

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan pembahasan pada bab sebelumnya, dapat ditarik kesimpulan dari Pembangunan Sistem informasi penentuan *Best Customer* dengan metode RFM:

1. Sistem informasi penentuan *Best Customer* dengan metode RFM berhasil dibangun untuk membantu proses bisnis di perusahaan, dalam hal ini SG Steel dan membantu manager dalam pengambilan keputusan untuk menentukan pelanggan terbaik, untuk diberikan penawaran khusus, seperti potongan harga dan lain-lain. Penawaran khusus tersebut, diberikan kepada pelanggan terbaik dengan cara dikirimkan email pada aplikasi SIPBC ini. Dengan dibuatnya aplikasi ini, dapat membantu proses bisnis pada SG Steel dan membantu meningkatkan hubungan antara pihak SG Steel dan pelanggannya.
2. Aplikasi yang telah dibangun ini dapat membantu proses bisnis perusahaan, khususnya SG Steel, serta dapat membantu proses pengambilan keputusan manager untuk menentukan pelanggan terbaik, hal ini dibuktikan dengan kurang lebih 70% responden dalam pengujian sistem mengatakan setuju.

6.2 Saran

Beberapa saran yang dapat diberikan dari proses analisa sampai pada pembuatan tugas akhir ini adalah sebagai berikut:

1. Perbaiki antarmuka sehingga lebih menarik.
2. Perbaiki interaksi antarmuka sistem dengan pengguna sehingga lebih *user friendly*.
3. Penambahan fungsi untuk menampilkan laporan dalam bentuk neraca saldo.
4. Penambahan fungsi untuk mengetahui arus kas.

Daftar Pustaka

- Anderson, Kristin, 2002. *Customer Relationship Management*. McGraw Hill Professional.
- Arini, Dwi, 2011. Sistem Informasi Akuntansi Penjualan dan Pembelian pada Toko Sumber Usaha. Naskah Publikasi, Jurusan Manajemen Informatika, STMIK AMIKOM Yogyakarta.
- Barnes, James G, (2003a). *Secrets Of Customer Relationship Management*. Penerbit Andi, Yogyakarta.
- Deck, S, 1999. "Mining Your Business," *Computerworld*, volume 33, number 20, pp. 94-98.
- Eko, Richardus, 2011. Metode RFM untuk Menentukan *Best Customer*. Penerbit Andi, Yogyakarta.
- Ghazali, M., dkk, 2008. *Konsep Sistem Informasi*. <http://ghanoz2480.files.wordpress.com/2008/07/ghanoz2480-konsep-sistem-informasi.pdf>.
- Hasanah, 2011. Sistem Informasi Penjualan dan Pembelian pada Toko Salam. Naskah Publikasi, Jurusan Manajemen Informatika, STMIK AMIKOM Yogyakarta.
- Jogiyanto HM, Prof.,Dr., MBA, Akt, 2005. *Analisis & Desain Sistem Informasi : Pendekatan terstruktur teori dan praktik aplikasi bisnis*. Andi Offset. Yogyakarta.
- Kalakota, R. And Robinson, 2001. *e-Business 2.0 : Roadmap for Success*. Addison-Wesley information technology series.
- O'Brien, James. A, 2005. *Introduction To Information System : Essential for The Business Enterprise*, 11th edition. McGraw Hill, New York.
- Olson, David; Yong Shi. 2009. *Introduction to Business Data Mining*. Mcgraw-Hill College.
- Strene, Jim, 2000. *Customer Service on The Internet : Building Relationship, Increasing Loyalty and Staying Competitive*, second edition. Wiley Computer Publishing.
- Sulistiyowati, 2009. Sistem Informasi Akuntansi Pembelian dan Penjualan Tunai pada Toko Besi "Cipta Putra" Sukoharjo. Naskah Publikasi, Jurusan Manajemen Informatika, STMIK AMIKOM Yogyakarta.
- Sutedjo, Budi dan Philip, John, (2003). *I-CRM : Membina Relasi Dengan Pelanggan Dot Com*. Penerbit Andi, Yogyakarta.
- Tsiptsis, Konstantinos; Antonios Chorianopoulos, 2009. *Data Mining Techniques in CRM*. John Wiley & Sons, Ltd.

Turban, Efraim, (2004). *Electronic Commerce A managerial Perspective*, Prentice Hall, New Jersey.

Yoeti, Oka A, (2000). *Customer Service Cara Efektif Memuaskan Pelanggan*, cetakan kedua, Pradnya Paramita, Jakarta.



SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SIPBC

(Sistem Informasi Penentuan Best Customer dengan Metode RFM)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Matius Andri / 100706301

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		SKPL-SIPBC		1/39
	Fakultas Teknologi Industri	Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperi ksa oleh								
Disetu jui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1	Pendahuluan	6
1.1	Tujuan.....	6
1.2	Lingkup Masalah.....	6
1.3	Definisi, Akronim dan Singkatan.....	7
1.4	Referensi.....	7
1.5	Deskripsi umum (Overview).....	8
2	Deskripsi Kebutuhan	8
2.1	Perspektif produk.....	8
2.2	Fungsi Produk.....	10
2.3	Karakteristik Pengguna.....	17
2.4	Batasan-batasan.....	17
2.5	Asumsi dan Ketergantungan.....	17
3	Kebutuhan khusus	17
3.1	Kebutuhan antarmuka eksternal.....	17
3.2	Kebutuhan fungsionalitas Perangkat Lunak....	19
4	Spesifikasi Rinci Kebutuhan	20
4.1	Spesifikasi Kebutuhan Fungsionalitas.....	20
5	ERD	39

Daftar Gambar

1	Arsitektur Perangkat Lunak SIPBC	9
2	Use Case Diagram	19



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIPBC (Sistem Informasi Penentuan *Best Customer* dengan Metode RFM) 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-SIPBC ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak SIPBC dikembangkan dengan tujuan untuk :

1. Menangani pengelolaan pencatatan pembelian.
2. Menangani pengelolaan transaksi penjualan.
3. Menangani pengelolaan data barang.
4. Menangani pengelolaan data *supplier*.
5. Menangani pengelolaan data *customer*.
6. Menangani pembuatan laporan penjualan dan pembelian.
7. Menangani analisis pelanggan terbaik.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIPBC-XXX	Kode yang merepresentasikan kebutuhan pada SIPBC (Sistem Informasi Penentuan <i>Best Customer</i> dengan Metode RFM) dimana XXX merupakan nomor fungsi produk.
SIPBC	Perangkat lunak sistem informasi pengelolaan berbasis desktop.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
SG Steel	SG Steel adalah sebuah toko yang memfokuskan pada penjualan alat-alat teknik, seperti b'esi, pipa, alat-alat keselamatan kerja, dan lain-lain.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Wiguna, Ellen Melissa, *Spesifikasi Kebutuhan Perangkat Lunak SPP (Sistem Penjualan Perabot)*, Universitas Atma Jaya Yogyakarta, 2012.
2. Andri, Matius, *Spesifikasi Perancangan Perangkat Lunak DRS (Delizio Resto System)*, Universitas Atma Jaya Yogyakarta, 2013.

1.5 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 SIPBC yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SIPBC tersebut.

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

2 Deskripsi Kebutuhan

2.1 Perspektif produk

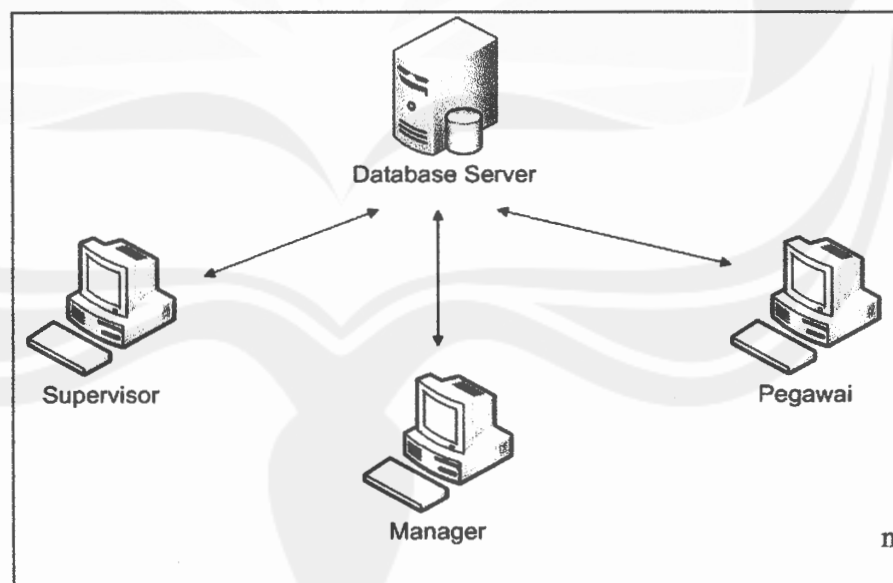
SIPBC merupakan perangkat lunak yang dikembangkan untuk membantu pengelolaan system informasi pada desktop. Sistem ini membantu "SG Steel" untuk membangun sebuah sistem informasi yang dapat meng-cover semua proses bisnis yang terjadi toko tersebut. Sistem ini juga dapat dipergunakan oleh SG Steel sendiri dan Sistem ini sendiri dibagi merupakan suatu aplikasi desktop.

Pada aplikasi ini, digunakan oleh pihak SG Steel. Aplikasi ini dapat menangani antara lain, pengelolaan

Program Studi Teknik Informatika	SKPL – SIPBC	8/ 39
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		

pegawai, pengelolaan barang, pengelolaan pencatatan pembelian, pengelolaan transaksi penjualan, dan khususnya pengelolaan *customer* yang memiliki tingkat royalti tertinggi (*best customer*) untuk mendapatkan perlakuan khusus dengan, yaitu mendapatkan potongan harga pada transaksi berikutnya dengan menggunakan metode RFM (*Recency, Frequency, Monetary*) untuk menentukan *score* tertinggi dalam pemilihan *Best Customer*, kemudian untuk melihat laporan transaksi penjualan pembelian secara periodik perbulan.

Perangkat lunak desktop SIPBC ini berjalan pada platform Windows XP/Vista/7 dan dibuat menggunakan bahasa pemrograman Microsoft Visual C#. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio 2008. Sedang untuk *database*, akan dipergunakan SQL Server.



Gambar 1. Arsitektur Perangkat lunak SIPBC

2.2 Fungsi Produk

Fungsi produk perangkat lunak SIPBC adalah sebagai berikut :

APLIKASI DESKTOP

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

Fungsi *Login* Merupakan fungsi yang digunakan oleh user untuk dapat masuk dalam sistem yang akan digunakan.

2. Fungsi *Pengelolaan Data Pegawai* (**SKPL-SIPBC-002**).

Fungsi pengelolaan data pegawai Merupakan fungsi yang digunakan Supervisor untuk mengelola data user dari karyawan yang meliputi user name, password, role user, nama asli, id pegawai.

Fungsi *Pengelolaan Data Pegawai* meliputi :

a. Fungsi *Add Data Pegawai* (**SKPL-SIPBC-002-01**).

Merupakan fungsi yang digunakan untuk menambahkan data Pegawai.

b. Fungsi *Edit Data Pegawai* (**SKPL-SIPBC-002-02**).

Merupakan fungsi yang digunakan untuk mengubah data Pegawai.

c. Fungsi *Display Data Pegawai* (**SKPL-SIPBC-002-03**).

Merupakan fungsi yang digunakan untuk menampilkan data Pegawai.

d. Fungsi *Search Data Pegawai* (**SKPL-SIPBC-002-04**).

Program Studi Teknik Informatika	SKPL - SIPBC	10/ 39
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 mensearch data Pegawai berdasarkan nama Pegawai atau role.

e. Fungsi *Reset Password* (**SKPL-SIPBC-002-05**)

Merupakan fungsi yang digunakan untuk mereset password berdasarkan ID Pegawai.

f. Fungsi *Delete Pegawai* (**SKPL-SIPBC-002-06**)

Merupakan fungsi yang digunakan untuk menghapus data pegawai berdasarkan ID Pegawai.

3. Fungsi *Ganti Password* (**SKPL-SIPBC-003**).

Fungsi *Ganti Password* Merupakan fungsi yang digunakan untuk mengubah password dari Pegawai.

4. Fungsi *Pengelolaan Barang* (**SKPL-SIPBC-004**).

Fungsi *Pengelolaan Barang* Merupakan fungsi yang digunakan supervisor untuk mengelola data barang.

Fungsi *Pengelolaan barang* meliputi:

a. Fungsi *Add Data Barang* (**SKPL-SIPBC-004-01**).

Merupakan fungsi yang digunakan untuk menambahkan data barang yang baru berupa Jenis barang , nama barang, dan harga.

b. Fungsi *Edit Data Barang* (**SKPL-SIPBC-004-02**).

Merupakan fungsi yang digunakan untuk mengubah data barang.

c. Fungsi *Delete Data Barang* (**SKPL-SIPBC-004-03**).

Program Studi Teknik Informatika	SKPL – SIPBC	11/ 39
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.

- d. Fungsi *Display Data Barang* (**SKPL-SIPBC-004-04**).

Merupakan fungsi yang digunakan untuk menampilkan data barang.

- e. Fungsi *Search Data Barang* (**SKPL-SIPBC-004-05**).

Merupakan fungsi yang digunakan untuk mencari data barang berdasarkan nama barang dan jenis barang.

5. Fungsi Pengelolaan Customer (**SKPL-SIPBC-005**).

Fungsi pengelolaan customer Merupakan fungsi yang digunakan supervisor untuk mengelola data Customer.

Fungsi Pengelolaan Customer Customer Customer :

- a. Fungsi *Add Data Customer* (**SKPL-SIPBC-005-01**).

Merupakan fungsi yang digunakan untuk menambahkan data Customer baru.

- b. Fungsi *Edit Data Customer* (**SKPL-SIPBC-005-02**).

Merupakan fungsi yang digunakan untuk mengubah data Customer.

- c. Fungsi *Display Data Customer* (**SKPL-SIPBC-005-04**).

Merupakan fungsi yang digunakan untuk menampilkan data Customer.

d. Fungsi *Search Data Customer* (**SKPL-SIPBC-005-05**).

Merupakan fungsi yang digunakan untuk mencari data Customer berdasarkan nama Customer.

6. Fungsi Pengelolaan Suplier (**SKPL-SIPBC-006**).

Fungsi pengelolaan suplier Merupakan fungsi yang digunakan supervisor untuk mengelola data suplier

Fungsi Pengelolaan suplier meliputi :

a. Fungsi *Add Data Suplier* (**SKPL-SIPBC-006-01**).

Merupakan fungsi yang digunakan untuk menambahkan data Suplier baru.

b. Fungsi *Edit Data Suplier* (**SKPL-SIPBC-006-02**).

Merupakan fungsi yang digunakan untuk mengubah data Suplier.

c. Fungsi *Display Data Suplier* (**SKPL-SIPBC-006-04**).

Merupakan fungsi yang digunakan untuk menampilkan data Suplier.

d. Fungsi *Search Data Suplier* (**SKPL-SIPBC-006-05**).

Merupakan fungsi yang digunakan untuk mencari data Suplier berdasarkan nama Suplier.

7. Fungsi *Pencatatan Transaksi Pembelian* (**SKPL-SIPBC-007**).

Fungsi pencatatan transaksi pembelian merupakan fungsi yang digunakan oleh Pegawai untuk mencatat dan mengelola pembelian dari SG Steel ke sistem.

Fungsi *Pencatatan Transaksi Pembelian* meliputi:

a. Fungsi *Add Data Penjualan* (**SKPL-SIPBC-007-01**).

Merupakan fungsi yang digunakan untuk menambah data pembelian.

b. Fungsi *Display Data Pembelian* (SKPL-SIPBC-007-02).

Merupakan fungsi yang digunakan untuk menampilkan data beserta status pembelian.

c. Fungsi *search data Pembelian* (SKPL-SIPBC-007-03).

Merupakan fungsi yang digunakan untuk mensearch data pembelian berdasarkan ID transaksi.

8. Fungsi *Pengelolaan Transaksi Penjualan* (**SKPL-SIPBC-008**).

Fungsi pengelolaan transaksi penjualan Merupakan fungsi yg digunakan Pegawai untuk melakukan transaksi penjualan. Fungsi pengelolaan transaksi penjualan meliputi:

a. Fungsi *Add Transaksi* (**SKPL-SIPBC-008-01**).

Merupakan fungsi yang digunakan untuk menambahkan data transaksi yang baru.

b. Fungsi *Display Data Penjualan* (**SKPL-SIPBC-008-02**).

Merupakan fungsi yang digunakan untuk menampilkan data beserta status penjualan.

c. Fungsi *search data Penjualan* (**SKPL-SIPBC-008-03**).

Merupakan fungsi yang digunakan untuk mensearch data penjualan berdasarkan ID transaksi penjualan.

d. Fungsi *Cetak Nota* (**SKPL-SIPBC-008-04**).

Merupakan fungsi yang digunakan untuk mencetak nota pembayaran.

9. Fungsi Laporan Transaksi (**SKPL-SIPBC-009**).

Fungsi Laporan Transaksi Merupakan fungsi yang dipergunakan oleh Supervisor untuk melihat laporan data transaksi penjualan pembelian

Fungsi laporan transaksi meliputi :

a. Fungsi Tampil Laporan Transaksi Penjualan (**SKPL-SIPBC-009-01**).

Merupakan fungsi yang digunakan untuk menampilkan laporan penjualan secara umum dan keseluruhan.

b. Fungsi Tampil Laporan Transaksi Pembelian (**SKPL-SIPBC-009-02**).

Merupakan fungsi yang digunakan untuk menampilkan laporan pembelian secara umum dan keseluruhan.

10. Fungsi Analisis Pelanggan Terbaik (**SKPL-SIPBC-0010**).

Fungsi Analisis Pelanggan Terbaik Merupakan fungsi yang dipergunakan oleh Supervisor untuk menghitung skor tertinggi dari analisis RFM untuk menentukan "Pelanggan Terbaik".

Fungsi Analisis Pelanggan Terbaik meliputi :

a. Fungsi Generate Pelanggan Terbaik (**SKPL-SIPBC-0010-01**).

Merupakan fungsi yang digunakan untuk menghitung skor tertinggi untuk mendapatkan pelanggan terbaik dengan menggunakan metode RFM, kemudian menampilkan data pelanggan terbaik tersebut.

b. Fungsi Kirim Email ke Pelanggan Terbaik (**SKPL-SIPBC-0010-02**).

Merupakan fungsi yang digunakan untuk mengirimkan Email ke Pelanggan Terbaik yang telah dihitung pada fungsi Generate Pelanggan Terbaik.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SIPBC adalah sebagai berikut :

1. Memahami pengoperasian Microsoft Windows.
2. Memahami penggunaan aplikasi SIPBC.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak SIPBC tersebut adalah :

1. Kebijakan Umum
Berpedoman pada tujuan dari pengembangan perangkat lunak SIPBC.
2. Keterbatasan perangkat keras
Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat desktop yang menggunakan sistem operasi Windows XP/Vista/7.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

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

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form.

Program Studi Teknik Informatika	SKPL – SIPBC	17/ 39
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		

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SIPBC adalah:

1. Perangkat dektop.
2. Perangkat Database Server

3.1.3 Antarmuka perangkat lunak

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

1. Nama : SQL Server 2005
Sumber : Microsoft

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

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

Sebagai sistem operasi untuk perangkat dektop.

3. Nama : Crystal Report
Sumber : Microsoft

Sebagai aplikasi untuk mencetak report/resi.

4. Nama : .NET Framework 2.0
Sumber : Microsoft.

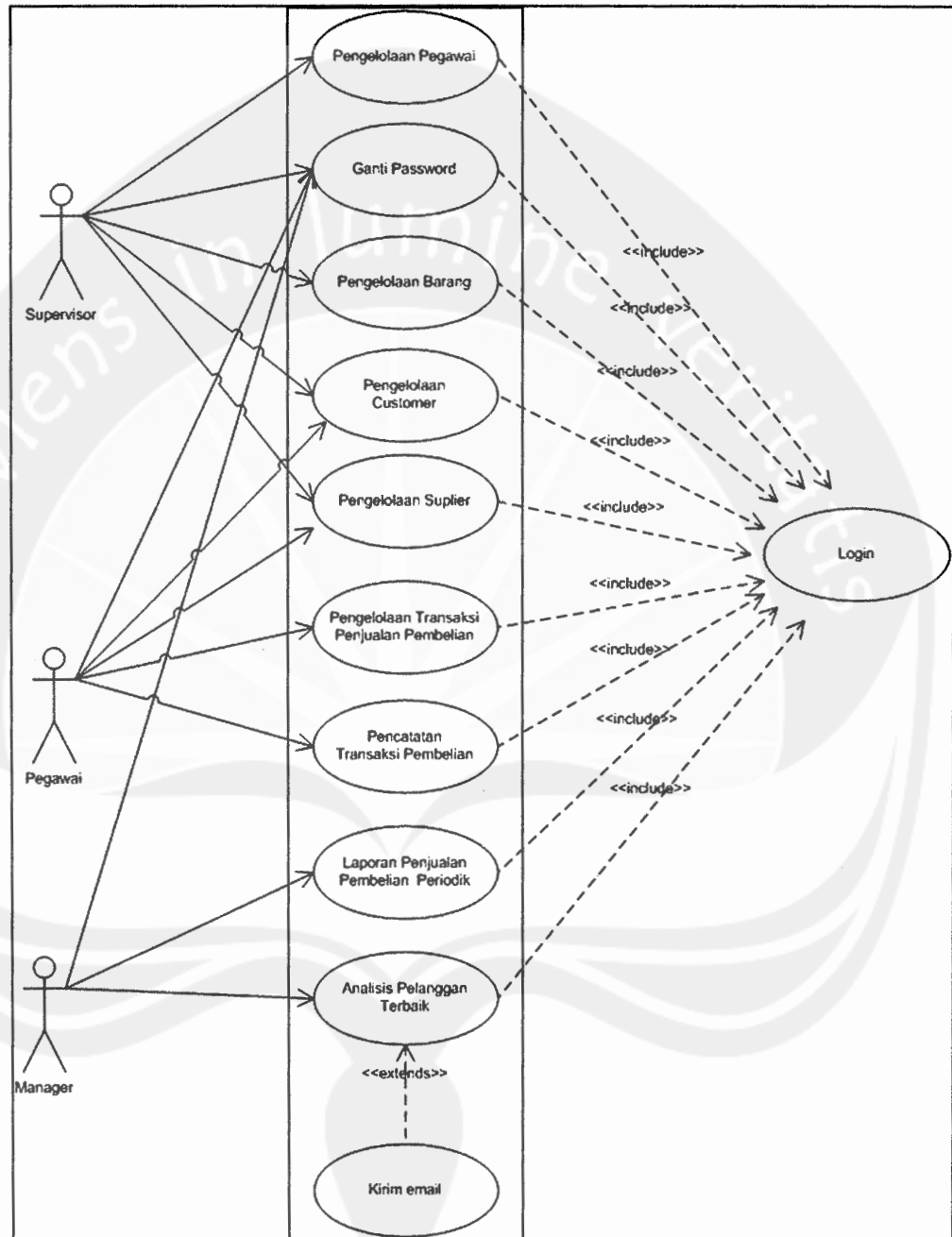
Sebagai aplikasi framework untuk menjalankan aplikasi SIPBC dalam sistem dektop.

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIPBC menggunakan protokol TCP/IP.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram SIPBC

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification : Login

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu email dari user dan password yang berupa rangkaian karakter.

2. Primary Actor

1. Supervisor
2. Pegawai
3. Manager

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 memasukkan id dan password
4. Sistem memeriksa id dan password yang diinputkan aktor
E-1 Password atau id user tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case ini selesai

5. Alternative Flow

none

6. Error Flow

E-1 Password atau id user tidak sesuai

1. Sistem menampilkan peringatan bahwa id user atau password tidak sesuai

2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

none

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

4.1.2 Use case Spesification : Ganti Password

1. Brief Description

Use Case ini digunakan oleh aktor untuk merubah password dari ID yang dimiliki.

2. Primary Actor

1. Supervisor
2. Pegawai
3. Manager

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan ganti password.
2. Sistem menampilkan antarmuka untuk mengedit password dengan memasukkan pasword lama terlebih dahulu, kemudian password yang baru dan konfirmasi password baru.
3. Aktor memasukkan password lama dan password baru dan konfirmasi password baru.
4. Aktor meminta sistem untuk menyimpan password baru dari aktor.
5. Sistem menyimpan password baru dari aktor ke dalam database.
6. Use Case ini selesai.

Program Studi Teknik Informatika	SKPL – SIPBC	21/ 39
----------------------------------	--------------	--------

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

none

6. Error Flow

none

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah masuk ke system.

5. PostConditions

1. Data password user dalam database telah terupdate.

4.1.3 Use case Spesification : Pengelolaan data pegawai

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data profile dari pegawai. Aktor dapat melakukan add data pegawai, edit data pegawai, display data pegawai, delete data pegawai, search data pegawai (bedasarkan id pegawai, nama pegawai, atau role), atau reset password.

2. Primary Actor

1. Supervisor

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pegawai.
2. Sistem memberikan pilihan untuk melakukan melakukan add data pegawai, edit data pegawai, display data pegawai, hapus data pegawai, search data pegawai (bedasarkan id

user, nama user, atau role pegawai), atau reset password pegawai.

3. Aktor memilih untuk melakukan add data pegawai

A-1 Aktor memilih untuk melakukan edit data pegawai

A-2 Aktor memilih untuk melakukan display data pegawai

A-3 Aktor memilih untuk melakukan delete data pegawai

A-4 Aktor memilih untuk melakukan search data pegawai

A-5 Aktor memilih untuk melakukan reset password pegawai

4. Aktor menginputkan data pegawai

5. Aktor meminta sistem untuk menyimpan data pegawai yang telah diinputkan.

6. Sistem mengecek 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 edit data pegawai

1. Sistem menampilkan data atau profile pegawai

2. Aktor mengedit data pegawai yang sudah ditampilkan

3. Aktor 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 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 display data pegawai

1. Sistem menampilkan data atau profile pegawai

2. Berlanjut ke Basic Flow langkah ke 8

A-3 Aktor memilih untuk melakukan delete data user

1. Sistem menampilkan data atau profile pegawai yang ingin dihapus

2. Aktor memilih untuk mengdelete data pegawai

3. Sistem menyimpan data pegawai yang ada ke database

4. Berlanjut ke Basic Flow langkah 8

A-4 Aktor memilih untuk melakukan search data pegawai

1. Aktor mengetik data pegawai yang ingin dicari berdasarkan id pegawai, nama pegawai atau role.

2. Sistem menampilkan data user yang dicari actor

3. Berlanjut ke Basic Flow langkah ke 8

A-5 Aktor memilih untuk melakukan reset password pegawai

1. Aktor mengetik id pegawai yang akan direset passwordnya.
2. Sistem melakukan reset password berdasarkan id yang diinputkan, dengan password default yaitu "userpass".
3. Berlanjut ke Basic Flow langkah 8.

6. Error Flow

E-1 Data user yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow Langkah ke 4

E-2 Data user yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
2. Kembali ke Alternative Flow A-1 Langkah ke 2

7. PreConditions

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

8. PostConditions

1. Data user di database telah terupdate

4.1.4 Use case Spesification : Pengelolaan Data Barang

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data barang. Aktor dapat melakukan tambah data barang, edit data barang, display data barang,

Program Studi Teknik Informatika	SKPL – SIPBC	25/ 39
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		

delete data barang atau cari data barang (bedasarkan nama barang dan jenis barang).

2. Primary Actor

1. Supervisor

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data barang.
2. Sistem memberikan pilihan untuk melakukan add data barang, edit data barang, display data barang, delete data barang atau cari data barang (bedasarkan nama barang).
3. Aktor memilih untuk melakukan add data barang
 - A-1 Aktor memilih untuk melakukan edit data barang
 - A-2 Aktor memilih untuk melakukan display data barang
 - A-3 Aktor memilih untuk melakukan delete data barang
 - A-4 Aktor memilih untuk melakukan search data barang
4. Aktor menginputkan data barang
5. Aktor meminta sistem untuk menyimpan data barang yang telah diinputkan
6. Sistem mengecek 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

Program Studi Teknik Informatika	SKPL – SIPBC	26/ 39
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 edit data barang

1. Sistem menampilkan data atau profile barang
2. Aktor mengedit data barang yang sudah ditampilkan
3. Aktor 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 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 display data barang

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

A-3 Aktor memilih untuk melakukan delete data barang

1. Sistem menampilkan data barang yang ingin dihapus
2. Aktor memilih untuk delete data barang
3. Sistem menyimpan data barang yang ada ke database
4. Berlanjut ke Basic Flow langkah 8

A-4 Aktor memilih untuk melakukan search data barang

1. Aktor mengetik data barang yang ingin dicari berdasarkan nama barang
2. Sistem menampilkan data barang yang dicari actor

3. Berlanjut ke Basic Flow langkah ke 8

6. Error Flow

E-1 Data barang yang diinputkan aktor salah

a. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah

b. Kembali ke Basic Flow Langkah ke 4

E-2 Data barang yang diinputkan aktor salah

a. Sistem memberikan pesan peringatan bahwa data yang diubah salah

b. Kembali ke Alternative Flow A-1 Langkah ke 2

7. PreConditions

1. Use Case Login telah dilakukan

2. Aktor telah memasuki sistem

8. PostConditions

1. Data barang di database telah terupdate

4.1.5 Use case Spesification : Pengelolaan Data Customer

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data customer. Aktor dapat melakukan add data customer, edit data customer, display data customer atau search data customer (bedasarkan nama customer).

2. Primary Actor

1. Supervisor

2. Pegawai

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data customer.
2. Sistem memberikan pilihan untuk melakukan add data customer, edit data customer, display data customer, delete data customer atau cari data customer (bedasarkan nama customer).
3. Aktor memilih untuk melakukan add data customer
 - A-1 Aktor memilih untuk melakukan edit data customer
 - A-2 Aktor memilih untuk melakukan display data customer
 - A-3 Aktor memilih untuk melakukan search data customer
4. Aktor menginputkan data customer
5. Aktor meminta sistem untuk menyimpan data customer yang telah diinputkan
6. Sistem mengecek 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 edit data customer
1. Sistem menampilkan data atau profile customer
 2. Aktor mengedit data customer yang sudah ditampilkan

3. Aktor 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 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 display data customer
 1. Sistem menampilkan data customer
 2. Berlanjut ke Basic Flow langkah ke 8
- A-3 Aktor memilih untuk melakukan search data customer
 1. Aktor mengetik data customer yang ingin dicari berdasarkan nama customer
 2. Sistem menampilkan data customer yang dicari aktor
 3. Berlanjut ke Basic Flow langkah ke 8

6. Error Flow

- E-1 Data customer yang diinputkan aktor salah
 1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
 2. Kembali ke Basic Flow Langkah ke 4
- E-1 Data customer yang diinputkan aktor salah
 1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
 2. Kembali ke Alternatif Flow A-1 langkah ke 2

7. PreConditions

Program Studi Teknik Informatika	SKPL – SIPBC	30/ 39
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 customer di database telah terupdate

4.1.6 Use case Spesification : Pengelolaan Data Suplier

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data suplier. Aktor dapat melakukan tambah data suplier, edit data suplier, display data suplier atau cari data suplier (bedasarkan nama suplier).

2. Primary Actor

1. Supervisor
2. Pegawai

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data suplier.
2. Sistem memberikan pilihan untuk melakukan add data suplier, edit data suplier, display data suplier atau cari data suplier (bedasarkan nama suplier).
3. Aktor memilih untuk melakukan add data suplier
 - A-1 Aktor memilih untuk melakukan edit data suplier
 - A-2 Aktor memilih untuk melakukan display data suplier

A-3 Aktor memilih untuk melakukan search data supplier

4. Aktor menginputkan data supplier
5. Aktor meminta sistem untuk menyimpan data supplier yang telah diinputkan
6. Sistem mengecek data supplier yang telah diinputkan

E-1 Data supplier yang diinputkan aktor salah

7. Sistem menyimpan data supplier ke database
8. Use Case selesai

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data supplier

1. Sistem menampilkan data atau profile supplier
2. Aktor mengedit data supplier yang sudah ditampilkan
3. Aktor meminta sistem untuk menyimpan data supplier yang telah diubah
4. Sistem melakukan pengecekan terhadap data supplier yang telah diubah

E-2 Data supplier yang telah diubah salah

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

A-2 Aktor memilih untuk melakukan display data supplier

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

A-3 Aktor memilih untuk melakukan delete data barang

1. Sistem menampilkan data barang yang ingin dihapus
2. Aktor memilih untuk delete data supplier
3. Sistem menyimpan data supplier yang ada ke database
4. Berlanjut ke Basic Flow langkah 8

A-4 Aktor memilih untuk melakukan search data supplier

1. Aktor mengetik data supplier yang ingin dicari berdasarkan nama supplier
2. Sistem menampilkan data supplier yang dicari actor
3. Berlanjut ke Basic Flow langkah ke 8

6. Error Flow

E-1 Data supplier yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow Langkah ke 4

E-2 Data supplier yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
2. Kembali ke Alternative Flow A-1 Langkah ke 2

7. PreConditions

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

8. PostConditions

1. Data supplier di database telah terupdate

4.1.7 Use case Spesification : Pengelolaan Pencatatan Transaksi Pembelian

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola pencatatan data transaksi pembelian. Aktor dapat melakukan Tambah data transaksi pembelian.

2. Primary Actor

1. Pegawai

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan pencatatan data transaksi pembelian.
2. Aktor melakukan tambah transaksi pembelian
3. Aktor menginputkan data pesanan
4. Sistem mengecek data pembelian yang telah diinputkan.

E-1 Data pembelian yang diinputkan aktor salah

5. Sistem akan menyimpan data transaksi pembelian yang baru kedalam database

5. Alternative Flow

none

6. Error Flow

E-1 Data pembelian yang diinputkan aktor salah

- a. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
- b. Kembali ke Basic Flow Langkah ke 2

7. PreConditions

1. Use Case Login telah dilakukan

Program Studi Teknik Informatika	SKPL – SIPBC	34/ 39
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 Transaksi pembelian telah disimpan dalam database

4.1.8 Use case Spesification : Pengelolaan Data Transaksi Penjualan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data transaksi penjualan. Aktor dapat melakukan Tambah transaksi penjualan dan cetak nota pembayaran.

2. Primary Actor

1. Pegawai

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data transaksi penjualan.
2. Sistem memberikan pilihan untuk melakukan Tambah transaksi penjualan.
3. Aktor memilih untuk melakukan tambah transaksi penjualan
4. Aktor menginputkan data pesanan
5. Sistem mengecek data penjualan yang telah diinputkan.

E-1 Data penjualan yang diinputkan aktor salah

6. Sistem akan menyimpan data transaksi penjualan yang baru kedalam database

Program Studi Teknik Informatika	SKPL – SIPBC	35/ 39
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. Sistem menampilkan detail transaksi penjualan customer dalam bentuk invoice
8. Aktor memilih untuk mencetak nota
9. Use case selesai

5. Alternative Flow

none

6. Error Flow

E-1 Data pesanan yang diinputkan aktor salah

- a. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
- b. Kembali ke Basic Flow Langkah ke 2

7. PreConditions

1. Use Case Login telah dilakukan
2. Aktor telah memasuki system

8. PostConditions

1. Data Transaksi penjualan telah disimpan dalam database

4.1.9 Use case Spesification : Laporan Transaksi

1. Brief Description

Use Case ini digunakan oleh aktor untuk melihat laporan transaksi penjualan atau pembelian.

2. Primary Actor

Manager

3. Supporting Actor

none

4. Basic Flow

1. Aktor memilih untuk melihat laporan pembelian.

A-1 Aktor memilih untuk melihat laporan penjualan

Program Studi Teknik Informatika	SKPL – SIPBC	36/ 39
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 tanggal pembuatan laporan dari awal transaksi sampai akhir transaksi
3. Sistem menampilkan laporan pembelian.
4. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melihat laporan penjualan :

1. Aktor memilih tanggal pembuatan laporan dari awal transaksi sampai akhir transaksi
2. Sistem menampilkan laporan penjualan periodik.
3. Use case selesai.

6. Error Flow

none

7. PreConditions

1. Aktor telah memasuki sistem.
2. Use Case login telah dilakukan (Manager).

8. PostConditions

1. Aktor mengetahui laporan pembelian.

4.1.10 Use case Spesification : Analisis Pelanggan Terbaik

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengidentifikasi pelanggan yang terbaik dari data transaksi penjualan yang diolah dengan menggunakan metode RFM. Aktor dapat melakukan generate skor tertinggi (untuk mendapatkan pelanggan yang terbaik) dan tampil pelanggan terbaik.

2. Primary Actor

1. Manager

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan generate skor tertinggi (untuk mendapatkan pelanggan yang terbaik).
A-1 Aktor memilih untuk kirim email ke pelanggan terbaik
2. Sistem menampilkan antarmuka generate skor tertinggi.
3. Aktor memilih tombol generate skor tertinggi.
4. Sistem menampilkan profile dari pelanggan terbaik dengan nilai skornya.
5. Use case selesai

5. Alternative Flow

A-1 Aktor memilih untuk kirim email ke pelanggan terbaik :

1. Sistem menampilkan antarmuka kirim email
2. Aktor memilih customer mana yang ingin dikirimkan email dan menginputkan bagian email.
3. Aktor menekan tombol "send".
4. Email terkirim ke customer yang dipilih
5. Use case selesai

6. Error Flow

none

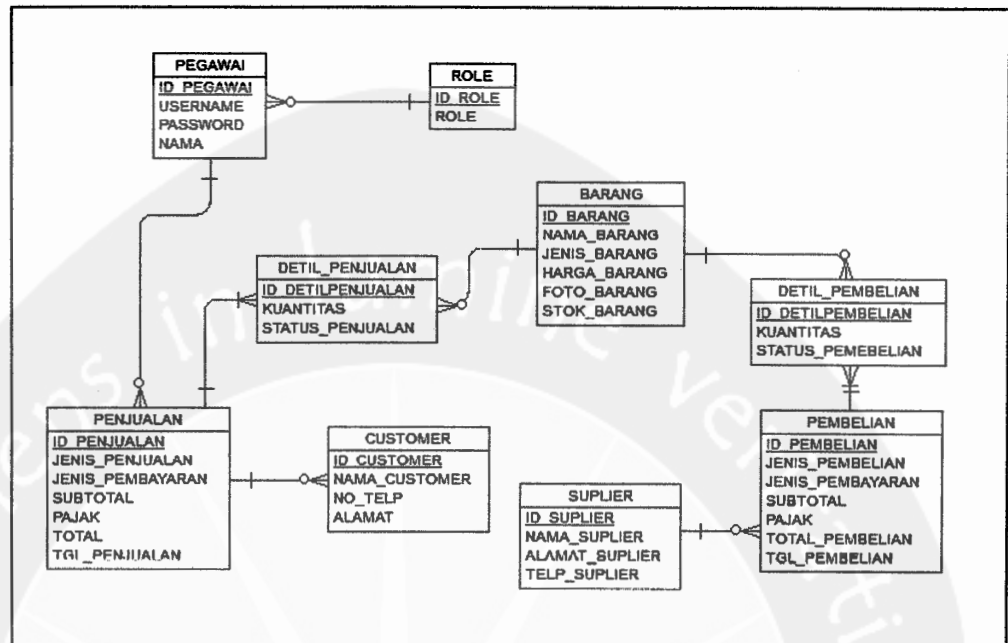
7. PreConditions

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

8. PostConditions

1. Aktor mengetahui pelanggan terbaik.

5. ERD



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SIPBC

(Sistem Informasi Penentuan *Best Customer*
dengan Metode RFM)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Matius Andri / 6301

Program Studi Teknik Informatika
Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL-SIPBC		1/56
		Revisi		

Program Studi Teknik Informatika	DPPL – SIPBC	1/ 56
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		

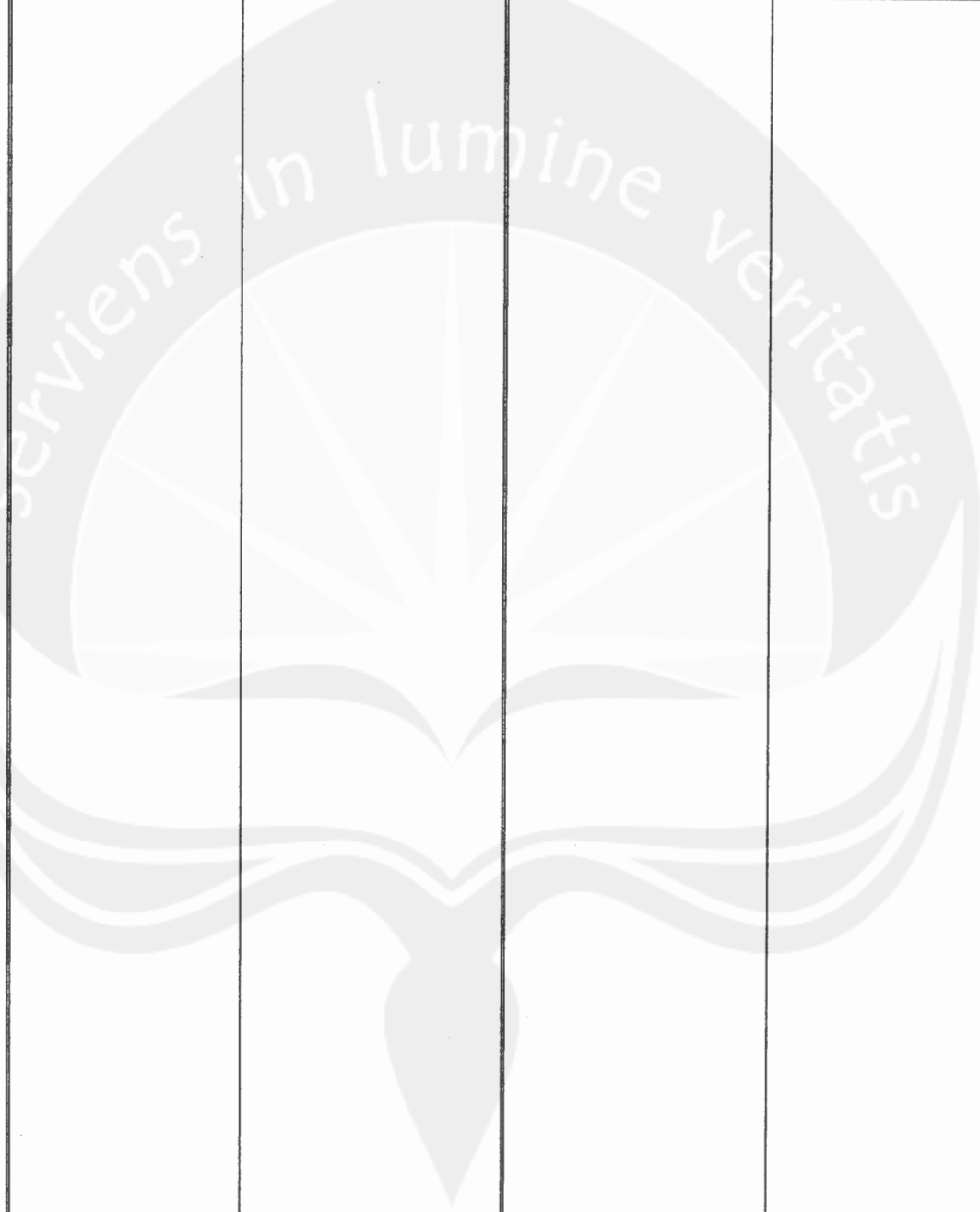
DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



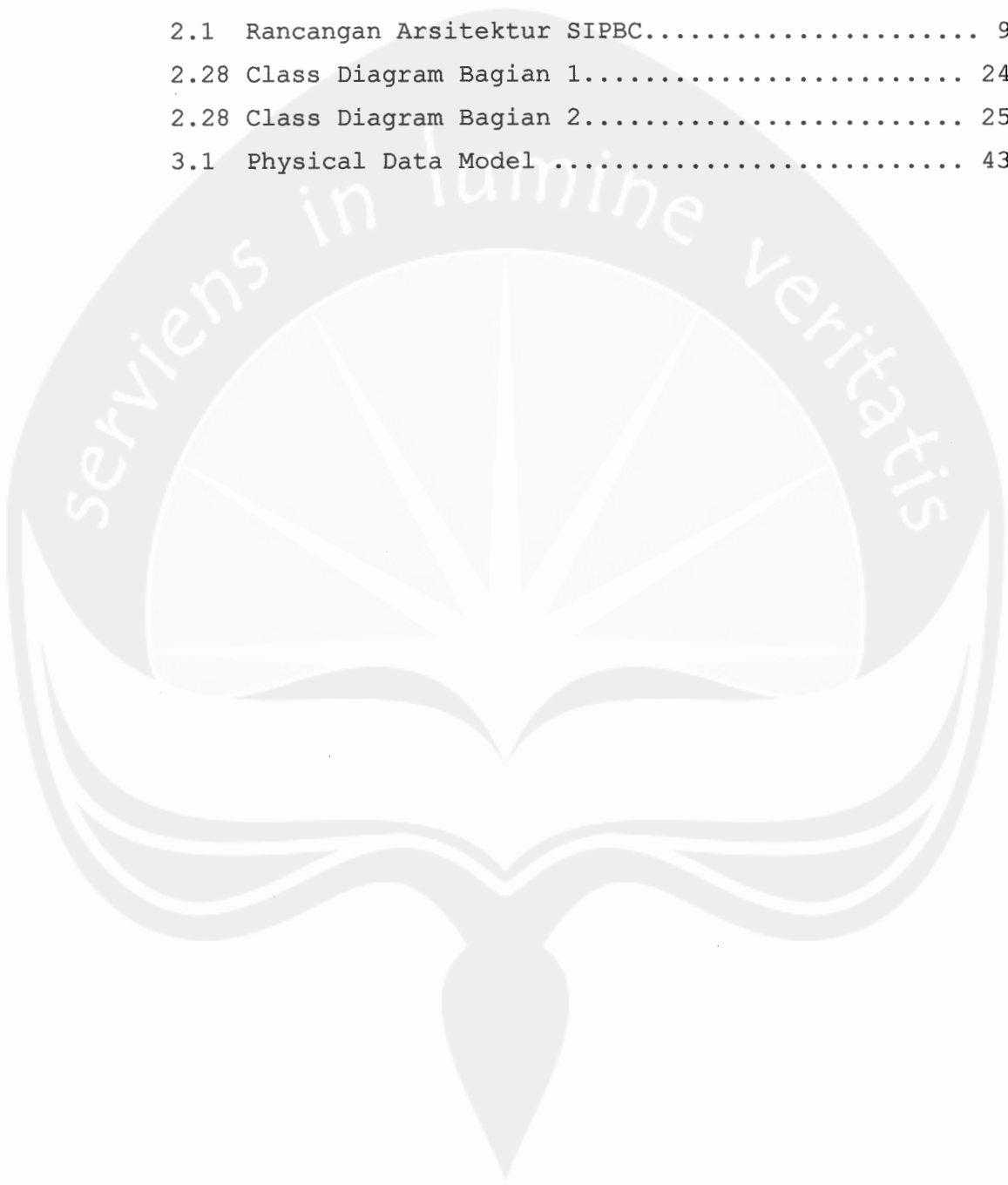
Daftar Isi

1.	Pendahuluan.....	7
1.1.	Tujuan	7
1.2.	Ruang Lingkup	7
1.3.	Definisi dan Akronim	7
1.4.	Referensi	8
2.	Perancangan Sistem	9
2.1.	Perancangan Arsitektur	9
2.2.	Perancangan Rinci	10
2.2.1.	Sequence Diagram	10
2.2.1.1	Autentikasi.....	10
2.2.1.1.1.	Login.....	10
2.2.1.1.2.	Pengelolaan Pegawai.....	11
2.2.1.2.1.	Add Data Pegawai.....	11
2.2.1.2.2.	Edit Data Pegawai.....	11
2.2.1.2.3.	Delete Data Pegawai.....	12
2.2.1.2.4.	Display Data Pegawai.....	12
2.2.1.2.5.	Reset Password Pegawai.....	13
2.2.1.3.	Ganti Password.....	13
2.2.1.4.	Pengelolaan Barang.....	14
2.2.1.4.1.	Add Data Barang.....	14
2.2.1.4.2.	Edit Data Barang.....	14
2.2.1.4.3.	Delete Data Barang.....	15
2.2.1.4.4.	Display Data Barang.....	15
2.2.1.5.	Pengelolaan Customer.....	16
2.2.1.5.1.	Add Data Customer.....	16
2.2.1.5.2.	Edit Data Customer.....	16
2.2.1.5.3.	Delete Data Customer.....	17
2.2.1.5.4.	Display Data Customer.....	17
2.2.1.6.	Pengelolaan Suplier.....	18
2.2.1.6.1.	Add Data Suplier.....	18
2.2.1.6.2.	Edit Data Suplier.....	18
2.2.1.6.3.	Delete Data Suplier.....	19

2.2.1.6.4. Display Data Suplier.....	19
2.2.1.7. Pengelolaan Transaksi Penjualan.....	20
2.2.1.7.1. Add Data Penjualan.....	20
2.2.1.7.2. Cetak Nota Penjualan.....	20
2.2.1.8. Pengelolaan Pencatatan Pembelian.....	21
2.2.1.8.1. Add Data Pembelian.....	21
2.2.1.9. Pengelolaan Laporan Transaksi.....	21
2.2.1.9.1. Cetak Laporan Transaksi Penjualan.....	21
2.2.1.9.2. Cetak Laporan Transaksi Pembelian.....	22
2.2.1.10. Analisis Pelanggan Terbaik.....	22
2.2.1.10.1. Generate Pelanggan Terbaik.....	22
2.2.1.10.2. Kirim Email untuk Best Customer.....	23
2.2.2. Class Diagram.....	24
2.2.3. Deskripsi Kelas.....	25
3. Perancangan Data	40
3.1. Dekomposisi Data	40
3.2. Physical Data Model	43
4. Deskripsi Perancangan Antarmuka	44

Daftar Gambar

2.1 Rancangan Arsitektur SIPBC.....	9
2.28 Class Diagram Bagian 1.....	24
2.28 Class Diagram Bagian 2.....	25
3.1 Physical Data Model	43



1 Pendahuluan

1.1 Tujuan

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

1.2 Ruang Lingkup

Perangkat Lunak SIPBC dikembangkan dengan tujuan untuk :

1. Menangani pengelolaan pencatatan pembelian.
2. Menangani pengelolaan transaksi penjualan.
3. Menangani pengelolaan data barang.
4. Menangani pengelolaan data *supplier*.
5. Menangani pengelolaan data customer.
6. Menangani pembuatan laporan penjualan dan pembelian.
7. Menangani analisis pelanggan terbaik.

Aplikasi ini dapat berjalan pada semua perangkat *desktop*.

1.3 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 akan dikembangkan.

SIPBC	Perangkat lunak sistem informasi pengelolaan berbasis desktop.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
SG Steel	SG Steel adalah sebuah toko yang memfokuskan pada penjualan alat-alat teknik, seperti pipa, besi, alat-alat keselamatan kerja, dan lain-lain.
Best Customer SG Steel	Konsumen SG Steel yang mendapatkan skor tertinggi setelah dihitung dengan menggunakan analisis RFM (<i>Recency, Frequency, Monetary</i>) dan akan mendapatkan perlakuan khusus seperti pemberian potongan harga, setelah.
RFM (<i>Recency, Frequency, Monetary</i>)	Merupakan metode yang digunakan untuk menganalisis data transaksi untuk mendapatkan pelanggan terbaik.

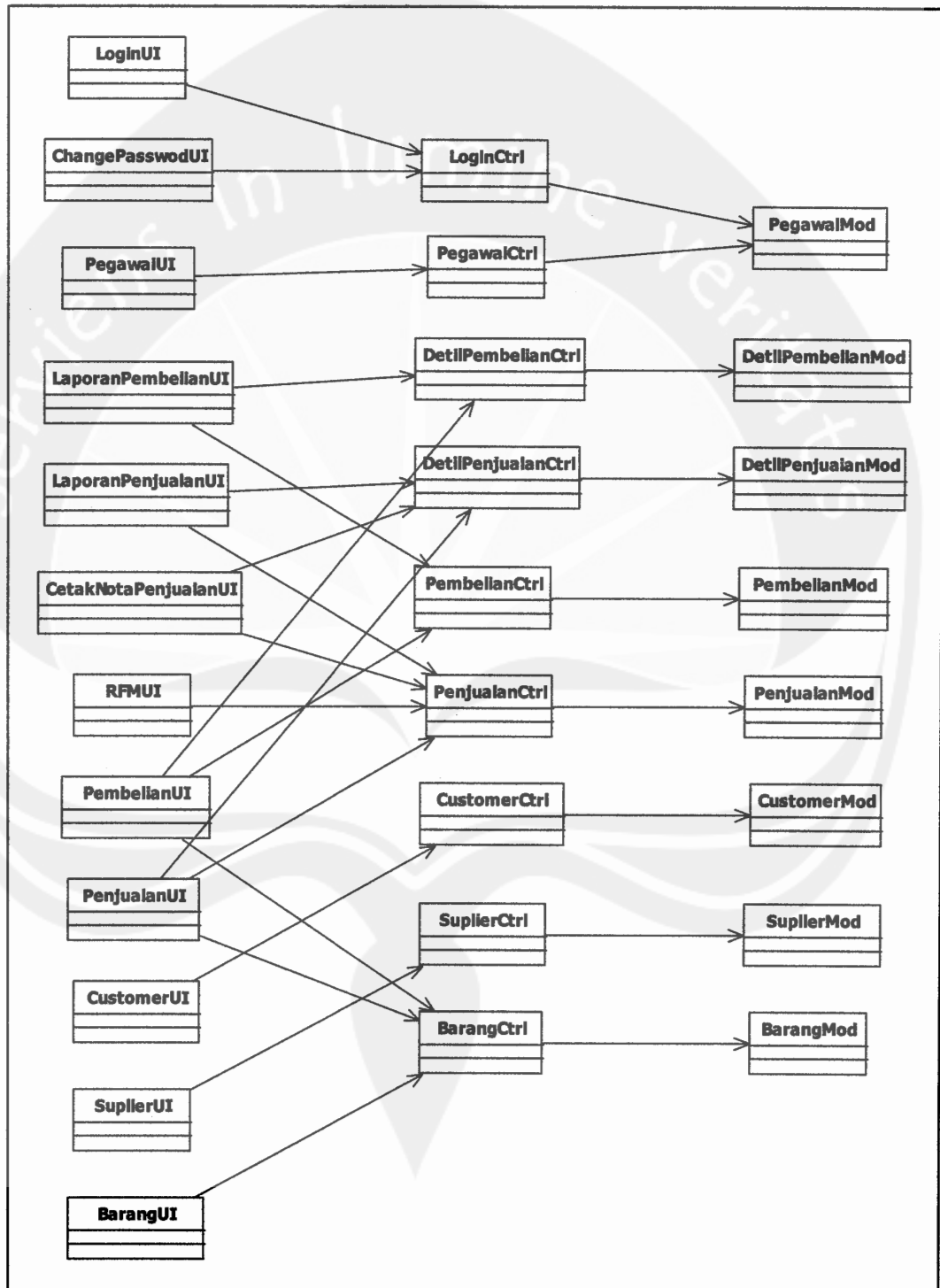
1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Wiguna, Ellen Melissa, *Deskripsi Perancangan Perangkat Lunak SIMARAPA-BB (Sistem Manajemen Rantai Pasok Bahan Bangunan)*, Universitas Atma Jaya Yogyakarta, 2012.
2. Andri, Matius, *Spesifikasi Perancangan Perangkat Lunak SIPBC (Sistem Informasi Penentuan Best Customer dengan Metode RFM)*, Universitas Atma Jaya Yogyakarta, 2014.

2 Analisis Model

2.1 Perancangan Arsitektur



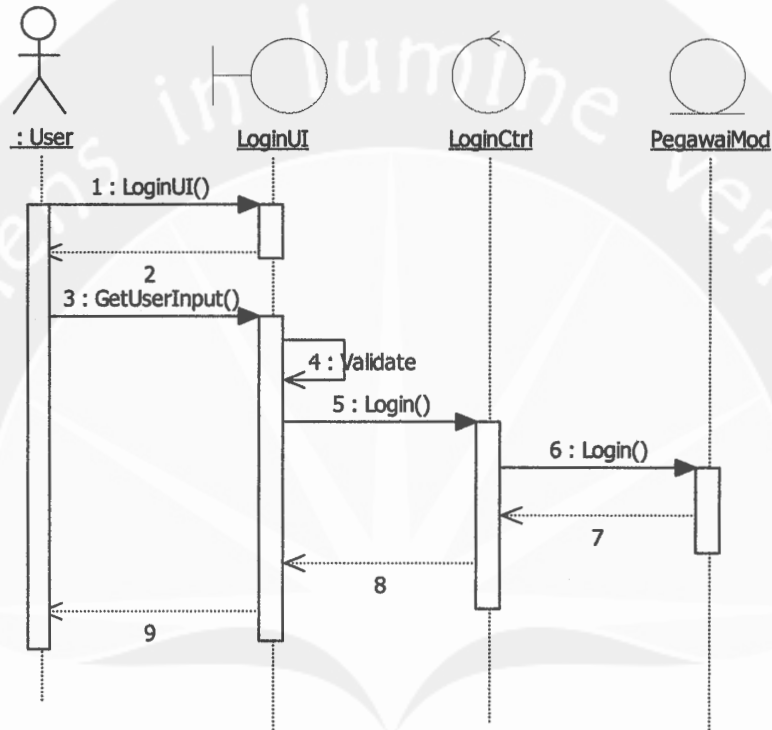
Gambar 2.1 Arsitektur Aplikasi SIPBC

2.2 Perancangan Rinci

2.2.1 Sequence Diagram

2.2.1.1 Autentikasi

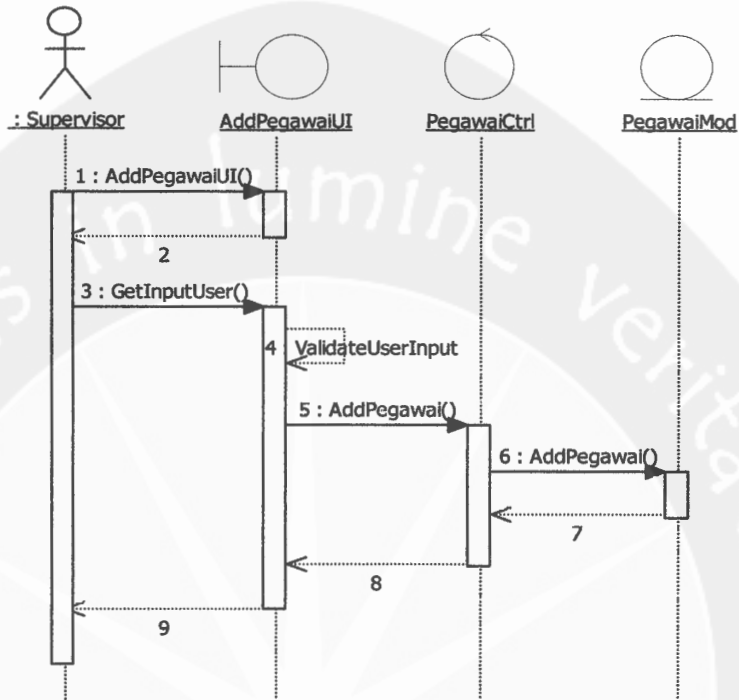
2.2.1.1.1 Login



Gambar 2.2 Sequence Diagram : Login

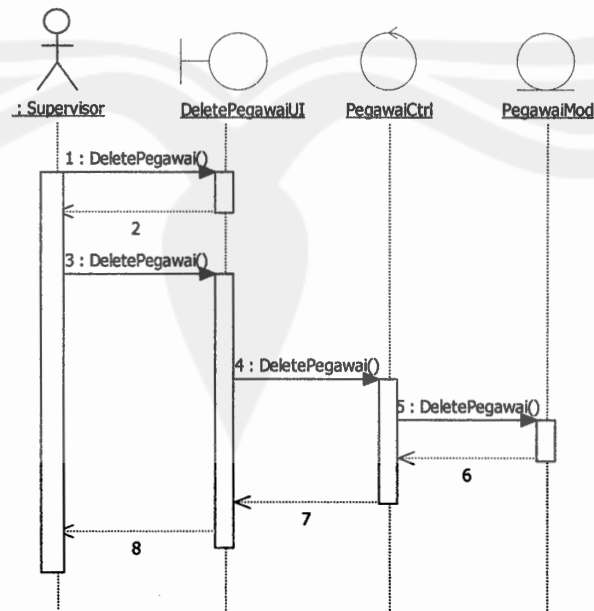
2.2.1.2 Pengelolaan Pegawai

2.2.1.2.1 Add Data Pegawai



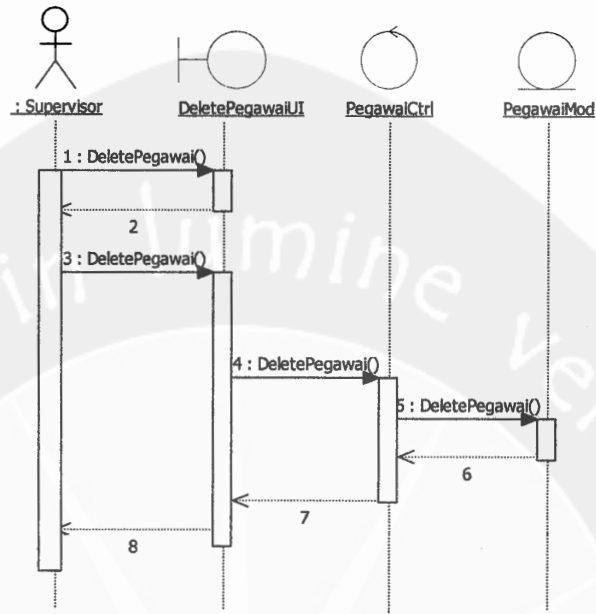
Gambar 2.3 Sequence Diagram : Add Pegawai

2.2.1.2.2 Edit Data Pegawai



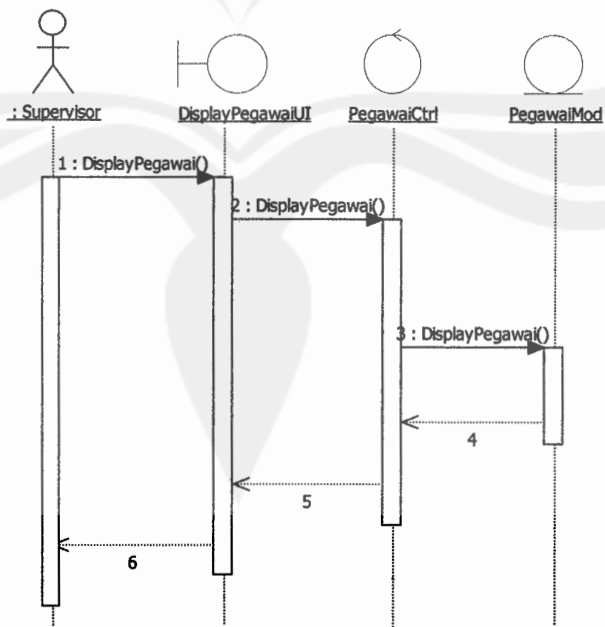
Gambar 2.4 Sequence Diagram : Edit Data Pegawai

2.2.1.2.3 Delete Data Pegawai



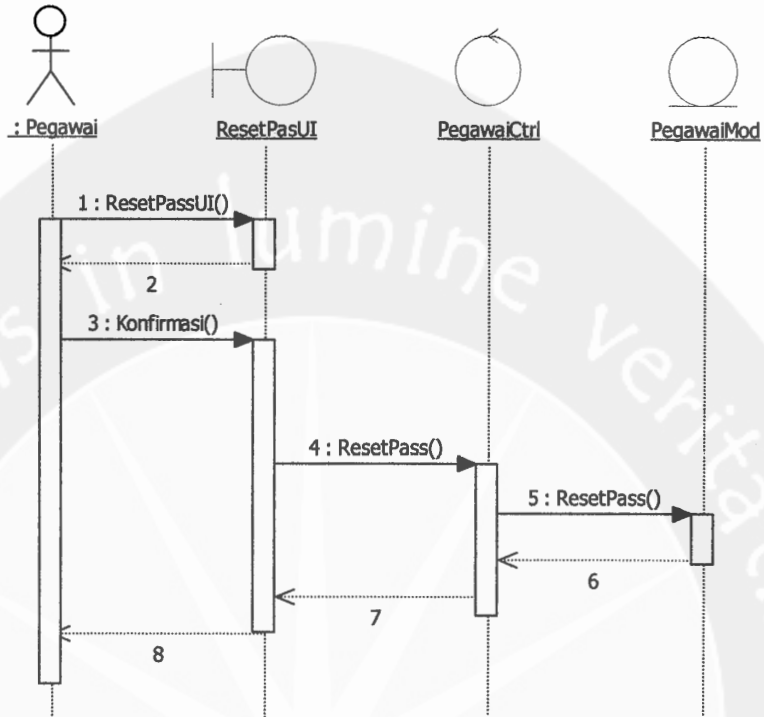
Gambar 2.5 Sequence Diagram : Delete Data Pegawai

2.2.1.2.4 Display Data Pegawai



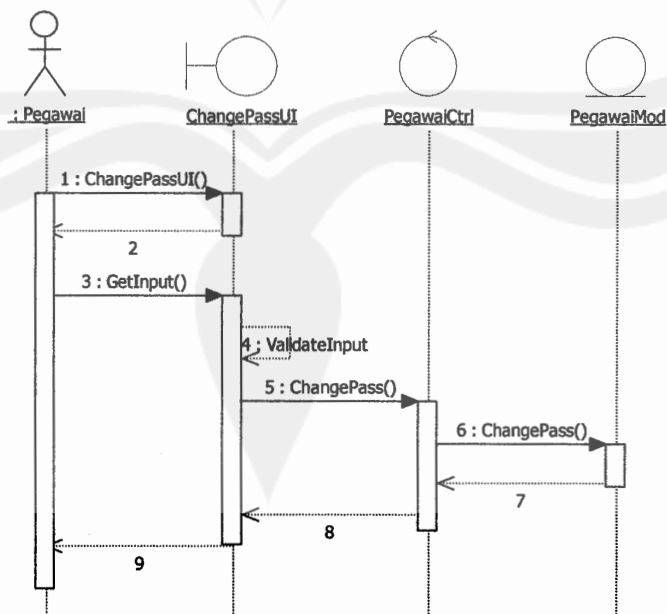
Gambar 2.6 Sequence Diagram : Display Data Pegawai

2.2.1.2.5 Reset Password Pegawai



Gambar 2.7 Sequence Diagram : Reset Password Pegawai

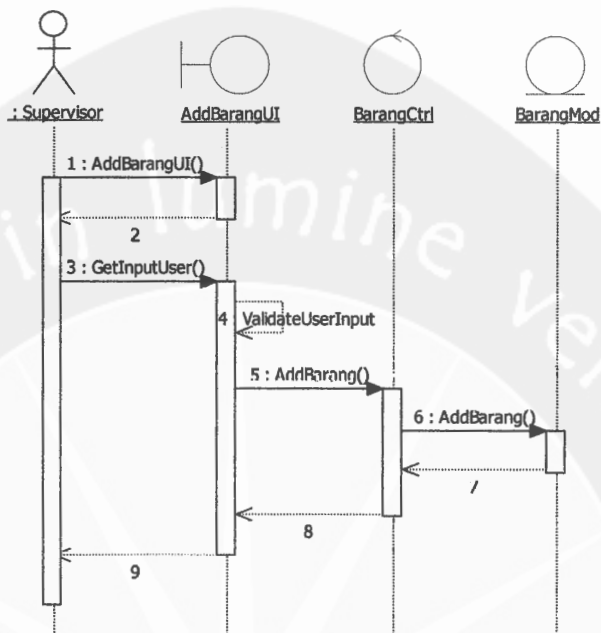
2.2.1.3 Ganti Password



Gambar 2.8 Sequence Diagram : Change Password Pegawai

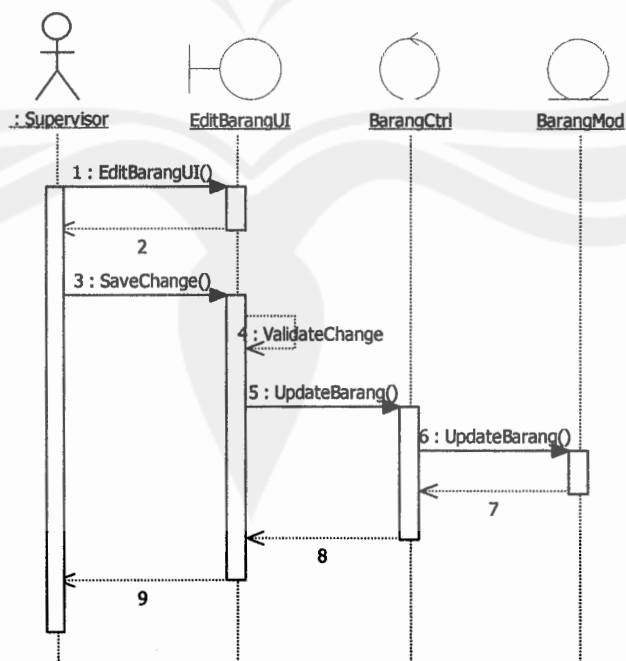
2.2.1.4 Pengelolaan Barang

2.2.1.4.1 Add Data Barang



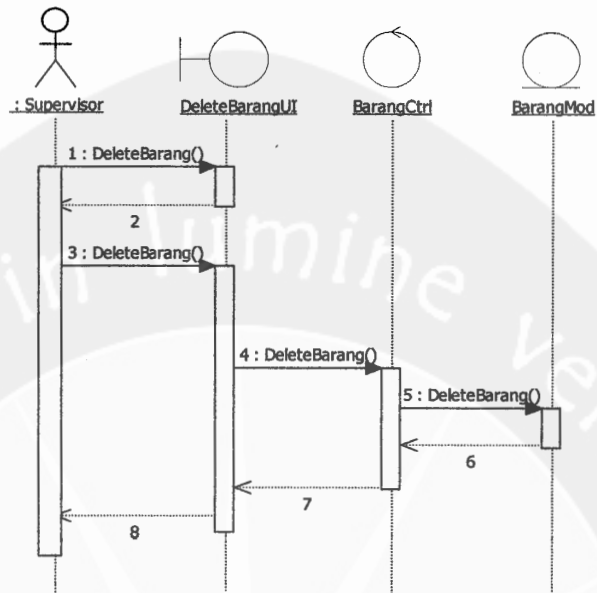
Gambar 2.9 Sequence Diagram : Add Data Barang

2.2.1.4.2 Edit Data Barang



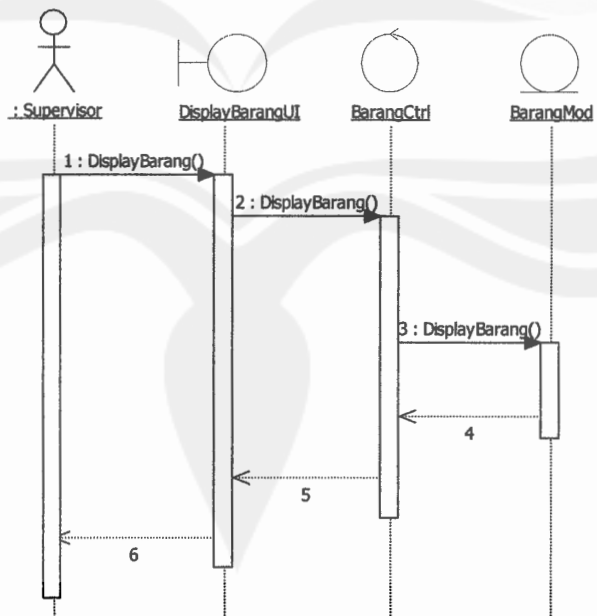
Gambar 2.10 Sequence Diagram : Edit Data Barang

2.2.1.4.3 Delete Data Barang



Gambar 2.11 Sequence Diagram : Delete Data Barang

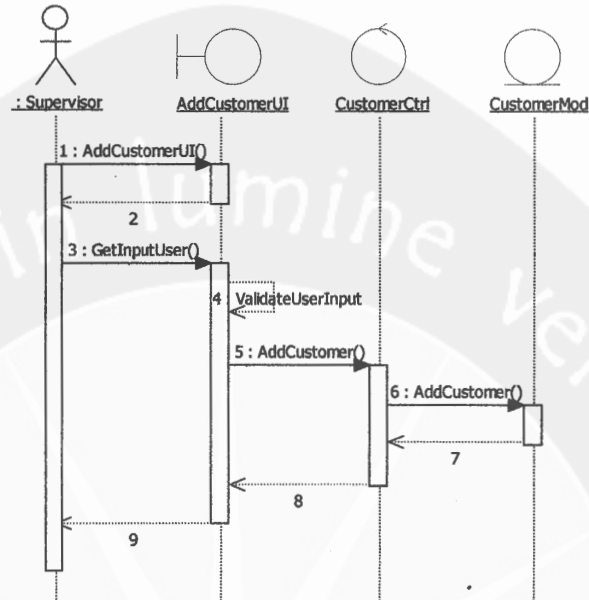
2.2.1.4.4 Display Data Barang



Gambar 2.12 Sequence Diagram : Display Data Barang

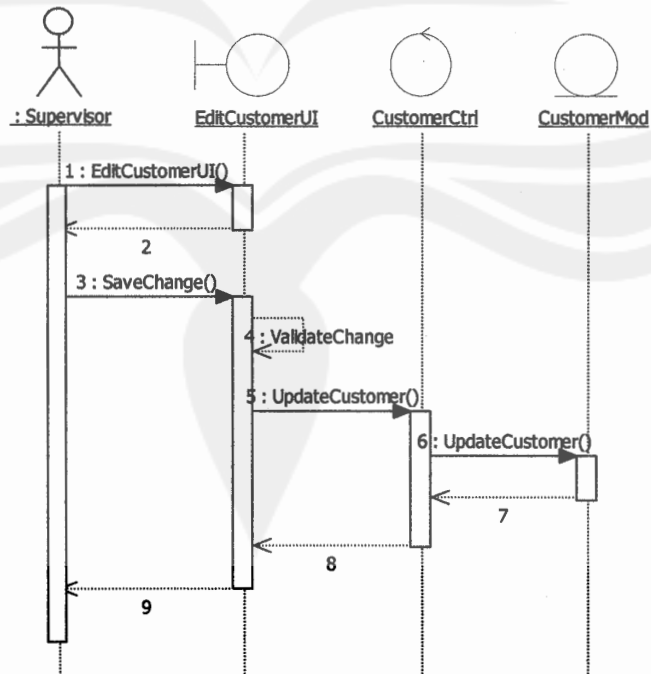
2.2.1.5 Pengelolaan Customer

2.2.1.5.1 Add Data Customer



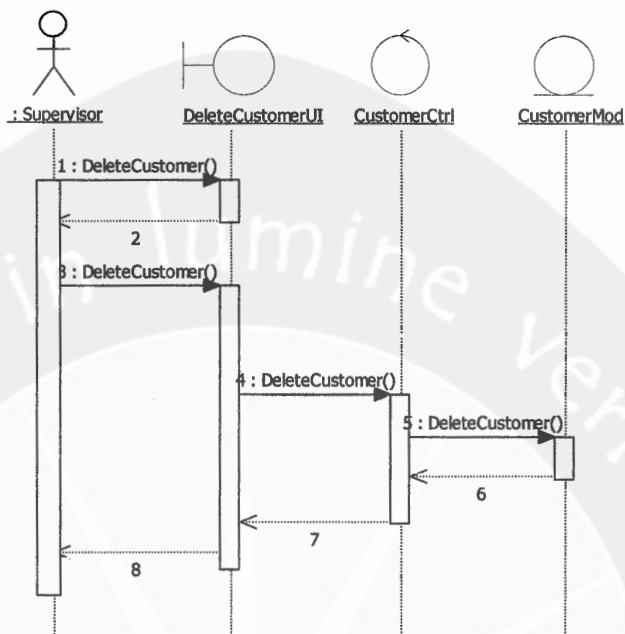
Gambar 2.13 Sequence Diagram : Add Data Customer

2.2.1.5.2 Edit Data Customer



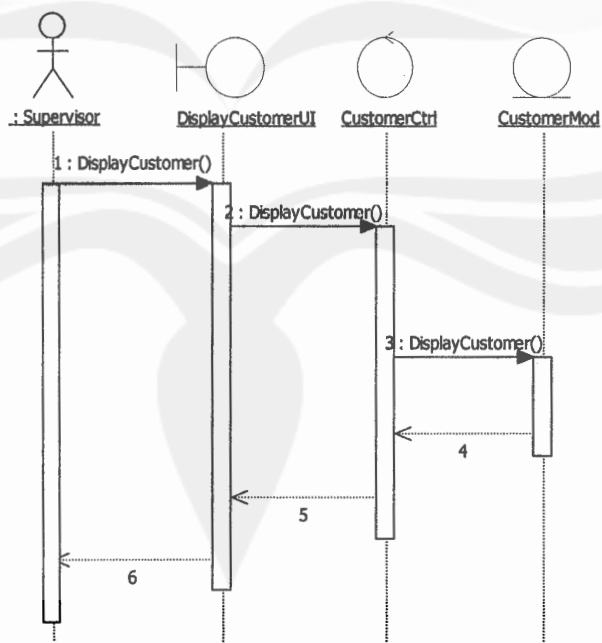
Gambar 2.14 Sequence Diagram : Add Data Customer

2.2.1.5.3 Delete Data Customer



Gambar 2.15 Sequence Diagram : Add Data Customer

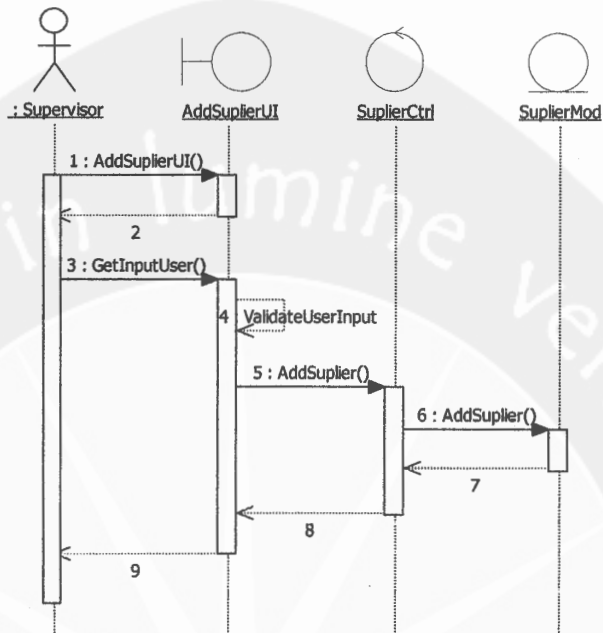
2.2.1.5.4 Display Data Customer



Gambar 2.16 Sequence Diagram : Add Data Customer

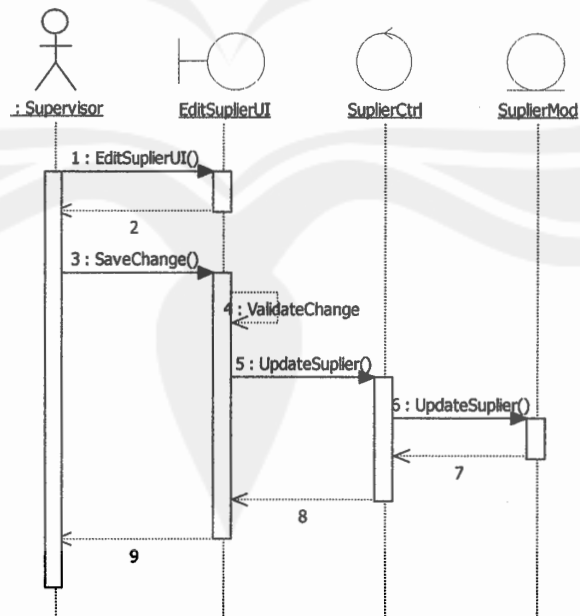
2.2.1.6 Pengelolaan Suplier

2.2.1.6.1 Add Data Suplier



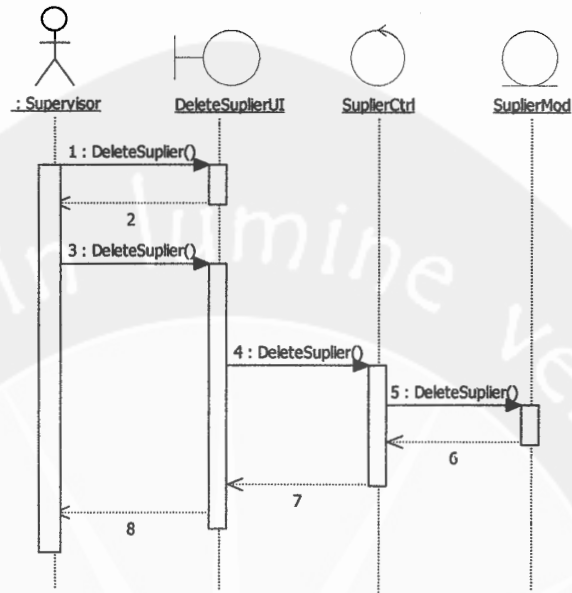
Gambar 2.17 Sequence Diagram : Add Data Suplier

2.2.1.6.2 Edit Data Suplier



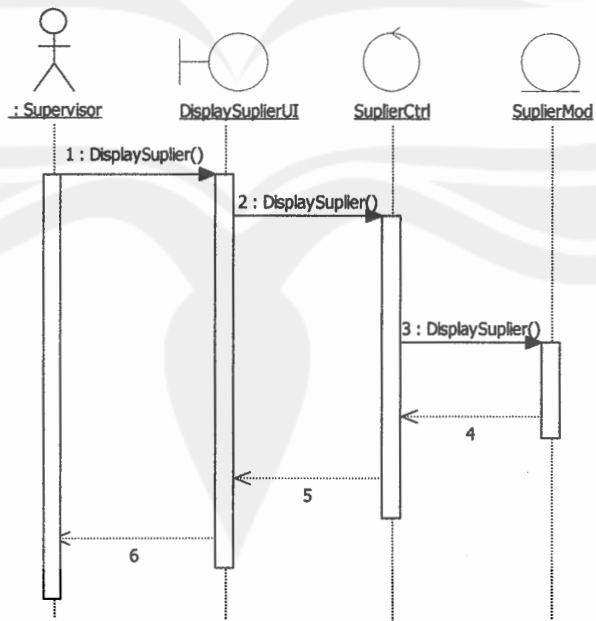
Gambar 2.18 Sequence Diagram : Edit Data Suplier

2.2.1.6.3 Delete Data Suplier



Gambar 2.19 Sequence Diagram : Delete Data Suplier

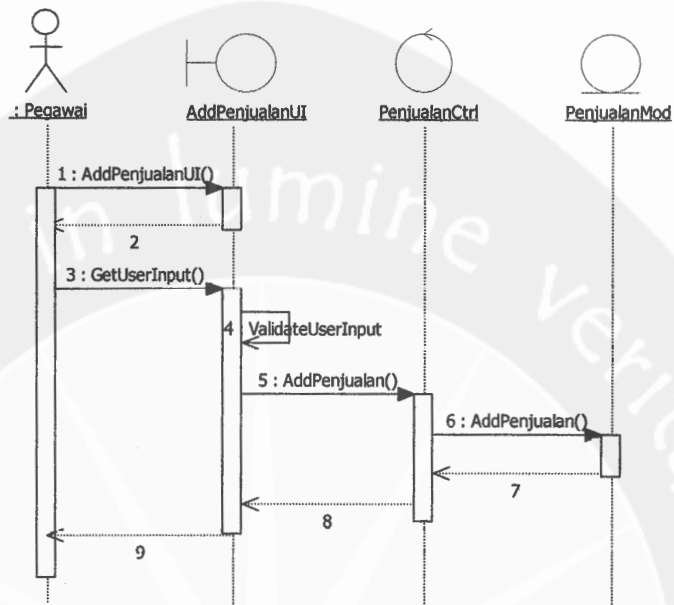
2.2.1.6.4 Display Data Suplier



Gambar 2.20 Sequence Diagram : Display Data Suplier

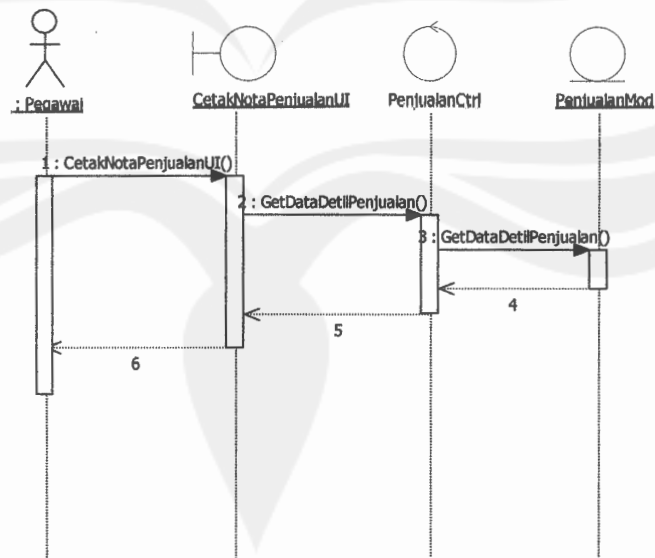
2.2.1.7 Pengelolaan Transaksi Penjualan

2.2.1.7.1 Add Data Penjualan



Gambar 2.21 Sequence Diagram : Add Data Penjualan

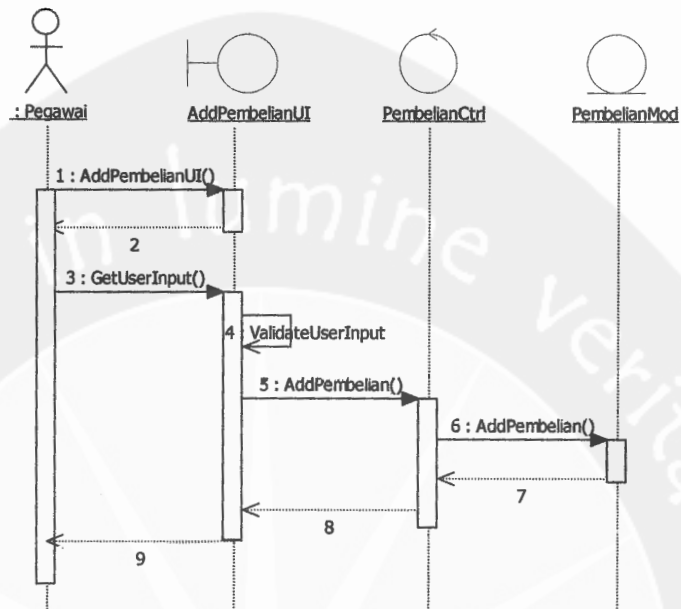
2.2.1.7.2 Cetak Nota Penjualan



Gambar 2.22 Sequence Diagram : Cetak Nota Penjualan

2.2.1.8 Pengelolaan Pencatatan Pembelian

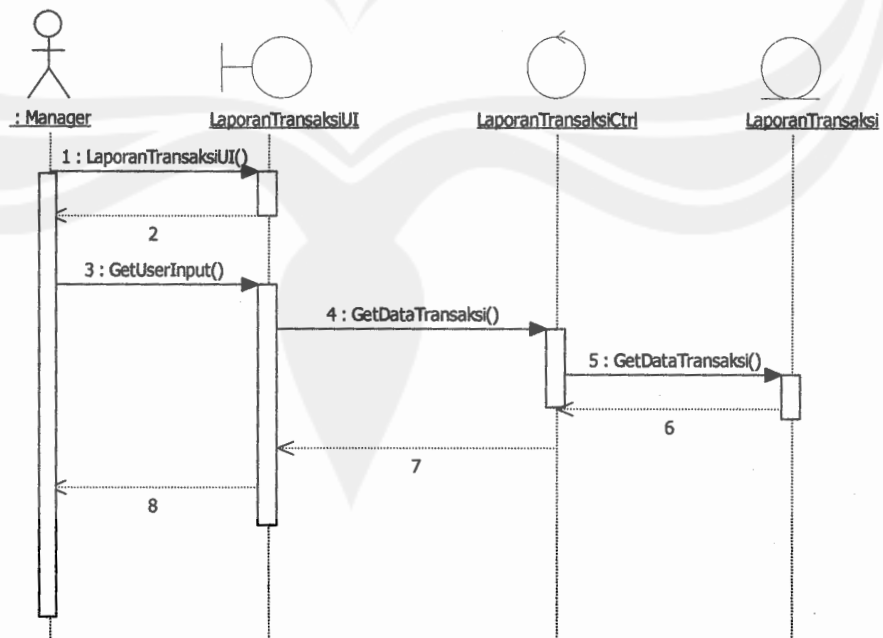
2.2.1.8.1 Add Data Pembelian



Gambar 2.23 Sequence Diagram : Add Data Pembelian

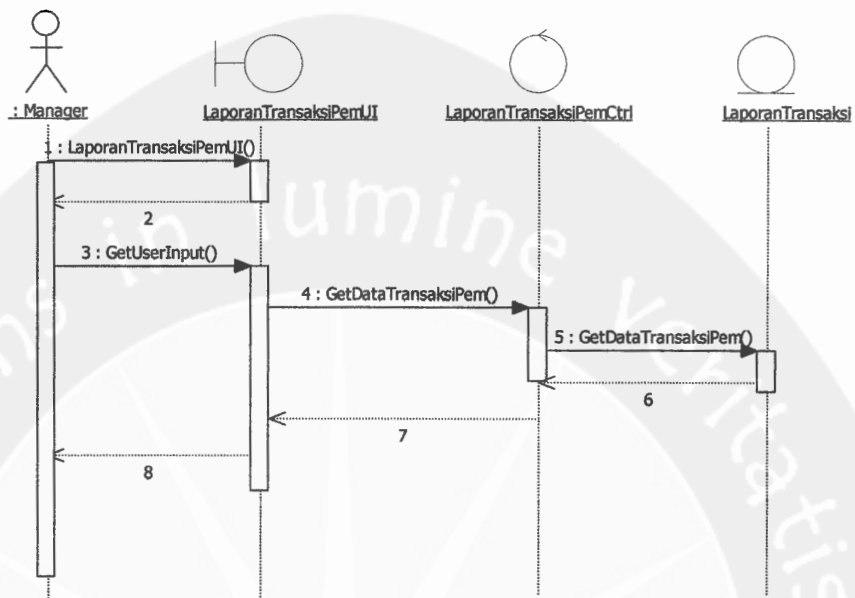
2.2.1.9 Pengelolaan Laporan Transaksi

2.2.1.9.1 Cetak Laporan Transaksi Penjualan



Gambar 2.24 Sequence Diagram : Cetak Laporan Transaksi Penjualan

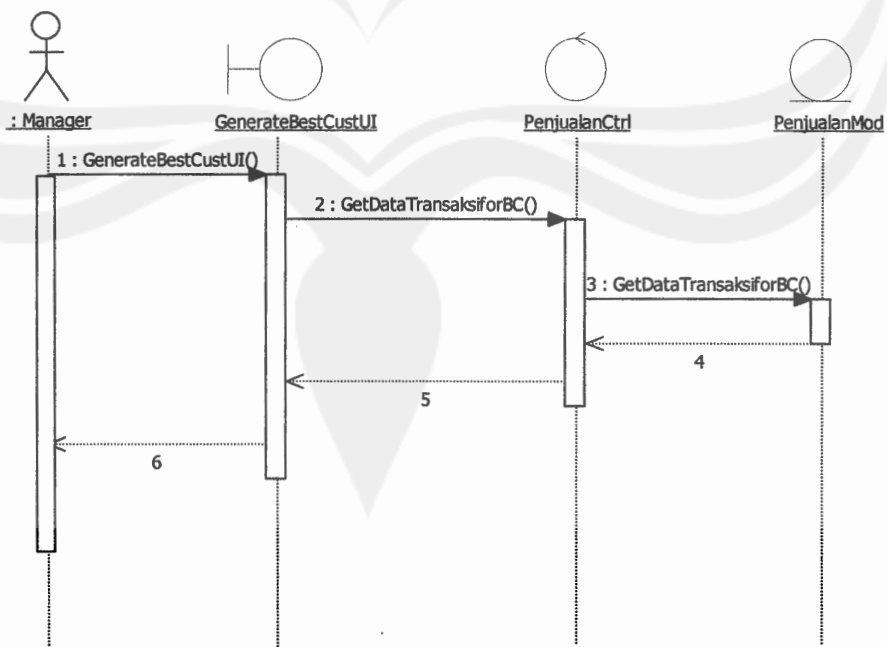
2.2.1.9.2 Cetak Laporan Transaksi Pembelian



Gambar 2.25 Sequence Diagram : Cetak Laporan Transaksi Pembelian

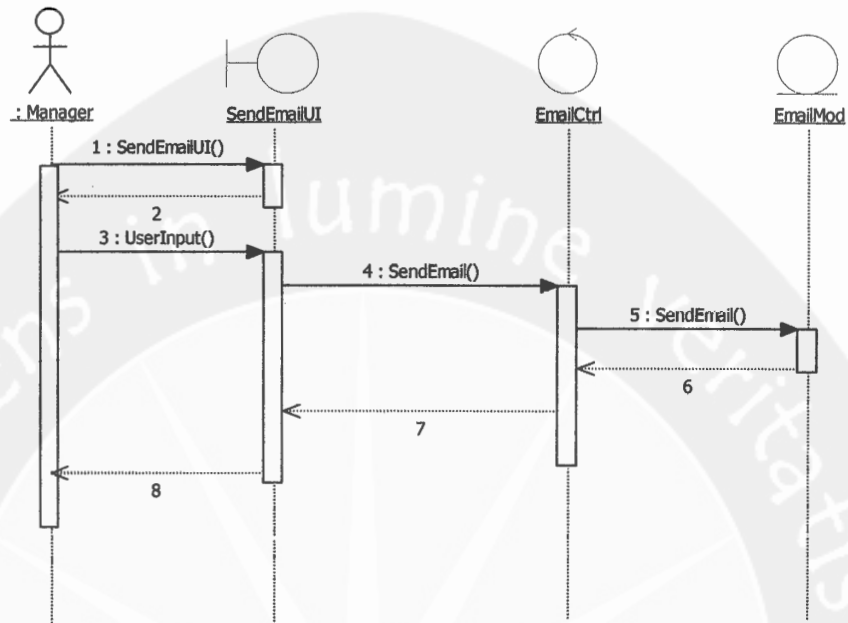
2.2.1.10 Analisis Pelanggan Terbaik

2.2.1.10.1 Generate Pelanggan Terbaik



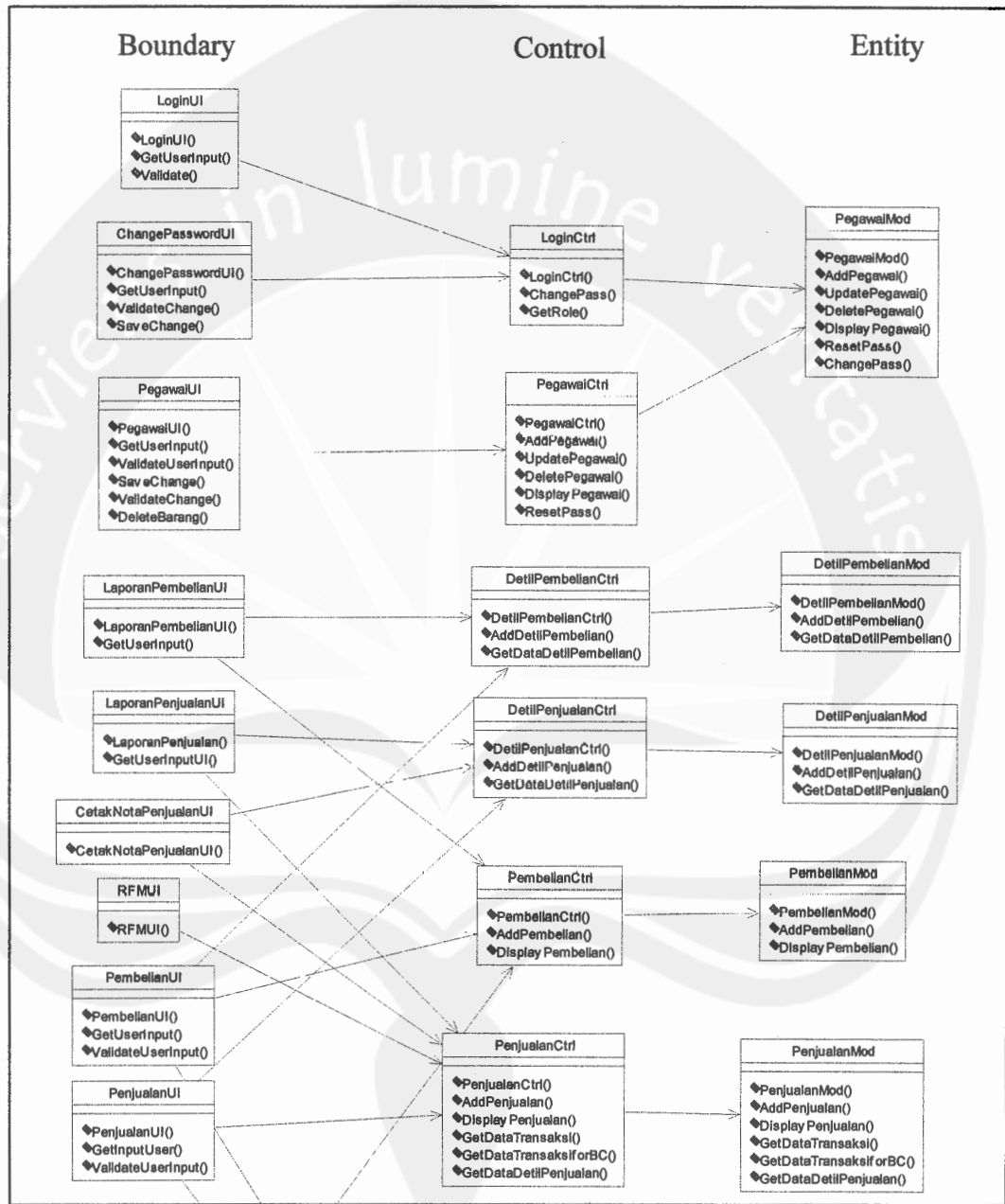
Gambar 2.26 Sequence Diagram : Generate Best Customer

2.2.1.10.2 Kirim Email untuk Best Customer

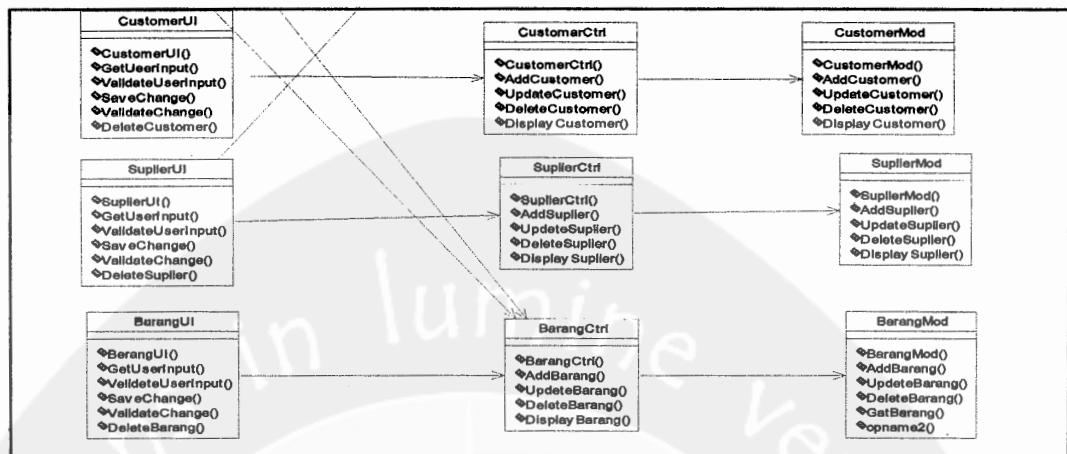


Gambar 2.27 Sequence Diagram : Kirim Email untuk Best Customer

2.2.2 Class Diagram



Gambar 2.28 Class Diagram Bagian 1



Gambar 2.29 Class Diagram Bagian 2

2.2.3 Class Diagram Specific Descriptions

2.2.3.1 Specific Design Class LoginUI

LoginUI	<<boundary>>
<pre> +LoginUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor. +validate():bool Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong. </pre>	

2.2.3.2 Specific Design Class ChangePasswordUI

ChangePasswordUI	<<boundary>>
<pre> +ChangePasswordUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +saveChanges():void </pre>	

Operasi ini digunakan untuk mengambil data-data pegawai yang diubah maupun tidak diubah oleh aktor.

+validateChanges():bool

Operasi ini digunakan untuk melakukan validasi terhadap semua data yang diubah maupun tidak diubah aktor.

+getUserInput():void

Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.

2.2.3.3 Specific Design Class PegawaiUI

PegawaiUI	<<boundary>>
<p>+PegawaiUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+saveChanges():void Operasi ini digunakan untuk mengambil data-data pegawai yang diubah maupun tidak diubah oleh aktor.</p> <p>+validateChanges():bool Operasi ini digunakan untuk melakukan validasi terhadap semua data yang diubah maupun tidak diubah aktor.</p> <p>+getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.</p> <p>+validateUserInput():bool Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong.</p> <p>+deletePegawai():void Operasi ini digunakan untuk menghapus data pegawai yang ada di dalam basis data.</p>	

2.2.3.4 Specific Design Class Barang

BarangUI	<<boundary>>
<pre>+BarangUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +saveChanges():void Operasi ini digunakan untuk mengambil data-data barang yang diubah maupun tidak diubah oleh aktor. +validateChanges():bool Operasi ini digunakan untuk melakukan validasi terhadap semua data yang diubah maupun tidak diubah aktor. +getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor. +validateUserInput():bool Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong. +deleteBarang():void Operasi ini digunakan untuk menghapus data barang yang ada di dalam basis data.</pre>	

2.2.3.5 Specific Design Class PenjualanUI

PenjualanUI	<<boundary>>
<pre>+PenjualanUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor. +validateUserInput():bool</pre>	

Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong.

2.2.3.6 Specific Design Class PembelianUI

PembelianUI	<<boundary>>
<p>+PembelianUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.</p> <p>+validateUserInput():bool Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong.</p>	

2.2.3.7 Specific Design Class CustomerUI

CustomerUI	<<boundary>>
<p>+CustomerUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+saveChanges():void Operasi ini digunakan untuk mengambil data-data customer yang diubah maupun tidak diubah oleh aktor.</p> <p>+validateChanges():bool Operasi ini digunakan untuk melakukan validasi terhadap semua data yang diubah maupun tidak diubah aktor.</p> <p>+getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.</p>	

+validateUserInput():bool

Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong.

+deleteCustomer():void

Operasi ini digunakan untuk menghapus data customer yang ada di dalam basis data.

2.2.3.8 Specific Design Class SuplierUI

SuplierUI	<<boundary>>
+SuplierUI() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+saveChanges():void Operasi ini digunakan untuk mengambil data-data suplier yang diubah maupun tidak diubah oleh aktor.	
+validateChanges():bool Operasi ini digunakan untuk melakukan validasi terhadap semua data yang diubah maupun tidak diubah aktor.	
+getUserInput():void Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.	
+validateUserInput():bool Operasi ini digunakan untuk melakukan validasi terhadap semua yang dimasukkan aktor, termasuk di dalamnya validasi terhadap field-field yang tidak dibolehkan kosong.	
+deleteSuplier():void Operasi ini digunakan untuk menghapus data suplier yang ada di dalam basis data.	

2.2.3.9 Specific Design Class LaporanPembelianUI

LaporanPembelianUI	<<boundary>>

+LaporanPembelianUI()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+getUserInput():void

Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.

2.2.3.10 Specific Design Class LaporanPenjualanUI

LaporanPenjualanUI

<<boundary>>

+LaporanPenjualanUI()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+getUserInput():void

Operasi ini digunakan untuk mengambil data yang dimasukkan oleh aktor.

2.2.3.11 Specific Design Class RFMUI

RFMUI

<<boundary>>

+RFMUI()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

2.2.3.12 Specific Design Class CetakNotaPenjualanUI

CetakNotaPenjualanUI

<<boundary>>

+CetakNotaPenjualanUI()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

2.2.3.13 Specific Design Class LoginCtrl

LoginCtrl	<<control>>
+LoginCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+ChangePass():void Operasi ini digunakan untuk password pegawai yang ada di dalam basis data.	
+GetRole() Operasi ini digunakan untuk mengambil data role pegawai.	

2.2.3.14 Specific Design Class PegawaiCtrl

PegawaiCtrl	<<control>>
+PegawaiCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+DisplayPegawai():DataTable Operasi ini digunakan untuk menampilkan data pegawai di dalam basis data.	
+updatePegawai():void Operasi ini digunakan untuk memperbaharui data pegawai lama dalam basis data dengan data yang baru.	
+addPegawai():void Operasi ini digunakan untuk memasukkan data pegawai baru ke dalam basis data.	
+deletePegawai():void Operasi ini digunakan untuk menghapus satu data pegawai dalam basis data.	
+ResetPass():void Operasi ini digunakan untuk mereset password pegawai yang ada di dalam basis data.	

+ChangePass():void

Operasi ini digunakan untuk password pegawai yang ada di dalam basis data.

2.2.3.15 Specific Design Class BarangCtrl

BarangCtrl	<<control>>
<p>+BarangCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+DisplayBarang():DataTable Operasi ini digunakan untuk menampilkan data barang di dalam basis data.</p> <p>+updateBarang():void Operasi ini digunakan untuk memperbaharui data barang lama dalam basis data dengan data yang baru.</p> <p>+addBarang():void Operasi ini digunakan untuk memasukkan data barang baru ke dalam basis data.</p> <p>+deleteBarang():void Operasi ini digunakan untuk menghapus satu data barang dalam basis data.</p>	

2.2.3.16 Specific Design Class PenjualanCtrl

PenjualanCtrl	<<control>>
<p>+PenjualanCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+AddPenjualan():void Operasi ini digunakan untuk menambahkan data penjualan di dalam basis data.</p> <p>+DisplayPenjualan():DataTable</p>	

Operasi ini digunakan untuk menampilkan data penjualan di dalam basis data.

+GetDataTransaksi():void

Operasi ini digunakan untuk mengambil data transaksi penjualan di dalam basis data.

+GetDataTransaksiforBC():void

Operasi ini digunakan untuk mengambil data transaksi penjualan untuk analisis Best Customer di dalam basis data.

+GetDataDetilPenjualan():void

Operasi ini digunakan untuk mengambil data detil transaksi penjualan di dalam basis data.

2.2.3.17 Specific Design Class PembelianCtrl

PembelianCtrl	<<control>>
<p>+PemeblianCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+AddPembelian():void Operasi ini digunakan untuk menambahkan data pembelian di dalam basis data</p> <p>+DisplayPembelian():DataTable Operasi ini digunakan untuk menampilkan data pembelian di dalam basis data.</p>	

2.2.3.18 Specific Design Class DetilPembelianCtrl

DetilPembelianCtrl	<<control>>
<p>+DetilPemeblianCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+AddDetilPembelian():void Operasi ini digunakan untuk menambahkan data detil pembelian</p>	

di dalam basis data

+GetDataDetilPembelian():DataTable

Operasi ini digunakan untuk mengambil data detil pembelian.

2.2.3.19 Specific Design Class DetilPenjualanCtrl

DetilPembelianCtrl	<<control>>
+DetilPenjualanCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+AddDetilPenjualan():void Operasi ini digunakan untuk menambahkan data detil penjualan di dalam basis data	
+GetDataDetilPenjualan():DataTable Operasi ini digunakan untuk mengambil data detil penjualan.	

2.2.3.20 Specific Design Class CustomerCtrl

CustomerCtrl	<<control>>
+CustomerCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+DisplayCustomer():DataTable Operasi ini digunakan untuk menampilkan data customer di dalam basis data.	
+updateCustomer():void Operasi ini digunakan untuk memperbaharui data customer lama dalam basis data dengan data yang baru.	
+addCustomer():void Operasi ini digunakan untuk memasukkan data customer baru ke dalam basis data.	
+deleteCustomer():void Operasi ini digunakan untuk menghapus satu data customer	

dalam basis data.

2.2.3.21 Specific Design Class SuplierCtrl

SuplierCtrl	<<control>>
+SuplierCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+DisplaySuplier():DataTable Operasi ini digunakan untuk menampilkan data suplier di dalam basis data.	
+updateSuplier():void Operasi ini digunakan untuk memperbaharui data suplier lama dalam basis data dengan data yang baru.	
+addSuplier():void Operasi ini digunakan untuk memasukkan data suplier baru ke dalam basis data.	
+deleteSuplier():void Operasi ini digunakan untuk menghapus satu data suplier dalam basis data.	

2.2.3.22 Specific Design Class PegawaiMod

PegawaiMod	<<entity>>
+PegawaiMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+DisplayPegawai():DataTable Operasi ini digunakan untuk menampilkan data pegawai di dalam basis data.	
+updatePegawai():void Operasi ini digunakan untuk memperbaharui data pegawai lama dalam basis data dengan data yang baru.	

+addPegawai():void

Operasi ini digunakan untuk memasukkan data pegawai baru ke dalam basis data.

+deletePegawai():void

Operasi ini digunakan untuk menghapus satu data pegawai dalam basis data.

+ResetPass():void

Operasi ini digunakan untuk mereset password pegawai yang ada di dalam basis data.

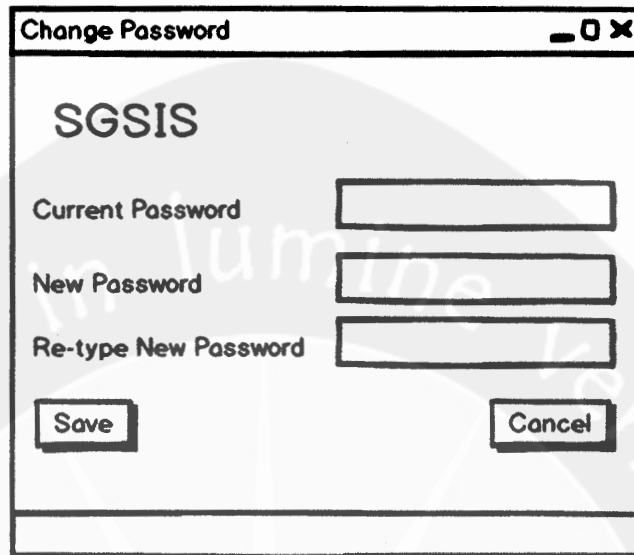
+ChangePass():void

Operasi ini digunakan untuk password pegawai yang ada di dalam basis data.

2.2.3.23 Specific Design Class BarangMod

BarangMod	<<entity>>
+BarangMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
!DisplayBarang():DataTable Operasi ini digunakan untuk menampilkan data barang di dalam basis data.	
+updateBarang():void Operasi ini digunakan untuk memperbaharui data barang lama dalam basis data dengan data yang baru.	
+addBarang():void Operasi ini digunakan untuk memasukkan data barang baru ke dalam basis data.	
+deleteBarang():void Operasi ini digunakan untuk menghapus satu data barang dalam basis data.	

4.9 Antarmuka Ubah Password



The image shows a 'Change Password' dialog box. The title bar reads 'Change Password' with a close button. The main content area is titled 'SGSIS'. It contains three text input fields labeled 'Current Password', 'New Password', and 'Re-type New Password'. At the bottom, there are two buttons: 'Save' and 'Cancel'.

Gambar 4.9 Rancangan Antarmuka Ubah Password

Antarmuka pada gambar 4.9 digunakan oleh semua aktor untuk mengubah password dengan mengisi field data *current password*, *New Password*, dan *Confirmation new password* yang telah disediakan, Kemudian mengeklik tombol 'Save' untuk melakukan penyimpanan data tersebut ke dalam basis data. Jika aktor ingin membatalkan perubahan password, cukup mengeklik tombol 'Cancel'.

4.10 Antarmuka Transaksi Penjualan

Sales Order Transaction

Sales Order Transaction

Invoice No:

Transaction Date:

Customer ID:

Customer Name:

Customer Address:

Customer Phone:

Email:

Item Name	Item Category	Sale Price	Unit	Item Stock
Chemical and Dust Respirator NP 306	Alat Keselamatan	57000	Pcs	22
Galvanis Wire Clips 10mm	Aksesoris	1500	Pcs	1000
Shackle Galvanis uk 10mm	Shackle	1500	Pcs	360

Item ID	Item Name	Quantity	Price	Total	Cancel
1	Shackle Galvanis uk 10mm	5	1500	7500	Cancel

Subtotal

Tax

Discount

Total

Gambar 4.10 Rancangan Antarmuka Transaksi Penjualan

Antarmuka pada gambar 4.10 digunakan oleh supervisor dan pegawai untuk menambahkan atau melakukan transaksi penjualan, untuk invoice no, telah degenerate secara otomatis oleh sistem, kemudian aktor diminta untuk memilih customer, setelah itu, aktor diminta memilih barang yang akan dijual dengan cara mengklik 2 kali pada table barang, dan barang tersebut akan tampil pada table order. Setelah itu, untuk menyimpan data transaksi aktor mengklik tombol 'Save', kemudian akan akan pilihan apakah aktor ingin mencetak nota pembayaran atau tidak. Dan jika ingin membatalkan transaksi, aktor memilih tombol 'Cancel'.

4.11 Antarmuka Transaksi Pembelian

Purchase Order Transaction

Purchase Order Transaction

Invoice No:

Transaction Date:

Supplier ID:

Supplier Name:

Supplier Address:

Supplier Phone:

Item Name	Item Category	Purchase Price	Unit	Item Stock
Chemical and Dust Respirator NP 306	Alat Keselamatan	57000	Pcs	22
Galvanis Wire Clips 10mm	Aksesoris	1500	Pcs	1000
Shackle Galvanis uk 10mm	Shackle	1500	Pcs	360

Item ID	Item Name	Quantity	Price	Total	Cancel
1	Shackle Galvanis uk 10mm	5	1500	7500	Cancel

Subtotal:

Tax:

Total:

Gambar 4.11 Rancangan Antarmuka Pencatatan Pembelian

Antarmuka pada gambar 4.11 digunakan oleh supervisor dan pegawai untuk menambahkan atau melakukan pencatatan pembelian, untuk invoice no, telah degenerate secara otomatis oleh sistem, kemudian aktor diminta untuk memilih supplier, setelah itu, aktor diminta memilih barang yang akan beli dengan cara mengklik 2 kali pada table barang, dan barang tersebut akan tampil pada table order. Setelah itu, untuk menyimpan data pencatatan pembelian aktor mengklik tombol 'Save' dan jika ingin membatalkan pencatatan, aktor memilih tombol 'Cancel'.

4.12 Antarmuka Laporan Transaksi Penjualan

The screenshot shows a window titled "Reporting" with a standard Windows-style title bar (minimize, maximize, close). Inside the window, there are two dropdown menus. The first is labeled "Month" and has "January" selected, with a list showing "February" and "March" as options. The second is labeled "Year" and has "2012" selected, with a list showing "2013" and "2014" as options. At the bottom of the window, there are two buttons: "View Report" on the left and "Cancel" on the right.

Gambar 4.12 Rancangan Antarmuka Laporan Transaksi Penjualan

Antarmuka pada gambar 4.12 digunakan oleh manager untuk melihat laporan penjualan pada periode tertentu. Jika manager ingin melihat laporannya maka klik tombol 'Show report' dan jika ingin membatalkan melihat laporan, manager memilih tombol 'Cancel'.

4.13 Antarmuka Laporan Transaksi Pembelian

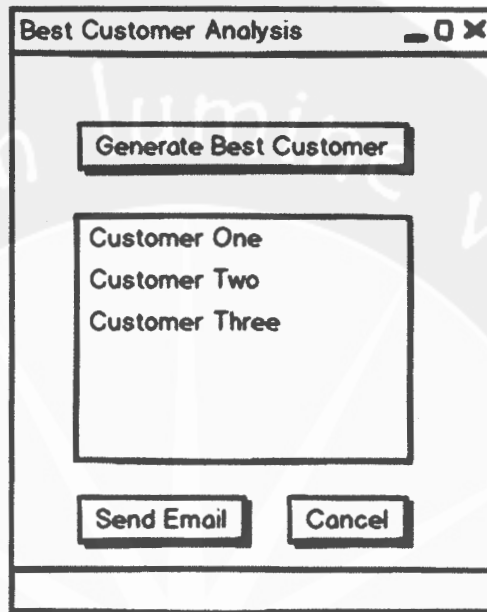
The screenshot shows a window titled "Reporting" with a standard Windows-style title bar (minimize, maximize, close). Inside the window, there are two dropdown menus. The first is labeled "Month" and has "January" selected, with a list showing "February" and "March" as options. The second is labeled "Year" and has "2012" selected, with a list showing "2013" and "2014" as options. At the bottom of the window, there are two buttons: "View Report" on the left and "Cancel" on the right.

Gambar 4.13 Rancangan Antarmuka Laporan Transaksi Pembelian

Antarmuka pada gambar 4.13 digunakan oleh manager untuk melihat laporan pembelian pada periode tertentu. Jika manager ingin melihat laporannya maka klik tombol 'Show report' dan jika ingin membatalkan melihat laporan, manager memilih tombol 'Cancel'.

report' dan jika ingin membatalkan melihat laporan, manager memilih tombol 'Cancel'.

4.14 Antarmuka Analisis Pelanggan Terbaik

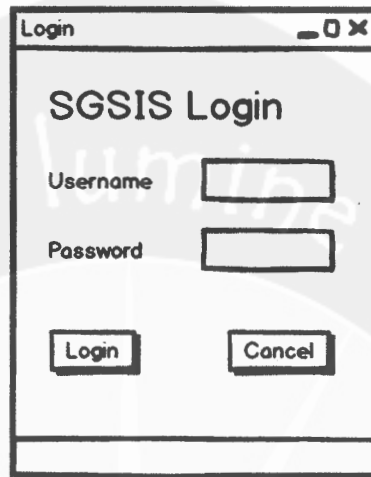


Gambar 4.14 Rancangan Antarmuka Analisis Pelanggan Terbaik

Antarmuka pada gambar 4.14 digunakan oleh manager untuk melihat pelanggan terbaik, berdasarkan analisis pada transaksi penjualan. Jika manager mengklik tombol 'Generate Best Customer', maka sistem akan menampilkan 3 nama teratas dari pelanggan terbaik. Jika manager ingin mengirimkan email untuk penawaran tertentu pada pelanggan terbaik, maka manager klik tombol 'Send Email', dan jika untuk melakukan pembatalan manager klik tombol 'Cancel'.

4 Perancangan Antarmuka

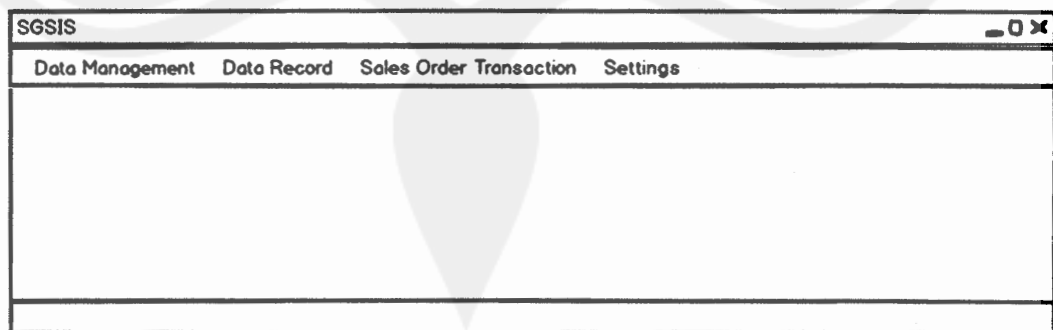
4.1 Antarmuka Login



Gambar 4.1 Rancangan Antarmuka Login

Antarmuka pada gambar 4.1 digunakan untuk melakukan *login* ke dalam sistem. Untuk mendapat akses ke dalam sistem, user harus memasukkan username dan password pada textbox yang telah disediakan, lalu menekan enter atau mengklik tombol 'Login'.

4.2 Antarmuka Navigasi Pengguna : Supervisor

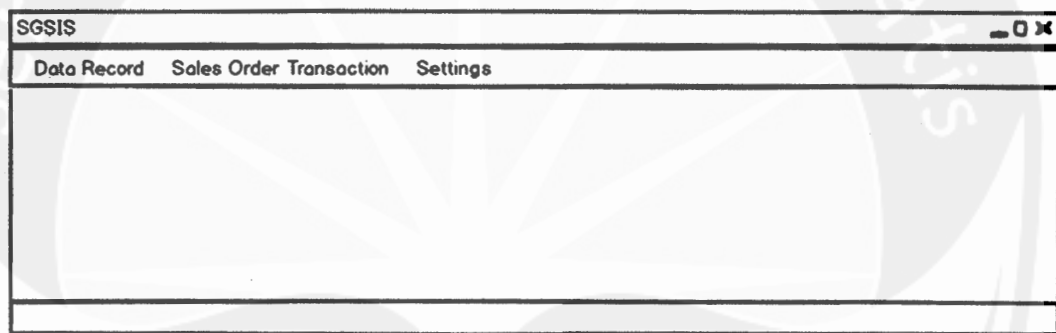


Gambar 4.2 Rancangan Navigasi Pengguna : Supervisor

Antarmuka pada gambar 4.2 adalah menu utama dalam role supervisor, digunakan untuk masuk ke dalam pengelolaan-pengelolaan data yang menjadi fungsionalitas sistem

untuk role supervisor. Jika menu 'Data Mangement' diklik akan muncul submenu 'Employee Management', 'Item Management', 'Customer Manegement', 'Supplier Management'. Jika menu Data Record diklik maka akan muncul submenu 'Pruchase Order Record'. Jika menu 'Sales Order Transaction' diklik maka akan muncul form 'Sales Order Transaction'. Jika menu setting diklik maka akan muncul submenu 'Change Password', 'Logout', 'Exit'.

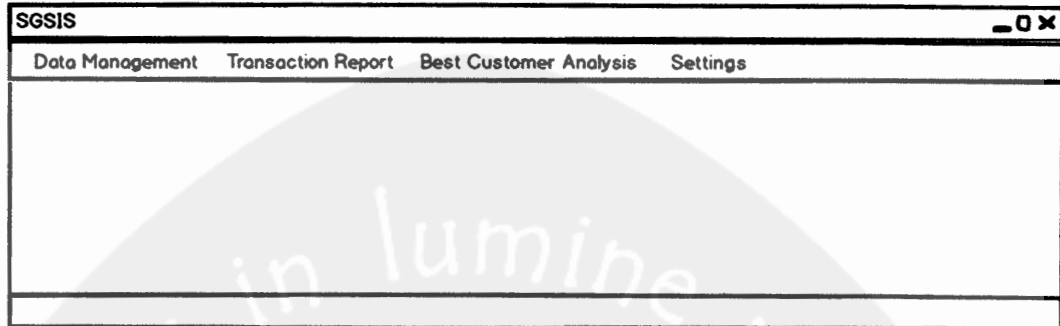
4.3 Antarmuka Navigasi Pengguna : Pegawai



Gambar 4.3 Rancangan Navigasi Pengguna : Pegawai

Antarmuka pada gambar 4.3 adalah menu utama dalam role pegawai, digunakan untuk masuk ke dalam pengelolaan-pengelolaan data yang menjadi fungsionalitas sistem untuk role pegawai. Jika menu Data Record diklik maka akan muncul submenu 'Purchase Order Record'. Jika menu 'Sales Order Transaction' diklik maka akan muncul form 'Sales Order Transaction'. Jika menu setting diklik maka akan muncul submenu 'Change Password', 'Logout', 'Exit'.

4.4 Antarmuka Navigasi Pengguna : Manager



Gambar 4.4 Rancangan Navigasi Pengguna : Manager

Antarmuka pada gambar 4.4 adalah menu utama dalam role manager, digunakan untuk masuk ke dalam pengelolaan-pengelolaan data yang menjadi fungsionalitas sistem untuk role manager. Jika menu 'Data Mangement' diklik akan muncul submenu 'Employee Management', 'Item Management', 'Customer Manegement', 'Suplier Management'. Jika menu *Transaction Report* diklik maka akan muncul form pembuatan laporan transaksi. Jika menu *Best Customer Analysis* diklik maka akan muncul form 'Best Customer Analysis'. Jika menu setting diklik maka akan muncul submenu 'Change Password', 'Logout', 'Exit'.

4.5 Antarmuka Pengelolaan Pegawai

The screenshot shows a window titled 'User Management'. At the top, there is a search bar with a magnifying glass icon and the text 'search'. Below the search bar is a table with three columns: 'Username', 'Employee Name', and 'Employee Role'. The table contains three rows of data. Below the table, there are four buttons: 'Add', 'Edit', 'Delete', and 'Reset Password'. To the right of these buttons are three input fields: 'Name', 'Username', and 'Role'. The 'Role' field is a dropdown menu with 'Supervisor' selected. At the bottom right of the window is a 'Cancel' button.

Username	Employee Name	Employee Role
GG	Guiltionus	Administrator
Tonzzzz	Tony Stark	Manoger
Sh3er	Sherlock	Pegawai

Gambar 4.5 Rancangan Antarmuka Pengelolaan Pegawai

Antarmuka pada gambar 4.5 digunakan oleh Supervisor dan Manager untuk mengelola data pegawai seperti menambahkan, menampilkan, mengubah, menghapus, dan menampilkan data pegawai ke dalam basis data. Data pegawai akan ditampilkan ketika form ini dipanggil. Data pegawai di sini berbentuk table. Untuk menambah pegawai, aktor mengklik tombol 'Add'. Untuk mengubah pegawai, aktor mengklik tombol 'Edit', untuk menghapus pegawai, aktor mengklik tombol 'Delete'. Dan untuk mereset password pegawai, aktor mengklik tombol *reset password*

4.6 Antarmuka Pengelolaan Barang

Item Management

Item Management

search

Item Name	Item Category	Purchase Price	Sale Price	Unit	Item Stock
Boju Pelampung uk. L	Alat Keselamatan	250000	300000	Pcs	40
Chemical and Dust Respirator NP 306	Alat Keselamatan	57000	90000	Pcs	22

Add Edit Delete

Item Name

Item Category

Purchase Price

Sale Price

Unit

Stock Cancel

Gambar 4.6 Rancangan Antarmuka Pengelolaan Barang

Antarmuka pada gambar 4.6 digunakan oleh Supervisor dan Manager untuk mengelola data barang seperti menambahkan, menampilkan, mengubah, menghapus, dan menampilkan data barang ke dalam basis data. Data barang akan ditampilkan ketika form ini dipanggil. Data barang di sini berbentuk table. Untuk menambah barang, aktor mengklik tombol 'Add'. Untuk mengubah barang, aktor mengklik tombol 'Edit'. Dan untuk menghapus barang, aktor mengklik tombol 'Delete'

4.7 Antarmuka Pengelolaan Customer

The screenshot shows a web application window titled "Customer Management". At the top, there is a search bar with a magnifying glass icon and the text "search". Below the search bar is a table with four columns: "Customer Name", "Customer Phone", "Customer Address", and "Email". A "Pick" button is located to the right of the table. The table contains three rows of customer data:

Customer Name	Customer Phone	Customer Address	Email
Salim (BCA)	0812556448	Jl. Trikora III no 9	Salim123@gmail.com
Mustika Andalas	0811781378	Jl. Kapten Marzuki	mustika.andalas@gmail.com
Jauhari (Perkapalan)	07117848471	Jl. Boombaru komplek	Shiptogo@yahoo.com

Below the table, there are five input fields for "Customer ID", "Customer Name", "Customer Address", "Customer Phone", and "Email". To the right of these fields are three buttons: "Add New Customer", "Edit Customer", and "Cancel".

Gambar 4.7 Rancangan Antarmuka Pengelolaan Customer

Antarmuka pada gambar 4.7 digunakan oleh Supervisor dan Manager untuk mengelola data customer seperti menambahkan, menampilkan, mengubah, menghapus, dan menampilkan data customer ke dalam basis data. Data customer akan ditampilkan ketika form ini dipanggil. Data customer di sini berbentuk table. Untuk menambah customer, aktor mengklik tombol 'Add'. Untuk mengubah customer, aktor mengklik tombol 'Edit'. Dan untuk menghapus customer, aktor mengklik tombol 'Delete'.

4.8 Antarmuka Pengelolaan Suplier

Supplier Name	Supplier Address	Supplier Phone	Pick
Bintang Kencana	Jl.Veteran no 19	0813712988208	
Tunas Logam	Jl. Cinde Welan no38	0711-377744	

Supplier ID	<input type="text"/>	<input type="button" value="Add New Suplier"/>
Supplier Name	<input type="text"/>	<input type="button" value="Edit Suplier"/>
Supplier Address	<input type="text"/>	
Supplier Phone	<input type="text"/>	<input type="button" value="Cancel"/>

Gambar 4.8 Rancangan Antarmuka Pengelolaan Suplier

Antarmuka pada gambar 4.8 digunakan oleh Supervisor dan Manager untuk mengelola data suplier seperti menambahkan, menampilkan, mengubah, menghapus, dan menampilkan data suplier ke dalam basis data. Data suplier akan ditampilkan ketika form ini dipanggil. Data suplier di sini berbentuk table. Untuk menambah suplier, aktor mengklik tombol 'Add'. Untuk mengubah suplier, aktor mengklik tombol 'Edit'. Dan untuk menghapus suplier, aktor mengklik tombol 'Delete'.

2.2.3.24 Specific Design Class PenjualanMod

PenjualanMod	<<entity>>
+PenjualanCtrl() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+AddPenjualan():void Operasi ini digunakan untuk menambahkan data penjualan di dalam basis data.	
+DisplayPenjualan():DataTable Operasi ini digunakan untuk menampilkan data penjualan di dalam basis data.	
+GetDataTransaksi():void Operasi ini digunakan untuk mengambil data transaksi penjualan di dalam basis data.	
+GetDataTransaksiforBC():void Operasi ini digunakan untuk mengambil data transaksi penjualan untuk analisis Best Customer di dalam basis data.	
+GetDataDetilPenjualan():void Operasi ini digunakan untuk mengambil data detil transaksi penjualan di dalam basis data.	

2.2.3.25 Specific Design Class PembelianMod

PembelianMod	<<entity>>
+PembelianMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+AddPembelian():void Operasi ini digunakan untuk menambahkan data pembelian di dalam basis data.	
+DisplayPembelian():DataTable Operasi ini digunakan untuk menampilkan data pembelian di	

dalam basis data.

2.2.3.26 Specific Design Class CustomerMod

CustomerMod	<<entity>>
<p>+CustomerMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+DisplayCustomer():DataTable Operasi ini digunakan untuk menampilkan data customer di dalam basis data.</p> <p>+updateCustomer():void Operasi ini digunakan untuk memperbaharui data customer lama dalam basis data dengan data yang baru.</p> <p>+addCustomer():void Operasi ini digunakan untuk memasukkan data customer baru ke dalam basis data.</p> <p>+deleteCustomer():void Operasi ini digunakan untuk menghapus satu data customer dalam basis data.</p>	

2.2.3.27 Specific Design Class SuplierMod

SuplierMod	<<entity>>
<p>+SuplierMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+DisplaySuplier():DataTable Operasi ini digunakan untuk menampilkan data suplier di dalam basis data.</p> <p>+updateSuplier():void Operasi ini digunakan untuk memperbaharui data suplier lama dalam basis data dengan data yang baru.</p>	

+addSupplier():void

Operasi ini digunakan untuk memasukkan data supplier baru ke dalam basis data.

+deleteSupplier():void

Operasi ini digunakan untuk menghapus satu data supplier dalam basis data.

2.2.3.28 Specific Design Class DetilPembelianCtrl

DetilPembelianMod	<<entity>>
<p>+DetilPembelianMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+AddDetilPembelian():void Operasi ini digunakan untuk menambahkan data detil pembelian di dalam basis data</p> <p>+GetDataDetilPembelian():DataTable Operasi ini digunakan untuk mengambil data detil pembelian.</p>	

2.2.3.29 Specific Design Class DetilPenjualanCtrl

DetilPembelianMod	<<entity>>
<p>+DetilPenjualanMod() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+AddDetilPenjualan():void Operasi ini digunakan untuk menambahkan data detil penjualan di dalam basis data</p> <p>+GetDataDetilPenjualan():DataTable Operasi ini digunakan untuk mengambil data detil penjualan.</p>	

3 Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Data PEGAWAI

Nama	Tipe	Panjang	Keterangan
ID_PEGAWAI	Integer	-	ID Pegawai, primary key
ID_ROLE	Integer	-	ID Role pada Pegawai, foreign key
USERNAME	Varchar	15	Username dari Pegawai
PASSWORD	Varchar	15	Password dari Pegawai
NAMA	Varchar	20	Nama dari Pegawai

3.1.2 Deskripsi Entitas Data ROLE

Nama	Tipe	Panjang	Keterangan
ID_ROLE	Integer	-	Id Role, primary key
ROLE	Varchar	10	Nama dari Role

3.1.3 Deskripsi Entitas Data BARANG

Nama	Tipe	Panjang	Keterangan
ID_BARANG	Integer	-	Id Supplier, primary key
NAMA_BARANG	Varchar	20	Nama dari barang
JENIS_BARANG	Varchar	20	Jenis dari barang
HARGA_BARANG	Float	20	Harga dari barang
UNIT_BARANG	Varchar	5	Unit dari barang
STOK_BARANG	Integer	-	Stok dari barang

3.1.4 Deskripsi Entitas Data CUSTOMER

Nama	Tipe	Panjang	Keterangan
ID_CUSTOMER	Varchar	10	Id Barang, primary key
NAMA_CUSTOMER	Varchar	20	Nama dari barang
NO_TELP	Varchar	20	Nomor Telpon dari Customer
ALAMAT	Varchar	50	Alamat dari Customer
EMAIL	Varchar	20	Email dari Customer

3.1.5 Deskripsi Entitas Data SUPLIER

Nama	Tipe	Panjang	Keterangan
ID_SUPLIER	Varchar	10	Id Barang, primary key
NAMA_SUPLIER	Varchar	20	Nama dari Suplier
NO_TELP	Varchar	20	Nomor Telpon dari Suplier
ALAMAT	Varchar	50	Alamat dari Suplier

3.1.6 Deskripsi Entitas Data PENJUALAN

Nama	Tipe	Panjang	Keterangan
ID_PENJUALAN	Varchar	20	Id Penjualan, primary key
ID_PEGAWAI	Integer	-	Id Pegawai yang bertanggung jawab pada penjualan ini, foreign key
ID_CUSTOMER	Varchar	10	Id Customer yang melakukan penjualan ini, foreign key
SUBTOTAL	Float	-	Subtotal pada transaksi penjualan
PAJAK	Float	-	Pajak dari transaksi penjualan
TOTAL	Float	-	Total harga dari transaksi penjualan
TGL_PENJUALAN	Datetime	-	Tanggal dan waktu yang terjadi saat transaksi penjualan
DISCOUNT	Float	-	Discount dari transaksi penjualan

3.1.7 Deskripsi Entitas Data PEMBELIAN

Nama	Tipe	Panjang	Keterangan
ID_PEMBELIAN	Varchar	20	Id Pembelian, primary key
ID_Supplier	Varchar	10	Id Suplier yang melakukan pembelian ini, foreign key

SUBTOTAL	Float	-	Subtotal pada transaksi pembelian
PAJAK	Float	-	Pajak dari transaksi pembelian
TOTAL_PEMBELIAN	Float	-	Total harga dari transaksi pembelian
TGL_PEMBELIAN	Datetime	-	Tanggal dan waktu yang terjadi saat transaksi pembelian

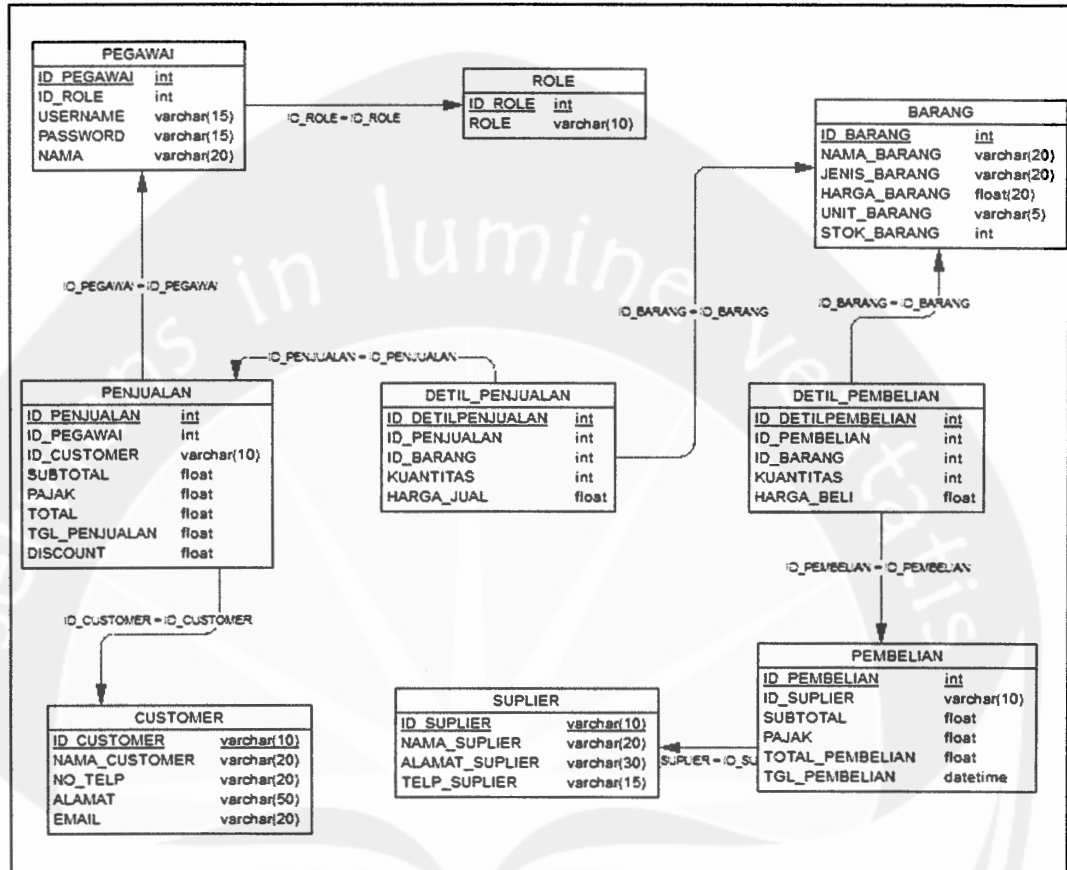
3.1.8 Deskripsi Entitas Data DETIL_PENJUALAN

Nama	Tipe	Panjang	Keterangan
ID_DETILPENJUALAN	Integer	-	Id Detil penjualan, primary key
ID_PENJUALAN	Varchar	20	Id penjualan dari detil penjualan, foreign key
ID_BARANG	Integer	-	Id Barang yang dijual, foreign key
KUANTITAS	Integer	-	Banyaknya Barang yang dijual
HARGA_JUAL	Float	-	Harga Jual Barang

3.1.9 Deskripsi Entitas Data DETIL_PEMBELIAN

Nama	Tipe	Panjang	Keterangan
ID_DETILPEMBELIAN	Integer	-	Id Detil pembelian, primary key
ID_PEMBELIAN	Varchar	20	Id pembelian dari detil pembelian, foreign key
ID_BARANG	Integer	-	Id Barang yang dibeli, foreign key
KUANTITAS	Integer	-	Banyaknya Barang yang dibeli
HARGA_BELI	Float	-	Harga Beli Barang

3.2 Physical Data Model



Gambar 3.1 Physical Data Model

4.15 Antarmuka Kirim Email

The image shows a window titled "Send Email" with a standard Windows-style title bar (minimize, maximize, close buttons). The window contains the following elements:

- To:** A text input field followed by a "Pick Customer" button.
- Subject:** A text input field.
- Message:** A text input field.
- Unit:** A larger text input area.
- Attachment:** A text input field followed by a "Browse" button.
- Buttons:** "Send" and "Cancel" buttons are located at the bottom of the window.

Gambar 4.15 Rancangan Antarmuka Kirim Email

Antarmuka pada gambar 4.15 digunakan oleh manager, untuk mengirimkan email dengan menggunakan aplikasi ini kepada pelanggan terbaik yang telah degenerate di antarmuka Analisis Pelanggan terbaik.