BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang dibuat.

Proses implementasi untuk prosedur dalam teknologi komputer akan menggunakan bahasa pemrograman. Pertimbangan untuk memilih bahasa pemrograman didasarkan pada dua hal yaitu, kemampuan bahasa itu untuk menangani dan mengimplementasikan proses-proses yang dirancang. Bahasa pemrograman yang digunakan dalam pengembangan sistem informasi persediaan *spare part* berbasis *web* di PT. Hariff DTE adalah PHP dengan *MySQL* untuk basis datanya.

5.1.1. Batasan Implementasi

Dalam mengimplementasikan perangkat lunak ada beberapa hal yang menjadi batasan implementasi, yaitu:

1. Tidak semua kebutuhan sistem informasi persedian *spare part* dikerjakan, tetapi hanya subsistem-subsistem yang mendukung seperti penerimaan *spare part*, permintaan *spare part*, pembelian *spare part* dan pengeluaran *spare part*.

2. Serta pembuatan laporan pengeluaran *spare part*, permintaan *spare part*, penerimaan *spare part* dan persediaan *spare part*.

5.1.2. Implementasi Perangkat Lunak

Perangkat lunak yang mendukung dalam pembuatan program ini adalah sebagai berikut:

1. PHP

   Digunakan untuk mengimplementasikan rancangan sistem karena menyediakan fasilitas yang memadai untuk membuat perangkat lunak berbasis *web*.

2. *MySQL*

   Digunakan sebagai basis data yang memfasilitasi penyimpanan data-data dengan bahasa pemrograman PHP.

Perangkat lunak yang diperlukan dalam perancangan sistem ini adalah sebagai berikut:

1. Sistem operasi Windows XP Professional Service Pack 2 atau Windows 7
2. Xampp versi 1.7.7
3. Adobe Dreamweaver CS3

5.1.3. Implementasi Perangkat Keras

Kebutuhan perangkat keras merupakan hal yang sangat penting karena nantinya sistem akan berjalan apabila didukung oleh perangkat keras yang sesuai

dengan kebutuhan sistem perangkat lunak. Perangkat keras yang diperlukan dalam perancangan sistem ini adalah sebagai berikut:

1. Processor : menggunakan minimal processor dengan kecepatan 2 GHz
2. Memory : 1 GB
3. Harddisk : 160 GB
4. Monitor : Resolusi 1024 x 768
5. Mouse dan keyboard sebagai alat input

5.1.4. Implementasi Basis Data (Sintaks SQL)

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana DBMS yang digunakan adalah MySQL. Adapun implementasi basis datanya dalam bahasa SQL adalah sebagai berikut:

CREATE DATABASE : `db_sparepart`

1. Tabel penerimaan_sp

   CREATE TABLE IF NOT EXISTS `penerimaan_sp` (`NO_RECEIVE` varchar(15) NOT NULL, `TANGGAL` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, `NO_PO` varchar(15) NOT NULL, KEY `NO_RECEIVE` (`NO_RECEIVE`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

2. Tabel pengeluaran _sp

   CREATE TABLE IF NOT EXISTS `pengeluaran_sp` (`NO_FAKTUR` varchar(15) NOT NULL, `TANGGAL` timestamp NOT NULL DEFAULT

CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, `regional` varchar(50) NOT NULL, `site` varchar(50) NOT NULL, `NO_FORM` varchar(25) NOT NULL, PRIMARY KEY (`NO_FAKTUR`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

3. Tabel permintaan _sp

   CREATE TABLE IF NOT EXISTS `permintaan_sp` (`NO_FORM` varchar(25) NOT NULL, `TANGGAL_FORM` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, `REGIONAL` varchar(50) NOT NULL, `SITE` varchar(10) NOT NULL, `STATUS` enum('0','1') NOT NULL, `PEMOHON` varchar(50) NOT NULL, PRIMARY KEY (`NO_FORM`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

4. Tabel purchase_order

   CREATE TABLE IF NOT EXISTS `purchase_order` (`NO_PO` varchar(25) NOT NULL, `KODE_SUPPLIER` varchar(50) NOT NULL, `TANGGAL_PO` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, `CARA_PEMBAYARAN` enum('cash','credit') NOT NULL, `status` enum('0','1') NOT NULL DEFAULT '0', `tanggal_terima` datetime NOT NULL, PRIMARY KEY (`NO_PO`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

5. Tabel regional

   CREATE TABLE IF NOT EXISTS `regional` (`REGIONAL` varchar(50) NOT NULL, `SITE` varchar(50) NOT NULL, PRIMARY KEY (`ID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=26 ;

6. Tabel spare_part

   CREATE TABLE IF NOT EXISTS `spare_part` (`PARTNUMBER` varchar(50) NOT NULL, `DESKRIPSI` varchar(100) NOT NULL, `VENDOR` varchar(25) NOT NULL, `KATEGORI` varchar(25) NOT NULL, PRIMARY KEY (`ID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=111 ;

7. Tabel stock

   CREATE TABLE IF NOT EXISTS `stock` (`IDPART` varchar(50) NOT NULL, `STOCK` int(11) NOT NULL, `SATUAN` varchar(10) NOT NULL, `HARGA` float NOT NULL, `SOH` int(11) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1;

8. Tabel supplier

   CREATE TABLE IF NOT EXISTS `supplier` (`KODE_SUPPLIER` varchar(30) NOT NULL, `NAMA` varchar(30) NOT NULL, `ALAMAT` varchar(50) NOT NULL, `TELEPON` varchar(12) NOT NULL, `FAX`

varchar(15) NOT NULL, `EMAIL` varchar(30) NOT NULL, `KONTAK_PERSON` varchar(30) NOT NULL, `jenisSp` varchar(50) NOT NULL, PRIMARY KEY (`KODE_SUPPLIER`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

9. Tabel transaksi

   CREATE TABLE IF NOT EXISTS `transaksi` (`IDTRXFORM` varchar(25) NOT NULL, `DATE` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP, `ID` int(50) NOT NULL AUTO_INCREMENT, `PARTNUMBER` varchar(50) NOT NULL, `QTY` int(11) NOT NULL, `STATUS` varchar(15) NOT NULL DEFAULT '0', PRIMARY KEY (`ID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=305 ;

10. Tabel t_kategori

    CREATE TABLE IF NOT EXISTS `t_kategori` (`KATEGORI` varchar(50) NOT NULL, PRIMARY KEY (`ID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=44 ;

11. Tabel user

    CREATE TABLE IF NOT EXISTS `user` (`NAMA` varchar(50) NOT NULL, `SANDI` varchar(50) NOT NULL, `GROUP_LOGIN` varchar(30) NOT NULL, `NO_TELEPON` varchar(15) NOT NULL, `REGIONAL`

varchar(50) NOT NULL, `EMAIL` varchar(50) NOT NULL, PRIMARY KEY (`ID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=13 ;

5.1.5. Implementasi Antar Muka

Implementasi antar muka dilakukan dengan membuat antar muka dari setiap menu yang dibutuhkan. Setiap halaman menu yang dibuat akan akan dibentuk sebuah *file* yang berekstensi *php*. *File-file* tersebut dapat diakses dan akan menjadi penghubung antara penggunan dengan sistem. Pada tahap implementasi antar muka ini lebih menitikberatkan pada implementasi antar muka menu utama dengan antar muka yang lain.

5.1.6. Implementasi Instalasi Program

Implementasi instalasi program dilakukan pada komputer (*localhost*). Adapun cara instalasinya seperti di bawah ini.

5.1.6.1. Instalasi Program di Localhost

Adapun implementasi instalasi program yang akan dilakukan di komputer (*localhost*) yaitu dengan berbagai tahapan, diantaranya :

1. Instal terlebih dahulu *webserver* (XAMPP).
2. Masukan file *website* ke dalam *folder …xampp/htdocs.*
3. Masukan file *database* ke dalam *folder …xampp/mysql/data.*

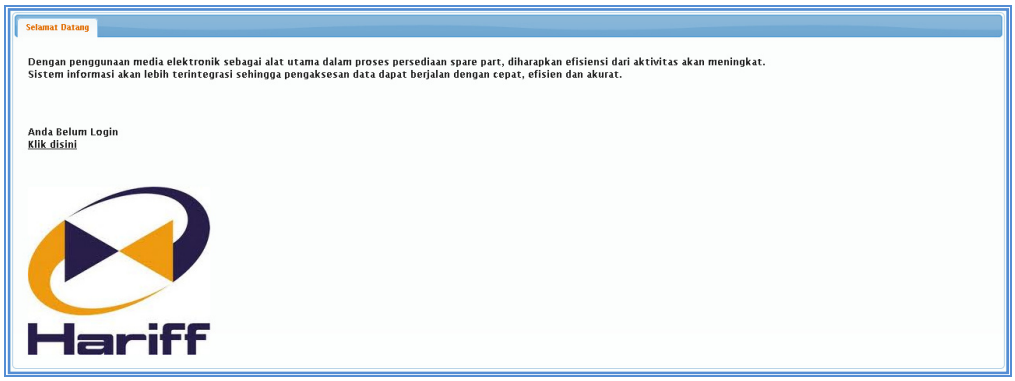
4. Panggil file *website* melalui *web browser Mozilla Firefox* dengan mengetikan alamat *localhost*/SI.
5. Sistem informasi persediaan *spare part* siap dijalankan.

5.1.7. Penggunaan Program

Untuk memulai menjalankan program maka sebelumnya harus mengaktifkan *apache* dan *MySql* yang ada di program *Xampp* kemudian buka *browser* (*Mozilla Firefox)*, tuliskan pada *address bar* *http://localhost/SI*, yang berfungsi untuk memanggil program yang berada di *folder htdocs.* Setelah klik *enter*, maka akan muncul tampilan program yang secara otomatis akan langsung menuju ke tampilan menu *login*.

1. Tampilan Utama

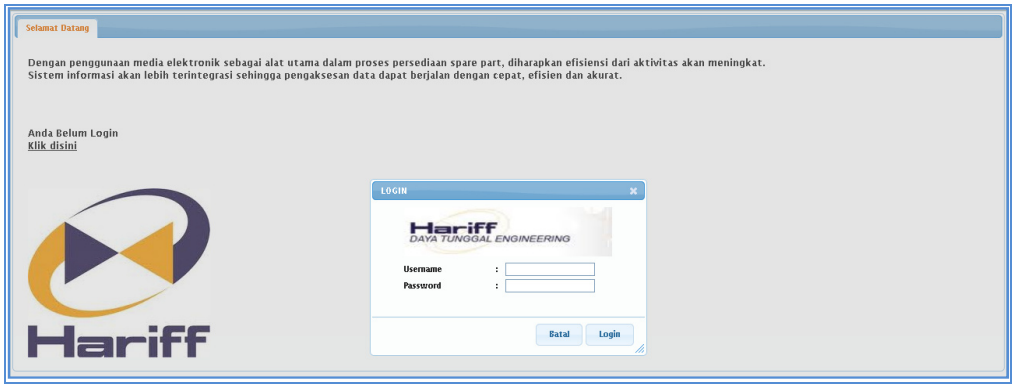
Tampilan yang akan muncul setelah memasukkan alamat di *address bar* terlihat pada gambar 5.1. Dalam tampilan utama terdapat menu untuk masuk (*login*) ke dalam program.



Gambar 5.1. Tampilan Utama

2. Tampilan Menu *Login*

Untuk dapat mengakses program, terlebih dahulu masukkan *username* dan *password* yang sebelumnya telah didaftarkan. Kemudian klik tombol *login* untuk masuk ke dalam program.



Gambar 5.2. Tampilan Menu *Login*

3. Tampilan Menu *User* SPM (*Spare Part Management*)

Apabila *login* berhasil, maka akan masuk ke dalam program sesuai dengan hak akses *user* masing-masing. Gambar di bawah ini adalah menu-menu yang dapat diakses oleh *user* sebagai *Spare Part Management*.



Gambar 5.3. Tampilan Menu *User* SPM (*Spare Part Management*)

Menu yang akan muncul dengan *user* yang masuk sebagai *Spare Part Management* (dan sekaligus bertindak sebagai *admin*) adalah sebagai berikut:

a. Master, terdiri dari:

   1) *User*

      Berfungsi untuk menambah *user*, edit *user*, *view user*, cari *user*, dan hapus *user*.

   2) *Supplier*

      Berfungsi untuk menambah *supplier*, edit *supplier*, *view supplier*, cari *supplier*, dan hapus *supplier*.

   3) *Spare Part*

      Berfungsi untuk menambah *spare part*, edit *spare part*, *view spare part*, cari *spare part*, dan hapus *spare part*.

b. Transaksi

   1) Permintaan *Spare Part*

      Berfungsi untuk membuat *form* permintaan *spare part*, yang berisi data regional, *site*, *spare part*, jumlah *spare part* yang diminta dan pemohon.

   2) Pengeluaran *Spare Part*

      Berfungsi untuk membuat faktur pengeluaran *spare part*, yang berisi nomor faktur, data regional, data *site*, nomor *form* permintaan *spare part* , data *spare part* yang akan dikeluarkan.

3) *Purchase Order*

Berfungsi untuk membuat faktur pembelian *spare part* kepada *supplier*, yang berisi nomor pembelian, nama *supplier*, cara pembelian, alamat kirim, data *spare part* yang akan dibeli beserta harganya.

4) Penerimaan *Spare Part*

Berfungsi untuk membuat faktur penerimaan *spare part*, yang berisi nomor penerimaan, nama *supplier*, nomor pembelian, dan data *spare part* yang diterima.

c. List

1) Daftar Permintaan SP

Berfungsi sebagai menu untuk menampilkan daftar permintaan *spare part* dari transaksi yang telah dilakukan.

2) Daftar Pengeluaran SP

Berfungsi sebagai menu untuk menampilkan daftar pengeluaran *spare part* dari transaksi yang telah dilakukan.

3) Daftar *Purchase Order*

Berfungsi sebagai menu untuk menampilkan daftar pembelian *spare part* dari transaksi yang telah dilakukan.

4) Daftar Penerimaan SP

Berfungsi sebagai menu untuk menampilkan daftar penerimaan *spare part* dari transaksi yang telah dilakukan.

d. Laporan

1) Rekap laporan

Berfungsi sebagai menu yang dapat menampilkan laporan sesuai jenis laporan yang dipilih dan tanggal yang diinginkan dari transaksi yang telah dilakukan.

Untuk yang login selain *admin* beberapa menu diatur oleh *admin*, hal tersebut dapat diatur hak aksesnya pada menu *user*.

4. Tampilan Menu User Manager

Apabila *login* berhasil, maka akan masuk ke dalam program sesuai dengan hak akses *user* masing-masing. Gambar di bawah ini adalah menu-menu yang dapat diakses oleh *user* sebagai *Manager*.



Gambar 5.4. Tampilan Menu *User* Manager

5. Tampilan Menu User FSS (*Field Service Staff*)

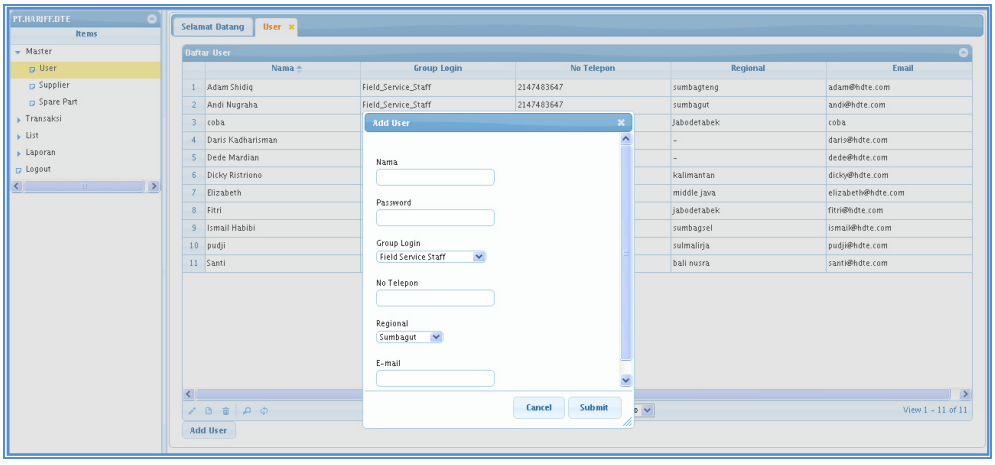
Apabila *login* berhasil, maka akan masuk ke dalam program sesuai dengan hak akses *user* masing-masing. Gambar di bawah ini adalah menu-menu yang dapat diakses oleh *user* sebagai *Field Service Staff*.



Gambar 5.5. Tampilan Menu *User* FSS (*Field Service Staff*)

6. Tampilan Menu *Input* Data *User*

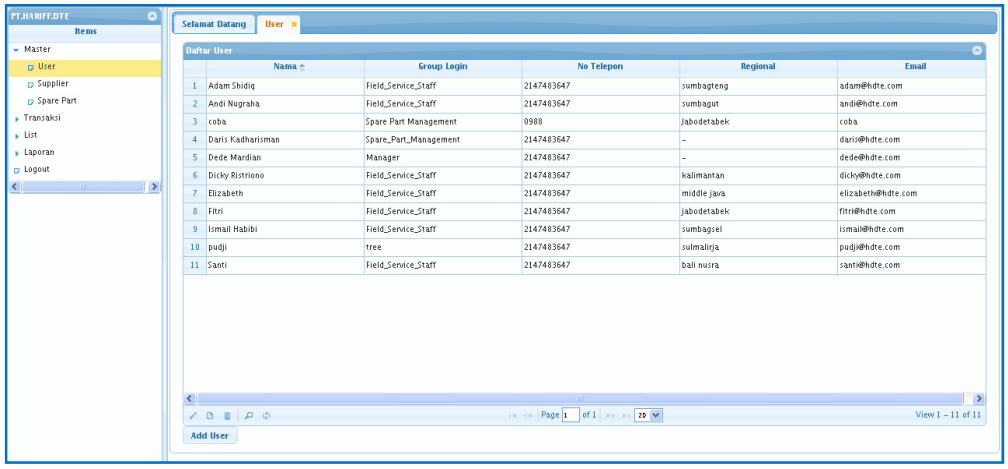
Menu *user* digunakan untuk *input* data *user* baru. Untuk *input* data baru tekan tombol *Add User*, lalu isikan data sesuai *form* yang tersedia, kemudian tekan tombol *Submit*.



Gambar 5.6. Tampilan Menu *Input* Data *User*

7. Tampilan Daftar *User*

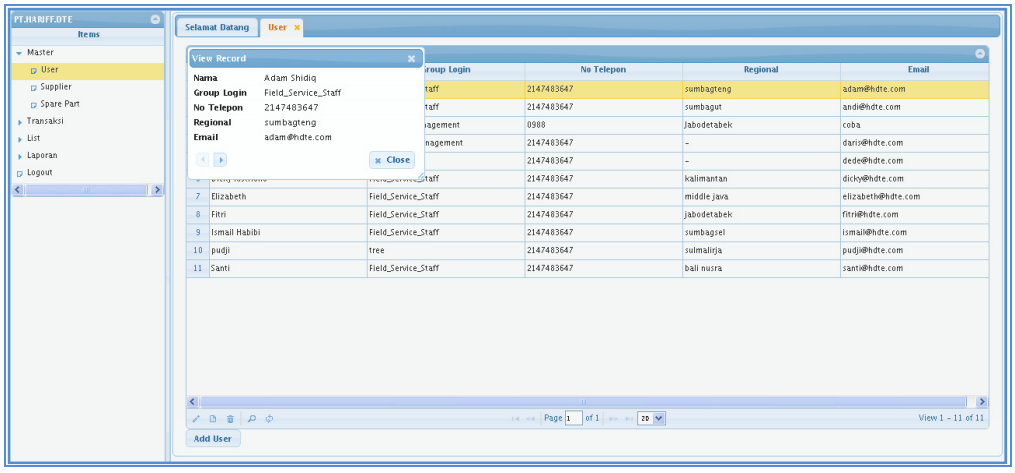
Berikut tampilan daftar *user* yang telah dimasukkan.



Gambar 5.7. Tampilan Daftar *User*

8. Tampilan Menu *View Record User*

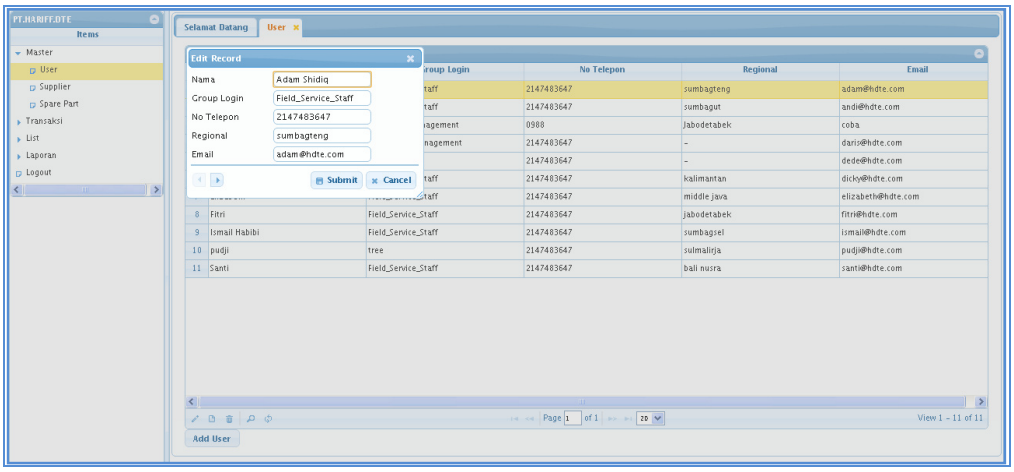
Untuk melihat daftar *user* secara detail, pilih menu *view selected rows* yang ada dikiri bawah.



Gambar 5.8. Tampilan Menu *View Record User*

9. Tampilan Menu *Edit Record User*

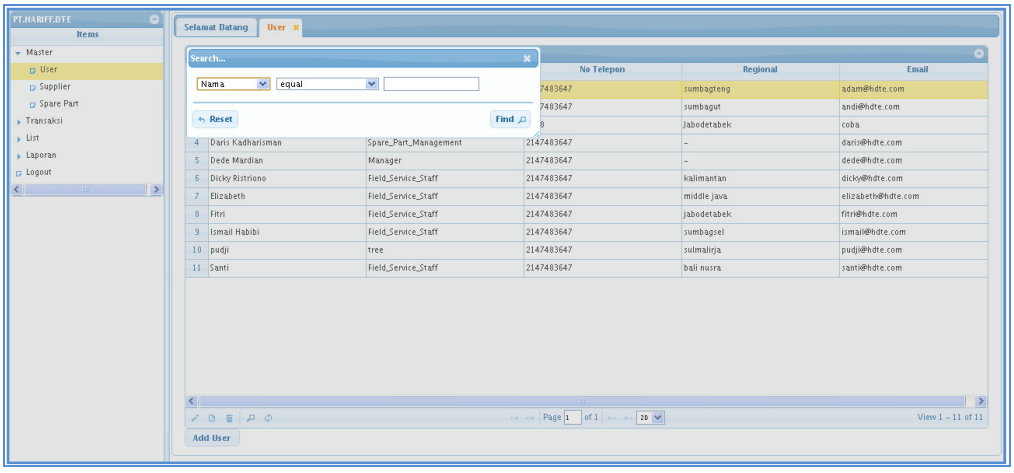
Untuk mengubah data *user*, pilih menu *edit selected rows* yang ada dikiri bawah. Kemudian klik tombol *Submit* apabila telah selesai.



Gambar 5.9. Tampilan Menu *Edit Record User*

10. Tampilan Menu *Search User*

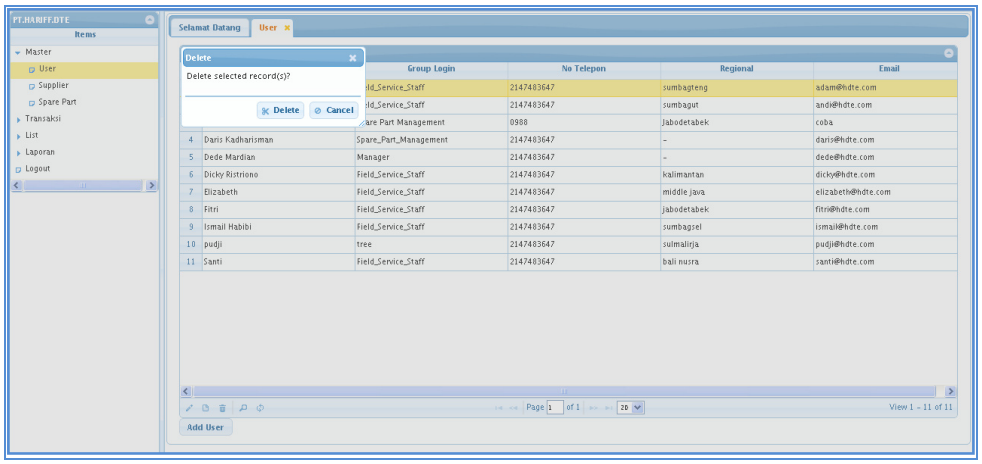
Untuk mencari data *user*, pilih menu *find records* yang ada dikiri bawah.



Gambar 5.10. Tampilan Menu *Search User*

11. Tampilan Menu *Delete User*

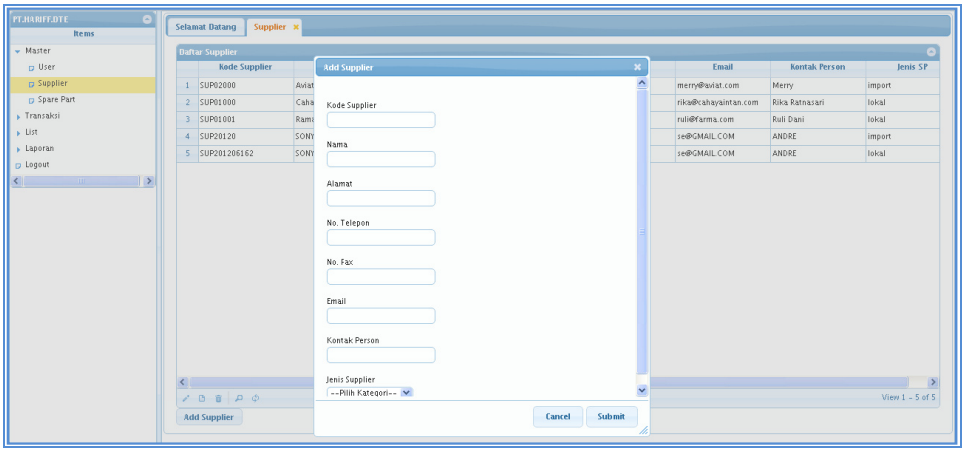
Untuk menghapus data *user*, pilih menu *delete selected rows* yang ada dikiri bawah. Kemudian klik tombol *Delete* untuk menghapus.



Gambar 5.11. Tampilan Menu *Delete User*

12. Tampilan Menu *Input* Data *Supplier*

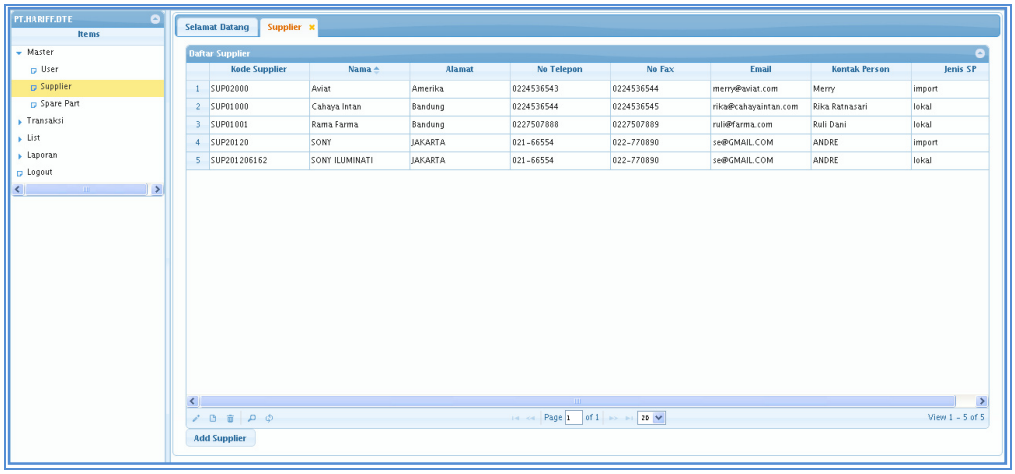
Menu *supplier* digunakan untuk *input* data *supplier* baru. Untuk *input* data baru tekan tombol *Add Supplier*, lalu isikan data sesuai *form* yang tersedia.



Gambar 5.12. Tampilan Menu *Input* Data *Supplier*

13. Tampilan Daftar *Supplier*

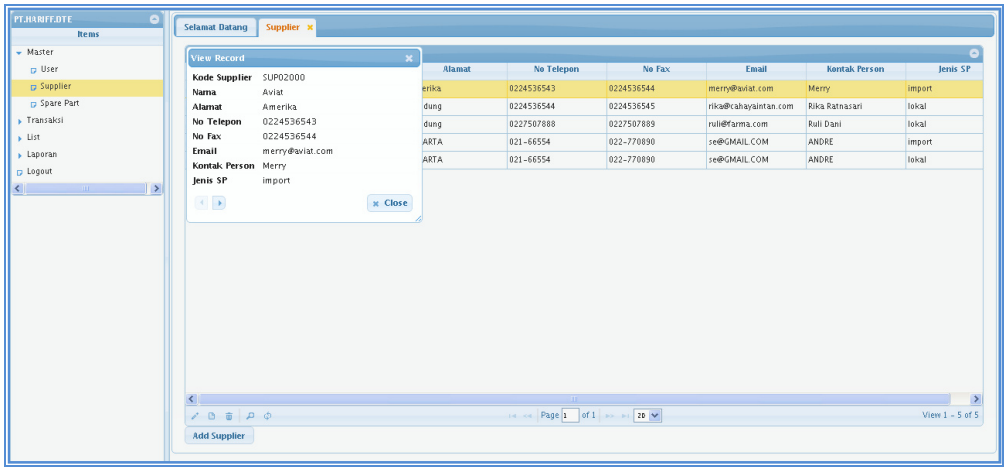
Berikut tampilan daftar *supplier* yang telah dimasukkan.



Gambar 5.13. Tampilan Daftar *Supplier*

14. Tampilan Menu *View Record Supplier*

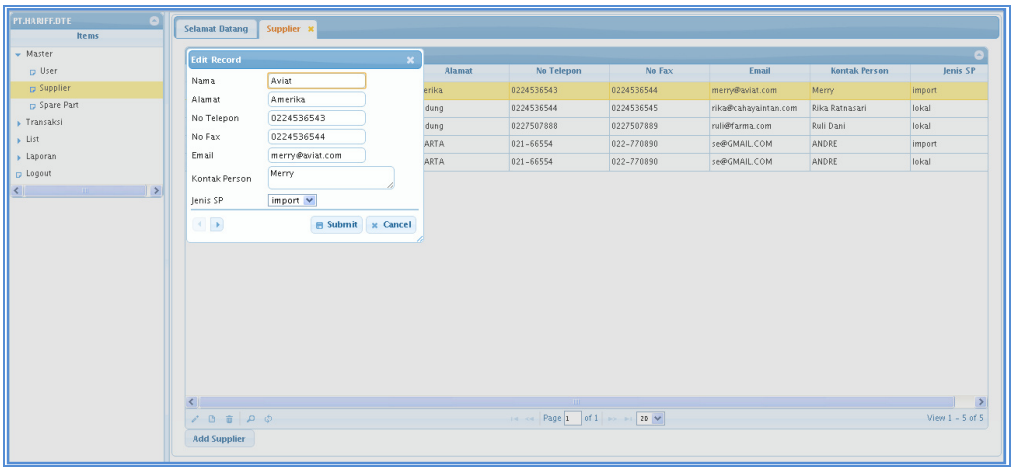
Untuk melihat data *supplier* secara detail, pilih menu *view selected rows* yang ada dikiri bawah.



Gambar 5.14. Tampilan Menu *View Record Supplier*

15. Tampilan Menu *Edit Record Supplier*

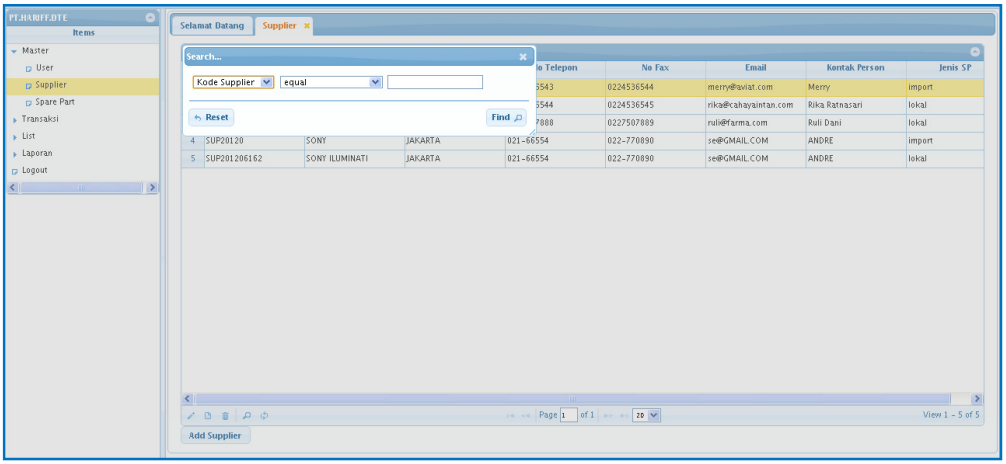
Untuk mengubah data *supplier*, pilih menu *edit selected rows* yang ada dikiri bawah. Kemudian klik tombol *Submit* apabila telah selesai.



Gambar 5.15. Tampilan Menu *Edit Record Supplier*

16. Tampilan Menu *Search Supplier*

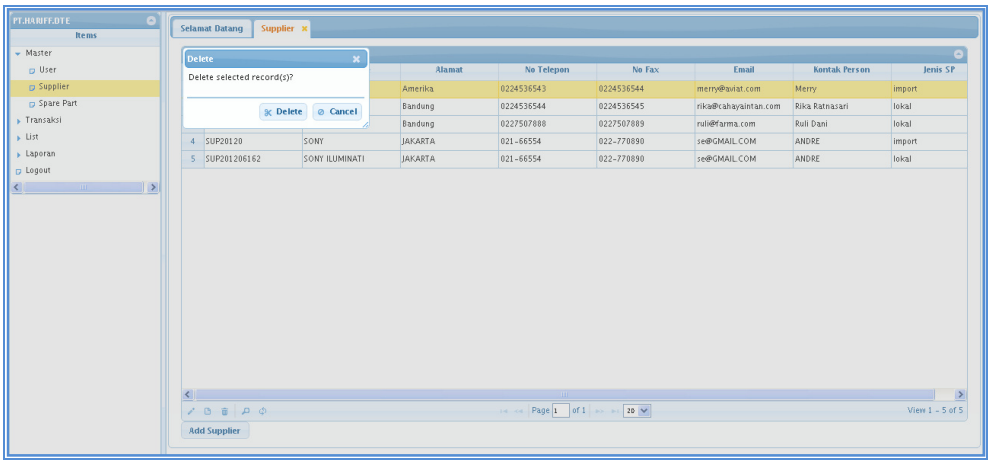
Untuk mencari data *supplier*, pilih menu *find records* yang ada dikiri bawah.



Gambar 5.16. Tampilan Menu *Search Supplier*

17. Tampilan Menu *Delete Supplier*

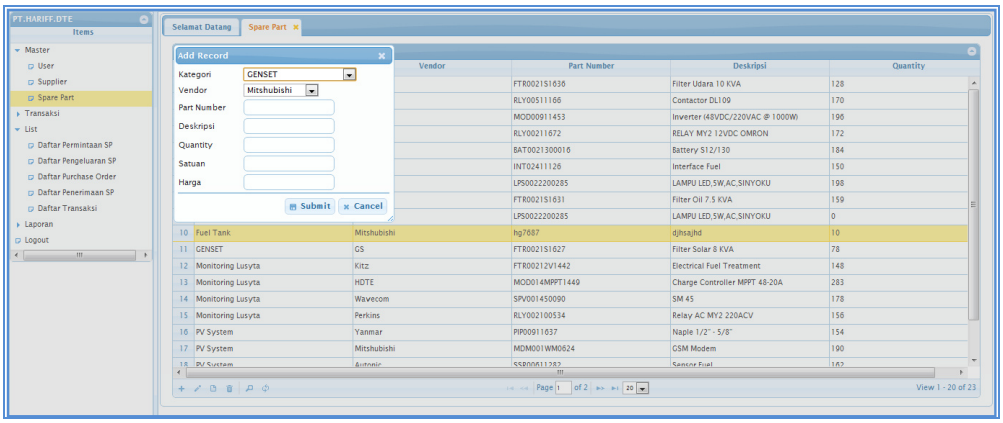
Untuk menghapus data *supplier*, pilih menu *delete selected rows* yang ada dikiri bawah. Kemudian klik tombol *Delete* untuk menghapus.



Gambar 5.17. Tampilan Menu *Delete Supplier*

18. Tampilan Menu *Input* Data *Spare Part*

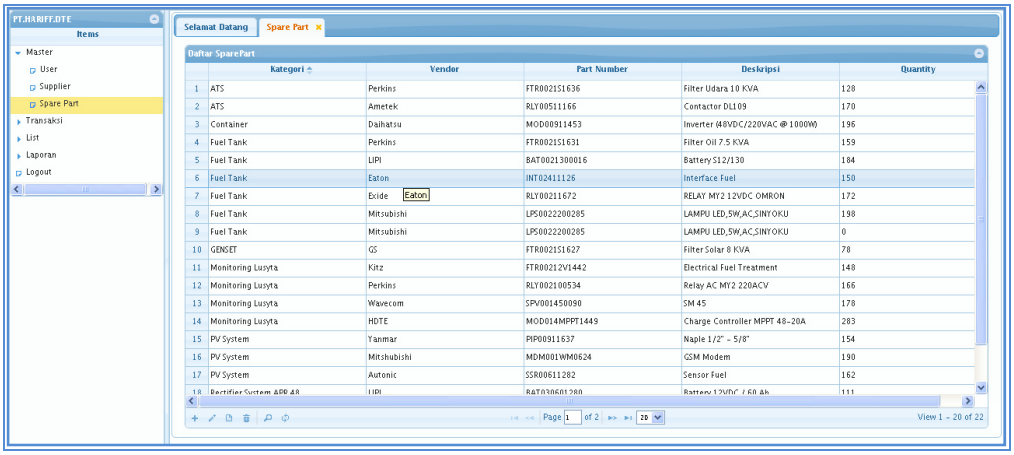
Menu *spare part* digunakan untuk *input* data *spare part* baru. Untuk *input* data baru tekan tombol *Add Part*, lalu isikan data sesuai *form* yang tersedia, kemudian tekan tombol *Submit*.



Gambar 5.18. Tampilan Menu *Input* Data *Spare Part*

19. Tampilan Daftar *Spare Part*

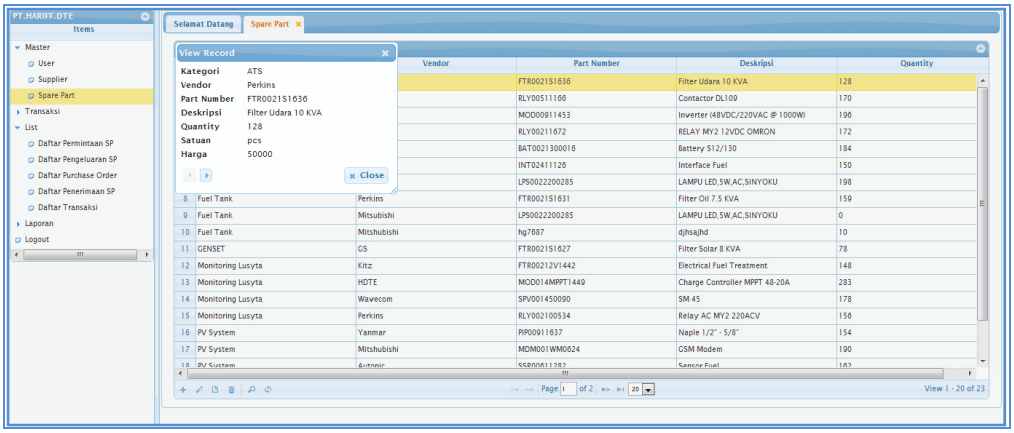
Berikut tampilan data *spare part* yang telah dimasukkan.



Gambar 5.19. Tampilan Daftar *Spare Part*

20. Tampilan Menu *View Record Spare Part*

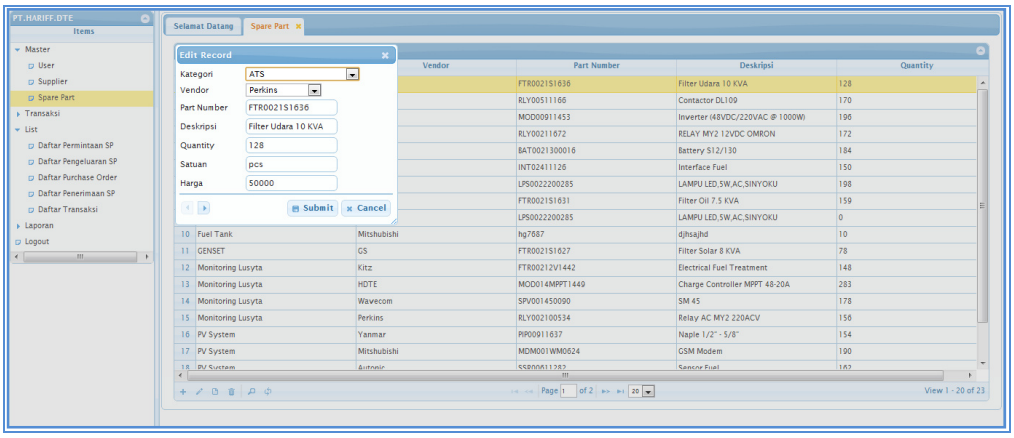
Untuk melihat data *spare part* secara detail, pilih menu *view selected rows* yang ada dikiri bawah.



Gambar 5.20. Tampilan Menu *View Record Spare Part*

21. Tampilan Menu *Edit Record Spare Part*

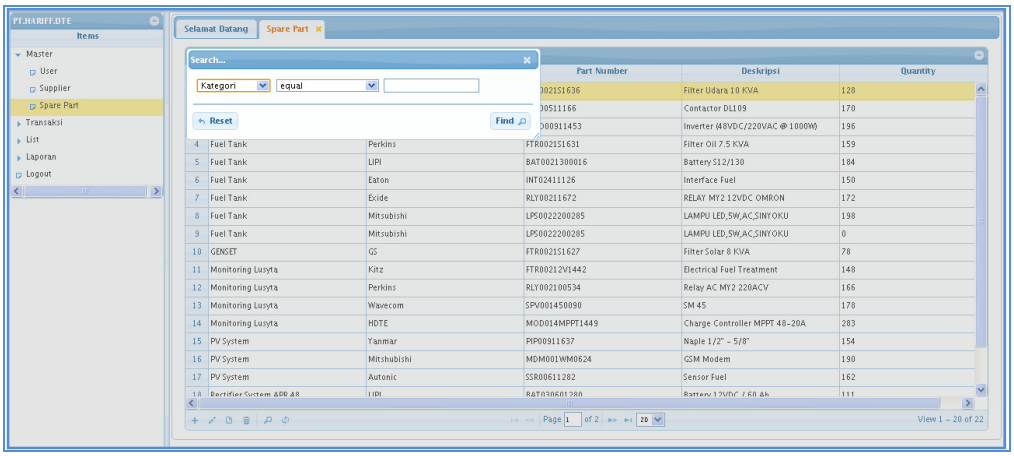
Untuk mengubah data *spare part*, pilih menu *edit selected rows* yang ada dikiri bawah. Kemudian klik tombol *Submit* apabila telah selesai.



Gambar 5.21. Tampilan Menu *Edit Record Spare Part*

22. Tampilan Menu *Search Spare Part*

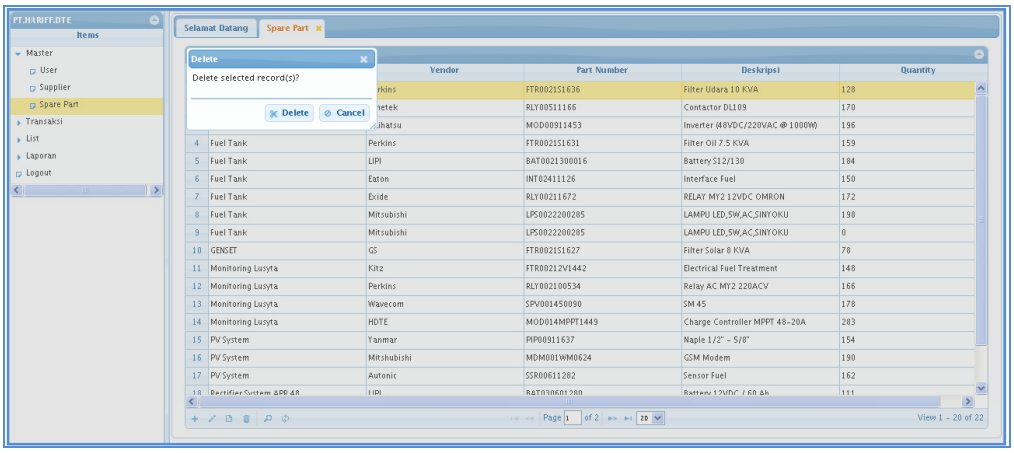
Untuk mencari data *spare part*, pilih menu *find records* yang ada dikiri bawah.



Gambar 5.22. Tampilan Menu *Search Spare Part*

23. Tampilan Menu *Delete Spare Part*

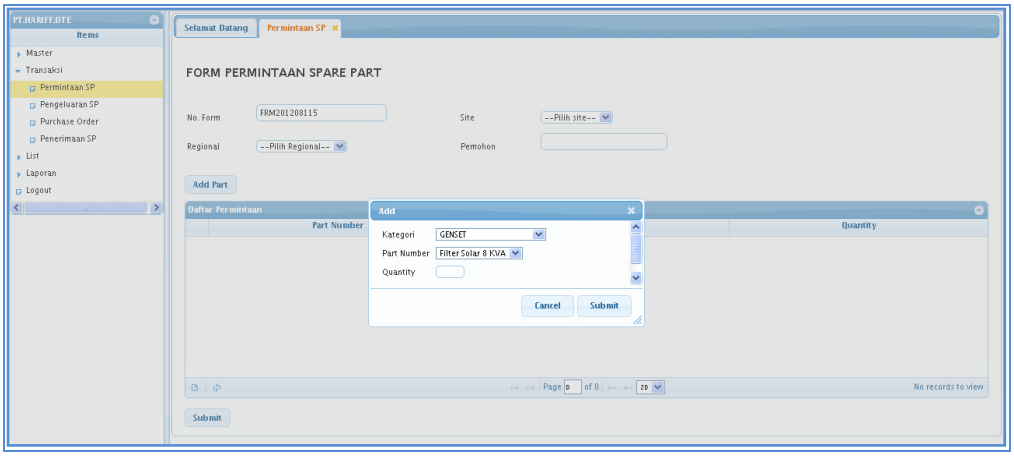
Untuk menghapus data *spare part*, pilih menu *delete selected rows* yang ada dikiri bawah. Kemudian klik tombol *Delete* untuk menghapus.



Gambar 5.23. Tampilan Menu *Delete Spare Part*

24. Tampilan Menu *Input Form* Permintaan SP

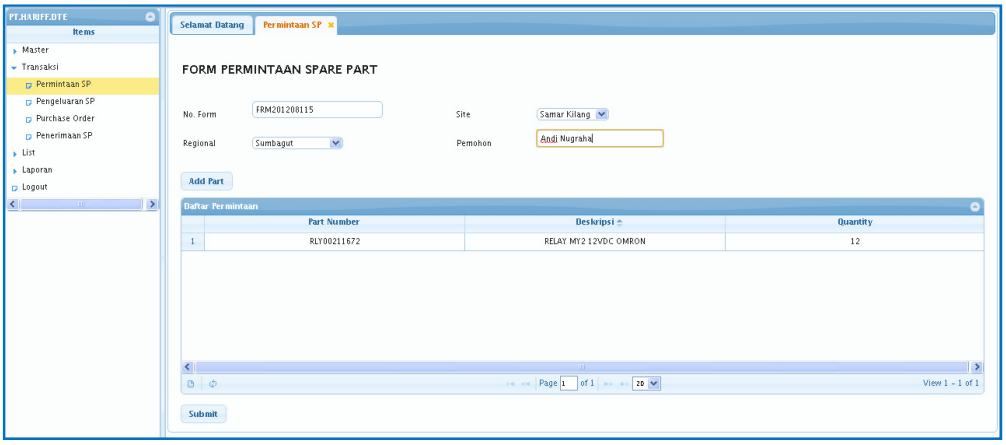
Menu permintaan *spare part* berfungsi untuk membuat *form* permintaan *spare part.* Untuk nomor *form* otomatis akan terisi ketika membuka menu ini, karena sudah diatur untuk *auto increment* mengikuti tahun dan bulan. Untuk regional, diisi sesuai dengan regional *user* berada, dan untuk *site* disesuaikan dengan peruntukkan *spare part* ini akan digunakan. Untuk pemohon diisi sama dengan *user* saat *login*. Tekan tombol *Add* untuk memasukkan *spare part* yang diperlukan berikut jumlahnya. Tekan tombol *Submit* apabila sudah memasukkan semua data-data yang dibutuhkan. Berikut tampilan menu untuk membuat *form* permintaan *spare part.*



Gambar 5.24. Tampilan Menu *Input Form* Permintaan SP

25. Tampilan *Form* Permintaan SP

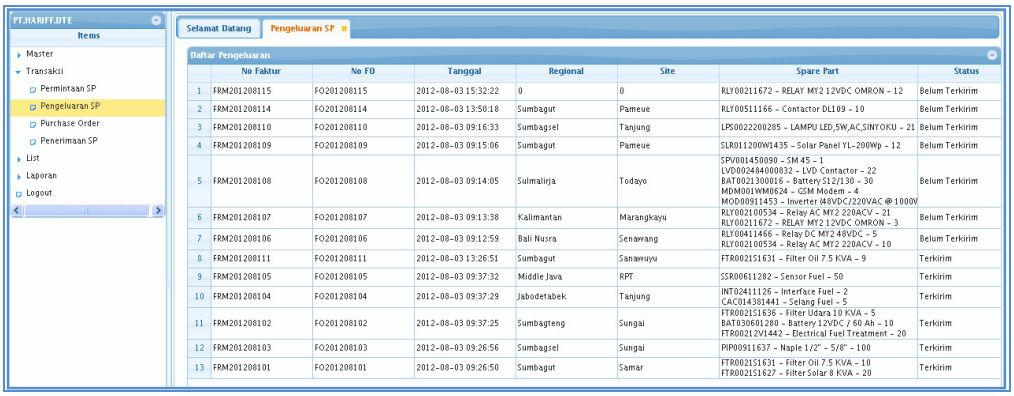
Berikut tampilan *form* permintaan *spare part* yang telah diisi data *spare part*.



Gambar 5.25. Tampilan *Form* Permintaan SP

26. Tampilan Menu *Input* Faktur Pengeluaran SP

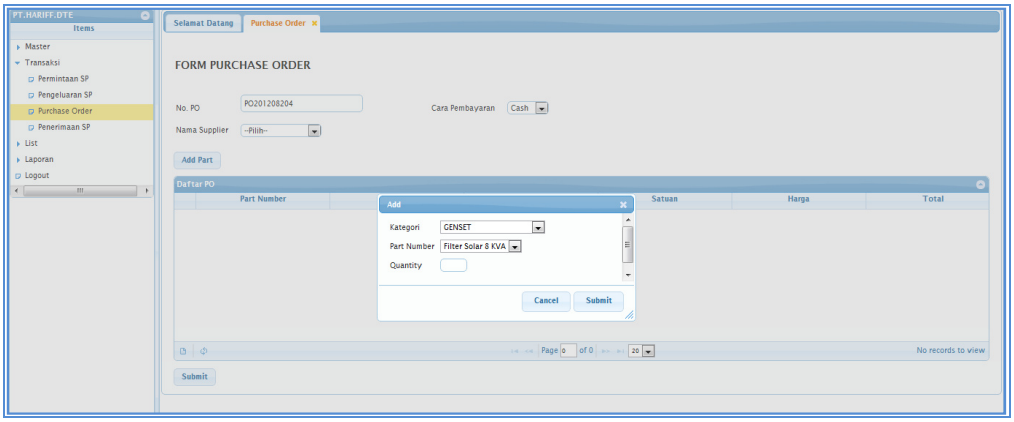
Menu pengeluaran *spare part* berfungsi untuk membuat *form* pengeluaran *spare part.* Ganti status menjadi terkirim, apabila *spare part* sudah siap untuk dikirim, dan nomor faktur akan mengikuti nomor form permintaannya. Berikut tampilan menu untuk membuat faktur pengeluaran *spare part.*



Gambar 5.26. Tampilan Menu *Input* Faktur Pengeluaran SP

27. Tampilan Menu *Input Purchase Order*

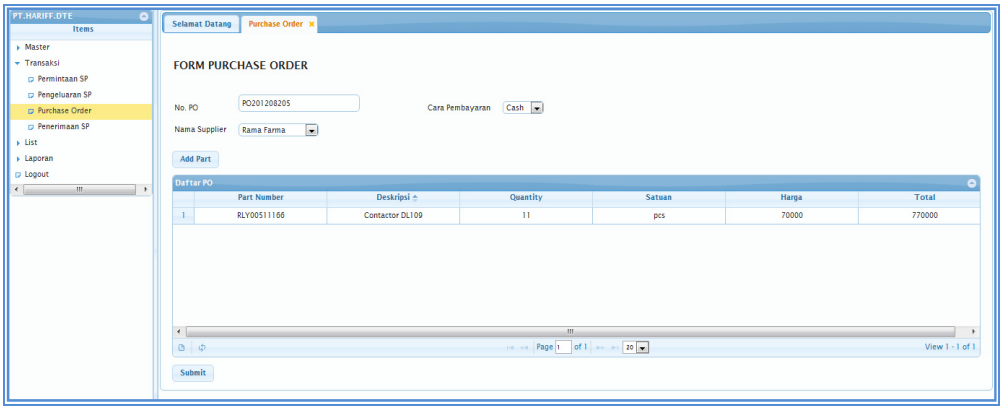
Menu *purchase order* berfungsi untuk membuat faktur *purchase order.* Untuk nomor *po* otomatis akan terisi ketika membuka menu ini, karena sudah diatur untuk *auto increment* mengikuti tahun dan bulan. Untuk nama *supplier* dipilih sesuai dengan nama *supplier* yang diperlukan, untuk cara pembayaran sesuaikan dengan kondisi keuangan perusahaan. Tekan tombol *Add* untuk memasukkan *spare part* yang datang berikut jumlahnya. Tekan tombol *Submit* apabila sudah memasukkan semua data-data. Berikut tampilan menu untuk membuat faktur *purchase order.*



Gambar 5.27. Tampilan Menu *Input Purchase Order*

28. Tampilan *Form Purchase Order*

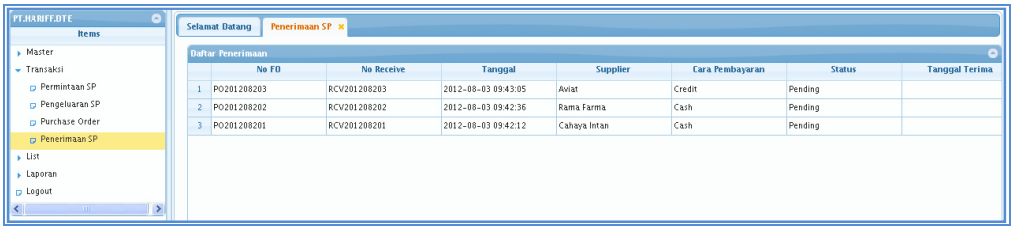
Berikut tampilan *form purchase order* yang telah diisi data *spare part*.



Gambar 5.28. Tampilan Menu *Form Purchase Order*

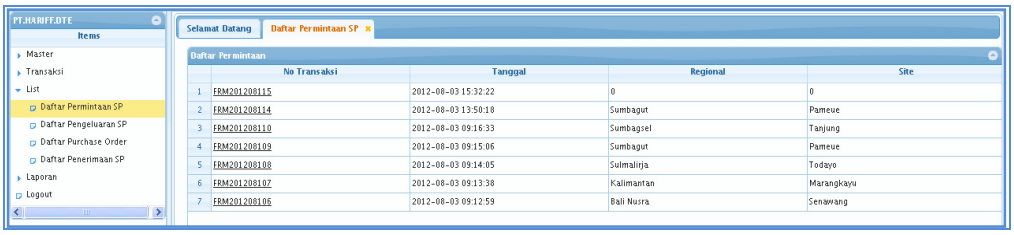
29. Tampilan Menu *Input* Faktur Penerimaan SP

Menu penerimaan *spare part* berfungsi untuk membuat faktur penerimaan *spare part.* Ganti status menjadi diterima, apabila *spare part* sudah diterima dari *supplier*, dan nomor faktur penerimaan akan mengikuti nomor pembeliaannya (PO). Berikut tampilan menu untuk membuat faktur penerimaan *spare part.*



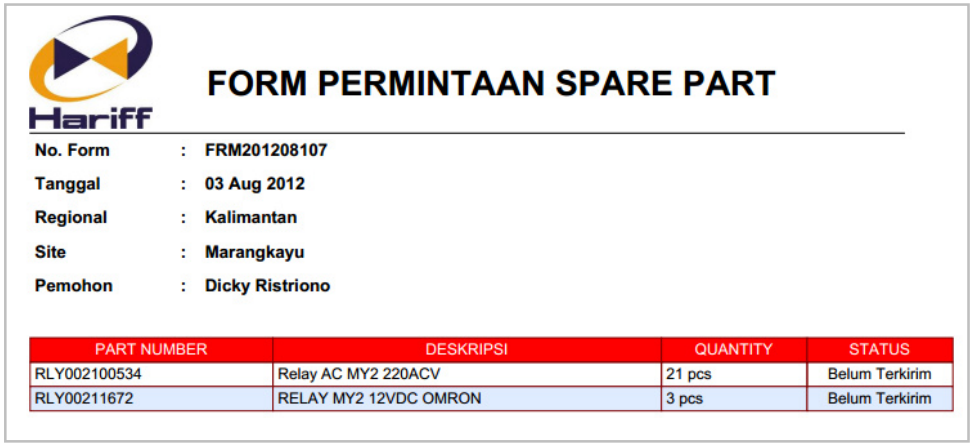
Gambar 5.29. Tampilan Menu *Input* Faktur Penerimaan SP

30. Tampilan Menu Daftar Permintaan SP



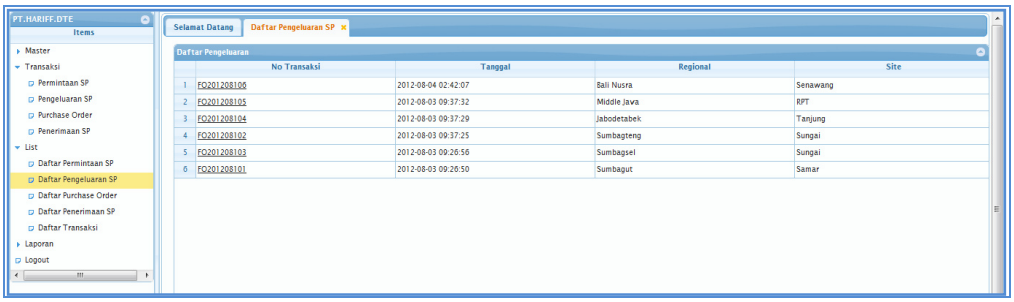
Gambar 5.30. Tampilan Menu Daftar Permintaan SP

31. Tampilan *Output Form* Permintaan SP



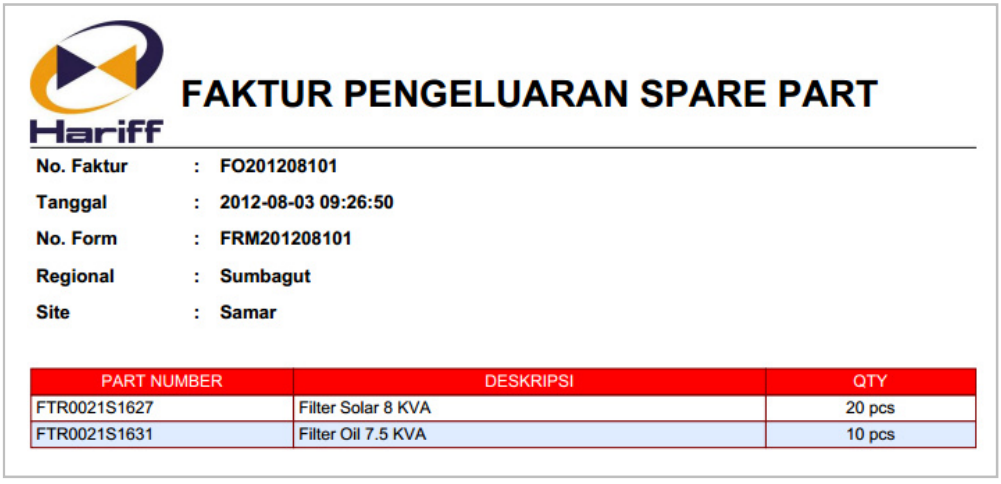
Gambar 5.31. Tampilan *Output Form* Permintaan SP

32. Tampilan Menu Daftar Pengeluaran SP



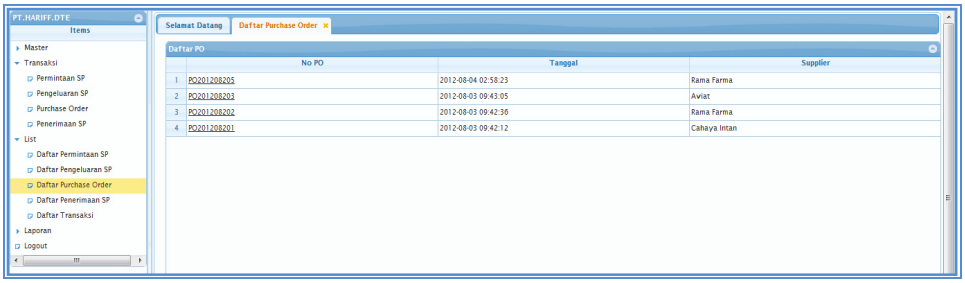
Gambar 5.32. Tampilan Menu Daftar Pengeluaran SP

33. Tampilan *Output* Faktur Pengeluaran SP



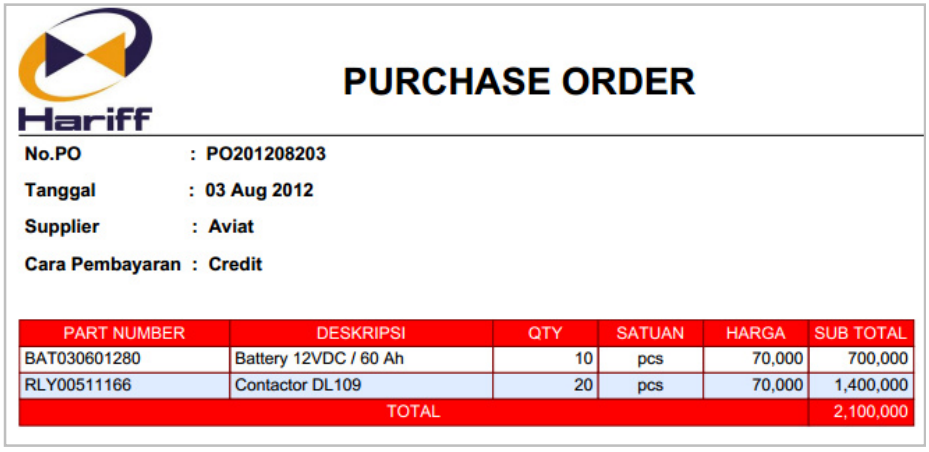
Gambar 5.33. Tampilan *Output* Faktur Pengeluaran SP

34. Tampilan Menu Daftar *Purchase Order*



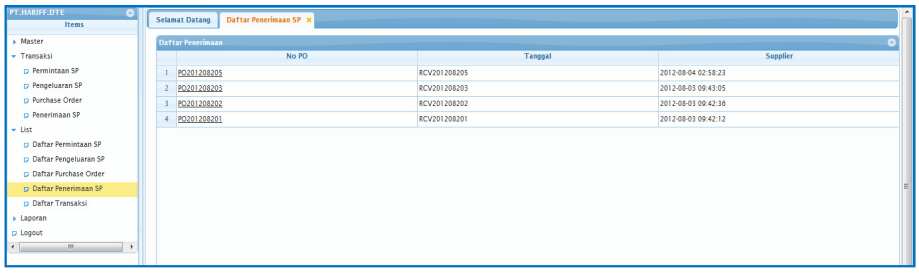
Gambar 5.34. Tampilan Menu Daftar *Purchase Order*

35. Tampilan *Output Form Purchase Order*



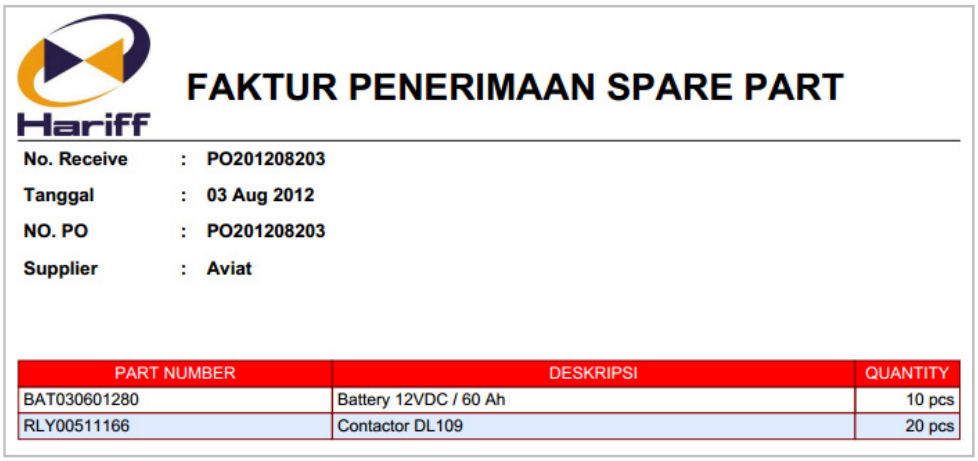
Gambar 5.35. Tampilan *Output Form Purchase Order*

36. Tampilan Menu Daftar Penerimaan *Spare Part*



Gambar 5.36. Tampilan Menu Daftar Penerimaan *Spare Part*

37. Tampilan *Output* Faktur Penerimaan SP



Hariff

FAKTUR PENERIMAAN SPARE PART

No. Receive	:	PO201208203
Tanggal	:	03 Aug 2012
NO. PO	:	PO201208203
Supplier	:	Aviat

PART NUMBER	DESKRIPSI	QUANTITY
BAT030601280	Battery 12VDC / 60 Ah	10 pcs
RLY00511166	Contactor DL109	20 pcs

Gambar 5.37. Tampilan Menu *Output* Faktur Penerimaan SP

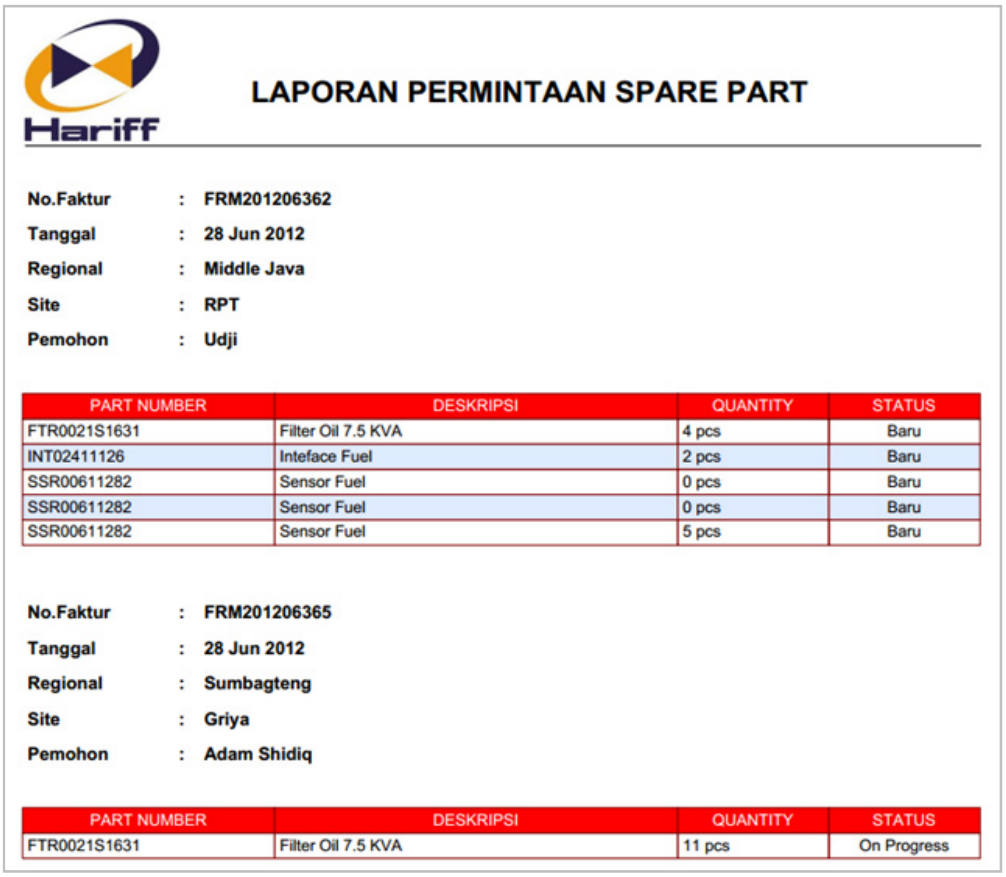
38. Tampilan Menu Rekap Laporan

Menu laporan berfungsi untuk melihat aktifitas atau transaksi yang telah dilakukan. Laporan ini ditujukan untuk manager. Periode laporan bisa dipilih sesuai dengan kebutuhan.



Gambar 5.38. Tampilan Menu Rekap Laporan

39. Tampilan *Output* Laporan Permintaan *Spare Part*



Hariff

LAPORAN PERMINTAAN SPARE PART

No.Faktur : FRM201206362
Tanggal : 28 Jun 2012
Regional : Middle Java
Site : RPT
Pemohon : Udji

PART NUMBER	DESKRIPSI	QUANTITY	STATUS
FTR0021S1631	Filter Oil 7.5 KVA	4 pcs	Baru
INT02411126	Inteface Fuel	2 pcs	Baru
SSR00611282	Sensor Fuel	0 pcs	Baru
SSR00611282	Sensor Fuel	0 pcs	Baru
SSR00611282	Sensor Fuel	5 pcs	Baru

No.Faktur : FRM201206365
Tanggal : 28 Jun 2012
Regional : Sumbagteng
Site : Griya
Pemohon : Adam Shidiq

PART NUMBER	DESKRIPSI	QUANTITY	STATUS
FTR0021S1631	Filter Oil 7.5 KVA	11 pcs	On Progress

Gambar 5.39. Tampilan *Output* Laporan Permintaan *Spare Part*

40. Tampilan Menu *Output* Laporan Pengeluaran *Spare Part*



Hariff

LAPORAN PENGELUARAN SPARE PART

No. Faktur : FO201206182
Tanggal : 29 Jun 2012
NO. Form :
Regional : Sumbagsel
Site : Tanjung

PART NUMBER	DESKRIPSI	QTY
RLY00411466	Relay DC MY2 48VDC, IEC255	2 pcs

No. Faktur : FO201206181
Tanggal : 29 Jun 2012
NO. Form :
Regional : Jabodetabek
Site : Tanjung

PART NUMBER	DESKRIPSI	QTY
FTR0021S1636	Filter Udara 10 KVA	0 pcs
FTR0021S1636	Filter Udara 10 KVA	0 pcs

Gambar 5. 40. Tampilan *Output* Laporan Pengeluaran *Spare Part*

41. Tampilan Menu *Output* Laporan Penerimaan *Spare Part*



Hariff

LAPORAN PENERIMAAN SPARE PART

No. Receive : RCV201206213
Tanggal : 28 Jun 2012
NO. PO : 222
Supplier : SONY

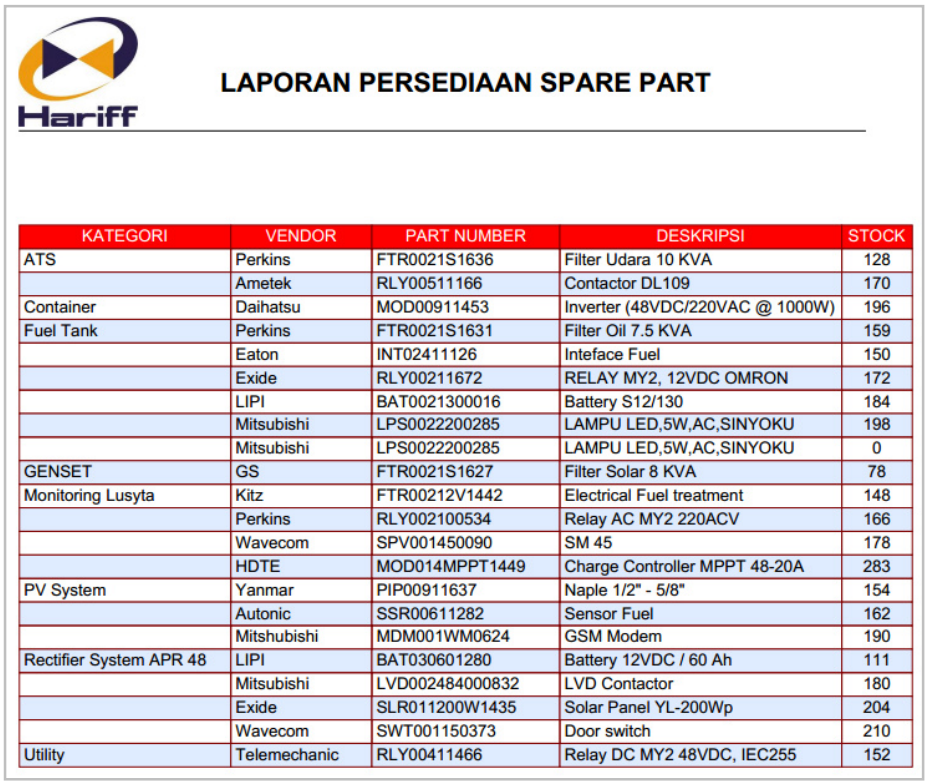
PART NUMBER	DESKRIPSI	QUANTITY
BAT030601280	Battery 12VDC / 60 Ah	10 pcs

No. Receive : RCV201206216
Tanggal : 28 Jun 2012
NO. PO : PO2012061
Supplier : Rama Farma

PART NUMBER	DESKRIPSI	QUANTITY
FTR0021S1631	Filter Oil 7.5 KVA	1 pcs

Gambar 5.41. Tampilan *Output* Laporan Penerimaan *Spare Part*

42. Tampilan *Output* Laporan Persediaan *Spare Part*



Hariff

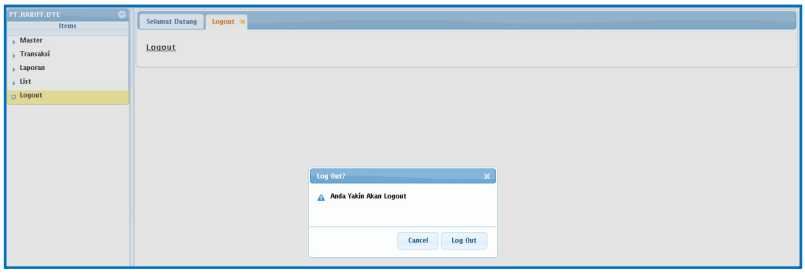
LAPORAN PERSEDIAAN SPARE PART

KATEGORI	VENDOR	PART NUMBER	DESKRIPSI	STOCK
ATS	Perkins	FTR0021S1636	Filter Udara 10 KVA	128
	Ametek	RLY00511166	Contactor DL109	170
Container	Daihatsu	MOD00911453	Inverter (48VDC/220VAC @ 1000W)	196
Fuel Tank	Perkins	FTR0021S1631	Filter Oil 7.5 KVA	159
	Eaton	INT02411126	Inteface Fuel	150
	Exide	RLY00211672	RELAY MY2, 12VDC OMRON	172
	LIPI	BAT0021300016	Battery S12/130	184
	Mitsubishi	LPS0022200285	LAMPU LED,5W,AC,SINYOKU	198
	Mitsubishi	LPS0022200285	LAMPU LED,5W,AC,SINYOKU	0
GENSET	GS	FTR0021S1627	Filter Solar 8 KVA	78
Monitoring Lusyta	Kitz	FTR00212V1442	Electrical Fuel treatment	148
	Perkins	RLY002100534	Relay AC MY2 220ACV	166
	Wavecom	SPV001450090	SM 45	178
	HDTE	MOD014MPPT1449	Charge Controller MPPT 48-20A	283
PV System	Yanmar	PIP00911637	Naple 1/2" - 5/8"	154
	Autonic	SSR00611282	Sensor Fuel	162
	Mitshubishi	MDM001WM0624	GSM Modem	190
Rectifier System APR 48	LIPI	BAT030601280	Battery 12VDC / 60 Ah	111
	Mitsubishi	LVD002484000832	LVD Contactor	180
	Exide	SLR011200W1435	Solar Panel YL-200Wp	204
	Wavecom	SWT001150373	Door switch	210
Utility	Telemechanic	RLY00411466	Relay DC MY2 48VDC, IEC255	152

Gambar 5.42. Tampilan *Output* Laporan Persediaan *Spare Part*

43. Tampilan Menu *Logout*

Setiap *user* dapat keluar dari program melalui menu *logout* yang ada di menu paling bawah.



Gambar 5.43. Tampilan Menu *Logout*

5.2. Pengujian

Pengujian sistem dilakukan untuk memeriksa kekompakan antar komponen sistem yang diimplementasikan.Tujuan utama dari pengujian sistem adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan.Berdasarkan skenario pengujian yang telah dirancang, dilakukan analisis terhadap pengujian yang telah dilakukan terhadap program yang dirancang.Analisis dilakukan dengan melihat hasil dari inputan terhadap sistem ini, apabila sesuai maka fungsi dinyatakan berhasil dan bila tidak sesuai maka dinyatakan tidak berhasil.

5.2.1. Rencana Pengujian

Pengujian perangkat lunak ini menggunakan metode pengujian *black box*. Hal ini dilakukan karena pengujian menggunakan *black box* berfokus pada persyaratan atau kebutuhan fungsional perangkat lunak yang dibuat (penguji dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program). Pengujian *black box* cenderung untuk menemukan hal-hal seperti berikut:

1. Fungsi yang tidak benar atau tidak ada
2. Kesalahan antarmuka
3. Kesalahan pada struktur data dan akses basis data
4. Kesalahan performansi
5. Kesalahan inisialisasi dan terminasi

Pengujian dilakukan dengan cara menjalankan program dan melakukan evaluasi terhadap fungsi-fungsi yang ada. Hasil dari pengujian akan menjadi acuan dalam menentukan tindakan atau kebijakan dalam merancang ulang program. Setelah mendapatkan hasil pengujian, dapat diambil kesimpulan terhadap rancangan program untuk mengetahui kelebihan dan kekurangan sistem.

Tabel 5.1. Rencana Pengujian

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Verifikasi Nama User	*Black Box*
	Verifikasi Password	*Black Box*
Master → User	Tambah User	*Black Box*
Transaksi → Permintaan Spare Part	Tambah Part Number	*Black Box*
Laporan	Laporan Persediaan Spare Part	*Black Box*

5.2.2. Kasus dan Hasil Pengujian

Pengujian program sistem informasi persediaan *spare part* ini dilakukan oleh staf pada bagian Spare Part Management (SPM). Adapun tabel pengujiannya dapat dilihat pada tabel di bawah ini:

Tabel 5.2. Pengujian Login *User*

Kasus dan Hasil Uji Login User (user: Spare Part Management)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
User: Daris Kadharisman	Dapat masuk ke menu selanjutnya	Dapat melakukan pengisian data secara benar dan yang diharapkannya	Berhasil
Password: daris0002	Dapat masuk ke menu selanjutnya	Dapat melakukan pengisian data secara benar dan yang diharapkannya	Berhasil
Klik tombol login	Dapat masuk ke menu selanjutnya	Tombol login dapat berfungsi sesuai dengan yang diharapkannya	Berhasil
User: Daris Khadarisman	Tidak dapat masuk ke menu selanjutnya	User tidak dapat login	Berhasil
Password: daris0002	Dapat masuk ke menu selanjutnya	User tidak dapat login	Tidak Berhasil
Klik tombol login	Tidak bisa masuk ke menu	User tidak dapat login	Berhasil

	selanjutnya		

Tabel 5.3. Pengujian Menu *User*

Kasus dan Hasil Uji Menu Master → User			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama: Test	Data dapat tersimpan	Data tersimpan	Berhasil
Password: Test0001	Data dapat tersimpan	Data tersimpan	Berhasil
Group_Login: *Tidak memilih*	Data dapat tersimpan	Data tidak muncul	Tidak Berhasil
No_Telepon: 000000	Data dapat tersimpan	Data tersimpan	Berhasil
Regional: Sumbagut	Data dapat tersimpan	Data tersimpan	Berhasil
Email: test@hdte.com	Data dapat tersimpan	Data tersimpan	Berhasil

Tabel 5.4. Pengujian Menu Permintaan *Spare Part*

Kasus dan Hasil Uji Menu Transaksi → Permintaan Spare Part			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
No.Form: *Auto increment*	Nomor muncul sesuai tahun dan bulan	Nomor muncul sesuai tahun dan bulan	Berhasil
Regional: Middle java	Data dapat tersimpan	Data tersimpan	Berhasil
Site: RPT Datalink Salem	Data dapat tersimpan	Data tersimpan	Berhasil
Pemohon: Test	Data dapat tersimpan	Data tersimpan	Berhasil
Part Number: FTR0021S1631	Data dapat tersimpan	Data tersimpan	Berhasil
Quantity: 1	Data dapat tersimpan	Data tersimpan	Berhasil

Tabel 5.5. Pengujian Menu Laporan

Kasus dan Hasil Uji Menu Laporan → Laporan Permintaan Spare Part			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tanggal Awal:	Tanggal muncul	Tanggal muncul	Berhasil

01/06/2012	sesuai yang dipilih	sesuai yang dipilih tahun dan bulan	
Tanggal Akhir: 30/06/2012	Tanggal muncul sesuai yang dipilih	Tanggal muncul sesuai yang dipilih	Berhasil
Jenis Laporan: Laporan Permintaan Spare Part	Data laporan muncul berupa PDF	Data laporan muncul berupa PDF	Berhasil

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan menggunakan metode pengujian *black box* terhadap sistem informasi persediaan *spare part* berbasis *web* di atas dapat diambil kesimpulan bahwa perangkat lunak secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan dan berfungsi sebagaimana mestinya.