

## BAB VI

### KESIMPULAN DAN SARAN

#### VI.1. Kesimpulan

Berdasarkan uraian dan pengujian yang telah dilakukan, kesimpulan dari pembangunan sistem ini adalah:

1. Sistem Informasi Spare Parts Terintegrasi Berbasis Web dengan menggunakan *framework* CodeIgniter telah berhasil dibangun dan setelah dilakukan pengujian, sistem informasi yang dibangun dikatakan handal.
2. Sistem Informasi Spare Parts Terintegrasi Berbasis Web telah berhasil dibangun dan berhasil dijalankan untuk mendukung proses bisnis yang dijalankan oleh PT. XYZ. Fungsi yang dikelola pada website ini yakni pengelolaan cabang, pegawai, kategori barang, barang, master barang, transaksi lunas, transaksi bon, pemesanan barang, penerimaan barang, retur transaksi lunas, retur transaksi bon, dan reporting.

#### VI.2. Saran

Walaupun pembangunan sistem sudah berjalan dengan baik, namun masih terdapat kekurangan yang dilakukan penulis. Maka dari itu, penulis memberikan saran:

1. Dilakukan pengembangan agar customer dapat melakukan pembelian barang secara online.

Program Studi Teknik Informatika	SKPL – SISP	259/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Dilakukan pengembangan agar vendor terhubung dengan sistem informasi ini untuk mempermudah pemesanan barang.



## DAFTAR PUSTAKA

- Abeka, S.O., 2012. User Satisfaction and Acceptance of Web Based Marketing Information System among Microfinance Institutions in Nairobi Region, Kenya. *International Journal of Academic Research in Business and Social Sciences*, 2(9), pp.368-83.
- Arifudzaki, B., Somantri, M. & FR, A., 2010. Aplikasi Sistem Informasi Persediaan Barang pada Perusahaan Ekspor Hasil Laut Berbasis Web. *TRANSMISI*, 12(4), pp.1-7.
- Bošnjaković, M., 2010. Multicriteria Inventory Model for Spare Parts. *Technical Gazette*, 17(4), pp.499-504.
- Curteanu, M., 2010. Using the Model-View-Controller for Creating Applications for Project Management. *Open Source Science Journal*, 2(4), pp.150-66.
- Daqiqil, I., 2011. [Online] Available at: [HYPERLINK "ftp://jaran.undip.ac.id/pustaka/Framework%20Codeigniter%20.pdf"](ftp://jaran.undip.ac.id/pustaka/Framework%20Codeigniter%20.pdf)  
<ftp://jaran.undip.ac.id/pustaka/Framework%20Codeigniter%20.pdf> [Accessed 18 Maret 2014].
- Hamzah, M.F., Machbob, M. & Ramdhon, Z., 2012. *Sistem Informasi Persediaan dan Penjualan Barang Berbasis Web pada Bengkel CK. Technic Tegal*. Yogyakarta: Sekolah Tinggi Manajemen Informatika dan Komputer Amikom.
- Handita, D.B., Umar & Fadillah, U., 2012. Sistem Informasi Penjualan Berbasis Web pada Usantex. *Jurnal Emitor*, 12(1), pp.26 - 33.

Program Studi Teknik Informatika	SKPL – SISP	261/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- Jia, W. & Zhou, W., 2005. *Distributed Network Systems From Concepts to Implementations*. Springer.
- Karim, A.N.M., Saad, M.F. & Haque, M., 2011. Development of a Prospective Web-Based Inventory System for. *Journal of Emerging Trends in Engineering and Applied Sciences (JETEAS)*, 2(1), pp.36-42.
- Mesquita, M.A.d. & do Rego, J.R., 2011. Spare parts inventory control. *Produção*, 21(4), pp.656-66.
- O'Brien, J.A. & Marakas, G.M., 2007. *Introduction to Information Systems*. 5th ed. Mc Graw Hill.
- Oliva, R., 2003. Managing The Transition From Products to Services. *IJSIM*, 14(2), pp.160-72.
- Paramarta, Y., 2012. *Sistem Informasi Penjualan Spare Part Berbasis Web pada Bengkel RSAC Motor Klaten*. Yogyakarta: Sekolah Tinggi Manajemen Informatika dan Komputer Amikom.
- Poerwanta, R., MT, Y. & Dr., E., 2013. Perancangan Sistem Inventory Spare Parts Mobil pada CV. Auto Parts Toyota Berbasis Aplikasi Java. *Jurnal TEKNOIF*, 1(2), pp.1 -5.
- Riyadi, A.S., Retnandi, E. & Deddy, A., 2012. Perancangan Sistem Informasi Berbasis Website Subsistem Guru di Sekolah Pesantren Persatuan Islam 99 Rancabango. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*, 9(40), pp.1-11.
- Sriyanto, Hartini, S. & Aldila, T.Y., 2011. Rancang Bangun Sistem Informasi E-Commerce Untuk Jaringan Penjualan Sepeda Motor Bekas Studi Kasus di Bedagan Motor Semarang. *Jurnal Ilmiah Ilmu Komputer*, 7(2), pp.231 - 236.

Program Studi Teknik Informatika	SKPL – SISP	262/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- Sugara, E.P.A., 2011. Sistem Informasi Pencarian Dan Penjualan Barang Berbasis Web Pada Toko Bagus. *Jurnal Teknologi dan Informatika*, 1(2), pp.158 - 166.
- Supaartagorn, C., 2011. PHP Framework For Database Management Based On MVC Pattern. *International Journal of Computer Science & Information Technology (IJCSIT)*, 3(2), pp.251-58.
- Turban, E., Rainer, R.K. & Potter, R.E., 2005. *Introduction to Information Technology*. John Wiley & Sons.
- Utpatadevi, N.L.P.P., Sudana, A.A.K.O. & Cahyawan, A.A.K.A., 2012. Implementation of MVC (Model-View-Controller) Architectural to Academic Management Information System with Android Platform Base. *International Journal of Computer Applications*, 57(8), pp.1-6.
- Wahana, K., 2006. *Apa dan Bagaimana E-Commerce*. Yogyakarta: Andi Offset.
- Wibisono, S., 2005. Enterprise Resource Planning (ERP) Solusi Sistem Informasi Terintegrasi. *Jurnal Teknologi Informasi DINAMIK*, 10(3), pp.150-59.
- Widayanto, A. & Wardati, I.U., 2013. Perancangan Sistem Informasi Penjualan Spare Part Mobil Pada Bengkel Samsi Motor Pacitan. *Indonesian Journal on Networking and Security*, 2(1), pp.1-7.
- Wiharjanto, Y., 2012. Perancangan Sistem Penjualan Tunai Berbasis Web Sebagai Sarana Informasi Produk Bagi Konsumen Pada PT. Warna AC. *Nominal*, 1(1), pp.6-11.

Program Studi Teknik Informatika	SKPL – SISP	263/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Yannopoulos, D.P., 2011. Impact of the Internet on Marketing Strategy Formulation. *International Journal of Business and Social Science*, 2(18), pp.1-7.





# LAMP IRAN

**SKPL**

**SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK**

**Pembangunan Sistem Informasi Spare Parts  
Terintegrasi Berbasis Web**

**(SISP)**

**Untuk :**

**Universitas Atma Jaya Yogyakarta**

**Dipersiapkan oleh:**

**Irene Deandra Indarto / 11. 07. 06483**

**Program Studi Teknik Informatika - Fakultas Teknologi  
Industri**

**Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika	<b>Nomor Dokumen</b>		<b>Halaman</b>
	Fakultas Teknologi Industri	<b>SKPL-SISP</b>		1/85
		<i>Revisi</i>		

Program Studi Teknik Informatika	SKPL – SISP	266/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

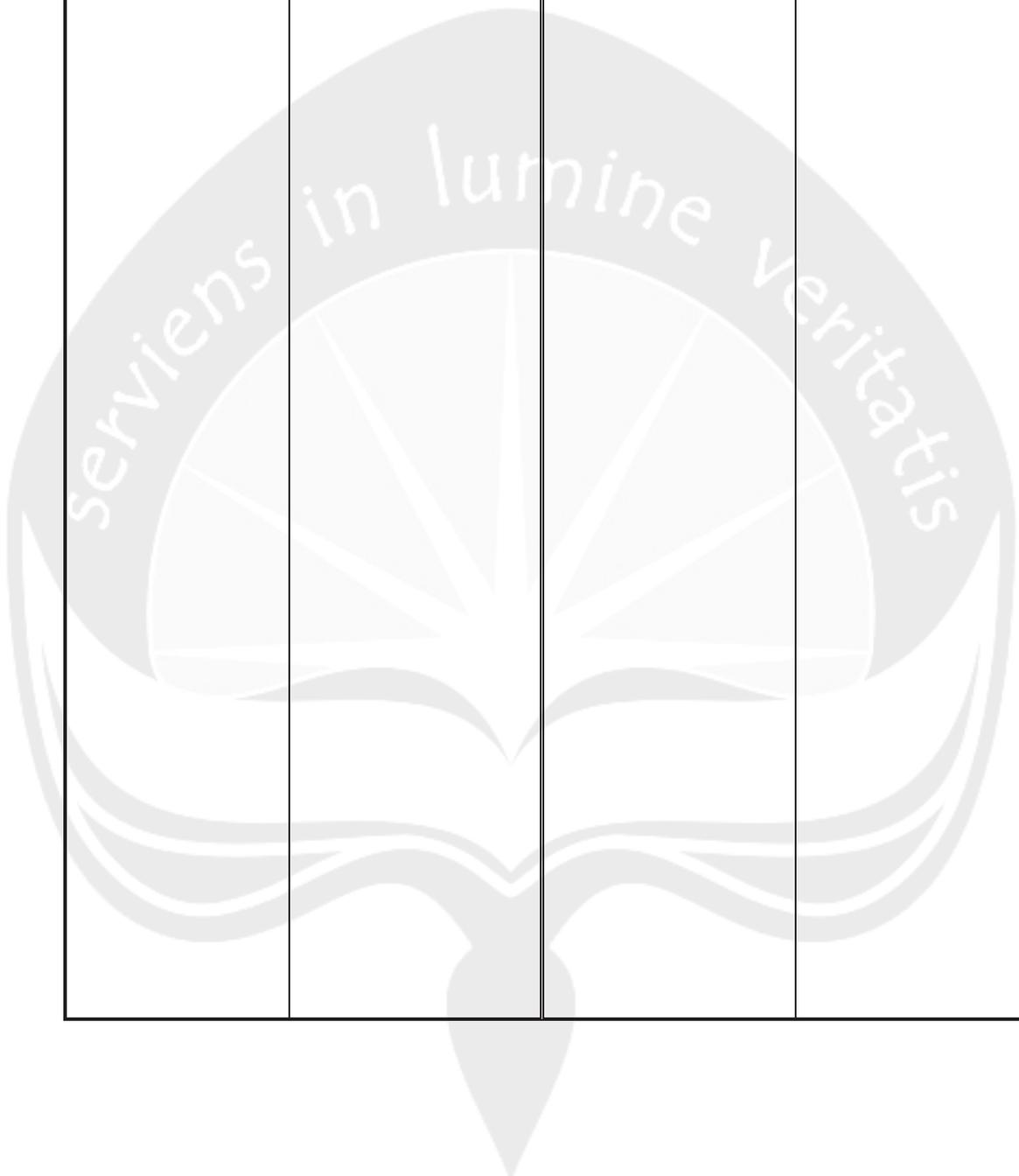
### DAFTAR PERUBAHAN

REVISI	DESKRIPSI
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F
DITULIS OLEH	IDI						
DIPERIKSA OLEH							
DISETUJUI OLEH							

## DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi

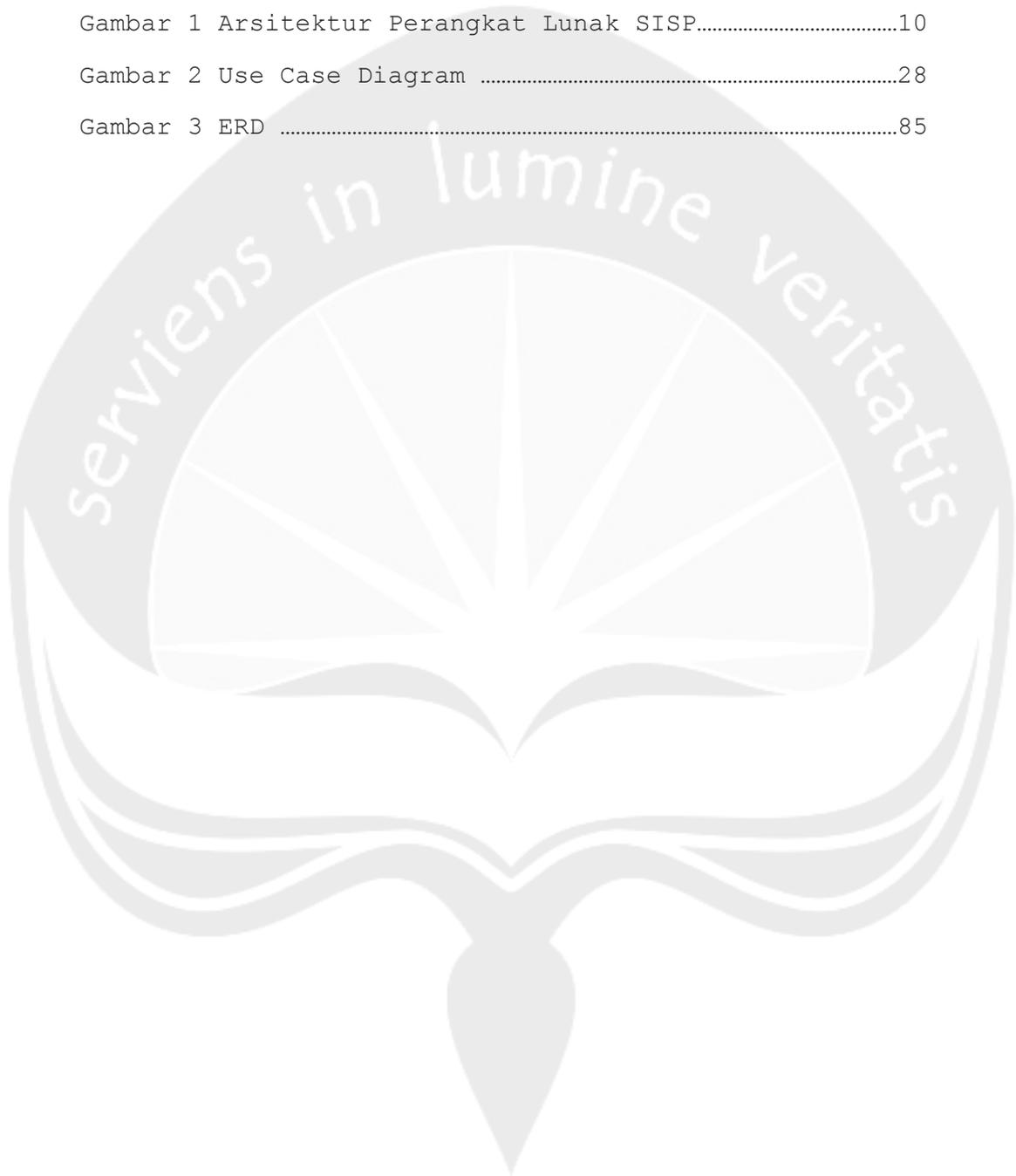


## DAFTAR ISI

1	PENDAHULUAN .....	271
1.1	Tujuan	271
1.2	Lingkup Masalah .....	271
1.3	Definisi, Akronim dan Singkatan .....	271
1.4	Referensi .....	273
1.5	Deskripsi umum (Overview) .....	273
2	DESKRIPSI KEBUTUHAN .....	274
2.1	Perspektif produk .....	274
2.2	Fungsi Produk .....	275
2.3	Karakteristik Pengguna .....	288
2.4	Batasan-batasan .....	288
2.5	Asumsi dan Ketergantungan .....	288
3	KEBUTUHAN KHUSUS .....	289
3.1	Kebutuhan antarmuka eksternal .....	289
3.2	Kebutuhan fungsionalitas Perangkat Lunak .....	291
4	SPESIFIKASI RINCI KEBUTUHAN .....	293
4.1	Spesifikasi Kebutuhan Fungsionalitas .....	293

## DAFTAR GAMBAR

Gambar 1 Arsitektur Perangkat Lunak SISP.....	10
Gambar 2 Use Case Diagram .....	28
Gambar 3 ERD .....	85



## Pendahuluan

### Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SISP (Sistem Informasi Spare Parts Terintegrasi Berbasis Web) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (feature-feature tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SISP ini juga mendefinisikan batasan perancangan perangkat lunak.

### Lingkup Masalah

Perangkat Lunak SISP dikembangkan dengan tujuan untuk:

1. Menangani pengelolaan yang terkait dengan cabang, vendor, pegawai, master barang, dan barang di PT. XYZ.
2. Menangani pengelolaan transaksi yang ada di PT. XYZ.
3. Menangani pengelolaan pemesanan dan penerimaan barang yang ada di PT. XYZ.
4. Menangani pengelolaan reporting transaksi penjualan di PT. XYZ.

Dan aplikasi web ini dapat berjalan dengan web browser.

### Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SISP-XXX	Kode yang merepresentasikan kebutuhan pada SISP (Sistem Informasi Spare Parts

Program Studi Teknik Informatika	SKPL – SISP	271/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

	Terintegrasi Berbasis Web) dimana XXX merupakan nomor fungsi produk.
SISP	Perangkat lunak pengelolaan web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
PT. XYZ	PT. XYZ adalah perusahaan yang bergerak di bidang penjualan spare parts serta aksesoris mobil dan memiliki beberapa cabang yang terletak di Yogyakarta.
Spare Parts	Bagian dari alat, mesin atau kendaraan yang disediakan untuk penggantian karena terjadinya keausan, kerusakan atau tidak berfungsinya komponen tersebut sesuai dengan standar kinerja yang diharapkan.
Transaksi	Persetujuan jual beli (dalam perdagangan) antara dua pihak.

Framework	Suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks.
PHP	Hypertext Preprocessor merupakan bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Code Igniter	Aplikasi open source yang berupa framework dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP.

## **Referensi**

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Thomas Hendri Hananto, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) SISU*, Universitas Atma Jaya Yogyakarta.
2. Dewi P. Palindih, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) WELSAP*, Universitas Atma Jaya Yogyakarta.

## **Deskripsi umum (Overview)**

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak SISP yang akan dikembangkan, mencakup perspektif

Program Studi Teknik Informatika	SKPL – SISP	273/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SISP tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SISP yang akan dikembangkan.

#### **Deskripsi Kebutuhan**

#### ***Perspektif produk***

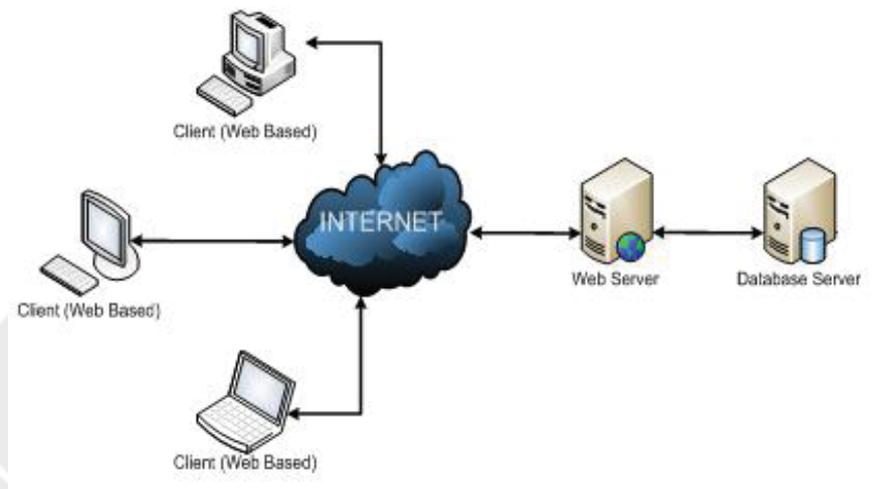
Sistem Informasi Spare Parts Terintegrasi Berbasis Web merupakan perangkat lunak yang dikembangkan untuk membantu pengelolaan di PT. XYZ. Sistem ini membantu PT. XYZ untuk mengelola proses bisnis yang berjalan di perusahaan yang bergerak di bidang penjualan spare parts tersebut. Sistem informasi ini dipergunakan oleh pihak internal dari PT. XYZ sendiri dan sistem ini berupa aplikasi web.

Sistem ini dapat digunakan oleh Kasir, Gudang, Administrator di setiap cabang PT. XYZ, serta Manager PT. XYZ. Melalui sistem tersebut, Kasir di PT. XYZ dapat melakukan pengelolaan transaksi lunas maupun bon, pengelolaan pemesanan barang, serta melakukan pengelolaan retur barang dari transaksi lunas maupun bon. Untuk Gudang dapat melakukan pengelolaan penerimaan barang, untuk Administrator dapat melakukan pengelolaan data cabang, pegawai, vendor, customer, kategori barang, master barang, dan barang, sedangkan Manager dapat melihat daftar transaksi lunas maupun bon, pemesanan barang, serta melihat reporting di website ini.

Website ini dapat berjalan pada web browser apapun, dan dibuat menggunakan bahasa pemrograman PHP dengan framework Code Igniter. Sedangkan untuk lingkungan pemrogramannya menggunakan Notepad++. Untuk *database*, digunakan MySQL.

Program Studi Teknik Informatika	SKPL – SISP	274/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika
--



**Gambar 1. Arsitektur Perangkat Lunak SISP**

### ***Fungsi Produk***

Fungsi produk perangkat lunak SISP adalah sebagai berikut :

**1. Fungsi *Login* (SKPL-SISP-001) .**

Merupakan fungsi yang digunakan oleh user dan untuk dapat masuk dalam sistem yang akan digunakan.

**2. Fungsi *Ubah Password* (SKPL-SISP-002) .**

Merupakan fungsi yang digunakan untuk mengubah password user.

**3. Fungsi *Pendaftaran User Baru* (SKPL-SISP-003) .**

Merupakan fungsi yang digunakan untuk melakukan pendaftaran user baru (*sign up*) untuk Kasir dan Gudang agar bisa melakukan login ke dalam sistem.

**4. Fungsi *Pengelolaan User* (SKPL-SISP-004) .**

Merupakan fungsi yang digunakan User untuk mengelola data user.

Fungsi *Pengelolaan User* meliputi :

a. Fungsi *Ubah Data User* (**SKPL-SISP-004-01**).

Merupakan fungsi yang digunakan untuk mengubah data user.

b. Fungsi *Tampil Data User* (**SKPL-SISP-004-02**).

Merupakan fungsi yang digunakan untuk menampilkan data user.

5. Fungsi *Pengelolaan Vendor* (**SKPL-SISP-005**).

Merupakan fungsi yang digunakan Administrator dan Kasir untuk mengelola data vendor PT. XYZ.

Fungsi *Pengelolaan Vendor* meliputi:

a. Fungsi *Tambah Vendor* (**SKPL-SISP-005-01**).

Merupakan fungsi yang digunakan untuk menambahkan data vendor.

b. Fungsi *Ubah Vendor* (**SKPL-SISP-005-02**).

Merupakan fungsi yang digunakan untuk mengubah data vendor.

c. Fungsi *Tampil Vendor* (**SKPL-SISP-005-03**).

Merupakan fungsi yang digunakan untuk menampilkan data vendor.

6. Fungsi *Pengelolaan Customer* (**SKPL-SISP-006**).

Merupakan fungsi yang digunakan Administrator dan Kasir untuk mengelola data customer PT. XYZ.

Fungsi *Pengelolaan Customer* meliputi:

a. Fungsi *Tambah Customer* (**SKPL-SISP-006-01**).

Program Studi Teknik Informatika	SKPL – SISP	276/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk menambahkan data customer.

b. Fungsi *Ubah Customer* (**SKPL-SISP-006-02**).

Merupakan fungsi yang digunakan untuk mengubah data customer.

c. Fungsi *Tampil Customer* (**SKPL-SISP-006-03**).

Merupakan fungsi yang digunakan untuk menampilkan data customer.

7. Fungsi *Pengelolaan Pegawai* (**SKPL-SISP-007**).

Merupakan fungsi yang digunakan Administrator untuk mengelola data pegawai PT. XYZ.

Fungsi *Pengelolaan Pegawai* meliputi:

a. Fungsi *Tambah Pegawai* (**SKPL-SISP-007-01**).

Merupakan fungsi yang digunakan untuk menambahkan data pegawai.

b. Fungsi *Ubah Pegawai* (**SKPL-SISP-007-02**).

Merupakan fungsi yang digunakan untuk mengubah data pegawai.

c. Fungsi *Hapus Pegawai* (**SKPL-SISP-007-03**).

Merupakan fungsi yang digunakan untuk menghapus data pegawai.

8. Fungsi *Pengelolaan Cabang* (**SKPL-SISP-008**).

Merupakan fungsi yang digunakan Administrator untuk mengelola data cabang PT. XYZ.

Fungsi *Pengelolaan Cabang* meliputi:

a. Fungsi *Tambah Cabang* (**SKPL-SISP-008-01**).

Merupakan fungsi yang digunakan untuk menambahkan data cabang.

Program Studi Teknik Informatika	SKPL – SISP	277/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

b. Fungsi *Ubah Cabang* (**SKPL-SISP-008-02**).

Merupakan fungsi yang digunakan untuk mengubah data cabang.

c. Fungsi *Hapus Cabang* (**SKPL-SISP-008-03**).

Merupakan fungsi yang digunakan untuk menghapus data cabang.

9. Fungsi *Pengelolaan Kategori Barang* (**SKPL-SISP-009**).

Merupakan fungsi yang digunakan Administrator untuk mengelola data kategori barang PT. XYZ.

Fungsi *Pengelolaan Kategori Barang* meliputi:

a. Fungsi *Tambah Kategori Barang* (**SKPL-SISP-009-01**).

Merupakan fungsi yang digunakan untuk menambahkan kategori barang.

b. Fungsi *Ubah Kategori Barang* (**SKPL-SISP-009-02**).

Merupakan fungsi yang digunakan untuk mengubah data kategori barang.

10. Fungsi *Pengelolaan Barang* (**SKPL-SISP-010**).

Merupakan fungsi yang digunakan Administrator untuk mengelola data barang PT. XYZ.

Fungsi *Pengelolaan Barang* meliputi:

a. Fungsi *Tambah Barang* (**SKPL-SISP-010-01**).

Merupakan fungsi yang digunakan untuk menambahkan data barang.

b. Fungsi *Ubah Barang* (**SKPL-SISP-010-02**).

Merupakan fungsi yang digunakan untuk mengubah data barang.

c. Fungsi *Hapus Barang* (**SKPL-SISP-010-03**).

Program Studi Teknik Informatika	SKPL – SISP	278/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk menghapus data barang.

**11. Fungsi *Pengelolaan Master Barang* (SKPL-SISP-011).**

Merupakan fungsi yang digunakan Administrator untuk mengelola data master barang dari vendor PT. XYZ.

Fungsi *Pengelolaan Master Barang* meliputi:

a. Fungsi *Tambah Master Barang* (SKPL-SISP-011-01).

Merupakan fungsi yang digunakan untuk menambahkan data master barang.

b. Fungsi *Ubah Master Barang* (SKPL-SISP-011-02).

Merupakan fungsi yang digunakan untuk mengubah data master barang.

c. Fungsi *Tampil Master Barang* (SKPL-SISP-011-03).

Merupakan fungsi yang digunakan untuk menampilkan data master barang.

d. Fungsi *Cari Master Barang* (SKPL-SISP-011-04).

Merupakan fungsi yang digunakan untuk mencari data barang berdasarkan nama master barang.

**12. Fungsi *Pengelolaan Transaksi Lunas* (SKPL-SISP-012).**

Merupakan fungsi yang digunakan Kasir untuk mengelola data transaksi lunas PT. XYZ.

Fungsi *Pengelolaan Transaksi Lunas* meliputi:

a. Fungsi *Generate Kode Transaksi Lunas* (SKPL-SISP-012-01).

Merupakan fungsi yang digunakan untuk melakukan generate kode transaksi lunas secara otomatis pada saat menambahkan transaksi lunas dengan format Cabang-Tanggal-Jumlah Transaksi pada hari itu.

b. Fungsi *Tambah Transaksi Lunas* (SKPL-SISP-012-02).

Program Studi Teknik Informatika	SKPL – SISP	279/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk menambahkan data transaksi lunas.

- c. Fungsi *Konfirmasi Transaksi Lunas* (**SKPL-SISP-012-03**).

Merupakan fungsi yang digunakan untuk melakukan konfirmasi data transaksi lunas.

- d. Fungsi *Batal Transaksi Lunas* (**SKPL-SISP-012-04**).

Merupakan fungsi yang digunakan untuk melakukan pembatalan terhadap suatu data transaksi lunas.

- e. Fungsi *Cetak Nota Transaksi Lunas* (**SKPL-SISP-012-05**).

Merupakan fungsi yang digunakan untuk menampilkan nota transaksi lunas.

- 13.** Fungsi *Pengelolaan Transaksi Bon* (**SKPL-SISP-013**).

Merupakan fungsi yang digunakan Kasir untuk mengelola data transaksi bon PT. XYZ.

Fungsi *Pengelolaan Transaksi Bon* meliputi:

- a. Fungsi *Generate Kode Transaksi Bon* (**SKPL-SISP-013-01**).

Merupakan fungsi yang digunakan untuk melakukan generate kode transaksi bon secara otomatis pada saat menambahkan transaksi bon dengan format Cabang-Tanggal-Jumlah Transaksi pada hari itu.

- b. Fungsi *Tambah Transaksi Bon* (**SKPL-SISP-013-02**).

Merupakan fungsi yang digunakan untuk menambahkan data transaksi bon.

- c. Fungsi *Konfirmasi Transaksi Bon* (**SKPL-SISP-013-03**).

Merupakan fungsi yang digunakan untuk melakukan konfirmasi data transaksi bon.

d. Fungsi *Batal Transaksi Bon* (**SKPL-SISP-013-04**).

Merupakan fungsi yang digunakan untuk melakukan pembatalan terhadap suatu data transaksi lunas.

e. Fungsi *Cetak Nota Transaksi Bon* (**SKPL-SISP-013-05**).

Merupakan fungsi yang digunakan untuk menampilkan nota transaksi bon.

f. Fungsi *Pelunasan Transaksi Bon* (**SKPL-SISP-013-06**).

Merupakan fungsi yang digunakan untuk melakukan pencatatan transaksi bon yang sudah dibayar (lunas).

**14.** Fungsi *Pengelolaan Pemesanan Barang* (**SKPL-SISP-014**).

Merupakan fungsi yang digunakan Kasir untuk mengelola data pemesanan barang PT. XYZ.

Fungsi *Pengelolaan Pemesanan Barang* meliputi:

a. Fungsi *Generate Kode Pemesanan Barang* (**SKPL-SISP-014-01**).

Merupakan fungsi yang digunakan untuk melakukan generate kode pemesanan barang secara otomatis pada saat menambahkan data pemesanan barang dengan format Cabang-Tanggal-Jumlah Pemesanan pada hari itu.

b. Fungsi *Tambah Pemesanan Barang* (**SKPL-SISP-014-02**).

Merupakan fungsi yang digunakan untuk menambahkan data pemesanan barang.

c. Fungsi *Batal Pemesanan Barang* (**SKPL-SISP-014-03**).

Merupakan fungsi yang digunakan untuk melakukan pembatalan terhadap suatu data pemesanan barang.

**15. Fungsi *Pengelolaan Detil Transaksi Lunas* (SKPL-SISP-015).**

Merupakan fungsi yang digunakan Kasir untuk mengelola data detil transaksi lunas PT. XYZ.

Fungsi *Pengelolaan Detil Transaksi Lunas* meliputi:

a. Fungsi *Tambah Detil Transaksi Lunas* (SKPL-SISP-015-01).

Merupakan fungsi yang digunakan untuk menambahkan data detil transaksi lunas.

b. Fungsi *Ubah Detil Transaksi Lunas* (SKPL-SISP-015-02).

Merupakan fungsi yang digunakan untuk mengubah data detil transaksi lunas.

c. Fungsi *Hapus Detil Transaksi Lunas* (SKPL-SISP-015-03).

Merupakan fungsi yang digunakan untuk menghapus data detil transaksi lunas.

d. Fungsi *Tampil Detil Transaksi Lunas* (SKPL-SISP-015-04).

Merupakan fungsi yang digunakan untuk menampilkan data detil transaksi lunas berdasarkan id transaksi.

**16. Fungsi *Pengelolaan Detil Transaksi Bon* (SKPL-SISP-016).**

Merupakan fungsi yang digunakan Kasir untuk mengelola data detil transaksi bon PT. XYZ.

Fungsi *Pengelolaan Detil Transaksi Bon* meliputi:

Program Studi Teknik Informatika	SKPL – SISP	282/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- a. Fungsi *Tambah Detil Transaksi Bon* (**SKPL-SISP-016-01**).

Merupakan fungsi yang digunakan untuk menambahkan data detil transaksi bon.

- b. Fungsi *Ubah Detil Transaksi Bon* (**SKPL-SISP-016-02**).

Merupakan fungsi yang digunakan untuk mengubah data detil transaksi bon.

- c. Fungsi *Hapus Detil Transaksi Bon* (**SKPL-SISP-016-03**).

Merupakan fungsi yang digunakan untuk menghapus data detil transaksi bon.

- d. Fungsi *Tampil Detil Transaksi Bon* (**SKPL-SISP-016-04**).

Merupakan fungsi yang digunakan untuk menampilkan data detil transaksi bon berdasarkan id transaksi.

17. Fungsi *Pengelolaan Detil Pemesanan Barang* (**SKPL-SISP-017**).

Merupakan fungsi yang digunakan Kasir untuk mengelola data detil pemesanan barang PT. XYZ.

Fungsi *Pengelolaan Detil Pemesanan Barang* meliputi:

- a. Fungsi *Tambah Detil Pemesanan Barang* (**SKPL-SISP-017-01**).

Merupakan fungsi yang digunakan untuk menambahkan data detil pemesanan barang.

- b. Fungsi *Ubah Detil Pemesanan Barang* (**SKPL-SISP-017-02**).

Merupakan fungsi yang digunakan untuk mengubah data detil pemesanan barang.

Program Studi Teknik Informatika	SKPL – SISP	283/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

c. Fungsi *Hapus Detil Pemesanan Barang* (**SKPL-SISP-017-03**).

Merupakan fungsi yang digunakan untuk menghapus data detil pemesanan barang.

d. Fungsi *Tampil Detil Pemesanan Barang* (**SKPL-SISP-017-04**).

Merupakan fungsi yang digunakan untuk menampilkan data detil pemesanan barang berdasarkan id pemesanan barang.

**18.** Fungsi *Retur Transaksi Lunas* (**SKPL-SISP-018**).

Merupakan fungsi yang digunakan Kasir untuk menambahkan data retur / pengembalian barang untuk transaksi lunas. Retur dapat dilakukan maksimal 2 hari setelah transaksi.

**19.** Fungsi *Retur Transaksi Bon* (**SKPL-SISP-019**).

Merupakan fungsi yang digunakan Kasir untuk menambahkan retur / pengembalian barang untuk transaksi bon. Retur dapat dilakukan maksimal 2 hari setelah transaksi.

**20.** Fungsi *Pengelolaan Penerimaan Barang* (**SKPL-SISP-020**).

Merupakan fungsi yang digunakan Gudang untuk melakukan pencatatan penerimaan barang PT. XYZ.

Fungsi *Pengelolaan Penerimaan Barang* meliputi:

a. Fungsi *Tambah Penerimaan Barang* (**SKPL-SISP-020-01**).

Merupakan fungsi yang digunakan untuk menambahkan data penerimaan barang.

b. Fungsi *Ubah Status Pemesanan Barang* (**SKPL-SISP-020-02**).

Program Studi Teknik Informatika	SKPL – SISP	284/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk mengubah status pesan dari pemesanan barang secara otomatis pada saat melakukan penerimaan barang. Jika barang sudah diterima namun belum lengkap akan diberi status 'NOTCOMPLETED', sedangkan bila barang sudah lengkap akan diberi status 'COMPLETED'.

**21. Fungsi Reporting (SKPL-SISP-021).**

Merupakan fungsi yang digunakan Manager untuk melihat laporan(reporting) PT. XYZ.

Fungsi Reporting meliputi:

a. Fungsi Laporan Penjualan Umum (SKPL-SISP-021-01).

Merupakan fungsi yang digunakan untuk melihat laporan penjualan secara umum untuk transaksi lunas dan transaksi bon di setiap cabang berdasarkan tanggal yang diinputkan.

b. Fungsi Laporan Penjualan Detil (SKPL-SISP-021-02).

Merupakan fungsi yang digunakan untuk melihat laporan penjualan secara detil untuk transaksi lunas dan transaksi bon di setiap cabang berdasarkan tanggal yang diinputkan.

c. Fungsi Laporan Bon (SKPL-SISP-021-03).

Merupakan fungsi yang digunakan untuk melihat laporan daftar bon di setiap cabang berdasarkan tanggal yang diinputkan.

d. Fungsi Laporan Daftar Barang Habis (SKPL-SISP-021-04).

Merupakan fungsi yang digunakan untuk melihat laporan daftar barang habis di setiap cabang.

e. Fungsi Grafik Laporan Penjualan (SKPL-SISP-021-05).

Program Studi Teknik Informatika	SKPL – SISP	285/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk melihat laporan penjualan transaksi lunas dan transaksi bon berdasarkan rentang tanggal yang diinputkan user dalam bentuk grafik.

**22. Fungsi *Tampil Pegawai* (SKPL-SISP-022).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk menampilkan data pegawai.

**23. Fungsi *Tampil Cabang* (SKPL-SISP-023).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk menampilkan data cabang.

**24. Fungsi *Tampil Kategori Barang* (SKPL-SISP-024).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk menampilkan data kategori barang.

**25. Fungsi *Tampil Barang* (SKPL-SISP-025).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk menampilkan semua data barang.

**26. Fungsi *Tampil Barang By Kategori* (SKPL-SISP-026).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk menampilkan data barang berdasarkan kategori yang dipilih.

**27. Fungsi *Cari Barang* (SKPL-SISP-027).**

Merupakan fungsi yang digunakan Administrator, Kasir, Gudang, dan Manager untuk mencari data barang berdasarkan nama barang.

**28. Fungsi Tampil Transaksi Lunas (SKPL-SISP-028).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk menampilkan data transaksi lunas.

**29. Fungsi Tampil Info Transaksi Lunas (SKPL-SISP-029).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk menampilkan informasi dari suatu data transaksi lunas.

**30. Fungsi Cari Transaksi Lunas (SKPL-SISP-030).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk mencari data transaksi lunas berdasarkan tanggal.

**31. Fungsi Tampil Transaksi Bon (SKPL-SISP-031).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk menampilkan data transaksi bon.

**32. Fungsi Tampil Info Transaksi Bon (SKPL-SISP-032).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk menampilkan informasi dari suatu data transaksi bon.

**33. Fungsi Cari Transaksi Bon (SKPL-SISP-033).**

Merupakan fungsi yang digunakan Administrator, Kasir, dan Manager untuk mencari data transaksi bon berdasarkan tanggal.

**34. Fungsi Tampil Pemesanan Barang (SKPL-SISP-034).**

Merupakan fungsi yang digunakan Kasir, Gudang, dan Manager untuk menampilkan data pemesanan barang.

**35. Fungsi Tampil Info Pemesanan Barang (SKPL-SISP-035).**

Program Studi Teknik Informatika	SKPL – SISP	287/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan Kasir, Gudang, dan Manager untuk menampilkan informasi dari suatu data pemesanan barang.

**36. Fungsi Cari Pemesanan Barang (SKPL-SISP-036).**

Merupakan fungsi yang digunakan Kasir, Gudang, dan Manager untuk mencari data pemesanan barang berdasarkan tanggal.

**37. Fungsi Tampil Penerimaan Barang (SKPL-SISP-037).**

Merupakan fungsi yang digunakan Manager untuk menampilkan data penerimaan barang.

**Karakteristik Pengguna**

Karakteristik dari pengguna perangkat lunak SISP adalah sebagai berikut :

1. Memahami pengoperasian Microsoft Windows.
2. Mengerti tentang internet dan web.
3. Mengerti perangkat lunak yang digunakan.

**Batasan-batasan**

Batasan-batasan dalam pengembangan perangkat lunak SISP tersebut adalah :

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak SISP.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

**Asumsi dan Ketergantungan**

Sistem ini dapat dijalankan pada web browser apapun.

Program Studi Teknik Informatika	SKPL – SISP	288/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## **Kebutuhan khusus**

### ***Kebutuhan antarmuka eksternal***

Kebutuhan antar muka eksternal pada perangkat lunak SISP meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

#### **Antarmuka pemakai**

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk halaman web.

#### **Antarmuka perangkat keras**

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SISP adalah:

1. Perangkat PC/Laptop/HP.
2. Perangkat Database Server.
3. Perangkat Web Server.

#### **Antarmuka perangkat lunak**

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak SISP adalah sebagai berikut :

1. Nama : MySQL  
Sumber : Oracle.

Sebagai database management system (DBMS) yang digunakan untuk menyimpan data di sisi server.

2. Nama : Windows XP/Vista/7/8  
Sumber : Microsoft.

Sebagai sistem operasi yang digunakan.

Program Studi Teknik Informatika	SKPL – SISP	289/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Nama : Apache  
Sumber : Apache Software Foundation.

Sebagai web server.

4. Nama : Code Igniter  
Sumber : EllisLab, Inc.

Sebagai framework untuk menjalankan aplikasi.

5. Nama : IE/Firefox/Chrome/dll  
Sumber : Microsoft/Mozilla/Google/dll

Sebagai web browser untuk membuka sistem web.

6. Nama : Notepad++  
Sumber : Don Ho.

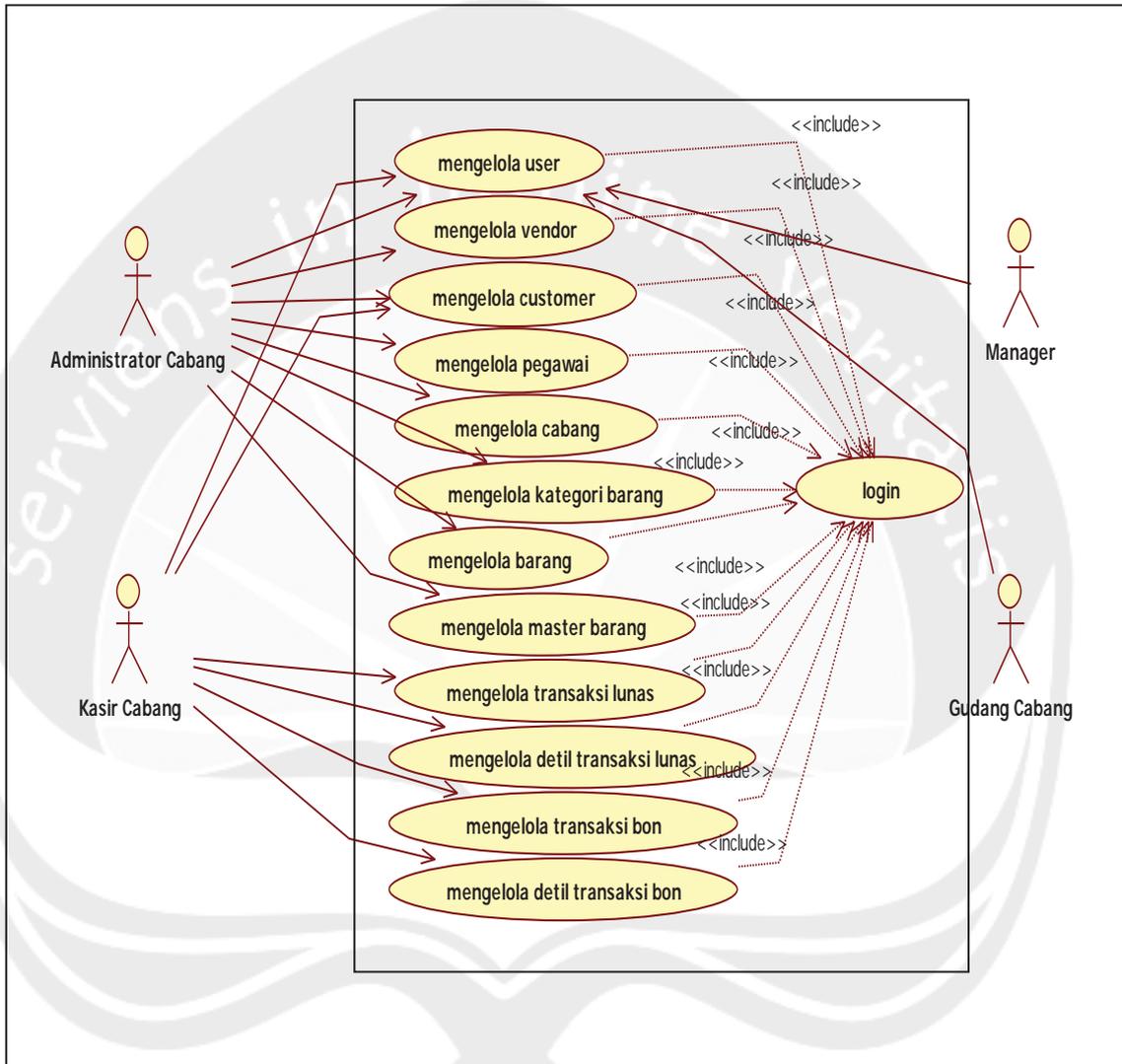
Sebagai aplikasi untuk editor program.

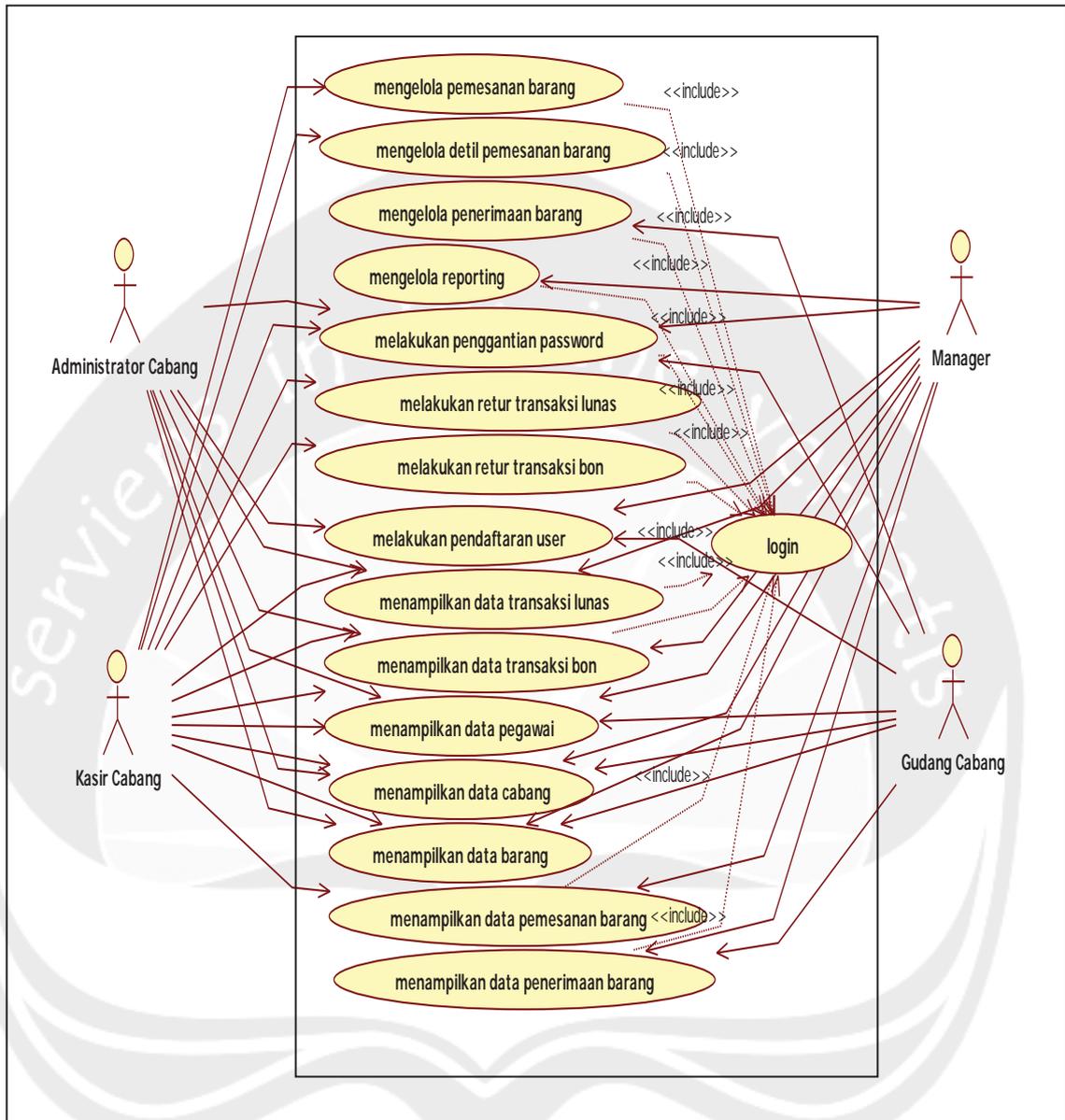
### **Antarmuka Komunikasi**

Antarmuka komunikasi perangkat lunak SISP menggunakan protocol HTTP.

## Kebutuhan fungsionalitas Perangkat Lunak

### 3.2.1 Use Case Diagram





Gambar 2. Use Case Diagram

## Spesifikasi Rinci Kebutuhan

### **Spesifikasi Kebutuhan Fungsionalitas**

#### **4.1.1 Use case Spesification : Login**

##### **1. Brief Description**

Use case ini digunakan untuk memperoleh akses masuk ke sistem yang digunakan oleh Administrator dan Kasir yang telah melakukan pendaftaran (*sign up*).

##### **2. Primary Actor**

1. Semua User

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login.
2. Sistem menampilkan antarmuka untuk login.
3. Aktor menginputkan username dan password.
4. Sistem melakukan validasi terhadap username dan password yang telah diinputkan.

E-1 Username atau password yang diinputkan aktor salah.

5. Sistem memberikan akses masuk ke dalam sistem.

6. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

E-1 Username atau password yang diinputkan aktor salah

Program Studi Teknik Informatika	SKPL – SISP	293/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-3.

#### **7. PreConditions**

1. Aktor telah melakukan pendaftaran (*sign up*).

#### **8. PostConditions**

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem sesuai role yang dimilikinya.

#### **4.1.2 Use case Spesification : Ubah Password**

##### **1. Brief Description**

Use case ini digunakan oleh aktor untuk mengubah password.

##### **2. Primary Actor**

1. Semua User

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengubah password.
2. Sistem menampilkan antarmuka untuk ubah password.
3. Aktor menginputkan password lama dan password baru.
4. Sistem melakukan validasi terhadap password lama dan password baru yang telah diinputkan.  
E-1 Password yang diinputkan aktor salah.
5. Sistem menyimpan password baru ke database.
6. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

Program Studi Teknik Informatika	SKPL – SISP	294/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-1 Password yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Password user telah terupdate di database.

#### **4.1.3 Use case Spesification : Pendaftaran User Baru**

##### **1. Brief Description**

Use case ini digunakan oleh aktor untuk melakukan pendaftaran user baru (*sign up*).

##### **2. Primary Actor**

1. Kasir
2. Gudang

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pendaftaran user baru.
2. Aktor menginputkan data user.
3. Aktor meminta sistem untuk menyimpan data user yang telah diinputkan.
4. Sistem melakukan pengecekan data user yang telah diinputkan.

E-1 Data user yang diinputkan aktor salah

Program Studi Teknik Informatika	SKPL – SISP	295/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Sistem menyimpan data user ke database.
6. Use case selesai.

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Data user yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-2.

#### **7. PreConditions**

none

#### **8. PostConditions**

1. Data user telah terdaftar di database.

### **4.1.4 Use case Spesification : Pengelolaan User**

#### **1. Brief Description**

Use case ini digunakan oleh aktor untuk mengelola data user, yaitu aktor dapat melakukan ubah data user dan tampil data user.

#### **2. Primary Actor**

1. Semua User

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data user.
2. Sistem memberikan pilihan untuk melakukan ubah data user dan tampil data user.

Program Studi Teknik Informatika	SKPL – SISP	296/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Aktor memilih untuk melakukan ubah data user
  - A-1 Aktor memilih untuk melakukan tampil data user
4. Sistem menampilkan data user.
5. Aktor mengubah data dan meminta sistem untuk menyimpan data user yang telah diubah.
6. Sistem melakukan pengecekan terhadap data user yang telah diubah.
  - E-1 Data user yang telah diubah aktor salah
7. Sistem menyimpan data user yang telah diubah ke database.
8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan tampil data user

- 1.Sistem menampilkan data user.
- 2.Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data user yang telah diubah aktor salah

- 1.Sistem memberi peringatan bahwa data yang diinputkan salah.
- 2.Kembali ke Basic Flow langkah ke-5.

#### **7. PreConditions**

none

#### **8. PostConditions**

1. Data user telah terupdate di database.

### **4.1.5 Use case Spesification : Pengelolaan Vendor**

#### **1. Brief Description**

Program Studi Teknik Informatika	SKPL – SISP	297/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

Use case ini digunakan oleh Administrator dan Kasir untuk mengelola data vendor. Aktor dapat melakukan tambah vendor, ubah data vendor, dan tampil data vendor.

## **2. Primary Actor**

1. Administrator
2. Kasir

## **3. Supporting Actor**

none

## **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data vendor.
2. Sistem memberikan pilihan untuk melakukan tambah vendor, ubah data vendor, dan tampil data vendor.
3. Aktor memilih untuk melakukan tambah vendor.  
A-1 Aktor memilih untuk melakukan ubah data vendor  
A-2 Aktor memilih untuk melakukan tampil data vendor
4. Aktor menginputkan data vendor.
5. Aktor meminta sistem untuk menyimpan data vendor yang telah diinputkan.
6. Sistem melakukan pengecekan data vendor yang telah diinputkan.  
E-1 Data vendor yang diinputkan aktor salah
7. Sistem menyimpan data vendor ke database.
8. Use case selesai.

## **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data vendor

Program Studi Teknik Informatika	SKPL – SISP	298/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan data vendor.
2. Aktor memilih data vendor yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data vendor yang telah diubah.
4. Sistem melakukan pengecekan terhadap data vendor yang telah diubah.

E-2 Data vendor yang telah diubah aktor salah

5. Sistem menyimpan data vendor yang telah diubah ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data vendor

1. Sistem menampilkan data vendor.
2. Berlanjut ke Basic Flow langkah ke-8.

## **6. Error Flow**

E-1 Data vendor yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data vendor yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

## **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – SISP	299/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 8. PostConditions

1. Data vendor telah terupdate di database.

### 4.1.6 Use case Spesification : Pengelolaan Customer

#### 1. Brief Description

Use case ini digunakan oleh Administrator dan Kasir untuk mengelola data customer. Aktor dapat melakukan tambah customer, ubah data customer, dan tampil data customer.

#### 2. Primary Actor

1. Administrator
2. Kasir

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data customer.
2. Sistem memberikan pilihan untuk melakukan tambah customer, ubah data customer, dan tampil data customer.
3. Aktor memilih untuk melakukan tambah customer.  
A-1 Aktor memilih untuk melakukan ubah data customer  
  
A-2 Aktor memilih untuk melakukan tampil data customer
4. Aktor menginputkan data customer.
5. Aktor meminta sistem untuk menyimpan data customer yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – SISP	300/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem melakukan pengecekan data customer yang telah diinputkan.

E-1 Data customer yang diinputkan aktor salah

7. Sistem menyimpan data customer ke database.

8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data customer

1. Sistem menampilkan data customer.

2. Aktor memilih customer yang akan diubah.

3. Aktor mengubah data dan meminta sistem untuk menyimpan data customer yang telah diubah.

4. Sistem melakukan pengecekan terhadap data customer yang telah diubah.

E-2 Data customer yang telah diubah aktor

salah

5. Sistem menyimpan data customer yang telah diubah ke database.

6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data customer

1. Sistem menampilkan data customer.

2. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data customer yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data customer yang telah diubah aktor salah

Program Studi Teknik Informatika	SKPL – SISP	301/ 523
----------------------------------	-------------	----------

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

**7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

**8. PostConditions**

1. Data customer telah terupdate di database.

**4.1.7 Use case Spesification : Pengelolaan Pegawai**

**1. Brief Description**

Use case ini digunakan oleh Administrator untuk mengelola data pegawai. Aktor dapat melakukan tambah pegawai, ubah data pegawai, dan hapus data pegawai.

**2. Primary Actor**

1. Administrator

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pegawai.
2. Sistem memberikan pilihan untuk melakukan tambah pegawai, ubah data pegawai, dan hapus data pegawai.
3. Aktor memilih untuk melakukan tambah pegawai.
  - A-1 Aktor memilih untuk melakukan ubah data pegawai
  - A-2 Aktor memilih untuk melakukan hapus data pegawai
4. Aktor menginputkan data pegawai.
5. Aktor meminta sistem untuk menyimpan data pegawai yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – SISP	302/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem melakukan pengecekan data pegawai yang telah diinputkan.

E-1 Data pegawai yang diinputkan aktor salah

7. Sistem menyimpan data pegawai ke database.

8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data pegawai

1.Sistem menampilkan data pegawai.

2.Aktor memilih pegawai yang akan diubah.

3.Aktor mengubah data dan meminta sistem untuk menyimpan data pegawai yang telah diubah.

4.Sistem melakukan pengecekan terhadap data pegawai yang telah diubah.

E-2 Data pegawai yang telah diubah aktor

salah

5.Sistem menyimpan data pegawai yang telah diubah ke database.

6.Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data pegawai

1.Sistem menampilkan data pegawai yang ingin dihapus.

2.Aktor menghapus data pegawai.

3.Sistem menyimpan data pegawai ke database.

4.Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data pegawai yang diinputkan aktor salah

Program Studi Teknik Informatika	SKPL – SISP	303/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
  2. Kembali ke Basic Flow langkah ke-4.
- E-2 Data pegawai yang telah diubah aktor salah
1. Sistem memberi peringatan bahwa data yang diubah salah.
  2. Kembali ke Alternative Flow A-1 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data pegawai telah terupdate di database.

### **4.1.8 Use case Spesification : Pengelolaan Cabang**

#### **1. Brief Description**

Use case ini digunakan oleh Administrator untuk mengelola data cabang. Aktor dapat melakukan tambah cabang, ubah data cabang, dan hapus data cabang.

#### **2. Primary Actor**

1. Administrator

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data cabang.
2. Sistem memberikan pilihan untuk melakukan tambah cabang, ubah data cabang, dan hapus data cabang.
3. Aktor memilih untuk melakukan tambah cabang.
  - A-1 Aktor memilih untuk melakukan ubah data cabang

Program Studi Teknik Informatika	SKPL – SISP	304/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-2 Aktor memilih untuk melakukan hapus data cabang

4. Aktor menginputkan data cabang.
5. Aktor meminta sistem untuk menyimpan data cabang yang telah diinputkan.
6. Sistem melakukan pengecekan data cabang yang telah diinputkan.

E-1 Data cabang yang diinputkan aktor salah

7. Sistem menyimpan data cabang ke database.
8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data cabang

1. Sistem menampilkan data cabang.
2. Aktor memilih cabang yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data cabang yang telah diubah.
4. Sistem melakukan pengecekan terhadap data cabang yang telah diubah.

E-2 Data cabang yang telah diubah aktor salah

5. Sistem menyimpan data cabang yang telah diubah ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data cabang

1. Sistem menampilkan data cabang yang ingin dihapus.
2. Aktor menghapus data cabang.
3. Sistem menyimpan data cabang ke database.

Program Studi Teknik Informatika	SKPL – SISP	305/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. Berlanjut ke Basic Flow langkah ke-8.

## 6. Error Flow

E-1 Data cabang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data cabang yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

## 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data cabang telah terupdate di database.

### 4.1.9 Use case Spesification : Pengelolaan Kategori Barang

#### 1. Brief Description

Use case ini digunakan oleh Administrator untuk mengelola data kategori barang. Aktor dapat melakukan tambah kategori barang, dan ubah data kategori barang.

#### 2. Primary Actor

1. Administrator

#### 3. Supporting Actor

none

Program Studi Teknik Informatika	SKPL – SISP	306/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data kategori barang.
2. Sistem memberikan pilihan untuk melakukan tambah kategori barang, dan ubah data kategori barang.
3. Aktor memilih untuk melakukan tambah kategori barang.  
A-1 Aktor memilih untuk melakukan ubah data kategori barang
4. Aktor menginputkan data kategori barang.
5. Aktor meminta sistem untuk menyimpan data kategori barang yang telah diinputkan.
6. Sistem melakukan pengecekan data kategori barang yang telah diinputkan.  
E-1 Data kategori barang yang diinputkan aktor salah
7. Sistem menyimpan data kategori barang ke database.
8. Use case selesai.

#### 5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data kategori barang
1. Sistem menampilkan data kategori barang.
  2. Aktor memilih kategori barang yang akan diubah.
  3. Aktor mengubah data dan meminta sistem untuk menyimpan data kategori barang yang telah diubah.

4. Sistem melakukan pengecekan terhadap data kategori barang yang telah diubah.

E-2 Data kategori barang yang telah diubah aktor salah

5. Sistem menyimpan data kategori barang yang telah diubah ke database.

6. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data kategori barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data kategori barang yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.

2. Kembali ke Alternative Flow A-1 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data kategori barang telah terupdate di database.

### **4.1.10 Use case Spesification : Pengelolaan Barang**

#### **1. Brief Description**

Use case ini digunakan oleh Administrator untuk mengelola data barang. Aktor dapat melakukan tambah barang, ubah data barang, dan hapus data barang.

#### **2. Primary Actor**

Program Studi Teknik Informatika	SKPL – SISP	308/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Administrator

### 3. Supporting Actor

none

### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data barang.
2. Sistem memberikan pilihan untuk melakukan tambah barang, ubah data barang, dan hapus data barang.
3. Aktor memilih untuk melakukan tambah barang.  
A-1 Aktor memilih untuk melakukan ubah data barang  
A-2 Aktor memilih untuk melakukan hapus data barang
4. Aktor menginputkan data barang.
5. Aktor meminta sistem untuk menyimpan data barang yang telah diinputkan.
6. Sistem melakukan pengecekan data barang yang telah diinputkan.  
E-1 Data barang yang diinputkan aktor salah
7. Sistem menyimpan data barang ke database.
8. Use case selesai.

### 5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data barang
1. Sistem menampilkan data barang.
  2. Aktor memilih barang yang akan diubah.
  3. Aktor mengubah data dan meminta sistem untuk menyimpan data barang yang telah diubah.
  4. Sistem melakukan pengecekan terhadap data barang yang telah diubah.

E-2 Data barang yang telah diubah aktor

Program Studi Teknik Informatika	SKPL – SISP	309/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

salah

5. Sistem menyimpan data barang yang telah diubah ke database.

6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data barang

1. Sistem menampilkan data barang yang ingin dihapus.

2. Aktor menghapus data barang.

3. Sistem menyimpan data barang ke database.

4. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data barang yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.

2. Kembali ke Alternative Flow A-1 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data barang telah terupdate di database.

### **4.1.11 Use case Spesification : Pengelolaan Master Barang**

#### **1. Brief Description**

Program Studi Teknik Informatika	SKPL – SISP	310/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Use case ini digunakan oleh Administrator untuk mengelola data master barang. Aktor dapat melakukan tambah master barang, ubah data master barang, tampil data master barang, dan cari data master barang.

## **2. Primary Actor**

1. Administrator

## **3. Supporting Actor**

none

## **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data master barang.
2. Sistem memberikan pilihan untuk melakukan tambah master barang, ubah data barang, tampil data barang, dan cari data barang.
3. Aktor memilih untuk melakukan tambah master barang.  
A-1 Aktor memilih untuk melakukan ubah data master barang  
A-2 Aktor memilih untuk melakukan tampil data master barang  
A-3 Aktor memilih untuk melakukan cari data master barang
4. Aktor menginputkan data master barang.
5. Aktor meminta sistem untuk menyimpan data master barang yang telah diinputkan.
6. Sistem melakukan pengecekan data master barang yang telah diinputkan.  
E-1 Data master barang yang diinputkan aktor salah
7. Sistem menyimpan data master barang ke database.
8. Use case selesai.

Program Studi Teknik Informatika	SKPL – SISP	311/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data master barang

1. Sistem menampilkan data master barang.
2. Aktor memilih master barang yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data master barang yang telah diubah.
4. Sistem melakukan pengecekan terhadap data master barang yang telah diubah.

E-2 Data master barang yang telah diubah aktor salah

5. Sistem menyimpan data master barang yang telah diubah ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data master barang

1. Sistem menampilkan data master barang.
2. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan cari data master barang

1. Aktor menginputkan nama master barang yang akan dicari.
2. Sistem menampilkan data master barang.
3. Berlanjut ke Basic Flow langkah ke-8.

## 6. Error Flow

E-1 Data master barang yang diinputkan aktor salah

Program Studi Teknik Informatika	SKPL – SISP	312/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data master barang yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.

2. Kembali ke Alternative Flow A-1 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data master barang telah terupdate di database.

#### **4.1.12 Use case Spesification : Pengelolaan Transaksi Lunas**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir untuk mengelola data transaksi bon. Aktor dapat melakukan tambah transaksi lunas, konfirmasi transaksi lunas, batal transaksi lunas, dan cetak nota transaksi lunas.

##### **2. Primary Actor**

1. Kasir

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data transaksi lunas.

Program Studi Teknik Informatika	SKPL – SISP	313/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Sistem memberikan pilihan untuk melakukan tambah transaksi lunas, konfirmasi transaksi lunas, batal transaksi lunas, dan cetak nota transaksi lunas.
3. Aktor memilih untuk melakukan tambah transaksi lunas.
  - A-1 Aktor memilih untuk melakukan konfirmasi transaksi lunas
  - A-2 Aktor memilih untuk melakukan batal transaksi lunas
  - A-3 Aktor memilih untuk melakukan cetak nota transaksi lunas
4. Sistem melakukan generate kode dan tanggal transaksi secara otomatis.
5. Sistem menyimpan data transaksi lunas ke database.
6. Use case selesai.

#### **5. Alternative Flow**

- A-1 Aktor memilih untuk melakukan konfirmasi transaksi lunas
1. Sistem menampilkan data transaksi lunas.
  2. Aktor menginputkan data konfirmasi transaksi lunas.
  3. Aktor menginputkan data dan meminta sistem untuk menyimpan data transaksi lunas yang diinputkan.
  4. Sistem melakukan pengecekan terhadap data transaksi lunas yang diinputkan.
- E-1 Data transaksi lunas yang diinputkan aktor salah

5. Sistem menyimpan data transaksi lunas yang telah diinputkan ke database.

6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan batal transaksi lunas

1. Sistem menampilkan data transaksi lunas.

2. Aktor melakukan pembatalan data transaksi lunas.

3. Sistem menghapus data transaksi lunas beserta detil transaksi lunas dalam database.

4. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan cetak nota transaksi lunas

1. Sistem menampilkan nota transaksi lunas.

2. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data transaksi lunas yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

#### **7. PreConditions**

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data transaksi lunas telah terupdate di database.

### **4.1.13 Use case Spesification : Pengelolaan Transaksi Bon**

#### **1. Brief Description**

Program Studi Teknik Informatika	SKPL – SISP	315/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Use case ini digunakan oleh Kasir untuk mengelola data transaksi bon. Aktor dapat melakukan tambah transaksi bon, konfirmasi transaksi bon, batal transaksi bon, cetak nota transaksi bon, dan pelunasan transaksi bon.

## **2. Primary Actor**

1. Kasir

## **3. Supporting Actor**

none

## **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data transaksi bon.
2. Sistem memberikan pilihan untuk melakukan tambah transaksi bon, konfirmasi transaksi bon, batal transaksi bon, cetak nota transaksi bon, dan pelunasan transaksi bon.
3. Aktor memilih untuk melakukan tambah transaksi bon.  
A-1 Aktor memilih untuk melakukan konfirmasi transaksi bon  
A-2 Aktor memilih untuk melakukan batal transaksi bon  
A-3 Aktor memilih untuk melakukan cetak nota transaksi bon  
A-4 Aktor memilih untuk melakukan pelunasan transaksi bon
4. Sistem melakukan generate kode dan tanggal transaksi secara otomatis.
5. Sistem menyimpan data transaksi bon ke database.

6. Use case selesai.

## 5. Alternative Flow

A-1 Aktor memilih untuk melakukan konfirmasi transaksi bon

1. Sistem menampilkan data transaksi bon.
2. Aktor menginputkan data konfirmasi transaksi bon.
3. Aktor menginputkan data dan meminta sistem untuk menyimpan data transaksi bon yang diinputkan.
4. Sistem melakukan pengecekan terhadap data transaksi bon yang diinputkan.

E-1 Data transaksi bon yang diinputkan actor salah

5. Sistem menyimpan data transaksi bon yang telah diinputkan ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan batal transaksi bon

1. Sistem menampilkan data transaksi bon.
2. Aktor melakukan pembatalan data transaksi bon.
3. Sistem menghapus data transaksi bon beserta detail transaksi bon dalam database.
4. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan cetak nota transaksi bon

1. Sistem menampilkan nota transaksi bon.
2. Berlanjut ke Basic Flow langkah ke-8.

A-4 Aktor memilih untuk melakukan pelunasan transaksi bon

Program Studi Teknik Informatika	SKPL – SISP	317/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan daftar transaksi bon.
2. Aktor memilih transaksi bon.
3. Sistem menampilkan data transaksi bon.
4. Aktor menginputkan data pelunasan transaksi bon.  
E-2 Data pelunasan transaksi bon yang diinputkan aktor salah
5. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data transaksi bon yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data pelunasan transaksi bon yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Alternative Flow A-7 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data transaksi bon telah terupdate di database.

### **4.1.14 Use case Spesification : Pengelolaan Pemesanan Barang**

#### **1. Brief Description**

Program Studi Teknik Informatika	SKPL – SISP	318/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Use case ini digunakan oleh Kasir untuk mengelola data pemesanan barang. Aktor dapat melakukan tambah pemesanan barang dan batal pemesanan barang.

## **2. Primary Actor**

1. Kasir

## **3. Supporting Actor**

none

## **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pemesanan barang.
2. Sistem memberikan pilihan untuk melakukan tambah pemesanan barang dan batal pemesanan barang.
3. Aktor memilih untuk membuat pemesanan barang baru.  
A-1 Aktor memilih untuk melakukan batal pemesanan barang
4. Sistem melakukan generate kode dan tanggal pemesanan barang secara otomatis.
5. Sistem menyimpan data pemesanan barang ke database.
6. Use case selesai.

## **5. Alternative Flow**

- A-1 Aktor memilih untuk melakukan batal pemesanan barang
1. Sistem menampilkan data pemesanan barang.
  2. Aktor melakukan pembatalan data pemesanan barang.
  3. Sistem menghapus data pemesanan barang beserta detail pemesanan barang dalam database.
  4. Berlanjut ke Basic Flow langkah ke-8.

Program Studi Teknik Informatika	SKPL – SISP	319/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 6. Error Flow

none

## 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data pemesanan barang telah terupdate di database.

### 4.1.15 Use case Spesification : Pengelolaan Detil Transaksi Lunas

#### 1. Brief Description

Use case ini digunakan oleh Kasir untuk mengelola detil transaksi lunas. Aktor dapat melakukan tambah detil transaksi lunas, ubah data detil transaksi lunas, hapus data detil transaksi lunas, dan tampil data detil transaksi lunas.

#### 2. Primary Actor

1. Kasir

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data detil transaksi lunas.
2. Sistem memberikan pilihan untuk melakukan tambah detil transaksi lunas, ubah data detil transaksi lunas, hapus data detil transaksi lunas, dan tampil data detil transaksi lunas.
3. Aktor memilih untuk melakukan tambah detil transaksi lunas.

Program Studi Teknik Informatika	SKPL – SISP	320/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-1 Aktor memilih untuk melakukan ubah data detail transaksi lunas

A-2 Aktor memilih untuk melakukan hapus data detail transaksi lunas

A-3 Aktor memilih untuk melakukan tampil data detail transaksi lunas

4. Aktor menginputkan data detail transaksi lunas.
5. Aktor meminta sistem untuk menyimpan data detail transaksi lunas yang telah diinputkan.
6. Sistem melakukan pengecekan data detail transaksi lunas yang telah diinputkan.

E-1 Data detail transaksi lunas yang diinputkan aktor salah

7. Sistem menyimpan data detail transaksi lunas ke database.

8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data detail transaksi lunas

1. Sistem menampilkan data detail transaksi lunas.
2. Aktor memilih detail transaksi lunas yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data detail transaksi lunas yang telah diubah.
4. Sistem melakukan pengecekan terhadap data detail transaksi lunas yang telah diubah.

Program Studi Teknik Informatika	SKPL – SISP	321/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-2 Data detail transaksi lunas yang telah diubah aktor salah

5. Sistem menyimpan data detail transaksi lunas yang telah diubah ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data detail transaksi lunas

1. Sistem menampilkan data detail transaksi lunas yang ingin dihapus.
2. Aktor menghapus data detail transaksi lunas.
3. Sistem menyimpan data detail transaksi lunas ke database.
4. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan tampil data detail transaksi lunas

1. Sistem menampilkan data detail transaksi lunas.
2. Berlanjut ke Basic Flow langkah ke-8.

## **6. Error Flow**

E-1 Data detail transaksi lunas yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data detail transaksi lunas yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

## **7. PreConditions**

Program Studi Teknik Informatika	SKPL – SISP	322/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data detil transaksi lunas telah terupdate di database.

#### **4.1.16 Use case Spesification : Pengelolaan Detil Transaksi Bon**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir untuk mengelola detil transaksi bon. Aktor dapat melakukan tambah detil transaksi bon, ubah data detil transaksi bon, hapus data detil transaksi bon, dan tampil data detil transaksi bon.

##### **2. Primary Actor**

1. Kasir

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data detil transaksi bon.
2. Sistem memberikan pilihan untuk melakukan tambah detil transaksi bon, ubah data detil transaksi bon, hapus data detil transaksi bon, dan tampil data detil transaksi bon.
3. Aktor memilih untuk melakukan tambah detil transaksi bon.  
A-1 Aktor memilih untuk melakukan ubah data detil transaksi bon  
  
A-2 Aktor memilih untuk melakukan hapus data detil transaksi bon

Program Studi Teknik Informatika	SKPL – SISP	323/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-3 Aktor memilih untuk melakukan tampil data detail transaksi bon

4. Aktor menginputkan data detail transaksi bon.
5. Aktor meminta sistem untuk menyimpan data detail transaksi bon yang telah diinputkan.
6. Sistem melakukan pengecekan data detail transaksi bon yang telah diinputkan.  
E-1 Data detail transaksi bon yang diinputkan aktor salah

7. Sistem menyimpan data detail transaksi bon ke database.

8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data detail transaksi bon

1. Sistem menampilkan data detail transaksi bon.
2. Aktor memilih detail transaksi bon yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data detail transaksi bon yang telah diubah.
4. Sistem melakukan pengecekan terhadap data detail transaksi bon yang telah diubah.

E-2 Data detail transaksi bon yang telah diubah aktor salah

5. Sistem menyimpan data detail transaksi bon yang telah diubah ke database.
6. Berlanjut ke Basic Flow langkah ke-8.

Program Studi Teknik Informatika	SKPL – SISP	324/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-2 Aktor memilih untuk melakukan hapus data detail transaksi bon

1. Sistem menampilkan data detail transaksi bon yang ingin dihapus.
2. Aktor menghapus data detail transaksi bon.
3. Sistem menyimpan data detail transaksi bon ke database.
4. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan tampil data detail transaksi bon

1. Sistem menampilkan data detail transaksi bon.
2. Berlanjut ke Basic Flow langkah ke-8.

#### **6. Error Flow**

E-1 Data detail transaksi bon yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data detail transaksi bon yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data detail transaksi bon telah terupdate di database.

Program Studi Teknik Informatika	SKPL – SISP	325/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### **4.1.17 Use case Spesification : Pengelolaan Detil Pemesanan Barang**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir untuk mengelola detil pemesanan barang. Aktor dapat melakukan tambah detil pemesanan barang, ubah data detil pemesanan barang, hapus data detil pemesanan barang, dan tampil data detil pemesanan barang.

##### **2. Primary Actor**

1. Kasir

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data detil pemesanan barang.
2. Sistem memberikan pilihan untuk melakukan tambah detil pemesanan barang, ubah data detil pemesanan barang, hapus data detil pemesanan barang, dan tampil data detil pemesanan barang.
3. Aktor memilih untuk melakukan tambah detil pemesanan barang.  
A-1 Aktor memilih untuk melakukan ubah data detil pemesanan barang  
A-2 Aktor memilih untuk melakukan hapus data detil pemesanan barang  
A-3 Aktor memilih untuk melakukan tampil data detil pemesanan barang
4. Aktor menginputkan data detil pemesanan barang.
5. Aktor meminta sistem untuk menyimpan data detil pemesanan barang yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – SISP	326/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem melakukan pengecekan data detail pemesanan barang yang telah diinputkan.

E-1 Data detail pemesanan barang yang diinputkan aktor salah

7. Sistem menyimpan data detail pemesanan barang ke database.

8. Use case selesai.

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data detail pemesanan barang

1. Sistem menampilkan data detail pemesanan barang.

2. Aktor memilih detail pemesanan barang yang akan diubah.

3. Aktor mengubah data dan meminta sistem untuk menyimpan data detail pemesanan barang yang telah diubah.

4. Sistem melakukan pengecekan terhadap data detail pemesanan barang yang telah diubah.

E-2 Data detail pemesanan barang yang telah diubah aktor salah

5. Sistem menyimpan data detail pemesanan barang yang telah diubah ke database.

6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data detail pemesanan barang

1. Sistem menampilkan data detail pemesanan barang yang ingin dihapus.

2. Aktor menghapus data detail pemesanan barang.

3. Sistem menyimpan data detail pemesanan barang ke database.

4. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan tampil data detail pemesanan barang

1. Sistem menampilkan data detail pemesanan barang.

2. Berlanjut ke Basic Flow langkah ke-8.

## 6. Error Flow

E-1 Data detail pemesanan barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data detail pemesanan barang yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.

2. Kembali ke Alternative Flow A-1 langkah ke-3.

## 7. PreConditions

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data detail pemesanan barang telah terupdate di database.

### 4.1.18 Use case Spesification : Retur Transaksi Lunas

#### 1. Brief Description

Use case ini digunakan oleh Kasir untuk mengelola retur transaksi lunas.

#### 2. Primary Actor

Program Studi Teknik Informatika	SKPL – SISP	328/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Kasir

### 3. Supporting Actor

none

### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan retur transaksi lunas.
2. Sistem menampilkan daftar transaksi lunas.
3. Aktor memilih data transaksi lunas.
4. Sistem menampilkan detail barang dari transaksi lunas.
5. Aktor menginputkan nama barang dan kuantitas barang yang diretur.
6. Sistem melakukan pengecekan data retur barang yang telah diinputkan.  
E-1 Data retur barang yang diinputkan aktor salah
7. Sistem menyimpan data retur transaksi lunas ke database.
8. Use case selesai.

### 5. Alternative Flow

none

### 6. Error Flow

E-1 Data retur barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-5.

### 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – SISP	329/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 8. PostConditions

1. Data retur transaksi lunas telah terupdate di database.

### 4.1.19 Use case Spesification : Retur Transaksi Bon

#### 1. Brief Description

Use case ini digunakan oleh Kasir untuk mengelola retur transaksi bon.

#### 2. Primary Actor

1. Kasir

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan retur transaksi bon.
2. Sistem menampilkan daftar transaksi bon.
3. Aktor memilih data transaksi bon.
4. Sistem menampilkan detail barang dari transaksi bon.
5. Aktor menginputkan nama barang dan kuantitas barang yang diretur.
6. Sistem melakukan pengecekan data retur barang yang telah diinputkan.  
E-1 Data retur barang yang diinputkan aktor salah
7. Sistem menyimpan data retur transaksi bon ke database.

Program Studi Teknik Informatika	SKPL – SISP	330/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. Use case selesai.

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Data retur barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-5.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data retur transaksi lunas telah terupdate di database.

### **4.1.20 Use case Spesification : Pengelolaan Penerimaan Barang**

#### **1. Brief Description**

Use case ini digunakan oleh Gudang untuk mengelola penerimaan barang.

#### **2. Primary Actor**

1. Gudang

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola penerimaan barang.
2. Sistem menampilkan daftar pemesanan barang.

Program Studi Teknik Informatika	SKPL – SISP	331/ 523
----------------------------------	-------------	----------

3. Aktor memilih pemesanan barang berdasarkan kode pesan yang diterima.
4. Sistem menampilkan detail pemesanan barang.
5. Aktor menginputkan data penerimaan barang.
6. Sistem melakukan pengecekan data penerimaan barang yang telah diinputkan.  
E-1 Data penerimaan barang yang diinputkan aktor salah
7. Sistem menyimpan data penerimaan barang ke database dan melakukan ubah status pemesanan barang secara otomatis.
8. Use case selesai.

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Data penerimaan barang yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-5.

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data penerimaan barang telah terupdate di database.

### **4.1.21 Use case Spesification : Reporting**

#### **1. Brief Description**

Use case ini digunakan oleh Manager untuk mengelola reporting. Aktor dapat melihat laporan penjualan umum,

Program Studi Teknik Informatika	SKPL – SISP	332/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

laporan penjualan detil, laporan bon, laporan daftar barang habis, dan grafik laporan penjualan.

## 2. Primary Actor

1. Manager

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola reporting.
2. Sistem memberikan pilihan untuk melihat laporan penjualan umum, laporan penjualan detil, laporan bon, laporan daftar barang habis, dan grafik laporan penjualan.
3. Aktor memilih untuk melihat laporan penjualan umum.  
A-1 Aktor memilih untuk melihat laporan penjualan detil  
A-2 Aktor memilih untuk melihat laporan bon  
A-3 Aktor memilih untuk melihat laporan daftar barang habis  
A-4 Aktor memilih untuk melihat grafik laporan penjualan
4. Aktor menginputkan tanggal laporan penjualan umum yang akan ditampilkan.
5. Sistem menampilkan data laporan penjualan umum.
6. Use case selesai.

## 5. Alternative Flow

Program Studi Teknik Informatika	SKPL – SISP	333/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-1 Aktor memilih untuk melihat laporan penjualan detil

1. Aktor menginputkan tanggal laporan penjualan detil yang akan ditampilkan.
2. Sistem menampilkan data laporan penjualan detil.
3. Berlanjut ke Basic Flow langkah ke-6.

A-2 Aktor memilih untuk melihat laporan bon

1. Aktor menginputkan tanggal laporan bon yang akan ditampilkan.
2. Sistem menampilkan data laporan bon.
3. Berlanjut ke Basic Flow langkah ke-6.

A-3 Aktor memilih untuk melihat laporan daftar barang habis

1. Sistem menampilkan data laporan daftar barang habis.
2. Berlanjut ke Basic Flow langkah ke-6.

A-4 Aktor memilih untuk melihat grafik laporan penjualan

1. Aktor menginputkan tanggal grafik penjualan yang akan ditampilkan.
2. Sistem menampilkan grafik penjualan transaksi lunas dan transaksi bon.
3. Berlanjut ke Basic Flow langkah ke-6.

## **6. Error Flow**

none

## **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## **8. PostConditions**

Program Studi Teknik Informatika	SKPL – SISP	334/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Aktor telah melihat laporan penjualan umum, laporan penjualan detil, laporan bon, laporan daftar barang habis, dan grafik laporan penjualan.

#### **4.1.22 Use case Spesification : Tampil Pegawai**

##### **1. Brief Description**

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk tampil data pegawai.

##### **2. Primary Actor**

1. Semua User

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan data pegawai.
2. Sistem menampilkan data pegawai.
3. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

none

##### **8. PostConditions**

1. Data pegawai telah tertampil di halaman data pegawai.

Program Studi Teknik Informatika	SKPL – SISP	335/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### **4.1.23 Use case Spesification : Tampil Cabang**

##### **1. Brief Description**

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk tampil data cabang.

##### **2. Primary Actor**

1. Semua User

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan data cabang.
2. Sistem menampilkan data cabang.
3. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

none

##### **8. PostConditions**

1. Data cabang telah tertampil di halaman data cabang.

#### **4.1.24 Use case Spesification : Tampil Kategori Barang**

##### **1. Brief Description**

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk tampil data kategori barang.

Program Studi Teknik Informatika	SKPL – SISP	336/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## **2. Primary Actor**

1. Semua User

## **3. Supporting Actor**

none

## **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan data kategori barang.
2. Sistem menampilkan data pegawai.
3. Use case selesai.

## **5. Alternative Flow**

none

## **6. Error Flow**

none

## **7. PreConditions**

none

## **8. PostConditions**

1. Data kategori telah tertampil di tabulasi list barang.

### **4.1.25 Use case Spesification : Tampil Barang**

#### **1. Brief Description**

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk menampilkan data barang.

#### **2. Primary Actor**

1. Semua User

#### **3. Supporting Actor**

none

Program Studi Teknik Informatika	SKPL – SISP	337/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk tampil data barang.
2. Sistem menampilkan data barang.
3. Use case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

none

#### 8. PostConditions

1. Data barang telah tertampil di halaman data barang.

#### 4.1.26 Use case Spesification : Tampil Barang By Kategori

##### 1. Brief Description

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk tampil data barang berdasarkan kategori.

##### 2. Primary Actor

1. Semua User

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan data barang berdasarkan kategori.

Program Studi Teknik Informatika	SKPL – SISP	338/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Aktor memilih salah satu kategori barang.
3. Sistem menampilkan data barang.
4. Use case selesai.

**5. Alternative Flow**

none

**6. Error Flow**

none

**7. PreConditions**

none

**8. PostConditions**

1. Data barang berdasarkan kategori yang dipilih telah tertampil di halaman data barang.

**4.1.27 Use case Spesification : Cari Barang**

**1. Brief Description**

Use case ini digunakan oleh Administrator, Kasir, Gudang, dan Manager untuk cari data barang.

**2. Primary Actor**

1. Semua User

**3. Supporting Actor**

none

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk cari data barang.
2. Aktor menginputkan nama barang yang akan dicari.
3. Sistem menampilkan data barang.
4. Use case selesai.

**5. Alternative Flow**

Program Studi Teknik Informatika	SKPL – SISP	339/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

none

#### **6. Error Flow**

none

#### **7. PreConditions**

none

#### **8. PostConditions**

1. Data barang telah tertampil di halaman data barang.

#### **4.1.28 Use case Spesification : Tampil Transaksi Lunas**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir dan Manager untuk tampil data transaksi lunas.

##### **2. Primary Actor**

1. Kasir
2. Manager

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk tampil data transaksi lunas.
2. Sistem menampilkan data transaksi lunas.
3. Use case selesai.

##### **5. Alternative Flow**

none

#### **6. Error Flow**

Program Studi Teknik Informatika	SKPL – SISP	340/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

none

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data transaksi lunas telah tertampil di halaman data transaksi lunas.

#### **4.1.29 Use case Spesification : Tampil Info Transaksi Lunas**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir dan Manager untuk tampil info detil dari data transaksi lunas.

##### **2. Primary Actor**

1. Kasir
2. Manager

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk tampil info detil dari data transaksi lunas.
2. Aktor memilih data transaksi lunas.
3. Sistem menampilkan info detil dari data transaksi lunas.
4. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

Program Studi Teknik Informatika	SKPL – SISP	341/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

none

#### **7. PreConditions**

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Info detil dari data transaksi lunas telah tertampil di halaman info transaksi lunas.

### **4.1.30 Use case Spesification : Cari Transaksi Lunas**

#### **1. Brief Description**

Use case ini digunakan oleh Kasir dan Manager untuk cari data transaksi lunas.

#### **2. Primary Actor**

1. Kasir
2. Manager

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk cari data transaksi lunas.
2. Aktor menginputkan tanggal transaksi lunas yang akan dicari.
3. Sistem menampilkan data transaksi lunas.
4. Use case selesai.

#### **5. Alternative Flow**

none

#### **6. Error Flow**

none

Program Studi Teknik Informatika	SKPL – SISP	342/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data transaksi lunas telah tertampil di halaman data transaksi lunas

### 4.1.31 Use case Spesification : Tampil Transaksi Bon

#### 1. Brief Description

Use case ini digunakan oleh Kasir dan Manager untuk tampil data transaksi bon.

#### 2. Primary Actor

1. Kasir
2. Manager

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk tampil data transaksi bon.
2. Sistem menampilkan data transaksi bon.
3. Use case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

## 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – SISP	343/ 523
----------------------------------	-------------	----------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

## 8. PostConditions

1. Data transaksi bon telah tertampil di halaman data transaksi bon.

### 4.1.32 Use case Spesification : Tampil Info Transaksi Bon

#### 1. Brief Description

Use case ini digunakan oleh Kasir dan Manager untuk tampil info detil dari data transaksi bon.

#### 2. Primary Actor

1. Kasir
2. Manager

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk tampil info detil dari data transaksi bon.
2. Aktor memilih data transaksi bon.
3. Sistem menampilkan info detil dari data transaksi bon.
4. Use case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – SISP	344/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 8. PostConditions

1. Info detil dari data transaksi bon telah tertampil di halaman info transaksi bon.

### 4.1.33 Use case Spesification : Cari Transaksi Bon

#### 1. Brief Description

Use case ini digunakan oleh Kasir dan Manager untuk cari data transaksi bon.

#### 2. Primary Actor

1. Kasir
2. Manager

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk cari data transaksi bon.
2. Aktor menginputkan tanggal transaksi bon yang akan dicari.
3. Sistem menampilkan data transaksi bon.
4. Use case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

Program Studi Teknik Informatika	SKPL – SISP	345/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data transaksi bon telah tertampil di halaman data transaksi bon.

#### **4.1.34 Use case Spesification : Tampil Pemesanan Barang**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir, Gudang, dan Manager untuk tampil data pemesanan barang.

##### **2. Primary Actor**

1. Kasir
2. Gudang
3. Manager

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk tampil data pemesanan barang.
2. Sistem menampilkan data pemesanan barang.
3. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

1. Use case Login telah dilakukan.

Program Studi Teknik Informatika	SKPL – SISP	346/ 523
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Data pemesanan barang telah tertampil di halaman data pemesanan barang.

#### **4.1.35 Use case Spesification : Tampil Info Pemesanan Barang**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir, Gudang, dan Manager untuk tampil info detil dari data pemesanan barang.

##### **2. Primary Actor**

1. Kasir
2. Gudang
3. Manager

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk tampil info detil dari data pemesanan barang.
2. Aktor memilih data pemesanan barang
3. Sistem menampilkan info detil dari data pemesanan barang.
4. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

##### **7. PreConditions**

Program Studi Teknik Informatika	SKPL – SISP	347/ 523
----------------------------------	-------------	----------

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### **8. PostConditions**

1. Info detil dari data pemesanan barang telah tertampil di halaman info pemesanan barang.

#### **4.1.36 Use case Spesification : Cari Pemesanan Barang**

##### **1. Brief Description**

Use case ini digunakan oleh Kasir, Gudang, dan Manager untuk cari data pemesanan barang.

##### **2. Primary Actor**

1. Kasir
2. Gudang
3. Manager

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk cari data pemesanan barang.
2. Aktor menginputkan tanggal pemesanan barang yang akan dicari
3. Sistem menampilkan data pemesanan barang.
4. Use case selesai.

##### **5. Alternative Flow**

none

##### **6. Error Flow**

none

Program Studi Teknik Informatika	SKPL – SISP	348/ 523
----------------------------------	-------------	----------

## 7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data pemesanan barang telah tertampil di halaman data pemesanan barang.

### 4.1.37 Use case Spesification : Tampil Penerimaan Barang

#### 1. Brief Description

Use case ini digunakan oleh Manager untuk tampil data penerimaan barang.

#### 2. Primary Actor

1. Manager

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk tampil data penerimaan barang.
2. Sistem menampilkan data penerimaan barang.
3. Use case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

## 7. PreConditions

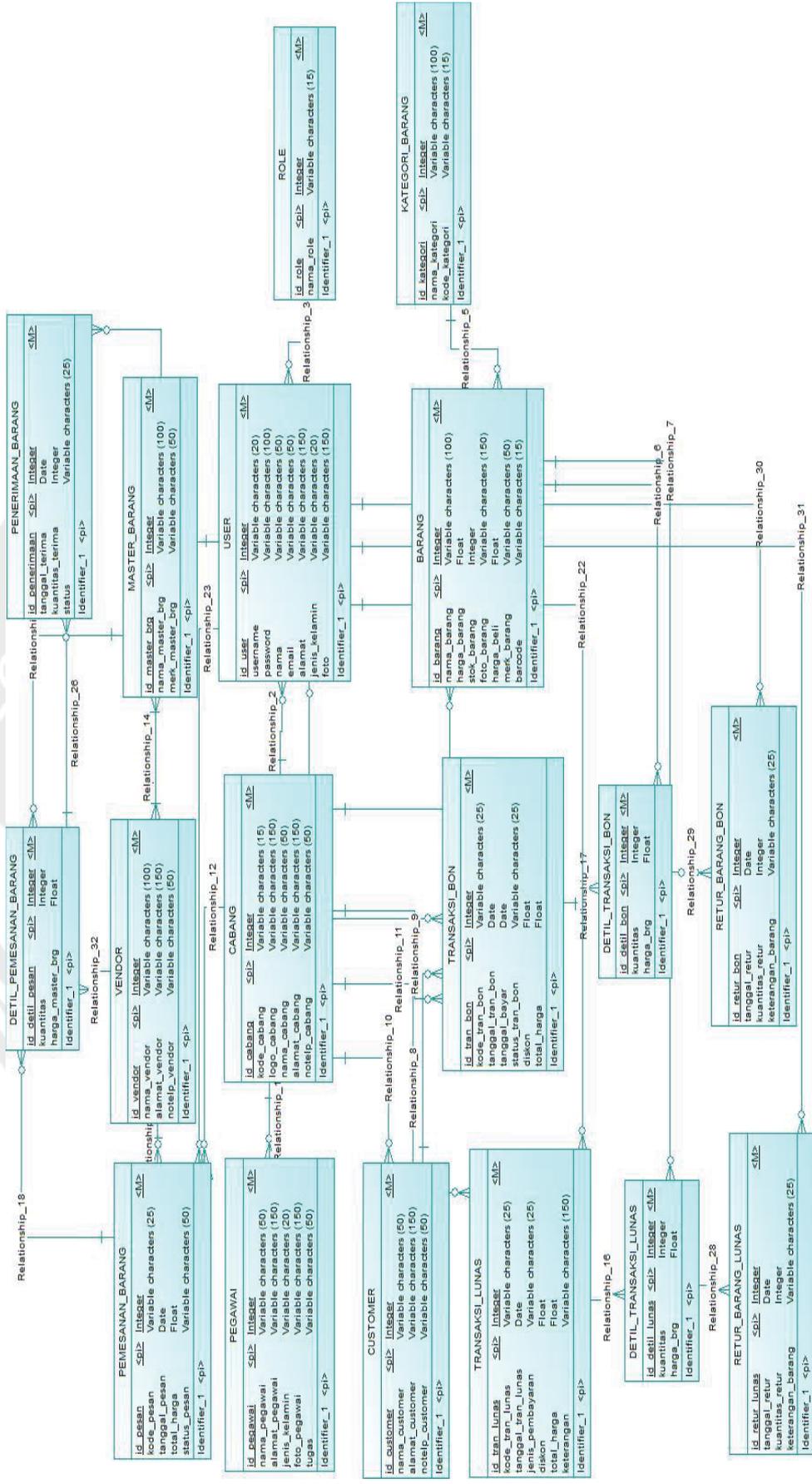
1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data penerimaan barang telah tertampil di halaman data penerimaan barang.



# Entity Relationship Diagram (ERD)



# DPPL

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Pembangunan Sistem Informasi Spare Parts  
Terintegrasi Berbasis Web

(SISP)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Irene Deandra Indarto / 11. 07. 06483

Program Studi Teknik Informatika - Fakultas Teknologi  
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		DPPL- SISP		1/153
	Fakultas Teknologi Industri	Revisi		

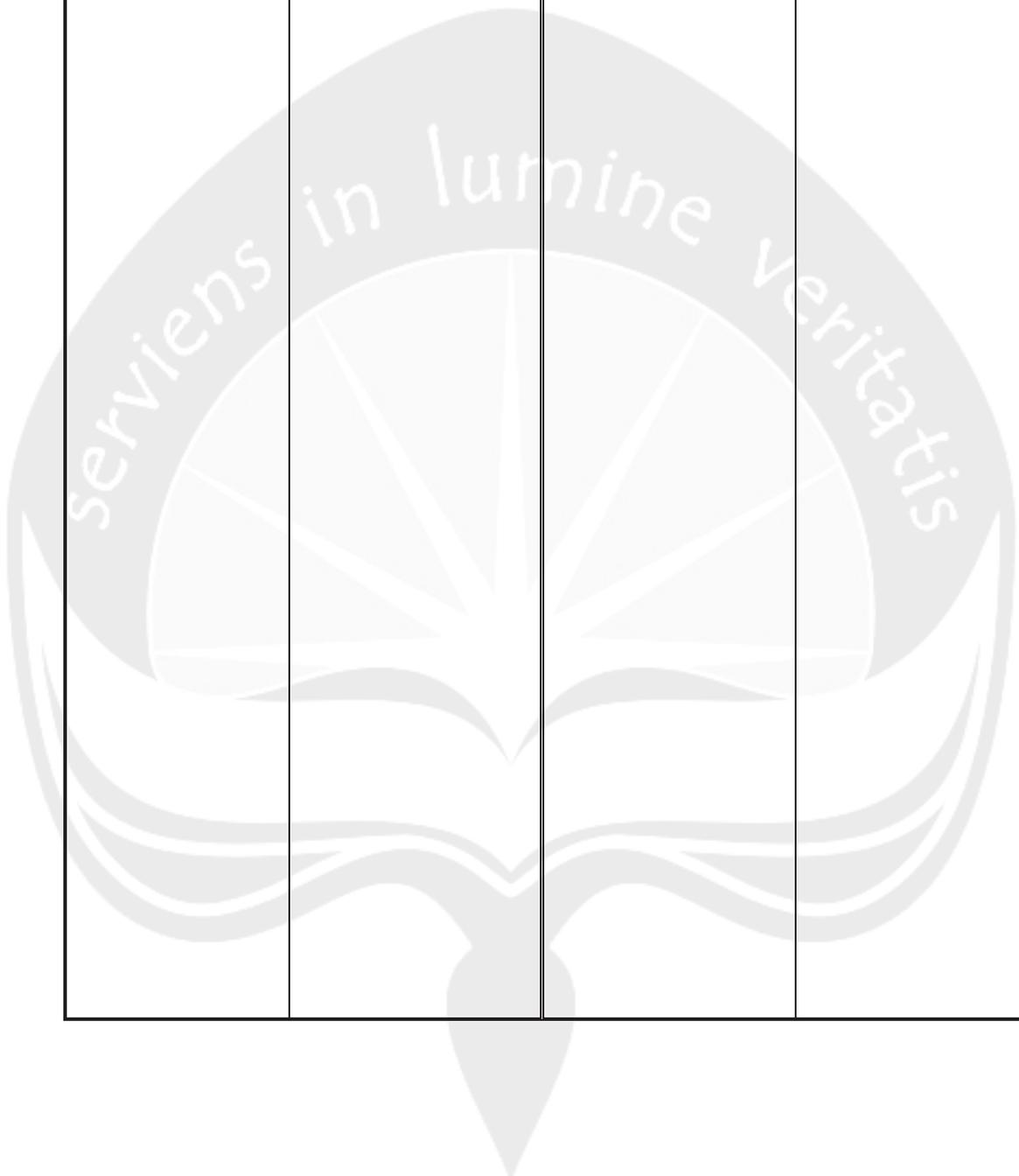
### DAFTAR PERUBAHAN

REVISI	DESKRIPSI
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F
DITULIS OLEH	IDI						
DIPERIKSA OLEH							
DISETUJUI OLEH							

### DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi



## Daftar Isi

1	Pendahuluan.....	12
1.1	Tujuan .....	12
1.2	Lingkup Masalah .....	12
1.3	Definisi dan Akronim .....	12
1.4	Referensi .....	14
2	Analisis Model .....	15
2.1	Perancangan Arsitektur .....	15
2.2	Perancangan Rinci .....	17
	Gambar 1.....	2
2.2.1	Sequence Diagram.....	17
2.2.1.1	Login.....	17
2.2.1.2	Ubah Password.....	17
2.2.1.3	Pendaftaran User Baru.....	18
2.2.1.4	Pengelolaan User.....	18
IV.3.6.20	2.2.1.4.1.....	Ubah Data User 51
IV.3.6.21	2.2.1.4.2.....	Tampil Data User 19
2.2.1.5	Pengelolaan Vendor.....	19
IV.3.6.22	2.2.1.5.1.....	Tambah Vendor 52
IV.3.6.23	2.2.1.5.2.....	Ubah Vendor 53
IV.3.6.24	2.2.1.5.3.....	Tampil Vendor 20
2.2.1.6	Pengelolaan Customer.....	20
IV.3.6.25	2.2.1.6.1.....	Tambah Customer 20
IV.3.6.26	2.2.1.6.2.....	Ubah Customer 21
IV.3.6.27	2.2.1.6.3.....	Tampil Customer 21
2.2.1.7	Pengelolaan Pegawai.....	22
IV.3.6.28	2.2.1.7.1.....	Tambah Pegawai 55
IV.3.6.29	2.2.1.7.2.....	Ubah Pegawai 56
IV.3.6.30	2.2.1.7.3.....	Hapus Pegawai 56
2.2.1.8	Pengelolaan Cabang.....	23
IV.3.6.31	2.2.1.8.1.....	Tambah Cabang 57
IV.3.6.32	2.2.1.8.2.....	Ubah Cabang 24
IV.3.6.33	2.2.1.8.3.....	Hapus Cabang 24
2.2.1.9	Pengelolaan Kategori Barang.....	25
IV.3.6.34	2.2.1.9.1.....	Tambah Kategori Barang 25
IV.3.6.35	2.2.1.9.2.....	Ubah Kategori Barang 26
2.2.1.10	Pengelolaan Barang.....	26
IV.3.6.36	2.2.1.10.1.....	Tambah Barang 59

IV.3.6.37.2.2.1.10.2	Ubah Barang	60
IV.3.6.38.2.2.1.10.3	Hapus Barang	60
2.2.1.11	Pengelolaan Master Barang	28
IV.3.6.39.2.2.1.11.1	Tambah Master Barang	61
IV.3.6.40.2.2.1.11.2	Ubah Master Barang	61
IV.3.6.41.2.2.1.11.3	Tampil Master Barang	29
IV.3.6.42.2.2.1.11.4	Cari Master Barang	29
2.2.1.12	Pengelolaan Transaksi Lunas	30
IV.3.6.43.2.2.1.12.1	Generate Kode Transaksi Lunas	62
IV.3.6.44.2.2.1.12.2	Tambah Transaksi Lunas	63
IV.3.6.45.2.2.1.12.3	Konfirmasi Transaksi Lunas	63
IV.3.6.46.2.2.1.12.4	Batal Transaksi Lunas	64
IV.3.6.47.2.2.1.12.5	Cetak Nota Transaksi Lunas	64
2.2.1.13	Pengelolaan Transaksi Bon	32
IV.3.6.48.2.2.1.13.1	Generate Kode Transaksi Bon	65
IV.3.6.49.2.2.1.13.2	Tambah Transaksi Bon	65
IV.3.6.50.2.2.1.13.3	Konfirmasi Transaksi Bon	33
IV.3.6.51.2.2.1.13.4	Batal Transaksi Bon	34
IV.3.6.52.2.2.1.13.5	Cetak Nota Transaksi Bon	34
IV.3.6.53.2.2.1.13.6	Pelunasan Transaksi Bon	67
2.2.1.14	Pengelolaan Pemesanan Barang	35
IV.3.6.54.2.2.1.14.1	Generate Kode Pemesanan Barang	68
IV.3.6.55.2.2.1.14.2	Tambah Pemesanan Barang	68
IV.3.6.56.2.2.1.14.3	Batal Pemesanan Barang	36
2.2.1.15	Pengelolaan Detil Transaksi Lunas	37
IV.3.6.57.2.2.1.15.1	Tambah Detil Transaksi Lunas	69
IV.3.6.58.2.2.1.15.2	Ubah Detil Transaksi Lunas	70
IV.3.6.59.2.2.1.15.3	Hapus Detil Transaksi Lunas	38
IV.3.6.60.2.2.1.15.4	Tampil Detil Transaksi Lunas	71
2.2.1.16	Pengelolaan Detil Transaksi Bon	39
IV.3.6.61.2.2.1.16.1	Tambah Detil Transaksi Bon	71

IV.3.6.62.2.2.1.16.2.....	Ubah Detil Transaksi Bon	72
IV.3.6.63.2.2.1.16.3.....	Hapus Detil Transaksi Bon	72
IV.3.6.64.2.2.1.16.4.....	Tampil Detil Transaksi Bon	40
2.2.1.17	Pengelolaan Detil Pemesanan Barang.....	40
IV.3.6.65.2.2.1.17.1.....	Tambah Detil Pemesanan Barang	73
IV.3.6.66.2.2.1.17.2.....	Ubah Detil Pemesanan Barang	41
IV.3.6.67.2.2.1.17.3.....	Hapus Detil Pemesanan Barang	41
IV.3.6.68.2.2.1.17.4.....	Tampil Detil Pemesanan Barang	42
2.2.1.18	Retur Transaksi Lunas.....	43
2.2.1.19	Retur Transaksi Bon.....	43
2.2.1.20	Pengelolaan Penerimaan Barang.....	44
IV.3.6.69.2.2.1.20.1.....	Tambah Penerimaan Barang	76
IV.3.6.70.2.2.1.20.2.....	Ubah Status Pemesanan Barang	44
2.2.1.21	Reporting.....	44
IV.3.6.71.2.2.1.21.1.....	Laporan Penjualan Umum	77
IV.3.6.72.2.2.1.21.2.....	Laporan Penjualan Detil	78
IV.3.6.73.2.2.1.21.3.....	Laporan Bon	78
IV.3.6.74.2.2.1.21.4.....	Laporan Daftar Barang Habis	46
IV.3.6.75.2.2.1.21.5.....	Grafik Laporan Penjualan	47
2.2.1.22	Tampil Pegawai.....	80
2.2.1.23	Tampil Cabang.....	80
2.2.1.24	Tampil Kategori Barang.....	81
2.2.1.25	Tampil Barang.....	81
2.2.1.26	Tampil Barang By Kategori.....	49
2.2.1.27	Cari Barang.....	82
2.2.1.28	Tampil Transaksi Lunas.....	83
2.2.1.29	Tampil Info Transaksi Lunas.....	83
2.2.1.30	Cari Transaksi Lunas.....	84
2.2.1.31	Tampil Transaksi Bon.....	52
2.2.1.32	Tampil Info Transaksi Bon.....	85
2.2.1.33	Cari Transaksi Bon.....	85
2.2.1.34	Tampil Pemesanan Barang.....	86
2.2.1.35	Tampil Info Pemesanan Barang.....	54
2.2.1.36	Cari Pemesanan Barang.....	87
2.2.1.37	Tampil Penerimaan Barang.....	87
Gambar 2.....		2
.2.2	Class Diagram.....	56
Gambar 3.....		2
.2.3	Class Diagram Specific Descriptions.....	59
2.2.3.1	Specific Design Class loginUI.....	59
2.2.3.2	Specific Design Class update_passwordUI.....	59
2.2.3.3	Specific Design insert_userUI.....	59

2.2.3.4	Specific Design userUI.....	60
2.2.3.5	Specific Design Class cabangUI.....	60
2.2.3.6	Specific Design pegawaiUI.....	61
2.2.3.7	Specific Design customerUI.....	61
2.2.3.8	Specific Design kategori_barangUI.....	62
2.2.3.9	Specific Design Class vendorUI.....	62
2.2.3.10	Specific Design Class master_barangUI.....	63
2.2.3.11	Specific Design Class pemesanan_barangUI.....	63
2.2.3.12	Specific Design Class detail_pemesanan_barangUI..	64
2.2.3.13	Specific Design Class penerimaan_barangUI.....	65
2.2.3.14	Specific Design Class barangUI.....	65
2.2.3.15	Specific Design Class tran_lunasUI.....	66
2.2.3.16	Specific Design Class detail_tran_lunasUI.....	67
2.2.3.17	Specific Design Class retur_lunasUI.....	68
2.2.3.18	Specific Design Class tran_bonUI.....	68
2.2.3.19	Specific Design Class detail_tran_bonUI.....	69
2.2.3.20	Specific Design retur_bonUI.....	70
2.2.3.21	Specific Design Class grafikUI.....	70
2.2.3.22	Specific Design reportUI.....	71
2.2.3.23	Specific Design Class login_ctrl.....	71
2.2.3.24	Specific Design Class user_ctrl.....	72
2.2.3.25	Specific Design cabang_ctrl.....	72
2.2.3.26	Specific Design Class pegawai_ctrl.....	73
2.2.3.27	Specific Design Class customer_ctrl.....	73
2.2.3.28	Specific Design Class kategori_barang_ctrl.....	74
2.2.3.29	Specific Design Class vendor_ctrl.....	74
2.2.3.30	Specific Design Class master_barang_ctrl.....	74
2.2.3.31	Specific Design Class pemesanan_barang_ctrl....	75
2.2.3.32	Specific Design Class barang_ctrl.....	77
2.2.3.33	Specific Design Class tran_lunas_ctrl.....	77
2.2.3.34	Specific Design Class tran_bon_ctrl.....	79
2.2.3.35	Specific Design Class reporting_ctrl.....	80
2.2.3.36	Specific Design Class user_model.....	81
2.2.3.37	Specific Design cabang_model.....	82
2.2.3.38	Specific Design pegawai_model.....	83
2.2.3.39	Specific Design Class customer_model.....	84
2.2.3.40	Specific Design Class kategori_model.....	85
2.2.3.41	Specific Design Class vendor_model.....	85
2.2.3.42	Specific Design Class master_barang_model.....	86
2.2.3.43	Specific Design Class pemesanan_barang_model...	87
2.2.3.44	Specific Design Class penerimaan_barang_model..	89
2.2.3.45	Specific Design Class barang_model.....	89
2.2.3.46	Specific Design Class tran_lunas_model.....	91
2.2.3.47	Specific Design Class retur_lunas_model.....	93
2.2.3.48	Specific Design Class tran_bon_model.....	93
2.2.3.49	Specific Design Class retur_bon_model.....	95
2.2.3.50	Specific Design Class grafik_model.....	96
2.2.3.51	Specific Design Class reporting_model.....	96
3	Perancangan Data .....	97
3.1	Dekomposisi Data .....	97
	Gambar 4. ....	3
.1.1	Deskripsi Entitas Data Role.....	97
	Gambar 5. ....	3
.1.2	Deskripsi Entitas Data User.....	97
	Gambar 6. ....	3
.1.3	Deskripsi Entitas Data Pegawai.....	97

Gambar 7.....	3
.1.4 Deskripsi Entitas Data Customer.....	98
Gambar 8.....	3
.1.5 Deskripsi Entitas Data Vendor.....	98
Gambar 9.....	3
.1.6 Deskripsi Entitas Data Cabang.....	98
Gambar 10.....	3
.1.7 Deskripsi Entitas Data Master Barang.....	99
Gambar 11.....	3
.1.8 Deskripsi Entitas Data Kategori Barang.....	99
Gambar 12.....	3
.1.9 Deskripsi Entitas Data Barang.....	99
Gambar 13.....	3
.1.10 Deskripsi Entitas Data Barang_Vendor.....	99
Gambar 14.....	3
.1.11 Deskripsi Entitas Data Transaksi_Lunas.....	100
Gambar 15.....	3
.1.12 Deskripsi Entitas Data Transaksi_Bon.....	101
Gambar 16.....	3
.1.13 Deskripsi Entitas Data Pemesanan_Barang.....	101
Gambar 17.....	3
.1.14 Deskripsi Entitas Data Detil_Transaksi_Lunas.....	102
Gambar 18.....	3
.1.15 Deskripsi Entitas Data Detil_Transaksi_Bon.....	102
Gambar 19.....	3
.1.16 Deskripsi Entitas Data Detil_Pemesanan_Barang.....	103
Gambar 20.....	3
.1.17 Deskripsi Entitas Data Retur_Barang_Lunas.....	103
Gambar 21.....	3
.1.18 Deskripsi Entitas Data Retur_Barang_Bon.....	104
Gambar 22.....	3
.1.19 Deskripsi Entitas Data Penerimaan_Barang.....	104
3.2 Physical Data Model.....	105
4 Perancangan Antarmuka.....	106
4.1 Antarmuka Home.....	106
4.2 Antarmuka Login.....	107
4.3 Antarmuka Ubah Password.....	108
4.4 Antarmuka Pendaftaran User Baru.....	109
4.5 Antarmuka Pengelolaan User.....	110
4.6 Antarmuka Pengelolaan Vendor.....	112
4.7 Antarmuka Pengelolaan Customer.....	114
4.8 Antarmuka Pengelolaan Pegawai.....	116
4.9 Antarmuka Pengelolaan Cabang.....	118
4.10 Antarmuka Pengelolaan Kategori Barang.....	120
4.11 Antarmuka Pengelolaan Barang.....	122
4.12 Antarmuka Pengelolaan Master Barang.....	124
4.13 Antarmuka Pengelolaan Transaksi Lunas.....	126
4.14 Antarmuka Pengelolaan Transaksi Bon.....	130
4.15 Antarmuka Pengelolaan Pemesanan Barang.....	134
4.16 Antarmuka Pengelolaan Detil Transaksi Lunas.....	136
4.17 Antarmuka Pengelolaan Detil Transaksi Bon.....	137
4.18 Antarmuka Pengelolaan Detil Pemesanan Barang.....	138
4.19 Antarmuka Retur Transaksi Lunas.....	139
4.20 Antarmuka Retur Transaksi Bon.....	140
4.21 Antarmuka Pengelolaan Penerimaan Barang.....	141
4.22 Antarmuka Reporting.....	142

4.23	Antarmuka Data Pegawai .....	145
4.24	Antarmuka Data Cabang .....	146
4.25	Antarmuka Data Kategori Barang .....	147
4.26	Antarmuka Data Barang .....	148
4.27	Antarmuka Data Transaksi Lunas .....	149
4.28	Antarmuka Data Transaksi Bon .....	151
4.29	Antarmuka Data Pemesanan Barang .....	153
4.30	Antarmuka Data Penerimaan Barang .....	155

## Daftar Gambar

Gambar 2.1	Perancangan Arsitektur.....	16
Gambar 2.2	Sequence Diagram: Login.....	17
Gambar 2.3	Sequence Diagram: Ubah Password.....	17
Gambar 2.4	Sequence Diagram: Pendaftaran User Baru.....	18
Gambar 2.5	Sequence Diagram: Ubah Data User.....	18
Gambar 2.6	Sequence Diagram: Tampil Data User.....	19
Gambar 2.7	Sequence Diagram: Tambah Vendor.....	19
Gambar 2.8	Sequence Diagram: Ubah Vendor.....	20
Gambar 2.9	Sequence Diagram: Tampil Vendor.....	20
Gambar 2.10	Sequence Diagram: Tambah Customer.....	21
Gambar 2.11	Sequence Diagram: Ubah Customer.....	21
Gambar 2.12	Sequence Diagram: Tampil Customer.....	22
Gambar 2.13	Sequence Diagram: Tambah Pegawai.....	22
Gambar 2.14	Sequence Diagram: Ubah Pegawai.....	23
Gambar 2.15	Sequence Diagram: Hapus Pegawai.....	23
Gambar 2.16	Sequence Diagram: Tambah Cabang.....	24
Gambar 2.17	Sequence Diagram: Ubah Cabang.....	24
Gambar 2.18	Sequence Diagram: Hapus Cabang.....	25
Gambar 2.19	Sequence Diagram: Tambah Kategori Barang.....	25
Gambar 2.20	Sequence Diagram: Ubah Kategori Barang.....	26
Gambar 2.21	Sequence Diagram: Tambah Barang.....	26
Gambar 2.22	Sequence Diagram: Ubah Barang.....	27
Gambar 2.23	Sequence Diagram: Hapus Barang.....	27
Gambar 2.24	Sequence Diagram: Tambah Master Barang.....	28
Gambar 2.25	Sequence Diagram: Ubah Master Barang.....	28
Gambar 2.26	Sequence Diagram: Tampil Master Barang.....	29
Gambar 2.27	Sequence Diagram: Cari Master Barang.....	29
Gambar 2.28	Sequence Diagram: Generate Kode Transaksi Lunas....	30
Gambar 2.29	Sequence Diagram: Tambah Transaksi Lunas.....	30
Gambar 2.30	Sequence Diagram: Konfirmasi Transaksi Lunas.....	31
Gambar 2.31	Sequence Diagram: Batal Transaksi Lunas.....	31
Gambar 2.32	Sequence Diagram: Cetak Nota Transaksi Lunas.....	32
Gambar 2.33	Sequence Diagram: Generate Kode Transaksi Bon.....	32
Gambar 2.34	Sequence Diagram: Tambah Transaksi Bon.....	33
Gambar 2.35	Sequence Diagram: Konfirmasi Transaksi Bon.....	33
Gambar 2.36	Sequence Diagram: Batal Transaksi Bon.....	34
Gambar 2.37	Sequence Diagram: Cetak Nota Transaksi Bon.....	34
Gambar 2.38	Sequence Diagram: Pelunasan Transaksi Bon.....	35
Gambar 2.39	Sequence Diagram: Generate Kode Pemesanan Barang...	35
Gambar 2.40	Sequence Diagram: Tambah Pemesanan Barang.....	36
Gambar 2.41	Sequence Diagram: Batal Pemesanan Barang.....	36

Gambar 2.42	Sequence Diagram: Tambah Detil Transaksi Lunas.....	37
Gambar 2.43	Sequence Diagram: Ubah Detil Transaksi Lunas.....	37
Gambar 2.44	Sequence Diagram: Hapus Detil Transaksi Lunas.....	38
Gambar 2.45	Sequence Diagram: Tampil Detil Transaksi Lunas.....	38
Gambar 2.46	Sequence Diagram: Tambah Detil Transaksi Bon.....	39
Gambar 2.47	Sequence Diagram: Ubah Detil Transaksi Bon.....	39
Gambar 2.48	Sequence Diagram: Hapus Detil Transaksi Bon.....	40
Gambar 2.49	Sequence Diagram: Tampil Detil Transaksi Bon.....	40
Gambar 2.50	Sequence Diagram: Tambah Detil Pemesanan Barang....	41
Gambar 2.51	Sequence Diagram: Ubah Detil Pemesanan Barang.....	41
Gambar 2.52	Sequence Diagram: Hapus Detil Pemesanan Barang.....	42
Gambar 2.53	Sequence Diagram: Tampil Detil Pemesanan Barang....	42
Gambar 2.54	Sequence Diagram: Retur Transaksi Lunas.....	43
Gambar 2.55	Sequence Diagram: Retur Transaksi Bon.....	43
Gambar 2.56	Sequence Diagram: Tambah Penerimaan Barang.....	44
Gambar 2.57	Sequence Diagram: Ubah Status Pemesanan Barang.....	44
Gambar 2.58	Sequence Diagram: Laporan Penjualan Umum.....	45
Gambar 2.59	Sequence Diagram: Laporan Penjualan Detil.....	45
Gambar 2.60	Sequence Diagram: Laporan Bon.....	46
Gambar 2.61	Sequence Diagram: Laporan Daftar Barang Habis.....	46
Gambar 2.62	Sequence Diagram: Grafik Laporan Penjualan.....	47
Gambar 2.63	Sequence Diagram: Tampil Pegawai.....	47
Gambar 2.64	Sequence Diagram: Tampil Cabang.....	48
Gambar 2.65	Sequence Diagram: Tampil Kategori Barang.....	48
Gambar 2.66	Sequence Diagram: Tampil Barang.....	49
Gambar 2.67	Sequence Diagram: Tampil Barang By Kategori.....	49
Gambar 2.68	Sequence Diagram: Cari Barang.....	50
Gambar 2.69	Sequence Diagram: Tampil Transaksi Lunas.....	50
Gambar 2.70	Sequence Diagram: Tampil Info Transaksi Lunas.....	51
Gambar 2.71	Sequence Diagram: Cari Transaksi Lunas.....	51
Gambar 2.72	Sequence Diagram: Tampil Transaksi Bon.....	52
Gambar 2.73	Sequence Diagram: Tampil Info Transaksi Bon.....	52
Gambar 2.74	Sequence Diagram: Cari Transaksi Bon.....	53
Gambar 2.75	Sequence Diagram: Tampil Pemesanan Barang.....	53
Gambar 2.76	Sequence Diagram: Tampil Info Pemesanan Barang.....	54
Gambar 2.77	Sequence Diagram: Cari Pemesanan Barang.....	54
Gambar 2.78	Sequence Diagram: Tampil Penerimaan Barang.....	55
Gambar 3.1	Physical Data Model.....	105
Gambar 4.1	Rancangan Antarmuka Home.....	106
Gambar 4.2	Rancangan Antarmuka Login.....	107
Gambar 4.3	Rancangan Antarmuka Home setelah Login.....	108
Gambar 4.4	Rancangan Antarmuka Ubah Password.....	108
Gambar 4.5	Rancangan Antarmuka Pengelolaan User.....	109
Gambar 4.6	Rancangan Antarmuka Pengelolaan User.....	110
Gambar 4.7	Rancangan Antarmuka Ubah Data User.....	111
Gambar 4.8	Rancangan Antarmuka Pengelolaan Vendor.....	112
Gambar 4.9	Rancangan Antarmuka Pengelolaan Tambah Vendor.....	113
Gambar 4.10	Rancangan Antarmuka Pengelolaan Customer.....	114
Gambar 4.11	Rancangan Antarmuka Pengelolaan Tambah Customer... ..	115
Gambar 4.12	Rancangan Antarmuka Pengelolaan Pegawai.....	116
Gambar 4.13	Rancangan Antarmuka Pengelolaan Tambah Pegawai....	117
Gambar 4.14	Rancangan Antarmuka Pengelolaan Cabang.....	118
Gambar 4.15	Rancangan Antarmuka Pengelolaan Tambah Cabang.....	119
Gambar 4.16	Rancangan Antarmuka Pengelolaan Kategori Barang... ..	120
Gambar 4.17	Rancangan Antarmuka Pengelolaan Tambah Kategori Barang .....	121

Gambar 4.18	Rancangan Antarmuka Pengelolaan Barang.....	122
Gambar 4.19	Rancangan Antarmuka Pengelolaan Tambah Barang.....	123
Gambar 4.20	Rancangan Antarmuka Pengelolaan Master Barang.....	124
Gambar 4.21	Rancangan Antarmuka Pengelolaan Tambah Master Barang .....	125
Gambar 4.22	Rancangan Antarmuka Pengelolaan Transaksi Lunas...	126
Gambar 4.23	Rancangan Antarmuka Pengelolaan Tambah Transaksi Lunas .....	127
Gambar 4.24	Rancangan Antarmuka Pengelolaan Konfirmasi dan Cetak Nota Transaksi Lunas.....	128
Gambar 4.25	Rancangan Antarmuka Nota Transaksi Lunas.....	129
Gambar 4.27	Rancangan Antarmuka Pengelolaan Tambah Transaksi Bon .....	131
Gambar 4.28	Rancangan Antarmuka Pengelolaan Konfirmasi dan Cetak Nota Transaksi Bon.....	132
Gambar 4.29	Rancangan Antarmuka Nota Transaksi Bon.....	133
Gambar 4.30	Rancangan Antarmuka Pengelolaan Pemesanan Barang..	134
Gambar 4.31	Rancangan Antarmuka Pengelolaan Tambah Pemesanan Barang.....	135
Gambar 4.32	Rancangan Antarmuka Pengelolaan Detil Transaksi Lunas .....	136
Gambar 4.33	Rancangan Antarmuka Pengelolaan Detil Transaksi Bon	137
Gambar 4.34	Rancangan Antarmuka Pengelolaan Detil Pemesanan Barang .....	138
Gambar 4.35	Rancangan Antarmuka Retur Transaksi Lunas.....	139
Gambar 4.36	Rancangan Antarmuka Retur Transaksi Bon.....	140
Gambar 4.37	Rancangan Antarmuka Pengelolaan Penerimaan Barang.	141
Gambar 4.38	Rancangan Antarmuka Reporting.....	142
Gambar 4.39	Rancangan Antarmuka Reporting Umum.....	143
Gambar 4.40	Rancangan Antarmuka Reporting Grafik Penjualan....	144
Gambar 4.41	Rancangan Antarmuka Data Pegawai.....	145
Gambar 4.42	Rancangan Antarmuka Data Cabang.....	146
Gambar 4.43	Rancangan Antarmuka Data Kategori Barang.....	147
Gambar 4.44	Rancangan Antarmuka Data Barang.....	148
Gambar 4.45	Rancangan Antarmuka Data Transaksi Lunas.....	149
Gambar 4.46	Rancangan Antarmuka Info Transaksi Lunas.....	150
Gambar 4.47	Rancangan Antarmuka Data Transaksi Bon.....	151
Gambar 4.48	Rancangan Antarmuka Info Transaksi Bon.....	152
Gambar 4.49	Rancangan Antarmuka Data Pemesanan Barang.....	153
Gambar 4.50	Rancangan Antarmuka Info Pemesanan Barang.....	154
Gambar 4.51	Rancangan Antarmuka Data Penerimaan Barang.....	155

## 1 Pendahuluan

### Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen tersebut akan digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap berikutnya.

### Lingkup Masalah

Perangkat Lunak SISP dikembangkan dengan tujuan untuk:

5. Menangani pengelolaan yang terkait dengan cabang, vendor, pegawai, master barang, dan barang di PT. XYZ.
6. Menangani pengelolaan transaksi yang ada di PT. XYZ.
7. Menangani pengelolaan pemesanan dan penerimaan barang yang ada di PT. XYZ.
8. Menangani pengelolaan reporting transaksi penjualan di PT. XYZ.

Dan aplikasi web ini dapat berjalan dengan web browser.

### Definisi dan Akronim

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD). Merupakan deskripsi dari perancangan produk/perangkat lunak yang

Program Studi Teknik Informatika	DPPL – SISP	12/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

	dikembangkan..
DPPL-SISP-XXX	Kode yang merepresentasikan kebutuhan pada SISP (Sistem Informasi Spare Parts Terintegrasi Berbasis Web) dimana XXX merupakan nomor fungsi produk.
SISP	Perangkat lunak pengelolaan web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
PT. XYZ	PT. XYZ adalah perusahaan yang bergerak di bidang penjualan spare parts serta aksesoris mobil dan memiliki beberapa cabang yang terletak di Yogyakarta.
Spare Parts	Bagian dari alat, mesin atau kendaraan yang disediakan untuk penggantian karena terjadinya keausan, kerusakan atau tidak berfungsinya komponen tersebut sesuai dengan standar kinerja yang diharapkan.
Transaksi	Persetujuan jual beli (dalam perdagangan) antara dua pihak.

Framework	Suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks.
PHP	Hypertext Preprocessor merupakan bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Code Igniter	Aplikasi open source yang berupa framework dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP.

### Referensi

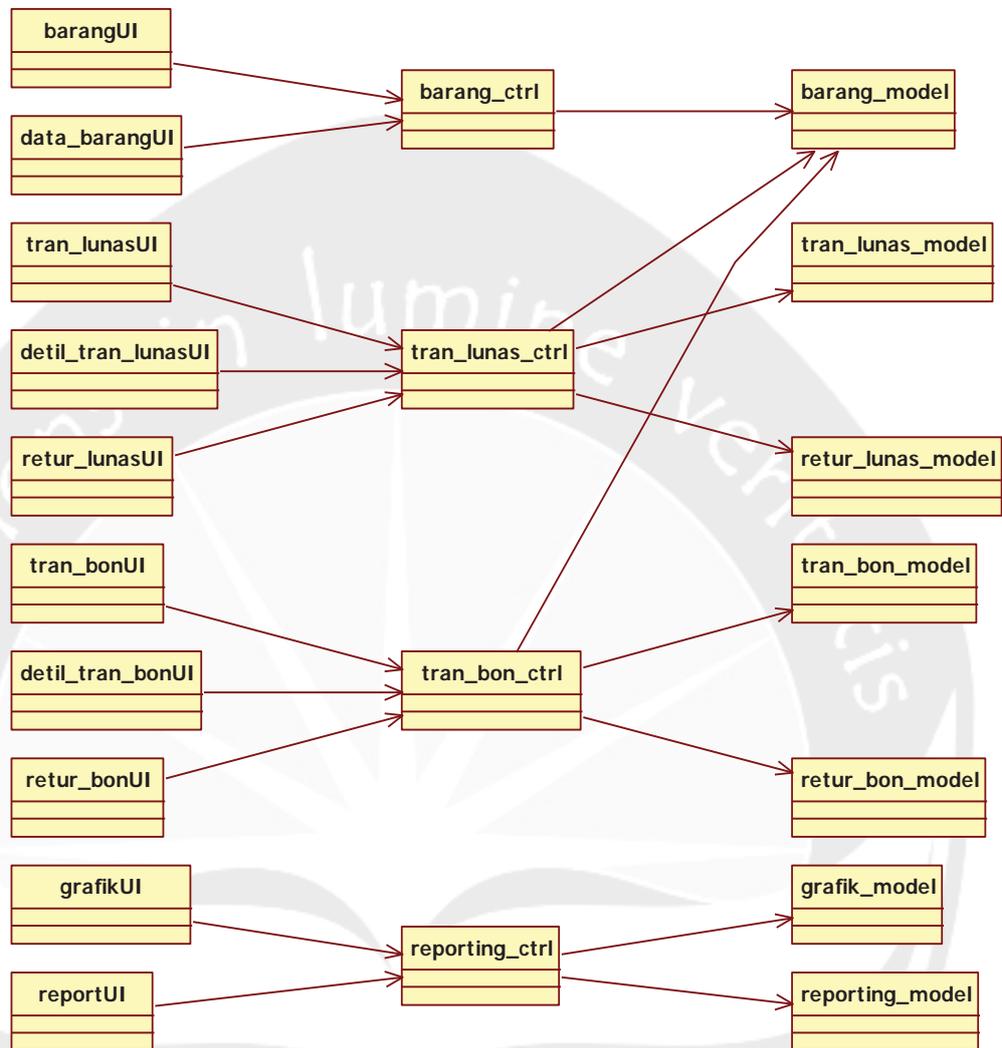
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Thomas Hendri Hananto, *Deskripsi Perancangan Perangkat Lunak (DPPL) SISU*, Universitas Atma Jaya Yogyakarta.
2. Dewi P. Palindih, *Deskripsi Perancangan Perangkat Lunak (DPPL) WELSAP*, Universitas Atma Jaya Yogyakarta.

## Analisis Model

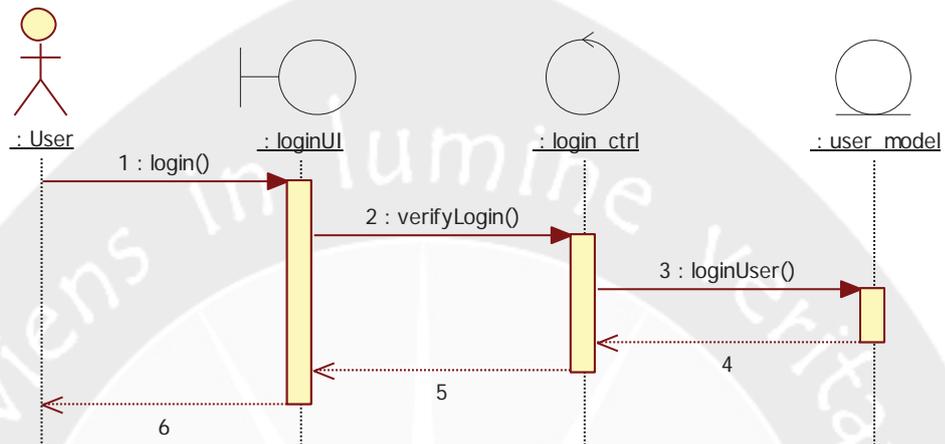
### Perancangan Arsitektur





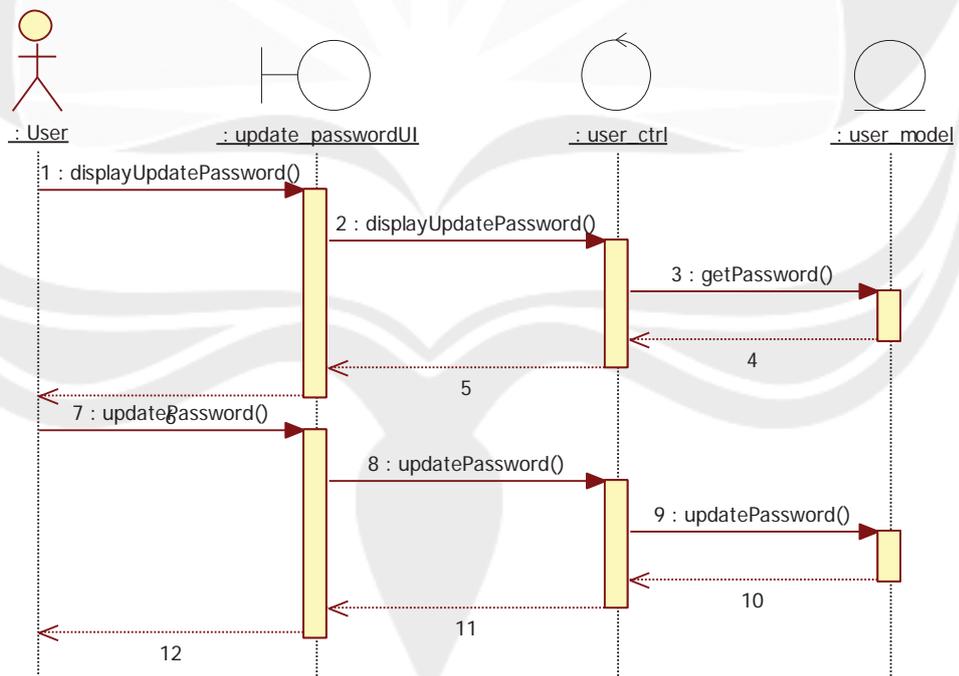
Gambar 0.1 Perancangan Arsitektur

**Perancangan Rinci**  
**Sequence Diagram**  
**Login**



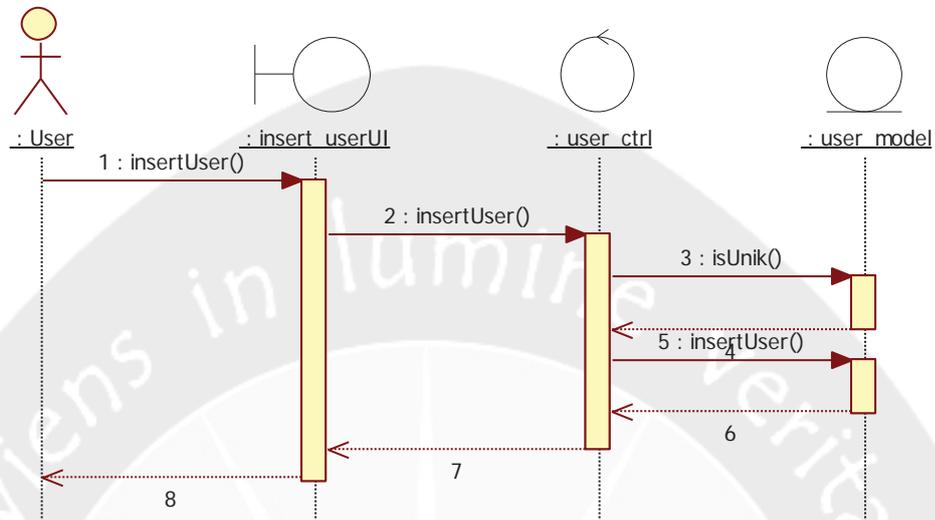
Gambar 0.2 Sequence Diagram: Login

**Ubah Password**



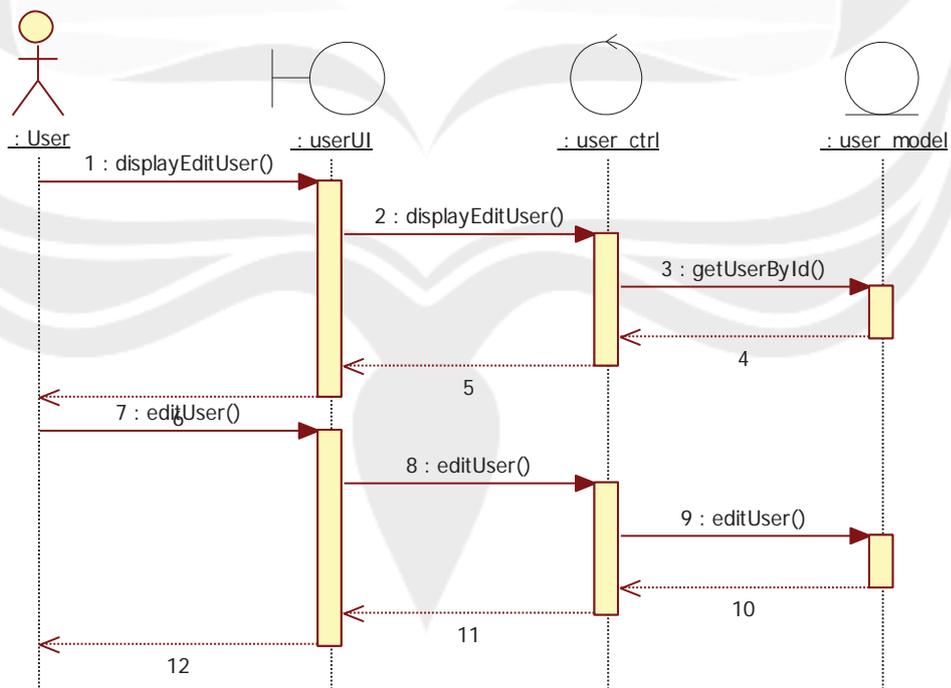
Gambar 0.3 Sequence Diagram: Ubah Password

## Pendaftaran User Baru



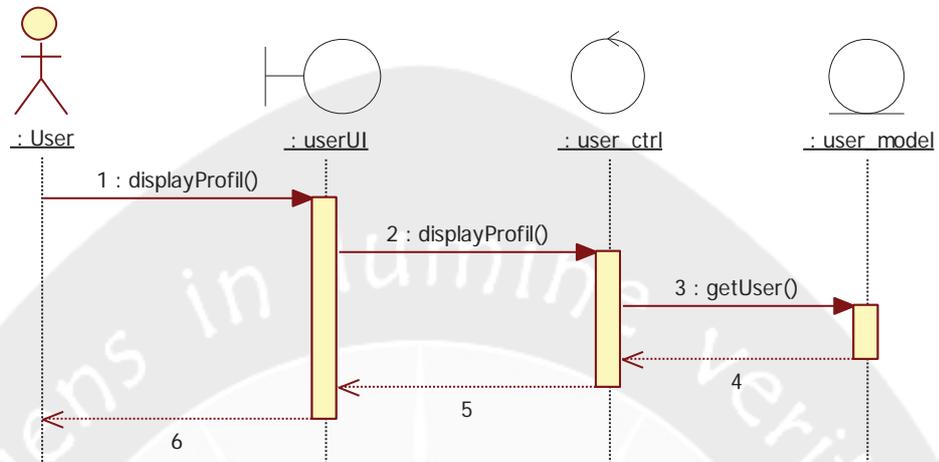
Gambar 0.4 Sequence Diagram: Pendaftaran User Baru

## Pengelolaan User Ubah Data User



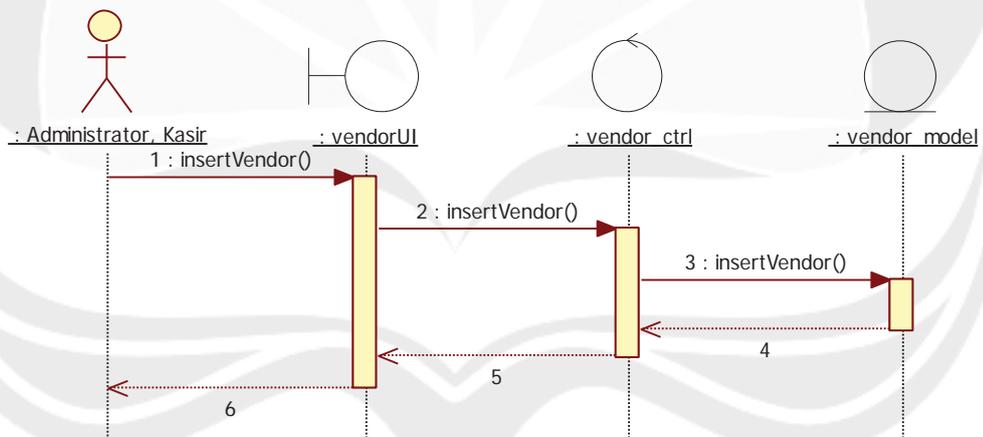
Gambar 0.5 Sequence Diagram: Ubah Data User

## Tampil Data User



Gambar 0.6 Sequence Diagram: Tampil Data User

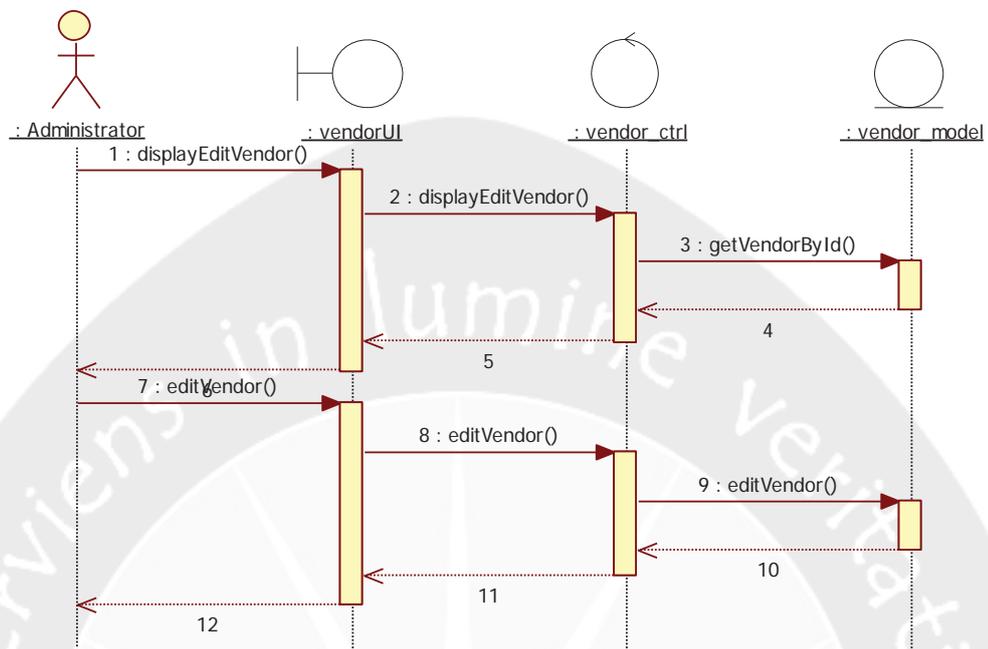
## Pengelolaan Vendor Tambah Vendor



Gambar 0.7 Sequence Diagram: Tambah Vendor

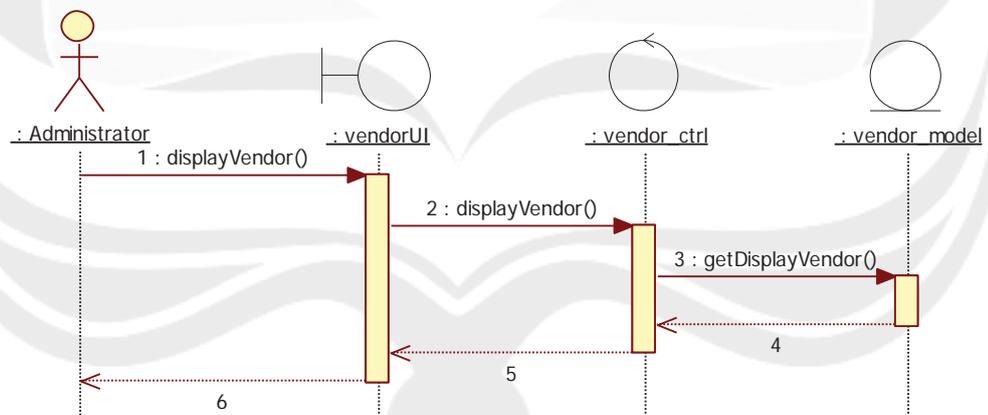
## Ubah Vendor

Program Studi Teknik Informatika	DPPL – SISP	19/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



Gambar 0.8 Sequence Diagram: Ubah Vendor

### Tampil Vendor

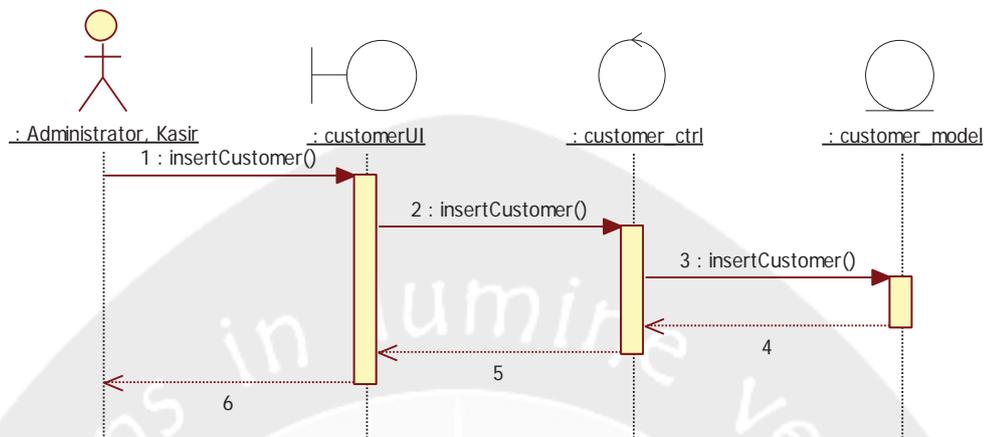


Gambar 0.9 Sequence Diagram: Tampil Vendor

### Pengelolaan Customer

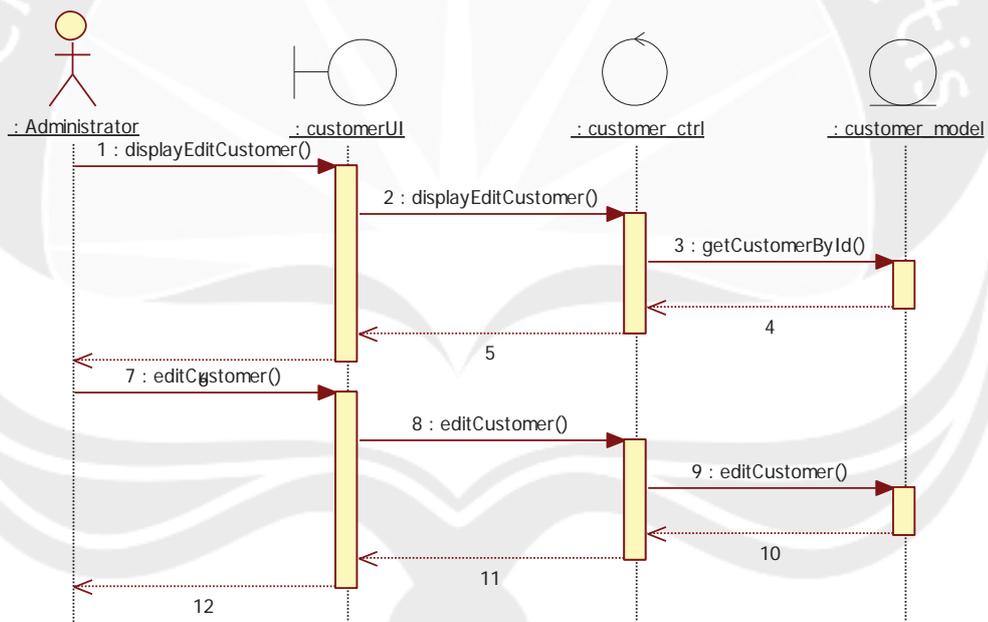
#### Tambah Customer

Program Studi Teknik Informatika	DPPL – SISP	20/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



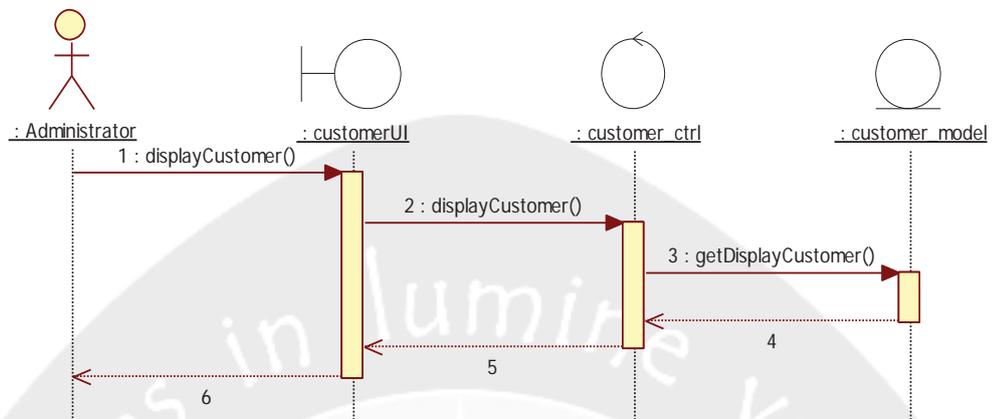
Gambar 0.10 Sequence Diagram: Tambah Customer

### Ubah Customer



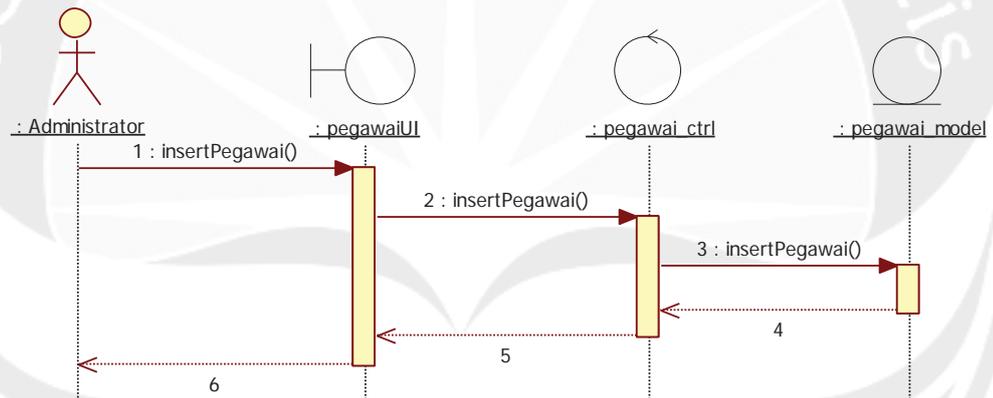
Gambar 0.11 Sequence Diagram: Ubah Customer

### Tampil Customer



Gambar 0.12 Sequence Diagram: Tampil Customer

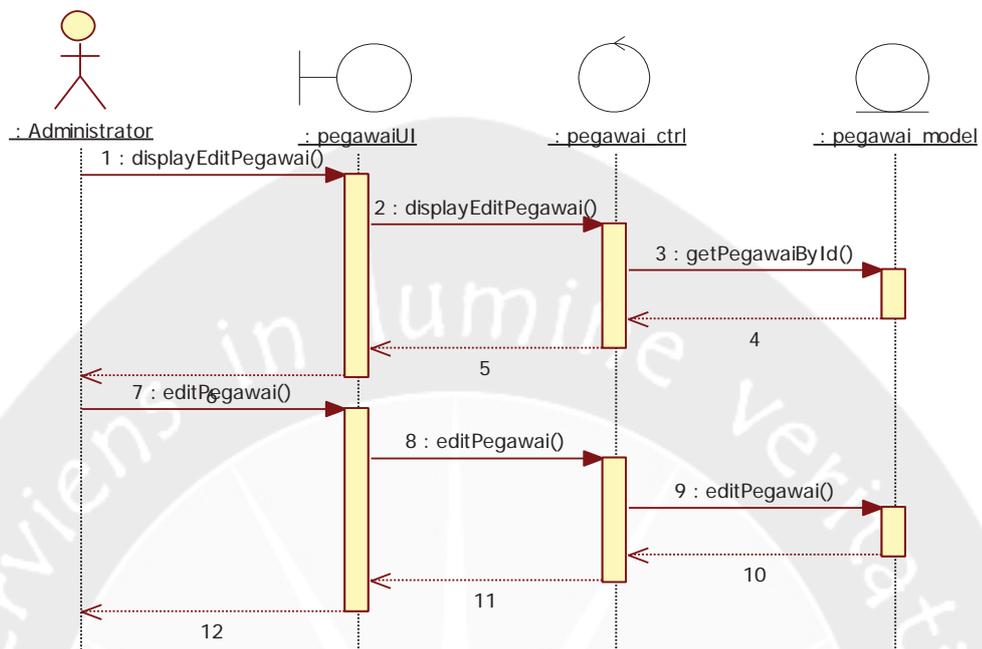
### Pengelolaan Pegawai Tambah Pegawai



Gambar 0.13 Sequence Diagram: Tambah Pegawai

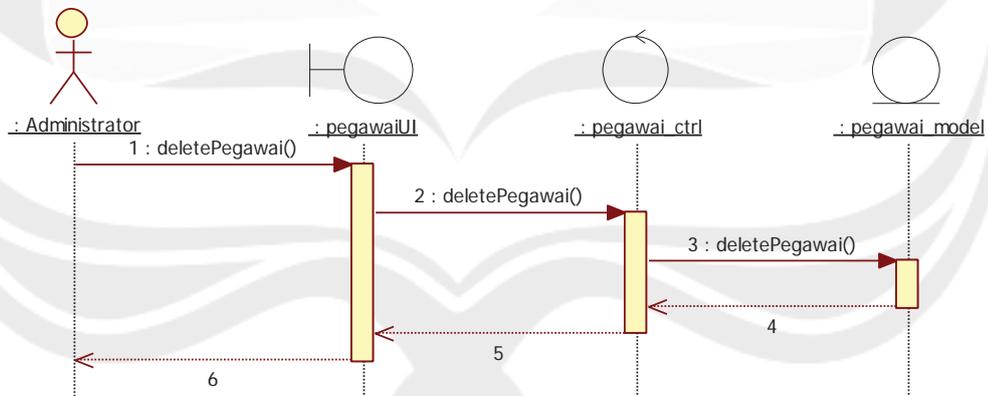
### Ubah Pegawai

Program Studi Teknik Informatika	DPPL – SISP	22/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



Gambar 0.14 Sequence Diagram: Ubah Pegawai

### Hapus Pegawai

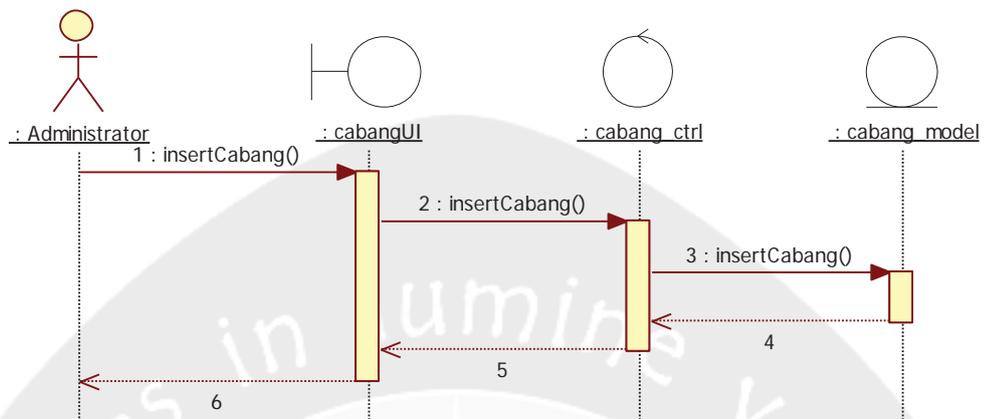


Gambar 0.15 Sequence Diagram: Hapus Pegawai

### Pengelolaan Cabang

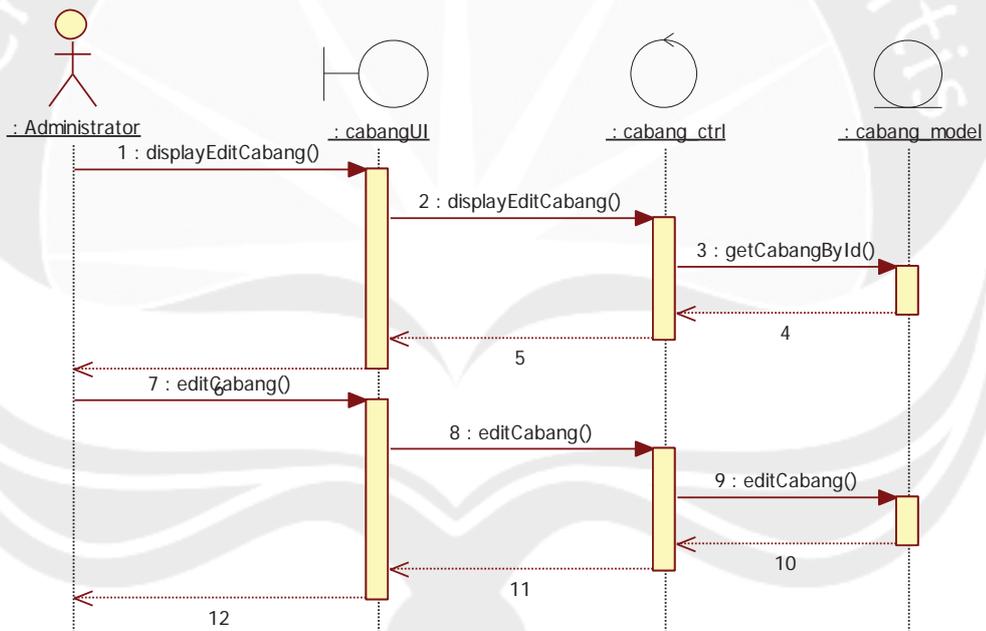
#### Tambah Cabang

Program Studi Teknik Informatika	DPPL – SISP	23/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



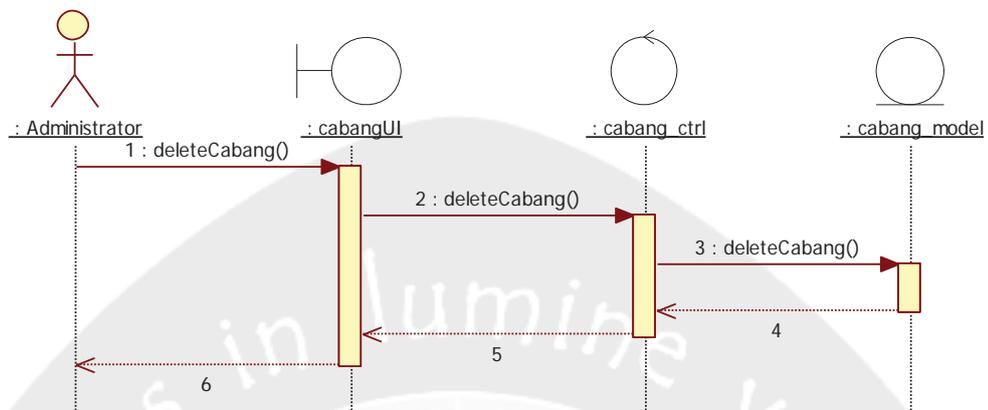
Gambar 0.16 Sequence Diagram: Tambah Cabang

### Ubah Cabang



Gambar 0.17 Sequence Diagram: Ubah Cabang

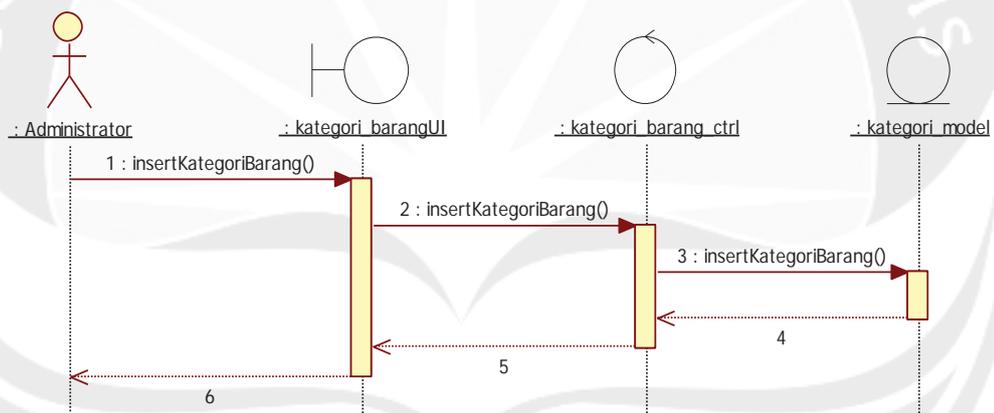
### Hapus Cabang



Gambar 0.18 Sequence Diagram: Hapus Cabang

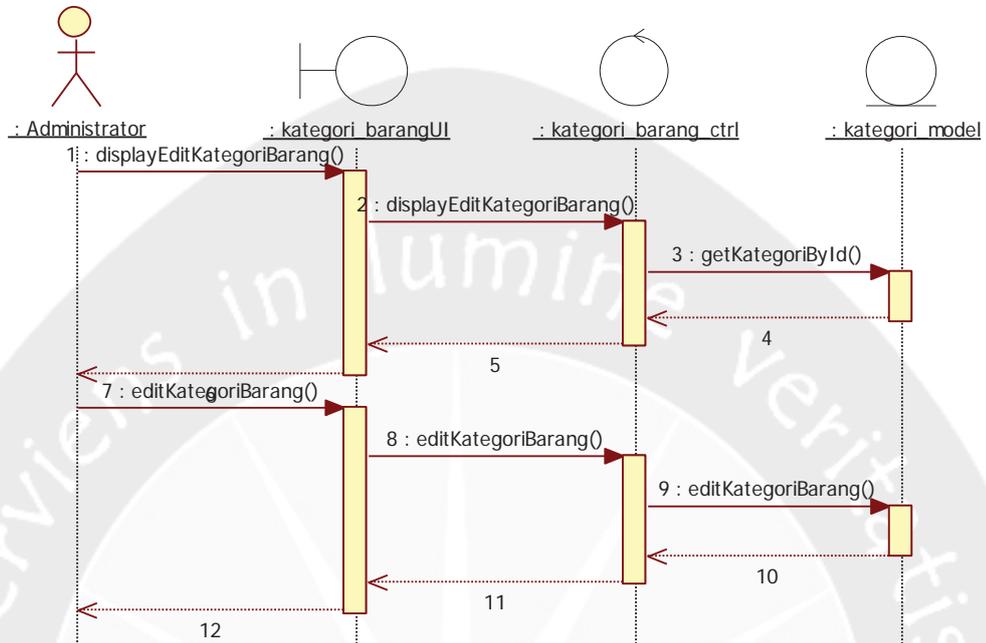
### Pengelolaan Kategori Barang

#### Tambah Kategori Barang



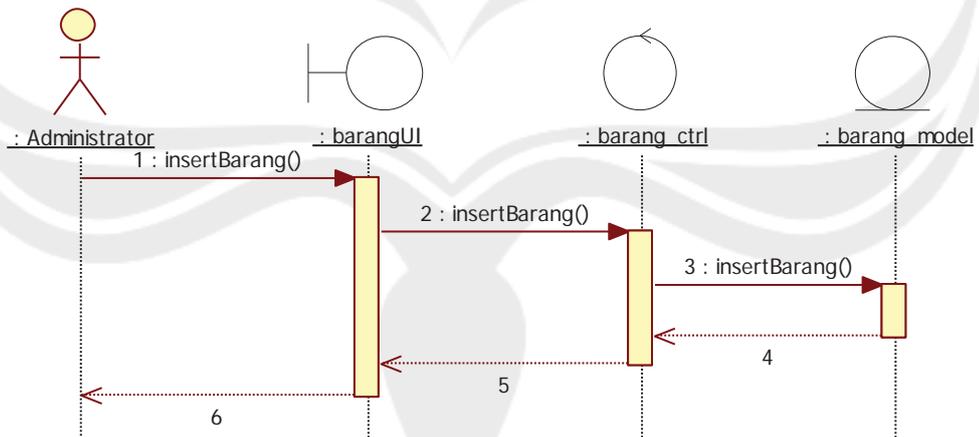
Gambar 0.19 Sequence Diagram: Tambah Kategori Barang

## Ubah Kategori Barang



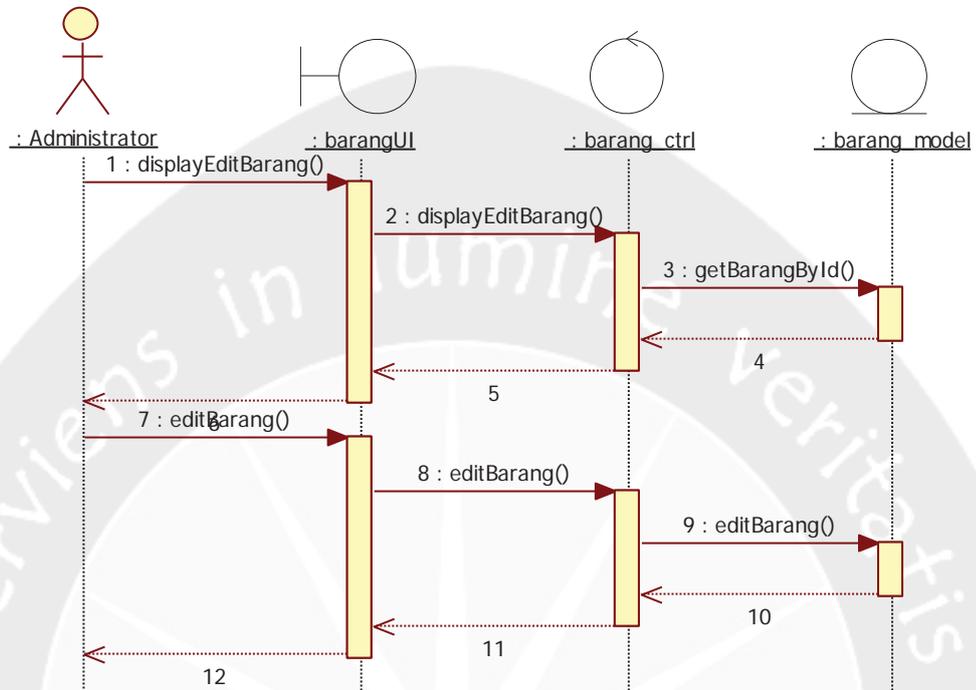
Gambar 0.20 Sequence Diagram: Ubah Kategori Barang

## Pengelolaan Barang Tambah Barang



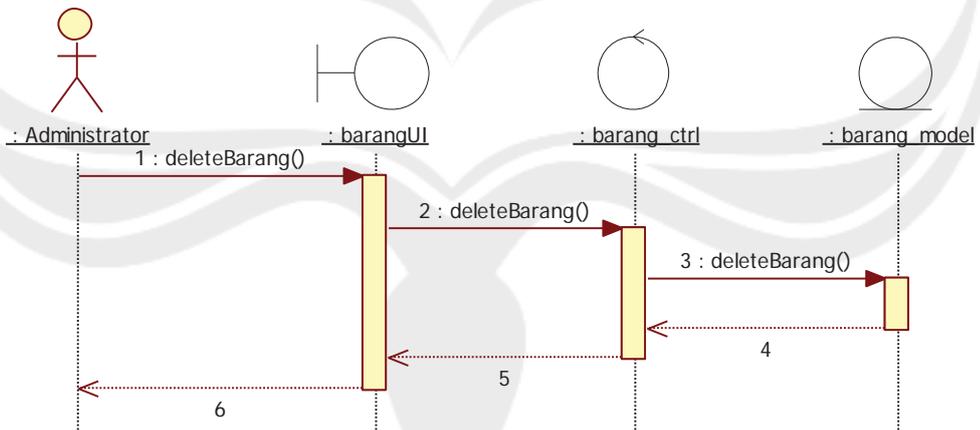
Gambar 0.21 Sequence Diagram: Tambah Barang

## Ubah Barang



Gambar 0.22 Sequence Diagram: Ubah Barang

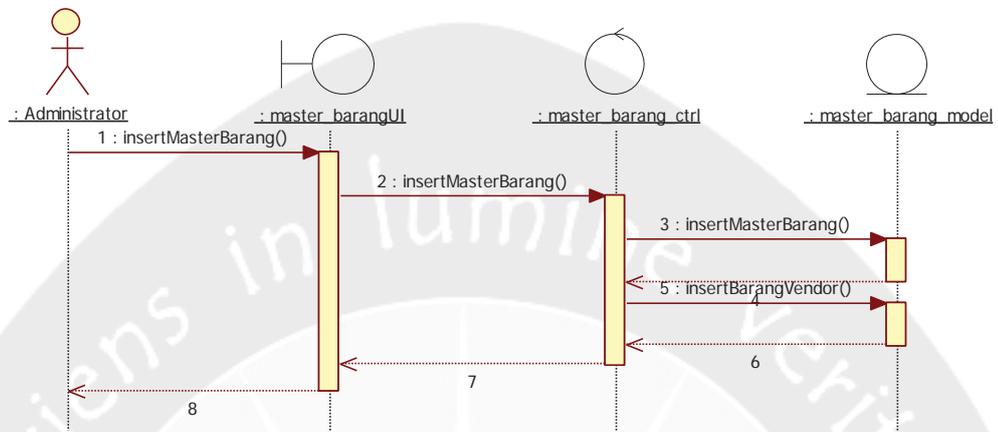
## Hapus Barang



Gambar 0.23 Sequence Diagram: Hapus Barang

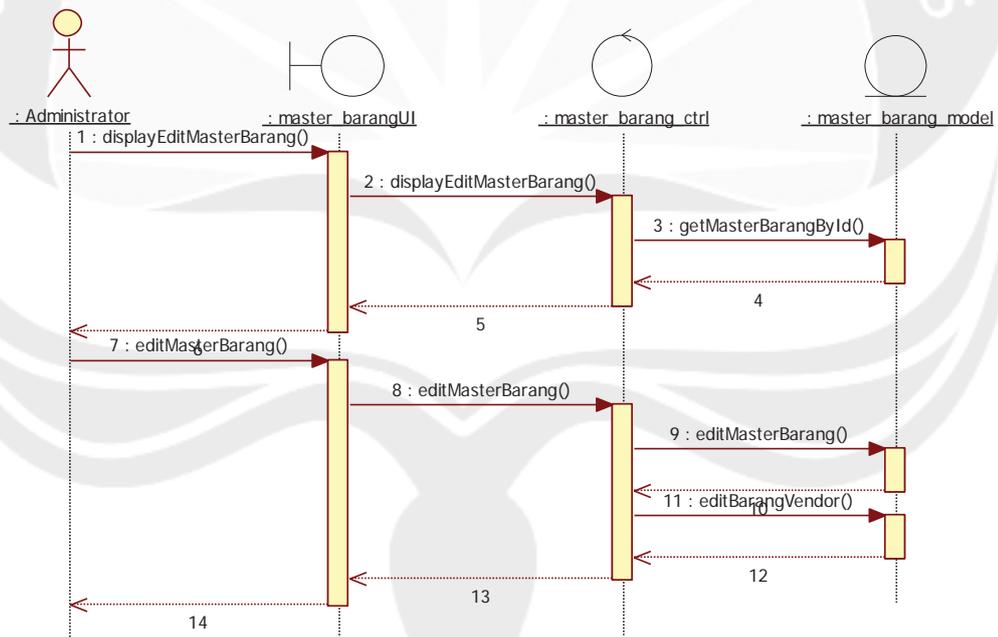
## Pengelolaan Master Barang

### Tambah Master Barang



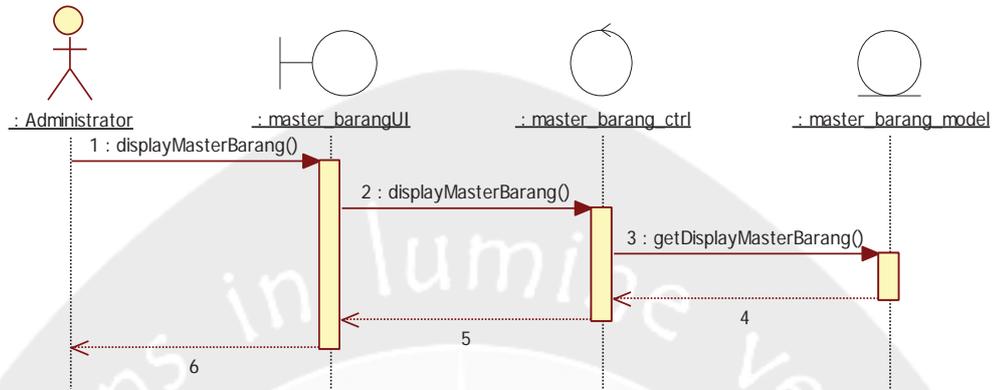
Gambar 0.24 Sequence Diagram: Tambah Master Barang

### Ubah Master Barang



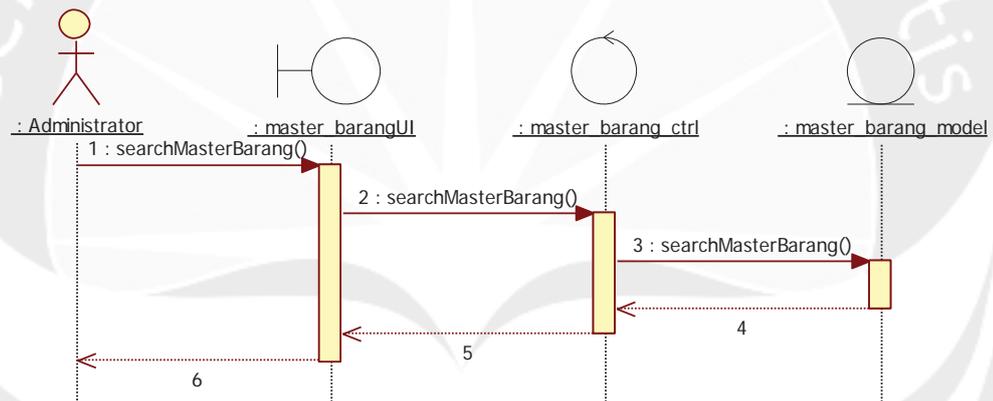
Gambar 0.25 Sequence Diagram: Ubah Master Barang

## Tampil Master Barang



Gambar 0.26 Sequence Diagram: Tampil Master Barang

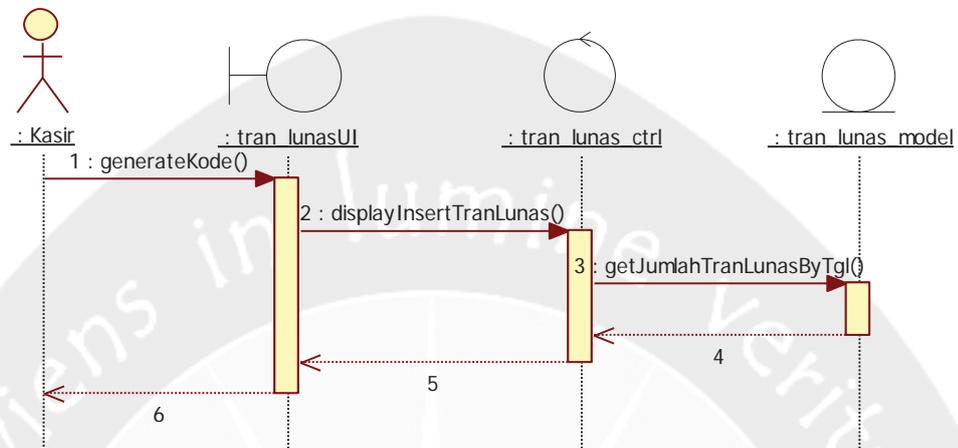
## Cari Master Barang



Gambar 0.27 Sequence Diagram: Cari Master Barang

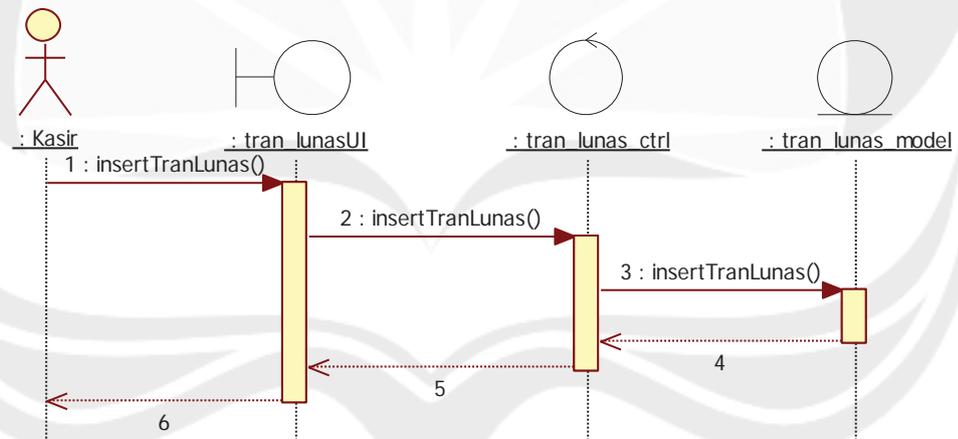
## Pengelolaan Transaksi Lunas

### Generate Kode Transaksi Lunas



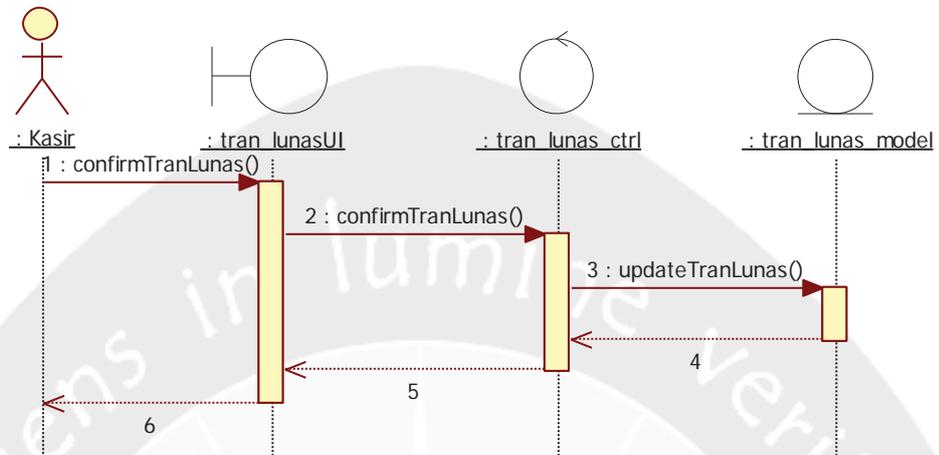
Gambar 0.28 Sequence Diagram: Generate Kode Transaksi Lunas

### Tambah Transaksi Lunas



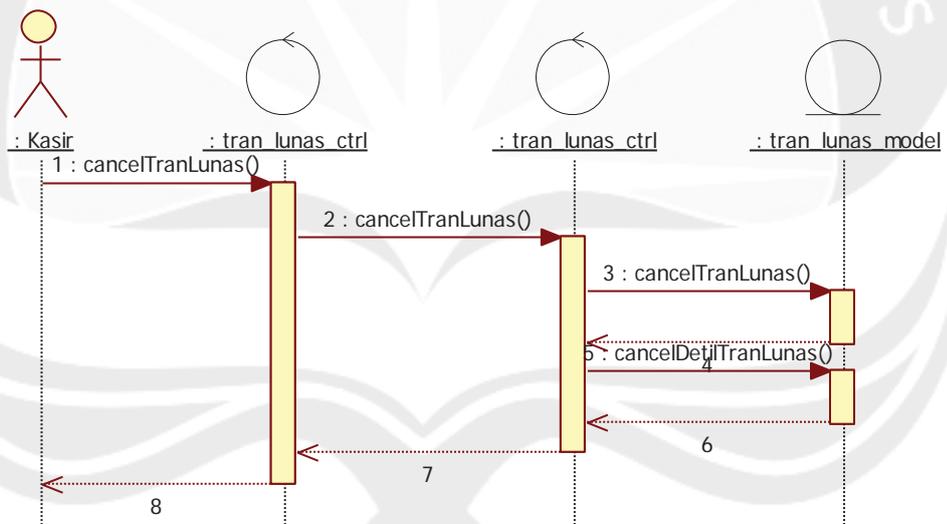
Gambar 0.29 Sequence Diagram: Tambah Transaksi Lunas

### Konfirmasi Transaksi Lunas



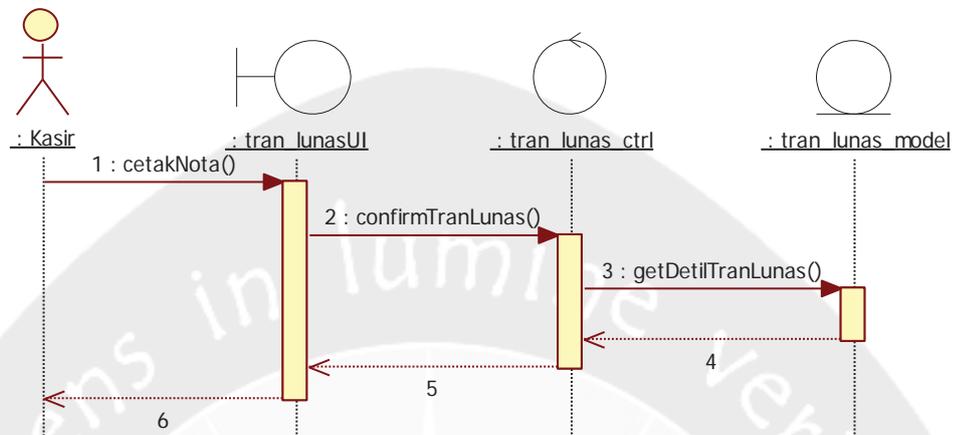
Gambar 0.30 Sequence Diagram: Konfirmasi Transaksi Lunas

### Batal Transaksi Lunas



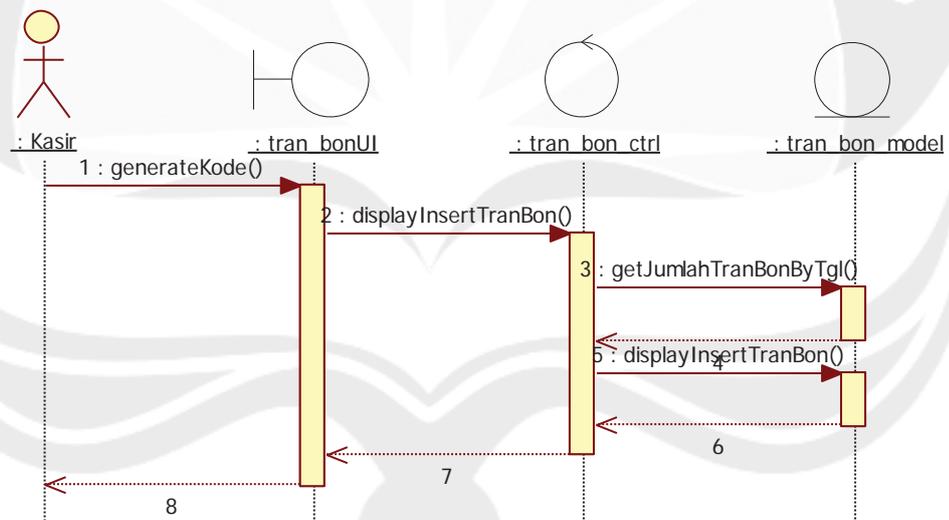
Gambar 0.31 Sequence Diagram: Batal Transaksi Lunas

### Cetak Nota Transaksi Lunas



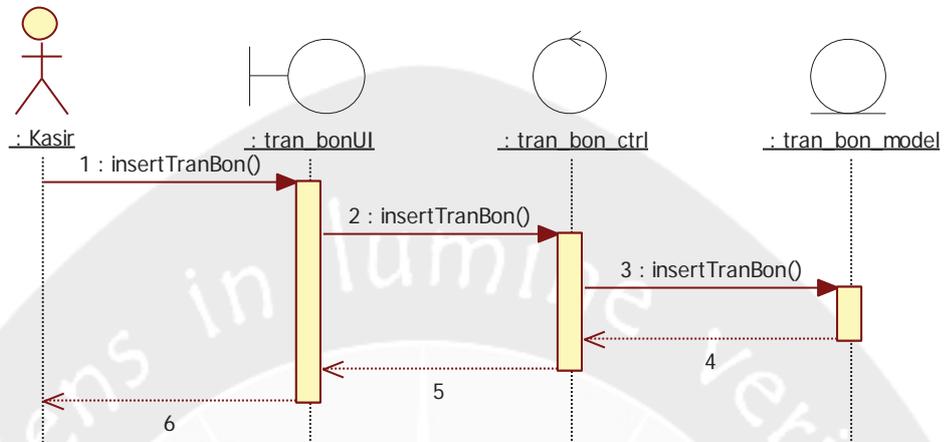
Gambar 0.32 Sequence Diagram: Cetak Nota Transaksi Lunas

### Pengelolaan Transaksi Bon Generate Kode Transaksi Bon



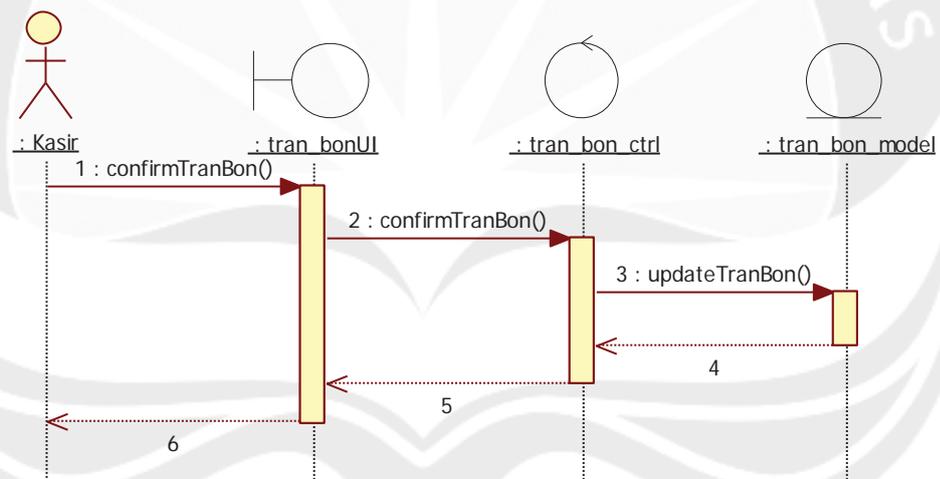
Gambar 0.33 Sequence Diagram: Generate Kode Transaksi Bon

### Tambah Transaksi Bon



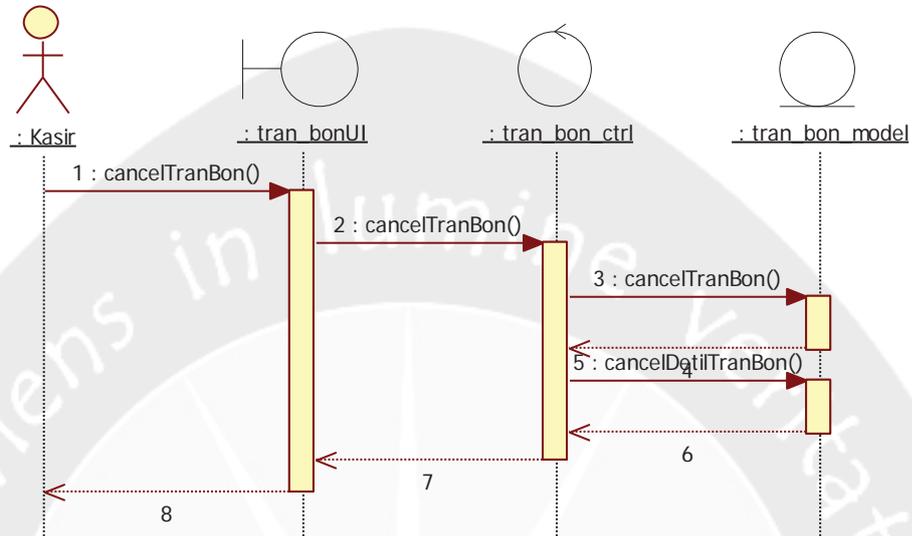
Gambar 0.34 Sequence Diagram: Tambah Transaksi Bon

### Konfirmasi Transaksi Bon



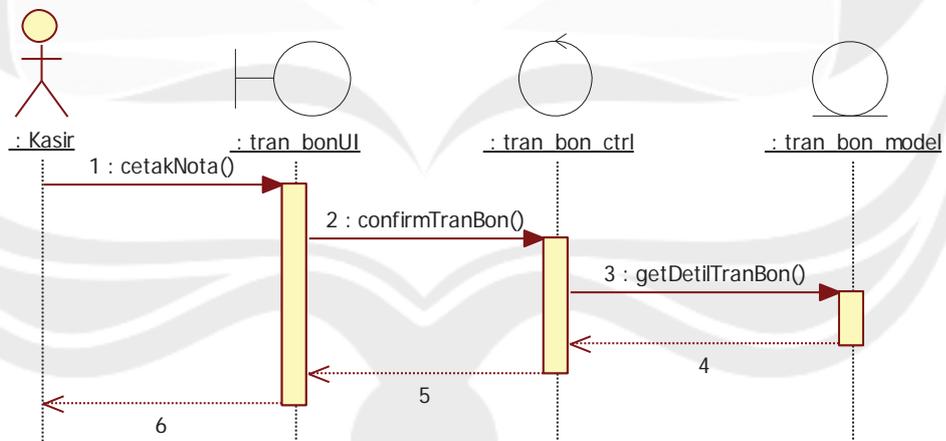
Gambar 0.35 Sequence Diagram: Konfirmasi Transaksi Bon

### Batal Transaksi Bon



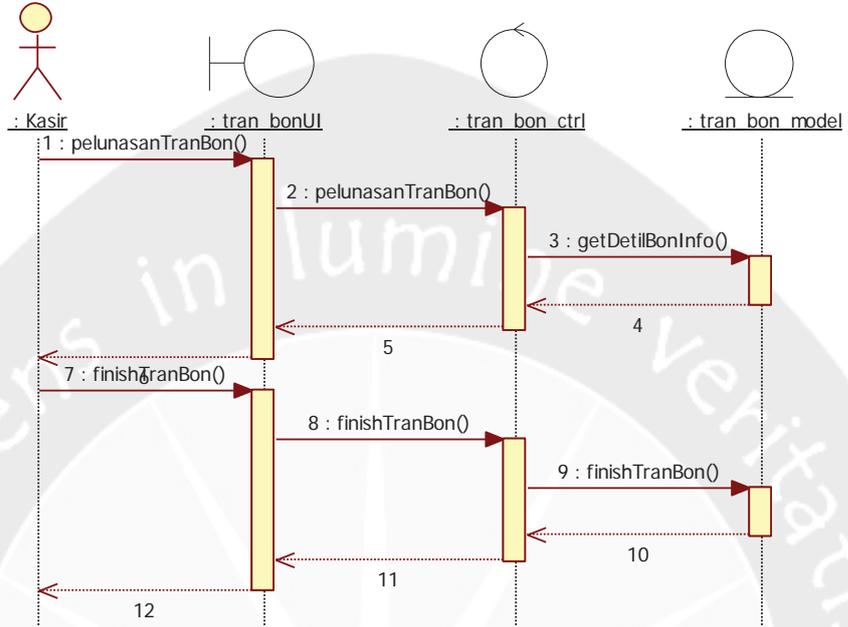
Gambar 0.36 Sequence Diagram: Batal Transaksi Bon

### Cetak Nota Transaksi Bon



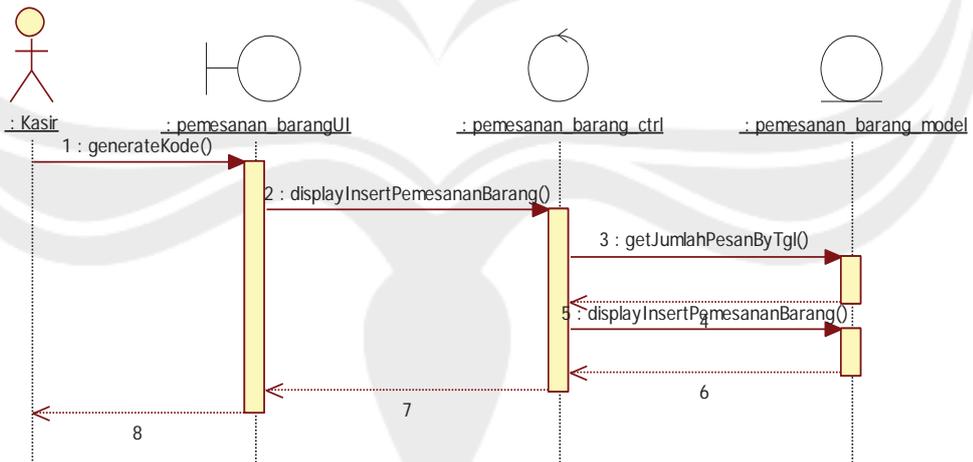
Gambar 0.37 Sequence Diagram: Cetak Nota Transaksi Bon

### Pelunasan Transaksi Bon



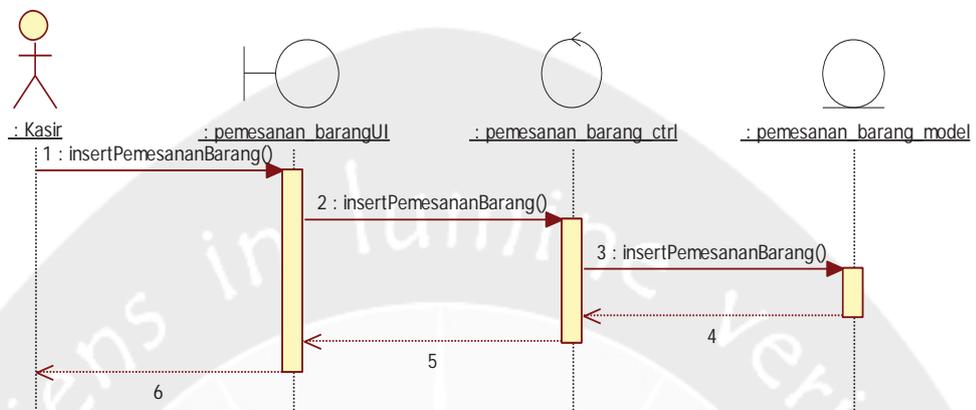
Gambar 0.38 Sequence Diagram: Pelunasan Transaksi Bon

### Pengelolaan Pemesanan Barang Generate Kode Pemesanan Barang



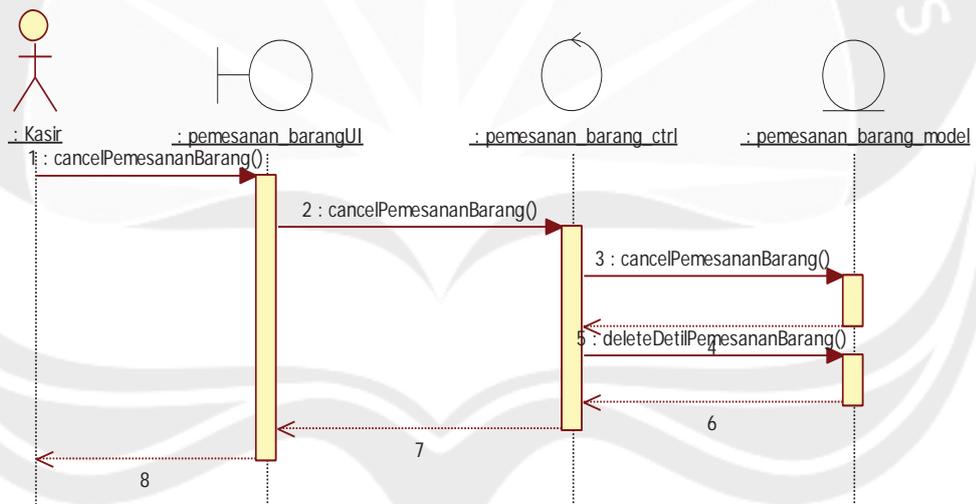
Gambar 0.39 Sequence Diagram: Generate Kode Pemesanan Barang

## Tambah Pemesanan Barang



Gambar 0.40 Sequence Diagram: Tambah Pemesanan Barang

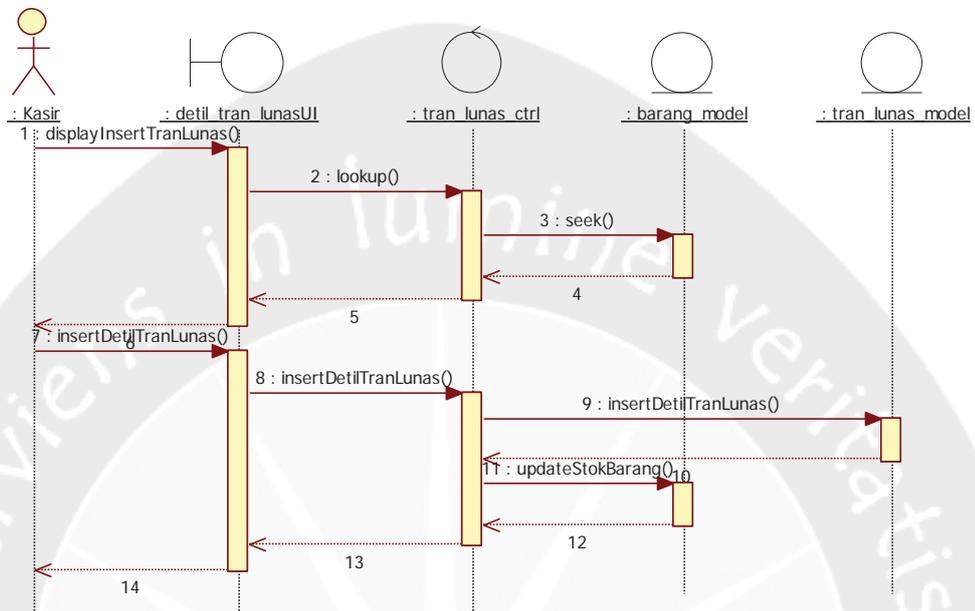
## Batal Pemesanan Barang



Gambar 0.41 Sequence Diagram: Batal Pemesanan Barang

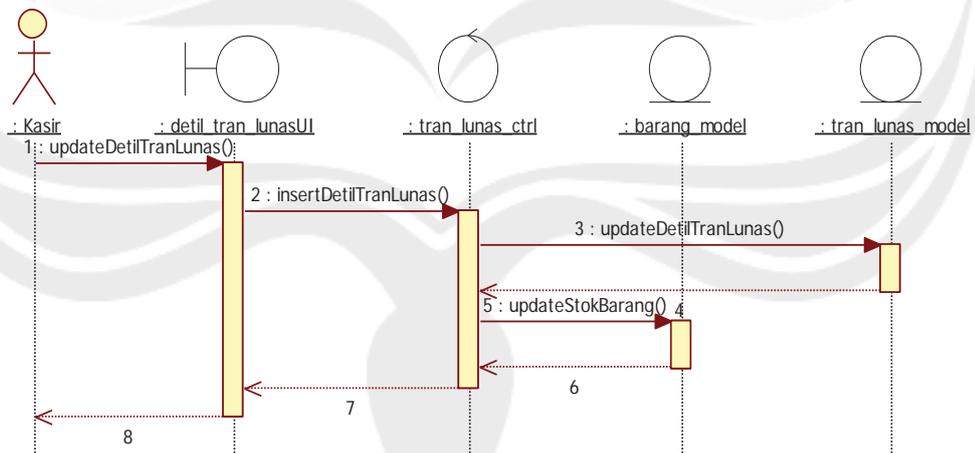
## Pengelolaan Detil Transaksi Lunas

### Tambah Detil Transaksi Lunas



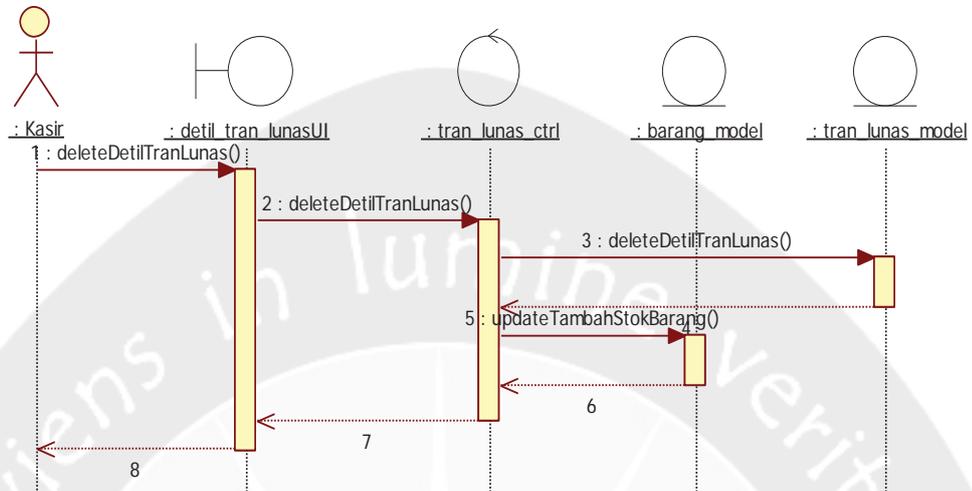
Gambar 0.42 Sequence Diagram: Tambah Detil Transaksi Lunas

### Ubah Detil Transaksi Lunas



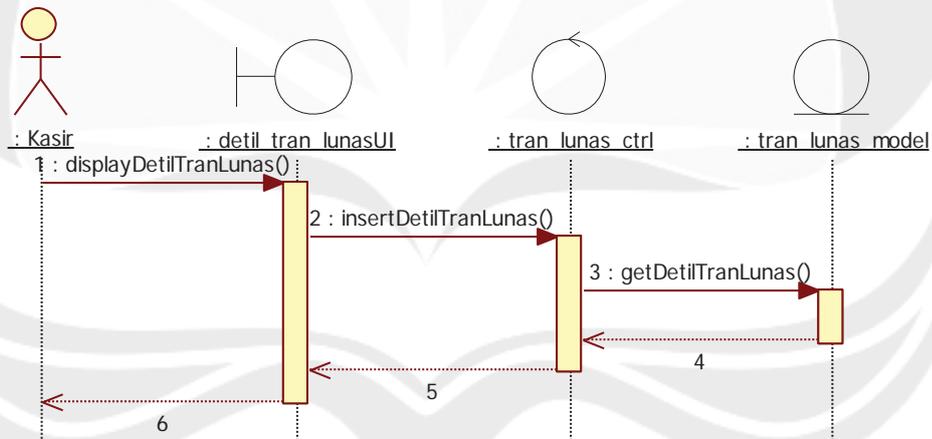
Gambar 0.43 Sequence Diagram: Ubah Detil Transaksi Lunas

### Hapus Detil Transaksi Lunas



Gambar 0.44 Sequence Diagram: Hapus Detil Transaksi Lunas

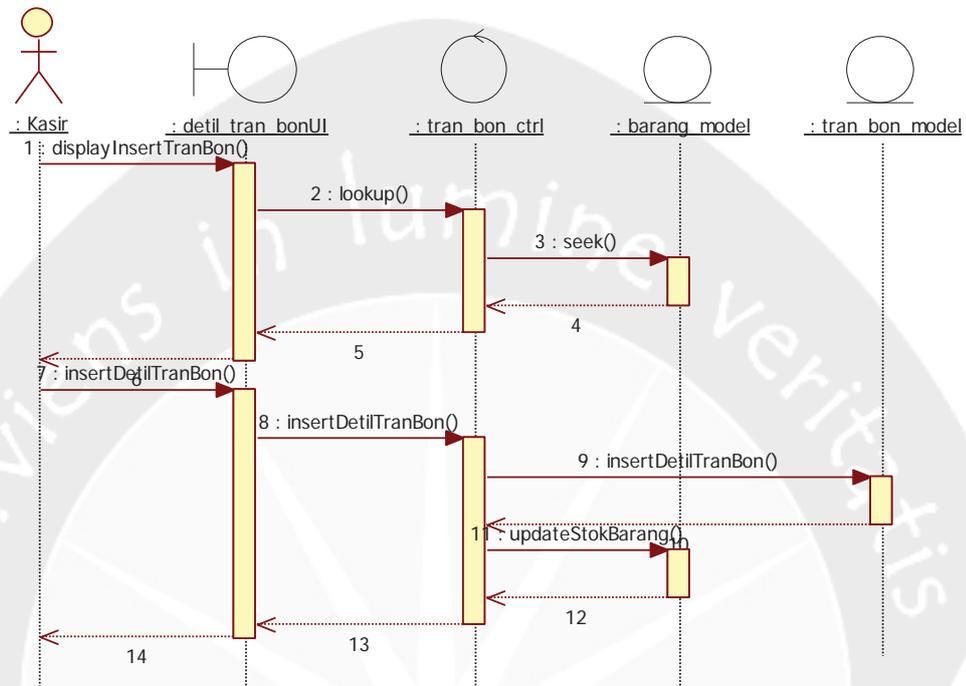
### Tampil Detil Transaksi Lunas



Gambar 0.45 Sequence Diagram: Tampil Detil Transaksi Lunas

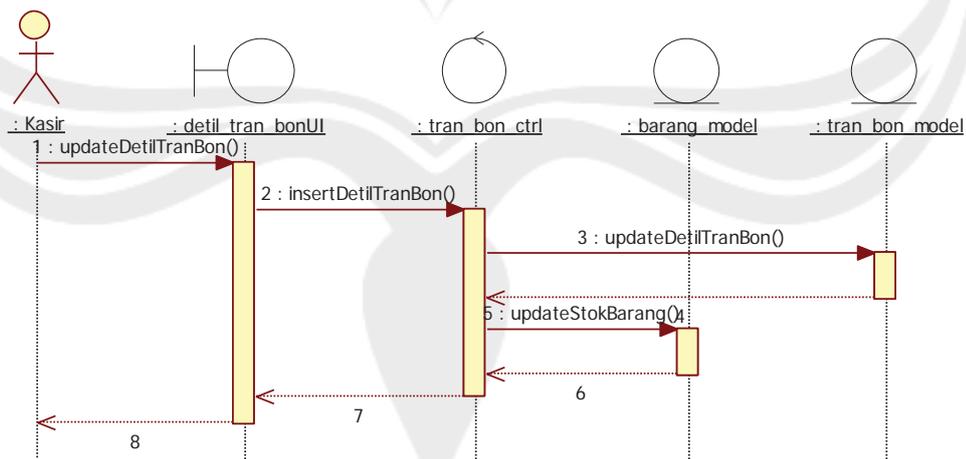
## Pengelolaan Detil Transaksi Bon

### Tambah Detil Transaksi Bon



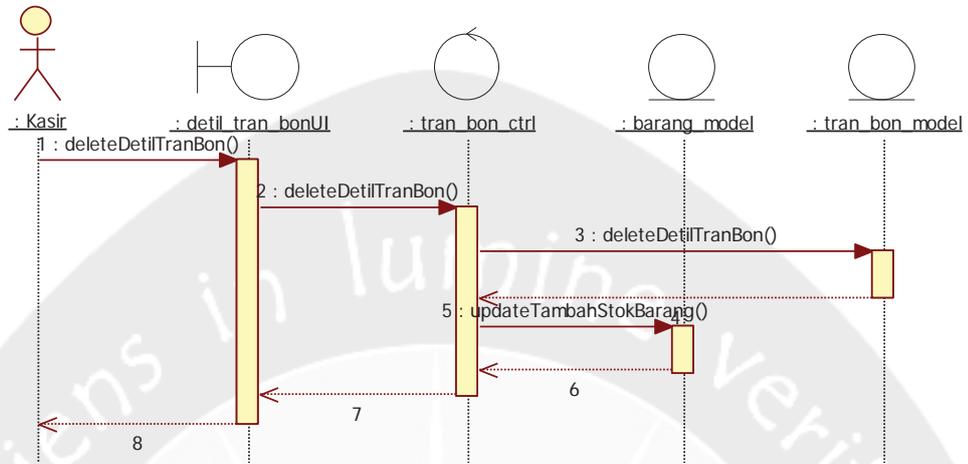
Gambar 0.46 Sequence Diagram: Tambah Detil Transaksi Bon

### Ubah Detil Transaksi Bon



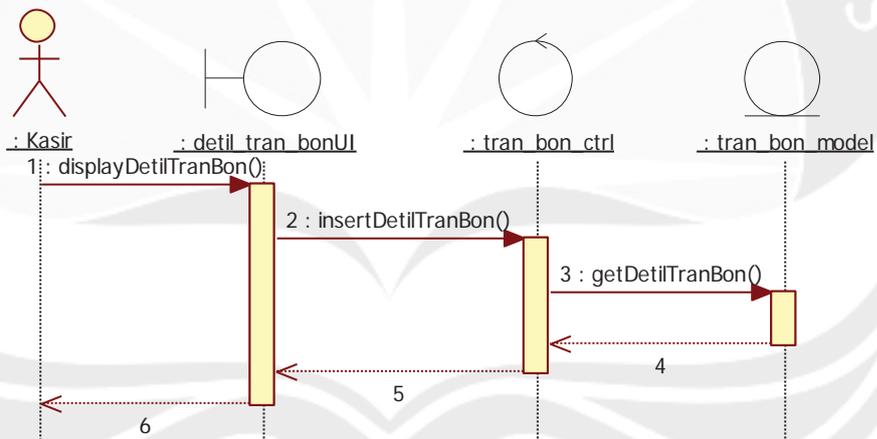
Gambar 0.47 Sequence Diagram: Ubah Detil Transaksi Bon

### Hapus Detil Transaksi Bon



Gambar 0.48 Sequence Diagram: Hapus Detil Transaksi Bon

### Tampil Detil Transaksi Bon

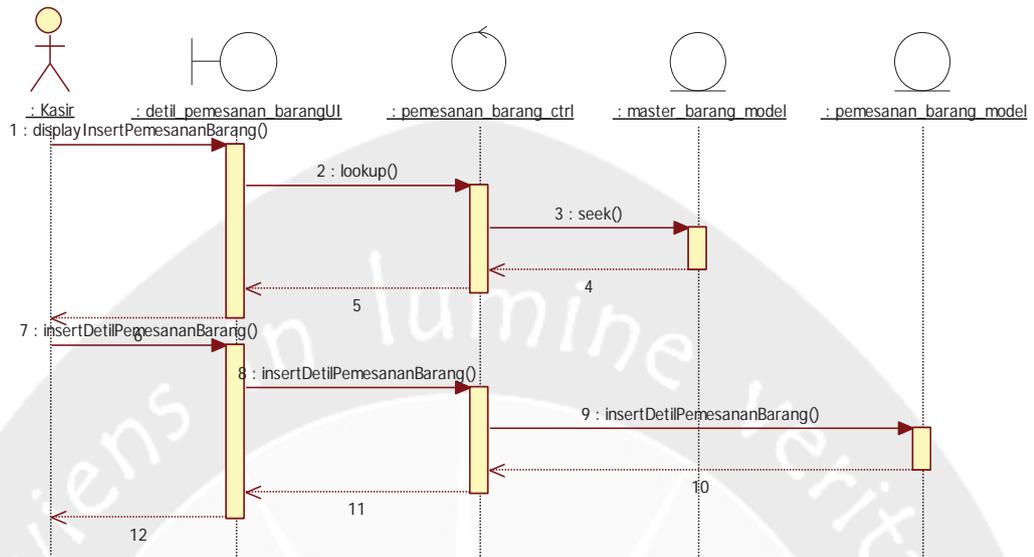


Gambar 0.49 Sequence Diagram: Tampil Detil Transaksi Bon

### Pengelolaan Detil Pemesanan Barang

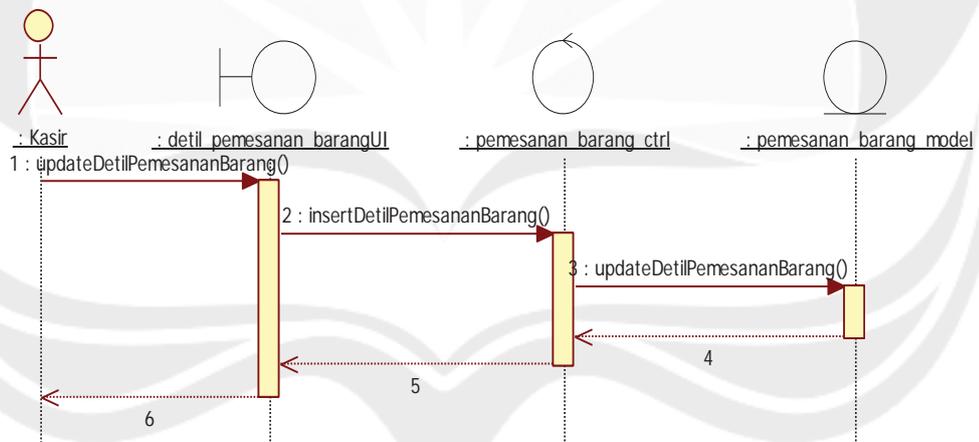
#### Tambah Detil Pemesanan Barang

Program Studi Teknik Informatika	DPPL – SISP	40/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



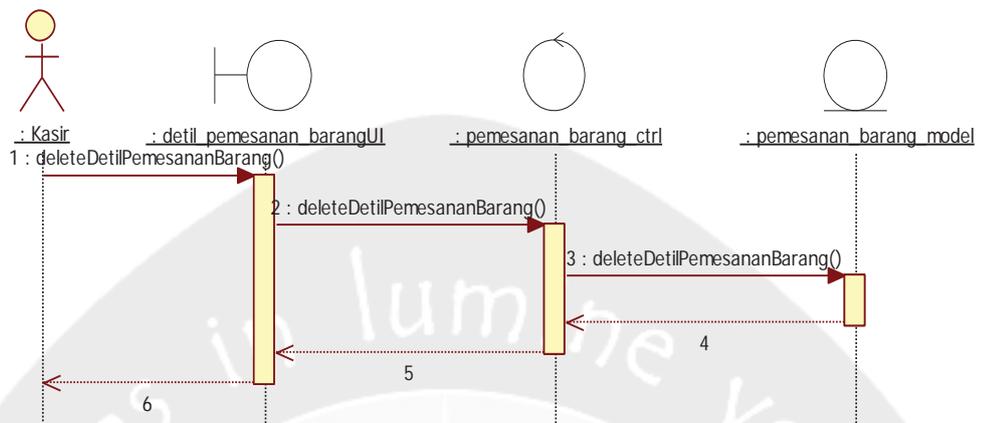
Gambar 0.50 Sequence Diagram: Tambah Detil Pemesanan Barang

### Ubah Detil Pemesanan Barang



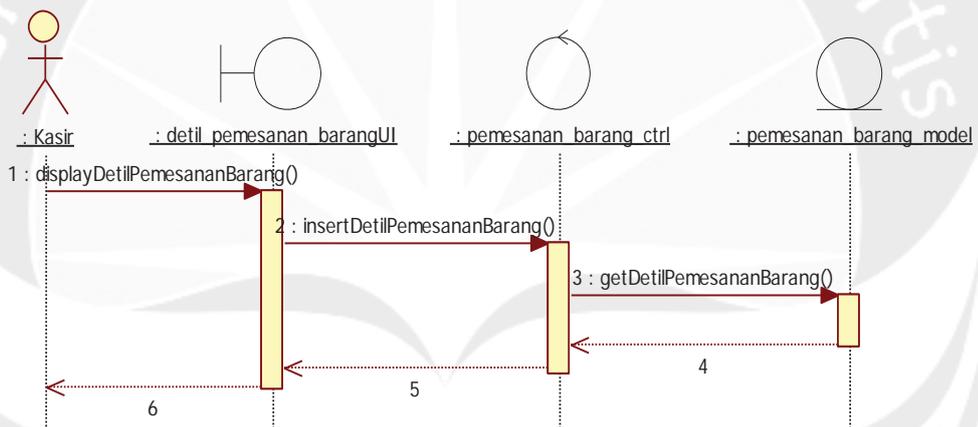
Gambar 0.51 Sequence Diagram: Ubah Detil Pemesanan Barang

### Hapus Detil Pemesanan Barang



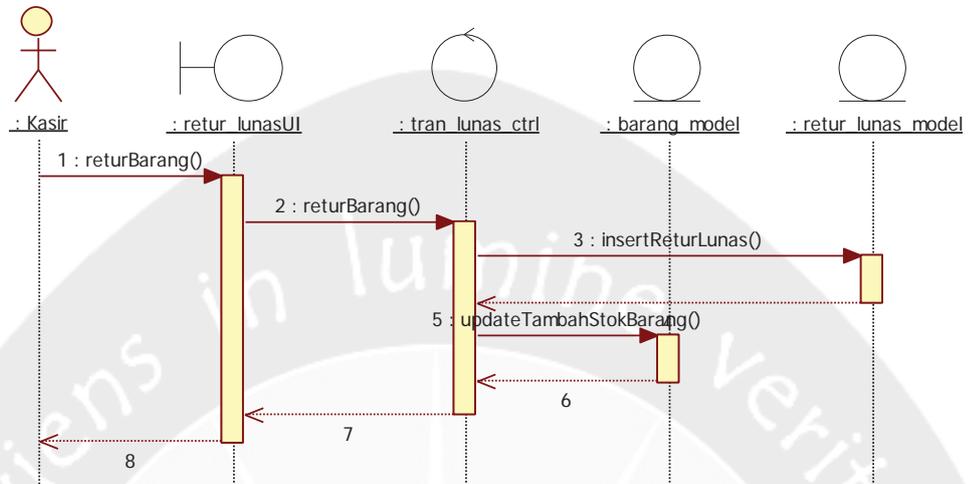
Gambar 0.52 Sequence Diagram: Hapus Detil Pemesanan Barang

### Tampil Detil Pemesanan Barang



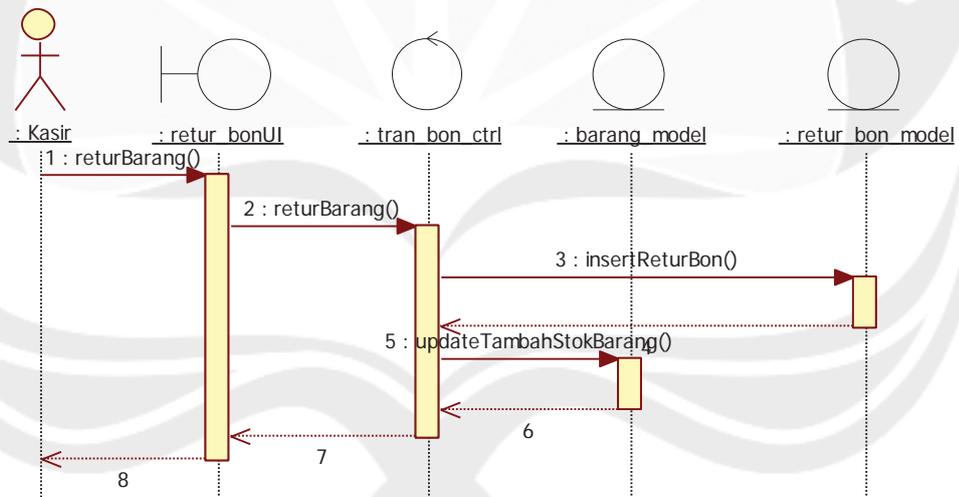
Gambar 0.53 Sequence Diagram: Tampil Detil Pemesanan Barang

## Retur Transaksi Lunas



Gambar 0.54 Sequence Diagram: Retur Transaksi Lunas

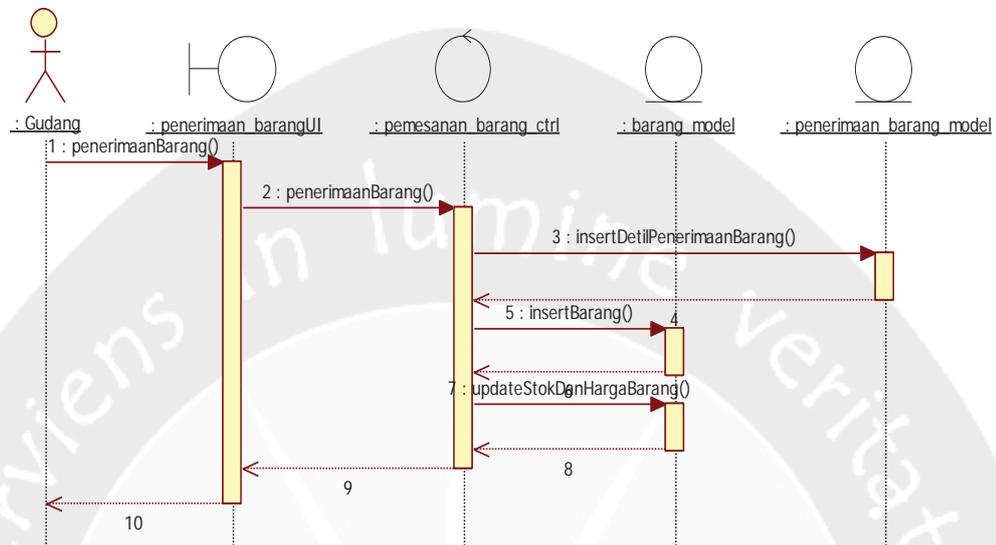
## Retur Transaksi Bon



Gambar 0.55 Sequence Diagram: Retur Transaksi Bon

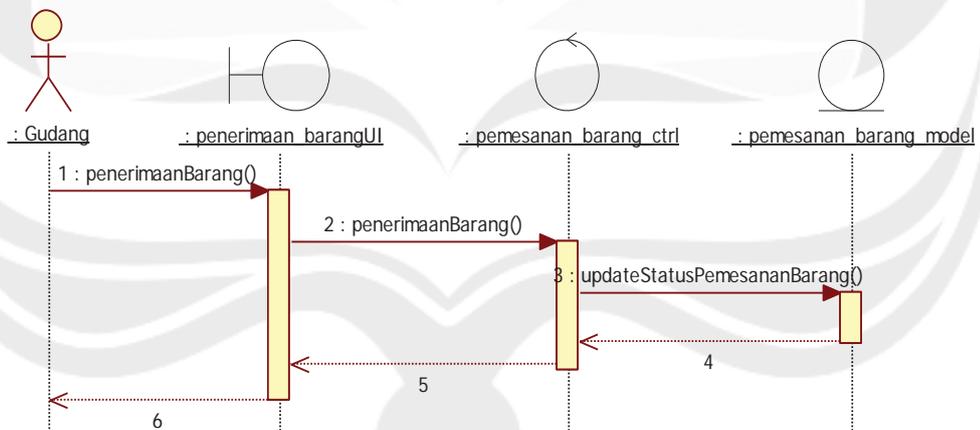
## Pengelolaan Penerimaan Barang

### Tambah Penerimaan Barang



Gambar 0.56 Sequence Diagram: Tambah Penerimaan Barang

### Ubah Status Pemesanan Barang

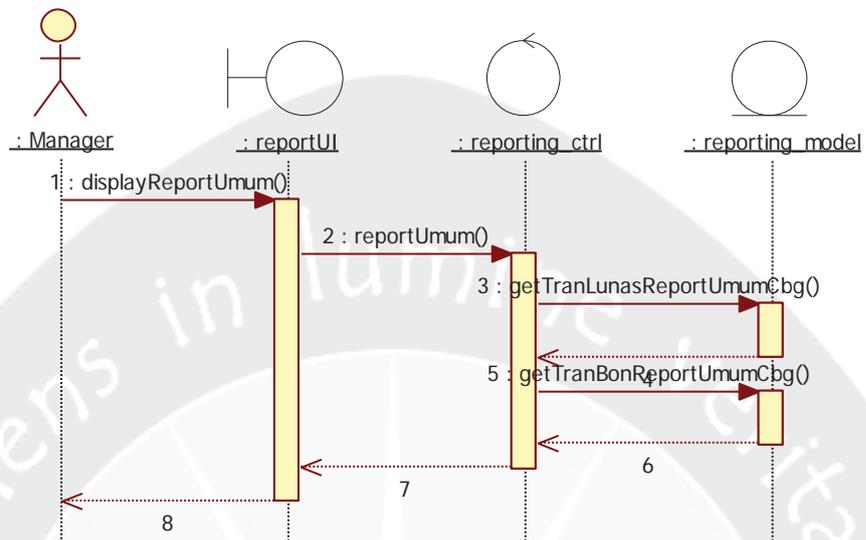


Gambar 0.57 Sequence Diagram: Ubah Status Pemesanan Barang

## Reporting

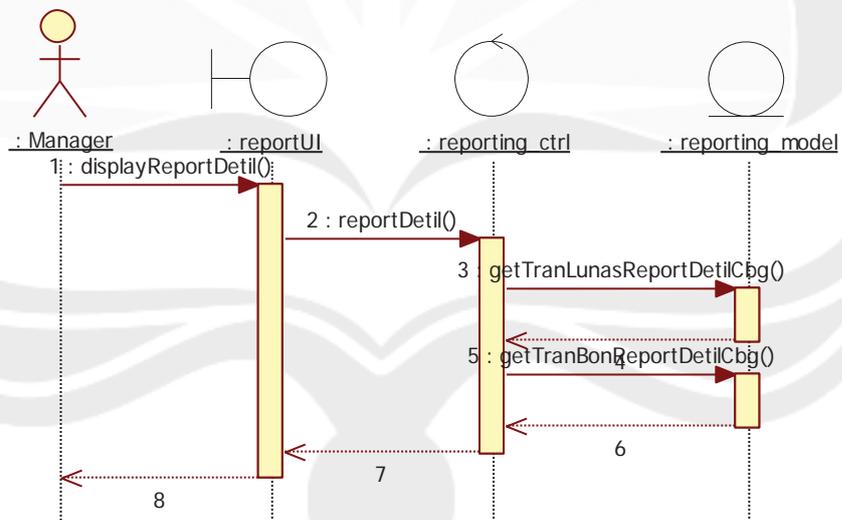
### Laporan Penjualan Umum

Program Studi Teknik Informatika	DPPL – SISP	44/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



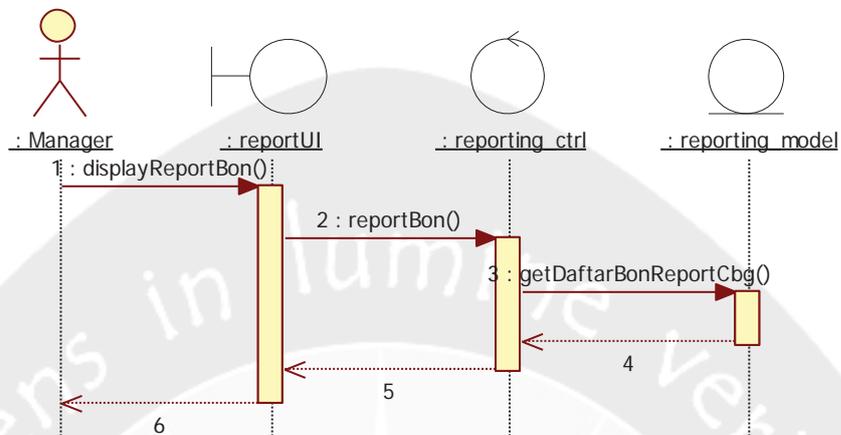
Gambar 0.58 Sequence Diagram: Laporan Penjualan Umum

### Laporan Penjualan Detil



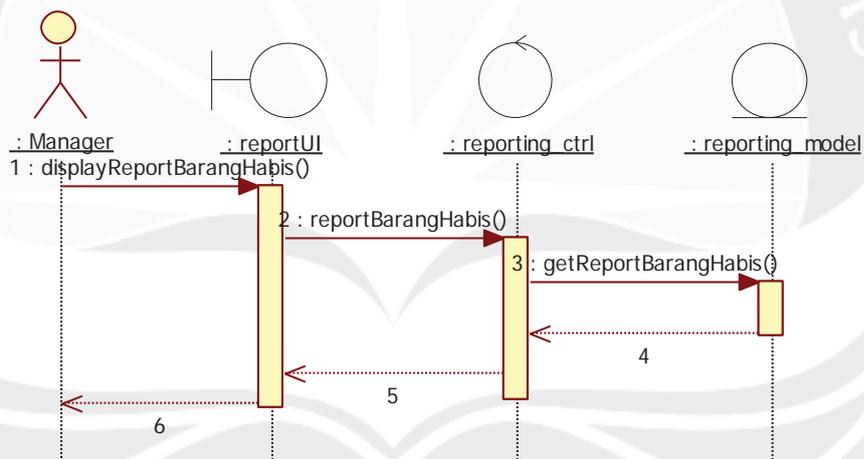
Gambar 0.59 Sequence Diagram: Laporan Penjualan Detil

## Laporan Bon



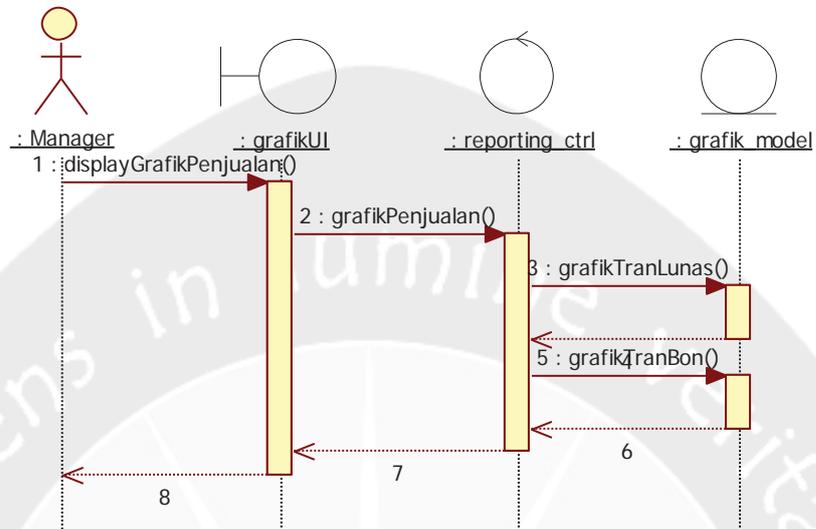
Gambar 0.60 Sequence Diagram: Laporan Bon

## Laporan Daftar Barang Habis



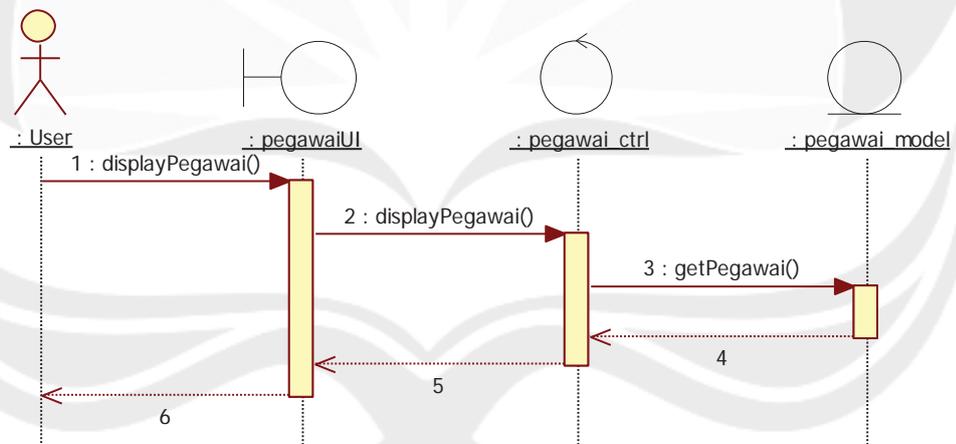
Gambar 0.61 Sequence Diagram: Laporan Daftar Barang Habis

## Grafik Laporan Penjualan



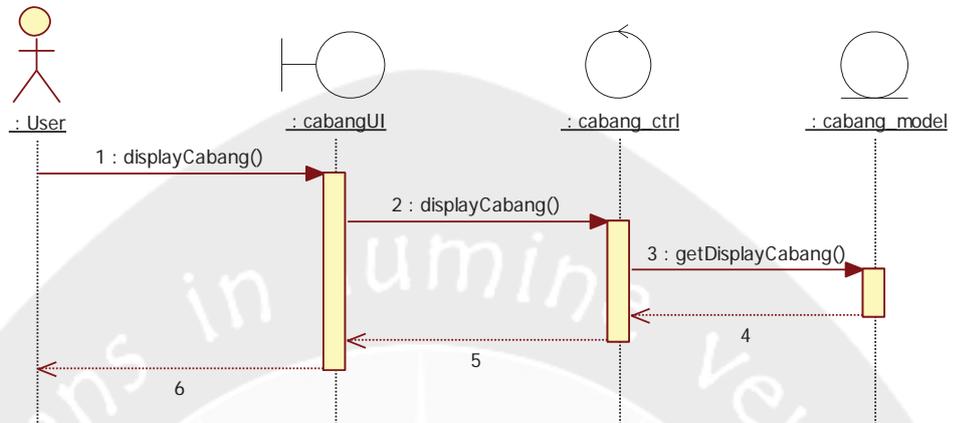
Gambar 0.62 Sequence Diagram: Grafik Laporan Penjualan

## Tampil Pegawai



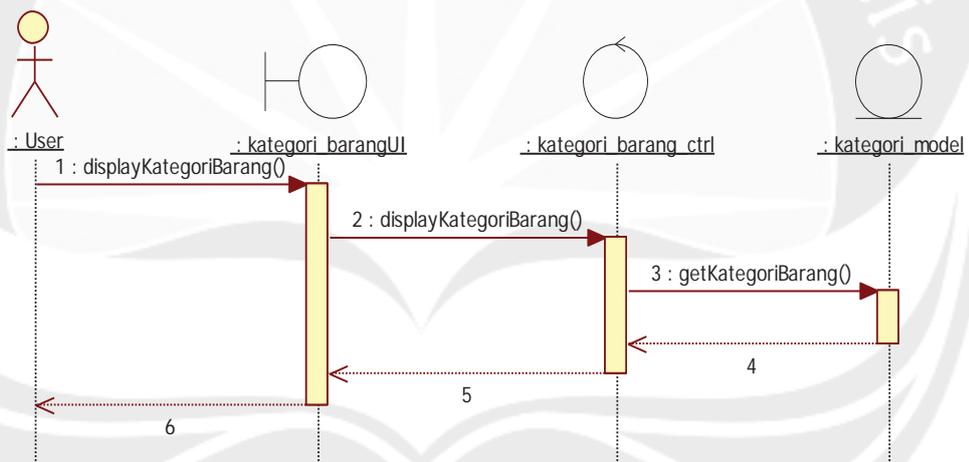
Gambar 0.63 Sequence Diagram: Tampil Pegawai

## Tampil Cabang



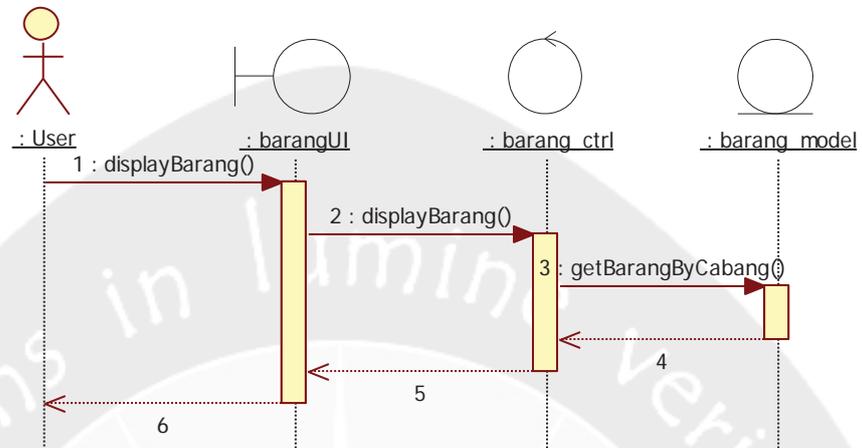
Gambar 0.64 Sequence Diagram: Tampil Cabang

## Tampil Kategori Barang



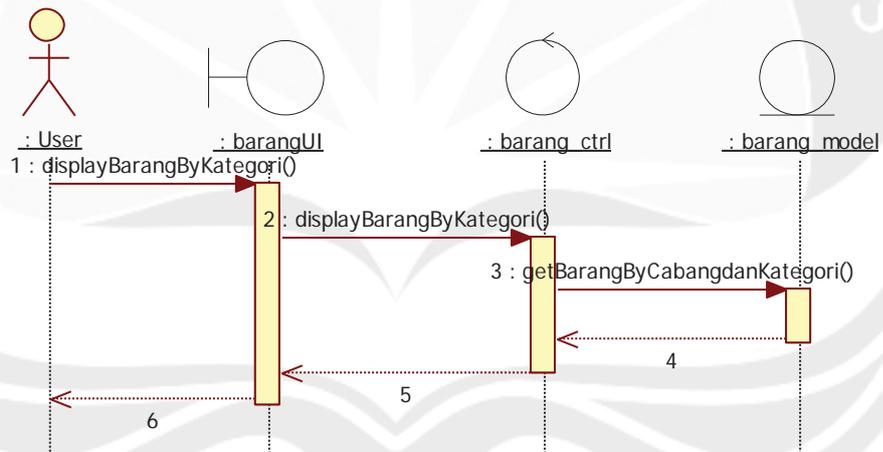
Gambar 0.65 Sequence Diagram: Tampil Kategori Barang

## Tampil Barang



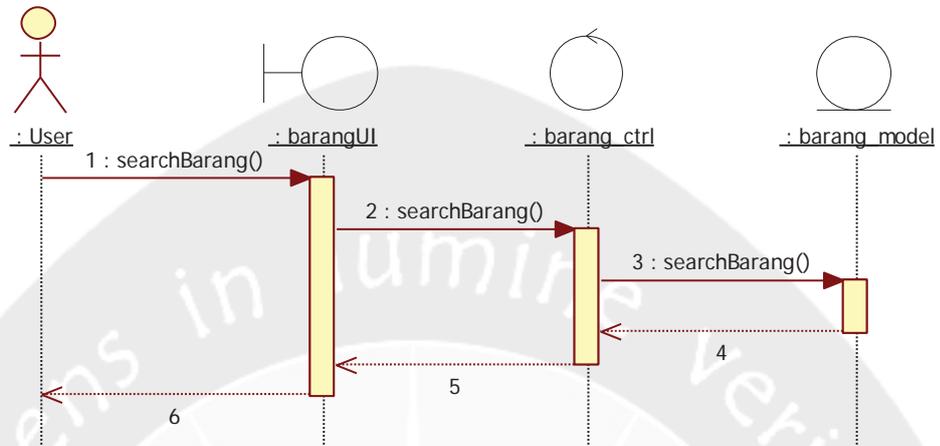
Gambar 0.66 Sequence Diagram: Tampil Barang

## Tampil Barang By Kategori



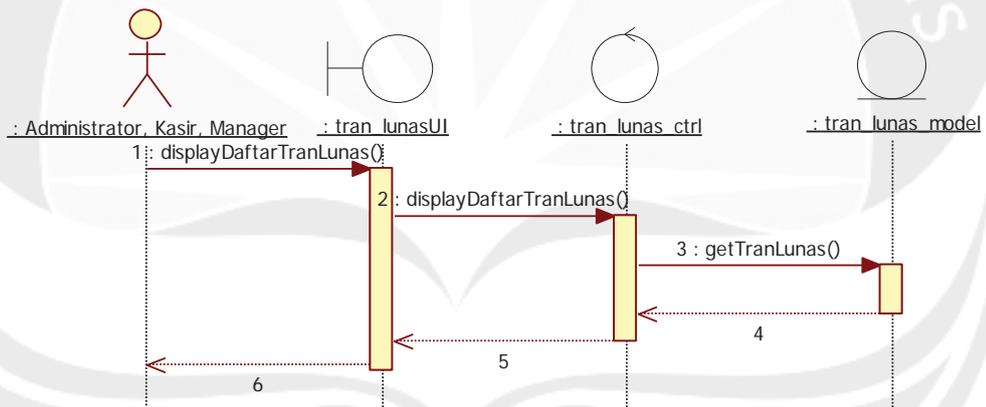
Gambar 0.67 Sequence Diagram: Tampil Barang By Kategori

## Cari Barang



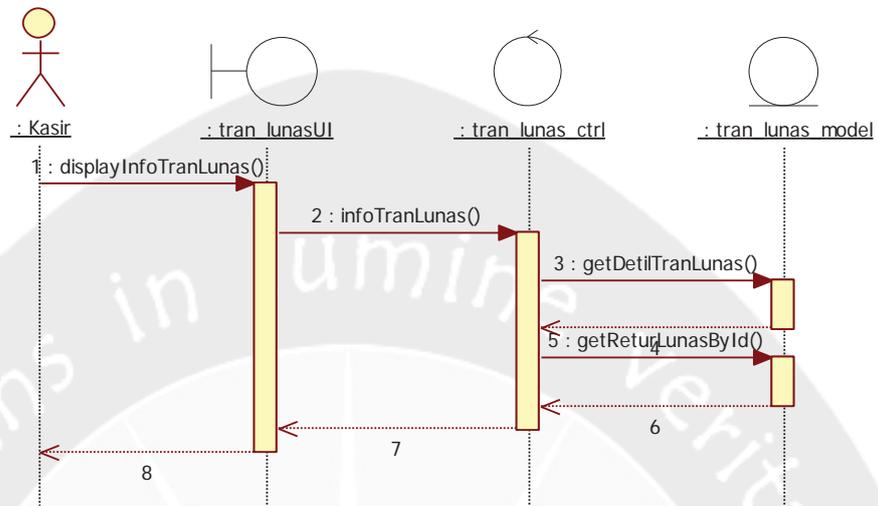
Gambar 0.68 Sequence Diagram: Cari Barang

## Tampil Transaksi Lunas



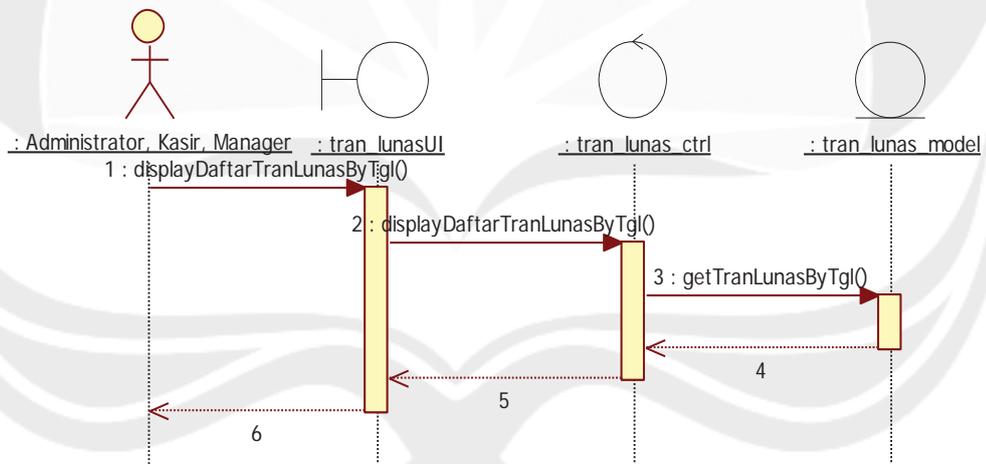
Gambar 0.69 Sequence Diagram: Tampil Transaksi Lunas

### Tampil Info Transaksi Lunas



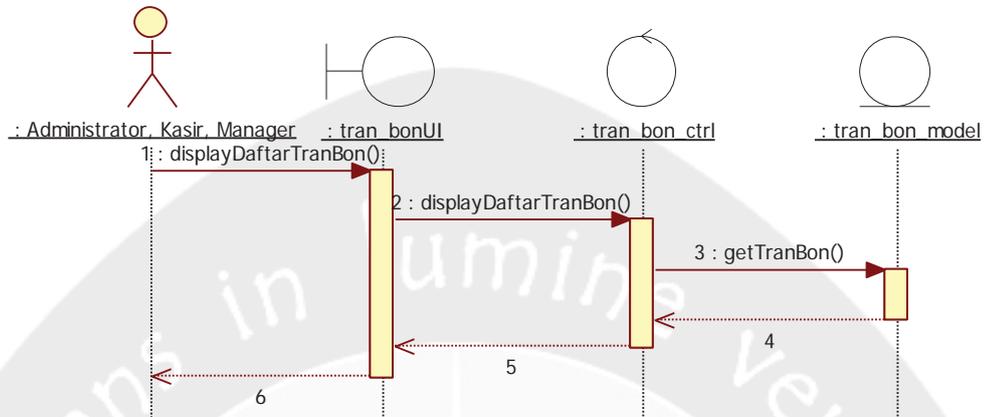
Gambar 0.70 Sequence Diagram: Tampil Info Transaksi Lunas

### Cari Transaksi Lunas



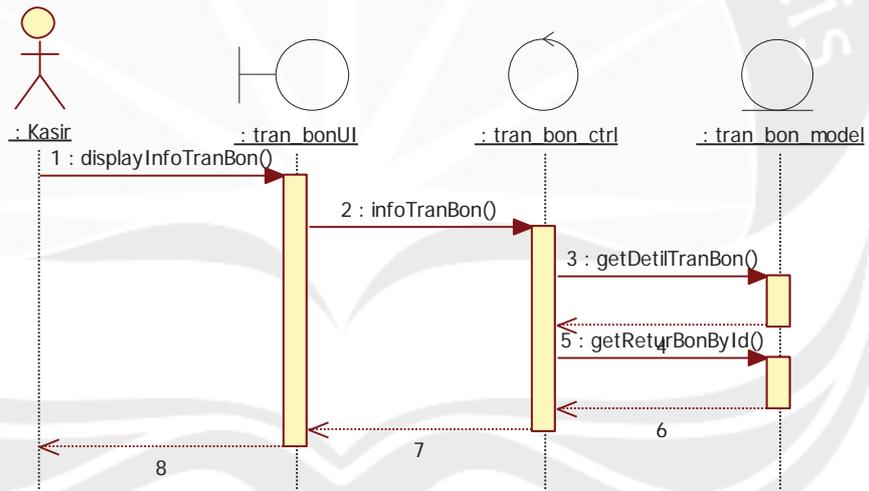
Gambar 0.71 Sequence Diagram: Cari Transaksi Lunas

## Tampil Transaksi Bon



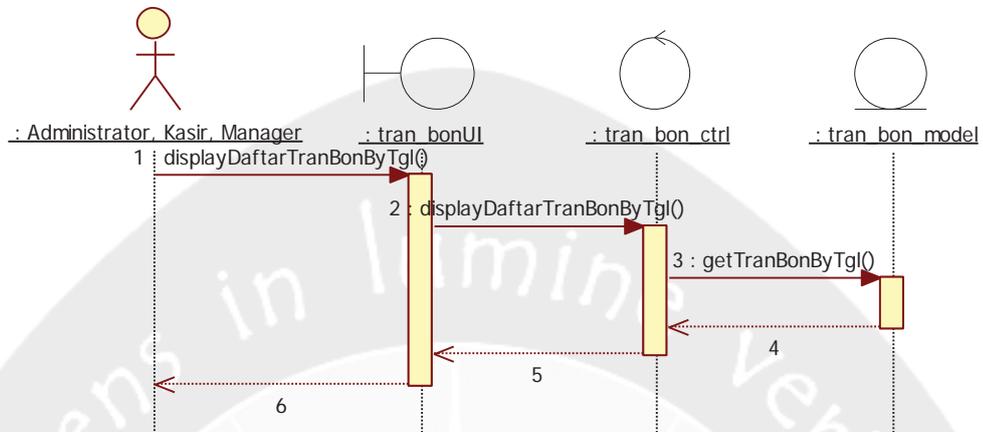
Gambar 0.72 Sequence Diagram: Tampil Transaksi Bon

## Tampil Info Transaksi Bon



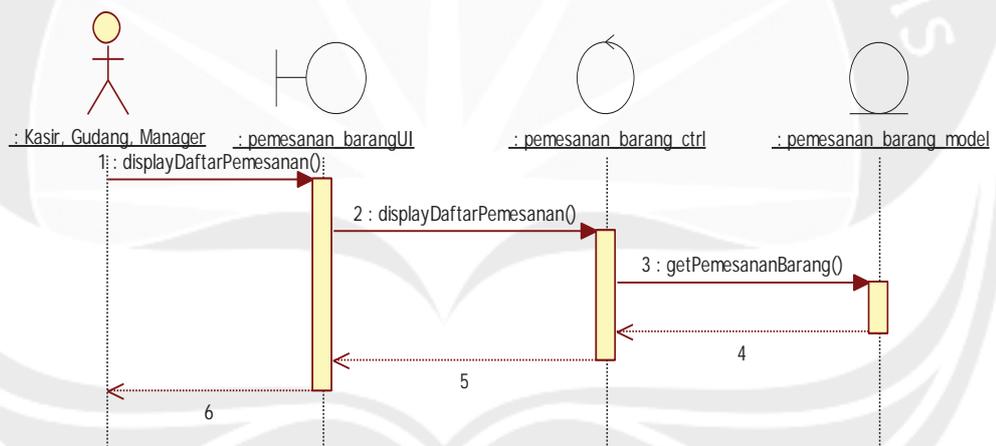
Gambar 0.73 Sequence Diagram: Tampil Info Transaksi Bon

### Cari Transaksi Bon



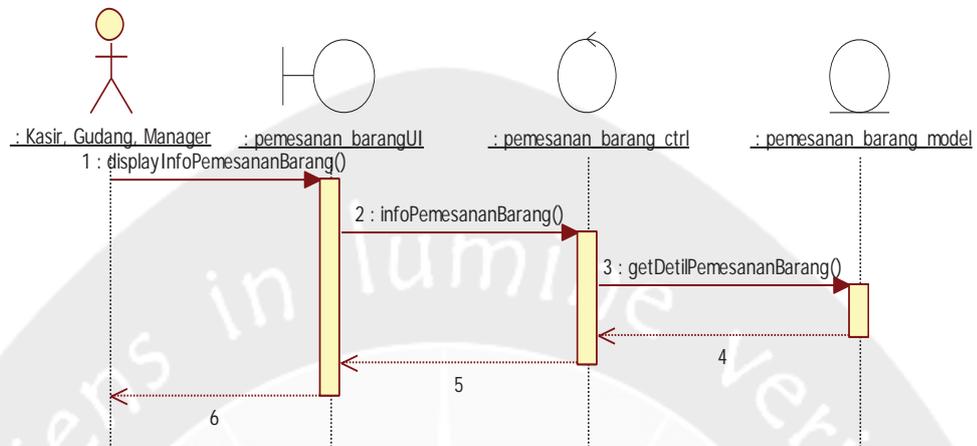
Gambar 0.74 Sequence Diagram: Cari Transaksi Bon

### Tampil Pemesanan Barang



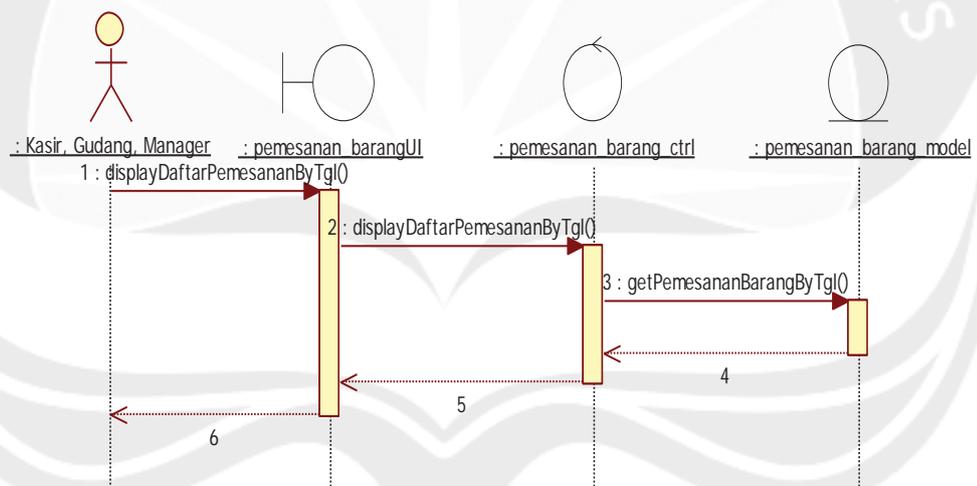
Gambar 0.75 Sequence Diagram: Tampil Pemesanan Barang

## Tampil Info Pemesanan Barang



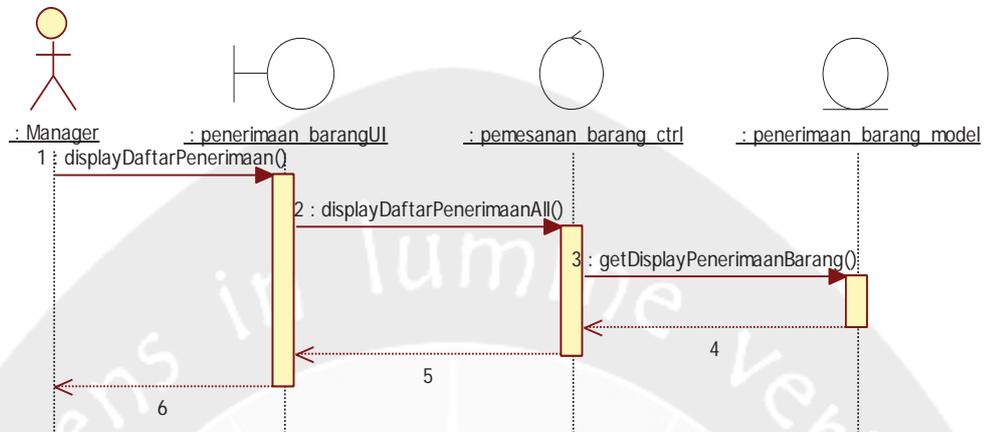
Gambar 0.76 Sequence Diagram: Tampil Info Pemesanan Barang

## Cari Pemesanan Barang



Gambar 0.77 Sequence Diagram: Cari Pemesanan Barang

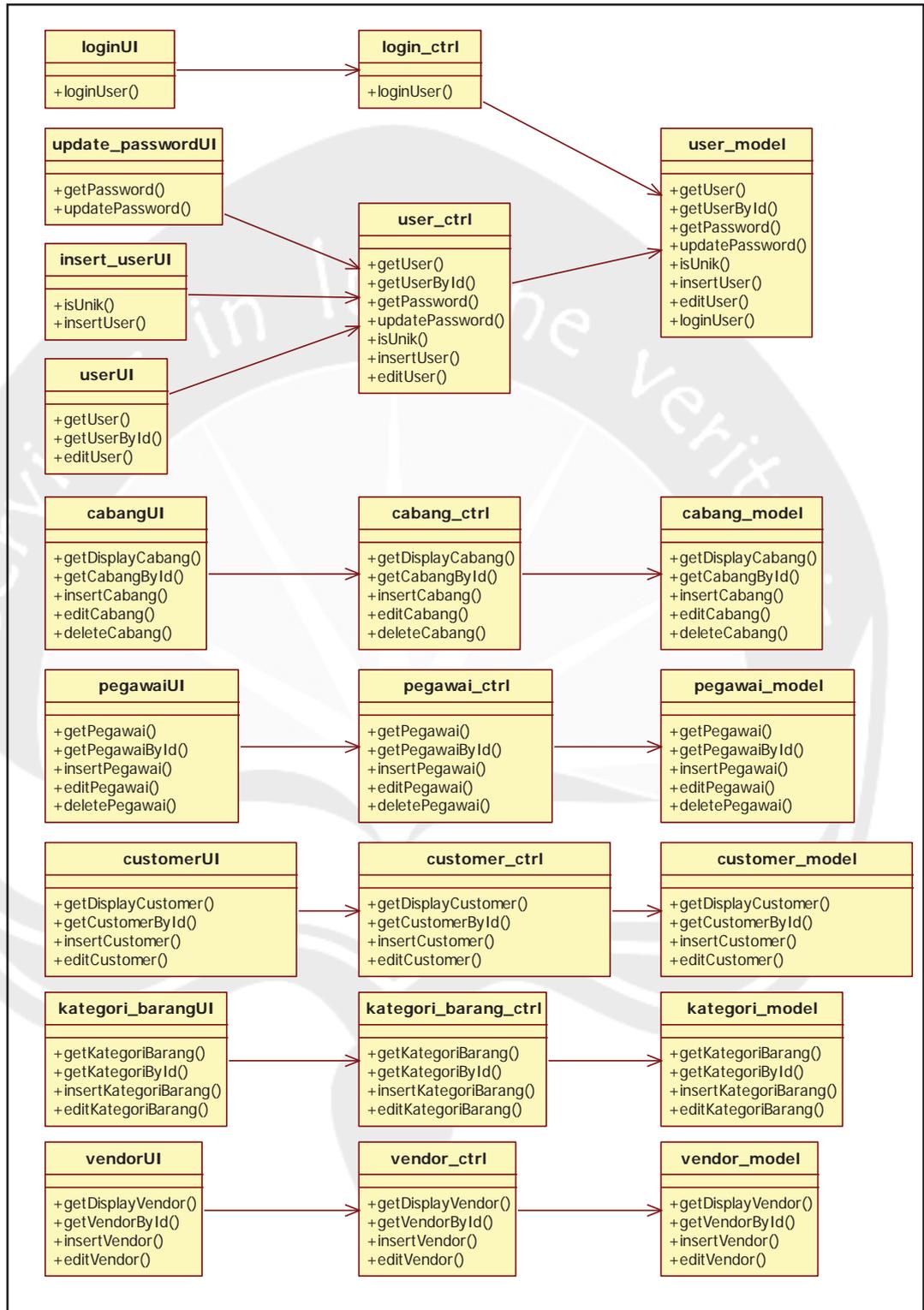
## Tampil Penerimaan Barang

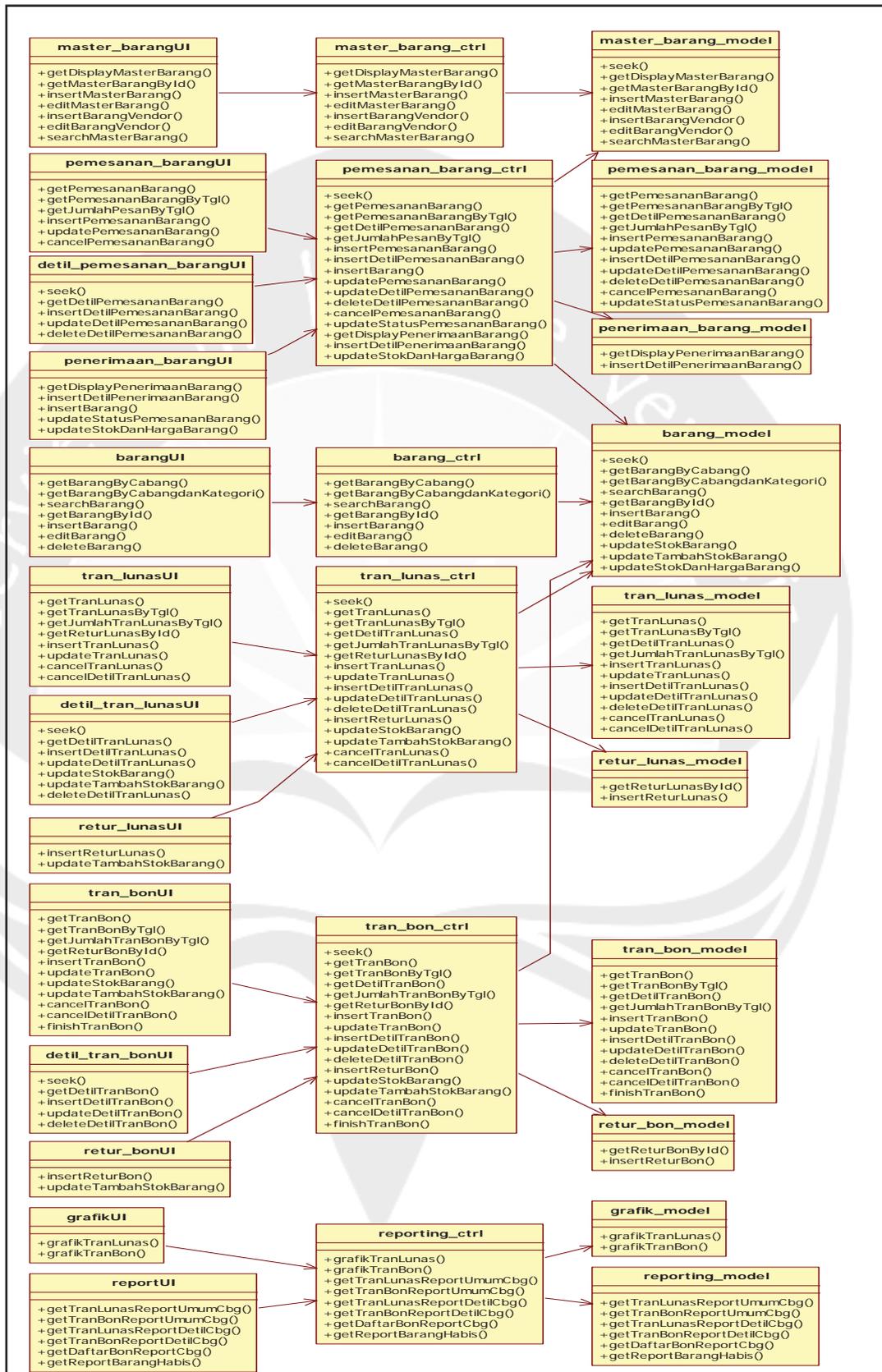


Gambar 0.78 Sequence Diagram: Tampil Penerimaan Barang

## Class Diagram







Gambar 0.79 Class Diagram

## Class Diagram Specific Descriptions

### Specific Design Class loginUI

<b>loginUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+loginUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+loginUser(username,password):bool Operasi ini digunakan untuk melakukan pengecekan apakah username dan password yang dimasukkan valid atau tidak sebelum memasuki sistem sesuai dengan role pengguna masing-masing.	

### Specific Design Class update\_passwordUI

<b>update_passwordUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+update_passwordUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+getPassword():void Operasi ini digunakan untuk mengambil password dari pengguna.	
+updatePassword(pass_lama,pass_baru,confirm_pass_baru):void Operasi ini digunakan untuk mengubah password dari pengguna.	

### Specific Design insert\_userUI

<b>insert_userUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+insert_userUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+isUnik():void Operasi ini digunakan untuk melakukan pengecekan apakah username sudah terdaftar atau belum.	

```
+insertUser(user):void
```

Operasi ini digunakan untuk menambah data user baru.

### Specific Design userUI

<b>userUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+userUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getUser():Tabel Operasi ini digunakan untuk menampilkan data user. +getUserById(id_user):void Operasi ini digunakan untuk mengambil data user yang dipergunakan pada saat akan melakukan ubah data user. +editUser(user):void Operasi ini digunakan untuk mengubah data user.</pre>	

### Specific Design Class cabangUI

<b>cabangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+cabangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDisplayCabang():Tabel Operasi ini digunakan untuk menampilkan data cabang. +getCabangById(id_cabang):void Operasi ini digunakan untuk mengambil data cabang yang dipergunakan pada saat akan melakukan ubah data cabang. +insertCabang(cabang):void Operasi ini digunakan untuk menambah data cabang baru. +editCabang(cabang):void Operasi ini digunakan untuk mengubah data cabang. +deleteCabang(id_cabang):void Operasi ini digunakan untuk menghapus data cabang.</pre>	

### Specific Design pegawaiUI

<b>pegawaiUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+pegawaiUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getPegawai():Tabel Operasi ini digunakan untuk menampilkan data pegawai. +getPegawaiById(id_pegawai):void Operasi ini digunakan untuk mengambil data pegawai yang dipergunakan pada saat akan melakukan ubah data pegawai. +insertPegawai(pegawai):void Operasi ini digunakan untuk menambah data pegawai baru. +editPegawai(pegawai):void Operasi ini digunakan untuk mengubah data pegawai. +deletePegawai(id_pegawai):void Operasi ini digunakan untuk menghapus data pegawai.</pre>	

### Specific Design customerUI

<b>customerUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+customerUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDisplayCustomer():Tabel Operasi ini digunakan untuk menampilkan data customer. +getCustomerById(id_customer):void Operasi ini digunakan untuk mengambil data customer yang dipergunakan pada saat akan melakukan ubah data customer. +insertCustomer(customer):void Operasi ini digunakan untuk menambah data customer baru. +editCustomer(customer):void Operasi ini digunakan untuk mengubah data customer.</pre>	

### Specific Design kategori\_barangUI

<b>kategori_barangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+kategori_barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getKategoriBarang():Tabel Operasi ini digunakan untuk menampilkan data kategori barang. +getKategoriById(id_kategori):void Operasi ini digunakan untuk mengambil data kategori barang yang dipergunakan pada saat akan melakukan ubah data kategori barang. +insertKategoriBarang(kategori):void Operasi ini digunakan untuk menambah data kategori barang baru. +editKategoriBarang(kategori):void Operasi ini digunakan untuk mengubah data kategori barang.</pre>	

### Specific Design Class vendorUI

<b>vendorUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+vendorUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDisplayVendor():Tabel Operasi ini digunakan untuk menampilkan data vendor. +getVendorById(id_vendor):void Operasi ini digunakan untuk mengambil data vendor yang dipergunakan pada saat akan melakukan ubah data vendor. +insertVendor(vendor):void Operasi ini digunakan untuk menambah data vendor baru. +editVendor(vendor):void Operasi ini digunakan untuk mengubah data vendor.</pre>	

### Specific Design Class master\_barangUI

<b>master_barangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> +master_barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDisplayMasterBarang():Tabel Operasi ini digunakan untuk menampilkan data master barang. +getMasterBarangById(id_master_brg):void Operasi ini digunakan untuk mengambil data master barang yang dipergunakan pada saat akan melakukan ubah data master barang. +insertMasterBarang(master_barang):void Operasi ini digunakan untuk menambah data master barang baru. +editMasterBarang(master_barang):void Operasi ini digunakan untuk mengubah data master barang. +insertBarangVendor(master_barang):void Operasi ini digunakan untuk menambah data barang vendor baru. +editBarangVendor(master_barang):void Operasi ini digunakan untuk mengubah data status harga dari barang vendor. +searchMasterBarang():void Operasi ini digunakan untuk mencari data master barang. </pre>	

### Specific Design Class pemesanan\_barangUI

<b>pemesanan_barangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> +pemesanan_barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getPemesananBarang():Tabel Operasi ini digunakan untuk menampilkan data pemesanan </pre>	

barang.

+getPemesananBarangByTgl():void  
Operasi ini digunakan untuk mencari data pemesanan barang berdasarkan tanggal.

+getJumlahPesanByTgl(id\_pesan):void  
Operasi ini digunakan untuk mendapatkan jumlah pemesanan barang berdasarkan tanggal yang dipergunakan untuk kode pemesanan barang.

+insertPemesananBarang(pemesanan\_barang):void  
Operasi ini digunakan untuk menambah data pemesanan barang baru.

+updatePemesananBarang(pemesanan\_barang):void  
Operasi ini digunakan untuk mengubah data pemesanan barang.

+cancelPemesananBarang(id\_pesan):void  
Operasi ini digunakan untuk melakukan pembatalan data pemesanan barang.

### Specific Design Class `detil_pemesanan_barangUI`

<code>detil_pemesanan_barangUI</code>	<code>&lt;&lt;boundary&gt;&gt;</code>
<pre>+detil_pemesanan_barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDetilPemesananBarang():Tabel Operasi ini digunakan untuk menampilkan data detil pemesanan barang. +seek():void Operasi ini digunakan untuk mendapatkan data barang dari master barang berdasarkan vendor tertentu. +insertDetilPemesananBarang(pemesanan_barang):void Operasi ini digunakan untuk menambah data detil pemesanan barang. +updateDetilPemesananBarang(pemesanan_barang):void Operasi ini digunakan untuk mengubah data detil pemesanan</pre>	

barang.  
+deleteDetilPemesananBarang(id\_detil\_pesan):void  
Operasi ini digunakan untuk menghapus data detil pemesanan barang.

### Specific Design Class penerimaan\_barangUI

<b>penerimaan_barangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+penerimaan_barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDisplayPenerimaanBarang():Tabel Operasi ini digunakan untuk menampilkan data penerimaan barang. +insertDetilPenerimaanBarang(penerimaan_barang):void Operasi ini digunakan untuk menambah data detil penerimaan barang baru. +insertBarang(barang):void Operasi ini digunakan untuk menambah data barang baru berdasarkan barang yang diterima. +updateStatusPemesananBarang(id_pesan):void Operasi ini digunakan untuk mengubah status pemesanan barang. +updateStokDanHargaBarang(id_barang):void Operasi ini digunakan untuk mengubah stok dan harga barang.</pre>	

### Specific Design Class barangUI

<b>barangUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+barangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getBarangByCabang():Tabel Operasi ini digunakan untuk menampilkan data barang</pre>	

berdasarkan cabang tertentu.

+getBarangByCabangdanKategori():Tabel  
 Operasi ini digunakan untuk menampilkan data barang berdasarkan cabang dan kategori barang tertentu.

+searchBarang():void  
 Operasi ini digunakan untuk mencari data barang.

+getBarangById(id\_barang):void  
 Operasi ini digunakan untuk mengambil data barang yang dipergunakan pada saat akan melakukan ubah data barang.

+insertBarang(barang):void  
 Operasi ini digunakan untuk menambah data barang baru.

+editBarang(barang):void  
 Operasi ini digunakan untuk mengubah data barang.

+deleteBarang(id\_barang):void  
 Operasi ini digunakan untuk menghapus data barang.

**Specific Design Class tran\_lunasUI**

<b>tran_lunasUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+tran_lunasUI() Default konstruktor, digunakan untuk inialisasi semua atribut dari kelas ini. +getTranLunas():Tabel Operasi ini digunakan untuk menampilkan data transaksi lunas. +getTranLunasByTgl():void Operasi ini digunakan untuk mencari data transaksi lunas berdasarkan tanggal. +getJumlahTranLunasByTgl():void Operasi ini digunakan untuk mendapatkan jumlah transaksi lunas berdasarkan tanggal yang dipergunakan untuk kode transaksi lunas. +getReturLunasById(id_tran_lunas):void Operasi ini digunakan untuk mendapatkan data retur pada</pre>	

suatu data transaksi lunas.

```
+insertTranLunas(tran_lunas):void
```

Operasi ini digunakan untuk menambah data transaksi lunas baru.

```
+updateTranLunas(tran_lunas):void
```

Operasi ini digunakan untuk mengubah data transaksi lunas.

```
+cancelTranLunas(id_tran_lunas):void
```

Operasi ini digunakan untuk melakukan pembatalan data transaksi lunas.

```
+cancelDetilTranLunas(id_detil_lunas):void
```

Operasi ini digunakan untuk melakukan pembatalan data detil transaksi lunas.

### Specific Design Class `detil_tran_lunasUI`

<code>detil_tran_lunasUI</code>	<code>&lt;&lt;boundary&gt;&gt;</code>
<pre>+detil_tran_lunasUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <pre>+getDetilTranLunas():Tabel</pre> <p>Operasi ini digunakan untuk menampilkan data detil transaksi lunas.</p> <pre>+seek():void</pre> <p>Operasi ini digunakan untuk mendapatkan data barang yang ada pada cabang tertentu.</p> <pre>+insertDetilTranLunas(tran_lunas):void</pre> <p>Operasi ini digunakan untuk menambah data detil transaksi lunas baru.</p> <pre>+updateDetilTranLunas(tran_lunas):void</pre> <p>Operasi ini digunakan untuk mengubah data detil transaksi lunas.</p> <pre>+updateStokBarang(barang):void</pre> <p>Operasi ini digunakan untuk mengubah stok barang dengan melakukan pengurangan stok barang.</p>	

```
+updateTambahStokBarang(barang):void
```

Operasi ini digunakan untuk mengubah stok barang dengan melakukan penambahan stok barang.

```
+deleteDetilTranLunas(id_detil_lunas):void
```

Operasi ini digunakan untuk menghapus data detil transaksi lunas.

### Specific Design Class retur\_lunasUI

<b>retur_lunasUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+retur_lunasUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <pre>+insertReturLunas(retur_lunas):void</pre> <p>Operasi ini digunakan untuk menambah data retur transaksi lunas baru.</p> <pre>+updateTambahStokBarang(barang):void</pre> <p>Operasi ini digunakan untuk mengubah stok barang dengan melakukan penambahan stok barang.</p>	

### Specific Design Class tran\_bonUI

<b>tran_bonUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+tran_bonUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <pre>+getTranBon():Tabel</pre> <p>Operasi ini digunakan untuk menampilkan data transaksi bon.</p> <pre>+getTranBonByTgl():void</pre> <p>Operasi ini digunakan untuk mencari data transaksi bon berdasarkan tanggal.</p> <pre>+getJumlahTranBonByTgl():void</pre> <p>Operasi ini digunakan untuk mendapatkan jumlah transaksi bon berdasarkan tanggal yang dipergunakan untuk kode transaksi</p>	

```

bon.
+getReturBonById(id_tran_bon):void
Operasi ini digunakan untuk mendapatkan data retur pada
suatu data transaksi bon.
+insertTranBon(transaksi_bon):void
Operasi ini digunakan untuk menambah data transaksi bon
baru.
+updateTranBon(transaksi_bon):void
Operasi ini digunakan untuk mengubah data transaksi bon.
+cancelTranBon(id_tran_bon):void
Operasi ini digunakan untuk melakukan pembatalan data
transaksi bon.
+cancelDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk melakukan pembatalan data detil
transaksi bon.
+finishTranBon(transaksi_bon):void
Operasi ini digunakan untuk melakukan pelunasan terhadap
suatu data transaksi bon.

```

### Specific Design Class detil\_tran\_bonUI

<b>detil_tran_bonUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> +detil_tran_bonUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDetilTranBon():Tabel Operasi ini digunakan untuk menampilkan data detil transaksi bon. +seek():void Operasi ini digunakan untuk mendapatkan data barang yang ada pada cabang tertentu. +insertDetilTranBon(transaksi_bon):void Operasi ini digunakan untuk menambah data detil transaksi bon baru. </pre>	

```
+updateDetilTranBon(transaksi_bon):void
Operasi ini digunakan untuk mengubah data detil transaksi bon.
+updateStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan melakukan pengurangan stok barang.
+updateTambahStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan melakukan penambahan stok barang.
+deleteDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk menghapus data detil transaksi bon.
```

#### Specific Design retur\_bonUI

<b>retur_bonUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+retur_bonUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +insertReturBon(retur_bon):void Operasi ini digunakan untuk menambah data retur transaksi bon baru. +updateTambahStokBarang(barang):void Operasi ini digunakan untuk mengubah stok barang dengan melakukan penambahan stok barang.</pre>	

#### Specific Design Class grafikUI

<b>grafikUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+grafikUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +grafikTranLunas():void Operasi ini digunakan untuk menampilkan grafik untuk data</pre>	

```

transaksi lunas.
+grafikTranBon():void
Operasi ini digunakan untuk menampilkan grafik untuk data
transaksi bon.

```

### Specific Design reportUI

<b>reportUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre> + reportUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getTranLunasReportUmumCbg():void Operasi ini digunakan untuk menampilkan report umum untuk data transaksi lunas. +getTranBonReportUmumCbg():void Operasi ini digunakan untuk menampilkan report umum untuk data transaksi bon. +getTranLunasReportDetilCbg():void Operasi ini digunakan untuk menampilkan report detil untuk data transaksi lunas. +getTranBonReportDetilCbg():void Operasi ini digunakan untuk menampilkan report detil untuk data transaksi bon. +getDaftarBonReportCbg():void Operasi ini digunakan untuk menampilkan report daftar bon. +getReportBarangHabis():void Operasi ini digunakan untuk menampilkan report daftar barang habis. </pre>	

### Specific Design Class login\_ctrl

<b>login_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre> +loginUser(username,password):bool Operasi ini digunakan untuk melakukan pengecekan apakah </pre>	

username dan password yang dimasukkan valid atau tidak sebelum memasuki sistem sesuai dengan role pengguna masing-masing.

### Specific Design Class user\_ctrl

<b>user_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getUser():Tabel Operasi ini digunakan untuk menampilkan data user. +getUserById():void Operasi ini digunakan untuk mengambil data user yang dipergunakan pada saat akan melakukan ubah data user. +getPassword():void Operasi ini digunakan untuk mengambil password dari pengguna. +updatePassword(pass_lama,pass_baru,confirm_pass_baru):void Operasi ini digunakan untuk mengubah password dari pengguna. +isUnik() Operasi ini digunakan untuk melakukan pengecekan apakah username sudah terdaftar atau belum. +insertUser(user):void Operasi ini digunakan untuk menambah data user baru. +editUser(user):void Operasi ini digunakan untuk mengubah data user.</pre>	

### Specific Design cabang\_ctrl

<b>cabang_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getDisplayCabang():Tabel Operasi ini digunakan untuk menampilkan data cabang. +getCabangById(id_cabang):void Operasi ini digunakan untuk mengambil data cabang yang dipergunakan pada saat akan melakukan ubah data cabang. +insertCabang(cabang):void</pre>	

Operasi ini digunakan untuk menambah data cabang baru.  
+editCabang(cabang):void  
Operasi ini digunakan untuk mengubah data cabang.  
+deleteCabang(id\_cabang):void  
Operasi ini digunakan untuk menghapus data cabang.

**Specific Design Class pegawai\_ctrl**

<b>pegawai_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getPegawai():Tabel Operasi ini digunakan untuk menampilkan data pegawai. +getPegawaiById(id_pegawai):void Operasi ini digunakan untuk mengambil data pegawai yang dipergunakan pada saat akan melakukan ubah data pegawai. +insertPegawai(pegawai):void Operasi ini digunakan untuk menambah data pegawai baru. +editPegawai(pegawai):void Operasi ini digunakan untuk mengubah data pegawai. +deletePegawai(id_pegawai):void Operasi ini digunakan untuk menghapus data pegawai.</pre>	

**Specific Design Class customer\_ctrl**

<b>customer_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getDisplayCustomer():Tabel Operasi ini digunakan untuk menampilkan data customer. +getCustomerById(id_customer):void Operasi ini digunakan untuk mengambil data customer yang dipergunakan pada saat akan melakukan ubah data customer. +insertCustomer(customer):void Operasi ini digunakan untuk menambah data customer baru. +editCustomer(customer):void Operasi ini digunakan untuk mengubah data customer.</pre>	

### Specific Design Class kategori\_barang\_ctrl

kategori_barang_ctrl	<<control>>
+getKategoriBarang():Tabel Operasi ini digunakan untuk menampilkan data kategori barang.	
+getKategoriById(id_kategori):void Operasi ini digunakan untuk mengambil data kategori barang yang dipergunakan pada saat akan melakukan ubah data kategori barang.	
+insertKategoriBarang(kategori):void Operasi ini digunakan untuk menambah data kategori barang baru.	
+editKategoriBarang(kategori):void Operasi ini digunakan untuk mengubah data kategori barang.	

### Specific Design Class vendor\_ctrl

vendor_ctrl	<<control>>
+getDisplayVendor():Tabel Operasi ini digunakan untuk menampilkan data vendor.	
+getVendorById(id_vendor):void Operasi ini digunakan untuk mengambil data vendor yang dipergunakan pada saat akan melakukan ubah data vendor.	
+insertVendor(vendor):void Operasi ini digunakan untuk menambah data vendor baru.	
+editVendor(vendor):void Operasi ini digunakan untuk mengubah data vendor.	

### Specific Design Class master\_barang\_ctrl

master_barang_ctrl	<<control>>
+getDisplayMasterBarang():Tabel Operasi ini digunakan untuk menampilkan data master barang.	

+getMasterBarangById(id\_master\_brg):void

Operasi ini digunakan untuk mengambil data master barang yang dipergunakan pada saat akan melakukan ubah data master barang.

+insertMasterBarang(master\_barang):void

Operasi ini digunakan untuk menambah data master barang baru.

+editMasterBarang(master\_barang):void

Operasi ini digunakan untuk mengubah data master barang.

+insertBarangVendor(master\_barang):void

Operasi ini digunakan untuk menambah data barang vendor baru.

+editBarangVendor(master\_barang):void

Operasi ini digunakan untuk mengubah data status harga dari barang vendor.

+searchMasterBarang():void

Operasi ini digunakan untuk mencari data master barang.

### Specific Design Class pemesanan\_barang\_ctrl

pemesanan_barang_ctrl	<<control>>
-----------------------	-------------

+seek():void

Operasi ini digunakan untuk mendapatkan data barang dari master barang berdasarkan vendor tertentu.

+getPemesananBarang():Tabel

Operasi ini digunakan untuk menampilkan data pemesanan barang.

+getPemesananBarangByTgl():void

Operasi ini digunakan untuk mencari data pemesanan barang berdasarkan tanggal.

+getDetilPemesananBarang():Tabel

Operasi ini digunakan untuk menampilkan data detil pemesanan barang.

+getJumlahPesanByTgl(id\_pesan):void

Operasi ini digunakan untuk mendapatkan jumlah pemesanan barang berdasarkan tanggal yang dipergunakan untuk kode pemesanan barang.

+insertPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk menambah data pemesanan barang baru.

+updatePemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk mengubah data pemesanan barang.

+insertDetilPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk menambah data detil pemesanan barang.

+insertBarang(barang):void

Operasi ini digunakan untuk menambah data barang baru berdasarkan barang yang diterima.

+updateDetilPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk mengubah data detil pemesanan barang.

+deleteDetilPemesananBarang(id\_detil\_pesan):void

Operasi ini digunakan untuk menghapus data detil pemesanan barang.

+cancelPemesananBarang(id\_pesan):void

Operasi ini digunakan untuk melakukan pembatalan data pemesanan barang.

+updateStatusPemesananBarang(id\_pesan):void

Operasi ini digunakan untuk mengubah status pemesanan barang.

+getDisplayPenerimaanBarang():Tabel

Operasi ini digunakan untuk menampilkan data penerimaan barang.

+insertDetilPenerimaanBarang(penerimaan\_barang):void

Operasi ini digunakan untuk menambah data detil penerimaan barang baru.

+updateStokDanHargaBarang(id\_barang):void

Operasi ini digunakan untuk mengubah stok dan harga barang.

Program Studi Teknik Informatika	DPPL – SISP	76/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### Specific Design Class barang\_ctrl

<b>barang_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getBarangByCabang():Tabel Operasi ini digunakan untuk menampilkan data barang berdasarkan cabang tertentu. +getBarangByCabangdanKategori():Tabel Operasi ini digunakan untuk menampilkan data barang berdasarkan cabang dan kategori barang tertentu. +searchBarang():void Operasi ini digunakan untuk mencari data barang. +getBarangById(id_barang):void Operasi ini digunakan untuk mengambil data barang yang dipergunakan pada saat akan melakukan ubah data barang. +insertBarang(barang):void Operasi ini digunakan untuk menambah data barang baru. +editBarang(barang):void Operasi ini digunakan untuk mengubah data barang. +deleteBarang(id_barang):void Operasi ini digunakan untuk menghapus data barang.</pre>	

### Specific Design Class tran\_lunas\_ctrl

<b>tran_lunas_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+seek():void Operasi ini digunakan untuk mendapatkan data barang yang ada pada cabang tertentu. +getTranLunas():Tabel Operasi ini digunakan untuk menampilkan data transaksi lunas. +getTranLunasByTgl():void Operasi ini digunakan untuk mencari data transaksi lunas berdasarkan tanggal. +getDetilTranLunas():Tabel</pre>	

Operasi ini digunakan untuk menampilkan data detail transaksi lunas.

+getJumlahTranLunasByTgl():void

Operasi ini digunakan untuk mendapatkan jumlah transaksi lunas berdasarkan tanggal yang dipergunakan untuk kode transaksi lunas.

+getReturLunasById(id\_tran\_lunas):void

Operasi ini digunakan untuk mendapatkan data retur pada suatu data transaksi lunas.

+insertTranLunas(tran\_lunas):void

Operasi ini digunakan untuk menambah data transaksi lunas baru.

+updateTranLunas(tran\_lunas):void

Operasi ini digunakan untuk mengubah data transaksi lunas.

+insertDetilTranLunas(tran\_lunas):void

Operasi ini digunakan untuk menambah data detail transaksi lunas baru.

+updateDetilTranLunas(tran\_lunas):void

Operasi ini digunakan untuk mengubah data detail transaksi lunas.

+deleteDetilTranLunas(id\_detil\_lunas):void

Operasi ini digunakan untuk menghapus data detail transaksi lunas.

+insertReturLunas(retur\_lunas):void

Operasi ini digunakan untuk menambah data retur transaksi lunas baru.

+updateStokBarang(barang):void

Operasi ini digunakan untuk mengubah stok barang dengan melakukan pengurangan stok barang.

+updateTambahStokBarang(barang):void

Operasi ini digunakan untuk mengubah stok barang dengan melakukan penambahan stok barang.

+cancelTranLunas(id\_tran\_lunas):void

Operasi ini digunakan untuk melakukan pembatalan data

transaksi lunas.  
+cancelDetilTranLunas(id\_detil\_lunas):void  
Operasi ini digunakan untuk melakukan pembatalan data detil transaksi lunas.

**Specific Design Class tran\_bon\_ctrl**

<b>tran_bon_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+seek():void Operasi ini digunakan untuk mendapatkan data barang yang ada pada cabang tertentu. +getTranBon():Tabel Operasi ini digunakan untuk menampilkan data transaksi bon. +getTranBonByTgl():void Operasi ini digunakan untuk mencari data transaksi bon berdasarkan tanggal. +getDetilTranBon():Tabel Operasi ini digunakan untuk menampilkan data detil transaksi bon. +getJumlahTranBonByTgl():void Operasi ini digunakan untuk mendapatkan jumlah transaksi bon berdasarkan tanggal yang dipergunakan untuk kode transaksi bon. +getReturBonById(id_tran_bon):void Operasi ini digunakan untuk mendapatkan data retur pada suatu data transaksi bon. +insertTranBon(transaksi_bon):void Operasi ini digunakan untuk menambah data transaksi bon baru. +updateTranBon(transaksi_bon):void Operasi ini digunakan untuk mengubah data transaksi bon. +insertDetilTranBon(transaksi_bon):void Operasi ini digunakan untuk menambah data detil transaksi bon baru.</pre>	

```

+updateDetilTranBon(transaksi_bon):void
Operasi ini digunakan untuk mengubah data detil transaksi
bon.
+deleteDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk menghapus data detil transaksi
bon.
+insertReturBon(retur_bon):void
Operasi ini digunakan untuk menambah data retur transaksi
bon baru.
+updateStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan
melakukan pengurangan stok barang.
+updateTambahStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan
melakukan penambahan stok barang.
+cancelTranBon(id_tran_bon):void
Operasi ini digunakan untuk melakukan pembatalan data
transaksi bon.
+cancelDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk melakukan pembatalan data detil
transaksi bon.
+finishTranBon(transaksi_bon):void
Operasi ini digunakan untuk melakukan pelunasan terhadap
suatu data transaksi bon.

```

### Specific Design Class reporting\_ctrl

<b>reporting_ctrl</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre> +grafikTranLunas():Grafik Operasi ini digunakan untuk menampilkan grafik untuk data transaksi lunas. +grafikTranBon():Grafik Operasi ini digunakan untuk menampilkan grafik untuk data transaksi bon. </pre>	

```

+getTranLunasReportUmumCbg():void
Operasi ini digunakan untuk menampilkan report umum untuk
data transaksi lunas.
+getTranBonReportUmumCbg():void
Operasi ini digunakan untuk menampilkan report umum untuk
data transaksi bon.
+getTranLunasReportDetilCbg():void
Operasi ini digunakan untuk menampilkan report detil untuk
data transaksi lunas.
+getTranBonReportDetilCbg():void
Operasi ini digunakan untuk menampilkan report detil untuk
data transaksi bon.
+getDaftarBonReportCbg():void
Operasi ini digunakan untuk menampilkan report daftar bon.
+getReportBarangHabis():void
Operasi ini digunakan untuk menampilkan report daftar barang
habis.

```

### Specific Design Class user\_model

user_model	<<entity>>
<pre> +ID_USER: int ID dari pengguna yang bersifat unik +USERNAME: string Username dari pengguna yang digunakan untuk login +PASSWORD: string Password dari pengguna yang digunakan untuk login +NAMA: string Nama dari pengguna +EMAIL: string Email dari pengguna +ALAMAT: string Alamat dari pengguna +JENIS_KELAMIN: string Jenis kelamin dari pengguna </pre>	

<p>+FOTO: string Letak dari foto pengguna yang diupload</p> <p>+ID_ROLE: int ID Role dari pengguna</p> <p>+ID_CABANG: int ID Cabang dari pengguna</p>
<p>+getUser():Tabel Operasi ini digunakan untuk menampilkan data user.</p> <p>+getUserById():void Operasi ini digunakan untuk mengambil data user yang dipergunakan pada saat akan melakukan ubah data user.</p> <p>+getPassword():void Operasi ini digunakan untuk mengambil password dari pengguna.</p> <p>+updatePassword(pass_lama,pass_baru,confirm_pass_baru):void Operasi ini digunakan untuk mengubah password dari pengguna.</p> <p>+isUnik() Operasi ini digunakan untuk melakukan pengecekan apakah username sudah terdaftar atau belum.</p> <p>+insertUser(user):void Operasi ini digunakan untuk menambah data user baru.</p> <p>+editUser(user):void Operasi ini digunakan untuk mengubah data user.</p> <p>+loginUser(username,password):bool Operasi ini digunakan untuk melakukan pengecekan apakah username dan password yang dimasukkan valid atau tidak sebelum memasuki sistem sesuai dengan role pengguna masing-masing.</p>

**Specific Design cabang\_model**

<b>cabang_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<p>+ID_CABANG: int ID dari cabang yang bersifat unik</p> <p>+KODE_CABANG: string</p>	

Kode dari cabang

+LOGO\_CABANG: string

Letak dari logo cabang yang diupload

+NAMA\_CABANG: string

Nama dari cabang

+ALAMAT\_CABANG: string

Alamat dari cabang

+NOTELEP\_CABANG: string

Nomor telepon dari cabang

+getDisplayCabang():Tabel

Operasi ini digunakan untuk menampilkan data cabang.

+getCabangById(id\_cabang):void

Operasi ini digunakan untuk mengambil data cabang yang dipergunakan pada saat akan melakukan ubah data cabang.

+insertCabang(cabang):void

Operasi ini digunakan untuk menambah data cabang baru.

+editCabang(cabang):void

Operasi ini digunakan untuk mengubah data cabang.

+deleteCabang(id\_cabang):void

Operasi ini digunakan untuk menghapus data cabang.

### Specific Design pegawai\_model

<b>pegawai_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<p>+ID_PEGAWAI: int ID dari pegawai yang bersifat unik</p> <p>+NAMA_PEGAWAI: string Nama dari pegawai</p> <p>+ALAMAT_PEGAWAI: string Alamat dari pegawai</p> <p>+JENIS_KELAMIN: string Jenis kelamin dari pegawai</p> <p>+FOTO_PEGAWAI: string Letak dari foto pegawai yang diupload</p> <p>+TUGAS: string</p>	

Tugas dari pegawai +ID_CABANG: int ID Cabang dari pegawai
+getPegawai():Tabel Operasi ini digunakan untuk menampilkan data pegawai. +getPegawaiById(id_pegawai):void Operasi ini digunakan untuk mengambil data pegawai yang dipergunakan pada saat akan melakukan ubah data pegawai. +insertPegawai(pegawai):void Operasi ini digunakan untuk menambah data pegawai baru. +editPegawai(pegawai):void Operasi ini digunakan untuk mengubah data pegawai. +deletePegawai(id_pegawai):void Operasi ini digunakan untuk menghapus data pegawai.

**Specific Design Class customer\_model**

<b>customer_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
+ID_CUSTOMER: int ID dari customer yang bersifat unik +NAMA_CUSTOMER: string Nama dari customer +ALAMAT_CUSTOMER: string Alamat dari customer +NOTELEP_CUSTOMER: string Nomor telepon dari customer +ID_CABANG: int ID Cabang dari customer	
+getDisplayCustomer():Tabel Operasi ini digunakan untuk menampilkan data customer. +getCustomerById(id_customer):void Operasi ini digunakan untuk mengambil data customer yang dipergunakan pada saat akan melakukan ubah data customer. +insertCustomer(customer):void Operasi ini digunakan untuk menambah data customer baru.	

```
+editCustomer(customer):void
```

Operasi ini digunakan untuk mengubah data customer.

### Specific Design Class kategori\_model

<b>kategori_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_KATEGORI: int ID dari kategori barang yang bersifat unik +KODE_KATEGORI: string Kode dari kategori barang +NAMA_KATEGORI: string Nama dari kategori barang</pre>	
<pre>+getKategoriBarang():Tabel Operasi ini digunakan untuk menampilkan data kategori barang. +getKategoriById(id_kategori):void Operasi ini digunakan untuk mengambil data kategori barang yang dipergunakan pada saat akan melakukan ubah data kategori barang. +insertKategoriBarang(kategori):void Operasi ini digunakan untuk menambah data kategori barang baru. +editKategoriBarang(kategori):void Operasi ini digunakan untuk mengubah data kategori barang.</pre>	

### Specific Design Class vendor\_model

<b>vendor_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_VENDOR: int ID dari vendor yang bersifat unik +NAMA_VENDOR: string Nama dari vendor +ALAMAT_VENDOR: string Alamat dari vendor +NOTELEP_VENDOR: string Nomor telepon dari vendor</pre>	

```
+getDisplayVendor():Tabel
```

Operasi ini digunakan untuk menampilkan data vendor.

```
+getVendorById(id_vendor):void
```

Operasi ini digunakan untuk mengambil data vendor yang dipergunakan pada saat akan melakukan ubah data vendor.

```
+insertVendor(vendor):void
```

Operasi ini digunakan untuk menambah data vendor baru.

```
+editVendor(vendor):void
```

Operasi ini digunakan untuk mengubah data vendor.

### Specific Design Class master\_barang\_model

<b>master_barang_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
----------------------------	-------------------------------

```
+ID_MASTER_BRG: int
```

ID dari master barang yang bersifat unik

```
+NAMA_MASTER_BRG: string
```

Nama dari master barang

```
+MERK_MASTER_BRG: string
```

Merk dari master barang

```
+HARGA_BARANG_VENDOR: float
```

Harga dari master barang

```
+STATUS: int
```

Status dari harga master barang berupa angka 0 atau 1.

```
+getDisplayMasterBarang():Tabel
```

Operasi ini digunakan untuk menampilkan data master barang.

```
+getMasterBarangById(id_master_brg):void
```

Operasi ini digunakan untuk mengambil data master barang yang dipergunakan pada saat akan melakukan ubah data master barang.

```
+insertMasterBarang(master_barang):void
```

Operasi ini digunakan untuk menambah data master barang baru.

```
+editMasterBarang(master_barang):void
```

Operasi ini digunakan untuk mengubah data master barang.

```
+insertBarangVendor(master_barang):void
```

Operasi ini digunakan untuk menambah data barang vendor baru.

+editBarangVendor(master\_barang):void

Operasi ini digunakan untuk mengubah data status harga dari barang vendor.

+searchMasterBarang():void

Operasi ini digunakan untuk mencari data master barang.

### Specific Design Class pemesanan\_barang\_model

pemesanan_barang_model	<<entity>>
<p>+ID_PESAN: int ID dari pemesanan barang yang bersifat unik</p> <p>+KODE_PESAN: string Kode dari pemesanan barang</p> <p>+TANGGAL_PESAN: date Tanggal dari pemesanan barang</p> <p>+TOTAL_HARGA: float Total dari pemesanan barang</p> <p>+STATUS_PESAN: string Status dari pemesanan barang</p> <p>+ID_CABANG: int ID Cabang dari pemesanan barang</p> <p>+ID_USER: int ID User dari pemesanan barang</p> <p>+ID_VENDOR: int ID Vendor dari pemesanan barang</p> <p>+ID_DETIL_PESAN: int ID dari detail pemesanan barang yang bersifat unik</p> <p>+KUANTITAS: int Kuantitas dari detail pemesanan barang</p> <p>+HARGA_MASTER_BRG: float Harga master barang dari detail pemesanan barang</p> <p>+ID_MASTER_BRG: int ID Master Barang dari detail pemesanan barang</p>	

+ID\_VENDOR: int

ID Vendor dari detil pemesanan barang

+ID\_PESAN: int

ID Pemesanan Barang dari detil pemesanan barang

+getPemesananBarang():Tabel

Operasi ini digunakan untuk menampilkan data pemesanan barang.

+getPemesananBarangByTgl():void

Operasi ini digunakan untuk mencari data pemesanan barang berdasarkan tanggal.

+getDetilPemesananBarang():Tabel

Operasi ini digunakan untuk menampilkan data detil pemesanan barang.

+getJumlahPesanByTgl(id\_pesan):void

Operasi ini digunakan untuk mendapatkan jumlah pemesanan barang berdasarkan tanggal yang dipergunakan untuk kode pemesanan barang.

+insertPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk menambah data pemesanan barang baru.

+updatePemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk mengubah data pemesanan barang.

+insertDetilPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk menambah data detil pemesanan barang.

+insertBarang(barang):void

Operasi ini digunakan untuk menambah data barang baru berdasarkan barang yang diterima.

+updateDetilPemesananBarang(pemesanan\_barang):void

Operasi ini digunakan untuk mengubah data detil pemesanan barang.

+deleteDetilPemesananBarang(id\_detil\_pesan):void

Operasi ini digunakan untuk menghapus data detil pemesanan barang.

```
+cancelPemesananBarang(id_pesan):void
Operasi ini digunakan untuk melakukan pembatalan data
pemesanan barang.
+updateStatusPemesananBarang(id_pesan):void
Operasi ini digunakan untuk mengubah status pemesanan
barang.
```

### Specific Design Class penerimaan\_barang\_model

<b>penerimaan_barang_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_PENERIMAAN: int ID dari penerimaan yang bersifat unik +TANGGAL_TERIMA: date Tanggal terima dari penerimaan barang +KUANTITAS_TERIMA: int Kuantitas terima dari penerimaan barang +STATUS: string Status terima dari penerimaan barang +ID_DETIL_PESAN: int ID Detil Pesan dari penerimaan barang +ID_USER: int ID User dari penerimaan barang</pre>	
<pre>+getDisplayPenerimaanBarang():Tabel Operasi ini digunakan untuk menampilkan data penerimaan barang. +insertDetilPenerimaanBarang(penerimaan_barang):void Operasi ini digunakan untuk menambah data detil penerimaan barang baru.</pre>	

### Specific Design Class barang\_model

<b>barang_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_BARANG: int ID dari barang yang bersifat unik +NAMA_BARANG: string Nama dari barang</pre>	

+HARGA\_BARANG: float  
 Harga dari barang

+STOK\_BARANG: int  
 Stok dari barang

+FOTO\_BARANG: string  
 Lokasi foto dari barang

+HARGA\_BELI: float  
 Harga beli dari barang

+MERK\_BARANG: string  
 Merk dari barang

+BARCODE: string  
 Barcode dari barang

+ID\_KATEGORI: int  
 ID Kategori dari barang

+ID\_CABANG: int  
 ID Cabang dari barang

+seek():void  
 Operasi ini digunakan untuk mendapatkan data barang yang ada pada cabang tertentu.

+getBarangByCabang():Tabel  
 Operasi ini digunakan untuk menampilkan data barang berdasarkan cabang tertentu.

+getBarangByCabangdanKategori():Tabel  
 Operasi ini digunakan untuk menampilkan data barang berdasarkan cabang dan kategori barang tertentu.

+searchBarang():void  
 Operasi ini digunakan untuk mencari data barang.

+getBarangById(id\_barang):void  
 Operasi ini digunakan untuk mengambil data barang yang dipergunakan pada saat akan melakukan ubah data barang.

+insertBarang(barang):void  
 Operasi ini digunakan untuk menambah data barang baru.

+editBarang(barang):void  
 Operasi ini digunakan untuk mengubah data barang.

```

+deleteBarang(id_barang):void
Operasi ini digunakan untuk menghapus data barang.
+updateStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan
melakukan pengurangan stok barang.
+updateTambahStokBarang(barang):void
Operasi ini digunakan untuk mengubah stok barang dengan
melakukan penambahan stok barang.
+updateStokDanHargaBarang(id_barang):void
Operasi ini digunakan untuk mengubah stok dan harga barang.

```

**Specific Design Class tran\_lunas\_model**

<b>tran_lunas_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre> +ID_TRAN_LUNAS: int ID dari transaksi lunas yang bersifat unik +KODE_TRAN_LUNAS: string Kode dari transaksi lunas +TANGGAL_TRAN_LUNAS: date Tanggal dari transaksi lunas +JENIS PEMBAYARAN: string Jenis pembayaran dari transaksi lunas +DISKON: float Diskon dari transaksi lunas +TOTAL_HARGA: float Total dari transaksi lunas +KETERANGAN: string Keterangan nomor credit card dari transaksi lunas +ID_CABANG: int ID Cabang dari transaksi lunas +ID_USER: int ID User dari transaksi lunas +ID_DETIL_LUNAS: int ID dari detail transaksi lunas yang bersifat unik +KUANTITAS: int </pre>	

Kuantitas dari detil transaksi lunas

+HARGA\_BRG: float

Harga barang dari detil transaksi lunas

+ID\_BARANG: int

ID Barang dari detil transaksi lunas

+ID\_TRAN\_LUNAS: int

ID Transaksi Lunas dari detil transaksi lunas

+getTranLunas():Tabel

Operasi ini digunakan untuk menampilkan data transaksi lunas.

+getTranLunasByTgl():void

Operasi ini digunakan untuk mencari data transaksi lunas berdasarkan tanggal.

+getDetilTranLunas():Tabel

Operasi ini digunakan untuk menampilkan data detil transaksi lunas.

+getJumlahTranLunasByTgl():void

Operasi ini digunakan untuk mendapatkan jumlah transaksi lunas berdasarkan tanggal yang dipergunakan untuk kode transaksi lunas.

+insertTranLunas(tran\_lunas):void

Operasi ini digunakan untuk menambah data transaksi lunas baru.

+updateTranLunas(tran\_lunas):void

Operasi ini digunakan untuk mengubah data transaksi lunas.

+insertDetilTranLunas(tran\_lunas):void

Operasi ini digunakan untuk menambah data detil transaksi lunas baru.

+updateDetilTranLunas(tran\_lunas):void

Operasi ini digunakan untuk mengubah data detil transaksi lunas.

+deleteDetilTranLunas(id\_detil\_lunas):void

Operasi ini digunakan untuk menghapus data detil transaksi lunas.

```
+cancelTranLunas(id_tran_lunas):void
Operasi ini digunakan untuk melakukan pembatalan data transaksi lunas.
+cancelDetilTranLunas(id_detil_lunas):void
Operasi ini digunakan untuk melakukan pembatalan data detil transaksi lunas.
```

### Specific Design Class retur\_lunas\_model

<b>retur_lunas_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_RETUR_LUNAS: int ID dari retur lunas yang bersifat unik +TANGGAL_RETUR: date Tanggal dari retur barang lunas +KUANTITAS_RETUR: int Kuantitas dari retur barang lunas +KETERANGAN_BARANG: string Keterangan barang dari retur barang lunas +ID_DETIL_LUNAS: int ID Detil Lunas dari retur barang lunas +ID_USER: int ID User dari retur barang lunas</pre>	
<pre>+getReturLunasById(id_tran_lunas):void Operasi ini digunakan untuk mendapatkan data retur pada suatu data transaksi lunas. +insertReturLunas(retur_lunas):void Operasi ini digunakan untuk menambah data retur transaksi lunas baru.</pre>	

### Specific Design Class tran\_bon\_model

<b>tran_bon_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+ID_TRAN_BON: int ID dari transaksi bon yang bersifat unik +KODE_TRAN_BON: string Kode dari transaksi bon</pre>	

<pre> +TANGGAL_TRAN_BON: date Tanggal dari transaksi bon +TANGGAL_BAYAR: date Tanggal pembayaran dari transaksi bon +STATUS_TRAN_BON: string Status dari transaksi bon +DISKON: float Diskon dari transaksi bon +TOTAL_HARGA: float Total dari transaksi bon +ID_CUSTOMER: int ID Customer dari transaksi bon +ID_CABANG: int ID Cabang dari transaksi bon +ID_USER: int ID User dari transaksi bon +ID_DETIL_BON: int ID dari detail transaksi bon yang bersifat unik +KUANTITAS: int Kuantitas dari detail transaksi bon +HARGA_BRG: float Harga barang dari detail transaksi bon +ID_BARANG: int ID Barang dari detail transaksi bon +ID_TRAN_BON: int ID Transaksi Bon dari detail transaksi bon </pre>
<pre> +getTranBon():Tabel Operasi ini digunakan untuk menampilkan data transaksi bon. +getTranBonByTgl():void Operasi ini digunakan untuk mencari data transaksi bon berdasarkan tanggal. +getDetilTranBon():Tabel Operasi ini digunakan untuk menampilkan data detail transaksi bon. </pre>

```

+getJumlahTranBonByTgl():void
Operasi ini digunakan untuk mendapatkan jumlah transaksi bon
berdasarkan tanggal yang dipergunakan untuk kode transaksi
bon.
+insertTranBon(transaksi_bon):void
Operasi ini digunakan untuk menambah data transaksi bon
baru.
+updateTranBon(transaksi_bon):void
Operasi ini digunakan untuk mengubah data transaksi bon.
+insertDetilTranBon(transaksi_bon):void
Operasi ini digunakan untuk menambah data detil transaksi
bon baru.
+updateDetilTranBon(transaksi_bon):void
Operasi ini digunakan untuk mengubah data detil transaksi
bon.
+deleteDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk menghapus data detil transaksi
bon.
+cancelTranBon(id_tran_bon):void
Operasi ini digunakan untuk melakukan pembatalan data
transaksi bon.
+cancelDetilTranBon(id_detil_bon):void
Operasi ini digunakan untuk melakukan pembatalan data detil
transaksi bon.
+finishTranBon(transaksi_bon):void
Operasi ini digunakan untuk melakukan pelunasan terhadap
suatu data transaksi bon.

```

**Specific Design Class retur\_bon\_model**

<b>retur_bon_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
+ID_RETUR_BON: int ID dari retur bon yang bersifat unik +TANGGAL_RETUR: date Tanggal dari retur barang bon	

<p>+KUANTITAS_RETUR: int Kuantitas dari retur barang bon</p> <p>+KETERANGAN_BARANG: string Keterangan barang dari retur barang bon</p> <p>+ID_DETIL_BON: int ID Detil Bon dari retur barang bon</p> <p>+ID_USER: int ID User dari retur barang bon</p>
<p>+getReturBonById(id_tran_bon):void Operasi ini digunakan untuk mendapatkan data retur pada suatu data transaksi bon.</p> <p>+insertReturBon(retur_bon):void Operasi ini digunakan untuk menambah data retur transaksi bon baru.</p>

**Specific Design Class grafik\_model**

<b>grafik_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<p>+grafikTranLunas():Grafik Operasi ini digunakan untuk menampilkan grafik untuk data transaksi lunas.</p> <p>+grafikTranBon():Grafik Operasi ini digunakan untuk menampilkan grafik untuk data transaksi bon.</p>	

**Specific Design Class reporting\_model**

<b>reporting_model</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<p>+getTranLunasReportUmumCbg():void Operasi ini digunakan untuk menampilkan report umum untuk data transaksi lunas.</p> <p>+getTranBonReportUmumCbg():void Operasi ini digunakan untuk menampilkan report umum untuk data transaksi bon.</p>	

```
+getTranLunasReportDetilCbg():void
Operasi ini digunakan untuk menampilkan report detil untuk data transaksi lunas.
+getTranBonReportDetilCbg():void
Operasi ini digunakan untuk menampilkan report detil untuk data transaksi bon.
+getDaftarBonReportCbg():void
Operasi ini digunakan untuk menampilkan report daftar bon.
+getReportBarangHabis():void
Operasi ini digunakan untuk menampilkan report daftar barang habis.
```

### Perancangan Data

### Dekomposisi Data

### Deskripsi Entitas Data Role

Nama	Tipe	Panjang	Keterangan
ID_ROLE	Integer	-	ID Role, primary key
NAMA_ROLE	Varchar	15	Nama Role

### Deskripsi Entitas Data User

Nama	Tipe	Panjang	Keterangan
ID_USER	Integer	-	ID User, primary key
USERNAME	Varchar	20	Username dari user
PASSWORD	Varchar	100	Password dari user
NAMA	Varchar	50	Nama dari user
EMAIL	Varchar	50	Email dari user
ALAMAT	Varchar	150	Alamat dari user
JENIS_KELAMIN	Varchar	20	Jenis kelamin dari user
FOTO	Varchar	150	Lokasi foto dari user
ID_ROLE	Integer	-	ID Role dari user, foreign key
ID_CABANG	Integer	-	ID Cabang dari user, foreign key

### Deskripsi Entitas Data Pegawai

Nama	Tipe	Panjang	Keterangan
------	------	---------	------------

Program Studi Teknik Informatika	DPPL – SISP	97/153
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

<b>ID_PEGAWAI</b>	Integer	-	<b>ID Pegawai, primary key</b>
NAMA_PEGAWAI	Varchar	50	Nama dari pegawai
ALAMAT_PEGAWAI	Varchar	150	Alamat dari pegawai
JENIS_KELAMIN	Varchar	20	Jenis kelamin dari pegawai
FOTO_PEGAWAI	Varchar	150	Lokasi foto dari pegawai
TUGAS	varchar	50	Tugas dari pegawai
ID_CABANG	Integer	-	ID Cabang dari pegawai, foreign key

### Deskripsi Entitas Data Customer

Nama	Tipe	Panjang	Keterangan
<b>ID_CUSTOMER</b>	Integer	-	<b>ID Customer, primary key</b>
NAMA_CUSTOMER	Varchar	50	Nama dari customer
ALAMAT_CUSTOMER	Varchar	150	Alamat dari customer
NOTELP_CUSTOMER	Varchar	50	Nomor telepon dari customer
ID_CABANG	Integer	-	ID Cabang dari customer, Foreign key

### Deskripsi Entitas Data Vendor

Nama	Tipe	Panjang	Keterangan
<b>ID_VENDOR</b>	Integer	-	<b>ID Vendor, primary key</b>
NAMA_VENDOR	Varchar	50	Nama dari vendor
ALAMAT_VENDOR	Varchar	150	Alamat dari vendor
NOTELP_VENDOR	Varchar	50	Nomor telepon dari vendor

### Deskripsi Entitas Data Cabang

Nama	Tipe	Panjang	Keterangan
<b>ID_CABANG</b>	Integer	-	<b>ID Cabang, primary key</b>
KODE_CABANG	Varchar	15	Kode dari cabang
LOGO_CABANG	Varchar	150	Logo dari cabang
NAMA_CABANG	Varchar	50	Nama dari cabang
ALAMAT_CABANG	Varchar	150	Alamat dari cabang
NOTELP_CABANG	Varchar	50	Nomor telepon dari

			cabang
--	--	--	--------

### Deskripsi Entitas Data Master\_Barang

Nama	Tipe	Panjang	Keterangan
ID_MASTER_BRG	Integer	-	ID Master Barang, primary key
NAMA_MASTER_BRG	Varchar	100	Nama dari master barang
MERK_MASTER_BRG	Varchar	50	Merk dari master barang

### Deskripsi Entitas Data Kategori\_Barang

Nama	Tipe	Panjang	Keterangan
ID_KATEGORI	Integer	-	ID Kategori, primary key
KODE_KATEGORI	Varchar	15	Kode dari kategori barang
NAMA_KATEGORI	Varchar	100	Nama dari kategori barang

### Deskripsi Entitas Data Barang

Nama	Tipe	Panjang	Keterangan
ID_BARANG	Integer	-	ID barang, primary key
NAMA_BARANG	Varchar	100	Nama dari barang
HARGA_BARANG	Float	-	Harga dari barang
STOK_BARANG	Integer	-	Stok dari barang
FOTO_BARANG	Varchar	150	Lokasi foto dari barang
HARGA_BELI	Float	-	Harga beli dari barang
MERK_BARANG	Varchar	50	Merk dari barang
BARCODE	Varchar	15	Barcode dari barang
ID_KATEGORI	Integer	-	ID Kategori dari barang, Foreign key
ID_CABANG	Integer	-	ID Cabang dari barang, Foreign key

### Deskripsi Entitas Data Barang\_Vendor

Nama	Tipe	Panjang	Keterangan
------	------	---------	------------

ID_MASTER_BRG	Integer	-	ID Master Barang, primary key dan foreign key
ID_VENDOR	Integer	-	ID Vendor, primary key dan foreign key
HARGA_BARANG_VENDOR	Float	-	Harga dari master barang
STATUS	Integer	-	Status dari harga master barang

### Deskripsi Entitas Data Transaksi\_Lunas

Nama	Tipe	Panjang	Keterangan
ID_TRAN_LUNAS	Integer	-	ID Transaksi Lunas, primary key
KODE_TRAN_LUNAS	Varchar	25	Kode dari transaksi lunas
TANGGAL_TRAN_LUNAS	Date	-	Tanggal dari transaksi lunas
JENIS PEMBAYARAN	Varchar	25	Jenis pembayaran dari transaksi lunas
DISKON	Float	-	Diskon dari transaksi lunas
TOTAL_HARGA	Float	-	Total dari transaksi lunas
KETERANGAN	Varchar	150	Keterangan nomor credit card dari transaksi lunas
ID_CABANG	Integer	-	ID Cabang dari transaksi lunas, Foreign key
ID_USER	Integer	-	ID User dari transaksi lunas, Foreign key

### Deskripsi Entitas Data Transaksi\_Bon

Nama	Tipe	Panjang	Keterangan
ID_TRAN_BON	Integer	-	ID Transaksi Bon, primary key
KODE_TRAN_BON	Varchar	25	Kode dari transaksi bon
TANGGAL_TRAN_BON	Date	-	Tanggal dari transaksi bon
TANGGAL_BAYAR	Date	-	Tanggal bayar dari transaksi bon
STATUS_TRAN_BON	Varchar	25	Status dari transaksi bon
DISKON	Float	-	Diskon dari transaksi bon
TOTAL_HARGA	Float	-	Total dari transaksi bon
ID_CUSTOMER	Integer	-	ID Customer dari transaksi bon, Foreign key
ID_CABANG	Integer	-	ID Cabang dari transaksi bon, Foreign key
ID_USER	Integer	-	ID User dari transaksi bon, Foreign key

### Deskripsi Entitas Data Pemesanan\_Barang

Nama	Tipe	Panjang	Keterangan
ID_PESAN	Integer	-	ID Pesan, primary key
KODE_PESAN	Varchar	25	Kode dari pemesanan barang
TANGGAL_PESAN	Date	-	Tanggal dari pemesanan barang
TOTAL_HARGA	Float	-	Total dari pemesanan barang
STATUS_PESAN	Varchar	50	Status dari pemesanan barang

ID_CABANG	Integer	-	ID Cabang dari pemesanan barang, Foreign key
ID_USER	Integer	-	ID User dari pemesanan barang, Foreign key
ID_VENDOR	Integer	-	ID Vendor dari pemesanan barang, Foreign key

### Deskripsi Entitas Data Detil\_Transaksi\_Lunas

Nama	Tipe	Panjang	Keterangan
ID_DETIL_LUNAS	Integer	-	ID Detil Transaksi Lunas, primary key
KUANTITAS	Integer	-	Kuantitas dari detil transaksi lunas
HARGA_BRG	Float	-	Harga barang dari detil transaksi lunas
ID_BARANG	Integer	-	ID Barang dari detil transaksi lunas, Foreign key
ID_TRAN_LUNAS	Integer	-	ID Transaksi Lunas dari detil transaksi lunas, Foreign key

### Deskripsi Entitas Data Detil\_Transaksi\_Bon

Nama	Tipe	Panjang	Keterangan
ID_DETIL_BON	Integer	-	ID Detil Transaksi Bon, primary key
KUANTITAS	Integer	-	Kuantitas dari detil transaksi bon
HARGA_BRG	Float	-	Harga barang dari detil transaksi lunas
ID_BARANG	Integer	-	ID Barang dari detil transaksi bon, Foreign key
ID_TRAN_BON	Integer	-	ID Transaksi Bon dari detil transaksi bon,

			Foreign key
--	--	--	-------------

### Deskripsi Entitas Data Detil\_Pemesanan\_Barang

Nama	Tipe	Panjang	Keterangan
<b>ID_DETIL_PESAN</b>	<b>Integer</b>	-	<b>ID Detil Pemesanan Barang, primary key</b>
KUANTITAS	Integer	-	Kuantitas dari detil pemesanan barang
HARGA_MASTER_BRG	Integer	-	Harga master marang dari detil pemesanan barang
ID_MASTER_BRG	Integer	-	ID Master Barang dari detil pemesanan barang, Foreign key
ID_VENDOR	Integer	-	ID Vendor dari detil pemesanan barang, Foreign key
ID_PESAN	Integer	-	ID Pemesanan Barang dari detil pemesanan barang, Foreign key

### Deskripsi Entitas Data Retur\_Barang\_Lunas

Nama	Tipe	Panjang	Keterangan
<b>ID_RETUR_LUNAS</b>	<b>Integer</b>	-	<b>ID Retur Barang Lunas, primary key</b>
TANGGAL_RETUR	Date	-	Tanggal dari retur barang lunas
KUANTITAS_RETUR	Integer	-	Kuantitas dari retur barang lunas
KETERANGAN_BARANG	Varchar	25	Keterangan barang dari retur barang lunas
ID_DETIL_LUNAS	Integer	-	ID Detil Lunas dari retur barang lunas, Foreign key
ID_USER	Integer	-	ID User dari retur barang lunas, Foreign key

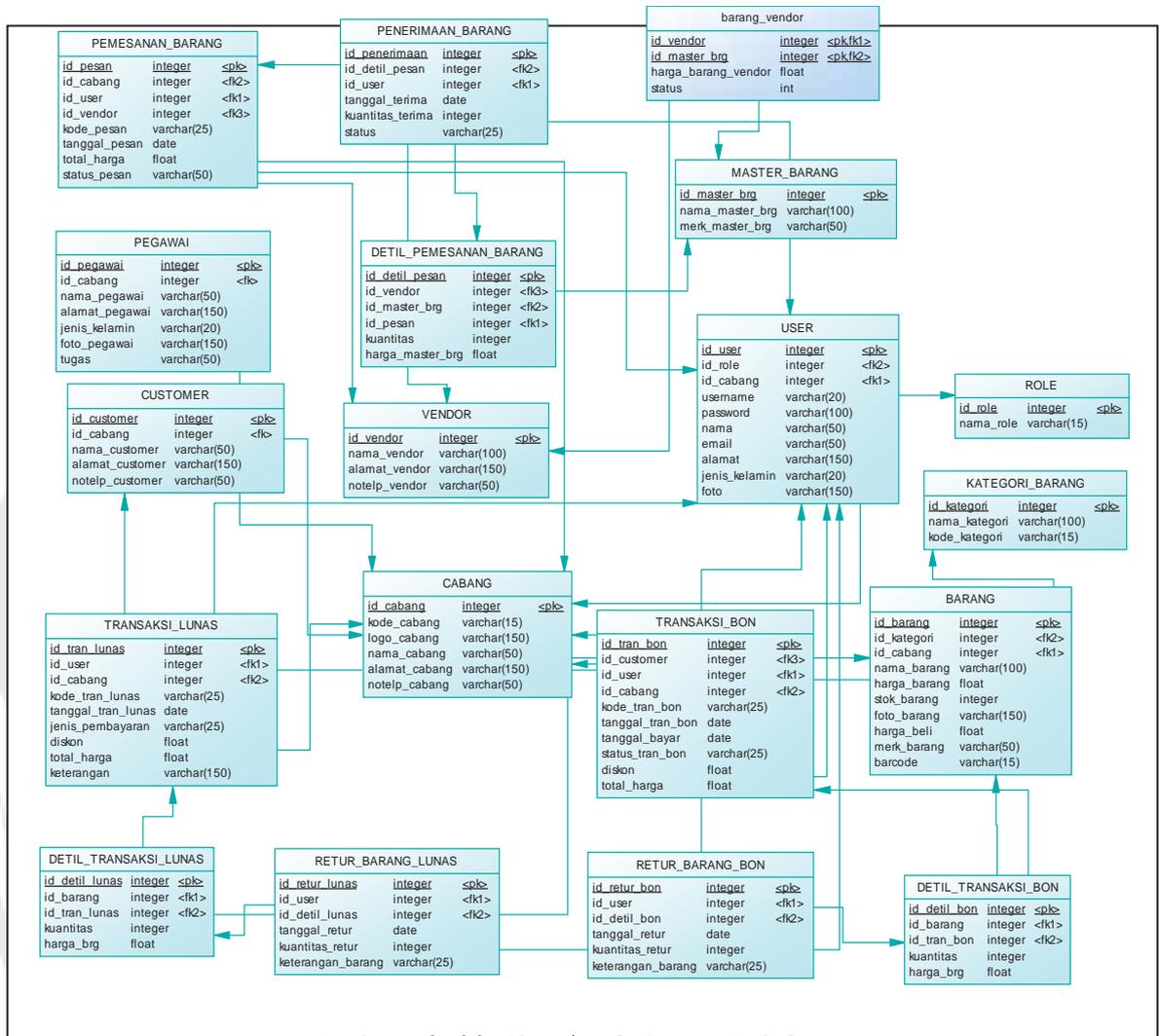
### Deskripsi Entitas Data Retur\_Barang\_Bon

Nama	Tipe	Panjang	Keterangan
ID_RETUR_BON	Integer	-	ID Retur Barang Bon, primary key
TANGGAL_RETUR	Date	-	Tanggal dari retur barang bon
KUANTITAS_RETUR	Integer	-	Kuantitas dari retur barang bon
KETERANGAN_BARANG	Varchar	25	Keterangan barang dari retur barang bon
ID_DETIL_BON	Integer	-	ID Detil Lunas dari retur barang bon, Foreign key
ID_USER	Integer	-	ID User dari retur barang bon, Foreign key

### Deskripsi Entitas Data Penerimaan\_Barang

Nama	Tipe	Panjang	Keterangan
ID_PENERIMAAN	Integer	-	ID Penerimaan Barang, primary key
TANGGAL_TERIMA	Integer	-	Tanggal terima dari penerimaan barang
KUANTITAS_TERIMA	Integer	-	Kuantitas terima dari penerimaan barang
STATUS	Varchar	25	Status terima dari penerimaan barang
ID_DETIL_PESAN	Integer	-	ID Detil Pesan dari penerimaan barang, Foreign key
ID_USER	Integer	-	ID User dari penerimaan barang, Foreign key

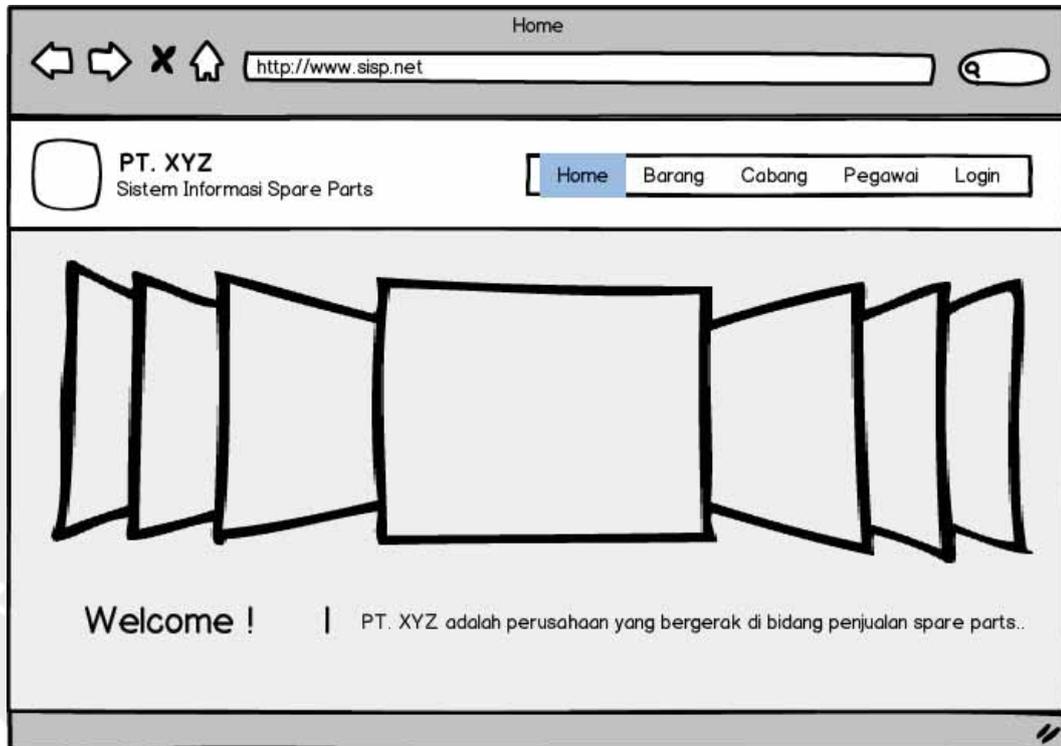
## Physical Data Model



Gambar 0.80 Physical Data Model

## Perancangan Antarmuka

### Antarmuka Home



Gambar 0.81 Rancangan Antarmuka Home

Antarmuka pada gambar 4.1 merupakan halaman awal yang akan ditampilkan ketika user membuka website SISP. Halaman ini memiliki navigasi menu di bagian atas, dan untuk halaman awal terdiri dari navigasi Home, Barang, Cabang, Pegawai, dan Login. Untuk masuk ke dalam sistem, user dapat melakukan login dengan memilih menu Login.

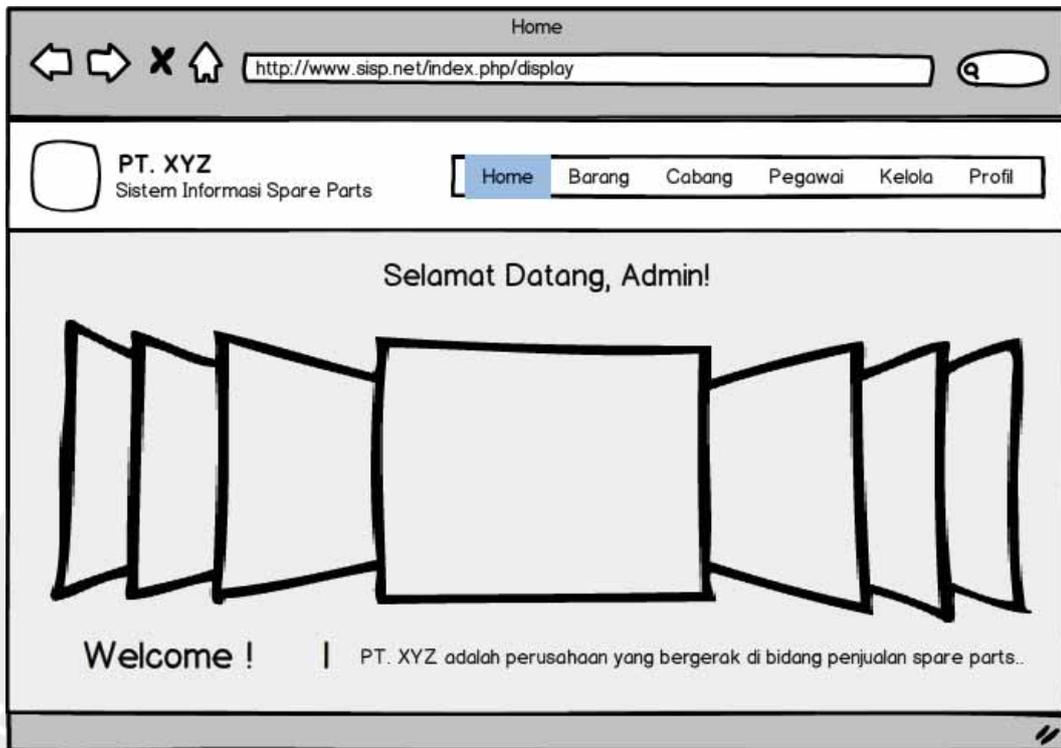
## Antarmuka Login

The image shows a web browser window with the following elements:

- Browser title: Login
- Address bar: http://www.sisp.net/displayLogin
- Page header: PT. XYZ Sistem Informasi Spare Parts
- Navigation menu: Home, Barang, Cabang, Pegawai, Login
- Main content: Login
- Form fields: Username, Password, Login button
- Footer: Belum terdaftar? Sign up [disini](#)

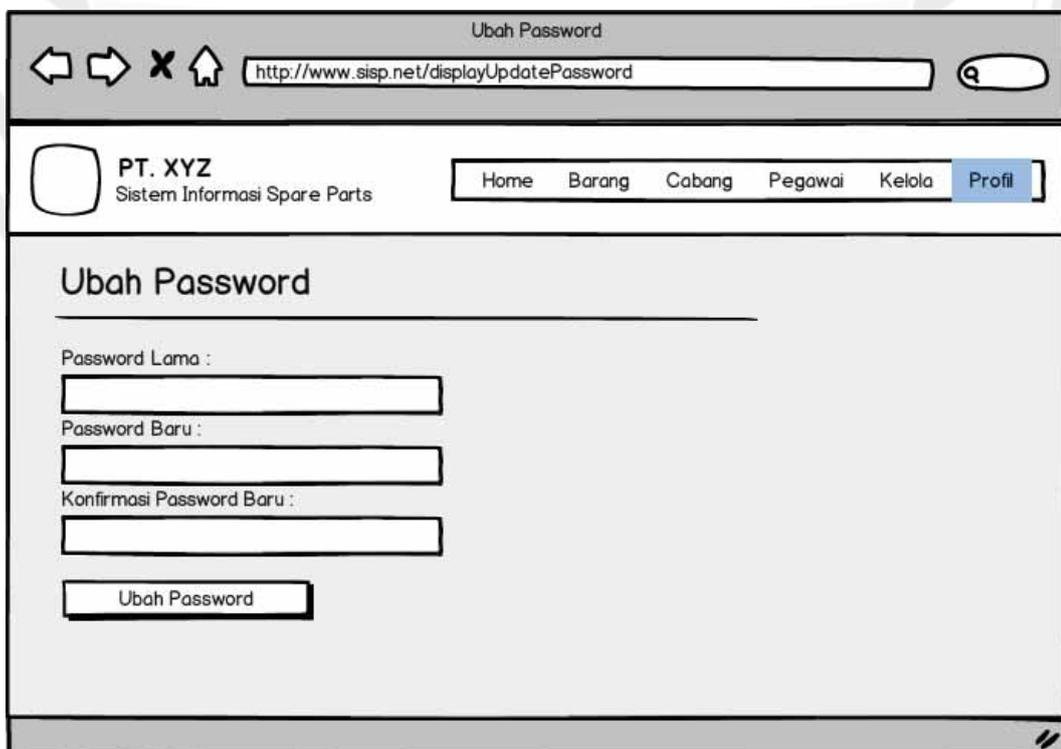
Gambar 0.82 Rancangan Antarmuka Login

Antarmuka pada gambar 4.2 digunakan untuk melakukan proses login ke dalam sistem. Untuk mendapat akses masuk ke dalam sistem, user harus menginputkan login username dan password dengan benar pada textbox yang telah disediakan di halaman ini. Pada saat tombol Login ditekan, sistem akan melakukan pengecekan username dan password yang diinputkan dengan data username dan password yang telah tersimpan di database. Jika data username dan password benar atau cocok, maka user akan masuk ke dalam sistem, sebaliknya jika id dan password salah atau tidak cocok maka akan muncul pesan peringatan. Jika user belum terdaftar, dapat dengan mengklik link *sign up here* yang terdapat di bawah form login.



Gambar 0.83 Rancangan Antarmuka Home setelah Login

#### Antarmuka Ubah Password



Gambar 0.84 Rancangan Antarmuka Ubah Password

Antarmuka pada gambar 4.4 akan muncul setelah pengguna melakukan login. Antarmuka ini digunakan untuk mengganti password dari user yang sudah login dimana inputannya terdiri dari password lama, password baru, dan konfirmasi password baru. Dan terdapat pengoperasian Ubah Password melalui tombol yang ada. Jika data data yang diinputkan benar, maka penggantian password user berhasil, sebaliknya jika data yang diinputkan salah, akan muncul pesan peringatan.

#### Antarmuka Pendaftaran User Baru

The screenshot shows a web browser window with the title 'Tambah User'. The address bar contains 'http://www.sisp.net/displayInsertUser'. The page header includes a logo for 'PT. XYZ Sistem Informasi Spare Parts' and a navigation menu with 'Home', 'Barang', 'Cabang', 'Pegawai', and 'Login'. The main content area is titled 'Pendaftaran User' and contains the following form fields:

- Nama : [input box]
- Email : [input box]
- Alamat : [input box]
- Jenis Kelamin :
  - Laki-laki
  - Perempuan
- Foto : [Choose File] No file chosen
- Cabang : [Dropdown menu with 'Babarsari' selected]
- Username : [input box]
- Password : [input box]

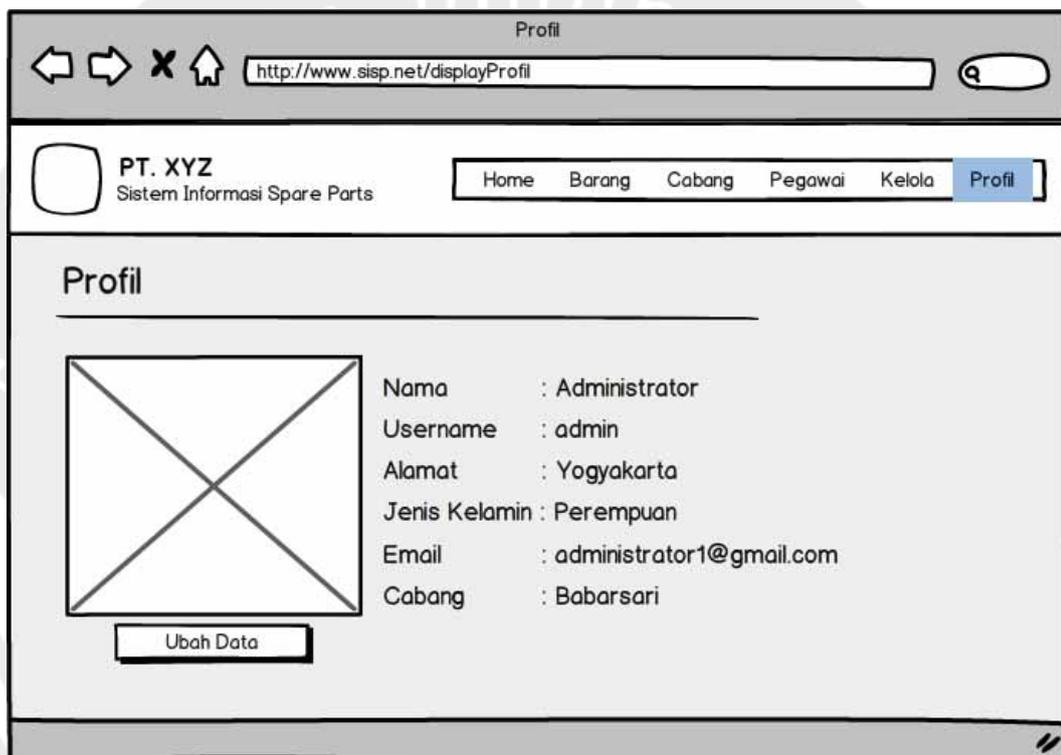
At the bottom of the form are two buttons: 'Insert' and 'Clear'.

Gambar 0.85 Rancangan Antarmuka Pendaftaran User Baru

Halaman pada gambar 4.5 merupakan halaman pendaftaran user baru. Halaman ini menampilkan *input box* seperti nama, email,

alamat, jenis kelamin, foto, cabang, username, dan password yang harus diisi user. Jika data yang diinputkan benar, maka user sudah terdaftar dalam sistem dan dapat melakukan login. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

#### Antarmuka Pengelolaan User



Gambar 0.86 Rancangan Antarmuka Pengelolaan User

Halaman pada gambar 4.6 merupakan halaman pengelolaan user yang menampilkan data user. User dapat mengubah data mereka dengan mengklik tombol Ubah Data yang terdapat di bawah foto. Halaman Ubah User akan ditampilkan seperti gambar 4.7.

Pengelolaan User

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai Kelola **Profil**

### Pengelolaan User

Nama :  
Administrator

Email :  
administrator1@gmail.com

Alamat :  
Yogyakarta

Jenis Kelamin :  
 Laki-laki  
 Perempuan

Foto :  
 No file chosen

Cabang :  
 ▼

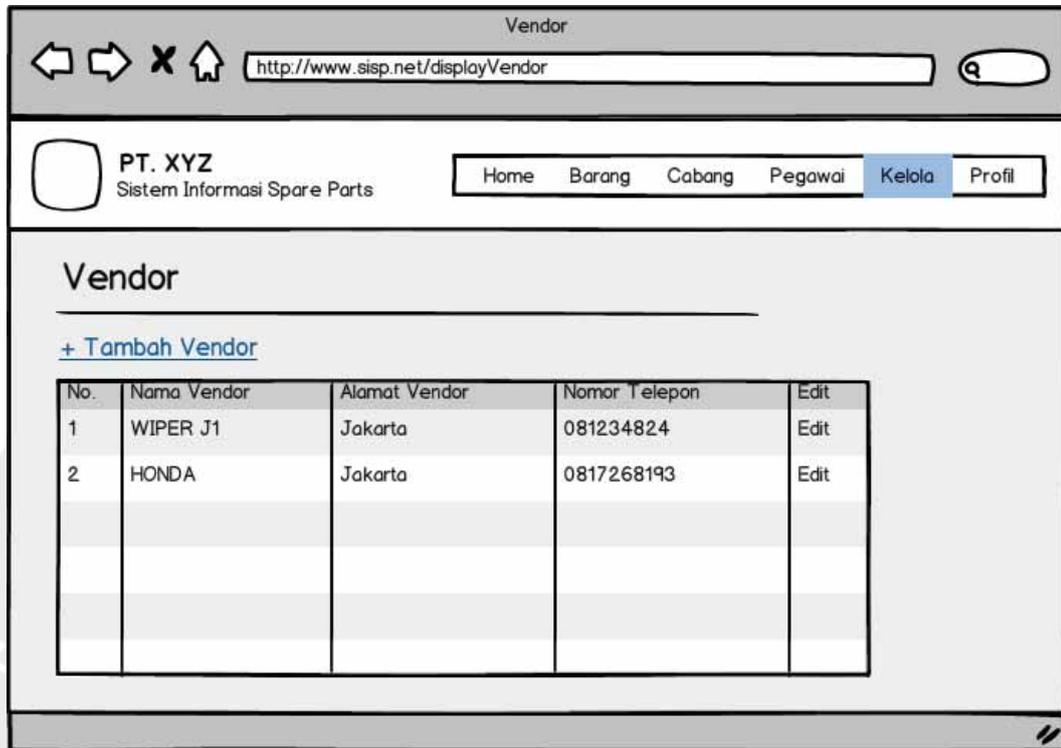
Username :  
admin

Password :

Gambar 0.87 Rancangan Antarmuka Ubah Data User

Halaman pada gambar 4.7 merupakan halaman untuk melakukan ubah data user. Pada *input box yang ada* telah terisi data user tersebut. User dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

## Antarmuka Pengelolaan Vendor



Gambar 0.88 Rancangan Antarmuka Pengelolaan Vendor

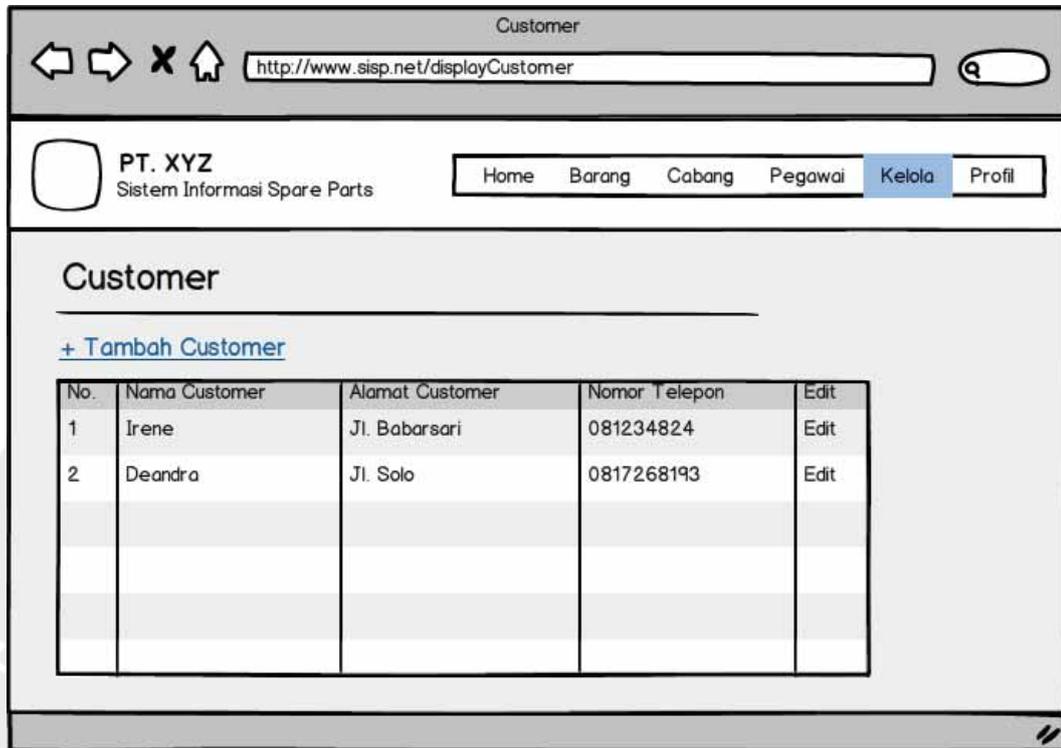
Halaman pada gambar 4.8 merupakan halaman pengelolaan vendor yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data-data vendor yang ada. Administrator dapat menambah atau mengubah data vendor. Untuk menambahkan data vendor, dapat dengan mengeklik link Tambah Vendor. Halaman Tambah Vendor akan ditampilkan seperti gambar 4.9. Untuk mengubah data vendor, dapat dengan mengklik icon edit yang ada.

The image shows a web browser window with the title 'Vendor'. The address bar contains the URL 'http://www.sisp.net/displayInsertVendor'. The page header features a logo for 'PT. XYZ Sistem Informasi Spare Parts' and a navigation menu with items: 'Home', 'Barang', 'Cabang', 'Pegawai', 'Kelola', and 'Profil'. The main content area is titled 'Pengelolaan Vendor' and contains a form with three input fields labeled 'Nama:', 'Alamat:', and 'Nomor Telepon:'. Below the input fields are two buttons: 'Insert' and 'Clear'.

Gambar 0.89 Rancangan Antarmuka Pengelolaan Tambah Vendor

Halaman pada gambar 4.9 merupakan halaman untuk menambahkan data vendor. Halaman ini menampilkan input box nama vendor, alamat vendor, dan nomor telepon vendor yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data vendor baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.9. Namun, pada *input box* yang ada telah terisi data vendor yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

## Antarmuka Pengelolaan Customer



Gambar 0.90 Rancangan Antarmuka Pengelolaan Customer

Halaman pada gambar 4.10 merupakan halaman pengelolaan customer yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data-data customer yang ada. Administrator dapat menambah atau mengubah data customer. Untuk menambahkan data customer, dapat dengan mengeklik link Tambah Customer. Halaman Tambah Customer akan ditampilkan seperti gambar 4.11. Untuk mengubah data customer, dapat dengan mengeklik icon edit yang ada.

The image shows a web browser window with the following elements:

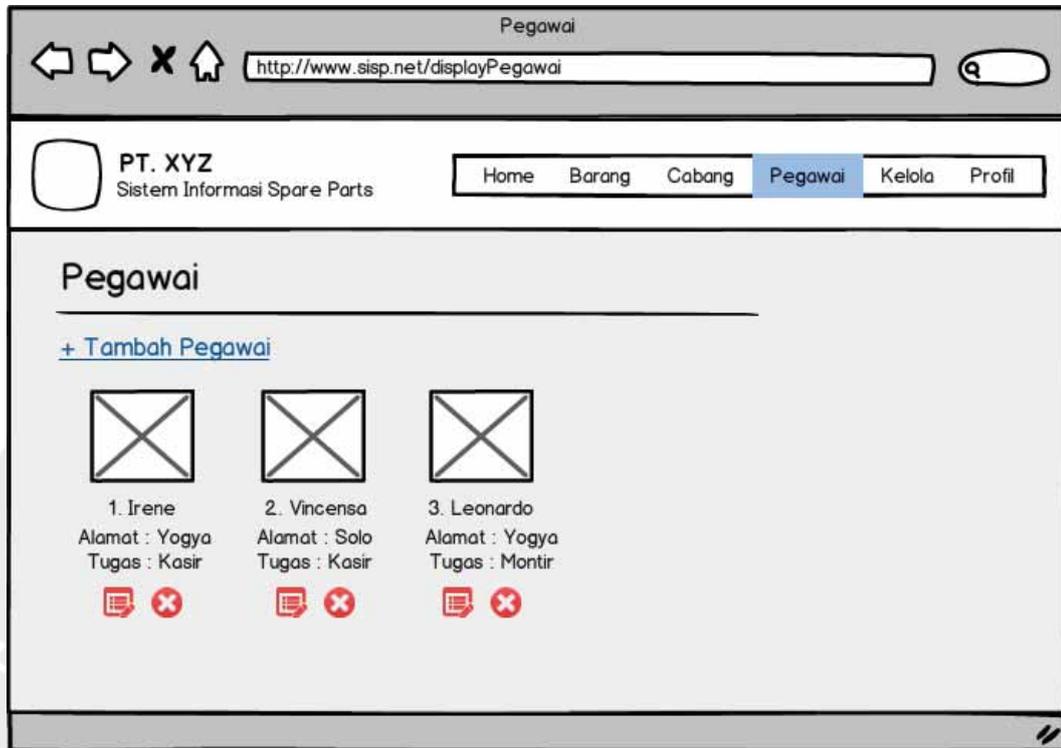
- Browser Title:** Customer
- Address Bar:** http://www.sisp.net/displayInsertCustomer
- Page Header:** PT. XYZ Sistem Informasi Spare Parts
- Navigation Menu:** Home, Barang, Cabang, Pegawai, **Kelola**, Profil
- Main Content Area:**
  - Title:** Pengelolaan Customer
  - Form Fields:**
    - Nama :
    - Alamat :
    - Nomor Telepon :
  - Buttons:** Insert, Clear

Gambar 0.91 Rancangan Antarmuka Pengelolaan Tambah Customer

Halaman pada gambar 4.11 merupakan halaman untuk menambahkan data customer. Halaman ini menampilkan input box nama customer, alamat customer, dan nomor telepon customer yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data customer baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.11. Namun, pada *input box* yang ada telah terisi data customer yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

## Antarmuka Pengelolaan Pegawai



Gambar 0.92 Rancangan Antarmuka Pengelolaan Pegawai

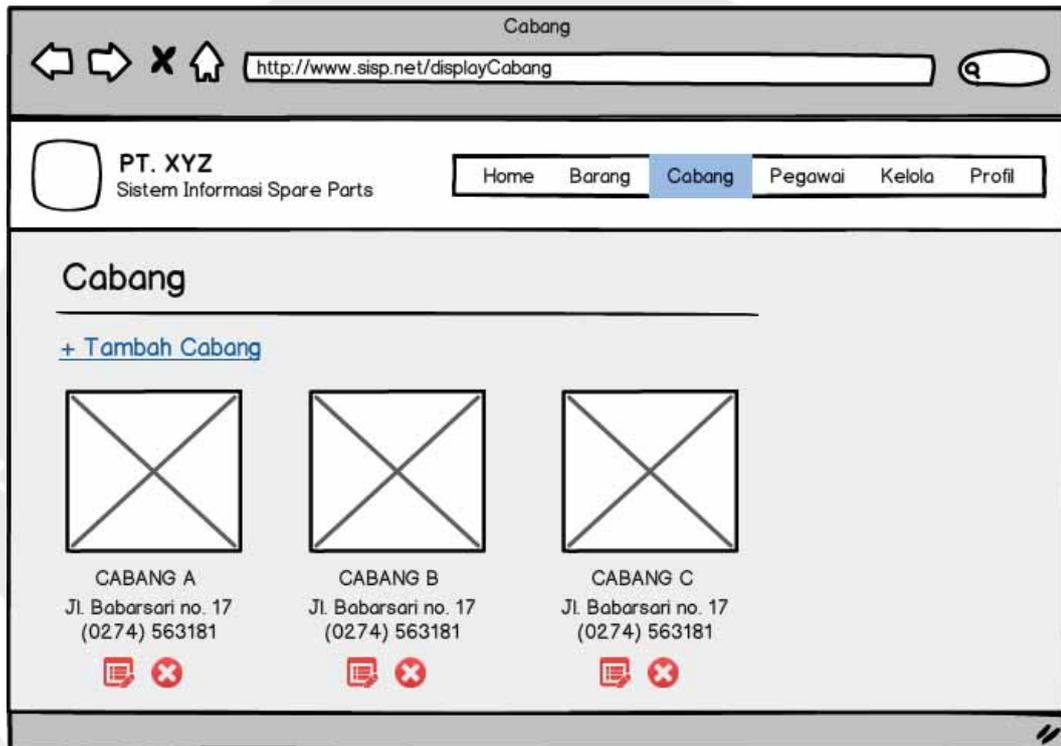
Halaman pada gambar 4.12 merupakan halaman pengelolaan pegawai yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data-data pegawai yang ada. Administrator dapat menambah, mengubah, atau menghapus data pegawai. Untuk menambahkan data pegawai, dapat dengan mengeklik link Tambah Pegawai. Halaman Tambah Pegawai akan ditampilkan seperti gambar 4.13. Untuk mengubah data customer, dapat dengan mengeklik icon edit yang ada. Sedangkan untuk menghapus data pegawai, dapat dengan mengeklik icon delete yang ada.

Gambar 0.93 Rancangan Antarmuka Pengelolaan Tambah Pegawai

Halaman pada gambar 4.13 merupakan halaman untuk menambahkan data pegawai. Halaman ini menampilkan input box nama, alamat, jenis kelamin, foto, tugas, dan cabang yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data pegawai baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.13. Namun, pada *input box* yang ada telah terisi data pegawai yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

Jika icon Delete dipilih, akan muncul konfirmasi penghapusan. Jika Administrator memilih 'Ya', maka data pegawai akan terhapus.

### Antarmuka Pengelolaan Cabang



Gambar 0.94 Rancangan Antarmuka Pengelolaan Cabang

Halaman pada gambar 4.14 merupakan halaman pengelolaan cabang yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data-data cabang yang ada. Administrator dapat menambah, mengubah, atau menghapus data cabang. Untuk menambahkan data cabang, dapat dengan mengeklik link Tambah Cabang. Halaman Tambah Cabang akan ditampilkan seperti gambar 4.15. Untuk mengubah data cabang, dapat dengan mengeklik icon edit yang ada. Sedangkan untuk menghapus data cabang, dapat dengan mengeklik icon delete yang ada.

The image shows a web browser window with the title 'Cabang'. The address bar contains the URL 'http://www.sisp.net/displayInsertCabang'. The page header features the logo for 'PT. XYZ Sistem Informasi Spare Parts' and a navigation menu with the following items: 'Home', 'Barang', 'Cabang', 'Pegawai', 'Kelola', and 'Profil'. The main content area is titled 'Pengelolaan Cabang' and contains a form for adding a new branch. The form includes the following fields and controls:

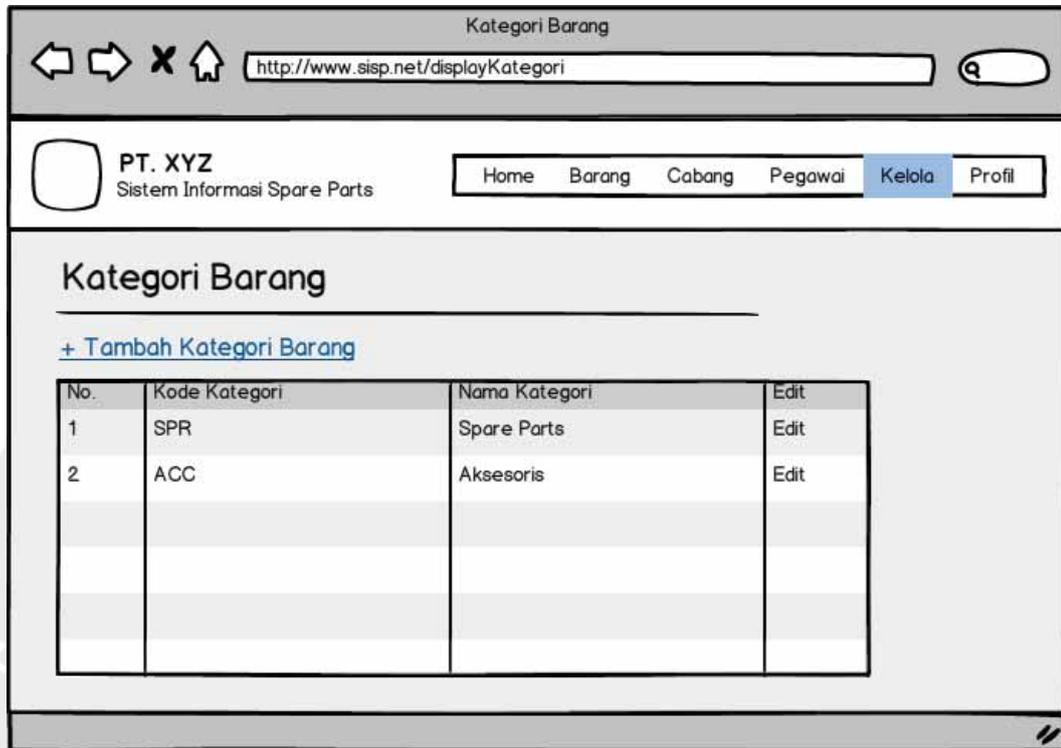
- Kode :** A text input field.
- Logo :** A file upload control with a 'Choose File' button and the text 'No file chosen'.
- Nama :** A text input field.
- Alamat :** A text input field.
- Nomor Telepon :** A text input field.
- Buttons:** 'Insert' and 'Clear' buttons at the bottom of the form.

Gambar 0.95 Rancangan Antarmuka Pengelolaan Tambah Cabang Halaman pada gambar 4.15 merupakan halaman untuk menambahkan data cabang. Halaman ini menampilkan input box kode cabang, logo, nama, alamat, dan nomor telepon yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data cabang baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.15. Namun, pada *input box* yang ada telah terisi data cabang yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

Jika icon Delete dipilih, akan muncul konfirmasi penghapusan. Jika Administrator memilih 'Ya', maka data cabang akan terhapus.

## Antarmuka Pengelolaan Kategori Barang



Gambar 0.96 Rancangan Antarmuka Pengelolaan Kategori Barang Halaman pada gambar 4.16 merupakan halaman pengelolaan kategori barang yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data-data barang berdasarkan kategori barang yang ada. Administrator dapat menambah atau mengubah kategori barang. Untuk menambahkan data kategori barang, dapat dengan mengeklik link Tambah Kategori Barang. Halaman Tambah Kategori Barang akan ditampilkan seperti gambar 4.17. Untuk mengubah data kategori barang, dapat dengan mengeklik icon edit yang ada.

Kategori Barang

http://www.sisp.net/displayInsertKategori

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Kelola** Profil

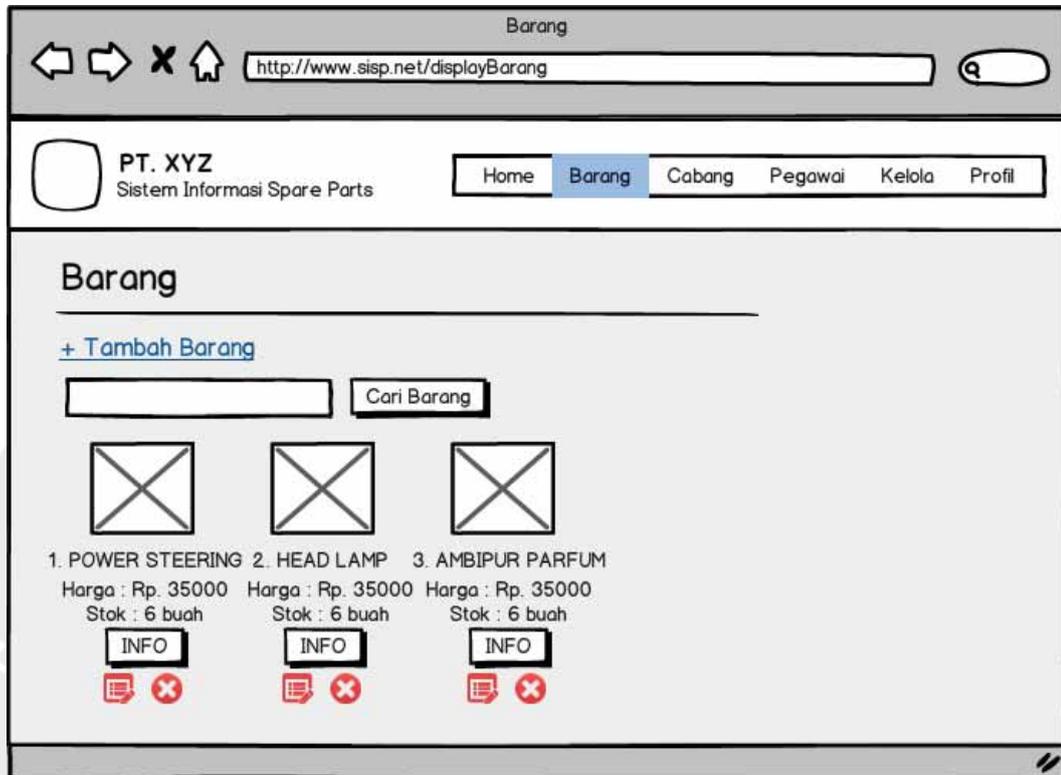
### Pengelolaan Kategori Barang

Kode Kategori :

Nama Kategori :

Gambar 0.97 Rancangan Antarmuka Pengelolaan Tambah Kategori Barang Halaman pada gambar 4.17 merupakan halaman untuk menambahkan data kategori barang. Halaman ini menampilkan input box kode dan nama kategori barang yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data kategori barang baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.17. Namun, pada *input box* yang ada telah terisi data kategori barang yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

## Antarmuka Pengelolaan Barang



Gambar 0.98 Rancangan Antarmuka Pengelolaan Barang

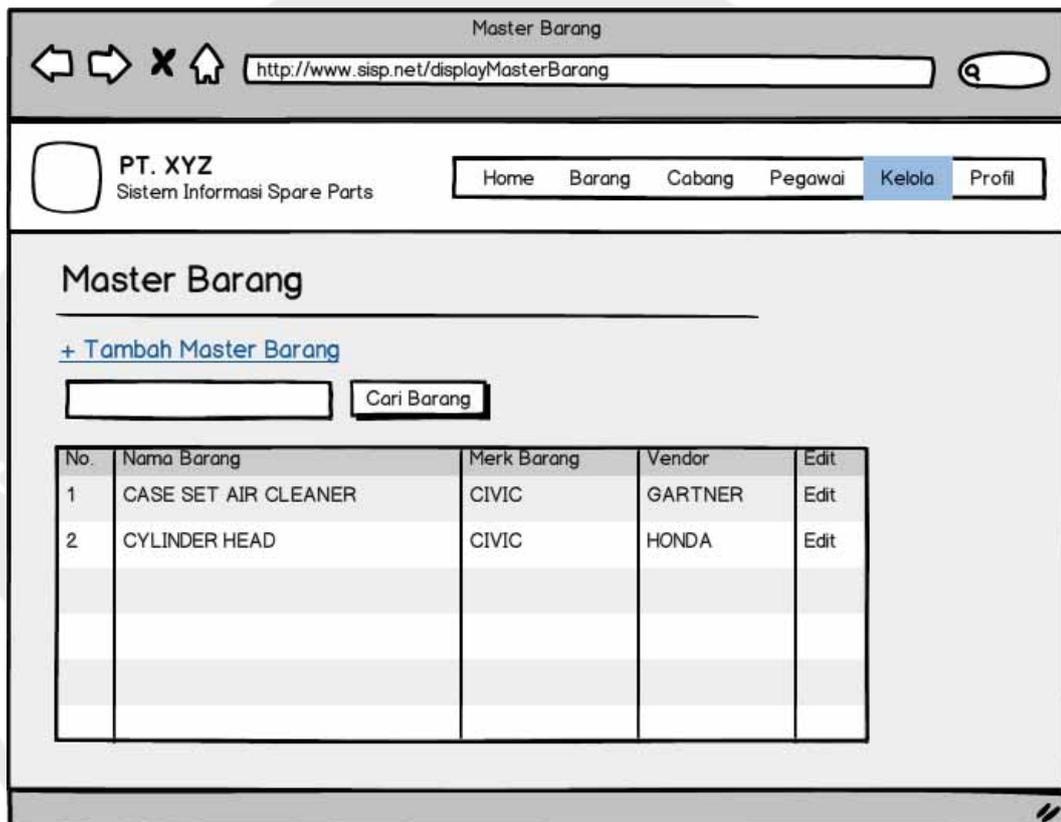
Halaman pada gambar 4.18 merupakan halaman pengelolaan barang yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data barang yang ada. Jika menu Barang diklik, maka akan muncul semua barang. Jika kursor diarahkan ke menu Barang, akan muncul sub menu kategori barang. Jika salah satu kategori barang tersebut diklik, maka akan muncul data barang sesuai dengan kategori yang dipilih. Administrator dapat menambah, mengubah, atau menghapus barang. Untuk menambahkan data barang, dapat dengan mengklik link Tambah Barang. Halaman Tambah Barang akan ditampilkan seperti gambar 4.19. Untuk mengubah data barang, dapat dengan mengklik icon edit yang ada. Sedangkan untuk menghapus data barang, dapat dengan mengklik icon delete yang ada.

Gambar 0.99 Rancangan Antarmuka Pengelolaan Tambah Barang Halaman pada gambar 4.19 merupakan halaman untuk menambahkan data barang. Halaman ini menampilkan input box nama, harga beli, harga jual, stok, foto, merk, keterangan barcode, dan kategori barang yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data barang baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.19. Namun, pada *input box* yang ada telah terisi data barang yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

Jika icon Delete dipilih, akan muncul konfirmasi penghapusan. Jika Administrator memilih 'Ya', maka data barang akan terhapus.

### Antarmuka Pengelolaan Master Barang



Gambar 0.100 Rancangan Antarmuka Pengelolaan Master Barang

Halaman pada gambar 4.20 merupakan halaman pengelolaan master barang yang hanya bisa diakses oleh Administrator. Halaman ini menampilkan data master barang yang ada. Administrator dapat menambah, mengubah, atau mencari master barang. Untuk menambahkan data master barang, dapat dengan mengklik link Tambah Master Barang. Halaman Tambah Master Barang akan ditampilkan seperti gambar 4.21. Untuk mengubah data master barang, dapat dengan mengklik icon edit yang ada. Untuk mencari barang, dapat dengan mengisi nama master barang yang ingin di cari di *input box* yang ada lalu tekan tombol Cari Barang.

Master Barang

http://www.sisp.net/displayInsertMasterBarang

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Kelola** Profil

### Pengelolaan Master Barang

Nama :

Merk :

Harga :

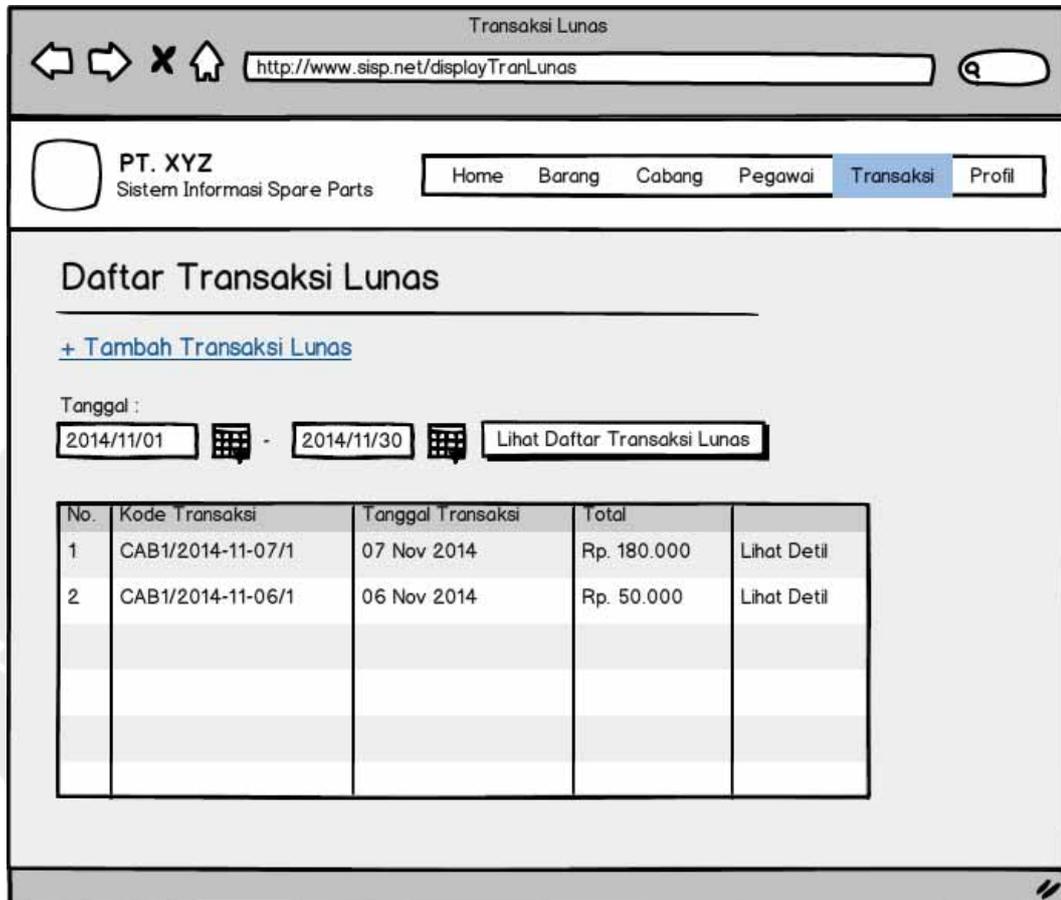
Vendor :

\* Tidak menemukan Vendor? Tambah data vendor terlebih dahulu [disini](#)

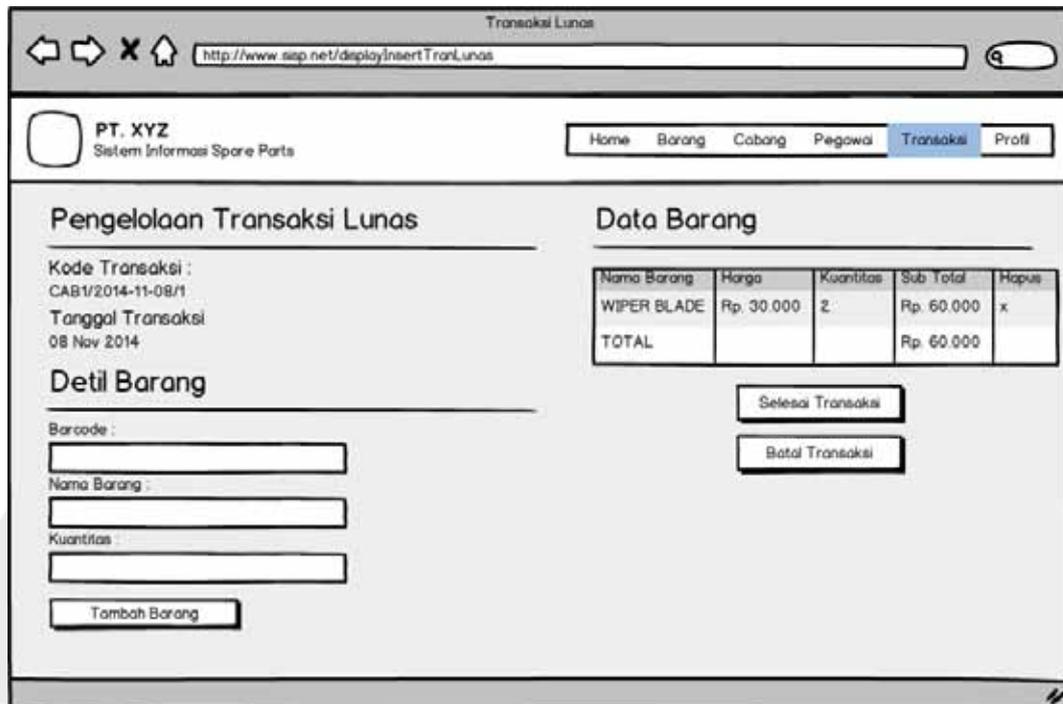
Gambar 0.101 Rancangan Antarmuka Pengelolaan Tambah Master Barang Halaman pada gambar 4.21 merupakan halaman untuk menambahkan data master barang. Halaman ini menampilkan input box nama, merk, harga, dan nama vendor yang harus diisi oleh Administrator. Jika data yang diinputkan benar, maka data master barang baru akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika icon Edit dipilih, akan ditampilkan halaman seperti gambar 4.21. Namun, pada *input box* yang ada telah terisi data master barang yang dipilih. Administrator dapat melakukan ubah data di *input box* tersebut, dan untuk menyimpan perubahan data dapat dengan mengklik tombol Ubah.

## Antarmuka Pengelolaan Transaksi Lunas



Gambar 0.102 Rancangan Antarmuka Pengelolaan Transaksi Lunas Halaman pada gambar 4.22 merupakan halaman pengelolaan transaksi lunas yang bisa diakses oleh Kasir. Halaman ini menampilkan data transaksi lunas pada tanggal hari ini. Kasir dapat menambah, konfirmasi, batal transaksi, cetak nota, melihat info transaksi, atau mencari data transaksi lunas. Untuk menambahkan data transaksi lunas, dapat dengan mengklik link Tambah Transaksi Lunas. Halaman Tambah Transaksi Lunas akan ditampilkan seperti gambar 4.23.



Gambar 0.103 Rancangan Antarmuka Pengelolaan Tambah Transaksi Lunas

Halaman pada gambar 4.23 merupakan halaman untuk menambahkan data transaksi lunas. Halaman ini menampilkan kode transaksi dan tanggal transaksi yang di generate secara otomatis. Data transaksi lunas akan ditambahkan ketika memasuki halaman ini. Lalu terdapat *input box* barcode, nama barang, dan kuantitas yang harus diisi oleh Kasir. Jika data yang diinputkan benar, maka detil transaksi lunas akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika tombol Selesai Transaksi dipilih, akan ditampilkan halaman seperti gambar 4.24 sedangkan jika tombol Batal Transaksi dipilih, maka data transaksi serta detil transaksi lunas akan dihapus dan kembali ke halaman seperti gambar 4.22.

Gambar 0.104 Rancangan Antarmuka Pengelolaan Konfirmasi dan Cetak Nota Transaksi Lunas

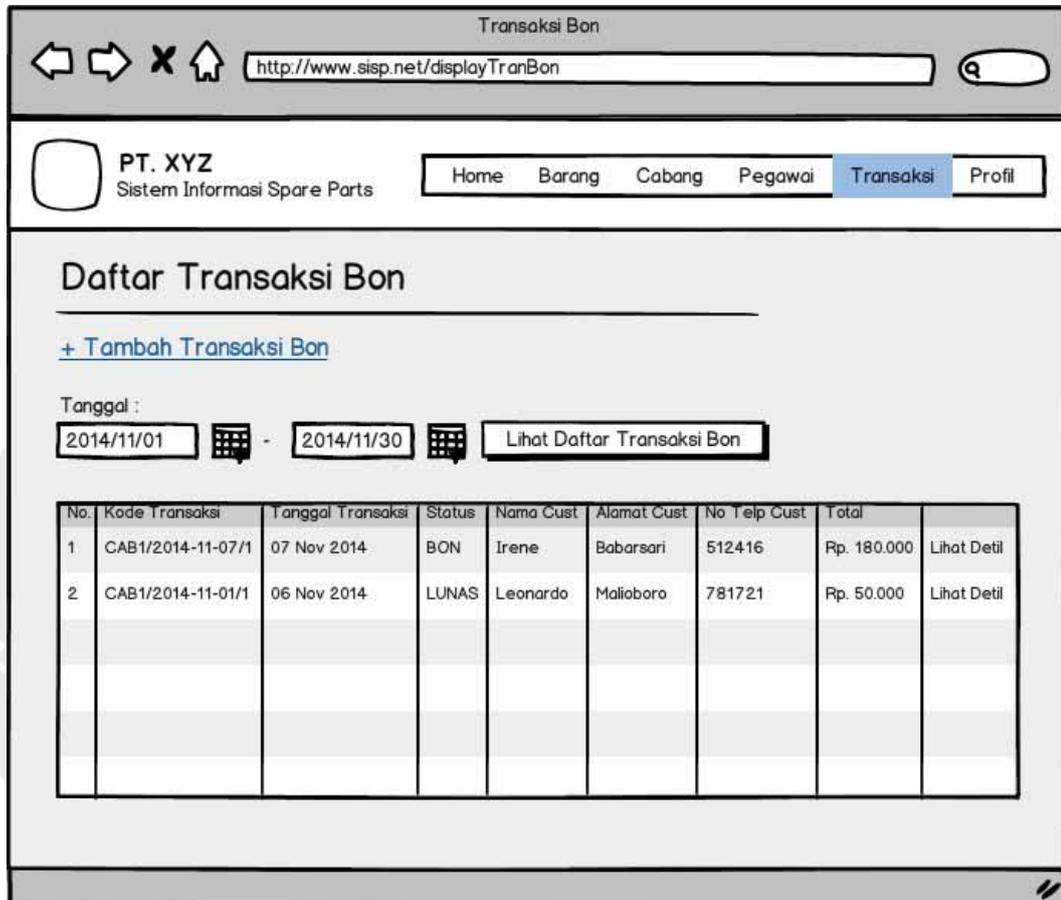
Halaman pada gambar 4.24 merupakan halaman untuk melakukan konfirmasi transaksi lunas. Lalu terdapat *input box* diskon, *combo box* jenis pembayaran, dan *input box* keterangan untuk nomor credit card yang harus diisi oleh Kasir. Jika data yang diinputkan benar, maka konfirmasi transaksi lunas akan dilakukan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika tombol Cetak Nota dipilih, maka konfirmasi transaksi lunas dilakukan dan akan ditampilkan nota seperti gambar 4.25 sedangkan jika tombol Batal Transaksi dipilih, maka data transaksi serta detail transaksi lunas akan dihapus dan kembali ke halaman seperti gambar 4.22.

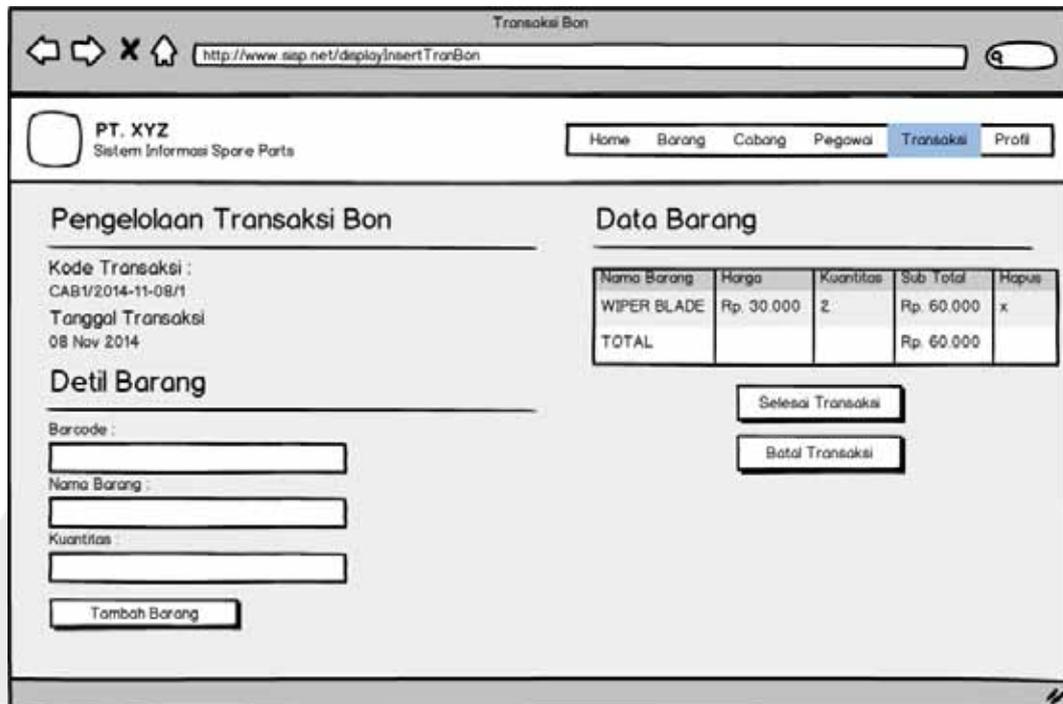
PT. XYZ Adisucipto	
Jl. Adisucipto no. 17	
Telp. (0274) 591823	
<hr/>	
2014-11-08 15:41:20	
CAB1/2014-11-08/1 #Leonardo	
WIPER BLADE	
2 X Rp. 30.000	Rp. 60.000
ITEMS : 2	
<b>SUBTOTAL</b>	<b>Rp. 60.000</b>
TERIMA KASIH	

Gambar 0.105 Rancangan Antarmuka Nota Transaksi Lunas  
 Halaman pada gambar 4.25 merupakan nota transaksi lunas yang dapat dicetak oleh Kasir. Di nota ini, terdapat informasi tanggal dan waktu transaksi, kode transaksi, serta nama Kasir. Lalu terdapat nama barang, kuantitas, subtotal, total, serta jumlah barang yang dibeli oleh Customer.

## Antarmuka Pengelolaan Transaksi Bon

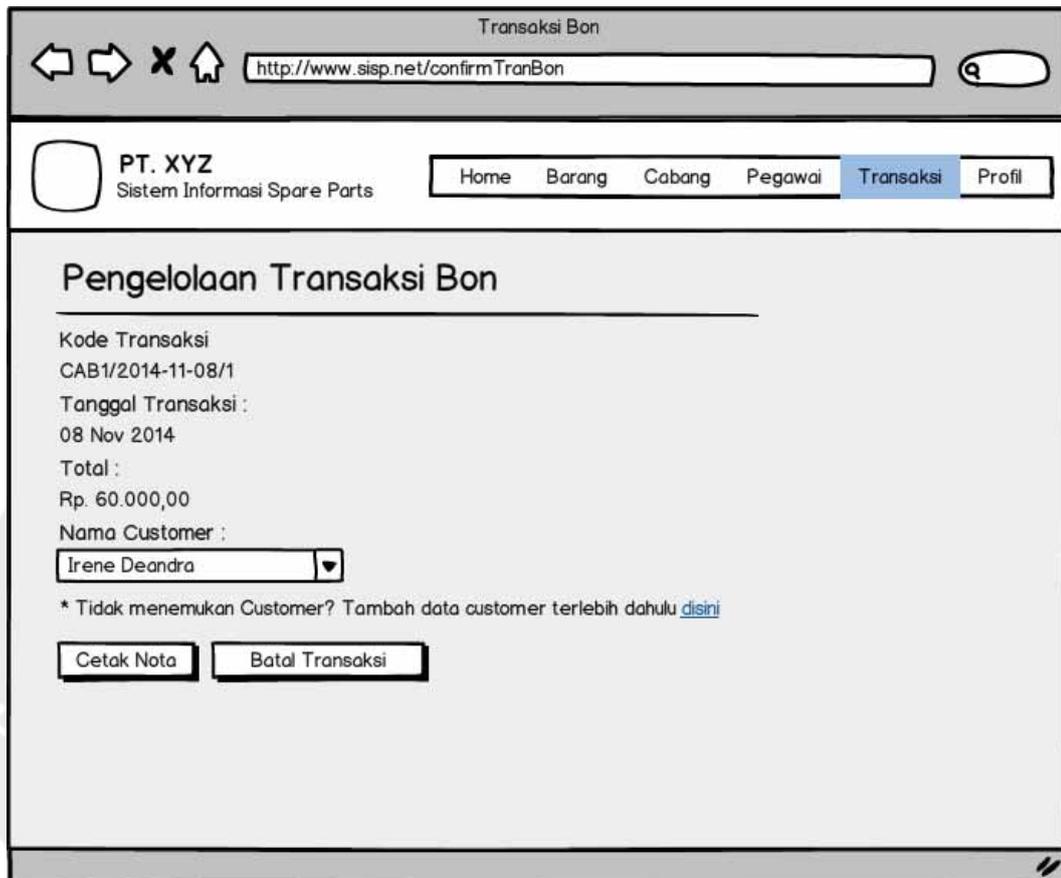


Gambar 0.106 Rancangan Antarmuka Pengelolaan Transaksi Bon Halaman pada gambar 4.26 merupakan halaman pengelolaan transaksi bon yang bisa diakses oleh Kasir. Halaman ini menampilkan data transaksi bon pada tanggal hari ini. Kasir dapat menambah, konfirmasi, batal transaksi, cetak nota, melihat info transaksi, mencari data transaksi lunas, atau melakukan pelunasan transaksi bon. Untuk menambahkan data transaksi bon, dapat dengan mengeklik link Tambah Transaksi Bon. Halaman Tambah Transaksi Bon akan ditampilkan seperti gambar 4.27.



Gambar 0.107 Rancangan Antarmuka Pengelolaan Tambah Transaksi Bon Halaman pada gambar 4.27 merupakan halaman untuk menambahkan data transaksi bon. Halaman ini menampilkan kode transaksi dan tanggal transaksi yang di generate secara otomatis. Data transaksi bon akan ditambahkan ketika memasuki halaman ini. Lalu terdapat *input box* barcode, nama barang, dan kuantitas yang harus diisi oleh Kasir. Jika data yang diinputkan benar, maka detil transaksi bon akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan.

Jika tombol Selesai Transaksi dipilih, akan ditampilkan halaman seperti gambar 4.28 sedangkan jika tombol Batal Transaksi dipilih, maka data transaksi serta detil transaksi bon akan dihapus dan kembali ke halaman seperti gambar 4.26.



Gambar 0.108 Rancangan Antarmuka Pengelolaan Konfirmasi dan Cetak Nota Transaksi Bon

Halaman pada gambar 4.28 merupakan halaman untuk melakukan konfirmasi transaksi bon. Lalu terdapat *combo box* nama customer yang melakukan pembelian yang harus diisi oleh Kasir. Jika data yang diinputkan benar, maka konfirmasi transaksi lunas akan dilakukan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika data customer belum terdapat di *combo box* tersebut, maka Kasir bias mengeklik link "disini" yang akan menuju ke halaman Tambah Customer.

Jika tombol Cetak Nota dipilih, maka konfirmasi transaksi bon dilakukan dan akan ditampilkan nota seperti gambar 4.29 sedangkan jika tombol Batal Transaksi dipilih, maka data transaksi serta detil transaksi bon akan dihapus dan kembali ke halaman seperti gambar 4.26.

PT. XYZ Adisucipto	
Jl. Adisucipto no. 17	
Telp. (0274) 591823	
2014-11-08 15:41:20	
CAB1/2014-11-08/1 #Leonardo	
WIPER BLADE	
2 X Rp. 30.000	Rp. 60.000
ITEMS : 2	
<b>SUBTOTAL</b>	<b>Rp. 60.000</b>
Nama : Irene Deandra	
Alamat : Jl. Babarsari 59 Yogyakarta	
TERIMA KASIH	

Gambar 0.109 Rancangan Antarmuka Nota Transaksi Bon Halaman pada gambar 4.29 merupakan nota transaksi bon yang dapat dicetak oleh Kasir. Di nota ini, terdapat informasi tanggal dan waktu transaksi, kode transaksi, serta nama Kasir. Lalu terdapat nama barang, kuantitas, subtotal, total, jumlah barang yang dibeli oleh Customer, serta informasi data customer.

## Antarmuka Pengelolaan Pemesanan Barang

Pemesanan Barang

http://www.sisp.net/displayPemesananBarang

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Transaksi** Profil

### Daftar Pemesanan Barang

[+ Tambah Pemesanan Barang](#)

Tanggal :

No.	Kode Pesan	Tanggal Pesan	Total	Vendor	Status	
1	CAB1/2014-10-30/1	30 Oct 2014	Rp. 270.000	HONDA	PROCESS	Lihat Detil
2	CAB1/2014-11-01/1	01 Nov 2014	Rp. 1.500.000	KIT	NOT COMPLETED	Lihat Detil

Gambar 0.110 Rancangan Antarmuka Pengelolaan Pemesanan Barang

Halaman pada gambar 4.30 merupakan halaman pengelolaan pemesanan barang yang bisa diakses oleh Kasir. Halaman ini menampilkan data pemesanan barang pada tanggal hari ini. Kasir dapat menambah, batal, batal transaksi, melihat info pemesanan barang, atau mencari data pemesanan barang. Untuk menambahkan data pemesanan barang, dapat dengan mengklik link Tambah Pemesanan Barang. Halaman Tambah Pemesanan Barang akan ditampilkan seperti gambar 4.31.

Gambar 0.111 Rancangan Antarmuka Pengelolaan Tambah Pemesanan Barang

Halaman pada gambar 4.31 merupakan halaman untuk menambahkan data pemesanan barang. Halaman ini menampilkan kode pemesanan dan tanggal pemesanan yang di generate secara otomatis. Terdapat *combo box* vendor yang harus diisi oleh Kasir. Jika data yang diinputkan benar, maka data pemesanan barang akan ditambahkan. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika data vendor belum terdapat di *combo box* tersebut, maka Kasir bias mengeklik link "disini" yang akan menuju ke halaman Tambah Vendor. Jika tombol Insert dipilih, akan dilanjutkan ke halaman pengelolaan Detil Pemesanan Barang.

## Antarmuka Pengelolaan Detil Transaksi Lunas

The screenshot shows a web browser window with the URL `http://www.ssp.net/displayInsertTranLunas`. The page header identifies the user as 'PT. XYZ Sistem Informasi Spare Parts' and includes a navigation menu with options: Home, Barang, Cabang, Pegawai, **Transaksi**, and Profil.

The main content area is divided into two sections:

- Pengelolaan Transaksi Lunas:** Displays transaction details:
  - Kode Transaksi: CAB1/2014-11-08/1
  - Tanggal Transaksi: 08 Nov 2014
- Data Barang:** A table showing the current transaction items:
 

Nama Barang	Harga	Kuantitas	Sub Total	Hapus
WIPER BLADE	Rp. 30.000	2	Rp. 60.000	x
<b>TOTAL</b>			Rp. 60.000	

Below the 'Data Barang' table are two buttons: 'Selesai Transaksi' and 'Batal Transaksi'. Under the 'Pengelolaan Transaksi Lunas' section, there is a 'Detil Barang' area with input fields for 'Barcode', 'Nama Barang', and 'Kuantitas', and a 'Tambah Barang' button.

Gambar 0.112 Rancangan Antarmuka Pengelolaan Detil Transaksi Lunas Halaman pada gambar 4.32 merupakan halaman untuk menambahkan detil transaksi lunas. Halaman ini menampilkan kode transaksi dan tanggal transaksi yang di generate secara otomatis. Lalu terdapat *input box* barcode, nama barang, dan kuantitas yang harus diisi oleh Kasir.

Jika tombol Tambah Barang dipilih, maka detil transaksi lunas akan ditambahkan dan akan ditampilkan di data barang yang ada di sebelah kanan halaman. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Untuk mengubah data detil transaksi lunas (kuantitas barang), dapat dengan mengisikan nama barang serta kuantitas, dan data barang akan terupdate. Untuk menghapus data detil transaksi lunas, dapat dengan mengeklik tanda 'X' yang ada pada data barang.

## Antarmuka Pengelolaan Detil Transaksi Bon

Transaksi Bon

http://www.ssp.net/display/insertTranBon

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Transaksi** Profil

### Pengelolaan Transaksi Bon

Kode Transaksi :  
CAB1/2014-11-08/1  
Tanggal Transaksi  
08 Nov 2014

### Data Barang

Nama Barang	Harga	Kuantitas	Sub Total	Hapus
WIPER BLADE	Rp. 30.000	2	Rp. 60.000	x
TOTAL			Rp. 60.000	

### Detil Barang

Barcode :

Nama Barang :

Kuantitas

Gambar 0.113 Rancangan Antarmuka Pengelolaan Detil Transaksi Bon Halaman pada gambar 4.33 merupakan halaman untuk menambahkan detil transaksi bon. Halaman ini menampilkan kode transaksi dan tanggal transaksi yang di generate secara otomatis. Lalu terdapat *input box* barcode, nama barang, dan kuantitas yang harus diisi oleh Kasir.

Jika tombol Tambah Barang dipilih, maka detil transaksi bon akan ditambahkan dan akan ditampilkan di data barang yang ada di sebelah kanan halaman. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Untuk mengubah data detil transaksi bon (kuantitas barang), dapat dengan mengisikan nama barang serta kuantitas, dan data barang akan terupdate. Untuk menghapus data detil transaksi bon, dapat dengan mengeklik tanda 'X' yang ada pada data barang.

## Antarmuka Pengelolaan Detil Pemesanan Barang

The screenshot shows a web browser window with the URL `http://www.sisp.net/display/insertDetilPemesananBarang`. The page header includes the logo for PT. XYZ (Sistem Informasi Spare Parts) and a navigation menu with options: Home, Barang, Cabang, Pegawai, Transaksi, and Profil. The main content area is divided into two sections:

- Pengelolaan Pemesanan Barang:** Displays order details:
  - Kode Pemesanan: CAB1/2014-11-08/1
  - Tanggal Pemesanan: 08 Nov 2014
  - Vendor: HONDA
- Data Barang:** A table showing the items in the order:
 

Nama Barang	Harga	Kuantitas	Sub Total	Hapus
WIPER BLADE	Rp. 30.000	2	Rp. 60.000	x
<b>TOTAL</b>			<b>Rp. 60.000</b>	

Below the 'Data Barang' table are two buttons: 'Selesai' and 'Batal'. In the 'Detil Barang' section, there are input fields for 'Nama Barang' and 'Kuantitas', and a 'Tambah Barang' button.

Gambar 0.114 Rancangan Antarmuka Pengelolaan Detil Pemesanan Barang

Halaman pada gambar 4.34 merupakan halaman untuk menambahkan detil pemesanan barang. Halaman ini menampilkan kode pemesanan, tanggal pemesanan yang di generate secara otomatis, dan nama vendor. Lalu terdapat *input box* nama barang dan kuantitas yang harus diisi oleh Kasir.

Jika tombol Tambah Barang dipilih, maka detil pemesanan barang akan ditambahkan dan akan ditampilkan di data barang yang ada di sebelah kanan halaman. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Untuk mengubah data detil pemesanan barang (kuantitas barang), dapat dengan mengisikan nama barang serta kuantitas, dan data barang akan terupdate. Untuk menghapus data detil pemesanan barang, dapat dengan mengeklik tanda 'X' yang ada pada data barang.

## Antarmuka Retur Transaksi Lunas

The screenshot shows a web browser window with the URL `http://www.sisp.net/infoTranLunas`. The page header includes the logo for PT. XYZ (Sistem Informasi Spare Parts) and a navigation menu with options: Home, Barang, Cabang, Pegawai, Transaksi, and Profil. The main content is divided into two columns:

- Info Transaksi Lunas:**
  - Kode Transaksi: CAB1/2014-11-08/1
  - Tanggal Transaksi: 08 Nov 2014
  - Total: Rp. 60.000,00
- Data Barang:**

Nama Barang	Harga	Kuantitas	Sub Total
Wiper Blade	Rp. 30.000	2	Rp. 60.000
<b>TOTAL</b>			<b>Rp. 60.000</b>
- Retur Barang:**
  - Nama Barang:
  - Kuantitas:
  - Keterangan Barang:
  -
- Data Retur Barang:**

Nama Barang	Harga	Kuantitas	Sub Total	Keterangan
Wiper Blade	Rp. 30.000	1	Rp. 30.000	RUSAK
<b>TOTAL</b>			<b>Rp. 30.000</b>	

At the bottom right of the 'Data Retur Barang' section, there is a  button.

Gambar 0.115 Rancangan Antarmuka Retur Transaksi Lunas

Halaman pada gambar 4.35 merupakan halaman untuk melakukan retur transaksi lunas. Halaman ini menampilkan informasi transaksi serta data barang yang dipilih. Lalu terdapat *input box* nama barang, kuantitas, serta keterangan barang yang harus diisi oleh Kasir.

Jika tombol Retur Barang dipilih, maka detail retur barang lunas akan ditambahkan dan akan ditampilkan di data retur barang yang ada di sebelah kanan halaman bagian bawah. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika tombol Selesai dipilih, maka akan kembali ke halaman yang menampilkan daftar transaksi lunas.

## Antarmuka Retur Transaksi Bon

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Transaksi** Profil

### Info Transaksi Bon

Nama Customer :  
Irene Deandra  
Kode Transaksi :  
CAB1/2014-11-08/1  
Tanggal Transaksi  
08 Nov 2014  
Total :  
Rp. 60.000,00

### Data Barang

Nama Barang	Harga	Kuantitas	Sub Total
Wiper Blade	Rp. 30.000	2	Rp. 60.000
TOTAL			Rp. 60.000

### Retur Barang

Nama Barang :

Kuantitas :

Keterangan Barang :  
RUSAK

### Data Retur Barang

Nama Barang	Harga	Kuantitas	Sub Total	Keterangan
Wiper Blade	Rp. 30.000	1	Rp. 30.000	RUSAK
TOTAL			Rp. 30.000	

Gambar 0.116 Rancangan Antarmuka Retur Transaksi Bon

Halaman pada gambar 4.36 merupakan halaman untuk melakukan retur transaksi bon. Halaman ini menampilkan informasi transaksi serta data barang yang dipilih. Lalu terdapat *input box* nama barang, kuantitas, serta keterangan barang yang harus diisi oleh Kasir.

Jika tombol Retur Barang dipilih, maka detail retur barang bon akan ditambahkan dan akan ditampilkan di data retur barang yang ada di sebelah kanan halaman bagian bawah. Sebaliknya, jika data yang diinputkan salah akan muncul pesan peringatan. Jika tombol Selesai dipilih, maka akan kembali ke halaman yang menampilkan daftar transaksi bon.

## Antarmuka Pengelolaan Penerimaan Barang

Penerimaan Barang

http://www.sisp.net/displayPenerimaanBarang

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai **Kelola** Profil

### Info Pemesanan Barang

Vendor :  
HONDA

Kode Pemesanan :  
CAB1/2014-10-30/1

Tanggal Pemesanan :  
30 Oct 2014

Total :  
Rp. 270.000,00

### Data Barang

No.	Nama Barang	Harga Barang	Kuantitas	Subtotal	Kuantitas Terima	Harga Jual
1	CYLINDER HEAD	Rp. 90.000	3	Rp. 270.000		
TOTAL				Rp. 270.000		

**Terima Barang**

Gambar 0.117 Rancangan Antarmuka Pengelolaan Penerimaan Barang Halaman pada gambar 4.37 merupakan halaman untuk melakukan penerimaan barang yang bisa diakses oleh Gudang. Halaman ini menampilkan informasi pemesanan barang yang sudah dipilih oleh Gudang dari halaman tampil pemesanan. Terdapat *input box* pada kolom kuantitas terima serta harga jual yang harus diisi oleh Gudang.

Jika tombol Terima Barang diklik, maka data akan ditambahkan ke table penerimaan barang dan status pemesanan barang akan terupdate secara otomatis. Jika barang yang diterima sudah lengkap, maka status akan berubah menjadi 'COMPLETED', sedangkan jika barang yang diterima belum lengkap, maka status akan berubah menjadi 'NOTCOMPLETED'. Setelah itu, akan menuju ke halaman yang menampilkan daftar pemesanan barang.

## Antarmuka Reporting

Report Umum

http://www.sisp.net/displayReportUmum

PT. XYZ  
Sistem Informasi Spare Parts

Home Barang Cabang Pegawai Transaksi Reporting Profil

### Reporting Penjualan Umum

Tanggal :  
2014/11/01 - 2014/11/01

Pilih Transaksi :

Transaksi Lunas  
 Transaksi Bon

Lihat Laporan

Gambar 0.118 Rancangan Antarmuka Reporting Halaman pada gambar 4.38 merupakan halaman untuk melakukan reporting yang bisa diakses oleh Manager. Terdapat *input box* tanggal serta *check box* jenis transaksi yang harus diisi oleh Manager.

Jika tombol Lihat Laporan diklik, maka akan ditampilkan report sesuai tanggal dan jenis transaksi yang sudah dipilih seperti pada gambar 4.39. Untuk laporan penjualan detail, laporan bon sama seperti gambar 4.39 hanya saja untuk laporan detail akan menampilkan nama barang beserta kuantitas yang telah dibeli oleh customer, sedangkan laporan daftar barang habis akan menampilkan data barang yang habis di setiap cabang. Untuk grafik laporan penjualan akan ditampilkan pada gambar 4.40.



PT. XYZ  
Jl. Babarsari no. 17 Yogyakarta  
Telp. (0274) 547812

## LAPORAN PENJUALAN UMUM

Tanggal 01 Nov 2014 - 30 Nov 2014

### TRANSAKSI LUNAS

Cabang Babarsari

No.	Kode Transaksi	Tanggal Transaksi	Subtotal
1	CAB1/2014-11-03/1	03 Nov 2014	Rp. 270.000
1	CAB1/2014-11-03/2	03 Nov 2014	Rp. 100.000
TOTAL			Rp. 370.000

### TRANSAKSI BON

Cabang Babarsari

No.	Kode Transaksi	Tanggal Transaksi	Subtotal
1	CAB1/2014-11-03/1	03 Nov 2014	Rp. 270.000
1	CAB1/2014-11-03/2	03 Nov 2014	Rp. 100.000
TOTAL			Rp. 370.000

Printed on : 08/11/2014 13:00 | PT. XYZ Yogyakarta Page 1/1

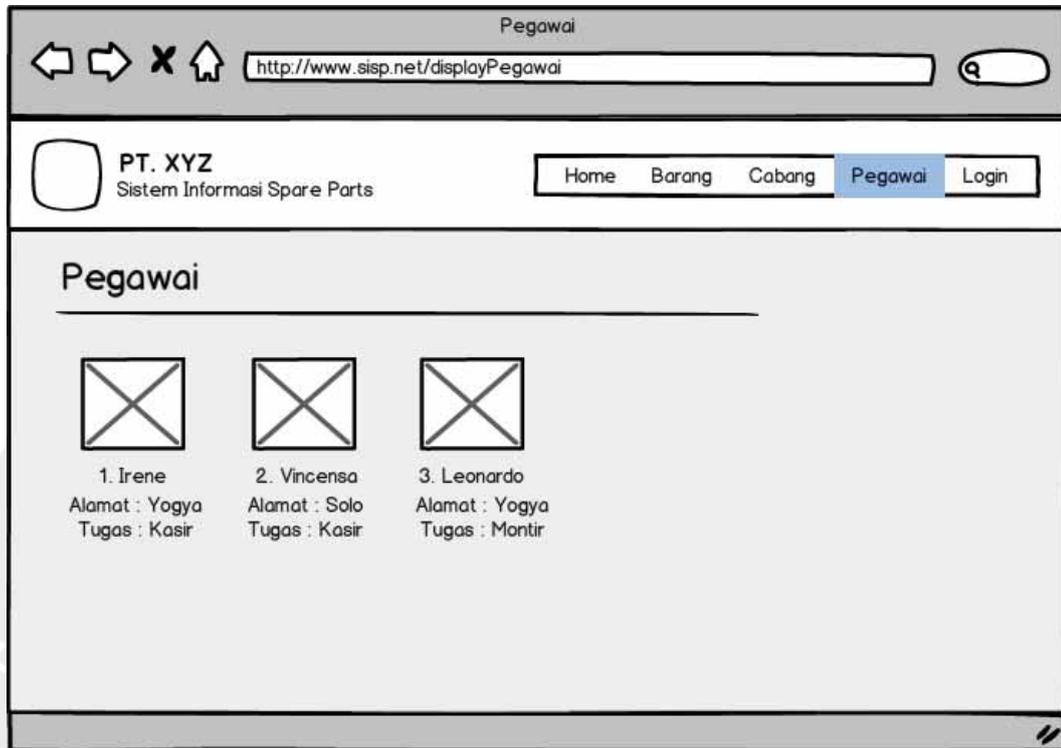
Gambar 0.119 Rancangan Antarmuka Reporting Umum  
Halaman pada gambar 4.39 merupakan halaman yang menampilkan laporan penjualan per cabang sesuai dengan rentang tanggal serta jenis transaksi yang dipilih oleh Manager.



Gambar 0.120 Rancangan Antarmuka Reporting Grafik Penjualan Halaman pada gambar 4.40 merupakan halaman yang grafik penjualan. Terdapat *input box* tanggal yang harus diisi oleh Manager.

Jika tombol Lihat Grafik diklik, maka akan muncul 2 grafik untuk Transaksi Lunas dan Transaksi Bon yang akan menunjukkan perbandingan jumlah penjualan di cabang-cabang PT. XYZ.

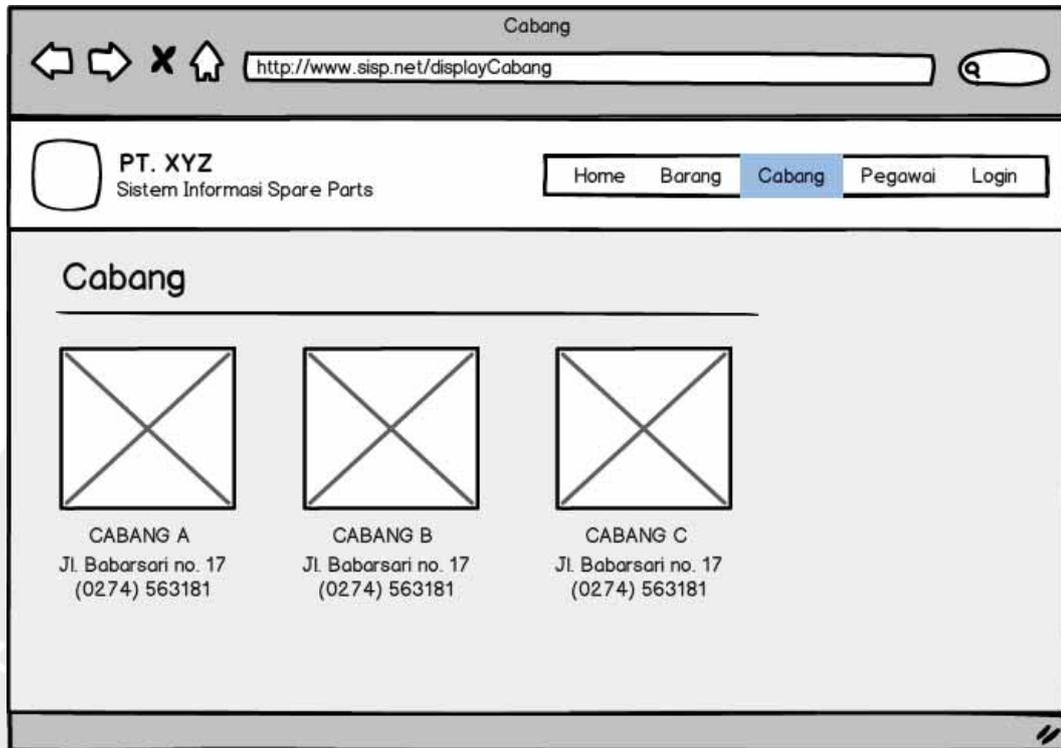
## Antarmuka Data Pegawai



Gambar 0.121 Rancangan Antarmuka Data Pegawai

Halaman pada gambar 4.41 merupakan halaman untuk menampilkan data pegawai yang bisa diakses oleh semua pengguna, yakni Administrator, Kasir, Gudang, dan Manager. Kasir, Gudang, dan Manager hanya dapat melihat data yang ada dan tidak dapat melakukan penambahan, pengubahan, atau penghapusan data. Sedangkan Administrator dapat menambah, mengubah, atau menghapus data pegawai.

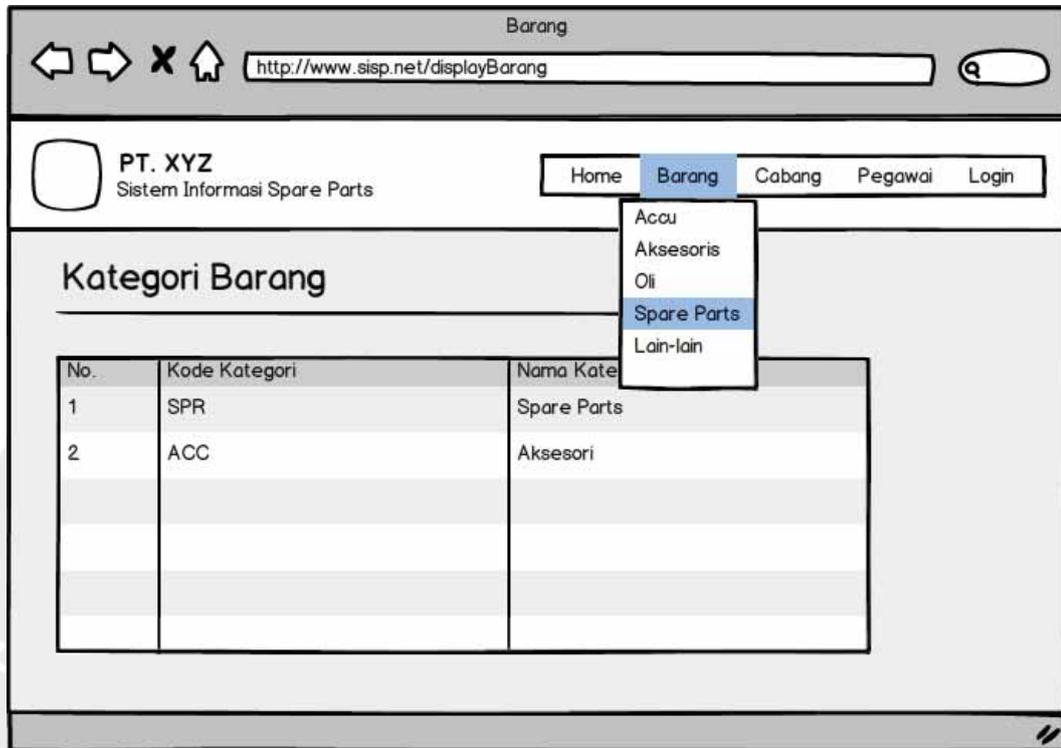
## Antarmuka Data Cabang



Gambar 0.122 Rancangan Antarmuka Data Cabang

Halaman pada gambar 4.42 merupakan halaman untuk menampilkan data cabang yang bisa diakses oleh semua pengguna, yakni Administrator, Kasir, Gudang, dan Manager. Kasir, Gudang, dan Manager hanya dapat melihat data yang ada dan tidak dapat melakukan penambahan, pengubahan, atau penghapusan data. Sedangkan Administrator dapat menambah, mengubah, atau menghapus data cabang.

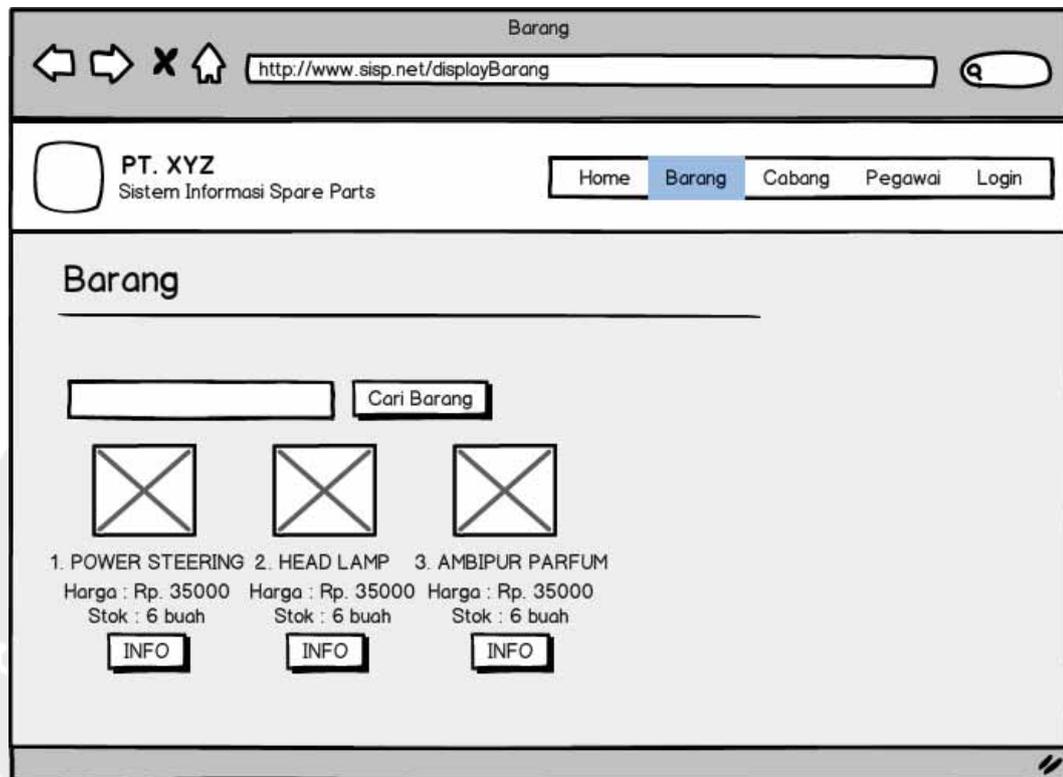
## Antarmuka Data Kategori Barang



Gambar 0.123 Rancangan Antarmuka Data Kategori Barang

Halaman pada gambar 4.43 merupakan halaman untuk menampilkan sub menu data kategori barang yang bisa diakses oleh semua pengguna, yakni Administrator, Kasir, Gudang, dan Manager. Kasir, Gudang, dan Manager hanya dapat melihat list kategori barang yang ada dan tidak dapat melakukan penambahan atau pengubahan data. Sedangkan Administrator dapat menambah atau mengubah data kategori barang.

## Antarmuka Data Barang

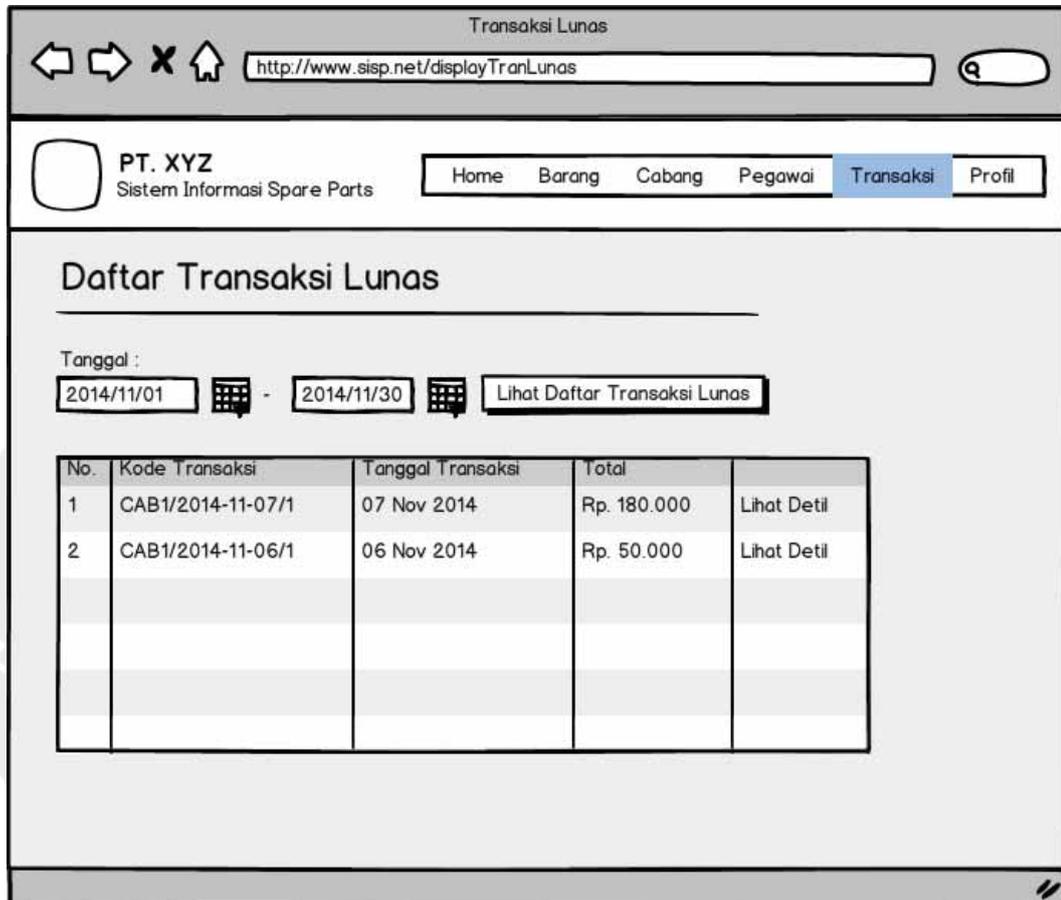


Gambar 0.124 Rancangan Antarmuka Data Barang

Halaman pada gambar 4.44 merupakan halaman untuk menampilkan data barang yang bisa diakses oleh semua pengguna, yakni Administrator, Kasir, Gudang, dan Manager. Kasir, Gudang, dan Manager dapat melihat serta mencari data barang yang ada dan tidak dapat melakukan penambahan, pengubahan, atau penghapusan data. Sedangkan Administrator dapat menambah, mengubah, atau menghapus data barang.

Jika menu Barang diklik, maka akan muncul semua barang. Jika kursor diarahkan ke menu Barang, akan muncul sub menu kategori barang. Jika salah satu kategori barang tersebut diklik, maka akan muncul data barang sesuai dengan kategori yang dipilih. Untuk mencari barang, dapat dengan mengisi nama barang yang ingin di cari di *input box* yang ada lalu tekan tombol Cari Barang. Jika tombol Info diklik, maka akan muncul informasi barang.

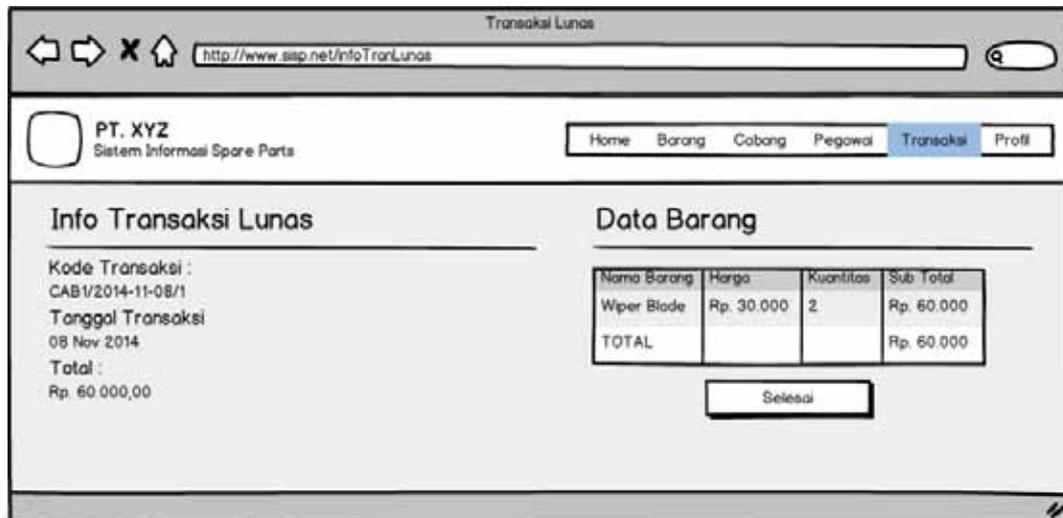
## Antarmuka Data Transaksi Lunas



Gambar 0.125 Rancangan Antarmuka Data Transaksi Lunas

Halaman pada gambar 4.45 merupakan halaman untuk menampilkan data transaksi lunas yang bisa diakses oleh Administrator, Kasir, dan Manager. Administrator dan Manager dapat melihat data, melihat informasi, serta mencari data transaksi lunas yang ada dan tidak dapat melakukan pengelolaan data. Sedangkan Administrator dapat melakukan pengelolaan data transaksi lunas.

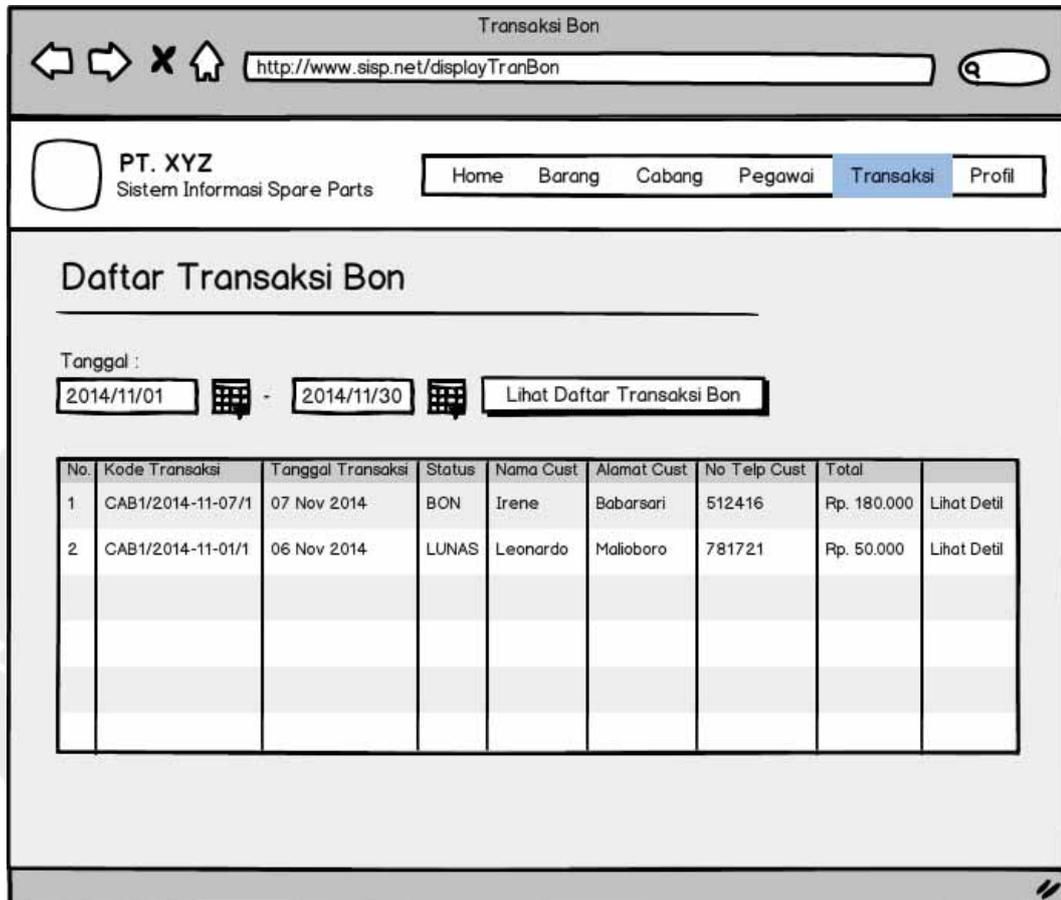
Halaman Info Transaksi Lunas akan ditampilkan seperti gambar 4.46. Untuk melihat info transaksi, dapat dengan mengeklik link Lihat Detil. Untuk mencari data transaksi lunas, dapat dengan mengisi tanggal di *input box* yang ada lalu tekan tombol Lihat Daftar Transaksi Lunas.



Gambar 0.126 Rancangan Antarmuka Info Transaksi Lunas

Halaman pada gambar 4.46 merupakan halaman yang menampilkan info transaksi lunas yang dipilih. Halaman ini menampilkan informasi kode, tanggal, serta total transaksi dan di bagian kanan halaman terdapat data barang. Jika tombol Selesai dipilih, akan kembali ke halaman yang menampilkan daftar transaksi lunas seperti gambar 4.45.

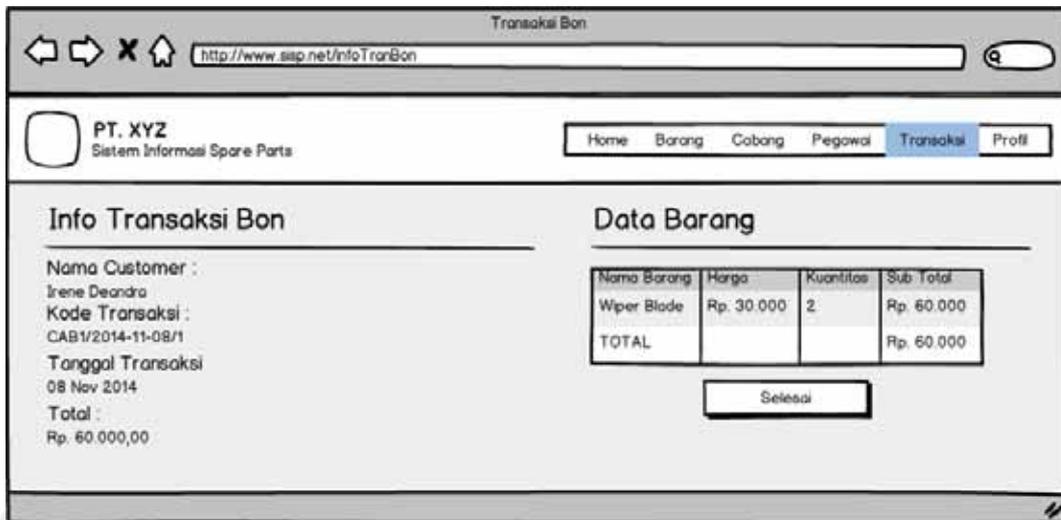
## Antarmuka Data Transaksi Bon



Gambar 0.127 Rancangan Antarmuka Data Transaksi Bon

Halaman pada gambar 4.47 merupakan halaman untuk menampilkan data transaksi bon yang bisa diakses oleh Administrator, Kasir, dan Manager. Administrator dan Manager dapat melihat data, melihat informasi, serta mencari data transaksi bon yang ada dan tidak dapat melakukan pengelolaan data. Sedangkan Administrator dapat melakukan pengelolaan data transaksi bon.

Halaman Info Transaksi Bon akan ditampilkan seperti gambar 4.48. Untuk melihat info transaksi, dapat dengan mengeklik link Lihat Detil. Untuk mencari data transaksi bon, dapat dengan mengisi tanggal di *input box* yang ada lalu tekan tombol Lihat Daftar Transaksi Bon.



Gambar 0.128 Rancangan Antarmuka Info Transaksi Bon Halaman pada gambar 4.48 merupakan halaman yang menampilkan info transaksi bon yang dipilih. Halaman ini menampilkan informasi customer, kode, tanggal, serta total transaksi dan di bagian kanan halaman terdapat data barang. Jika tombol Selesai dipilih, akan kembali ke halaman yang menampilkan daftar transaksi lunas seperti gambar 4.47.

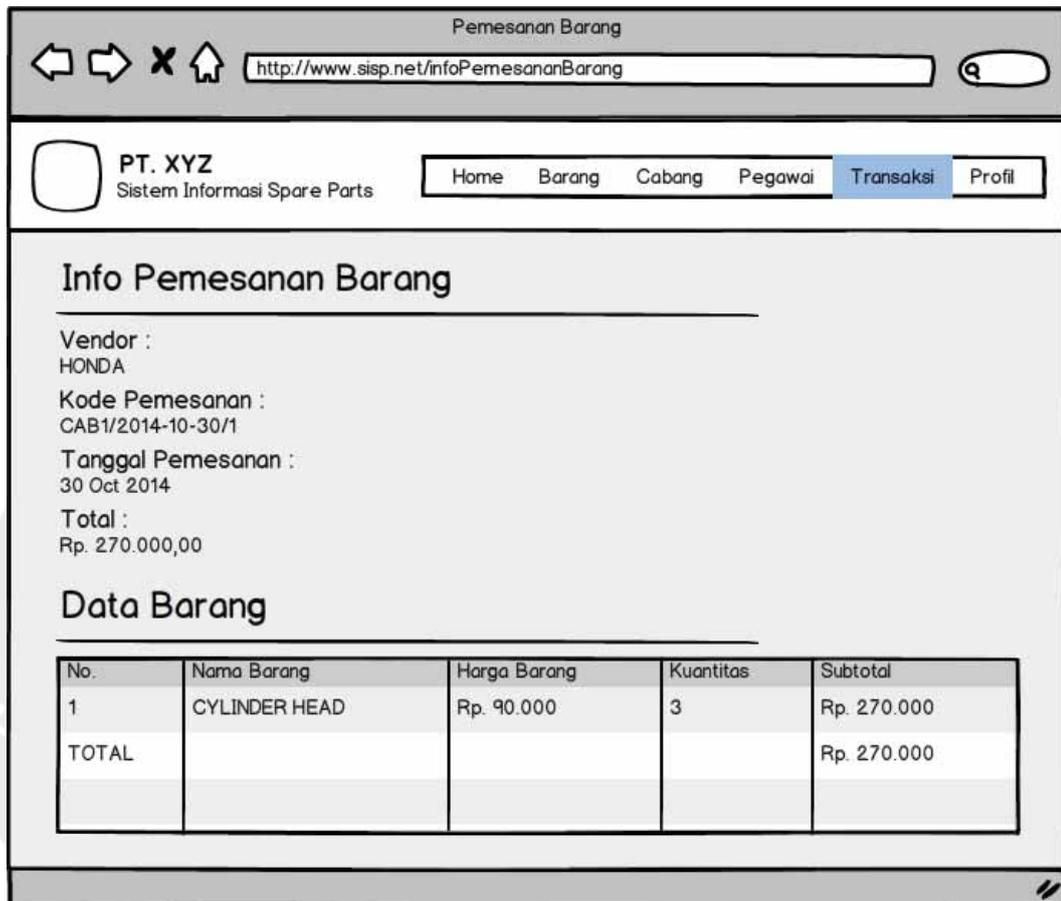
## Antarmuka Data Pemesanan Barang

No.	Kode Pesan	Tanggal Pesan	Total	Vendor	Status	
1	CAB1/2014-10-30/1	30 Oct 2014	Rp. 270.000	HONDA	PROCESS	Lihat Detil
2	CAB1/2014-11-01/1	01 Nov 2014	Rp. 1.500.000	KIT	NOT COMPLETED	Lihat Detil

Gambar 0.129 Rancangan Antarmuka Data Pemesanan Barang

Halaman pada gambar 4.49 merupakan halaman untuk menampilkan data pemesanan barang yang bisa diakses oleh Kasir, Gudang, dan Manager. Gudang dan Manager dapat melihat data, melihat informasi, serta mencari data pemesanan barang yang ada dan tidak dapat melakukan pengelolaan data. Sedangkan Kasir dapat melakukan pengelolaan data pemesanan barang.

Halaman Info Pemesanan Barang akan ditampilkan seperti gambar 4.50. Untuk melihat info pemesanan barang, dapat dengan mengklik link Lihat Detil. Untuk mencari data pemesanan barang, dapat dengan mengisi tanggal di *input box* yang ada lalu tekan tombol Lihat Daftar Pemesanan Barang.



Gambar 0.130 Rancangan Antarmuka Info Pemesanan Barang  
Halaman pada gambar 4.50 merupakan halaman yang menampilkan info pemesanan barang yang dipilih. Halaman ini menampilkan informasi vendor, kode, tanggal, serta total pemesanan. Lalu terdapat data barang yang dipesan.

## Antarmuka Data Penerimaan Barang

The screenshot shows a web browser window with the title 'Penerimaan Barang'. The address bar contains the URL 'http://www.sisp.net/displayDaftarPenerimaan'. The page header includes the company logo 'PT. XYZ Sistem Informasi Spare Parts' and a navigation menu with items: Home, Barang, Cabang, Pegawai, Transaksi (highlighted), Reporting, and Profil. The main content area is titled 'Daftar Penerimaan Barang' and displays data for 'Cabang Babarsari'. Below this, there is a section for 'Cabang Seturan' with the message '- Tidak terdapat data penerimaan barang -'.

No.	Kode Pesan	Nama Barang	Kuantitas Terima	Tanggal	Tanggal Terima	Status
1	CAB1/2014-11-07/1	HEAD LAMP FREED RH	2	25 Oct 2014	05 Nov 2014	NOTCOMPLETED
2	CAB1/2014-11-07/1	HEAD LAMP FREED LH	5	25 Oct 2014	05 Nov 2014	COMPLETED

Cabang Seturan  
- Tidak terdapat data penerimaan barang -

Gambar 0.131 Rancangan Antarmuka Data Penerimaan Barang Halaman pada gambar 4.51 merupakan halaman untuk menampilkan data penerimaan barang yang diterima semua cabang yang hanya bisa diakses oleh Manager.