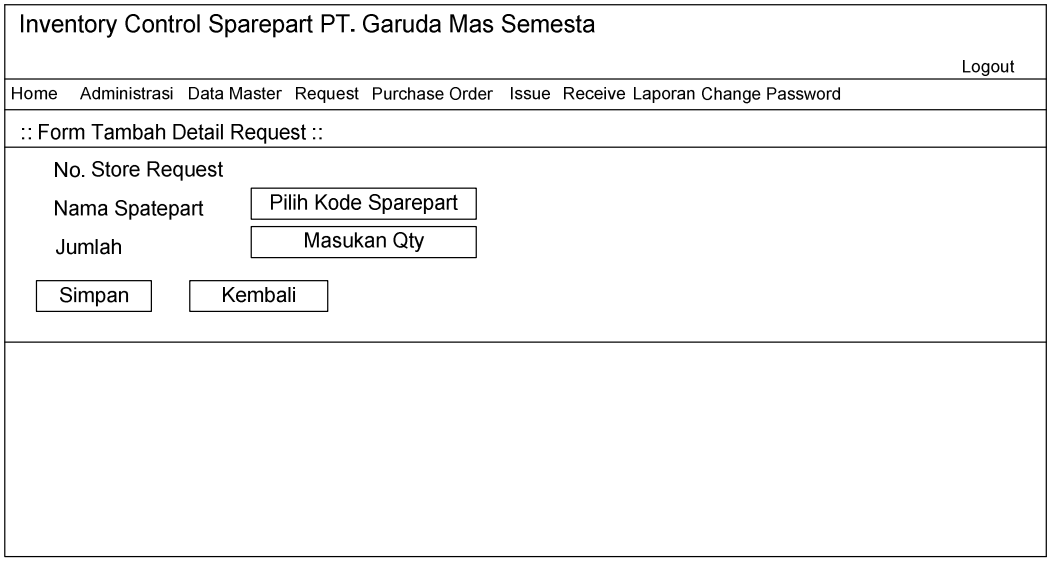
Gambar 3.40 Data Store Request

3.2.3.9 Tampilan Data Detail Request



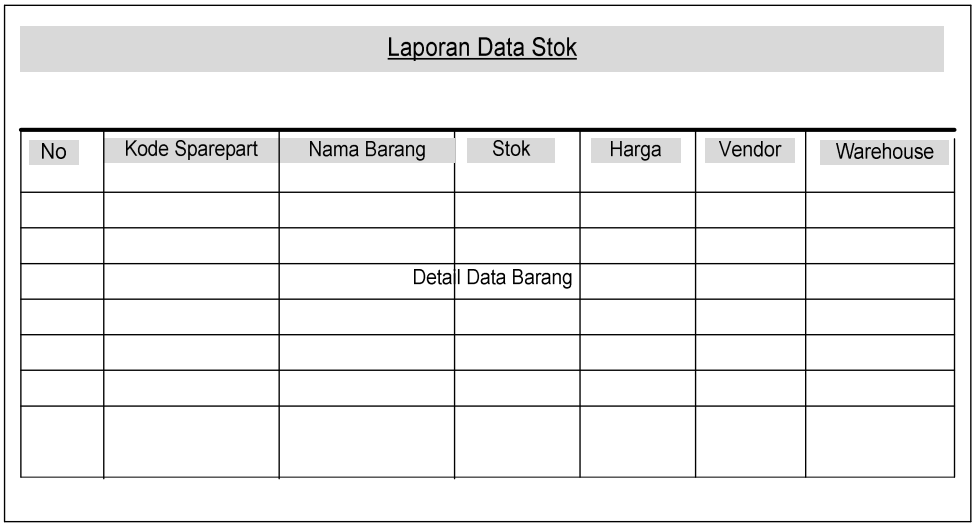
Gambar 3.41 Data Detail Request

3.2.3.10 Tampilan Tambah Barang



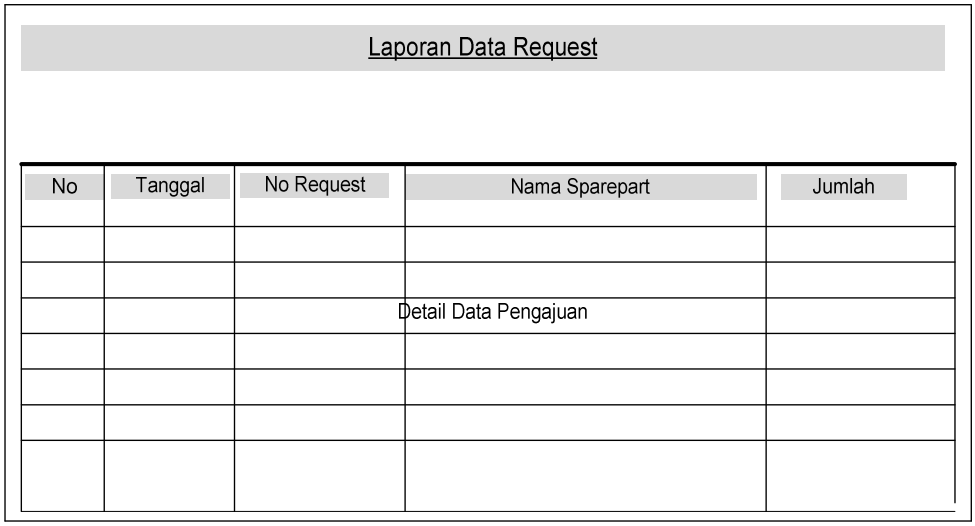
Gambar 3.42 Data Tambah Barang

3.2.3.11 Tampilan Laporan Data Stok



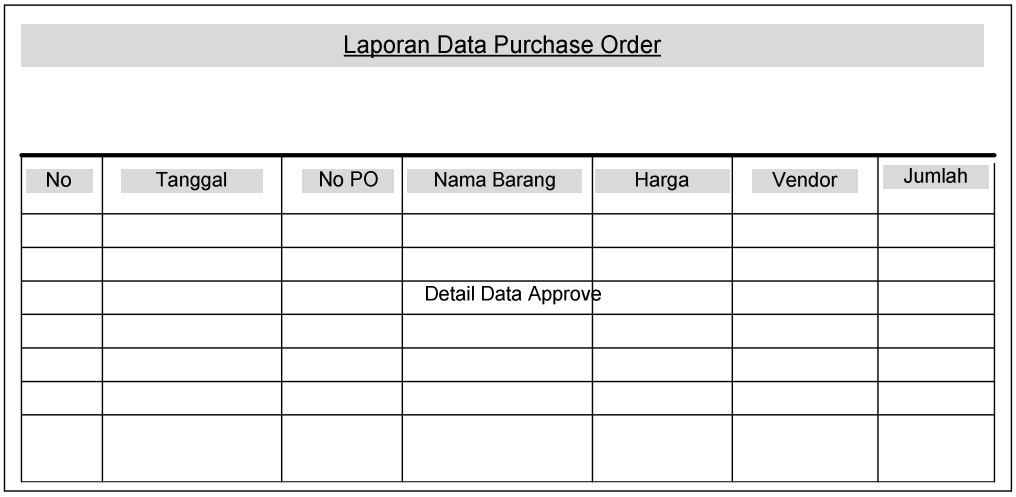
Gambar 3.43 Laporan Data Stok

3.2.3.12 Tampilan Laporan Pengajuan Barang / Data Request



Gambar 3.44 Laporan Pengajuan Barang / Data Request

3.2.3.13 Tampilan Laporan Purchase Order



Gambar 3.45 Laporan Purchase Order