

**ANALISIS DAN PERANCANGAN SISTEM INFORMASI PEMESANAN
BARANG
(STUDI KASUS : PT. TRIVIA NUSANTARA)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
AFIFAHLYA ALIF RIZYOMI
145150400111095



PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

ANALISIS DAN PERANCANGAN SISTEM INFORMASI PEMESANAN BARANG
(STUDI KASUS : PT. TRIVIA NUSANTARA)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Afifahya Alif Rizyomi
145150400111095

Skripsi ini telah diuji dan dinyatakan lulus pada
30 Juli 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II


Mochamad Chandra Saputra, S.Kom., M.T., M.Eng.
NIK. 2016098601061001


Welly Purnomo, S.T., M.Kom.
NIK. 2017088101171001

Mengetahui
Ketua Jurusan Sistem Informasi




Herman Tolle, Dr. Eng., S.T., M.T
NIP. 19740823 200012 1 001



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 30 Juli 2018



Afitahlya Alif Rizyomi
NIM. 145150400111095



KATA PENGANTAR

Puji syukur Alhamdulillah penulis panjatkan kehadiran Allah SWT yang telah memberikan rahmat taufik dan hidayah Nya sehingga penulis dapat menyelesaikan laporan skripsi yang berjudul “Analisis dan Perancangan Sistem Informasi Pemesanan Barang (Studi Kasus: PT. Trivia Nusantara Surabaya)”.

Penulisan skripsi ini merupakan salah satu syarat bagi penulis untuk memperoleh gelar sarjana komputer di Universitas Brawijaya. Dalam pengerjaannya skripsi ini tidak lepas dari hambatan dan kesulitan. Namun berkat bimbingan, dukungan, doa, dan motivasi dari semua pihak skripsi hambatan dan kesulitan dapat diatasi dengan baik.

Dalam kesempatan ini penulis ingin menyampaikan rasa syukur dan terima kasih kepada :

1. Bapak Mochamad Chandra Saputra, S.Kom.,M.T.,M.Eng selaku dosen pembimbing satu dan bapak Welly Purnomo, S. ST, M.T selaku dosen pembimbing dua yang telah memberikan banyak ilmu, saran, kritik dan motivasi untuk penyelesaian skripsi ini,
2. Bapak Suprpto, S.T., M.T. selaku Ketua Program Studi Sistem Informasi Universitas Brawijaya,
3. Bapak Herman Tolle, Dr. Eng., S.T., M.T. selaku Ketua Jurusan Sistem Informasi Universitas Brawijaya,
4. Bapak Ismiarta Aknuranda, S.T, M.Sc, Ph.D selaku dosen Penasihat Akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi,
5. Kedua orang tua penulis Alm. Bapak Mulyo Akbar dan Ibu Darmi Utami beserta keluarga besar penulis atas segala motivasi, nasehat, doa, dan dukungan baik secara moril dan materiil dalam penyelesaian skripsi ini,
6. Pegawai PT. Trivia Nusantara yang telah mengizinkan penulis untuk dijadikan objek penelitian,
7. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu dalam penyelesaian skripsi ini.

Semoga semua kebaikan yang telah diberikan kepada penulis segera dibalas oleh Allah SWT dengan kebaikan yang lebih. Semoga skripsi ini dapat memberikan manfaat kepada pembaca.

Malang, 30 Juli 2018

Penulis

afifahyaalif@ub.ac.id

ABSTRAK

Afifahya Alif Rizyomi, Analisis dan Perancangan Sistem Informasi Pemesanan Barang (Studi Kasus: PT. Trivia Nusantara)

Pembimbing: Mochamad Chandra Saputra , S.Kom., M.Eng. dan Welly Purnomo, S.T., M.Kom.

PT. Trivia Nusantara adalah suatu usaha di bidang jasa *advertising* dan *retail* alat tulis kantor. Setelah melakukan observasi didapatkan permasalahan dalam hal pemesanan barang. Yaitu kurangnya dukungan sistem informasi menyebabkan proses pemesanan barang menjadi lebih lama dan memakan banyak biaya dalam melakukan komunikasi. Semakin berkembangnya teknologi berdampak pada semakin ketatnya persaingan bisnis, hal ini memicu PT. Trivia Nusantara untuk membangun sistem informasi pemesanan barang agar dapat menunjang operasional perusahaan. Dikarenakan PT. Trivia Nusantara tidak memiliki tenaga ahli TI sehingga pimpinan perusahaan berencana membangun sistem informasi pemesanan barang dengan bekerja sama dengan vendor. Berdasarkan konsep SDLC (*Software Development Life Cycle*) hal awal yang dibutuhkan adalah sebuah analisis kebutuhan dan perancangan sistem yang terdokumentasi. Analisis dan perancangan sistem yang dilakukan menggunakan metodologi *Ripple* pada 4 fase awal. Pada fase *genesis* dan fase *requirement* dilakukan analisis kebutuhan. Sedangkan perancangan dilakukan pada fase *analysis* dan fase *design*. Pentingnya melakukan evaluasi kebutuhan maka pada penelitian ini evaluasi perancangan yang digunakan yaitu metode *consistency analysis* terbukti memiliki nilai presentase 100% konsisten dan termasuk kategori *correctness* sedangkan pada uji *correctness* yang membuktikan bahwa kebutuhan sistem dengan perancangan sistem bersifat konsisten dan benar.

Kata kunci: *Ripple, Consistency Analysis, Correctness*

ABSTRACT

Afifahlya Alif Rizyomi, Analisis dan Perancangan Sistem Informasi Pemesanan Barang (Studi Kasus: PT. Trivia Nusantara)

Pembimbing: Mochamad Chandra Saputra , S.Kom., M.Eng. dan Welly Purnomo, S.T., M.Kom.

PT. Trivia Nusantara is a business in advertising and retail of office stationery. After doing an observation, there is a problem in ordering products process. Which lack of information system support causes the ordering process of goods to be longer and takes a lot of cost in communication. Growth of technology impact on the increasingly business competition, this trigger PT. Trivia Nusantara to build an information system in order to support the company's operations. Due to PT. Trivia Nusantara also does not have an IT expert so the company's leader plans to build an information system by working with vendors. Based on the concept of SDLC (Software Development Life Cycle) the initial thing required is a documented needs analysis and design. System analysis and design performed using Ripple methodology in the initial 4 phases. Genesis phase and requirement phase used to requirements analysis. While the design in the analysis phase and design phase. The importance of evaluating the requirements, then in this study the design evaluation used consistency analysis method proved to have a 100% percentage value consistent and included the correctness category. And the correctness test proving that system requirements with system design is consistent and correct.

Keywords: *Ripple, Consistency Analysis, Corretness*

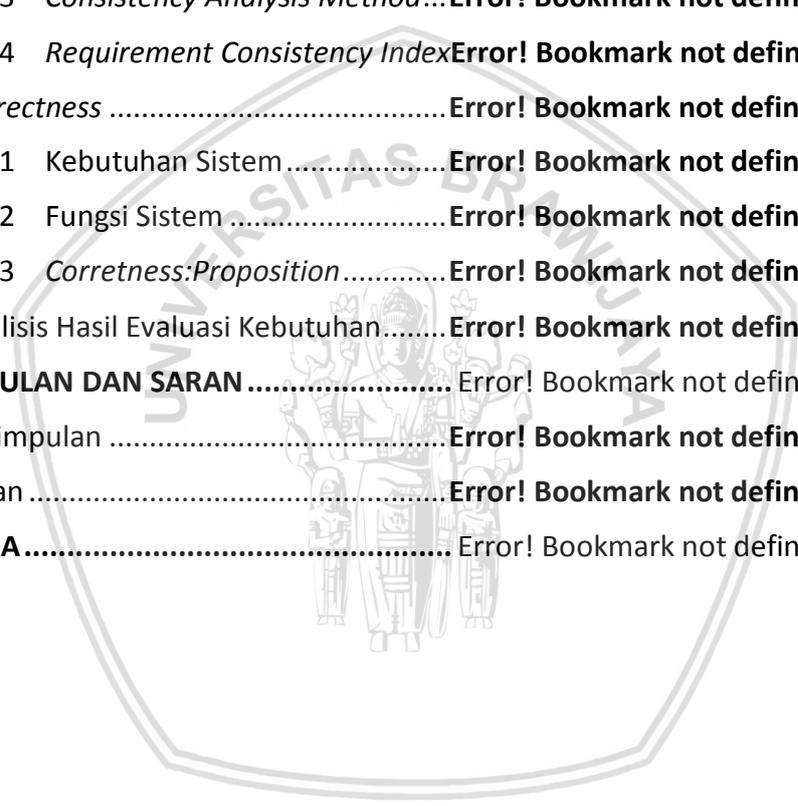
DAFTAR ISI

PENGESAHAN	2
PERNYATAAN ORISINALITAS	3
KATA PENGANTAR	4
ABSTRAK.....	5
ABSTRACT	6
DAFTAR ISI.....	7
DAFTAR TABEL.....	10
DAFTAR GAMBAR	11
DAFTAR LAMPIRAN.....	14
BAB 1 PENDAHULUAN.....	Error! Bookmark not defined.
1.1 Latar belakang	Error! Bookmark not defined.
1.2 Rumusan masalah.....	Error! Bookmark not defined.
1.3 Tujuan	Error! Bookmark not defined.
1.4 Manfaat	Error! Bookmark not defined.
1.5 Batasan masalah.....	Error! Bookmark not defined.
1.6 Sistematika pembahasan.....	Error! Bookmark not defined.
BAB 2 LANDASAN KEPUSTAKAAN.....	Error! Bookmark not defined.
2.1 Tinjauan Pustaka.....	Error! Bookmark not defined.
2.2 Profil PT. Trivia Nusantara	Error! Bookmark not defined.
2.2.1 Visi dan Misi	Error! Bookmark not defined.
2.2.2 Struktur Organisasi	Error! Bookmark not defined.
2.3 Pengertian Sistem Informasi	Error! Bookmark not defined.
2.4 Konsep Analisis dan Perancangan	Error! Bookmark not defined.
2.5 Pemrograman Berorientasi Obyek	Error! Bookmark not defined.
2.6 Metodologi <i>Ripple</i>	Error! Bookmark not defined.
2.7 Diagram UML.....	Error! Bookmark not defined.
2.7.1 <i>Use Case Diagram</i>	Error! Bookmark not defined.
2.7.2 <i>Activiy Diagram</i>	Error! Bookmark not defined.
2.7.3 <i>Communication Diagram</i>	Error! Bookmark not defined.
2.7.4 <i>Sequence Diagram</i>	Error! Bookmark not defined.
2.7.5 <i>Class Diagram</i>	Error! Bookmark not defined.



- 2.8 *Consistency Analysis: Requirement Configuration Structure* **Error! Bookmark not defined.**
- 2.9 *Correctness* **Error! Bookmark not defined.**
- BAB 3 METODOLOGI PENELITIAN** **Error! Bookmark not defined.**
- 3.1 *Studi Literatur* **Error! Bookmark not defined.**
- 3.2 *Observasi dan Pengumpulan Data* **Error! Bookmark not defined.**
 - 3.2.1 *Wawancara* **Error! Bookmark not defined.**
 - 3.2.2 *Observasi* **Error! Bookmark not defined.**
- 3.3 *Analisis Kebutuhan Sistem*..... **Error! Bookmark not defined.**
- 3.4 *Perancangan Sistem* **Error! Bookmark not defined.**
- 3.5 *Evaluasi Perancangan* **Error! Bookmark not defined.**
- 3.6 *Pengambilan Kesimpulan* **Error! Bookmark not defined.**
- BAB 4 ANALISIS DAN PERANCANGAN**..... **Error! Bookmark not defined.**
- 4.1 *User Story*..... **Error! Bookmark not defined.**
- 4.2 *Fase Genesis*..... **Error! Bookmark not defined.**
 - 4.2.1 *Projek Genesis*..... **Error! Bookmark not defined.**
 - 4.2.2 *Membuat Glossary*..... **Error! Bookmark not defined.**
- 4.3 *Fase Requirements*..... **Error! Bookmark not defined.**
 - 4.3.1 *Business Actor List*..... **Error! Bookmark not defined.**
 - 4.3.2 *Business Use Case List*..... **Error! Bookmark not defined.**
 - 4.3.3 *Business Activity Diagram*..... **Error! Bookmark not defined.**
 - 4.3.4 *Business Use Case Details* **Error! Bookmark not defined.**
 - 4.3.5 *User Interface Sketches*..... **Error! Bookmark not defined.**
 - 4.3.6 *System Actor List*..... **Error! Bookmark not defined.**
 - 4.3.7 *System Use Case List* **Error! Bookmark not defined.**
 - 4.3.8 *System Use Case Diagram* **Error! Bookmark not defined.**
 - 4.3.9 *System Use Case Details* **Error! Bookmark not defined.**
 - 4.3.10 *Supplementary Requirements*.. **Error! Bookmark not defined.**
- 4.4 *Fase Analysis*..... **Error! Bookmark not defined.**
 - 4.4.1 *Analysis Class Diagram* **Error! Bookmark not defined.**
 - 4.4.2 *Analysis Attributes List*..... **Error! Bookmark not defined.**
 - 4.4.3 *Communication Diagram* **Error! Bookmark not defined.**
 - 4.4.4 *Operation List*..... **Error! Bookmark not defined.**
- 4.5 *Fase Design* **Error! Bookmark not defined.**

4.5.1	<i>Class Diagram</i>	Error! Bookmark not defined.
4.5.2	<i>Sequence Diagram</i>	Error! Bookmark not defined.
4.5.3	<i>Database Schema</i>	Error! Bookmark not defined.
4.5.4	Penyelesaian <i>User Interface</i>	Error! Bookmark not defined.
BAB 5	EVALUASI PERANCANGAN	Error! Bookmark not defined.
5.1	<i>Consistency Analysis:Requirement Configuration Structure</i>	Error! Bookmark not defined.
5.1.1	<i>Layers and Configuration Items</i>	Error! Bookmark not defined.
5.1.2	<i>Requirement Configuration Structure</i>	Error! Bookmark not defined.
5.1.3	<i>Consistency Analysis Method</i> ...	Error! Bookmark not defined.
5.1.4	<i>Requirement Consistency Index</i>	Error! Bookmark not defined.
5.2	<i>Correctness</i>	Error! Bookmark not defined.
5.2.1	Kebutuhan Sistem	Error! Bookmark not defined.
5.2.2	Fungsi Sistem	Error! Bookmark not defined.
5.2.3	<i>Corretness:Proposition</i>	Error! Bookmark not defined.
5.3	Analisis Hasil Evaluasi Kebutuhan.....	Error! Bookmark not defined.
BAB 6	KESIMPULAN DAN SARAN	Error! Bookmark not defined.
6.1	Kesimpulan	Error! Bookmark not defined.
6.2	Saran	Error! Bookmark not defined.
DAFTAR PUSTAKA	Error! Bookmark not defined.



DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka 1	Error! Bookmark not defined.
Tabel 2.2 Tinjauan Pustaka 2	Error! Bookmark not defined.
Tabel 2.3 Tinjauan Pustaka 3	Error! Bookmark not defined.
Tabel 2.4 Artefak pada Metodologi <i>Ripple</i>	Error! Bookmark not defined.
Tabel 2.5 Penyesuaian Metodologi <i>Ripple</i> dengan Penelitian	Error! Bookmark not defined.
Tabel 2.6 Simbol <i>Use Case Diagram</i>	Error! Bookmark not defined.
Tabel 2.7 Notasi pada <i>Activity Diagram</i>	Error! Bookmark not defined.
Tabel 2.8 Notasi pada <i>Communication Diagram</i>	Error! Bookmark not defined.
Tabel 2.9 Notasi pada <i>Sequence Diagram</i>	Error! Bookmark not defined.
Tabel 2.10 Notasi pada <i>Class Diagram</i>	Error! Bookmark not defined.
Tabel 3.1 Penggunaan Metode <i>Ripple</i> pada Penelitian	Error! Bookmark not defined.
Tabel 4.1 <i>Glossary</i>	Error! Bookmark not defined.
Tabel 4.2 Deskripsi Fitur Sistem	Error! Bookmark not defined.
Tabel 4.3 Kebutuhan Fungsional Sistem	Error! Bookmark not defined.
Tabel 4.4 <i>System Supplementary Requirements</i>	Error! Bookmark not defined.
Tabel 4.5 <i>Analysis Attributes List</i>	Error! Bookmark not defined.
Tabel 4.6 <i>Operation List</i>	Error! Bookmark not defined.
Tabel 5.1 Masukan <i>Business Layer</i>	Error! Bookmark not defined.
Tabel 5.2 Masukan <i>Process Layer</i>	Error! Bookmark not defined.
Tabel 5.3 Masukan <i>Requirements Layer</i>	Error! Bookmark not defined.
Tabel 5.4 Masukan <i>Specification Layer</i>	Error! Bookmark not defined.
Tabel 5.5 Kebutuhan Sistem	Error! Bookmark not defined.
Tabel 5.6 Fungsi Sistem	Error! Bookmark not defined.

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi PT. Trivia Nusantara**Error! Bookmark not defined.**

Gambar 2.2 Contoh *Activity Diagram***Error! Bookmark not defined.**

Gambar 2.3 Contoh *Communication Diagram* Mengganti Password**Error! Bookmark not defined.**

Gambar 2.4 Contoh *Sequence Diagram* Administrasi Mahasiswa**Error! Bookmark not defined.**

Gambar 3.1 Tahapan Penelitian**Error! Bookmark not defined.**

Gambar 4.1 Aturan Penomoran *Business Use Case*.**Error! Bookmark not defined.**

Gambar 4.2 *Business Activity Diagram* untuk SIPB-BUC-01**Error! Bookmark not defined.**

Gambar 4.3 *Business Activity Diagram* untuk SIPB-BUC-02**Error! Bookmark not defined.**

Gambar 4.4 *Business Activity Diagram* untuk SIPB-BUC-03**Error! Bookmark not defined.**

Gambar 4.5 *Business Activity Diagram* untuk SIPB-BUC-04**Error! Bookmark not defined.**

Gambar 4.6 *User Interfaces Sketches 1 (Login)***Error! Bookmark not defined.**

Gambar 4.7 *User Interfaces Sketches 2 (Halaman Produk Barang)***Error! Bookmark not defined.**

Gambar 4.8 *User Interfaces Sketches 3 (Menampilkan Detail Barang)***Error! Bookmark not defined.**

Gambar 4.9 *User Interfaces Sketches 4 (Trolis Pelanggan)***Error! Bookmark not defined.**

Gambar 4.10 *User Interfaces Sketches 5 (Status Pesanan Pelanggan)***Error! Bookmark not defined.**

Gambar 4.11 *User Interfaces Sketches 6 (Profil PT. Trivia Nusantara)***Error! Bookmark not defined.**

Gambar 4.12 *User Interfaces Sketches 7 (Kontak PT. Trivia Nusantara)***Error! Bookmark not defined.**

Gambar 4.13 *User Interfaces Sketches 8 (Form Data Perusahaan Pelanggan)***Error! Bookmark not defined.**

Gambar 4.14 *User Interfaces Sketches 9 (Halaman Detail Pemesanan)***Error! Bookmark not defined.**

Gambar 4.15 *User Interfaces Sketches 10 (Halaman Verifikasi)***Error! Bookmark not defined.**

Gambar 4.16 *User Interfaces Sketches 11 (Halaman Produk Admin)***Error! Bookmark not defined.**

Gambar 4.17 *User Interfaces Sketches 12 (Menambahkan Produk)***Error! Bookmark not defined.**

Gambar 4.18 *User Interfaces Sketches 13 (Mengedit Produk)***Error! Bookmark not defined.**



Gambar 4.19 *User Interfaces Sketches 14* (Halaman Pesanan Admin)**Error! Bookmark not defined.**

Gambar 4.20 *User Interfaces Sketches 15* (Halaman Cetak Pesanan Admin)**Error! Bookmark not defined.**

Gambar 4.21 *User Interfaces Sketches 16* (Halaman Lembar *Purchase Order*)**Error! Bookmark not defined.**

Gambar 4.22 *User Interfaces Sketches 17* (Halaman Tagihan Admin)**Error! Bookmark not defined.**

Gambar 4.23 *User Interfaces Sketches 18* (Halaman Cetak Tagihan Admin)**Error! Bookmark not defined.**

Gambar 4.24 *User Interfaces Sketches 19* (Surat Penagihan) ...**Error! Bookmark not defined.**

Gambar 4.25 *User Interfaces Sketches 20* (Halaman Utama Bagian Gudang)**Error! Bookmark not defined.**

Gambar 4.26 *User Interfaces Sketches 21* (Halaman Mengubah Status Pesanan)**Error! Bookmark not defined.**

Gambar 4.27 *User Interfaces Sketches 22* (Halaman Cetak *Purchase Order* Bag. Gudang)**Error! Bookmark not defined.**

Gambar 4.28 *User Interfaces Sketches 23* (Halaman Cetak Tagihan Bag. Gudang)**Error! Bookmark not defined.**

Gambar 4.29 Aturan Penomoran *System Use Case*.....**Error! Bookmark not defined.**

Gambar 4.30 *Use Case Diagram***Error! Bookmark not defined.**

Gambar 4.31 *Analysis Class Diagram***Error! Bookmark not defined.**

Gambar 4.32 *Communication Diagram* untuk *Use Case* Buat Akun : (SIPB-F-01).....**Error! Bookmark not defined.**

Gambar 4.33 *Communication Diagram* untuk *Use Case Login* : (SIPB-F-02)**Error! Bookmark not defined.**

Gambar 4.34 *Communication Diagram* untuk *Use Case Tambah Produk* : (SIPB-F-03)...**Error! Bookmark not defined.**

Gambar 4.35 *Communication Diagram* untuk *Use Case Edit Produk* : (SIPB-F-04).....**Error! Bookmark not defined.**

Gambar 4.36 *Communication Diagram* untuk *Use Case Hapus Produk* : (SIPB-F-05).....**Error! Bookmark not defined.**

Gambar 4.37 *Communication Diagram* untuk *Use Case Lihat informasi Produk* : (SIPB-F-06)**Error! Bookmark not defined.**

Gambar 4.38 *Communication Diagram* untuk *Use Case Cari Produk* : (SIPB-F-07).....**Error! Bookmark not defined.**

Gambar 4.39 *Communication Diagram* untuk *Use Case Pesan Barang* : (SIPB-F-08).....**Error! Bookmark not defined.**

Gambar 4.40 *Communication Diagram* untuk *Use Case* Konfirmasi Pesanan : (SIPB-F-09)
.....**Error! Bookmark not defined.**

Gambar 4.41 *Communication Diagram* untuk *Use Case* Lihat PO : (SIPB-F-010) **Error!
Bookmark not defined.**

Gambar 4.42 *Communication Diagram* untuk *Use Case* Cetak PO : (SIPB-F-11)..... **Error!
Bookmark not defined.**

Gambar 4.43 *Communication Diagram* untuk *Use Case* Lihat Rekap Penjualan : (SIPB-F-12)
.....**Error! Bookmark not defined.**

Gambar 4.44 *Communication Diagram* untuk *Use Case* Cetak Rekap Penjualan : (SIPB-F-13)
.....**Error! Bookmark not defined.**

Gambar 4.45 *Communication Diagram* untuk *Use Case* Status PO : (SIPB-F-14)..... **Error!
Bookmark not defined.**

Gambar 4.46 *Communication Diagram* untuk *Use Case* Lihat Tagihan : (SIPB-F-15).....**Error!
Bookmark not defined.**

Gambar 4.47 *Communication Diagram* untuk *Use Case* Cetak Tagihan : (SIPB-F-07-02) **Error!
Bookmark not defined.**

Gambar 4.48 *Class Diagram***Error! Bookmark not defined.**

Gambar 4.49 *Sequence Diagram* untuk *Use Case* Buat Akun : (SIPB-F-01)**Error! Bookmark
not defined.**

Gambar 4.50 *Sequence Diagram* untuk *Use Case* Login : (SIPB-F-02)**Error! Bookmark not
defined.**

Gambar 4.51 Berikut ini adalah *Sequence Diagram* untuk *Use Case* Tambah Produk : (SIPB-F-03)
.....**Error! Bookmark not defined.**

Gambar 4.52 *Sequence Diagram* untuk *Use Case* Edit Produk : (SIPB-F-04)**Error! Bookmark
not defined.**

Gambar 4.53 *Sequence Diagram* untuk *Use Case* Hapus Produk : (SIPB-F-05)**Error! Bookmark
not defined.**

Gambar 4.54 *Sequence Diagram* untuk *Use Case* Lihat informasi Produk : (SIPB-F-06)..**Error!
Bookmark not defined.**

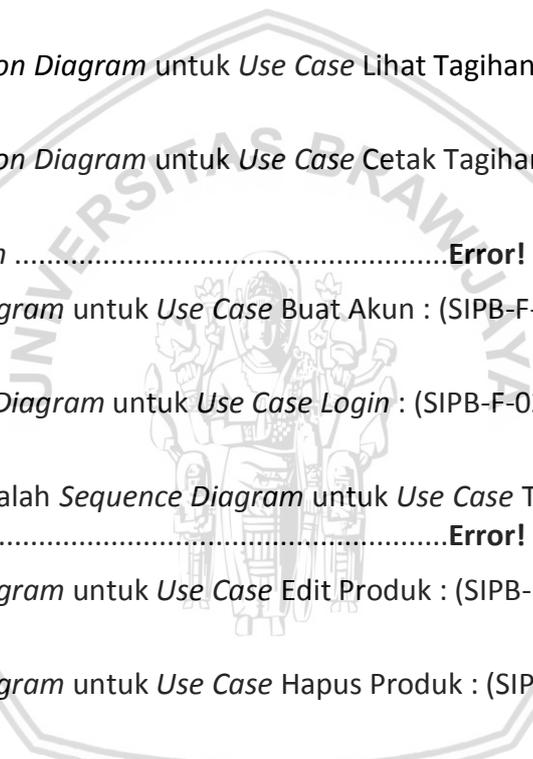
Gambar 4.55 *Sequence Diagram* untuk *Use Case* Cari Produk : (SIPB-F-07)**Error! Bookmark
not defined.**

Gambar 4.56 *Sequence Diagram* untuk *Use Case* Konfirmasi Pesanan : (SIPB-F-09)..... **Error!
Bookmark not defined.**

Gambar 4.57 *Sequence Diagram* untuk *Use Case* Pesan Barang : (SIPB-F-08)**Error! Bookmark
not defined.**

Gambar 4.58 *Sequence Diagram* untuk *Use Case* Lihat PO : (SIPB-F-10)**Error! Bookmark not
defined.**

Gambar 4.59 *Sequence Diagram* untuk *Use Case* Cetak PO : (SIPB-F-11)**Error! Bookmark not
defined.**



Gambar 4.60 *Sequence Diagram* untuk *Use Case* Lihat Rekap PO : (SIPB-F-12) **Error! Bookmark not defined.**

Gambar 4.61 *Sequence Diagram* untuk *Use Case* Cetak Rekap PO : (SIPB-F-13) **Error! Bookmark not defined.**

Gambar 4.62 *Sequence Diagram* untuk *Use Case* Ubah Status PO : (SIPB-F-14) **Error! Bookmark not defined.**

Gambar 4.63 *Sequence Diagram* untuk *Use Case* Lihat Tagihan : (SIPB-F-15) **Error! Bookmark not defined.**

Gambar 4.64 *Sequence Diagram* untuk *Use Case* Cetak Tagihan: (SIPB-F-16) **Error! Bookmark not defined.**

Gambar 4.65 *Database Schema*..... **Error! Bookmark not defined.**

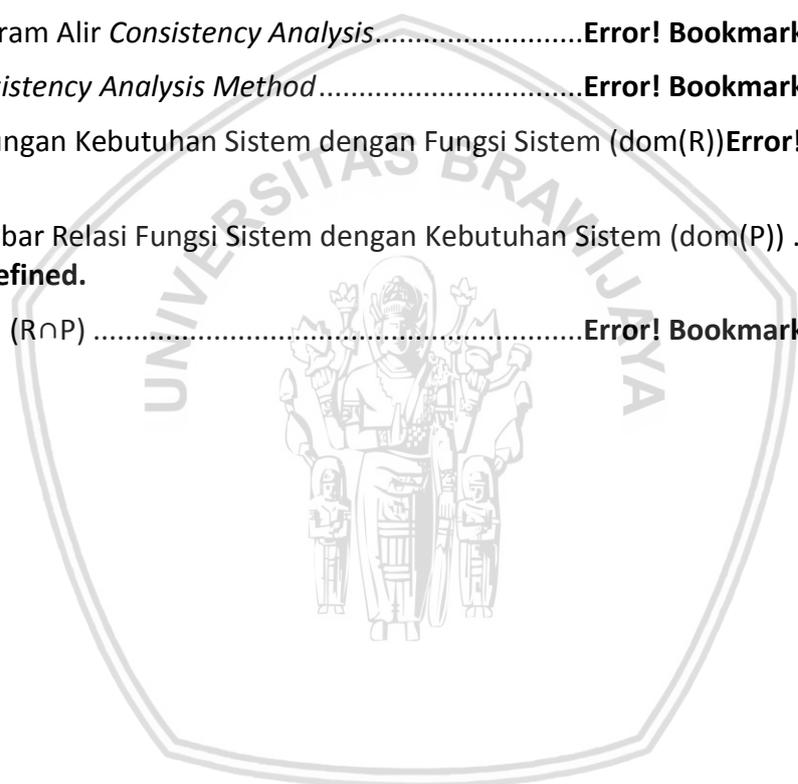
Gambar 5.1 Diagram Alir *Consistency Analysis*..... **Error! Bookmark not defined.**

Gambar 5.2 *Consistency Analysis Method* **Error! Bookmark not defined.**

Gambar 5.3 Hubungan Kebutuhan Sistem dengan Fungsi Sistem (dom(R)) **Error! Bookmark not defined.**

Gambar 5.4 Gambar Relasi Fungsi Sistem dengan Kebutuhan Sistem (dom(P)) **Error! Bookmark not defined.**

Gambar 5.5 Hasil (R \cap P) **Error! Bookmark not defined.**



DAFTAR LAMPIRAN

LAMPIRAN 1 WAWANCARA.....**Error! Bookmark not defined.**



BAB 1 PENDAHULUAN

1.1 Latar belakang

Dengan adanya perkembangan teknologi informasi yang pesat, berdampak pada semakin ketatnya persaingan bisnis yang ada. PT. Trivia Nusantara merupakan perusahaan yang bergerak di bidang jasa *advertising* dan penyedia alat tulis kantor. Berdasarkan hasil wawancara diketahui bahwa terdapat kegiatan operasional dilakukan secara konvensional yang sebenarnya kegiatan tersebut dapat dibantu dengan sebuah sistem informasi seperti pencatatan penjualan, pemesanan barang dari pelanggan dan administrasi pemesanan barang. Tidak semua kegiatan operasional didukung oleh sistem informasi.

Setelah melakukan observasi diketahui PT. Trivia Nusantara tidak memiliki sebuah divisi IT, hal itu menyebabkan operasional perusahaan hanya menggunakan *Microsoft Office* sebagai penunjang kinerja perusahaan. Selama ini pemesanan barang dilakukan via telepon setelah itu melakukan verifikasi kepada bagian keuangan pelanggan melalui telepon, hal ini dianggap kurang efektif dikarenakan selain memakan waktu yang lama juga menggunakan anggaran biaya yang cukup besar. PT. Trivia Nusantara memiliki keinginan untuk mengubah kegiatan operasional yang bersifat konvensional menjadi berbasis sistem informasi dikarenakan adanya tuntutan dari pelanggan agar dapat melakukan proses pemesanan barang lebih cepat dan efektif.

PT. Trivia Nusantara bekerja sama dengan vendor untuk mengembangkan sistem pemesanan barang yang baru. Dalam proses mengembangkan sistem informasi pemesanan barang maka perlu dilakukan tahapan pengembangan sistem sesuai dengan aktivitas dalam *Software Development Life Cycle* (SDLC). Berdasarkan konsep SDLC hal awal yang dibutuhkan adalah sebuah analisis kebutuhan dan perancangan sistem yang terdokumentasi. Hal ini bermaksud untuk memberikan gambaran sistem seperti apa yang diinginkan perusahaan.

Menurut Whitten, dan Bentley (2007) analisis kebutuhan sistem merupakan fase paling kritis dari sebuah proyek. Analisis kebutuhan merupakan kegiatan yang wajib dilakukan karena apabila tidak, dapat menyebabkan kegagalan dalam memenuhi kebutuhan organisasi atau pengguna. Menurut narasumber perlu ada bantuan pihak lain dalam melakukan analisis kebutuhan dikarenakan pihak organisasi merasa kesulitan dalam melakukan pendefinisian kebutuhan. Dengan adanya analisis kebutuhan diharapkan pihak organisasi dapat menambah pandangan sistem seperti apa yang ingin dibangun.

Setelah melakukan analisis kebutuhan, tahap yang dilakukan adalah perancangan, yang mana digunakan untuk memodelkan dan memberikan gambaran rinci dari kebutuhan sistem yang akan dikembangkan. Mengimplementasi sistem tanpa melalui analisis dan perancangan yang tepat dapat mengakibatkan ketidakpuasan pengguna dan sering menjadikan sistem tidak lagi digunakan (Kendall, 2011). Menurut Mitch Kapor dalam Pressman (2010) menjelaskan bahwa perancangan perangkat lunak merupakan tempat kualitas perangkat lunak ditetapkan.

Dalam melakukan analisis dan perancangan terdapat beberapa pendekatan, pendekatan yang digunakan dalam penelitian ini menggunakan pendekatan berorientasi objek (*Object Oriented Analysis and Design*). Dikarenakan pendekatan berorientasi objek merupakan sebuah pendekatan untuk memikirkan suatu masalah dengan menggunakan model yang

dibuat menurut konsep sekitar dunia nyata (Kendall, 2011). Pendekatan OOAD yang digunakan menggunakan metodologi *Ripple*. Alasan menggunakan metodologi *Ripple* adalah dengan menggunakan metode ini pemangku kepentingan dapat terlibat dalam pengembangan di tahap awal, sehingga dapat meningkatkan komunikasi antara pengembang dan perusahaan. Selain itu metodologi *Ripple* memiliki *output* yang jelas pada setiap fase sehingga dapat mengetahui apa yang harus dilakukan setiap tahapnya. *Ripple* menggabungkan tahapan dalam SDLC dengan notasi UML (*Unified Modelling Language*). Dikarenakan penelitian ini hanya sampai pada analisis dan perancangan maka penggunaan metodologi *Ripple* dalam penelitian ini terdiri dari empat fase yaitu *Genesis*, *Requirement*, *Analysis* dan *Design*.

Berdasarkan data survei yang dilakukan oleh Standish Group, alasan umum kegagalan sebuah proyek bukanlah karena hal teknis tapi terletak pada aspek non-teknis yaitu tahapan analisis kebutuhan. Oleh karena itu diperlukan sebuah evaluasi perancangan terhadap analisis kebutuhan, pada penelitian ini evaluasi yang digunakan adalah *Consistency Analysis* dan *Correctness*. *Consistency Analysis* digunakan untuk melakukan analisis konsistensi pada hasil perancangan sistem dengan pemanfaatan hubungan antar elemen perancangan dan mengukur prosentase konsistensi antar kebutuhan, proses bisnis, proses dan spesifikasi yang ada pada perancangan sistem (Nistala dan Kumari, 2013). Sedangkan uji *Correctness* digunakan untuk melakukan koreksi terhadap kesesuaian spesifikasi kebutuhan dari sistem dengan fungsi dari sistem yang akan dikembangkan (Mili dan Tchier, 2015).

Atas dasar uraian di atas, penulis mengambil topik penelitian analisis perancangan sistem informasi berjudul "Analisis Dan Perancangan Sistem Informasi Pemesanan Barang (Studi Kasus : PT. Trivia Nusantara Surabaya)". Penelitian ini bertujuan untuk membantu menganalisis kebutuhan dan mendokumentasikan perancangan sistem. Harapan dari penelitian ini adalah untuk memberikan solusi dan jawaban atas permasalahan yang dihadapi PT. Trivia Nusantara berupa hasil analisis kebutuhan dan sebuah perancangan sistem untuk pemesanan barang.

1.2 Rumusan masalah

Dari uraian latar belakang di atas, rumusan masalah untuk penelitian ini adalah:

1. Bagaimana hasil penerapan fase *Genesis* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara ?
2. Bagaimana hasil penerapan fase *Requirements* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara ?
3. Bagaimana hasil penerapan fase *Analysis* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara ?
4. Bagaimana hasil penerapan fase *Design* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara ?

5. Bagaimana hasil evaluasi perancangan terhadap analisis kebutuhan menggunakan *Consistency Analysis* dan *Correcteness* pada Sistem Informasi Pemesanan Barang pada PT. Trivia Nusantara?

1.3 Tujuan

Dari rumusan masalah diatas dapat disimpulkan bahwa tujuan penelitian ini adalah :

1. Melakukan penerapan fase *Genesis* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara.
2. Melakukan penerapan fase *Requirements* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara.
3. Melakukan penerapan fase *Analysis* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara.
4. Melakukan penerapan fase *Design* pada metodologi *Ripple* dalam melakukan analisis dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara.
5. Melakukan evaluasi perancangan terhadap analisis kebutuhan menggunakan *Consistency Analysis* dan *Correcteness* pada Sistem Informasi Pemesanan Barang pada PT. Trivia Nusantara.

1.4 Manfaat

Manfaat yang diharapkan pada penelitian ini adalah didapatkan dokumen kebutuhan dan perancangan sistem informasi pemesanan barang pada PT. Trivia Nusantara yang dapat digunakan sebagai dasar pembangunan sistem oleh vendor.

1.5 Batasan masalah

Batasan masalah dari penelitian ini adalah

1. Melakukan analisis dan perancangan akan disesuaikan dengan prosedur pengelolaan pemesanan barang yang diterapkan pada PT. Trivia Nusantara.
2. Analisis dan perancangan sistem akan dilakukan dengan metode OOAD (*Object Oriented Analysis and Design*) menggunakan metodologi *Ripple* fase *Genesis*, *Requirements*, *Analysis* dan *Design*.
3. Arsitektur sistem yang digunakan berorientasi obyek, sehingga dimodelkan menggunakan diagram UML (*Unified Modelling Language*) meliputi *use case diagram*, *activity diagram*, *class diagram*, *sequence diagram* dan *communication diagram*.
4. Metode yang digunakan untuk evaluasi kebutuhan terhadap proses manajemen pembelian pemesanan barang dan fungsi sistem yang dirancang adalah metode *consistency analysis*.
5. Perancangan yang dibuat akan diuji dengan menggunakan *correcteness* untuk mengetahui sejauh mana implementasi penuh dan fungsi-fungsi yang diperlukan telah tercapai.
6. Data yang diperoleh merupakan hasil wawancara dan observasi dengan calon pengguna sistem dan *owner*, hasil penelitian dan literatur yang berkaitan dengan penelitian.

7. Hasil dari pengerjaan berupa dokumentasi kebutuhan pengguna dan perancangan sistem manajemen pemesanan barang.

1.6 Sistematika pembahasan

Dalam penyusunan proposal skripsi ini, terdapat tujuh bab dengan pokok bahasan yang sudah ditentukan. Seperti berikut:

Bab I Pendahuluan

Berisi tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan.

Bab II Landasan Kepustakaan

Berisi tentang pembahasan mengenai teori-teori yang menjadi landasan dalam melakukan penelitian.

Bab III Metodologi Penelitian

Berisi tentang bagaimana proses pengerjaan penelitian mulai dari awal hingga akhir penelitian.

Bab IV Analisis dan Perancangan

Berisi tentang pembahasan analisis ruang lingkup sistem yang akan digunakan, analisis permasalahan yang timbul pada kondisi saat ini, dan analisis kebutuhan sistem sesuai dengan hasil wawancara, observasi dan metode pengumpulan data lainnya. Serta berisi tentang proses representasi hasil dari analisis kebutuhan sistem ke dalam perancangan sistem informasi pemesanan barang. Hasil dari analisis kebutuhan akan dimodelkan dengan bentuk diagram UML.

BAB VI Evaluasi Perancangan

Berisi tentang pembahasan hasil dari evaluasi perancangan sistem terhadap kebutuhan dari PT. Trivia Nusantara yang dibangun.

BAB VI Kesimpulan dan Saran

Berisi tentang kesimpulan dan saran yang akan diperoleh dari analisis hasil perancangan sistem informasi pemesanan barang.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Tinjauan pustaka yang berkaitan dengan penelitian ini adalah penelitian yang dilakukan oleh Anggit Wirasto pada tahun 2013 dengan judul “Perancangan Sistem Informasi Penanggulangan Aids Menggunakan Pendekatan Berorientasi Objek” . Metodologi dan hasil penelitian ini terdapat pada tabel 2.1.

- a. **Tabel Error! No text of specified style in document..1 Tinjauan Pustaka 1**

Masalah	Tujuan
Penelitian ini berdasar pada permasalahan mengenai pelaporan penyuluhan AIDS yang memiliki format berbeda dan memakan waktu yang lama. Sistem Informasi Penanggulangan AIDS menerapkan prinsip pendekatan berorientasi objek, sehingga dapat Metode pengembangan yang digunakan adalah metode Ripple dengan fokus utama pada tahap analisis dan perancangan.	Penelitian ini bertujuan untuk mengetahui pola rancangan apa saja yang dapat diterapkan pada sistem tersebut dan menjelaskan bahwa dengan menggunakan metode Ripple memberikan keuntungan dimana dapat memisahkan logika program dan antar muka pengguna agar struktur program jadi mudah dimengerti dan mudah melakukan <i>maintenance</i> .

Penelitian yang berkaitan dengan penelitian ini adalah penelitian yang dilakukan oleh Widyasari Puspa Permata Witra pada tahun 2017 dengan judul “Analisis dan Perancangan Sistem Informasi Pengelolaan Skripsi dan Praktek Kerja Lapangan”. Metodologi dan hasil penelitian ini terdapat pada tabel 2.2

- b. **Tabel Error! No text of specified style in document..2 Tinjauan Pustaka 2**

Masalah	Tujuan
Penelitian ini berdasar pada permasalahan untuk melakukan <i>controlling</i> mahasiswa yang belum mengumpulkan laporan skripsi dan praktik kerja nyata. Metode perancangan yang digunakan adalah metode <i>Ripple</i> , dari fase <i>genesis</i> sampai <i>design</i> .	Penelitian ini bertujuan untuk memberikan rancangan sistem agar dapat mempermudah dalam melakukan <i>monitoring</i> pengumpulan skripsi dan praktik kerja nyata. Metode <i>Ripple</i> yang digunakan pada penelitian ini digunakan untuk melakukan analisis kebutuhan dan merancang sistem informasi.

Penelitian selanjutnya dari Kamalrudin & Sidek pad tahun 2015 berjudul “*A Review On Software Requirements and Consistency Management*” yang dijelaskan pada tabel 2.2 kajian pustaka 2. Penelitian tersebut bertujuan untuk meninjau terhadap definisi 3C yang merupakan *correctness*, *completeness*, dan

correctness terhadap kebutuhan sistem. Metodologi dan hasil penelitian ini terdapat pada tabel 2.3

c. **Tabel Error! No text of specified style in document..3 Tinjauan Pustaka 3**

Metodologi	Hasil
<p>Penelitian ini bertujuan untuk memahami definisi dari 3C. Selanjutnya penelitian ini menjelaskan mengenai tinjauan menyeluruh terkait teknik pengelolaan konsistensi yang telah diidentifikasi. Penelitian ini mengidentifikasi berbagai kesenjangan yang ada dalam proses memvalidasi dan mengelola konsistensi kebutuhan untuk menghindari penemuan kembali. Penelitian ini didukung dengan representasi <i>heat map</i> terkait jenis kontribusi, teknik, spesifikasi dan semantik yang digunakan dalam manajemen konsistensi.</p>	<p>Hasil dari penelitian tersebut berupa pembahasan gagasan 3C konsistensi, kebenaran, dan kelengkapan. Dalam pengelolaan konsisten <i>heat map</i> digunakan untuk menyajikan grafik dan dilakukan penyederhanaan dan mengelompokkan jenis kontribusi, spesifikasi, semantik dan teknik yang digunakan dalam konsistensi. Selain itu juga membandingkan pendekatan yang ada dan mengidentifikasi kelemahan dan kelebihan pendekatan tersebut.</p>

2.2 Profil PT. Trivia Nusantara

PT. Trivia Nusantara berdiri pada tahun 1996, sebelumnya perusahaan ini bernama PT. Saharta Mahardika. Perusahaan ini bergerak di bidang alat tulis perkantoran. Namun seiring dengan berjalannya waktu, perusahaan semakin berkembang dan bergerak dalam bidang penyediaan jasa percetakan dan *advertising*.

2.2.1 Visi dan Misi

Berikut adalah visi dan misi dari PT. Trivia Nusantara

Visi :

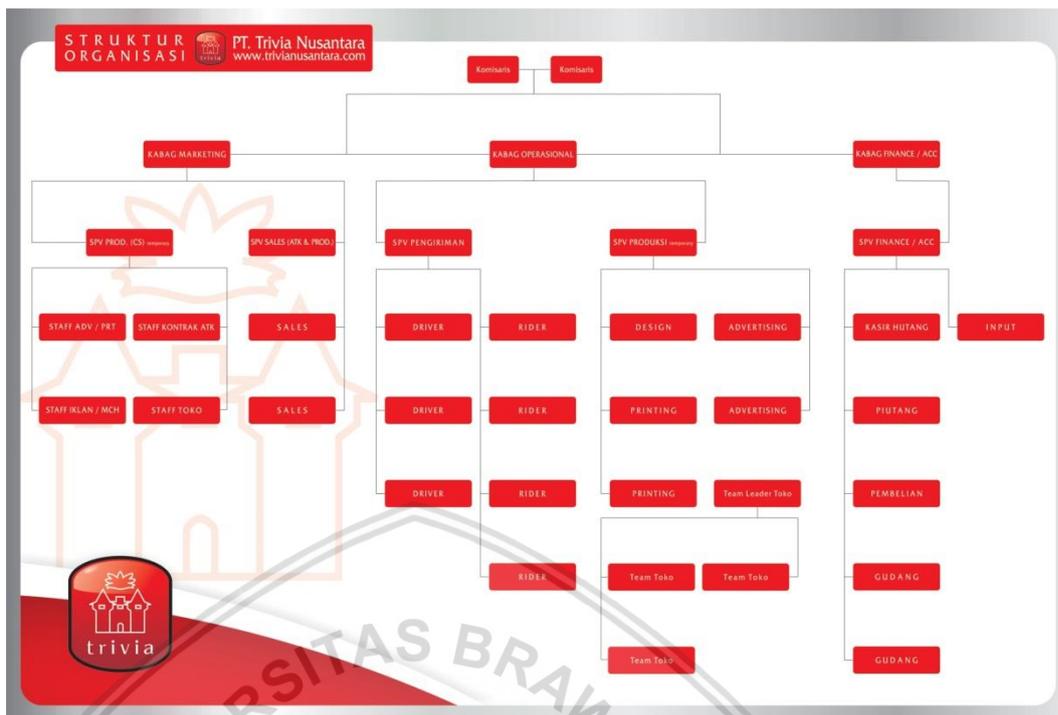
Sebagai sebuah perusahaan berskala nasional yang dibangun oleh profesional *team work* didalamnya.

Misi :

Mensuplai kebutuhan kantor konsumen secara aktif, *men-support* konsumen sebagai *partner* niaga serta memberikan optimal *service* kepada konsumen.

2.2.2 Struktur Organisasi

Struktur organisasi dari PT. Trivia Nusantara ada pada Gambar 2.1



d. **Gambar Error! No text of specified style in document..1 Struktur Organisasi PT. Trivia Nusantara**

2.3 Pengertian Sistem Informasi

Menurut O'Brien pada tahun 2010, sistem informasi merupakan kombinasi sumber daya yang terorganisir dari manusia, perangkat keras, piranti lunak, jaringan komputer, dan data yang mengumpulkan, mengubah, dan mendistribusikan informasi pada suatu organisasi. Sistem informasi menggunakan sumber daya manusia, *hardware*, *software* data jaringan, dan aktivitas pengendalian yang mengubah data menjadi informasi.

Sistem informasi adalah sekumpulan komponen yang saling terhubung yang mengumpulkan, memanipulasi, menyimpan, menyebarkan data dan informasi, dan memberikan mekanisme umpan balik untuk mencapai tujuan. Mekanisme umpan balik tersebut yang dapat membantu organisasi untuk mencapai tujuan mereka, seperti meningkatkan keuntungan atau meningkatkan pelayanan pada pelanggan. Sistem informasi dapat digunakan pada bidang bisnis untuk meningkatkan pendapatan dan mengurangi biaya (Stair dan Renolds, 2010). Sistem informasi memberikan informasi kepada semua tingkat manajemen, mulai dari manajemen tingkat bawah, manajemen tingkat menengah, hingga manajemen tingkat atas.

Sistem Informasi menurut John Burch dan Gary Grudnitski (1986) terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*). Terdapat enam blok yang harus saling berinteraksi satu sama lain untuk mencapai sasaran dalam satu kesatuan antara lain :

- Blok Masukan
Input mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.
- Blok Model
Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.
- Blok Keluaran
Produk yang dihasilkan dari sistem informasi adalah keluaran yang merupakan informasi yang baik serta bermanfaat dan dokumentasi yang berguna untuk semua tingkatan manajemen semua pemakai sistem.
- Blok Basis Data
Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan untuk mengakses atau memanipulasinya digunakan perangkat lunak yang disebut dengan DBMS (*Database Management Systems*). Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Perlu dilakukan pengorganisasian terhadap basis data yang ada agar informasi yang dihasilkannya baik dan efisiensi kapasitas penyimpanannya.
- Blok Kendali
Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, kecurangan-kecurangan, kegagalan-kegagalan yang terjadi di dalam sistem, ketidakefisienan, sabotase, dan lain sebagainya. Sehingga beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun dapat langsung segera diperbaiki jika seandainya hal-hal yang disebutkan di atas terjadi.

2.4 Konsep Analisis dan Perancangan

2.4.1 Konsep Analisis Sistem

Menurut Pressman (2010) analisis sistem adalah kegiatan menemukan atau mengidentifikasi masalah, mengevaluasi, membuat model serta membuat spesifikasi sistem. Dalam melakukan analisis ada beberapa langkah yang harus dilewati yaitu :

1. *Identify* merupakan sebuah proses mengenal masalah yang ada. Mengapa sebuah organisasi membutuhkan sebuah sistem atau masalah apa yang membuat organisasi membuat sebuah sistem.
2. *Understand* merupakan proses bagaimana memahami permasalahan dan memahami kebutuhan pengguna dan bisa memilah mana kebutuhan primer dan sekunder dari sistem
3. *Analyse* adalah menganalisis hasil dari kebutuhan yang diberikan pengguna apakah sudah sesuai dengan kebutuhan organisasi dan manajemen organisasi itu sendiri.

4. *Report* dengan kata lain laporan hasil diserahkan pada *Steering Committee* yang nantinya akan diteruskan ke manajemen. Pihak manajemen bersama-sama dengan *Steering Committee* dan pemakai sistem akan mempelajari temuan-temuan dan analisis yang telah dilakukan oleh analisis sistem yang disajikan dalam laporan ini.

2.4.2 Konsep Perancangan Sistem

Perancangan adalah sebuah proses mendefinisikan sesuatu yang akan dikerjakan dengan teknik yang bermacam-macam serta didalamnya berisi tentang deskripsi arsitektur komponen, detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya (Rizky,2011). Menurut Pressman pada tahun 2010, perancangan perangkat lunak adalah proses dimana analisis diterjemahkan menjadi sebuah “*blueprint*” untuk membangun perangkat lunak. Awalnya *blueprint*, menggambarkan pandangan menyeluruh perangkat lunak, yaitu desain diwakili pada abstraksi tingkat tinggi yang dapat langsung ditelusuri pada sistem tertentu, objektif, data yang lebih rinci, fungsional, dan perilaku persyaratan.

2.5 Pemrograman Berorientasi Obyek

Metodologi berorientasi obyek merupakan strategi pembangunan sistem yang mengorganisasikan sistem sebagai kumpulan obyek yang berisi data dan operasi yang diberlakukan terhadapnya (Sukamto dan Shalahuddin, 2014). Sedangkan menurut Kendall dan Kendall pada tahun 2011, *Object Oriented Analysis and Design* merupakan sebuah pendekatan untuk memikirkan sesuatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata.

Menurut Rosa dan Shalahudin pada tahun 2015 ada beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek :

1. *Class*, kelas adalah kumpulan objek-objek dengan karakteristik yang sama, secara teknis kelas adalah sebuah struktur tertentu dalam pembuatan perangkat lunak.
2. *Object*, objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstrak.
3. *Method*, *method* merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek.
4. *Attribute*, atribut adalah variabel global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek.
5. Antarmuka (*Interface*), pada umumnya digunakan agar kelas yang lain tidak mengakses langsung ke suatu kelas, namun hanya mengakses antarmukanya saja.
6. Generalisasi dan Spesialisasi, menunjukkan hubungan antar kelas dan objek yang umum dengan kelas dan objek yang khusus.

7. Komunikasi antar Objek, komunikasi antar-objek dilakukan lewat pesan yang dikirim dan satu objek ke objek lainnya.
8. *Package, package* adalah sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam *package* yang berbeda.

2.6 Metodologi Ripple

Metodologi *Ripple* adalah suatu metodologi yang diturunkan dari metodologi RUP (*Rational Unified Process*) dan lebih sederhana daripada metodologi RUP (O'Docherty, 2005) Pembangunan sebuah sistem dengan menggunakan metodologi *Ripple* terdiri dari sembilan fase, yaitu : *genesis phase, requirement phase, analysis phase, design phase, class specification phase, implementation phase, testing phase, deployment phase, maintenance phase*. Masing-masing fase memiliki beberapa langkah dan beberapa artefak yang dihasilkan sebagai keluaran. Pada tabel 2.4 pada kolom *phase* akan menjelaskan fase-fase yang terdapat dalam metodologi *Ripple*, pada kolom *artifacts* akan menjelaskan artefak apa saja yang dihasilkan dalam setiap fase dan pada kolom UML akan menjelaskan artefak mana saja yang termasuk ke dalam diagram UML.

e. **Tabel Error! No text of specified style in document..4 Artefak pada Metodologi Ripple**

Artifacts by phase			
Phase		Artifacts	UML
Genesis		Mission statement or informal requirement	No
		Roles	No
		Responsibilities	No
		Project Plan	No
		Work book	No
		Glossary	No
		Test plan	No
Requirements	Business	Actor list	No
		Use case list	No
		Use case detail	No
		Activity diagram	Yes
		Communication diagram	Yes
	System	Actor list	No
		Use case list	No
		Use case detail	No
		Use case diagram	Yes
		Use case survey	No
	UI sketches	No	
Analysis		Class diagram	Yes
		Communication diagram	Yes
Design	System	Deployment diagram	Yes
		Layer diagram	No

Artifacts by phase			
Phase	Artifacts		UML
	Subsystem	Class diagram	Yes
		Sequence diagram	Yes
		Database diagram	No
Class Specification	Comments	No	
Implementation	Source code	No	
Testing	Test reports	No	
Deployment	Shrink wrapped solution	No	
	Manuals	No	
	Training materia	No	
Maintenance	Fault reports	No	
	Increment plans	No	

Sumber : O'Dochtery (2005)

Berikut ini tahapan-tahapan yang dilakukan dalam penerapan fase Ripple dalam penelitian :

- *Genesis*
 - a. *Project Genesis* adalah tahapan untuk mendapatkan ide apa yang diinginkan oleh pelanggan atau mendapatkan apa yang dibutuhkan pelanggan serta mendapatkan dokumentasi kebutuhan dari pelanggan dan menghasilkan sebuah informal dokumen kebutuhan pelanggan.
 - a. *Producing a Glossary*, adalah tahapan untuk membuat sebuah *glossary* yang digunakan untuk mencatat definisi dari terminologi proyek dan memperbarui *glossary* selama melakukan pengembangan. Dalam sebuah *glossary* terdapat beberapa *relationship*, berikut ini adalah definisi dari setiap *relationship* dalam *glossary* :
 - *Business Actor* : Aktor yang menjalankan proses bisnis
 - *Business Object*: Objek yang muncul pada sub-fase *business*.
 - *System Actor*: Aktor yang berinteraksi dengan sistem
 - *System Object*: Objek yang muncul pada sub-fase *system*.
 - *Analysis Object*: Objek yang muncul di *analysis model*
 - *Deployment Artifact*: Sesuatu yang dihasilkan dalam sistem (sebuah file)
 - *Design Object*: Objek yang muncul pada fase *design*.
- *Business Requirement*
 - a. Membuat *business actor list* (dalam bentuk deskripsi).
 - b. Membuat *business use case list* (dalam bentuk deskripsi).
 - c. Mengilustrasikan *business use case* menggunakan *activity diagram (optional)*.
 - d. Mengilustrasikan *business use case* menggunakan *communication diagram (optional)*.
 - e. Membuat *use case detail*.

- *System Requirements (with pelanggan)*
 - a. Melakukan *brainstorm* dengan menggunakan sketsa UI.
 - b. Membuat *system actor list* (dalam bentuk deskripsi).
 - c. Membuat *system use case diagram*.
 - d. Membuat *system use case details*.

Use case details digunakan untuk mendeskripsikan bagaimana suatu sistem dapat berinteraksi dengan aktor. Dalam *Ripple* terdapat urutan yang digunakan untuk membuat sebuah *use case details* yaitu

1. Kode *use case*, judul dan *relationship* dengan *use case* yang lainnya.
 2. Penjelasan mengenai *precondition* yaitu kondisi spesifik yang harus terpenuhi sebelum sebuah UC bisa diinisiasi atau dieksekusi oleh aktor primer.
 3. Langkah langkah untuk menjalankan *use case* yaitu jalur interaksi yang mengarahkan pada skenario yang berhasil sehingga tujuan aktor bisa terpenuhi.
 4. Penjelasan mengenai *postcondition* yaitu kondisi spesifik yang harus terjadi ketika UC berhasil dijalankan atau dieksekusi secara lengkap, sebagai representasi dari tujuan yang ingin dicapai oleh aktor primer. Jumlah kondisi akhir bisa lebih dari 1 keadaan.
 5. *Abnormal path* yaitu jalur alternatif dari interaksi yang terjadi antar aktor dengan sistem yang mencakup pencabangan(pilihan) maupun skenario yang gagal sehingga tujuan aktor tidak terpenuhi.
- e. Membuat permintaan atau persyaratan tambahan untuk sistem.
 - f. Membuat *system use case priorities*.
- *Analysis*
 - a. Membuat sebuah analisis *class diagram*.
 - b. Membuat sebuah *attribute list* (dalam bentuk deskripsi).
 - c. Merealisasikan *use case* dengan menggunakan *communication diagram*.
 - d. Membuat sebuah *operation list* (berserta deskripsi).
 - *Subsystem Design*
 - a. Melakukan pemetaan *class* ke *business layer classes* : *class list* (dengan deskripsi) , *class diagram*, *field list* (dengan deskripsi).
 - b. Menampilkan realisasi layanan bisnis dan mencatat hasilnya dalam *sequence diagram*.
 - c. Membuat skema database.
 - d. Melakukan finalisasi dalam desain UI.

Penjelasan tersebut merupakan sebuah teori dasar dari metodologi *Ripple*. Tabel 2.5 akan menampilkan penyesuaian antara teori dasar metodologi dan penelitian ini, penyesuaian yang dimaksud adalah pemilihan fase-fase ataupun langkah-langkah yang akan digunakan dalam penelitian ini. Pada Tabel 2.5 akan menjelaskan fase yang terdapat dalam *Ripple*, langkah-langkah untuk mengerjakan fase tersebut serta memberikan penjelasan mengenai penyesuaian

metode dengan topik penelitian yang diharapkan dapat memperoleh hasil sesuai tujuan penelitian.



f. **Tabel Error! No text of specified style in document..5 Penyesuaian Metodologi Ripple dengan Penelitian**

Fase	Langkah	Penyesuaian	
Genesis	<i>Project Genesis (with pelanggan)</i>	Digunakan	
	<i>Assigning Responsibilities</i>	Tidak digunakan, dikarenakan fokus penelitian ini bukan mengenai manajemen proyek	
	Membuat <i>Workbook</i>	Tidak digunakan, dikarenakan penelitian ini bertujuan untuk membuat perancangan sistem saja	
	Membuat <i>Glossary</i>	Digunakan	
	Membuat <i>Project Plan</i>	Tidak digunakan, dikarenakan fokus penelitian ini bukan mengenai manajemen proyek	
	Membuat <i>Test Plan</i>	Tidak digunakan, dikarenakan pada penelitian ini perancangan yang digunakan hanyalah pengujian perancangan	
Requirements	Membuat <i>business actor list</i> (dalam bentuk deskripsi).	Digunakan	
	Membuat <i>business use case list</i> (dalam bentuk deskripsi).	Digunakan	
	Mengilustrasikan <i>business use cases</i> menggunakan <i>activity diagrams. (optional)</i>	Digunakan	
	Mengilustrasikan <i>business use cases</i> menggunakan <i>communication diagrams. (optional)</i>	Tidak digunakan, dikarenakan yang dibutuhkan dalam penelitian ini adalah urutan proses setiap <i>use case</i> dan sudah digambarkan dengan <i>activity diagram</i>	
	Membuat <i>business use case details</i>	Digunakan	
	System	Melakukan <i>brainstorm</i> dengan menggunakan sketsa UI	Digunakan
		Membuat <i>system actor list</i> (dalam bentuk deskripsi)	Digunakan

g.

h. Tabel 2.5 Penyesuaian Metodologi *Ripple* dengan Penelitian (Lanjutan)

Fase		Langkah	Penyesuaian
Requirements	System	Membuat <i>system use case list</i> (dalam bentuk deskripsi)	Digunakan
		Membuat <i>system use case diagram</i>	Digunakan
		Membuat <i>system use case survey</i>	Tidak digunakan, dikarenakan hubungan aktor dengan <i>use case</i> telah digambarkan pada <i>use case diagram</i>
		Membuat <i>system use case details</i>	Digunakan
		Membuat permintaan atau persyaratan tambahan untuk sistem	Digunakan
		Membuat <i>use case priorities</i>	Tidak digunakan, dikarenakan urutan implementasi <i>use case</i> tidak sesuai dengan tujuan penelitian
Analysis		Membuat analisis <i>class diagram</i>	Digunakan
		Membuat analisis <i>attributes list</i> (dalam bentuk deskripsi)	Digunakan
		Menggunakan <i>state machines</i> untuk memodelkan entitas kompleks dan dicatat hasilnya dalam <i>state machine diagrams (optional)</i>	Tidak digunakan, dikarenakan fungsi yang sakma yaitu urutan proses sistem sudah digambarkan menggunakan <i>use case diagram</i> pada fase <i>Requirements</i>
		Membuat realisasi <i>use case</i> menggunakan <i>communication diagrams (optional)</i>	Digunakan
		Membuat <i>operation list</i> (deskripsi)	Digunakan
Design	System	Membuat <i>technology choices</i>	Tidak digunakan, dikarenakan tujuan penelitian adalah membuat perancangan sistem yang bersifat umum

i. **Tabel 2.5 Penyesuaian Metodologi *Ripple* dengan Penelitian (Lanjutan)**

Fase		Langkah	Penyesuaian
Design	System	Membuat <i>deployment diagram</i> dan <i>layer diagram</i>	Tidak digunakan, dikarenakan tujuan enelitian lebih mengarah kepada perancangan sistem dari pada sub-sistem
	Subsystem	Membuat <i>class diagrams</i>	Digunakan
		Membuat <i>sequence diagrams</i>	Digunakan
		Membuat <i>skema database</i>	Digunakan
		Membuat penyelesaian desain UI	Digunakan
Class Spesification		Membuat <i>informal spesification</i> dalam <i>design</i>	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
		Membuat <i>source code</i> untuk setiap <i>class</i>	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
Implementation		Membuat <i>unit test</i>	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
Testing		Membuat <i>implementation code</i>	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
		Menguju sistem dan memperbaiki kesalahan yang mungkin muncul	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian

j. Tabel 2.5 Penyesuaian Metodologi *Ripple* dengan Penelitian (Lanjutan)

Fase	Langkah	Penyesuaian	Fase
<i>Deployment</i>		Membuat manual prosedur	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
		Meng- <i>install</i> kode pada sistem konsumen	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
		Melatih konsumen agar dapat memahami sistem	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
<i>Maintenance</i>		Memperbaiki kesalahan yang mungkin muncul selama proses instalasi	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian
		Melakukan peningkatan sistem yang didasarkan pada pelanggan's <i>feedback</i>	Tidak digunakan, dikarenakan termasuk dalam batasan masalah yang mana tidak sesuai dengan topik penelitian

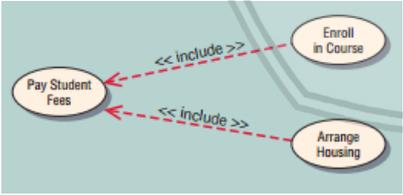
2.7 Diagram UML

Pada perkembangan teknologi software (perangkat lunak), diperlukan adanya standarisasi bahasa atau pemodelan terhadap software yang akan dibuat agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Pada perkembangan teknik pemrograman berorientasi obyek, muncul standarisasi bahasa pemodelan untuk pembangunan software yang dibangun dengan teknik pemrograman berorientasi obyek yaitu UML (Unified Modelling Language). UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks pendukung (Sukamto & Shalahuddin, 2014).

2.7.1 Use Case Diagram

Menurut Whitten dan Bentley (2007), *use case diagram* dipakai untuk menggambarkan relasi antara sistem, sistem eksternal dan user dengan kasus yang disesuaikan dengan langkah-langkah yang telah ditentukan. *Use case diagram*. Pada Tabel 2.6 akan menjelaskan simbol yang digunakan pada *use case diagram* dan juga deskripsi penggunaan simbol tersebut.

k. **Tabel Error! No text of specified style in document..6 Simbol Use Case Diagram**

Simbol	Deskripsi
	<i>Use case</i> mendeskripsikan fungsi dari sebuah sistem dilihat dari sudut pandang pengguna.
	<i>Actors</i> merupakan sesuatu yang berinteraksi dengan sistem untuk saling bertukar informasi. <i>Actors</i> dapat berupa manusia, suatu organisasi atau sistem informasi.
	<i>Include relationship</i> atau disebut juga <i>uses relationship</i> menggambarkan situasi dalam suatu <i>use case</i> yang termasuk dalam <i>use case</i> lainnya. Bertujuan untuk mengurangi redundansi di antara dua <i>use case</i> atau lebih dengan menggabungkan langkah-langkah yang sama tersebut. Sebuah panah putus-putus yang mengarah pada suatu <i>use case</i> menandakan memiliki hubungan dengan <i>use case</i> lainnya.
	<i>Extends relationship</i> bertujuan untuk menggambarkan situasi dimana sebuah <i>use case</i> memiliki <i>behavior</i> yang memungkinkan <i>use case</i> lain dapat menggunakannya sebagai dasar variasi ataupun pengecualian.



Tabel 2.6 Simbol Use Case Diagram (Lanjutan)

Simbol	Deskripsi
	<p><i>Generalizes relationship</i> atau <i>inheritance</i> menunjukkan bahwa satu hal lebih umum daripada hal lain. Hubungan ini memungkinkan terjadi antara dua aktor maupun dua <i>use case</i>.</p>
	<p><i>Associations</i> adalah sebuah relasi antara seorang <i>actor</i> dengan sebuah <i>use case</i> di mana terjadi interaksi antara mereka. Asosiasi dengan panah tertutup (1) di ujung yang menyentuh <i>use case</i> mengindikasikan bahwa <i>actor</i> di ujung yang satu lagi melakukan <i>use case</i> tersebut. Sedangkan asosiasi tanpa panah (2) mengindikasikan sebuah interaksi dari <i>use case</i> ke <i>actor</i> yang menerima hasil dari <i>use case</i> tersebut.</p>

Sumber: Whitten dan Bentley (2007)

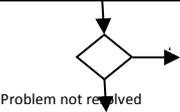
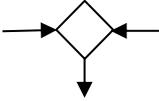
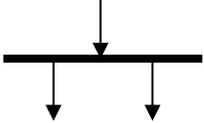
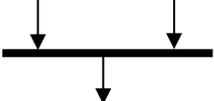
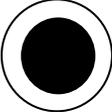
2.7.2 Activiy Diagram

Menurut Whitten dan Bentley (2007), activity diagram merupakan gambaran dari alur yang berurutan dari aktivitas *use case* atau proses bisnis. *Activity diagram* digunakan untuk melihat bagaimana sebuah aksi dan reaksi saat sebuah *use case* dilakukan. Dari diagram ini dapat terlihat bagaimana awal suatu proses berjalan dan akhirnya. Pada Tabel 2.8 akan dijelaskan simbol-simbol yang digunakan dalam membuat *activity diagram* dan juga deskripsi dari simbol tersebut.

I. Tabel Error! No text of specified style in document..7 Notasi pada Activity Diagram

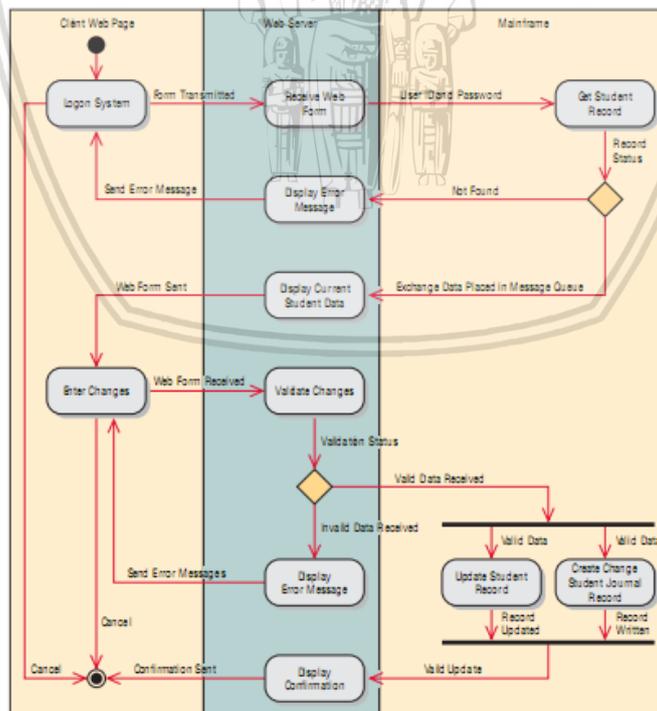
Simbol	Deskripsi
	<p><i>Initial node</i> menggambarkan titik mulai suatu proses.</p>
	<p><i>Actions</i> menggambarkan aktivitas yang terjadi.</p>
	<p><i>Flow</i> (alur) merupakan panah dalam diagram yang mengindikasikan alur antar-<i>actions</i>.</p>



Simbol	Deskripsi
	<i>Decision</i> digunakan apabila pada suatu proses terjadi seleksi kondisi.
	<i>Merge</i> digunakan untuk menggabungkan <i>flow</i> yang sebelumnya berpisah karena suatu seleksi kondisi atau <i>decision</i> .
	<i>Fork</i> digunakan untuk menggambarkan suatu proses yang dapat dilakukan bersamaan
	<i>Join</i> digunakan untuk menggabungkan suatu aktivitas yang sebelumnya dipisahkan oleh <i>fork</i> .
	<i>Activity final</i> digunakan untuk menggambarkan bahwa suatu proses telah berakhir

1. Sumber: Whitten dan Bentley (2007)

2. Sedangkan menurut Sugiarti (2013), *activity diagram* juga dapat digunakan untuk memodelkan *action* yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari *action* tersebut. Contoh *activity diagram* terdapat pada Gambar 2.3



m. **Gambar Error! No text of specified style in document..2 Contoh Activity Diagram**

Sumber: Kendall dan Kendall (2011)

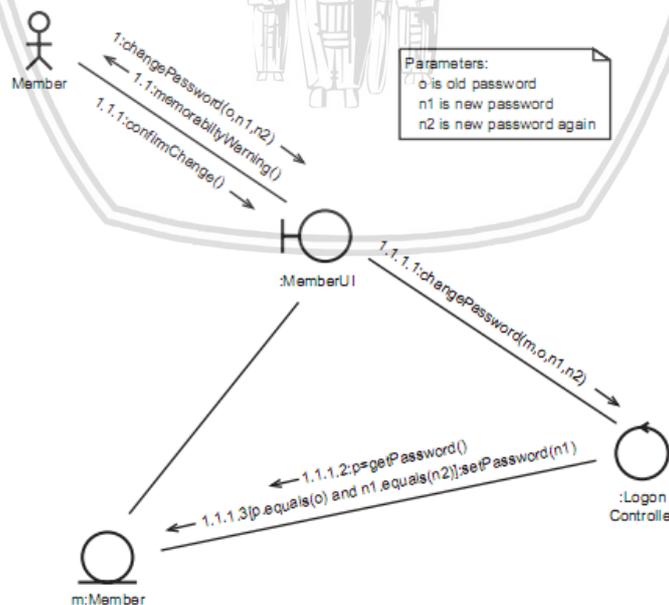
2.7.3 Communication Diagram

Menurut Whitten dan Bentley (2007), *communication diagram* menggambarkan interaksi dari objek-objek dengan pesan. *Communication diagram* memiliki kemiripan, terutama kesamaan simbol dengan *sequence diagram*, akan tetapi *sequence diagram* lebih fokus pada waktu dan urutan pesan, sedangkan *communication diagram* lebih berfokus pada organisasi struktural dari objek-objek.

Pada pembuatan *communication diagram* terdapat beberapa simbol yang digunakan, yang akan dijelaskan pada tabel berikut.

n. **Tabel Error! No text of specified style in document..8 Notasi pada Communication Diagram**

Simbol	Deskripsi
	<i>Entity</i> , entitas yang mempunyai atribut yang memiliki data yang bisa direkam.
	<i>Boundary</i> , menghubungkan user dengan sistem.
	<i>Control</i> , untuk mengontrol aktivitas-aktivitas yang dilakukan oleh sebuah kegiatan.
	<i>Message</i> , pengiriman pesan.



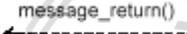
o. **Gambar Error! No text of specified style in document..3 Contoh Communication Diagram Mengganti Password**

Sumber : O'Dochtery (2005)

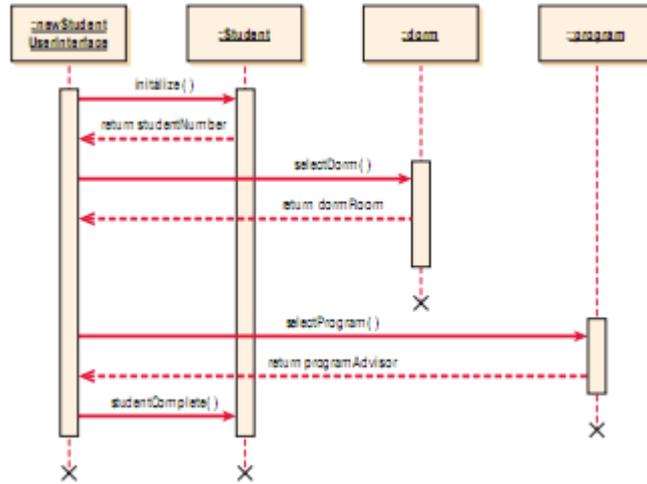
2.7.4 Sequence Diagram

Sequence diagram secara grafis menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada sekuensi sebuah use case atau operasi. Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima oleh objek dalam sekuensi atau timing (Sugiarti, 2013). Pesan dapat berupa sinyal atau panggilan terhadap suatu operasi. Notasi pesan untuk panggilan terhadap suatu operasi dapat dituliskan dalam sintaks UML atau sintaks bahasa pemrograman tertentu. Pada Tabel 2.9 akan dijelaskan simbol yang digunakan dalam membuat *sequence diagram* dan juga deskripsi penggunaan simbol tersebut.

p. **Tabel Error! No text of specified style in document..9 Notasi pada Sequence Diagram**

Simbol	Deskripsi
	<i>Class</i> , ditampilkan dengan bentuk persegi panjang pada umumnya dituliskan nama <i>class</i> tersebut ditengahnya.
	<i>Message</i> , pengiriman pesan.
	<i>Return Values</i> , ditampilkan dengan garis panah terputus yang menggambarkan hasil dari pengiriman pesan. Digambarkan arah dari kanan ke kiri.
	Garis kehidupan (<i>Lifelines</i>), Garis vertical putus-putus yang memanjang kebawah dari simbol <i>actor</i> dan sistem yang mengindikasikan urutan kehidupan.
	<i>Bar</i> aktivasi, <i>Bar</i> didalam garis kehidupan (<i>lifetime</i>) yang menunjukkan periode waktu ketika peserta aktif dalam interaksi.

3. Menurut Pressman (2010), *sequence diagram* digunakan untuk menunjukkan komunikasi dinamis antara objek selama pelaksanaan tugas. Contoh *sequence diagram* terdapat pada Gambar berikut



Gambar Error! No text of specified style in document..4 Contoh Sequence Diagram Administrasi Mahasiswa

Sumber : Kendall dan Kendall (2011)

2.7.5 Class Diagram

Class diagram memodelkan struktur sistem dari segi pendefinisian kelas-kelas sesuai dengan sistem yang akan dibuat. Setiap kelas memiliki atribut dan metode operasi (Sukamto & Shalahuddin, 2014). Atribut merupakan suatu variabel yang dimiliki kelas. Sedangkan operasi merupakan fungsi fungsi yang dimiliki suatu kelas. *Class diagram* digunakan untuk menggambarkan relasi antar kelas yang ada. Susunan struktur diagram kelas yang baik harus memiliki jenis jenis kelas seperti, Kelas *View* adalah kelas digunakan untuk menampilkan sistem, Kelas *Controller* adalah kelas yang menangani jalannya alur suatu sistem dan Kelas *Model* adalah kelas yang menyimpan data dari sebuah sistem. Definisi symbol yang digunakan dalam *class diagram* dan kegunaannya dijelaskan pada Tabel 2.7

q. Tabel Error! No text of specified style in document..10 Notasi pada Class Diagram

Simbol	Deskripsi
	<p><i>Attribute</i> adalah sekumpulan data yang dimiliki oleh objek. <i>Behavior</i> adalah kumpulan dari sesuatu yang dapat dilakukan oleh objek dan terkait dengan fungsi-fungsi yang bertindak pada data objek (atribut). Pada siklus berorientasi objek, perilaku objek merujuk kepada metode, operasi, atau fungsi.</p>
	<p><i>Inheritance</i>, menunjukkan bahwa satu kelas merupakan turunan dari kelas lain.</p>
	<p><i>Association</i>, menunjukkan bahwa objek dari satu kelas berhubungan dengan kelas lain.</p>
	<p><i>Agregation</i>, menunjukkan bahwa contoh objek dari satu kelas terdiri dari contoh objek dari kelas lain.</p>
	<p><i>Composition</i>, menunjukkan hubungan dimana satu kelas bertanggung jawab atas pembuatan dan perusakan bagian-bagian dalam kelas lainnya. Jika satu kelas rusak, maka kelas lain juga rusak.</p>

4. Sumber: Whitten dan Bentley (2007)

2.8 Consistency Analysis: Requirement Configuration Structure

Requirement consistency analysis merupakan metode untuk melakukan analisis konsistensi pada hasil perancangan sistem dengan pemanfaatan hubungan antar elemen perancangan (Nistala & Kumari, 2013). Dalam penerapannya terdapat 4 langkah kerja yaitu:

1. *Layers and Configuration Items*

Tahap ini mendeskripsikan asal dari 4 layer yang akan dianalisis. Layer tersebut antara lain :

- a) *Business layer* yang berisi tujuan organisasi yang diperoleh dari proses yang berjalan pada sebuah organisasi.
- b) *Process layer* yang berisi proses dan sub-proses yang harus ada untuk mencapai tujuan organisasi.

- c) *Requirements layer* yang berisi kunci dari kebutuhan sistem berdasarkan proses dan sub-proses.
- d) *Specification layer* yang menghasilkan analisis kebutuhan dalam bentuk spesifikasi kebutuhan.

2. Configuration Structure

Tahap ini memberikan panduan dalam identifikasi *layer* dan menghubungkan 4 layer pada komponen yang pertama. Setiap elemen pada tiap layer akan dijelaskan pada tahap ini.

3. Consistency Analysis

Tahap ini berguna untuk memberikan validasi dari tahap kedua, dengan cara menggambarkan hubungan antara empat layer yang telah didefinisikan dengan digambarkan dalam bentuk diagram *consistency analysis*.

4. Requirement Consistency Index

Requirement Consistency Index berfungsi untuk melakukan perhitungan terhadap persentasi konsistensi dalam pendefinisian kebutuhan. Proses perhitungan RCI dituliskan pada persamaan :

$$RCI = A/(B+C)$$

Keterangan :

A : Jumlah elemen kebutuhan yang konsisten.

B : Jumlah total elemen kebutuhan.

C : Jumlah elemen kebutuhan yang terdefinisi secara tidak benar.

2.9 Correctness

Fokus pengujian perangkat lunak adalah dengan melihat sistem kandidat pada kebutuhan yang dipilih dan memeriksa apakah program kandidat tersebut memiliki fungsi yang sesuai dengan spesifikasi kebutuhannya (Mili & Tchier, 2014). Studi tentang *correctness* program mengarah pada tingkat granularitas yang tidak sesuai. Khususnya mengarah pada asumsi tentang perilaku sistem pada tahap tertentu dalam pelaksanaannya.

Setiap kebutuhan dan fungsi yang telah didapatkan akan didefinisikan setelah itu dilakukan pemetaan antara himpunan kebutuhan dengan himpunan fungsi dan sebaliknya. Hasil pemetaan tersebut akan dicocokkan dengan kategori pada evaluasi *correctness*. Berikut ini kategori dan penilaian pada evaluasi *correctness* (diumpamakan R adalah mewakili kebutuhan dan mewakili fungsi):

- *Correctness*

Berkategori *correctness* apabila hasil pemetaan antara himpunan kebutuhan dengan himpunan fungsi sama dengan hasil pemetaan himpunan kebutuhan beririsan dengan himpunan fungsi.

$$(\text{dom}(R \cap P)) = \text{nilai}(\text{dom}(R))$$

- *Partially Correctness*

Berkategori *partially correctness* apabila hasil antara himpunan kebutuhan dengan himpunan fungsi sama dengan hasil pemetaan irisan antara himpunan kebutuhan dengan himpunan fungsi.

$$(\text{dom}(R \cap P)) = \text{nilai}(\text{dom}(R)) \cap \text{nilai}(\text{dom}(P))$$

- *Terminate Correctness*

Berkategori *terminate correctness* apabila himpunan kebutuhan merupakan *subset* dari himpunan fungsi.

Berikut ini perumpamakan R berisi daftar spesifikasi kebutuhan yang diperlukan oleh pengguna.

$$R = \{(0,0), (0,1), (0,2), (1,1), (1,2), (1,3), (2,2), (2,3), (2,4), (3,3), (3,4), (3,5)\}$$

Terdapat tiga program yang memiliki fungsi yang disesuaikan dengan spesifikasi dan diumpamakan kedalam himpunan P.

$$P1 = \{(0,1), (1,2), (2,3), (3,4)\}$$

$$P2 = \{(0,0), (1,2), (2,4), (4,8), (5,10), (6,12)\}$$

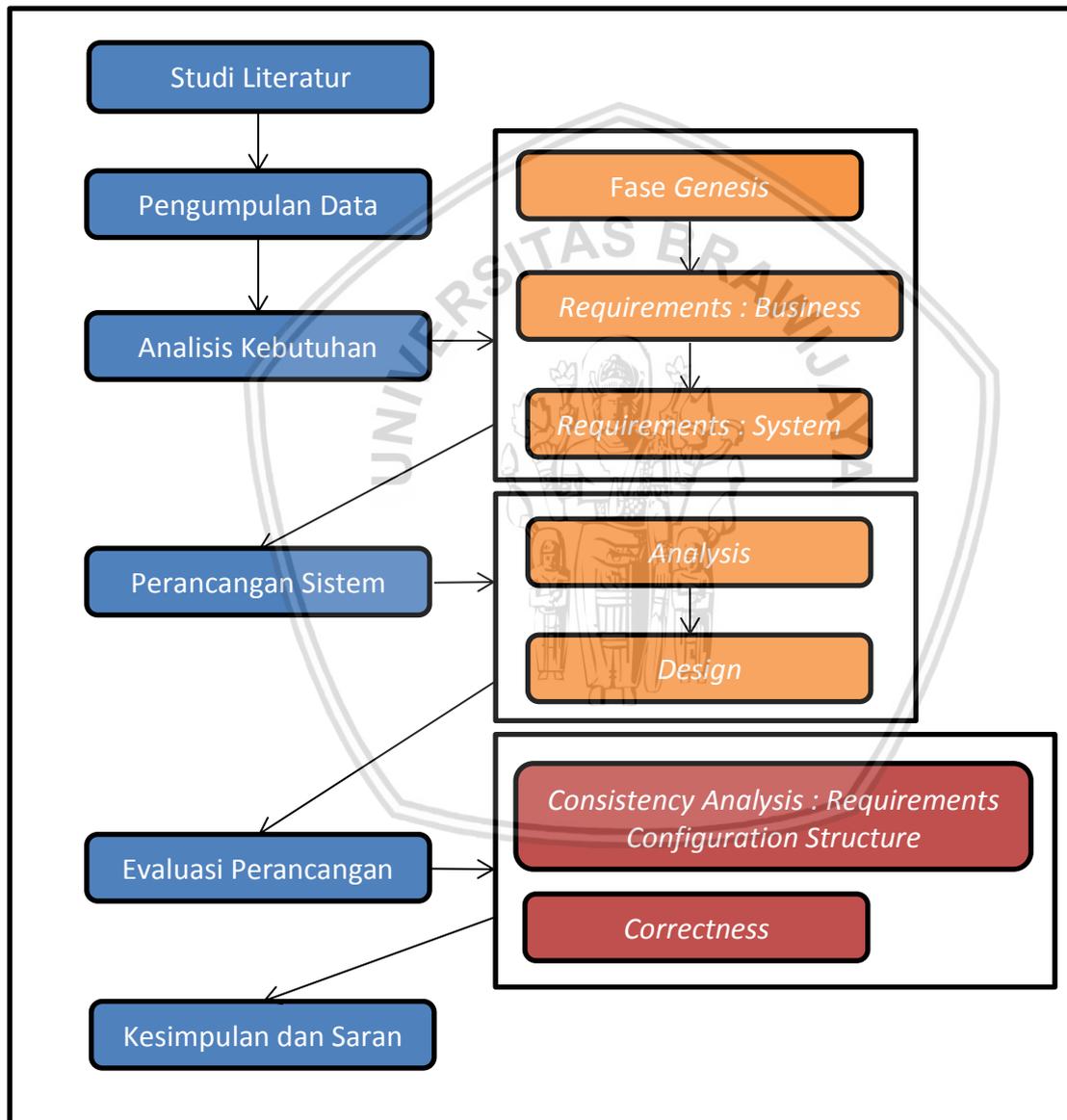
$$P3 = \{(0,0), (1,2), (2,4), (3,6), (4,8), (5,10), (6,12)\}$$

Ketiga kandidat program di atas akan dipetakan dengan kebutuhan yang telah didefinisikan sebelumnya.

- P1 dikatakan sesuai dengan spesifikasi kebutuhan atau termasuk dalam kategori *correctness* karena setiap fungsi yang dimiliki p1 terdapat pada spesifikasi kebutuhan. Dan apabila dipetakan pada setiap kebutuhan maka
- P2 dikatakan tidak sesuai dengan spesifikasi kebutuhan atau termasuk dalam kategori *partially correctness* karena sebagian fungsi yang dimiliki p2 terdapat pada spesifikasi kebutuhan dan sebagian lagi tidak dimiliki.
- P3 dikatakan tidak sesuai dengan spesifikasi kebutuhan atau termasuk dalam kategori *terminate normally* karena fungsi yang dimiliki p3 tidak sesuai dengan spesifikasi kebutuhan.

BAB 3 METODOLOGI PENELITIAN

Metodologi penelitian merupakan mekanisme penyelesaian masalah penelitian dan menjelaskan mengenai metode yang digunakan selama penelitian skripsi. Gambar 3.1 menjelaskan tahap – tahap yang dilakukan dalam penelitian analisis dan perancangan sistem informasi pemesanan barang. Tipe penelitian yang digunakan dalam analisis dan perancangan sistem informasi pengelolaan proses pemesanan barang PT. Trivia Nusantara adalah tipe penelitian implementatif-perancangan. Diagram alir tahap penelitian yang akan dilakukan dalam skripsi ini terdapat pada Gambar 3.1.



Gambar 0.1 Tahapan Penelitian

Penelitian ini dimulai dari tahap studi literatur penulis mencari sumber literatur untuk landasan teori untuk pengerjaan penelitian, selanjutnya tahap observasi dan pengumpulan data yang dilakukan kepada *stakeholder* dari obyek sistem akan dikembangkan, tahap ketiga melakukan analisis dan perancangan sistem dengan metode *Ripple* pada fase awal yaitu fase *genesis* sampai dengan fase *design*. Tahap keempat adalah

fase evaluasi perancangan yang digunakan untuk mengukur konsistensi dan kesesuaian hasil perancangan dengan kebutuhan sistem informasi pemesanan barang. Tahap terakhir yaitu pengambilan keputusan yang berisi mengenai kesimpulan dari penelitian dan saran apa yang perlu dilakukan untuk mendukung penelitian ini kedepannya. Pada Tabel 3.1 akan dijelaskan serangkaian fase dan langkah yang akan digunakan untuk melakukan penelitian yang telah disesuaikan dengan metodologi yang digunakan.

Tabel 0.1 Penggunaan Metode *Ripple* pada Penelitian

Fase		Langkah	Output
Genesis		<i>Project Genesis (with pelanggan)</i>	Deskripsi
		Membuat <i>Glossary</i>	Deskripsi
Requirements	Business	Membuat <i>business actor list</i> (deskripsi)	Deskripsi
		Membuat <i>business use case list</i> (deskripsi)	Deskripsi
		Mengilustrasikan <i>business use cases</i> menggunakan <i>activity diagram</i> (optional)	<i>Activity Diagram</i> untuk proses bisnis
		Membuat <i>business use case detail</i>	Deskripsi
	Systems	<i>Brainstorm system interaction</i> dengan menggunakan sketsa UI	Sketsa awal UI
		Membuat <i>system actor list</i>	Deskripsi
		Membuat <i>system use case list</i>	Deskripsi
		Membuat <i>system use case diagram</i>	<i>Use case diagram</i>
		Membuat <i>system use case details</i>	Deskripsi
		Membuat <i>supplementary requirements</i> untuk sistem	Deskripsi
		Membuat <i>analysis class diagram</i>	<i>Class diagram</i> secara umum
Analysis		Membuat <i>analysis class diagram</i>	<i>Class diagram</i> secara umum

Tabel 3.1 Penggunaan Metode *Ripple* pada Penelitian (Lanjutan)

Fase	Langkah	Output
Analysis	Membuat <i>analysis atributes list</i>	Deskripsi
	Membuat realisasi use case menggunakan <i>communication</i>	<i>Communication diagram</i>

		<i>diagram (optional)</i>	
		Membuat deskripsi <i>operation list</i>	Deskripsi
<i>Design</i>	<i>Subsystem</i>	Membuat <i>class diagram</i>	<i>Class diagram</i>
		Membuat <i>sequence diagram</i>	<i>Sequence diagram</i>
		Membuat <i>database schema</i>	<i>Database schema</i>

3.1 Studi Literatur

Pada tahap ini, literatur dan referensi dari berbagai macam sumber seperti jurnal, buku, *e-book* atau media *online* dikumpulkan untuk mendapatkan teori-teori dasar dan pendukung penelitian ini.

- Penelitian sebelumnya yang pernah dilakukan
- Pemograman Berorientasi Objek
- Metodologi *Ripple*
- Kebutuhan Fungsional dan Non-Fungsional
- *Unified Modelling Diagram* (UML)
- *Consistency Analysis*
- *Correctness*

3.2 Observasi dan Pengumpulan Data

Pada tahap ini, penulis melakukan observasi dan pengumpulan data langsung di tempat penulis melakukan penelitian dan pada *Stakeholder* terkait. Dalam penelitian ini metode penggalan dan analisis kebutuhan yang digunakan adalah wawancara dan observasi.

3.2.1 Wawancara

Pengumpulan data secara wawancara yang dilakukan secara lisan untuk mendapatkan informasi dari pegawai PT. Trivia Nusantara. Data yang dibutuhkan berupa proses bisnis dan kendala yang sering terjadi secara detail dalam jalannya proses pemesanan barang. Teknik wawancara yang digunakan dalam penelitian ini menggunakan teknik wawancara semiterstruktur. Teknik wawancara semiterstruktur bertujuan untuk menentukan permasalahan secara lebih terbuka, sehingga pihak yang diwawancara dapat lebih mengeluarkan ide dan pendapatnya.

Pedoman yang digunakan dalam melakukan wawancara meliputi daftar pertanyaan dan *domain objective* didapatkan dari "Jogiyanto, H.M., 1990, Analisis dan Desain Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis (Yogyakarta : Andi Offset, hal 145-149", yang merefrensi " Joseph W. Wilkinson, Accounting and Information Systems, (New York : John Wiley and Sons, 1982), hal 585".

3.2.2 Observasi

Dalam observasi ini peneliti menggunakan metode observasi partisipasi pasif, yaitu peneliti terlibat dengan kegiatan sehari-hari orang yang sedang diamati atau yang digunakan

sebagai sumber data penelitian. Dengan observasi partisipasi pasif ini, data yang diperoleh akan lebih lengkap, tajam, dan sampai mengetahui pada tingkat makna dari setiap perilaku yang tampak. Dalam hal ini peneliti hanya mengamati saja dalam beberapa waktu tanpa terjun langsung ke dalam kegiatan.

3.3 Analisis Kebutuhan Sistem

Setelah melakukan observasi dan pengumpulan data yang sudah dilakukan pada tahap sebelumnya maka tahap selanjutnya adalah melakukan analisis kebutuhan yang mana digunakan untuk menemukan kebutuhan fungsional dan non-fungsional. Kebutuhan pengguna secara umum dibagi menjadi dua kategori utama, yaitu Kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional menggambarkan apa yang sistem lakukan, sedangkan kebutuhan non-fungsional menggambarkan batasan-batasan yang harus dipenuhi oleh sistem dalam hal batasan kualitas dan batasan penggunaan (*usability*).

Selain itu analisis kebutuhan yang dilakukan disini menggunakan beberapa tahap dari metodologi *Ripple* yaitu pada tahap *project genesis* dan pada fase *requirements*, pada fase ini akan terlihat apa saja kebutuhan dari segi bisnis organisasi. Pada tahap *project genesis* aktivitas yang dilakukan adalah mendefinisikan apa saja kebutuhan pelanggan, mendapatkan dokumentasi kebutuhan pelanggan dan membuat *glossary*. *Glossary* yang dimaksud adalah membuat sebuah glossarium mengenai singkatan apa saja yang digunakan dalam proses analisis kebutuhan sehingga apabila terjadi sesuatu dapat dimudahkan dengan adanya glosarium.

Setelah itu pada fase *requirements* kegiatan yang dilakukan yaitu membuat *business actor list*, membuat *business use case list*, mengilustrasikan *business use cases* menggunakan *activity diagram*, membuat *business use case detail*, membuat *system actor list*, membuat *system use case list*, membuat *system use case diagram*, membuat *system use case details* dan membuat *supplementary requirement*. Pada tahap ini akan dilakukan analisis kebutuhan terhadap proses bisnis yang ada pada organisasi. Dari *business actor list* akan terlihat aktor yang berhubungan langsung dengan proses bisnis yang ada. Setelah itu pada *business use case list* akan terlihat kegiatan apa saja yang dilakukan selama proses bisnis yang ada. Pada aktivitas mengilustrasikan *business use cases* menggunakan *activity diagram*, akan terlihat urutan urutan proses yang terjadi pada organisasi selama proses bisnis berlangsung.

Sedangkan pada fase *requirement* sub-fase *system* tahap pertama akan diidentifikasi aktor dalam sistem. Setelah itu memetakan antara kebutuhan pengguna dan fungsi sistem dalam *system use case list*. Lalu menggambarkan *use case diagram* dan memberikan detail interaksi sistem dengan aktor pada *system use case details*. Pada tahap akhir akan menjelaskan kebutuhan tambahan dalam sistem yang akan dibangun.

3.4 Perancangan Sistem

Tahapan selanjutnya setelah melakukan analisis kebutuhan adalah melakukan perancangan sistem. Pada tahap ini penulis melanjutkan tahapan yang ada pada metodologi *Ripple* yaitu masih pada tahap *requirements*, *analysis* dan *design*. Pada fase *analysis* aktivitas yang dilakukan adalah membuat analisis *class diagram* yang akan dibuat, membuat analisis pada *attribute list*, membuat *communication diagram* dan membuat deskripsi

operation list. Pada fase *design* kegiatan yang dilakukan adalah membuat *class diagram*, *sequence diagram*, *database schema* dan membuat penyelesaian desain UI.

3.5 Evaluasi Perancangan

Pada tahap ini, penulis melakukan evaluasi kesesuaian untuk mengukur presentase konsistensi kebutuhan sistem terhadap proses bisnis, proses sistem, kebutuhan, dan spesifikasi sistem pemesanan barang, dengan menggunakan metode *consistency analysis*. Metode *correctness* juga digunakan untuk menilai kesesuaian antara kebutuhan pengguna dengan fungsi pada sistem yang dirancang.

3.6 Pengambilan Kesimpulan

Dari hasil pengujian dapat ditarik kesimpulan mengenai hasil validasi kebutuhan dan perancangan pada penelitian. Kesimpulan tersebut mengacu pada rumusan masalah yang telah dituliskan sebelumnya pada Bab 1 dan berdasarkan hasil analisis kebutuhan dan perancangan pada Bab IV dan Bab V.



BAB 4 ANALISIS DAN PERANCANGAN

4.1 User Story

PT. Trivia Nusantara merupakan sebuah perusahaan *retail* yang melayani penyediaan alat tulis perkantoran pada beberapa perusahaan di Indonesia. Direktur utama PT. Trivia Nusantara bermaksud untuk membangun sebuah sistem informasi. Untuk memudahkan para pelanggan dalam melakukan pemesanan barang dan jumlah yang harus dibayarkan ketika membeli suatu barang, maka perusahaan perlu menyediakan sebuah layanan berbasis web (*web service*). *Web service* yang disediakan dapat digunakan untuk menampilkan produk dan mendapatkan jumlah total pembelian yang harus dibayar.

Dalam sistem yang ingin dibangun, perusahaan menginginkan adanya akun pada setiap pengguna. Pelanggan dapat membuat akun dengan mengisi *form* kelengkapan seperti nama perusahaan, kontak perusahaan dan email bagian keuangan. Pembuatan akun oleh pelanggan hanya bisa dilakukan apabila pelanggan telah menjalin kerjasama dengan PT. Trivia Nusantara.

Sebelum melakukan pemesanan barang, sistem informasi pemesanan barang ini memerlukan seorang admin yang bertugas untuk mengelola data-data yang dibutuhkan terkait produk yang dijual. Hal ini sesuai dengan keinginan manajemen perusahaan yaitu untuk membangun sebuah aplikasi berbasis web yang dapat memudahkan pelanggannya dalam mengetahui barang yang akan dipesan secara lengkap meliputi nama produk, harga produk, *merk* produk, deskripsi dan juga gambar produk. Dengan adanya deskripsi produk dapat mempercepat proses pemesanan karena tidak perlu lagi melakukan kepada pihak perusahaan mengenai penjelasan barang yang dijual.

Setelah melakukan *login*, pelanggan melakukan pemesanan mengenai barang yang diinginkan, terdapat sebuah halaman pemesanan yang dapat menampilkan gambar produk, dan informasi singkat produk. Apabila terdapat kesulitan dalam menemukan barang yang akan dipesan mengingat produk alat tulis sangat banyak maka kolom *search* perlu disediakan. Setelah menemukan barang yang diinginkan. Setelah melakukan proses pembelian barang, pelanggan melakukan proses *check out* untuk memastikan barang mana saja yang akan dibeli. Disini pelanggan diminta untuk mengisi *form* yang dibuat yang berisi kelengkapan alamat dan *contact person* bagian keuangan pelanggan. Setelah mendapatkan alamat dan kelengkapan perusahaan, akan dihasilkan kelengkapan pemesanan yang menampilkan jumlah total pemesanan termasuk dengan biaya pengiriman.

Dikarenakan pelanggan PT. Trivia rata-rata sebuah perusahaan besar maka pembayaran perlu melakukan konfirmasi terhadap bagian keuangan pelanggan perusahaan yang bersangkutan sesuai yang telah diisikan pada formulir sebelumnya. Oleh karena itu sistem yang disediakan diharapkan dapat mengirimkan sebuah notifikasi kepada bagian keuangan pelanggan via *email*, yang mana berisi kelengkapan pemesanan beserta biaya pengiriman. Setelah itu

bagian keuangan pelanggan pelanggan dapat melakukan konfirmasi melalui *email*.

Setelah bagian keuangan pelanggan pelanggan melakukan konfirmasi pemesanan, secara otomatis pesanan berubah menjadi sebuah *purchase order* (PO) dan bagian gudang dapat mengakses PO tersebut agar dapat dilakukan pengecekan barang yang *ready stock* dan dalam kondisi yang baik. Apabila ada kerusakan barang atau kurangnya stok dalam gudang maka bagian gudang akan melakukan *restock*. Apabila barang yang dipesan sudah siap selanjutnya adalah melakukan *packing* barang, bagian gudang akan melakukan *update* terhadap status pemesanan barang. Status pemesanan barang hanya dapat dilakukan oleh orang yang memiliki otoritas dalam sistem seperti admin dan petugas. Terdapat beberapa status pemesanan barang yaitu, "Menunggu Konfirmasi", "Packing barang", "Sedang Dikirim", "Barang Diterima", "Belum Lunas" dan "Lunas". Setelah itu bagian gudang dapat mencetak PO barang yang berisi informasi detail tentang PO barang tersebut. Surat PO yang dihasilkan, dicetak rangkap dua satu sebagai arsip PT. Trivia Nusantara dan juga dibawa saat barang dikirim.

Setelah barang dikemas, bagian gudang akan mencetak dokumen tagihan yang akan ikut dikirimkan ke alamat pelanggan. Kurir akan mengantarkan barang sesuai dengan alamat yang tertera pada dokumen PO. Setelah barang tiba di alamat pelanggan akan dilakukan *checklist* penerimaan barang yang dipesan sesuai dengan *list* barang pada dokumen PO. Setelah perusahaan pelanggan menerima barang pesanan, kurir akan memberikan dokumen tagihan yang ditujukan kepada bagian keuangan pelanggan. Pembayaran akan dilakukan oleh pihak pelanggan pada umumnya menjadi piutang sampai jatuh tempo yang telah disepakati. Setelah proses pemesanan barang selesai, setiap bulan petugas dapat mendapatkan rekap PO sebagai salah satu evaluasi terhadap pelayanan perusahaan

4.2 Fase *Genesis*

Fase ini adalah fase awal dari metodologi *Ripple*. Pada fase ini akan menjelaskan apa saja yang dibutuhkan oleh PT. Trivia Nusantara dalam memudahkan pelanggan melakukan proses pemesanan barang.

4.2.1 *Projek Genesis*

Projek genesis adalah fase untuk mendapatkan konsep apa yang dibutuhkan oleh PT. Trivia Nusantara mengenai sistem yang ingin dibangun. Berdasarkan penjelasan *user story* di atas dan hasil wawancara dengan pegawai PT. Trivia Nusantara terdapat beberapa kegiatan yang dapat didukung pelaksanaannya menggunakan sebuah sistem. Berikut adalah kebutuhan pengguna pada sistem informasi pemesanan barang yang akan dibangun:

1. Kegiatan penggunaan *user* untuk membedakan hak aksesnya.
2. Kegiatan mengelola informasi produk.
3. Kegiatan mendapatkan informasi produk.

4. Kegiatan melakukan pemesanan barang.
5. Kegiatan melakukan konfirmasi .
6. Kegiatan mengelola pesanan.
7. Kegiatan mengelola tagihan

4.2.2 Membuat *Glossary*

Dalam setiap *glossary* mendefinisikan istilah atau sebuah definisi yang sesuai. *Glossary* dibuat untuk memudahkan apabila di kemudian hari sistem akan diperiksa, pemeriksa dapat mengetahui istilah atau sinonim yang digunakan. Berikut adalah daftar istilah yang berhubungan dengan analisis dan perancangan sistem informasi pemesanan barang di PT. Trivia Nusantara. Pada Tabel 4.1 akan menjelaskan nama istilah yang digunakan pada sistem sedangkan deskripsi menggambarkan secara umum dan juga menjelaskan peran istilah tersebut dalam perancangan sistem yang akan dibangun.

Tabel 0.1 *Glossary*

No.	Istilah	Deskripsi
1.	Pelanggan (<i>Business Actor</i> dan <i>System Actor</i>)	Seseorang yang melakukan pemesanan kepada PT. Triva Nusantara.
2.	Petugas (<i>Business Actor</i>)	Seseorang yang bertugas untuk mengelola penjualan dan berhubungan langsung dengan pelanggan.
3.	Bagian Gudang (<i>Business Actor</i>)	Seseorang yang melakukan <i>packing</i> barang yang telah dipesan oleh pelanggan.
4.	Kurir (<i>Business Actor</i>)	Seseorang yang melakukan pengiriman barang ke alamat tujuan pemesanan.
5.	Bagian Keuangan Pelanggan (<i>Business Actor</i> dan <i>System Actor</i>)	Bagian keuangan pelanggan merupakan sebuah departemen yang mengecek apakah pelanggan telah melakukan pemesanan sesuai dengan kebutuhan perusahaan.
6.	Pengguna (<i>System Actor</i>)	Seseorang yang belum dikenali perannya dalam sistem.
7.	Pegawai Perusahaan (<i>System Actor</i>)	Pegawai perusahaan disini merupakan bagian dari PT. Trivia Nusantara. Pegawai perusahaan adalah seseorang yang berperan dalam memproses pemesanan pelanggan.
8.	Admin (<i>System Actor</i>)	Seseorang yang mengelola sistem supaya dapat berjalan dengan maksimal.
9.	<i>Purchase Order</i>	Merupakan sebuah satu set pemesanan

No.	Istilah	Deskripsi
	(<i>Business Objec</i> dan <i>System Object</i>)	yang telah dikonfirmasi oleh bagian keuangan pelanggan.
10.	Dokumen <i>Purchase Order</i> (<i>Deployment Artifact</i>)	Merupakan sebuah dokumen berisi informasi mengenai barang yang dipesan.
11.	Tagihan (<i>Business Object, System Object, Analysis Object</i> dan <i>Design Object</i>)	Merupakan sebuah objek untuk melakukan penagihan pembayaran.
12.	Surat Tagihan (<i>Deployment Artifact</i>)	Merupakan sebuah dokumen yang ditujukan untuk perusahaan pelanggan sebagai alat untuk melakukan penagihan pembayaran penagihan.
13.	Rekap <i>Purchase Order</i> (<i>Deployment Artifact</i>)	Merupakan sebuah dokumen yang digunakan untuk memberikan informasi rekap PO selama bulan tertentu.
14.	Produk (<i>Business Object, System Object, Analysis Object</i> dan <i>Design Object</i>)	Merupakan produk yang dijual oleh PT. Trivia Nusantara
15.	C_Produk (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk mengatur alur jalannya data yang berhubungan dengan produk.
16.	E_Produk (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk menyimpan data yang berhubungan dengan produk.
17.	C_PurchaseOrder (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk mengatur alur jalannya data yang berhubungan dengan <i>purchase order</i> .
18.	E_PurchaseOrder (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk menyimpan data yang berhubungan dengan <i>purchase order</i> .
19.	C_Auth (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk mengatur alur jalannya data yang berhubungan dengan pelanggan.
20.	E_Pelanggan (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan sebuah <i>class</i> yang digunakan untuk menyimpan data yang berhubungan dengan pelanggan.
21.	V_Purchase Order (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan halaman yang berhubungan dengan objek <i>purchase order</i> .

No.	Istilah	Deskripsi
22.	V_Pengguna (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan halaman yang berhubungan dengan pengguna.
23.	V_Autentikasi (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan halaman yang berhubungan dengan autentikasi pengguna
24.	V_Produk (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan halaman yang berhubungan dengan produk.
25.	V_Konfirmasi (<i>Analysis object</i> dan <i>Design Object</i>)	Merupakan halaman untuk melakukan konfirmasi bagi bagian keuangan pelanggan

Kesimpulan dari fase ini adalah terdapat dua tahap yaitu Proyek *Genesis* dan Membuat *Glossary*. Pada tahap proyek *genesis* didapatkan konsep sistem seperti apa yang diinginkan oleh pelanggan. Pada tahap ini didapatkan tujuh kebutuhan pengguna yaitu penggunaan *user* untuk membedakan hak aksesnya, mengelola informasi produk, mendapatkan informasi produk, melakukan pemesanan barang, mengelola pesanan, melakukan konfirmasi dan mengelola tagihan.

Sedangkan pada tahap membuat *glossary* melakukan dokumentasi penggunaan istilah apa saja yang ada dalam perancangan sistem untuk memudahkan dalam melakukan satu pemahaman apabila sistem akan dilakukan *maintenance*. Dalam tahap ini didapatkan 25 istilah.

4.3 Fase Requirements

Fase ini menjelaskan tentang kebutuhan dari segi proses bisnis pemesanan barang. Pada fase ini akan mendefinisikan *Business Actor List*, *Business Use Case List*, *Business Activity Diagram*, *User Interface Sketches*, *Business Use Case Detail*, *System Actor List*, *System Use Case List*, *System Use Case Diagram*, *System Use Case Details* dan *Supplementary Requirements* dalam sistem yang akan dibangun.

4.3.1 Business Actor List

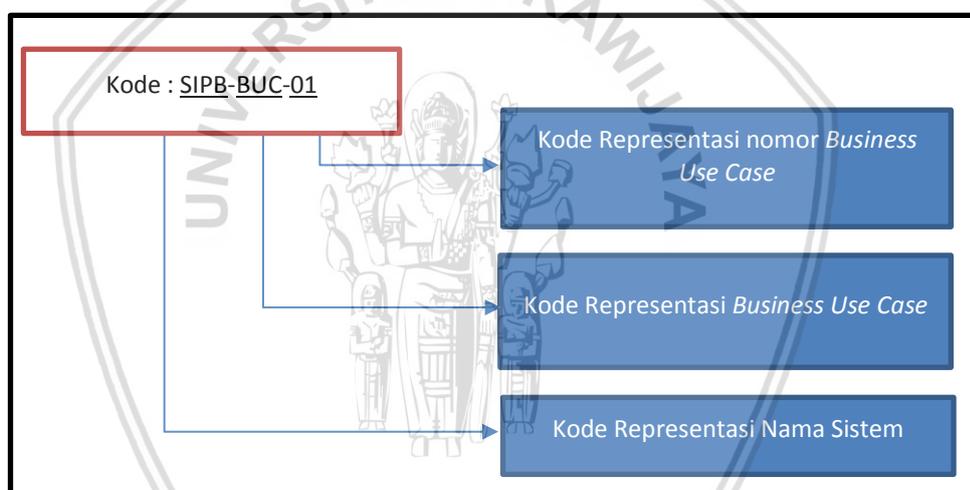
Business Actor List merupakan kumpulan beberapa aktor yang ikut berkecimpung dalam proses bisnis yang dilakukan. *Business Actor* disini dapat berupa seseorang, departemen atau sebuah sistem perangkat lunak terpisah yang berperan dalam sebuah proses bisnis. Berikut adalah *business actor* yang terdapat pada kegiatan pemesanan barang PT. Trivia Nusantara.

1. Pelanggan : Seseorang yang melakukan pemesanan barang
2. Petugas: Seseorang yang bertugas dalam menjelaskan produk dan mencatat barang yang dipesan oleh pelanggan, serta mencetak dokumen yang dibutuhkan sebagai lembar penagihan.

3. Bagian Keuangan Pelanggan: Departemen yang bertugas dalam menyeleksi kebutuhan pembelian dan melakukan konfirmasi barang yang dipesan.
4. Bagian Gudang : Departemen yang bertugas untuk melakukan *packing* barang yang dipesan pelanggan.
5. Kurir : Seseorang yang bertugas mengantar barang ke alamat pelanggan.

4.3.2 Business Use Case List

Business Use Case List merupakan kumpulan definisi dari beberapa *Business Use Case*. *Business Use Case* merupakan deskripsi dari beberapa cara yang digunakan dalam sebuah bisnis, baik oleh pelanggannya, para penggunanya serta interaksi dengan sistem yang lain. *Business use case* harus mempunyai kode dan dijelaskan dengan judul yang singkat. Pemberian nomor bertujuan untuk mempermudah pengguna atau vendor untuk mengidentifikasi kebutuhan pada sistem, berikut merupakan aturan penomoran *business use case* pada sistem informasi pemesanan barang.



Gambar 0.1 Aturan Penomoran *Business Use Case*.

Berikut ini adalah *business use case list* dari proses bisnis pemesanan barang PT. Trivia Nusantara diantaranya:

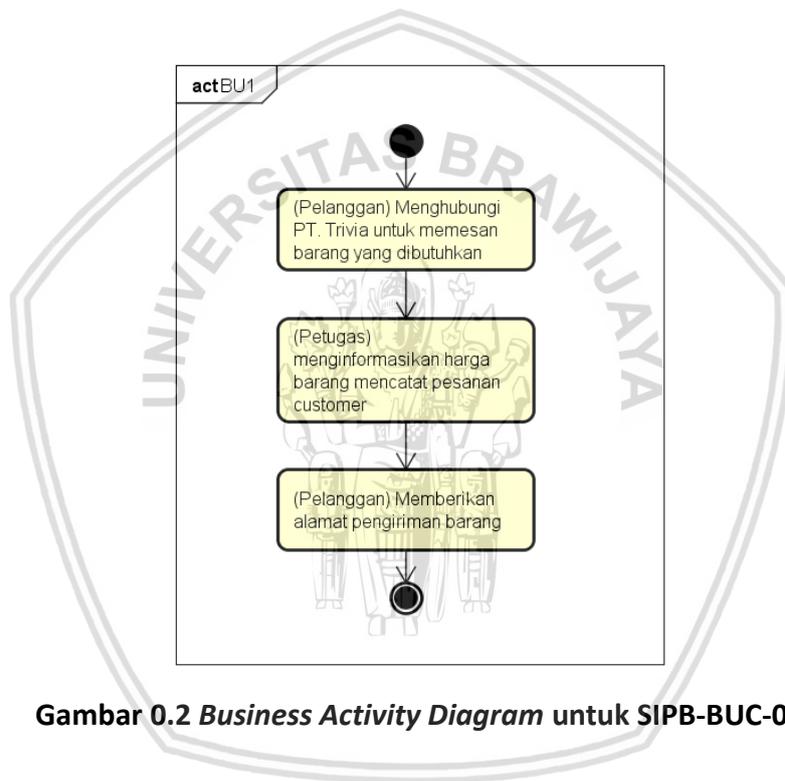
1. SIPB-BUC-01: Pelanggan melakukan pemesanan barang: Pelanggan melakukan pemesanan barang kepada PT. Trivia Nusantara melalui telepon.
2. SIPB-BUC-02: Petugas melakukan konfirmasi kepada bagian keuangan pelanggan : Petugas PT. Trivia Nusantara melakukan konfirmasi terhadap barang yang dipesan serta menginformasikan total pembelian barang tersebut.
3. SIPB-BUC-03: Bagian gudang memeriksa *stock* barang pesanan: Bagian gudang memeriksa ketersediaan barang yang dipesan.

4. SIPB-BUC-04: Kurir mengirim barang dan memberikan surat penagihan: Kurir mengirimkan barang pesanan ke alamat pelanggan serta memberikan surat penagihan pembayaran

4.3.3 Business Activity Diagram

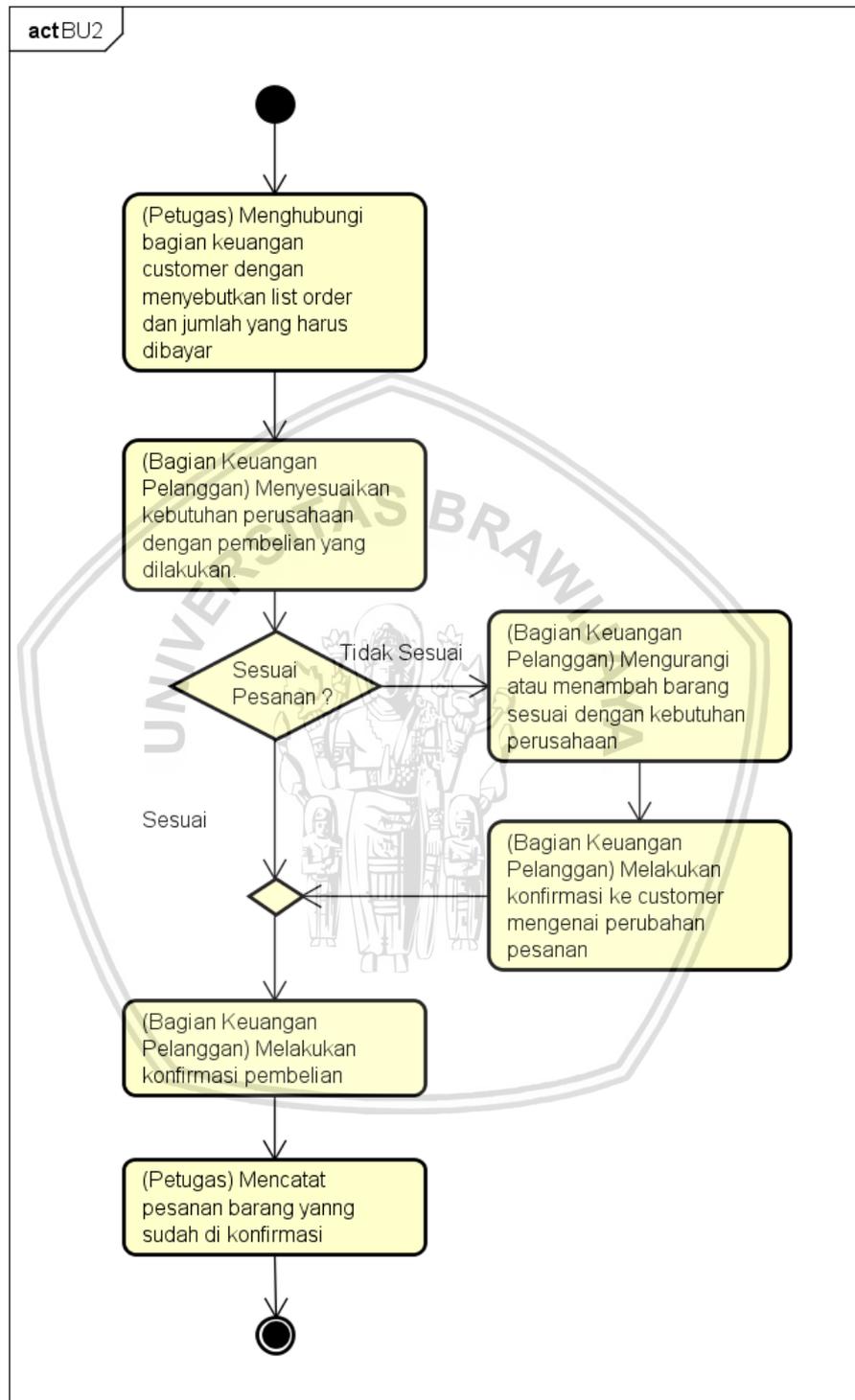
Business activity diagram adalah sebuah diagram yang digunakan untuk merepresentasikan aliran kegiatan dalam suatu proses bisnis. Dengan *activity diagram* akan terlihat awal sebuah proses dan hubungannya dengan proses yang lain, sampai dengan tujuan akhir suatu proses tercapai. Berikut adalah *business activity diagram* yang terjadi pada proses pemesanan barang:

Business activity diagram SIPB-BUC-01 : Pelanggan melakukan pemesanan barang.



Gambar 0.2 Business Activity Diagram untuk SIPB-BUC-01

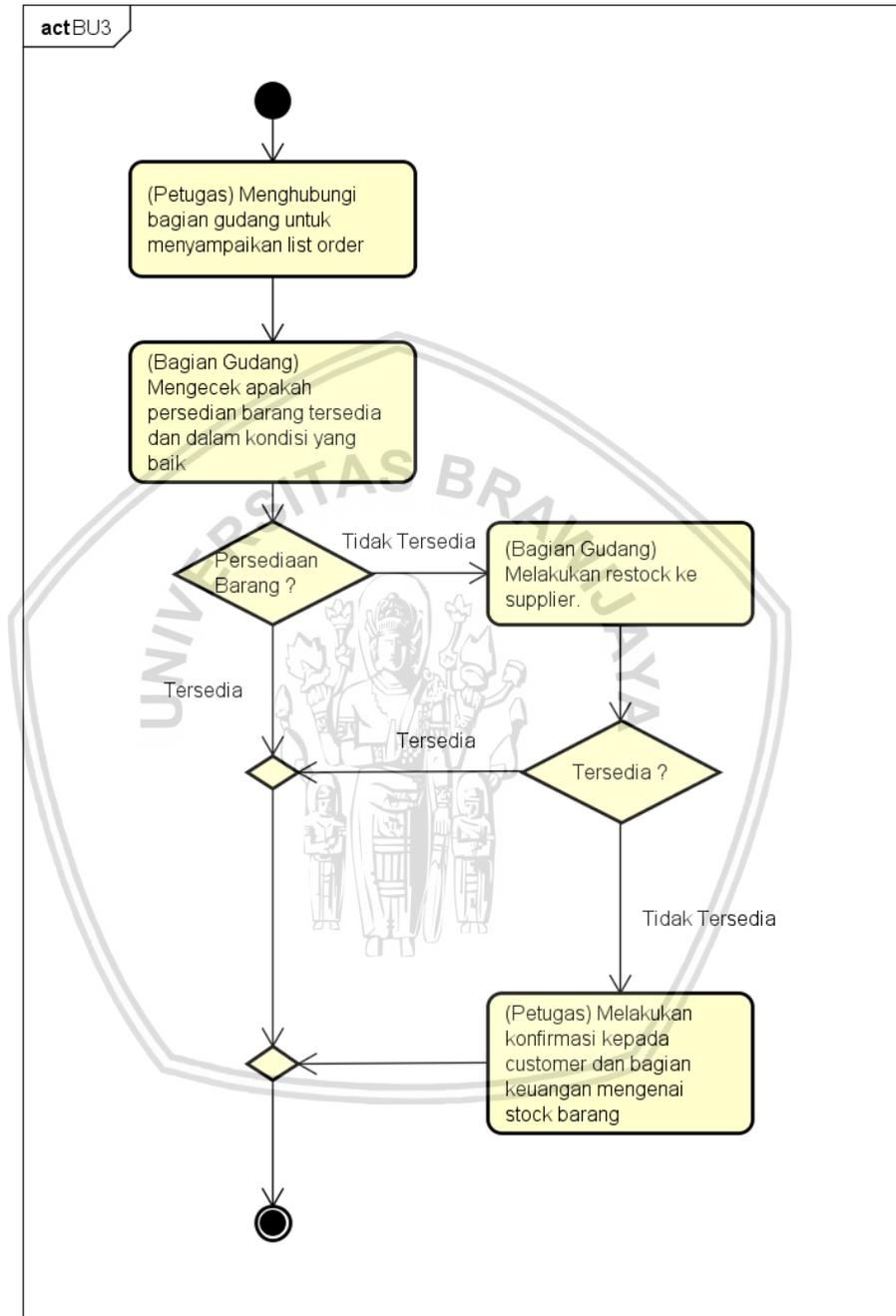
Business activity diagram untuk SIPB-BUC-02 : Petugas melakukan konfirmasi kepada bagian keuangan pelanggan.



Gambar 0.3 Business Activity Diagram untuk SIPB-BUC-02



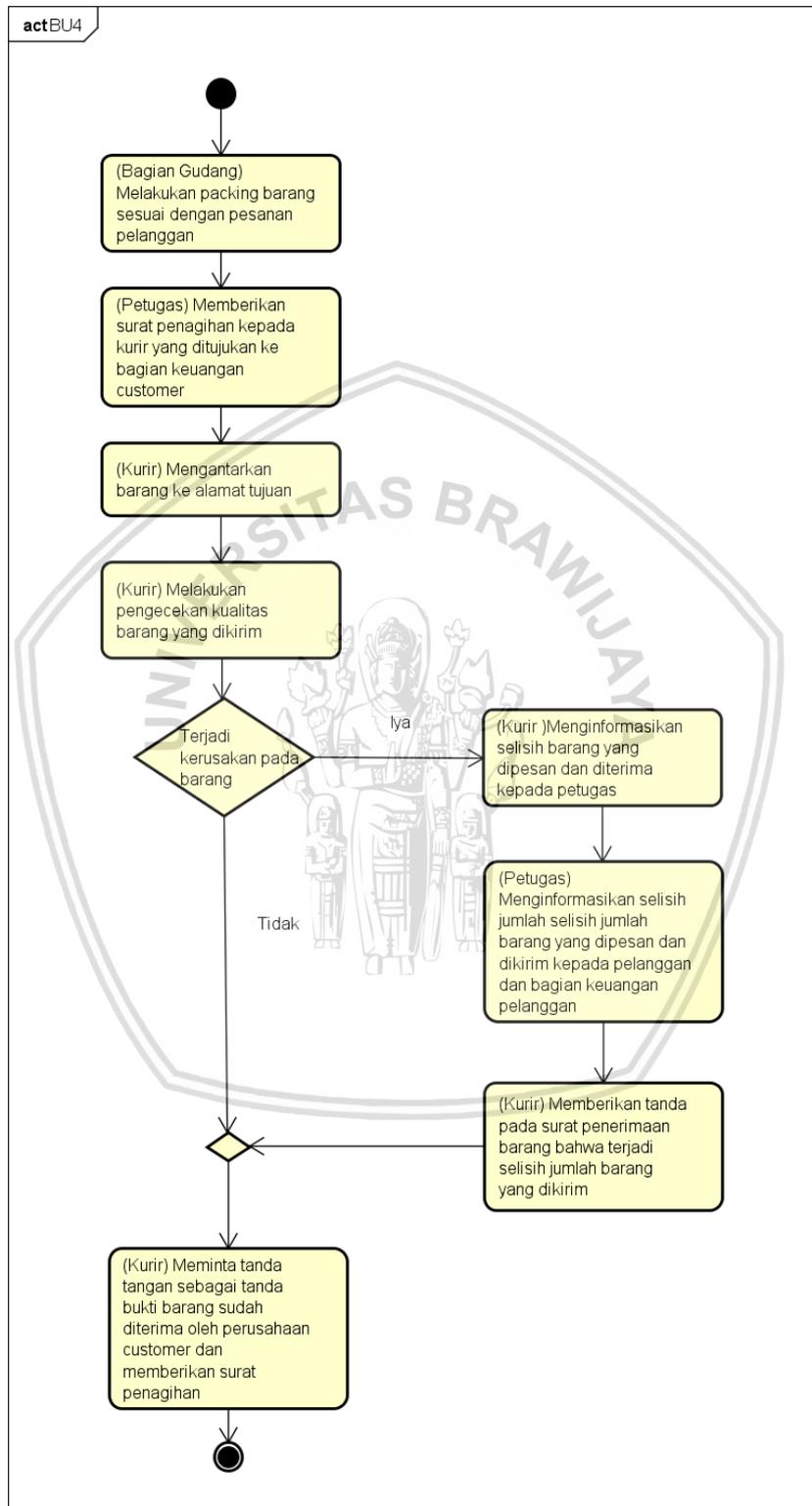
Business activity diagram untuk SIPB-BUC-03 : Bagian gudang memeriksa stock barang pesanan.



Gambar 0.4 Business Activity Diagram untuk SIPB-BUC-03



Business activity diagram untuk SIPB-BUC-04 : Kurir mengirim barang dan memberikan surat penagihan.



Gambar 0.5 Business Activity Diagram untuk SIPB-BUC-04



4.3.4 Business Use Case Details

Business use case details merupakan suatu bentuk deskripsi yang menjelaskan langkah-langkah yang terdapat pada *business use case*. Berikut adalah *business use case details* dari proses pemesanan barang pada PT. Trivia Nusantara.

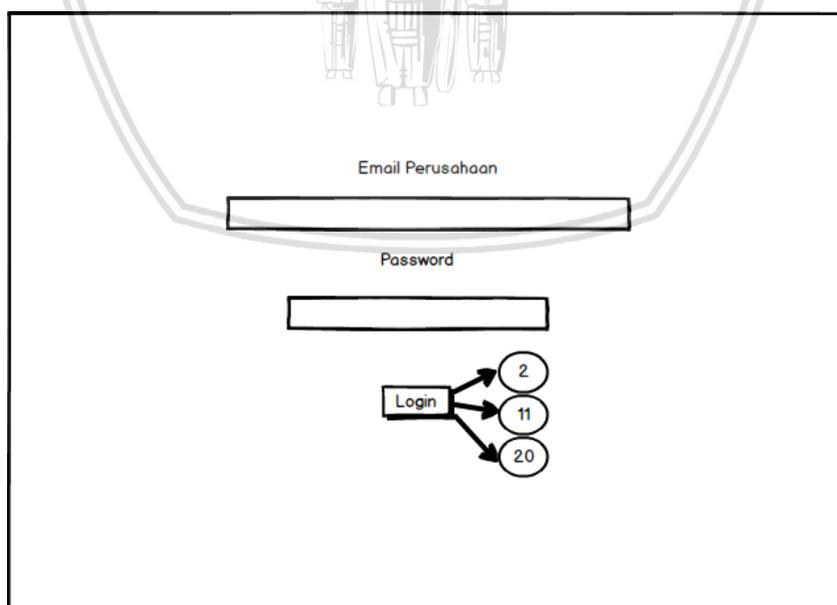
- SIPB-BUC-01 : Pelanggan melakukan pemesanan barang.
 1. Pelanggan menghubungi PT. Trivia Nusantara untuk memesan barang yang dibutuhkan.
 2. Petugas menginformasikan harga barang mencatat pesanan pelanggan.
 3. Pelanggan memberikan alamat pengiriman barang.
- SIPB-BUC-02 : Petugas melakukan konfirmasi kepada bagian keuangan pelanggan.
 1. Petugas menghubungi bagian keuangan pelanggan dengan menyebutkan *list order* dan jumlah yang harus dibayar.
 2. Bagian keuangan pelanggan menyesuaikan kebutuhan perusahaan dengan pembelian yang dilakukan.
 3. Apabila ada ketidaksesuaian maka bagian keuangan pelanggan akan mengurangi atau menambah barang sesuai dengan kebutuhan perusahaan.
 4. Dikarenakan status pemesanan telah diubah maka bagian keuangan pelanggan melakukan konfirmasi kepada pelanggan mengenai perubahan status pemesanan.
 5. Bagian keuangan pelanggan melakukan konfirmasi pembelian.
 6. Petugas mencatat pesanan barang yang sudah dikonfirmasi.
- SIPB-BUC-03 : Bagian Gudang memeriksa *stock* barang pesanan.
 1. Petugas menghubungi bagian gudang untuk menyampaikan *list order*.
 2. Bagian gudang mengecek apakah persediaan barang tersedia dan kondisi barang dalam kondisi baik.
 3. Apabila barang yang dipesan dalam keadaan *out of stock* atau beberapa dalam keadaan buruk maka bagian gudang melakukan *restock* ke *supplier*.
 4. Bagian gudang menunggu barang tersedia apabila barang pada *supplier* tidak tersedia maka petugas melakukan konfirmasi kepada pelanggan dan bagian keuangan pelanggan mengenai barang yang *out of stock*.
- SIPB-BUC-04 : Kurir mengirimkan barang dan memberikan surat penagihan.
 1. Bagian Gudang menyiapkan barang yang dipesan serta melakukan *packing*.

2. Petugas membuat surat tagihan dan memberikan kepada kurir, yang nantinya akan diberikan kepada bagian keuangan pelanggan.
3. Kurir mengirim barang ke alamat tujuan
4. Kurir melakukan *quality control* terhadap barang yang dikirim.
5. Apabila terjadi kerusakan kurir akan menginformasikan jumlah selisih barang kepada petugas.
6. Petugas menginformasikan selisih barang yang dipesan dengan barang yang dikirim ke pelanggan dan bagian keuangan pelanggan.
7. Kurir menuliskan perbedaan tersebut ke lembar penerimaan barang.
8. Kurir meminta tanda tangan sebagai tanda bukti barang sudah diterima oleh perusahaan pelanggan dan memberikan surat penagihan.

4.3.5 User Interface Sketches

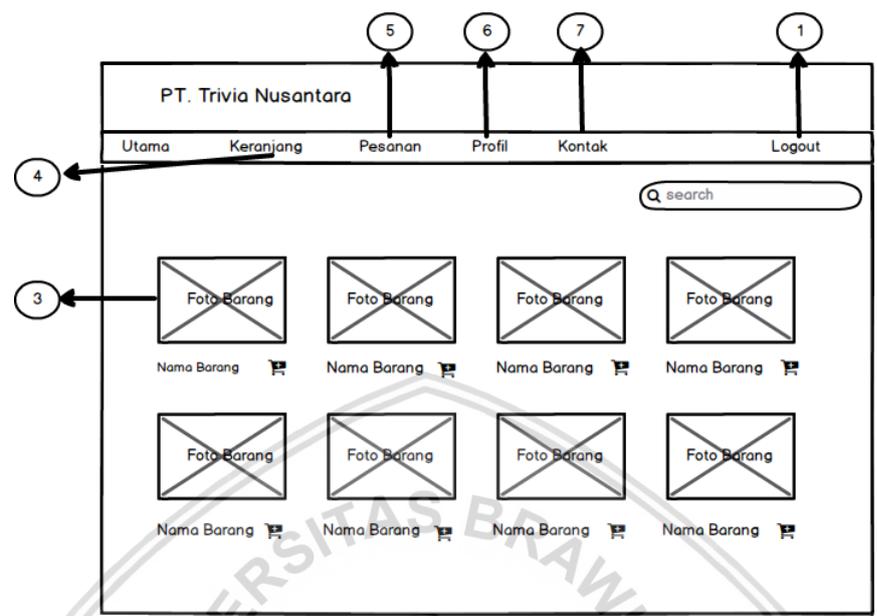
User Interfaces Sketches digunakan untuk melakukan *brainstorming* kepada PT. Trivia Nusantara mengenai tampilan sistem yang akan dibangun. *User Interfaces Sketches* akan digambarkan pada gambar 4.6-4.28. Nomor yang tertera pada gambar adalah suatu reaksi apabila sistem bekerja maka akan melanjutkan ke halaman UI tersebut.

User Interfaces Sketches 1 merupakan halaman untuk melakukan *login*. Setelah melakukan *login* maka akan teridentifikasi perannya dalam sistem apabila pengguna teridentifikasi menjadi pelanggan maka akan dilanjutkan ke UI 2, apabila menjadi admin dilanjutkan ke UI 11 dan apabila menjadi pegawai perusahaan dilanjutkan ke UI 20.



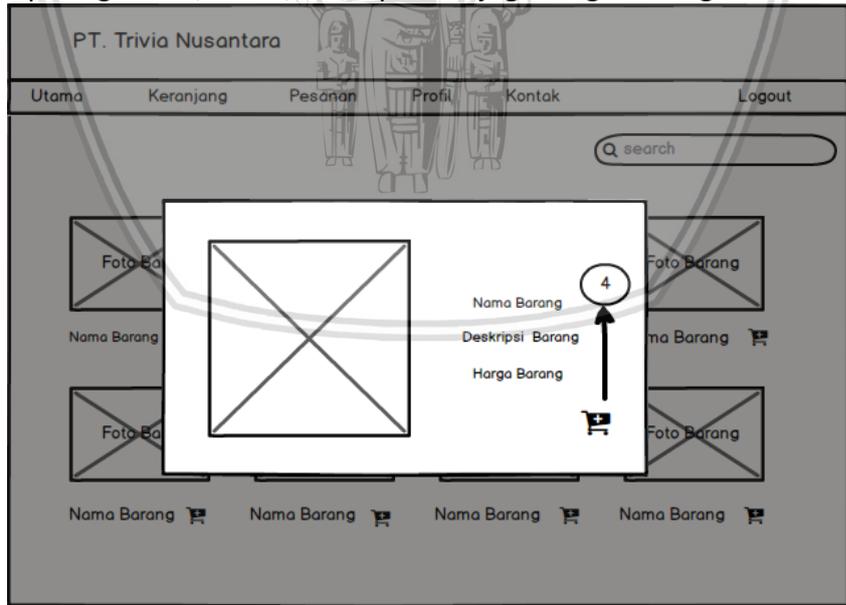
Gambar 0.6 User Interfaces Sketches 1 (Login)

User Interfaces Sketches 2 merupakan sebuah halaman utama bagi pelanggan, halaman ini digunakan untuk menampilkan produk barang yang dijual.



Gambar 0.7 User Interfaces Sketches 2 (Halaman Produk Barang)

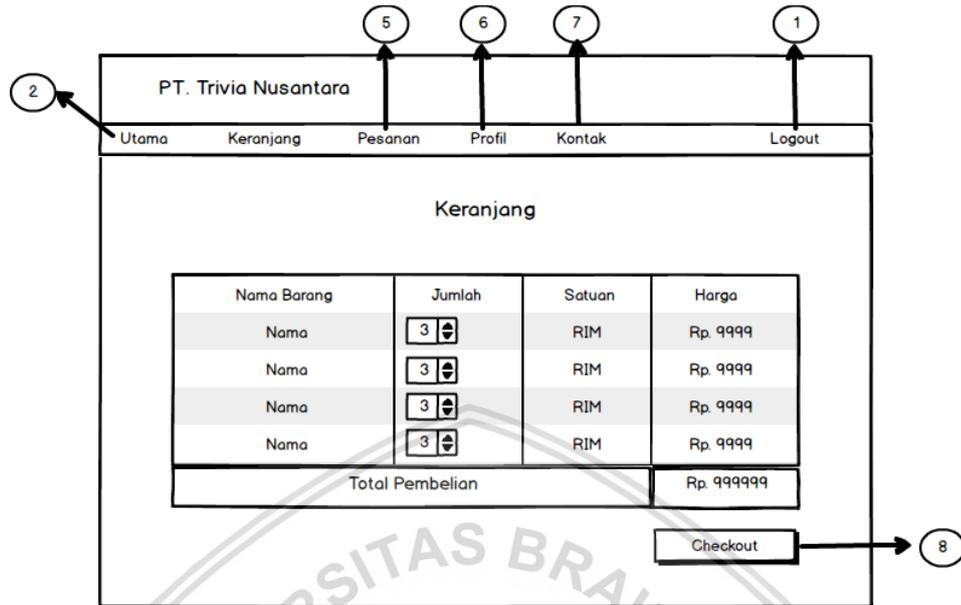
User Interfaces Sketches 3 merupakan halaman untuk menampilkan detail produk seperti gambar, nama, deskripsi dan juga harga barang.



Gambar 0.8 User Interfaces Sketches 3 (Menampilkan Detail Barang)

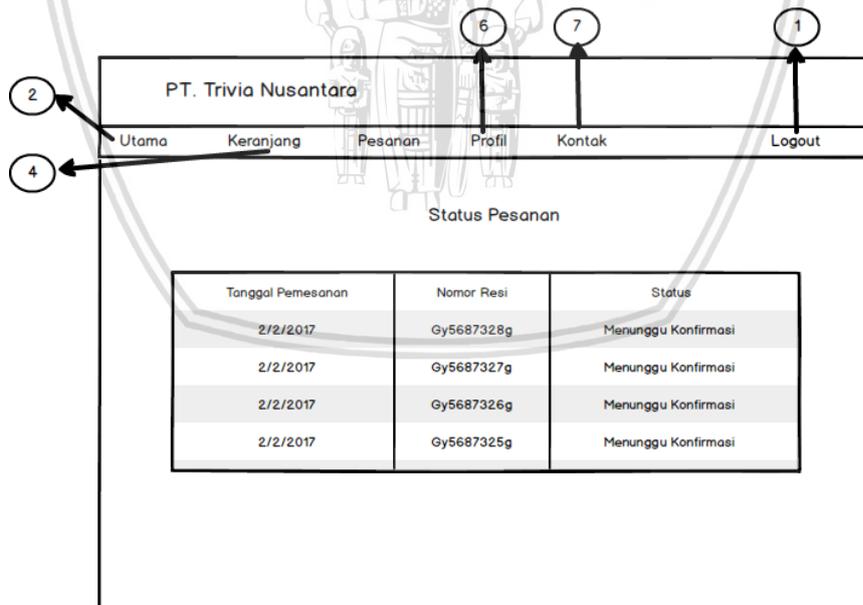


User Interfaces Sketches 4 merupakan halaman untuk menampilkan *list* barang apa saja yang akan dibeli oleh pelanggan.



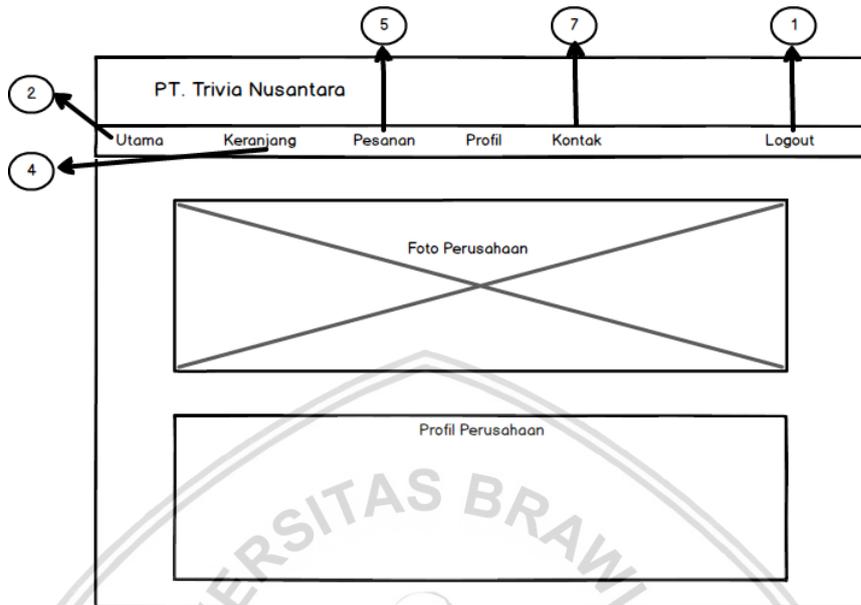
Gambar 0.9 User Interfaces Sketches 4 (Troli Pelanggan)

User Interfaces Sketches 5 merupakan halaman yang digunakan untuk menampilkan *list* status PO yang dilakukan oleh pelanggan.



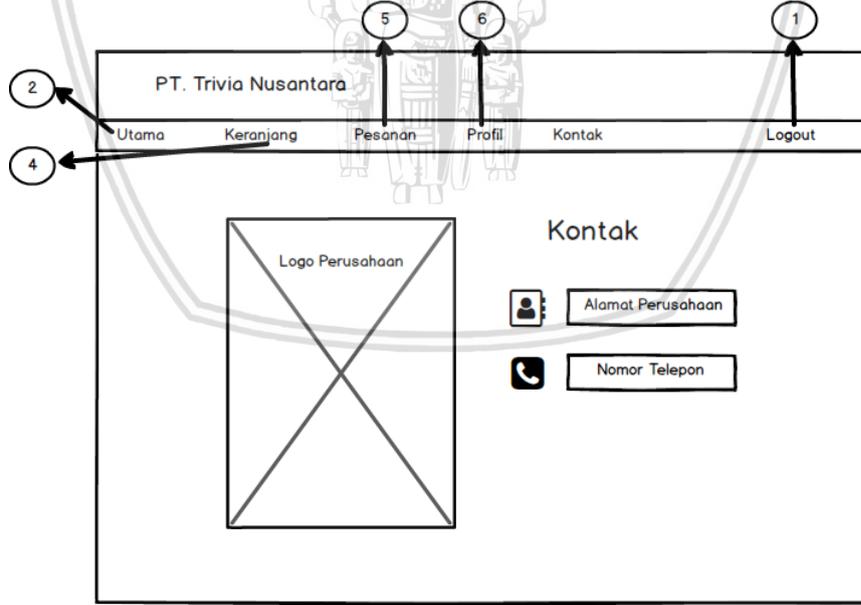
Gambar 0.10 User Interfaces Sketches 5 (Status Pesanan Pelanggan)

User Interfaces Sketches 6 merupakan halaman yang menampilkan profil PT. Trivia Nusantara



Gambar 0.11 User Interfaces Sketches 6 (Profil PT. Trivia Nusantara)

User Interfaces Sketches 7 merupakan halaman yang digunakan untuk menampilkan informasi kontak PT. Trivia Nusantara



Gambar 0.12 User Interfaces Sketches 7 (Kontak PT. Trivia Nusantara)



User Interfaces Sketches 8 merupakan halaman yang digunakan untuk mengisi form data perusahaan setelah melakukan pemesanan barang.

Gambar 0.13 User Interfaces Sketches 8 (Form Data Perusahaan Pelanggan)

User Interfaces Sketches 9 merupakan halaman yang digunakan untuk menampilkan kelengkapan pemesanan yaitu produk, jumlah dan juga alamat pengiriman beserta detail PO .

Nama Barang	Jumlah	Satuan	Harga
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Total Pembelian			Rp. 999999

Gambar 0.14 User Interfaces Sketches 9 (Halaman Detail Pemesanan)



User Interfaces Sketches 10 merupakan halaman yang digunakan oleh bagian keuangan pelanggan untuk melakukan konfirmasi pemesanan.

PT. Trivia Nusantara

Purchase Order

Nama Perusahaan : PT. XYZ
 Bagian/Divisi : Divisi Produksi
 Contact Person : 08xxxx
 Alamat Pengiriman : JL. xyz

Gy5687328g

Nama Barang	Jumlah	Satuan	Harga
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Total Pembelian			Rp. 999999

Verifikasi

Gambar 0.15 *User Interfaces Sketches 10* (Halaman Verifikasi)

User Interfaces Sketches 11 merupakan halaman utama admin, halam ini berisi *list* produk yang dijual.

PT. Trivia Nusantara

Produk Pesanan Tagihan Logout

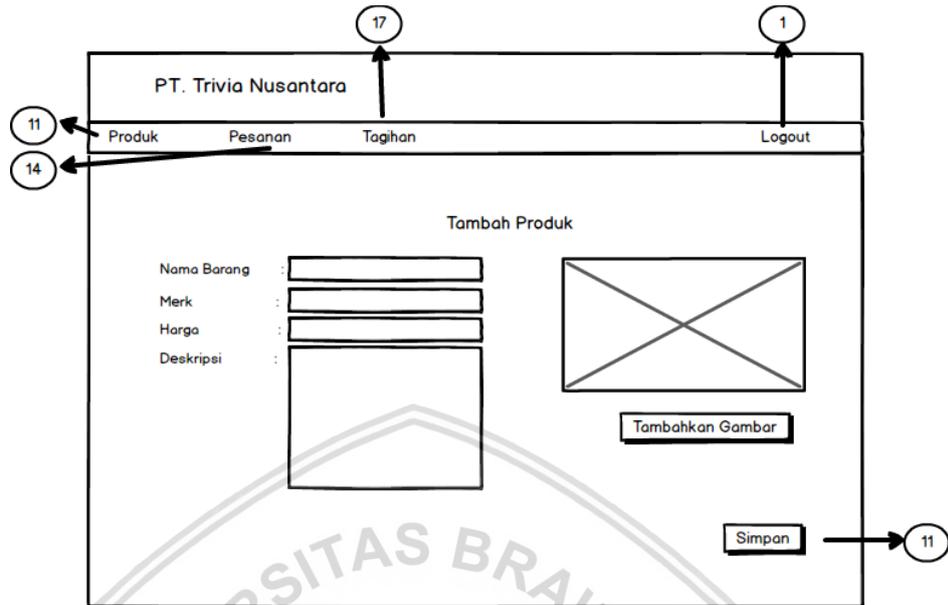
Produk

+ Tambahkan Produk

No	Kode Barang	Nama Barang	Merk	Gambar	Deskripsi	Ket
1	PJ1234	Pensil Isi	Joyko	<input type="checkbox"/>	Pensil isi mekanik	Edit
2	PJ1234	Pensil Isi	Joyko	<input type="checkbox"/>	Pensil isi mekanik	Edit
3	PJ1234	Pensil Isi	Joyko	<input type="checkbox"/>	Pensil isi mekanik	Edit
4	PJ1234	Pensil Isi	Joyko	<input type="checkbox"/>	Pensil isi mekanik	Edit

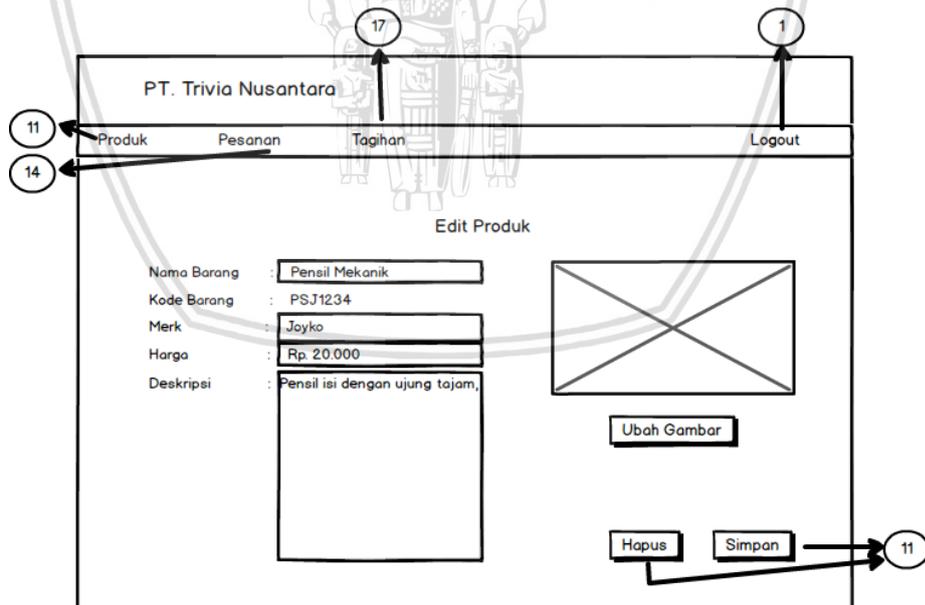
Gambar 0.16 *User Interfaces Sketches 11* (Halaman Produk Admin)

User Interfaces Sketches 12 merupakan halaman yang digunakan untuk menambah produk.



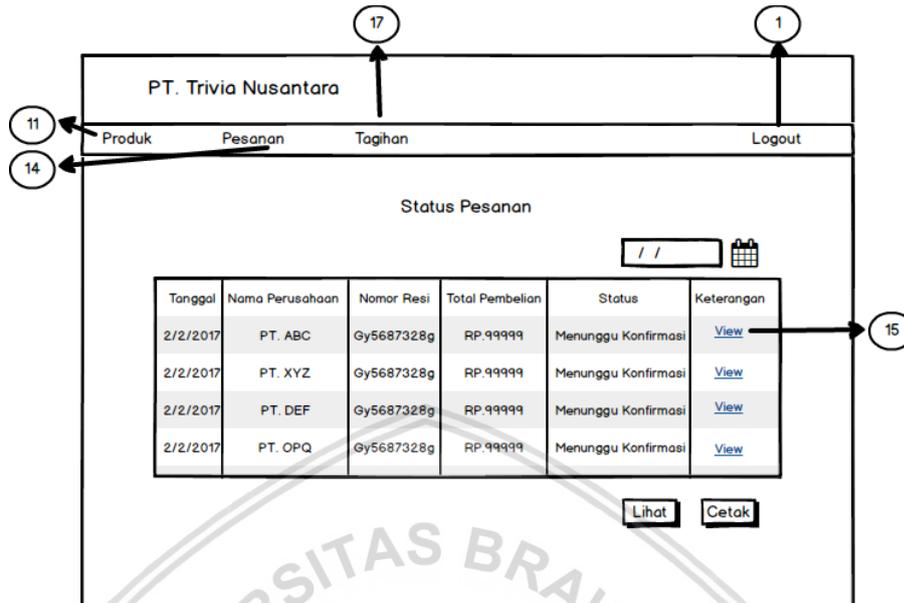
Gambar 0.17 *User Interfaces Sketches 12* (Menambahkan Produk)

User Interfaces Sketches 13 merupakan halaman yang digunakan untuk mengedit detail produk.



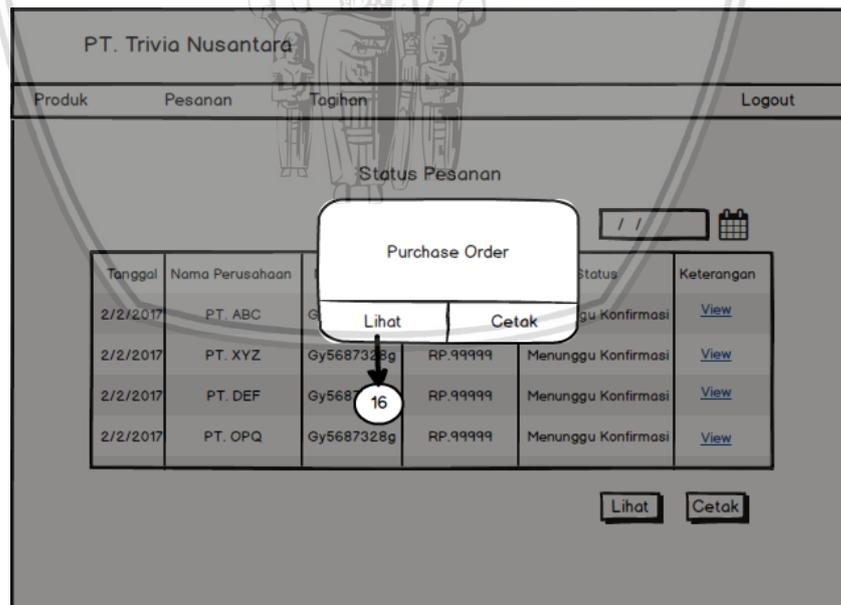
Gambar 0.18 *User Interfaces Sketches 13* (Mengedit Produk)

User Interfaces Sketches 14 merupakan halaman yang digunakan untuk menampilkan status PO bagi admin.



Gambar 0.19 User Interfaces Sketches 14 (Halaman Pesanan Admin)

User Interfaces Sketches 15 merupakan halaman yang digunakan oleh admin untuk mencetak lembar PO



Gambar 0.20 User Interfaces Sketches 15 (Halaman Cetak Pesanan Admin)

User Interfaces Sketches 16 merupakan lembar PO

PT. Trivia Nusantara

Purchase Order

Nama Perusahaan : PT. XYZ
Bagian/Divisi : Divisi Produksi
Contact Person : 08xxxx
Alamat Pengiriman : JL. xyz



Gy5687328g

Nama Barang	Jumlah	Satuan	Harga
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Nama	3	RIM	Rp. 9999
Total Pembelian			Rp. 999999

Gambar 0.21 User Interfaces Sketches 16 (Halaman Lembar Purchase Order)

User Interfaces Sketches 17 merupakan halaman yang menampilkan tagihan.

PT. Trivia Nusantara

17

1

11 Produk Pesanan Tagihan Logout

14

Tagihan

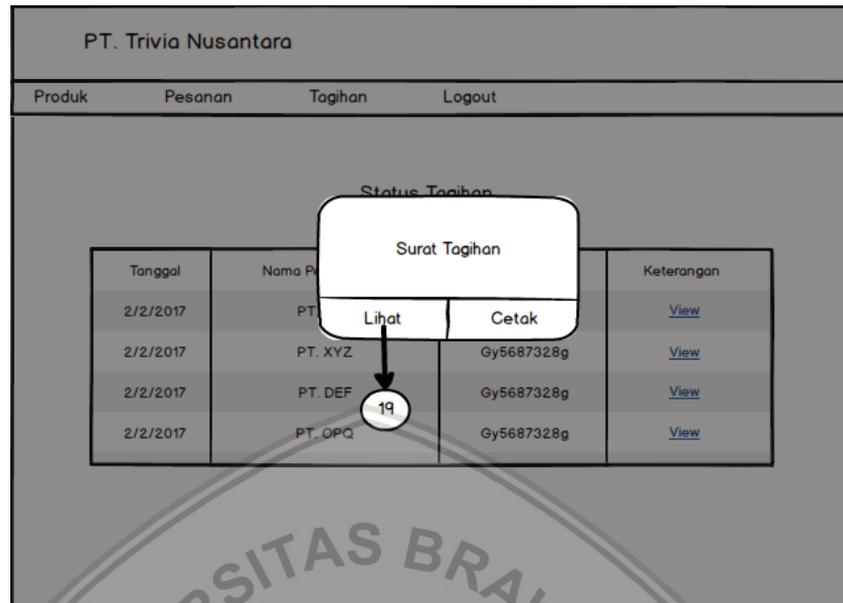
Tanggal	Nama Perusahaan	Nomor Resi	Keterangan
2/2/2017	PT. ABC	Gy5687328g	View
2/2/2017	PT. XYZ	Gy5687328g	View
2/2/2017	PT. DEF	Gy5687328g	View
2/2/2017	PT. OPQ	Gy5687328g	View

18

Gambar 0.22 User Interfaces Sketches 17 (Halaman Tagihan Admin)

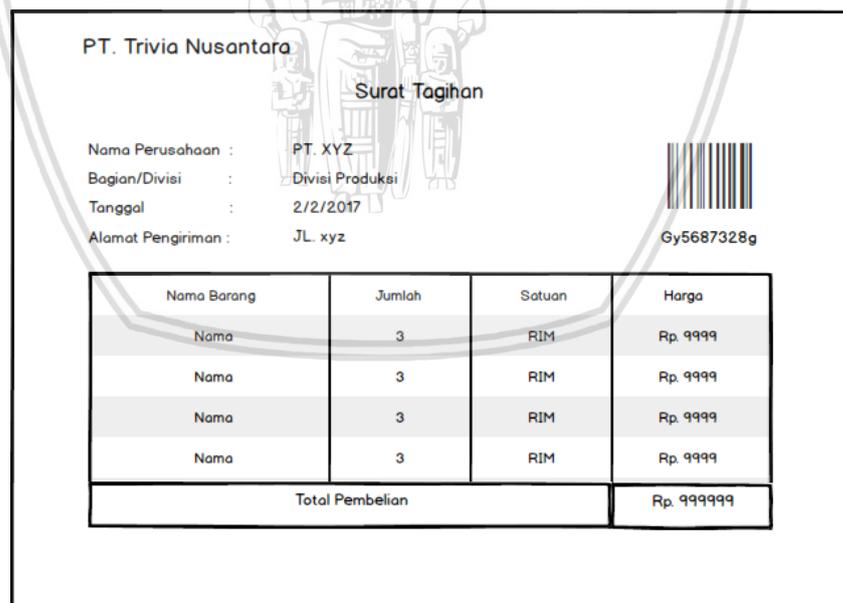


User Interfaces Sketches 18 digunakan untuk menampilkan pilihan mengenai surat tagihan apakah ingin di cetak atau ditampilkan.



Gambar 0.23 User Interfaces Sketches 18 (Halaman Cetak Tagihan Admin)

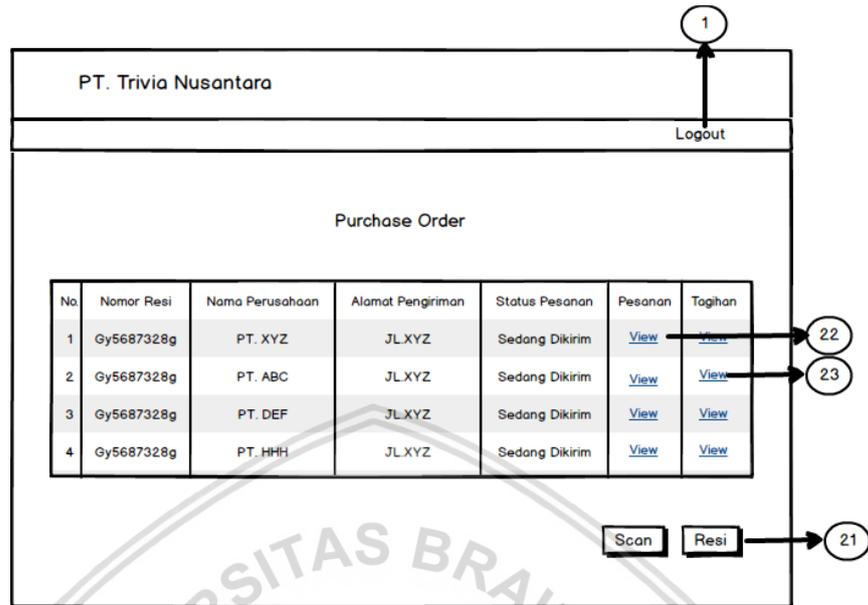
User Interfaces Sketches 19 merupakan tampilan surat tagihan.



Gambar 0.24 User Interfaces Sketches 19 (Surat Penagihan)

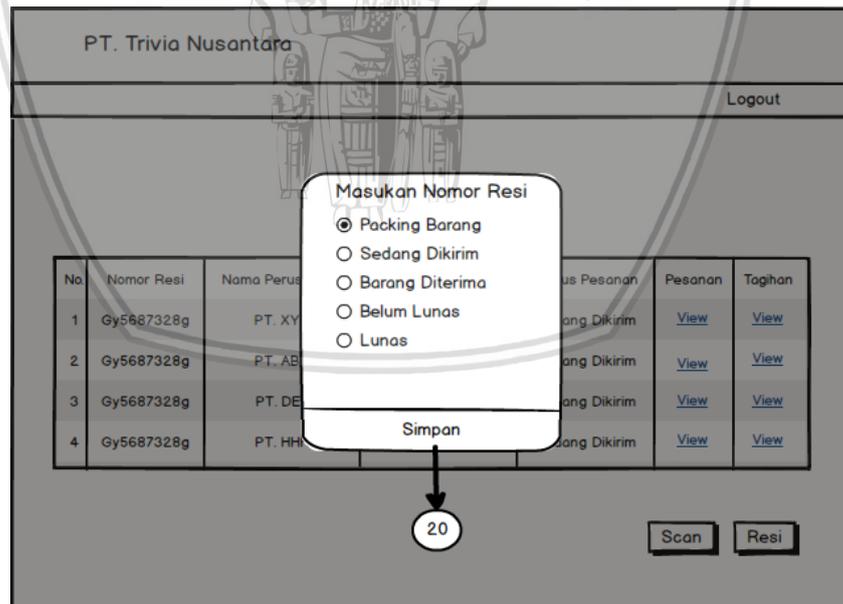


User Interfaces Sketches 20 merupakan halaman utama sistem bagi bagian gudang.



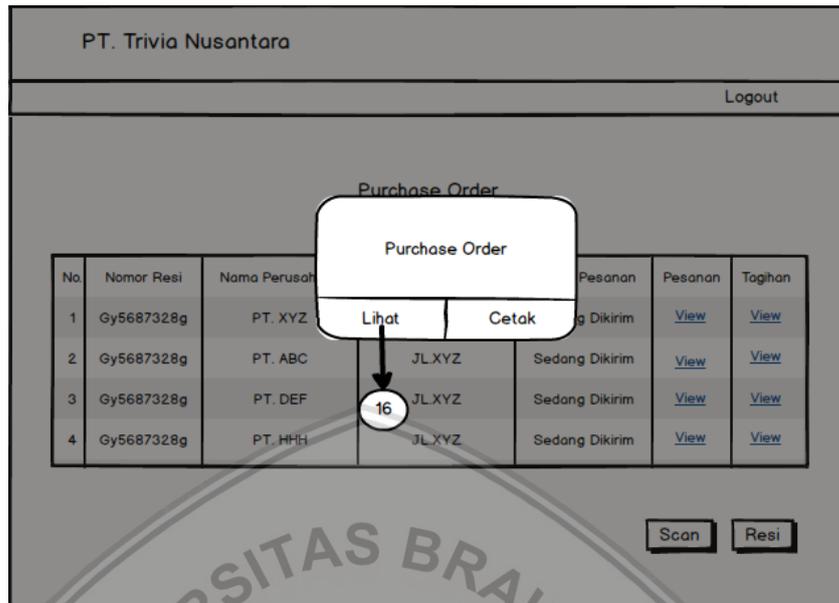
Gambar 0.25 User Interfaces Sketches 20 (Halaman Utama Bagian Gudang)

User Interfaces Sketches 21 merupakan halaman yang digunakan bagi bagian gudang untuk mengubah status PO



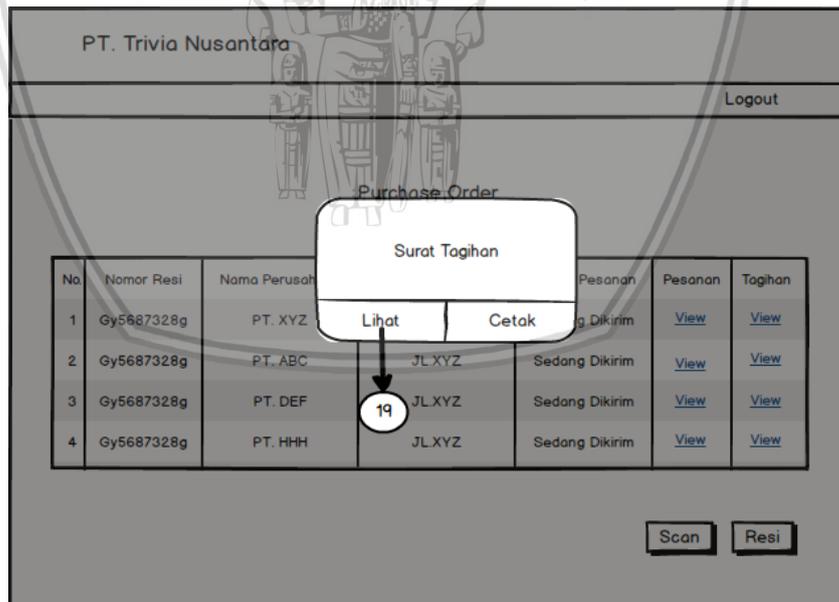
Gambar 0.26 User Interfaces Sketches 21 (Halaman Mengubah Status Pesanan)

User Interfaces Sketches 22 merupakan halaman yang digunakan oleh bagian gudang untuk memilih lembar PO ingin dicetak atau ditampilkan



Gambar 0.27 *User Interfaces Sketches 22* (Halaman Cetak Purchase Order Bag. Gudang)

User Interfaces Sketches 23 merupakan halaman yang digunakan oleh bagian gudang untuk memilih surat tagihan ingin dicetak atau ditampilkan



Gambar 0.28 *User Interfaces Sketches 23* (Halaman Cetak Tagihan Bag. Gudang)

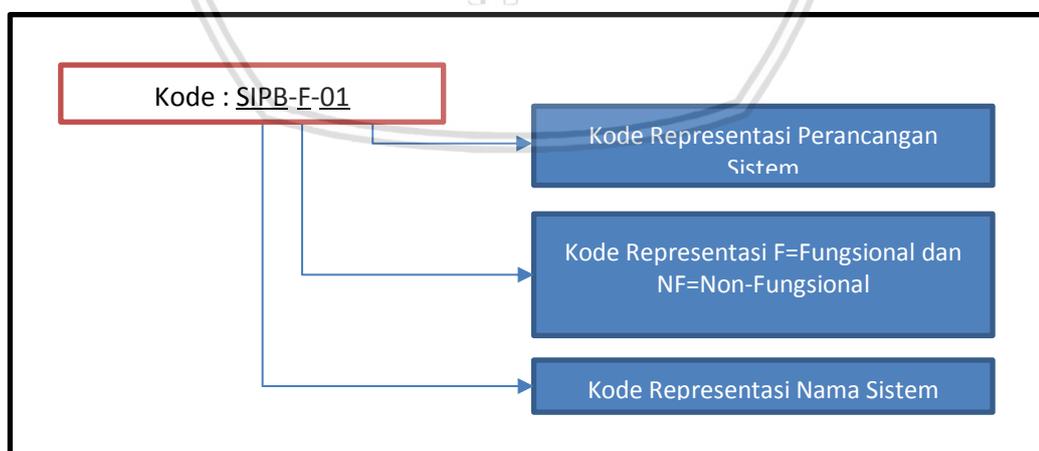
4.3.6 System Actor List

System Actor List adalah seseorang yang turut berperan aktif dalam menggunakan sistem. *System Actor* dapat berupa seseorang ataupun sistem eksternal. Berikut adalah *system actor* yang terdapat pada proses pemesanan barang PT. Trivia Nusantara.

1. Pengguna : Pengguna merupakan orang yang mengakses sistem dari luar dan hanya dapat melihat tampilan awal sistem dan melakukan *login* untuk melakukan autentikasi terhadap pengguna tersebut untuk menggunakan sistem lebih lanjut.
2. Pelanggan : Pelanggan merupakan seseorang yang dapat mengakses sistem untuk melakukan proses pemesanan barang.
3. Admin: Admin merupakan seseorang yang memiliki hak untuk melakukan manajemen data pada sistem informasi.
4. Bagian keuangan pelanggan: Bagian keuangan pelanggan merupakan sebuah departemen atau divisi dari perusahaan pelanggan yang memiliki hak untuk melakukan konfirmasi pembelian barang.
5. Pegawai Perusahaan: Pegawai perusahaan merupakan seseorang yang memiliki hak untuk mengubah status pesanan barang.

4.3.7 System Use Case List

System Use Case List merupakan kumpulan definisi dari beberapa *system use case*. *System use case* merupakan deskripsi statis dari beberapa cara yang digunakan oleh *system actor* dalam sebuah sistem yang diusulkan. *System use case* harus mempunyai kode dan dijelaskan dengan judul yang singkat. Pemberian nomor bertujuan untuk mempermudah pengguna atau vendor untuk mengidentifikasi kebutuhan pada sistem, berikut merupakan aturan penomoran sistem informasi pemesanan barang.



Gambar 0.29 Aturan Penomoran System Use Case

Dalam sub-bab *Project Genesis* disebutkan terdapat beberapa kegiatan yang dapat dibantu oleh sistem dan berdasarkan analisis proses bisnis, wawancara dan *user story* yang dilakukan menghasilkan beberapa fitur yaitu:

Tabel 0.2 Deskripsi Fitur Sistem

Fitur	Deskripsi
Penggunaan Akun	Sistem dapat digunakan untuk melakukan pendaftaran akun dan menyimpan data pengguna. Data tersebut meliputi nama perusahaan, alamat perusahaan, <i>email</i> perusahaan, <i>email</i> bagian keuangan. Serta sistem juga dapat mengidentifikasi hak akses setiap pengguna sistem, sehingga informasi dan layanan yang disediakan oleh sistem berbeda setiap jenis pengguna.
Mengelola Informasi Produk	Sistem dapat digunakan untuk mengelola informasi produk seperti menambahkan, menghapus atau mengubah informasi produk dan perubahan tersebut disimpan dalam sistem. Informasi produk yang disimpan meliputi nama produk, harga produk, <i>merk</i> produk, deskripsi produk serta gambar produk.
Mendapat Informasi Produk	Sistem dapat digunakan untuk menampilkan informasi produk yang dipilih oleh pengguna serta sistem dapat digunakan untuk mencari produk yang diinginkan oleh pelanggan.
Memesan Barang	Sistem dapat digunakan untuk melakukan pemesanan barang dan menyimpan pesanan barang tersebut ke dalam sistem. Sistem dapat digunakan dengan mudah sehingga pelanggan dapat memanfaatkan sistem dengan baik.
Konfirmasi Pesanan	Sistem dapat digunakan untuk mengirimkan informasi barang yang dipesan dan melakukan konfirmasi pemesanan melalui <i>email</i> .
Mengelola Pesanan	Sistem dapat digunakan untuk menampilkan pesanan yang dilakukan oleh pelanggan, mendapatkan rekap pemesanan selama rentang waktu yang diinginkan pengguna serta dapat memberikan status pesanan barang sehingga pelanggan dapat mengetahui status barang yang dipesan.
Mengelola Tagihan	Sistem dapat digunakan untuk mendapatkan dokumen tagihan secara otomatis.

Setelah didapatkan fitur sistem didapatkan beberapa spesifikasi kebutuhan sistem. Terdapat dua jenis kebutuhan sistem yaitu kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional merupakan kebutuhan yang harus dipenuhi oleh sistem agar fitur sistem dapat berjalan dengan baik. Sedangkan kebutuhan non-fungsional menentukan kualitas lain yang harus dimiliki oleh sistem seperti yang terkait dengan kegunaan, keandalan, kinerja, dan dukungan sistem. Berikut ini adalah kebutuhan fungsional dari sistem yang akan dibangun.

Tabel 0.3 Kebutuhan Fungsional Sistem

Fitur	Deskripsi Spesifikasi
Penggunaan Akun	Sistem dapat digunakan untuk membuat akun dengan memasukan nama perusahaan, alamat perusahaan, <i>email</i> perusahaan, <i>email</i> bagian keuangan perusahaan, dan nomor telepon perusahaan.
	Sistem dapat digunakan untuk melakukan <i>login</i> sehingga dapat diketahui hak akses pengguna yang masuk kedalam sistem.
Mengelola Informasi Produk	Sistem dapat digunakan untuk menambah informasi produk meliputi nama produk, harga produk, <i>merk</i> produk, deskripsi produk serta gambar produk.
	Sistem dapat digunakan untuk mengubah informasi produk
	Sistem dapat digunakan untuk menghapus informasi produk
Mendapat Informasi Produk	Sistem dapat digunakan untuk melihat informasi produk
	Sistem dapat digunakan untuk mencari informasi produk sesuai dengan masukan yang diberikan pelanggan dalam sistem.
Memesan Barang	Sistem dapat digunakan untuk melakukan pemesanan barang.
Mengelola Pesanan	Sistem dapat menampilkan dokumen PO.
	Sistem dapat digunakan untuk mencetak PO
	Sistem dapat menampilkan rekap penjualan sesuai dengan rentang waktu yang dimasukkan oleh pengguna.
	Sistem dapat digunakan untuk mencetak rekap penjualan.
	Sistem dapat mengubah status PO.
Konfirmasi Pesanan	Sistem dapat digunakan untuk mengirim <i>form</i> konfirmasi pesanan kepada <i>email</i> bagian keuangan.
Mengelola Tagihan	Sistem dapat menampilkan dokumen tagihan
	Sistem dapat mencetak dokumen tagihan

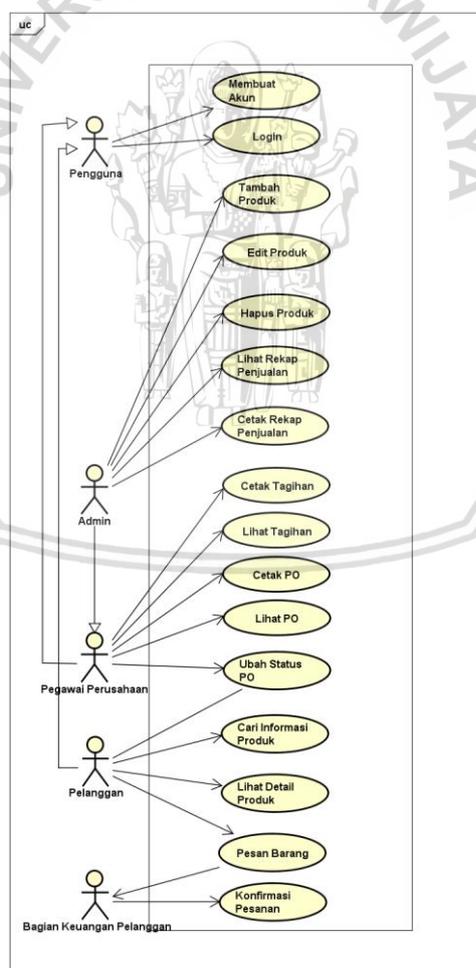
Dari hasil kebutuhan fungsional tersebut akan di jadikan dasar pembuatan sebuah *use case*. Berikut ini adalah *use case* pada sistem informasi pemesanan barang.

1. (SIPB-F-01): Buat Akun: Pengguna membuat akun.
2. (SIPB-F-02): Login: Pengguna melakukan *login*.
3. (SIPB-F-03): Tambah Produk: Admin menambah produk
4. (SIPB-F-04): Edit Produk : Admin mengubah informasi produk
5. (SIPB-F-05): Hapus Produk : Admin menghapus produk
6. (SIPB-F-06): Lihat informasi Produk: Pelanggan mendapatkan informasi produk
7. (SIPB-F-07): Cari Produk : Pelanggan mencari produk dengan nama produk
8. (SIPB-F-08): Pesan Barang: Pelanggan memesan barang
9. (SIPB-F-09): Konfirmasi Pesanan: Bagian Keuangan Pelanggan melakukan konfirmasi pesanan.

10. (SIPB-F-10): Lihat PO: Pegawai Perusahaan dan Pelanggan melihat dokumen
11. (SIPB-F-11): Cetak PO : Pegawai Perusahaan mencetak dokumen PO
12. (SIPB-F-12): Lihat Rekap Penjualan : Admin melihat rekap penjualan pada bulan tertentu
13. (SIPB-F-13): Cetak Rekap Penjualan : Admin mencetak rekap penjualan pada bulan tertentu.
14. (SIPB-F-14): Ubah status PO: pegawai perusahaan mengubah status PO.
15. (SIPB-F-15): Lihat Tagihan: Pegawai Perusahaan dapat melihat dokumen tagihan.
16. (SIPB-F-16): Cetak Tagihan Pegawai perusahaan dapat mencetak dokumen tagihan.

4.3.8 System Use Case Diagram

Use case diagram adalah suatu diagram yang menunjukkan hubungan antara *system actor list* dengan *system use case*. *System use case diagram* dapat memberikan sekilas bagaimana sistem akan digunakan oleh pengguna. Berikut adalah *use case diagram* dari proses pemesanan barang PT. Trivia Nusantara.



Gambar 0.30 Use Case Diagram

4.3.9 System Use Case Details

System use case details akan menjelaskan alur jalannya proses yang terjadi pada setiap *use case* dari sisi aktor dan sistem berdasarkan *use case diagram* yang telah dibuat. *Use case details* memiliki dua skenario yaitu *basic course* dan *alternate course*, *basic course* akan menjelaskan skenario sistem yang normal sedangkan *alternate course* akan menjelaskan skenario sistem yang tidak berjalan normal atau terjadi *error*. Berikut *use case details* pada proses pemesanan barang.

1. *System use case details* membuat akun (SIPB-F-01)

(SIPB-F-01):Buat Akun

Preconditions: Pengguna belum terdaftar dalam sistem menjadi pelanggan.

1. Pengguna mengisi semua kolom pada formulir data member. Data yang diisi adalah nama perusahaan, alamat perusahaan, *email* perusahaan, *email* bagian keuangan pelanggan, nomor telepon perusahaan dan *password* (*password* harus terdiri dari 8 karakter) lalu memilih fungsi simpan. Pengguna hanya bisa mengakses form buat akun apabila diberikan link buat akun oleh PT. Trivia Nusantara.
2. Sistem menyimpan data
3. Sistem menampilkan form login.

Postconditions: Pengguna terdaftar dalam sistem

Abnormal Path : A1. Jika memiliki nama perusahaan yang sama maka sistem akan menampilkan form membuat akun dan juga notifikasi bahwa gagal membuat akun.

2. *System use case details* login (SIPB-F-02)

(SIPB-F-02): *Login*

Precondition: Pengguna sudah terdaftar dalam sistem.

1. Pengguna memasukan *email* perusahaan dan *password* yang sudah didaftarkan lalu memilih fungsi *login*.
2. Sistem memeriksa apakah data yang dimasukan oleh *user* sudah terdaftar dalam sistem atau belum
3. Pengguna teridentifikasi perannya dalam sistem.

Postcondition: Pengguna berhasil ke dalam sistem dan teridentifikasi perannya dalam sistem.

3. *System use case details* tambah produk (SIPB-F-03)

(SIPB-F-03): Tambah Produk

Precondition: Admin berhasil melakukan *login*.

1. Admin memilih "Tambahkan Produk"
2. Sistem menampilkan form untuk menambahkan produk yang berisi data nama produk, harga produk, merk produk, deskripsi produk dan gambar
3. Admin menginputkan data dan memilih gambar produk lalu memilih fungsi simpan.
4. Sistem menyimpan data produk
5. Sistem menampilkan halaman produk admin.

Postcondition: Data produk tersimpan dalam sistem

Abnormal Path: Apabila gambar yang disimpan lebih dari 500 KB maka sistem akan menampilkan pop-up bahwa gambar yang di *upload* harus dibawah 500KB dan kembali ke halaman edit produk.

4. System use case details edit produk (SIPB-F-04)

(SIPB-F-04) : Edit Produk

Precondition: Data produk sudah tersimpan dalam sistem.

1. Admin memilih menu *edit* pada *list* produk yang ingin dihapus
2. Sistem menampilkan *form* untuk mengedit produk
3. Admin mengubah informasi produk sesuai yang diinginkan lalu memilih fungsi simpan.
4. Sistem menyimpan perubahan data produk ke dalam sistem.
5. Sistem menampilkan halaman produk.

Postcondition: Data produk diperbarui.

5. System use case details hapus produk (SIPB-F-05)

(SIPB-F-05) : Hapus Produk

Precondition: Admin berhasil *login* dan data produk sudah tersimpan dalam sistem.

1. Admin memilih menu *edit* pada *list* produk yang ingin dihapus.
2. Sistem menampilkan *form* untuk mengedit produk
3. Admin memilih fungsi hapus.
4. Sistem menampilkan *popup* untuk melakukan konfirmasi mengenai penghapusan produk dari sistem.
5. Admin melakukan konfirmasi hapus.
6. Sistem menghapus data produk yang dipilih dari sistem.
7. Sistem menampilkan halaman *home* admin.

Postcondition: Data produk tersebut dihapus dalam sistem.

6. System use case details lihat informasi produk (SIPB-F-06)

(SIPB-F-06): Lihat Informasi Produk

Precondition: Pelanggan berhasil melakukan *login*.

1. Pelanggan memilih gambar produk yang yang diinginkan
2. Sistem menampilkan informasi produk

Postcondition: Sistem dapat menampilkan kelengkapan produk seperti harga, merk dan juga gambar produk yang dipilih.

7. *System use case details* cari produk (SIPB-F-07)

(SIPB-F-07): Cari Produk

Precondition: Pelanggan berhasil melakukan *login*.

1. Pelanggan mengisi nama barang dan memilih fungsi cari.
2. Sistem menampilkan daftar informasi produk sesuai dengan yang dimasukan pelanggan.

Postcondition: Pelanggan menemukan informasi produk sesuai dengan kata kunci yang dimasukan.

8. *System use case details* pesan barang (SIPB-F-08)

(SIPB-F-08): Pesan Barang

Precondition: Pelanggan berhasil melakukan *login*.

1. Pelanggan memilih produk mana yang ingin dipesan
2. Sistem menampilkan detail produk yang diinginkan
3. Pelanggan memilih fungsi tambah untuk menambahkan produk yang ingin dibeli.
4. Sistem menyimpan produk yang ingin dibeli, yang diset secara otomatis 1.
5. Pelanggan masuk ke halaman *purchase order*, sistem menampilkan produk apa saja yang telah disimpan dalam *purchase order* beserta jumlah barang yang dipesan. Pelanggan memasukan jumlah barang sesuai dengan kebutuhan pelanggan. Setelah itu pelanggan memilih fungsi simpan.
6. Sistem menyimpan barang yang telah dipesan.
7. Sistem menampilkan data pelanggan dan juga form alamat pengiriman barang
8. Pelanggan memasukan alamat pengiriman barang dan memilih fungsi simpan.
9. Sistem menyimpan data alamat pengiriman barang.
10. Sistem mengambil data pelanggan.
11. Sistem mengambil detail pesanan pelanggan.
12. Sistem membuat dokumen pemesanan yang berisi kelengkapan data pelanggan, alamat pengiriman dan detail barang yang dipesan.
13. Sistem menampilkan dokumen pemesanan.

14. Sistem membuat sebuah form yang digunakan untuk mengirim *email* yang mana akan dilakukan konfirmasi oleh bagian keuangan pelanggan.
15. Sistem mengirimkan *email* kepada bagian keuangan pelanggan.
16. Sistem menyimpan status PO pelanggan menjadi "menunggu konfirmasi" dan menyimpan status keuangan menjadi "invalid".

Postcondition: Sistem menyimpan pesanan barang dan kelengkapan pelanggan serta email form konfirmasi telah terkirim.

9. *System use case details* konfirmasi pesanan (SIPB-F-09)

(SIPB-F-09) : Konfirmasi Pesanan

Precondition : -

1. Bagian keuangan pelanggan memeriksa apakah pesanan yang dilakukan sudah sesuai dengan kebutuhan perusahaan, apabila dirasa sudah sesuai maka bagian keuangan pelanggan akan menekan fungsi konfirmasi
2. Sistem menampilkan halaman baru yang berisi detail pemesanan dan pilihan untuk melakukan konfirmasi pesanan.
3. Bagian keuangan pelanggan memilih fungsi konfirmasi.
4. Sistem mengubah status konfirmasi menjadi "valid". Apabila selama 2 minggu bagian keuangan pelanggan belum melakukan konfirmasi maka secara otomatis sistem akan menghapus PO yang telah dilakukan.

Postcondition : Pesanan berubah menjadi sebuah PO

10. *System use case details* lihat PO (SIPB-F-10)

(SIPB-F-10): Lihat PO

Precondition : Aktor berhasil melakukan *login*.

Melihat PO sebagai Pegawai Perusahaan

1. Pegawai Perusahaan memilih pesanan yang diinginkan untuk ditampilkan surat *purchase order* dengan memilih fungsi *view* pada kolom PO . Pada halaman ini sistem hanya menampilkan daftar PO dengan status konfirmasi "valid".
2. Sistem menampilkan popup surat PO ingin ditampilkan atau dicetak.
3. Pegawai perusahaan memilih lihat
4. Sistem membuat dokumen PO.
5. Sistem menampilkan surat PO.

Postcondition: Sistem dapat menampilkan surat PO.

11. *System use case details* cetak PO (SIPB-F-11)

(SIPB-F-11) : Cetak PO

Precondition : Pegawai Perusahaan berhasil melakukan *login*

1. Pegawai Perusahaan memilih pesanan yang diinginkan untuk ditampilkan surat PO dengan memilih fungsi *view* pada kolom PO . Pada halaman ini sistem hanya menampilkan daftar PO dengan status konfirmasi “valid”.
2. Sistem menampilkan popup surat PO ingin ditampilkan atau dicetak.
3. Pegawai perusahaan memilih cetak
4. Sistem membuat dokumen PO.
5. Sistem mencetak surat PO melalui *hardware* yang terhubung dengan *device*.

Postcondition: Sistem mencetak surat PO dengan *hardware* cetak yang terhubung dengan *device*.

12. *System use case details* lihat rekap penjualan (SIPB-12)

(SIPB-F-12): Lihat Rekap Penjualan

Precondition : Admin berhasil melakukan *login*.

1. Admin menginputkan batasan tanggal yang diinginkan untuk ditampilkan dan memilih fungsi cari.
2. Sistem menampilkan daftar PO sesuai dengan batasan tanggal yang diinginkan. Pada halaman ini sistem hanya menampilkan daftar PO dengan status konfirmasi “valid”.
3. Admin memilih fungsi lihat pada halaman pesanan
4. Sistem membuat dokumen rekap PO selama bulan yang diinputkan.
5. Sistem akan menampilkan hasil rekap penjualan berupa sebuah dokumen.

Postcondition : Sistem dapat menampilkan rekap penjualan

13. *System use case details* cetak rekap penjualan (SIPB-F-13)

(SIPB-F-13) : Cetak Rekap Penjualan

Precondition : Admin berhasil melakukan login

1. Admin menginputkan bulan yang diinginkan untuk ditampilkan dan memilih fungsi cari.
2. Sistem menampilkan daftar PO sesuai dengan batasan tanggal yang diinginkan. Pada halaman ini sistem hanya menampilkan daftar PO dengan status konfirmasi “valid”.
3. Admin memilih fungsi cetak pada halaman pesanan
4. Sistem membuat dokumen rekap PO selama bulan yang diinputkan.
5. Sistem akan mencetak hasil rekap penjualan berupa sebuah dokumen.

Postcondition : Sistem mencetak rekap penjualan dengan *hardware* cetak yang terhubung dengan *device*.

14. *System use case details* ubah status PO (SIPB-F-14)

(SIPB-F-14) : Ubah Status PO

Precondition: Pegawai Perusahaan berhasil melakukan login

1. Pegawai perusahaan memilih fungsi “ubah status” pada daftar PO yang memiliki status konfirmasi “valid”.
2. Sistem menampilkan pilihan status PO yaitu “Packing barang”, “Sedang Dikirim”, “Barang Diterima”, “Belum Lunas” dan “Lunas”. Lalu pegawai Perusahaan memilih status PO.
3. Sistem merubah status PO barang sesuai dengan pilihan petugas perusahaan.

Postcondition: Status pesanan berubah

15. *System use case details* lihat tagihan (SIPB-F-15)

(SIPB-F-15) : Lihat Tagihan

Precondition : Pegawai Perusahaan berhasil melakukan *login*

1. Pegawai Perusahaan memilih pesanan yang diinginkan untuk ditampilkan surat tagihan dengan memilih fungsi *view*. Sistem hanya menampilkan PO yang memiliki status konfirmasi “valid”.
2. Sistem menampilkan popup surat PO ingin ditampilkan atau dicetak.
3. Pegawai perusahaan memilih lihat
4. Sistem membuat dokumen PO.
5. Sistem menampilkan surat PO.

Postcondition: Sistem dapat menampilkan surat tagihan.

16. *System use case details* cetak tagihan (SIPB-F-16)

(SIPB-F-16) : Cetak Tagihan

Precondition : Pegawai Perusahaan berhasil melakukan *login*

1. Pegawai Perusahaan memilih pesanan yang diinginkan untuk ditampilkan surat *purchase order* dengan memilih fungsi *view* pada kolom PO . Pada halaman ini sistem hanya menampilkan daftar PO dengan status konfirmasi “valid”.
2. Sistem menampilkan popup surat PO ingin ditampilkan atau dicetak.
3. Pegawai perusahaan memilih cetak
4. Sistem membuat dokumen PO.
5. Sistem mencetak surat PO berupa sebuah dokumen.

Postcondition: Sistem mencetak surat PO dengan *hardware* cetak yang terhubung dengan *device*.

4.3.10 **Supplementary Requirements**

Supplementary requirements merupakan suatu *deskripsi* untuk menjelaskan kebutuhan-kebutuhan non-fungsional yang tidak terdapat dalam

system use case list. Tabel berikut berisi tentang *supplementary requirements* dari sistem informasi pemesanan barang :

Tabel 0.4 System Supplementary Requirements

No.	Nama Kebutuhan	Deskripsi Kebutuhan
1.	<i>Ergonomy</i> (SIPB-NF-01)	Desain antarmuka sistem harus <i>user friendly</i> dan jelas alurnya.

Kesimpulan dari fase ini terdapat 10 tahap yaitu tahap *business actor list*, *business use case list*, *business activity diagram*, *business use case details*, *user interface sketches*, *system actor list*, *system use case list*, *system use case diagram*, *system use case details* dan *supplementary requirement*.

Pada sub-fase *business* tahap *business actor list* dilakukan analisis aktor siapa saja yang turut berperan dalam jalannya proses bisnis pemesanan barang pada PT. Trivia Nusantara. Dari tahap ini dihasilkan 5 aktor yaitu pelanggan, bagian keuangan pelanggan, petugas, bagian gudang dan kurir. Pada sub fase yang sama tahap *business use case list* akan menyebutkan proses bisnis apa saja yang ada dalam proses pemesanan barang, dalam tahap ini didapatkan empat proses bisnis yaitu pelanggan memesan barang, melakukan konfirmasi pemesanan ke bagian keuangan pelanggan, melakukan *packing barang* dan melakukan penagihan. Setelah mendapatkan *business use case list* tahapan selanjutnya adalah memodelkan proses bisnis tersebut kedalam sebuah *activity diagram*. Dari *business activity diagram* tersebut akan didapatkan proses apa saja yang ada untuk menjalankan proses bisnis tersebut. Hal itu akan dijelaskan pada tahap selanjutnya yaitu *business use case details*.

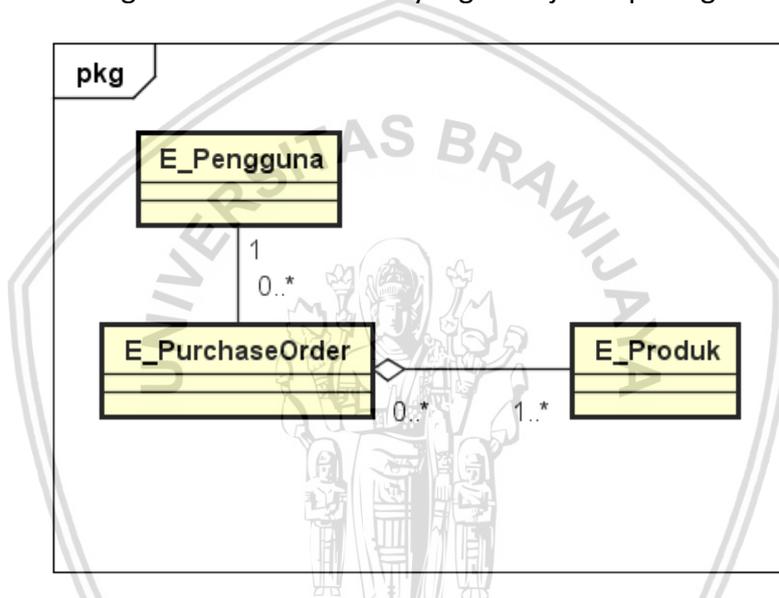
Setelah itu pada sub-fase *system* diawali dengan tahap melakukan *brainstorm user interface sketches* yang mana memberikan gambaran tampilan UI sistem yang akan dibangun. Setelah itu melakukan analisis terhadap sistem diawali dengan menganalisis aktor yang terlibat dalam sistem. Aktor yang sudah teridentifikasi dituliskan dalam tahap *system actor list*. Pada tahap selanjutnya yaitu *system use case list* kegiatan yang dilakukan adalah melakukan analisis interaksi yang dapat dilakukan antara aktor dengan sistem, masukan dari *use case list* didapatkan dari keluaran tahap *project genesis* dan analisis proses bisnis. Pada tahap ini dihasilkan 16 *use case*. Setelah mendapatkan *use case*. Pada tahap *use case diagram*, *use case* yang sudah didefinisikan sebelumnya digambarkan perannya dalam sistem dan aktor. Setelah itu akan dilakukan penggambaran secara deskripsi bagaimana aktor dan sistem berinteraksi untuk melakukan suatu *use case* hal itu di tuliskan pada tahap *system use case details*. Tahap yang sebelumnya menggambarkan kebutuhan fungsional sistem yang diinginkan namun terdapat kebutuhan fungsional dalam sistem yang dituliskan pada tahap *supplementary requirement*. Terdapat satu *supplementary requirement* yang dihasilkan yaitu *ergonomy*.

4.4 Fase Analysis

Fase *analysis* bertujuan untuk membuat gambaran perancangan awal yang disesuaikan dengan kebutuhan yang telah didefinisikan. Fase ini terdiri dari membuat *analysis class diagram*, membuat *analysis attributes list*, realisasi *use case* dengan menggunakan *communication diagram* dan membuat *operation list*.

4.4.1 Analysis Class Diagram

Class diagram adalah suatu diagram yang menggambarkan struktur objek statis dalam sebuah sistem, menunjukkan suatu sistem tersusun dari kelas-kelas apa saja serta hubungan apa yang terbentuk diantara kelas tersebut. Pada fase *analysis*, pembuat *class diagram* hanya mencantumkan nama kelas dan objek beserta hubungannya. Berikut ini adalah *analysis class diagram* dari proses pemesanan barang PT. Trivia Nusantara yang ditunjukkan pada gambar dibawah :



Gambar 0.31 Analysis Class Diagram

4.4.2 Analysis Attributes List

Attributes adalah properti dari sebuah objek, setiap *attribute* dalam UML dapat diberikan sebuah tipe data, baik itu tipe data dari sebuah kelas maupun tipe data primitif. Jika seseorang analis memilih untuk menentukan tipe data, maka penulisannya adalah disebelah kanan nama *attribute*, setelah titik dua (:). *Analysis attribute list* didapatkan dari hasil masukan yang dilakukan oleh aktor sistem. Berikut ini adalah *analysis attributes list* beserta tipe data dan entitas dari proses pemesanan barang PT. Trivia Nusantara :

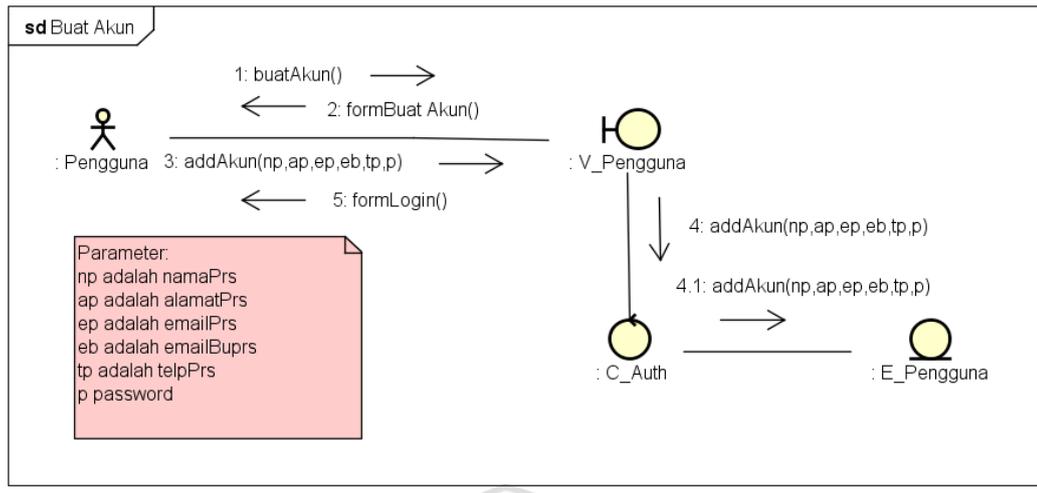
Tabel 0.5 Analysis Attributes List.

No.	Nama Attribut	Class	Deskripsi
1	namaPrs : Varchar	E_Pengguna	Nama perusahaan pelanggan

No.	Nama Attribut	Class	Deskripsi
2	alamatPrs: Varchar	E_Pengguna	Alamat perusahaan pelanggan
3	emailPrs: Varchar	E_Pengguna	Email perusahaan secara umum
4	emailBuprs : Varchar	E_Pengguna	Email bagian keuangan pelanggan
5	telpPrs : Varchar	E_Pengguna	Telepon perusahaan pelanggan
6	password: Varchar	E_Pengguna	Password yang digunakan pelanggan untuk melakukan <i>login</i> .
7	namaProduk: Varchar	E_Produk	Nama produk .
8	merkProduk: Varchar	E_Produk	Merk produk
9	hargaProduk:int	E_Produk	Harga Produk
10	deskripsiProduk:LongText	E_Produk	Penjelasan singkat mengenai produk seperti kegunaan dan spesifikasi produk
11	gambarProduk:BLOB	E_Produk	Gambar produk
12	jumlah:int	E_PurchaseOrder	Menyimpan jumlah barang yang dipesan
13	alamat:Varchar(100)	E_PurchaseOrder	Alamat pengiriman barang
14	status: Varchar	E_PurchaseOrder	Status barang yang dipesan
15	tanggal :Date	E_PurchaseOrder	Tanggal pemesanan barang dilakukan

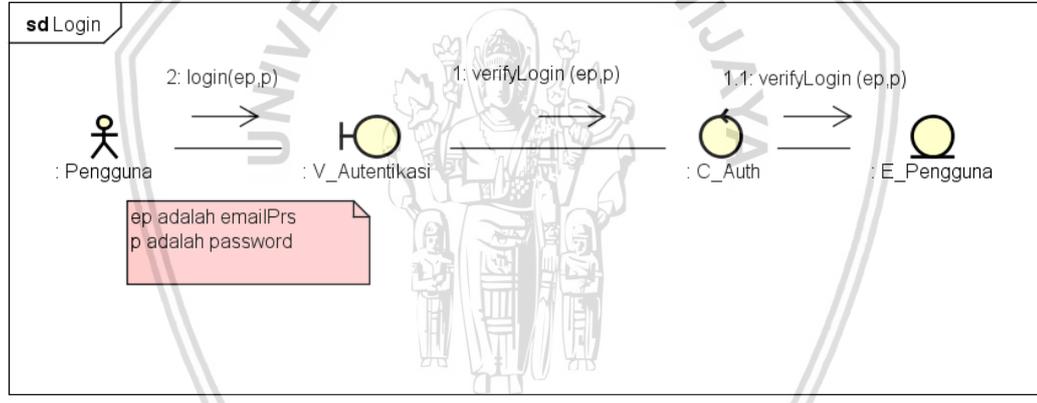
4.4.3 Communication Diagram

Communication Diagram adalah suatu diagram UML yang menggambarkan hubungan antar kelas dengan kelas yang lain serta pesan yang disampaikan dalam suatu sistem sesuai dengan proses yang ada dalam *use case*. *Communication diagram* didapatkan dari interaksi sistem dengan aktor yang sebelumnya telah dijelaskan di *use case details*. Berikut ini adalah *communication diagram* dari proses pemesanan barang PT. Trivia Nusantara :



Gambar 0.32 Communication Diagram untuk Use Case Buat Akun : (SIPB-F-01)

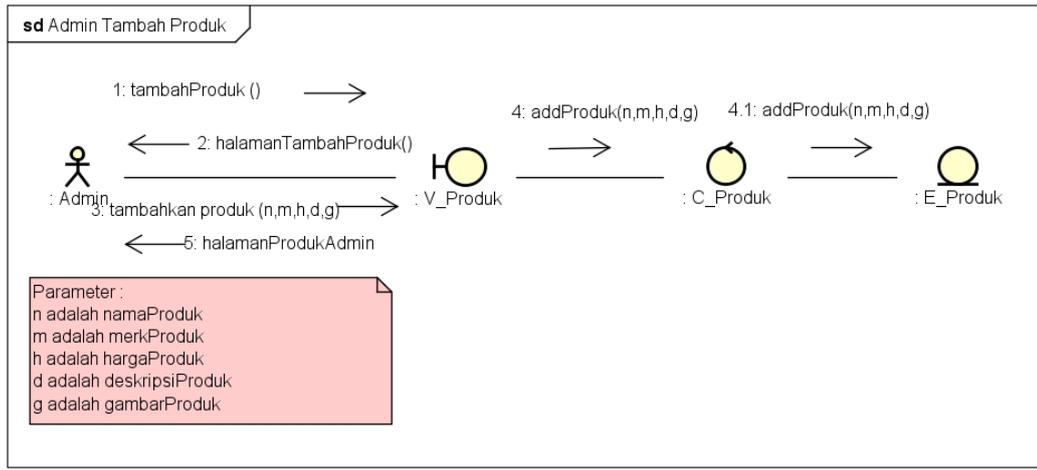
Communication Diagram untuk Use Case Buat Akun : (SIPB-F-01) dapat diketahui pada gambar 4.32 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary V_Pengguna*, *controller C_Auh* dan *entity E_pengguna*.



Gambar 0.33 Communication Diagram untuk Use Case Login : (SIPB-F-02)

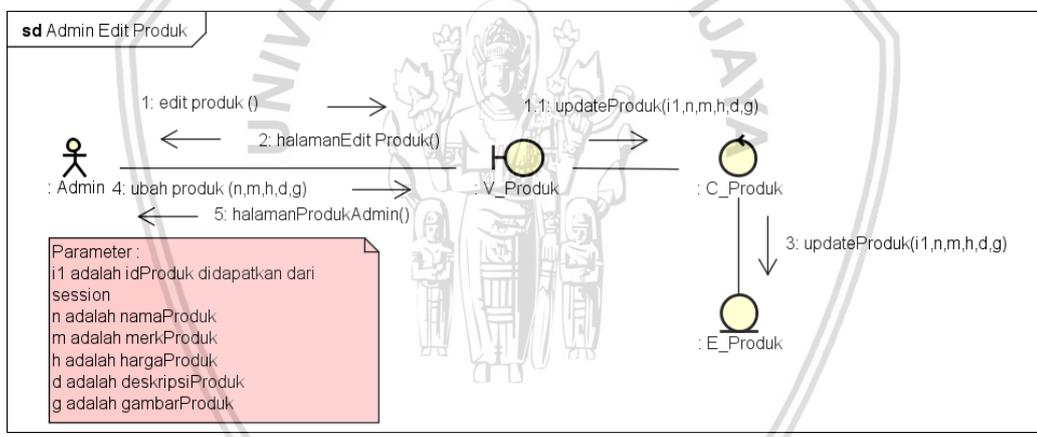
Communication Diagram untuk Use Case Login : (SIPB-F-02) dapat diketahui pada gambar 4.33 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary V_Autentikasi*, *controller C_Auth* dan *entity E_Pengguna*.





Gambar 0.34 Communication Diagram untuk Use Case Tambah Produk : (SIPB-F-03)

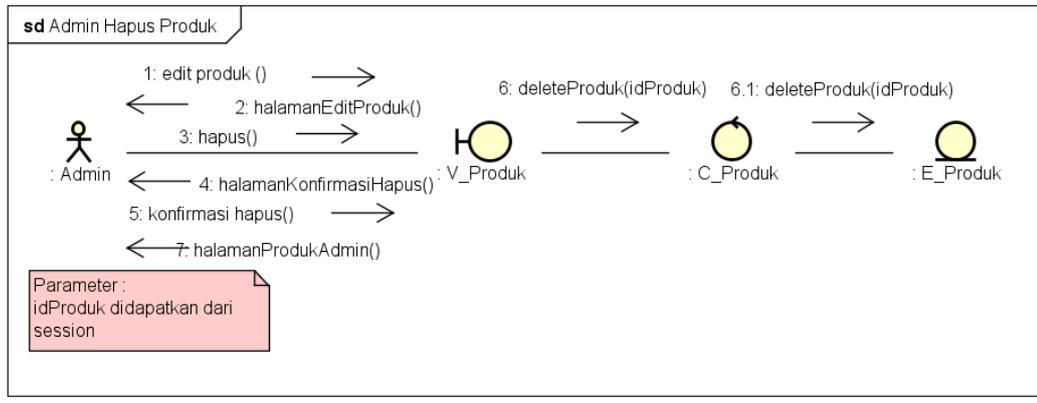
Communication Diagram untuk Use Case Tambah Produk : (SIPB-F-03) dapat diketahui pada gambar 4.34 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_Produk, *controller* C_Produk dan *entity* E_Produk.



Gambar 0.35 Communication Diagram untuk Use Case Edit Produk : (SIPB-F-04)

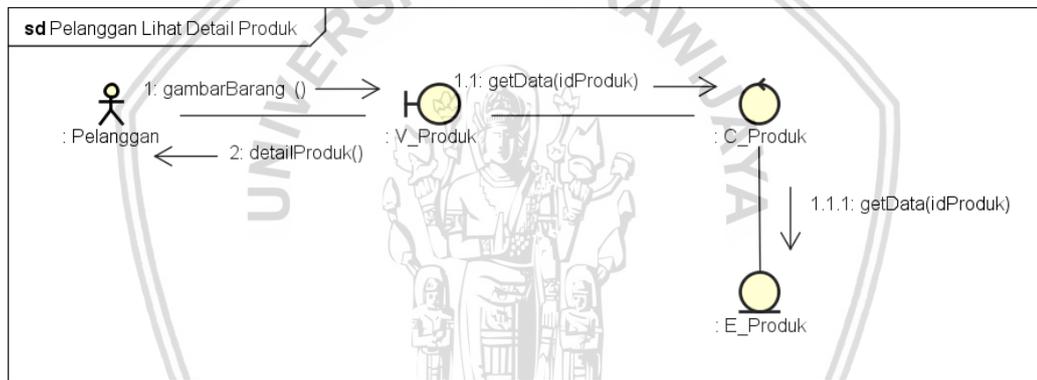
Communication Diagram untuk Use Case Edit Produk : (SIPB-F-04) dapat diketahui pada gambar 4.35 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_Produk, *controller* C_Produk dan *entity* E_Produk.





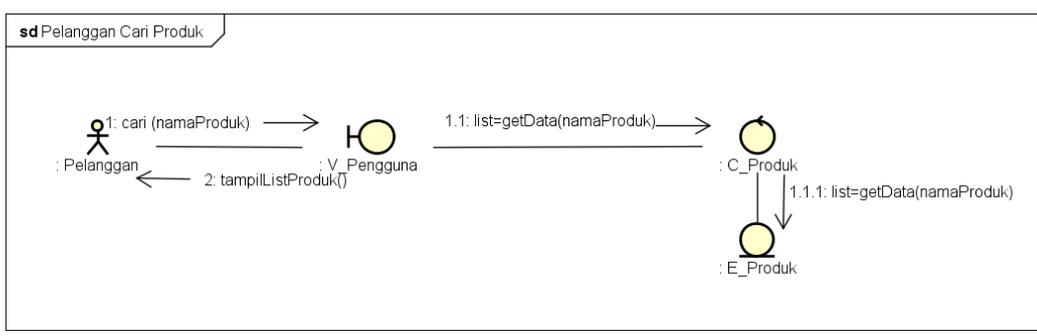
Gambar 0.36 Communication Diagram untuk Use Case Hapus Produk : (SIPB-F-05)

Communication Diagram untuk Use Case Hapus Produk : (SIPB-F-05) dapat diketahui pada gambar 4.36 bahwa kelas dalam sistem yang berinteraksi yaitu boundary V_Produk, controller C_Produk dan entity E_Produk.



Gambar 0.37 Communication Diagram untuk Use Case Lihat informasi Produk : (SIPB-F-06)

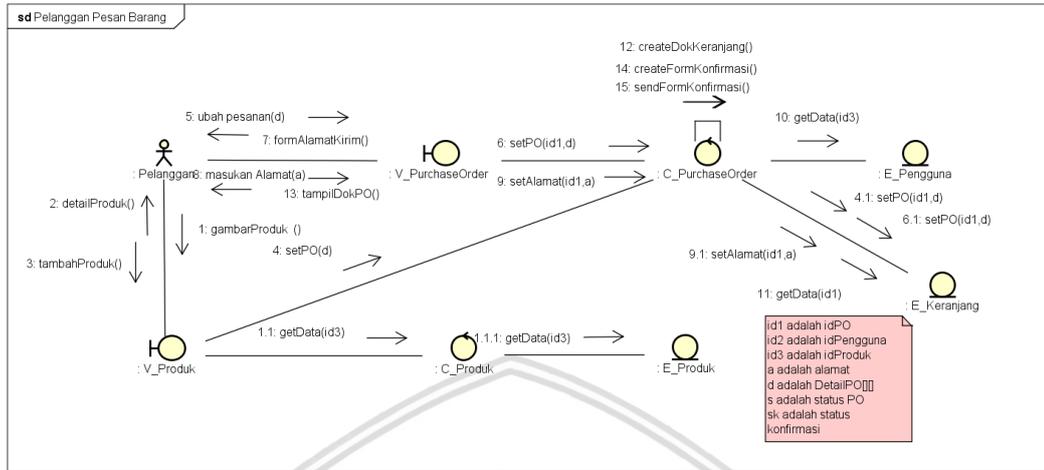
Communication Diagram untuk Use Case Lihat informasi Produk : (SIPB-F-06) dapat diketahui pada gambar 4.37 bahwa kelas dalam sistem yang berinteraksi yaitu kelas boundary V_Produk, controller C_Produk dan entity E_Produk.



Gambar 0.38 Communication Diagram untuk Use Case Cari Produk : (SIPB-F-07)

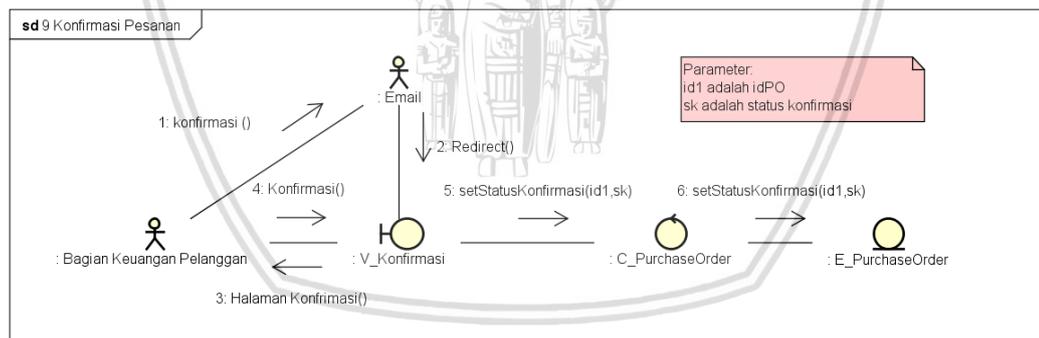


Communication Diagram untuk Use Case Cari Produk : (SIPB-F-03-02) dapat diketahui pada gambar 4.38 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_Produk, *controller* C_Produk dan *entity* E_Produk.



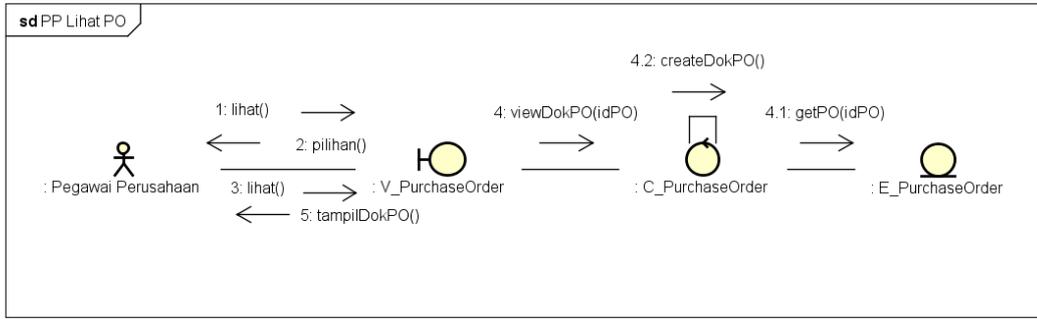
Gambar 0.39 Communication Diagram untuk Use Case Pesan Barang : (SIPB-F-08)

Communication Diagram untuk Use Case Pesan Barang : (SIPB-F-08) dapat diketahui pada gambar 4.39 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *boundary* V_Produk, *controller* C_PurchaseOrder,, *controller* C_Produk , *entity* E_Produk, *entity* E_Pengguna dan *entity* E_PurchaseOrder.



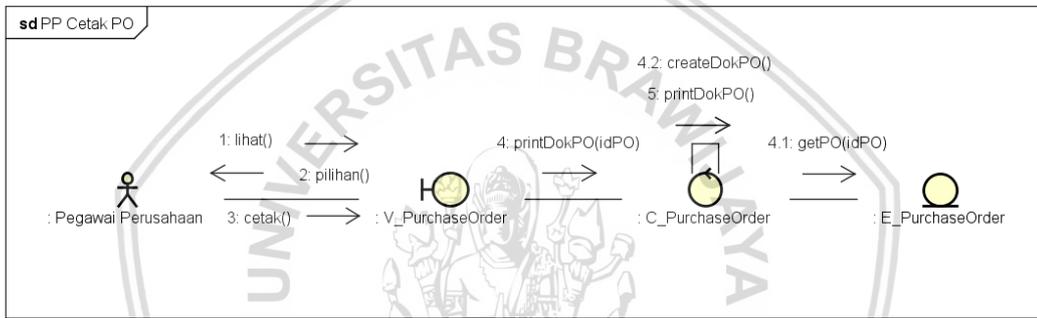
Gambar 0.40 Communication Diagram untuk Use Case Konfirmasi Pesanan : (SIPB-F-09)

Communication Diagram untuk Use Case Konfirmasi Pesanan : (SIPB-F-09) dapat diketahui pada gambar 4.40 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.



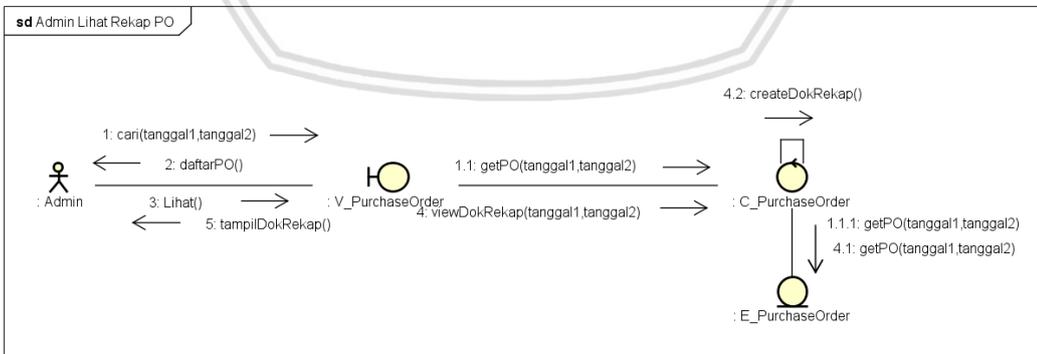
Gambar 0.41 Communication Diagram untuk Use Case Lihat PO : (SIPB-F-010)

Communication Diagram untuk Use Case Lihat PO : (SIPB-F-10) pada gambar 4.41 dapat diketahui bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder .



Gambar 0.42 Communication Diagram untuk Use Case Cetak PO : (SIPB-F-11)

Communication Diagram untuk Use Case Cetak PO : (SIPB-F-11) pada gambar 4.42 dapat diketahui bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.

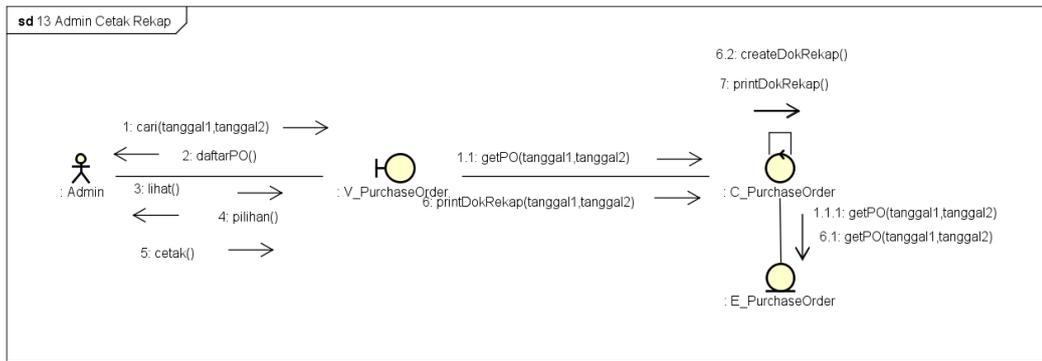


Gambar 0.43 Communication Diagram untuk Use Case Lihat Rekap Penjualan : (SIPB-F-12)

Communication Diagram untuk Use Case Lihat Rekap Penjualan : (SIPB-F-12) dapat diketahui pada gambar 4.43 bahwa kelas dalam sistem yang berinteraksi

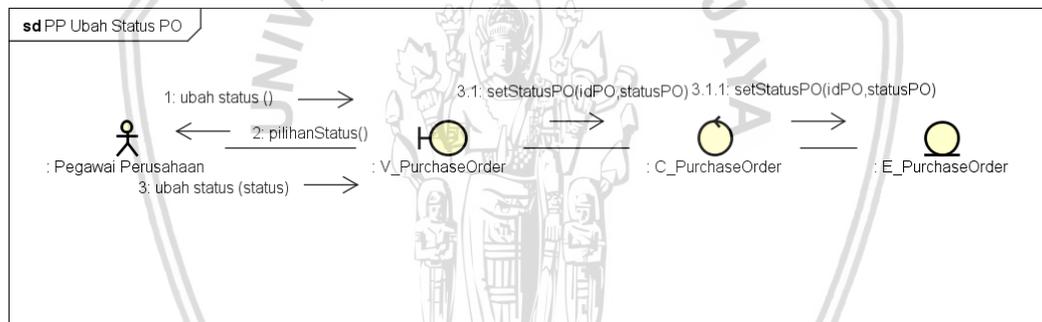


yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.



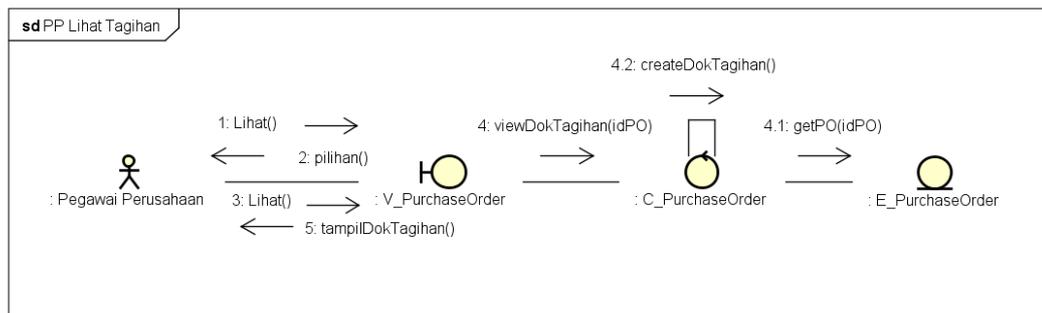
Gambar 0.44 Communication Diagram untuk Use Case Cetak Rekap Penjualan : (SIPB-F-13)

Communication Diagram untuk Use Case Cetak Rekap Penjualan : (SIPB-F-13) dapat diketahui pada gambar 4.44 bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.



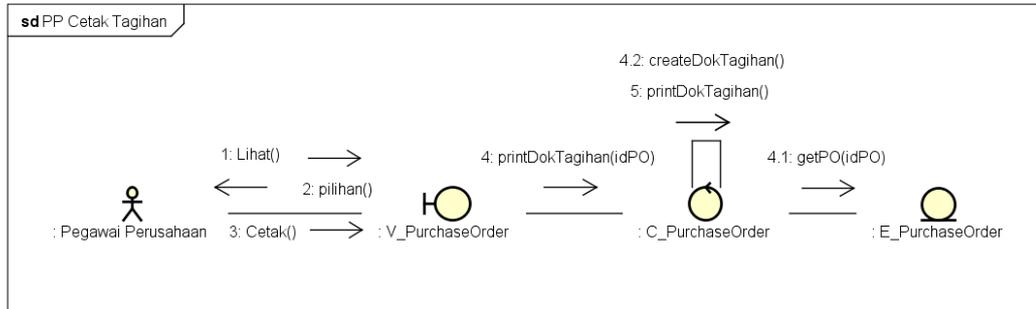
Gambar 0.45 Communication Diagram untuk Use Case Status PO : (SIPB-F-14)

Communication Diagram untuk Use Case Status PO : (SIPB-F-14) pada gambar 4.45 dapat diketahui bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.



Gambar 0.46 Communication Diagram untuk Use Case Lihat Tagihan : (SIPB-F-15)

Communication Diagram untuk *Use Case* Lihat Tagihan : (SIPB-F-15) pada gambar 4.46 dapat diketahui bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.



Gambar 0.47 *Communication Diagram* untuk *Use Case* Cetak Tagihan : (SIPB-F-07-02)

Communication Diagram untuk *Use Case* Lihat Tagihan : (SIPB-F-16) pada gambar 4.47 dapat diketahui bahwa kelas dalam sistem yang berinteraksi yaitu kelas *boundary* V_PurchaseOrder, *controller* C_PurchaseOrder dan *entity* E_PurchaseOrder.

4.4.4 *Operation List*

Setiap pesan yang terdapat pada *communication diagram* memiliki kesesuaian dengan *operation* pada *class diagram*, sehingga seorang analis harus mendokumentasikan seluruh *operation* yang digunakan dalam rangka untuk memiliki satu set lengkap realisasi *use case*. *Operation list* terdapat pada tabel 4.7.

Tabel 0.6 Operation List

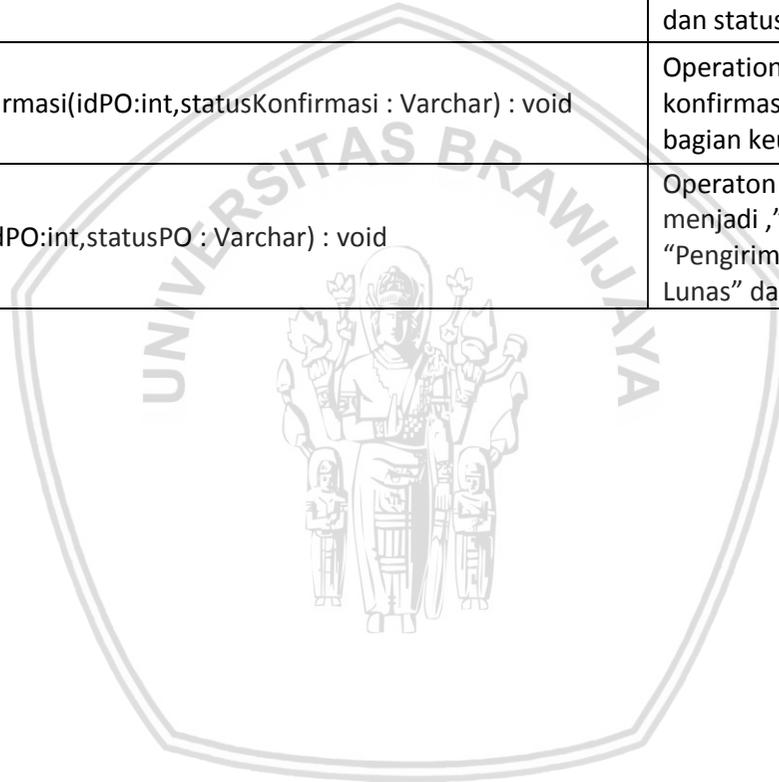
No.	Class	Operation	Deskripsi
1	C_Auth	addAkun(namaPrs : String, alamatPrs: String, emailPrs: String, emailBuprs : String, telpPrs :String, password:String):void	Operation yang digunakan untuk membuat akun baru input yang digunakan berisi data kelengkapan pengguna
2	C_Auth	verifyLogin(emailPrs:String,pasword:String): void	Operation yang digunakan untuk melakukan verifikasi email dan password yang dimasukan.
3	C_Produk	addProduk(idGambar :int,gambarProduk:BLOB,namaProduk:String, merkProduk:String,hargaProduk:int,deskripsiProduk:LongText):void	Operation yang digunakan untuk menambahkan produk baru inputan yang digunakan merupakan kelengkapan data produk.
4	C_Produk	deleteProduk(idProduk:int):void	Operation yang digunakan untuk menghapus produk dengan suatu idProduk tertentu.
5	C_Produk	getData(idProduk:int) :E_Produk	Operation yang digunakan untuk mendapatkan data produk sesuai dengan idProduk
6	C_Produk	getData(namaProduk:String):E_Produk	Operation yang digunakan untuk mengambil data produk dengan suatu nama produk tertentu
7	C_Produk	updateProduk(idProduk:int, gambarProduk:BLOB, namaProduk:String, merkProduk:String, hargaProduk:int,deskripsiProduk:LongText):void	Operation yang digunakan untuk melakukan update data produk.
8	C_PurchaseOrder	createDokPO():void	Operation yang digunakan untuk membuat sebuah dokumen berisi kelengkapan Purchase Order.
9	C_PurchaseOrder	createDokRekapPO(tanggal1:Date,tanggal 2:Date):void	Operation yang digunakan untuk membuat dokumen yang berisi rekap Purchase Order dalam suatu bulan tertentu

No.	Class	Operation	Deskripsi
10	C_PurchaseOrder	createDokTagihan(idTagihan:int):void	Operation yang digunakan untuk membuat sebuah dokumen berisi kelengkapan Tagihan.
11	C_PurchaseOrder	createFormKonfirmasi():void	Operation yang digunakan untuk membuat form konfirmasi yang mana nantinya form tersebut akan dikirimkan ke email Bagian Keuangan Pelanggan.
12	C_PurchaseOrder	getPOtanggal1:Date, tanggal 2:Date):E_PurchaseOrder	Operation yang digunakan untuk mendapatkan sebuah daftar data PO pada suatu bulan tertentu
13	C_PurchaseOrder	printDokPO(idPO:int):void	Operation yang digunakan untuk mencetak dokumen PO pada id po tertentu
14	C_PurchaseOrder	printDokTagihan(idPO:int):void	Operation yang digunakan untuk mencetak dokumen Tagihan pada id tagihan tertentu
15	C_PurchaseOrder	printRekapPO(tanggal1:Date,tanggal2:Date):void	Operation yang digunakan untuk mencetak dokumen rekap purchase order pada bulan tertentu
16	C_PurchaseOrder	printDokPO():void	Operation yang digunakan untuk mencetak dokumen PO
17	C_PurchaseOrder	printDokTagihan():void	Operation yang digunakan untuk mencetak dokumen Tagihan
18	C_PurchaseOrder	printDokRekap():void	Operation yang digunakan untuk mencetak dokumen rekap purchase order.
19	C_PurchaseOrder	sendFormKonfirmasi():void	Operation yang digunakan untuk mengirimkan email kepada email bagian keuangan pelanggan form untuk melakukan konfirmasi pemesanan.
20	C_PurchaseOrder	setAlamat(idPO:int,alamat:Varchar(100)):void	Operation yang digunakan untuk menyimpan alamat dengan idPO tertentu

No.	Class	Operation	Deskripsi
21	C_PurchaseOrder	setPO(DetailPO[][]:[E_Produk][int]):void	Operation yang digunakan untuk memasukan data PO berupa nama barang dan jumlah.
22	C_PurchaseOrder	setPO(idPO:int, DetailPO[][]:[E_Produk],[int]):void	Operation yang digunakan untuk merubah barang dan jumlah barang yang dipesan dengan idPo tertentu
23	C_PurchaseOrder	setStatusKonfirmasi(idPO:int,statusKonfirmasi : Varchar) : void	Operation yang digunakan untuk merubah status konfirmasi apabila pesanan telah dikonfirmasi oleh bagian keuangan pelanggan
24	C_PurchaseOrder	setStatusPO(idPO:int,statusPO : Varchar) : void	Operaton yang digunakan untuk mengubah status PO menjadi ,”Menunggu Konfirmasi”, ”Packing barang”, ”Pengiriman Barang”, ”Barang Diterima” ,”Belum Lunas” dan ”Lunas”.
25	C_PurchaseOrder	viewDokPO(idPO:int):DokumenPO	Operation yang digunakan untuk menampilkan dokumen purchase order.
26	C_PurchaseOrder	viewDokTagihan(idPO:int):void	Operation yang digunakan untuk menampilkan dokumen tagihan.
27	C_PurchaseOrder	viewRekapPO(bulan:Date):void	Operation yang digunakan untuk menampilkan dokumen rekap purchase order pada bulan tertentu sesuai yang diinginkan oleh admin.
28	E_Pengguna	addAkun(namaPrs :String,alamatPrs:String,emailPrs:String, emailBuPrs:String,telpPrs:String,password:String):void	Operation yang digunakan untuk membuat akun baru input yang digunakan berisi data kelengkapan pengguna

No.	Class	Operation	Deskripsi
29	E_Pengguna	getData(idPengguna:int):E_Pengguna	Operation yang digunakan untuk mendapatkan data pelanggan sesuai dengan idPelanggan.
30	E_Pengguna	verifyLogin(emailPrs:String,password:String):void	Operation yang digunakan untuk melakukan verifikasi email dan password yang dimasukan.
31	E_Produk	addProduk(idGambar :int,gambarProduk:BLOB,namaProduk:String, merkProduk:String,hargaProduk:int,deskripsiProduk:LongText):void	Operation yang digunakan untuk menambahkan produk baru inputan yang digunakan merupakan kelengkapan data produk.
32	E_Produk	deleteProduk(idProduk:int):void	Operation yang digunakan untuk menghapus produk dengan suatu idProduk tertentu.
33	E_Produk	getData(namaProduk:String):E_Produk	Operation yang digunakan untuk mendapatkan data produk sesuai dengan nama produk.
34	E_Produk	getData(idProduk:int) :E_Produk	Operation yang digunakan untuk mendapatkan data produk sesuai dengan idProduk
35	E_Produk	updateProduk(idProduk:int, idGambar :int,gambarProduk:BLOB,namaProduk:String, merkProduk:String,hargaProduk:int,deskripsiProduk:LongText):void	Operation yang digunakan untuk melakukan update data produk.
36	E_PurchaseOrder	getPO(tanggal1 : Date,tanggal2:Date):E_PurchaseOrder	Operation yang digunakan untuk mendapatkan sebuah daftar data PO pada suatu bulan tertentu
37	E_PurchaseOrder	setAlamat(idPO:int,alamat:Varchar(100))	Operation yang digunakan untuk menyimpan alamat pengiriman

No.	Class	Operation	Deskripsi
38	E_PurchaseOrder	setPO(DetailPO[[[]]:[E_Produk][int]):void	Operation yang digunakan untuk menyimpan barang yang dipesan beserta jumlahnya .
39	E_PurchaseOrder	setStatus(idPO : int, statusPO : Varchar, statusKonfirmasi : Varchar) : void	Operation yang digunakan untuk merubah status PO dan status konfirmasi
40	E_PurchaseOrder	setStatusKonfirmasi(idPO:int,statusKonfirmasi : Varchar) : void	Operation yang digunakan untuk merubah status konfirmasi apabila pesanan telah dikonfirmasi oleh bagian keuangan pelanggan
41	E_PurchaseOrder	setStatusPO(idPO:int,statusPO : Varchar) : void	Operaton yang digunakan untuk mengubah status PO menjadi ,”Menunggu Konfirmasi”, ”Packing barang”, ”Pengiriman Barang”, ”Barang Diterima” ,”Belum Lunas” dan ”Lunas”.



Kesimpulan dari fase *analysis* terdapat 4 tahap yaitu *analysis class diagram*, *analysis attribute list*, *communication diagram* dan *operation list*. Pada tahap pertama yaitu *analysis diagram* kegiatan yang dilakukan adalah melakukan analisis objek dan merancang *class diagram*. Pada tahap selanjutnya yaitu *analysis attribute list* kegiatan yang dilakukan menganalisis *attribute* apa saja yang ada dalam suatu *class* masukan dari tahap ini bersumber pada *system use case details*. Setelah mendapatkan atribut yang ada maka menggambarkan *communication diagram* yang digunakan untuk mengetahui hubungan setiap kelas yang ada. Menggambar *communication diagram* harus berdasar kepada urutan sudah dituliskan pada tahap *system use case details*. Setelah menggambarkan *communication diagram*, akan didapatkan fungsi fungsi yang digunakan lalu fungsi tersebut dituliskan di tahap selanjutnya yaitu tahap *operation list*. Dari hasil penggambaran *communication diagram* didapatkan 41 *operation*.

4.5 Fase Design

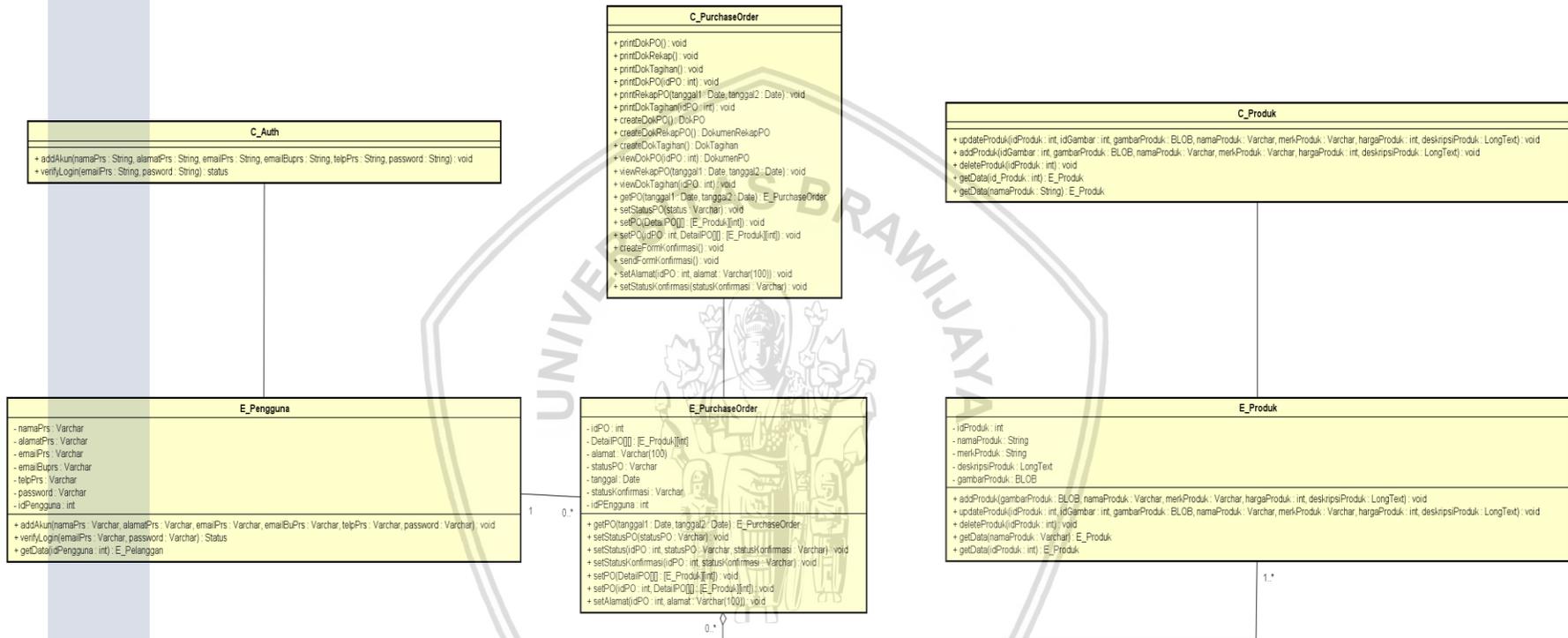
Fase *design* bertujuan untuk membuat perancangan sesuai dengan kebutuhan yang telah didefinisikan. Fase ini menghasilkan keluaran berupa *class diagram*, *sequence diagram* dan penyelesaian desain antar muka.

4.5.1 Class Diagram

Class diagram menggambarkan struktur objek statis dalam sebuah sistem, menunjukkan suatu sistem tersusun dari kelas-kelas apa dan hubungan apa yang terbentuk diantara kelas tersebut. Pada fase *design*, pembuatan *class diagram* mencantumkan nama kelas, atribut, *function*, beserta hubungannya. Masukan dari *class diagram* didapatkan dari hasil operasi dan atribut yang sebelumnya telah digunakan pada *communication diagram*.

Terlihat dari diagram tersebut memiliki beberapa hubungan agregasi (*whole-part*) yaitu pada kelas *E_purchaseOrder* dan kelas *E_Produk*. Pada kelas entitas lainnya dapat ditemukan hubungan asosiasi yaitu pada kelas *E_PurchaseOrder* dengan kelas *E_pengguna*, *E_Pengguna* dengan *C_Auth*, *E_PurchaseOrder* dengan *C_PurchaseOrder*, *E_Produk* dengan *C_Produk*. Berikut ini adalah *class diagram* dari proses pemesanan barang PT. Trivia Nusantara yang akan ditunjukkan pada Gambar 4.48.

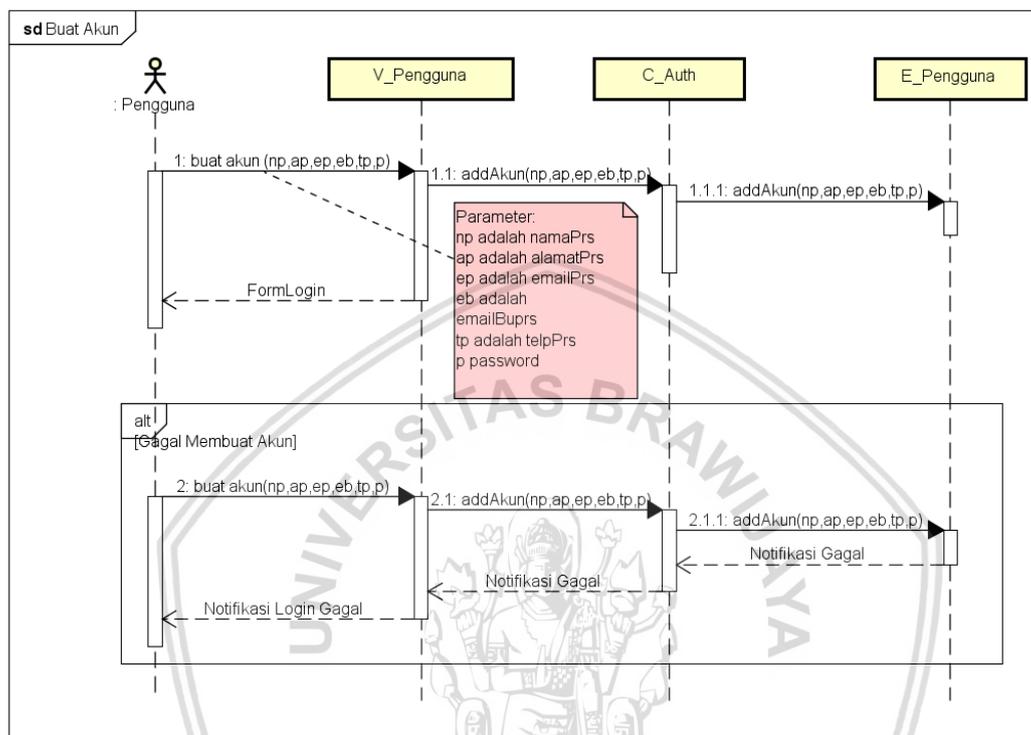
pkg



Gambar 0.48 Class Diagram

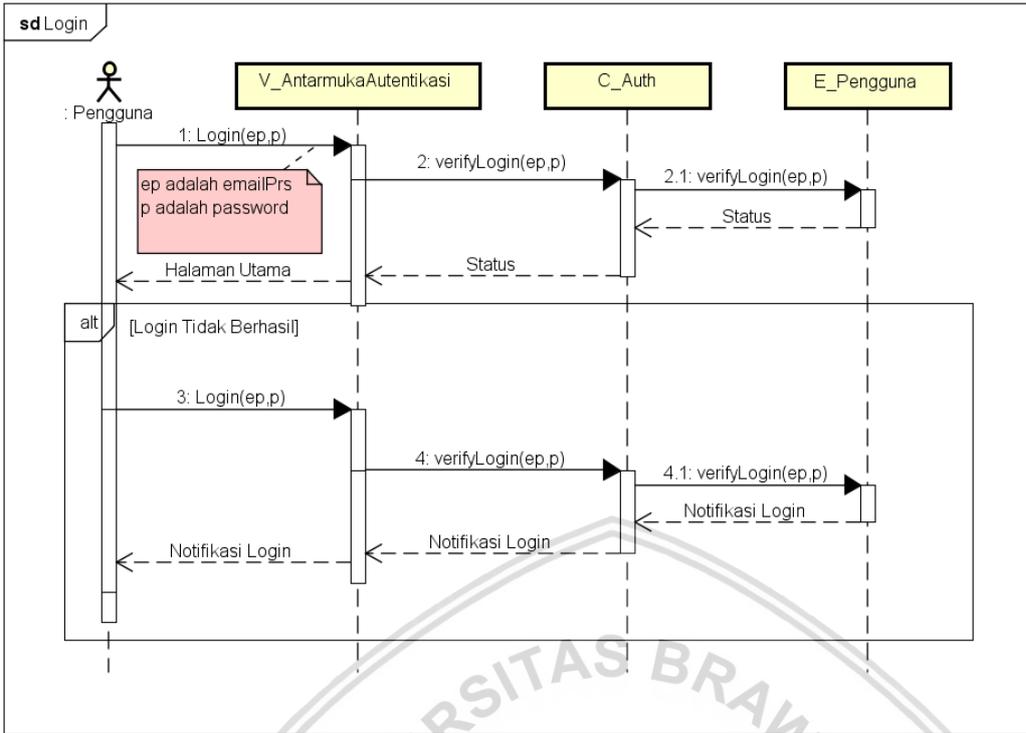
4.5.2 Sequence Diagram

Sequence diagram menggambarkan bagaimana sebuah objek berinteraksi dengan satu sama lain melalui pesan sekuensi sebuah use case atau operasi. Berbeda dengan communication diagram, sequence diagram lebih menampilkan urutan setiap proses yang ada. Sequence diagram mendapatkan masukan dari use case details dan communication diagram.



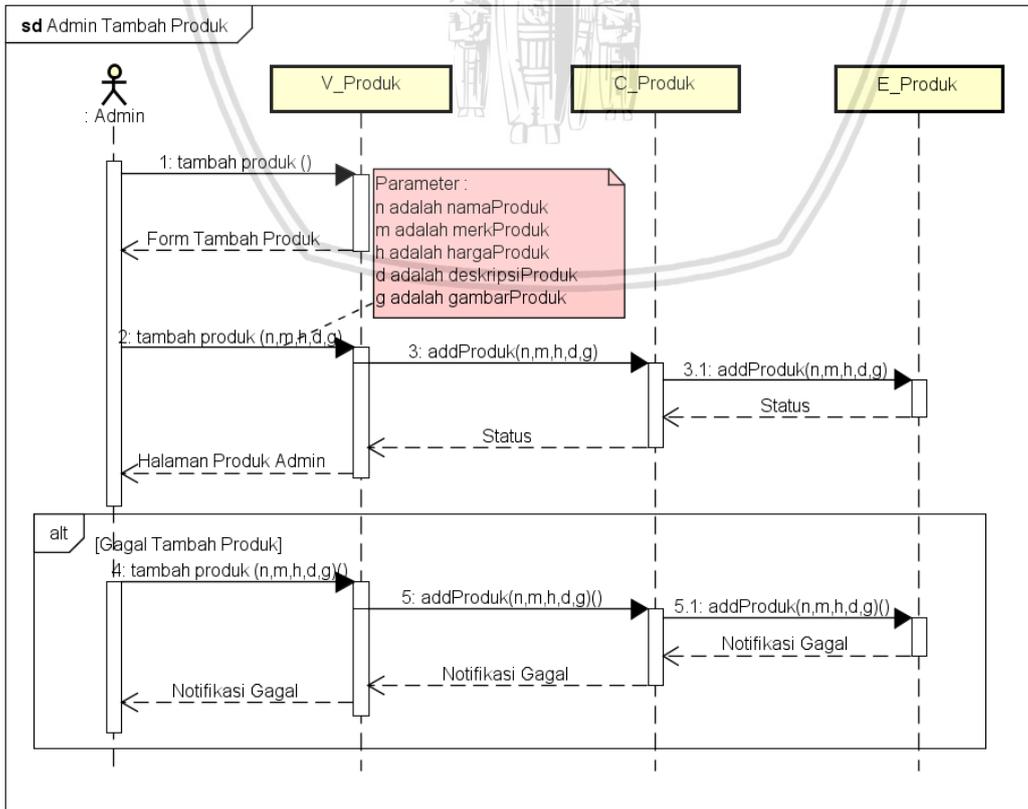
Gambar 0.49 Sequence Diagram untuk Use Case Buat Akun : (SIPB-F-01)

Gambar 4.49 adalah Sequence Diagram untuk Use Case Buat Akun : (SIPB-F-01). Alur proses ini diawali dengan pengguna mengisi kelengkapan untuk membuat akun meliputi nama perusahaan, alamat perusahaan, email perusahaan, email bagian keuangan perusahaan, telepon perusahaan dan password. Pelanggan akan memilih fungsi simpan dan sistem akan menjalankan operasi addAkun() dengan parameter yang telah dimasukan pengguna. Dalam sequence diagram ini terdapat abnormal path yang mana apabila data yang menjadi masukan ada kesamaan dengan data yang tersimpan dalam sistem, maka sistem akan menampilkan notifikasi gagal.



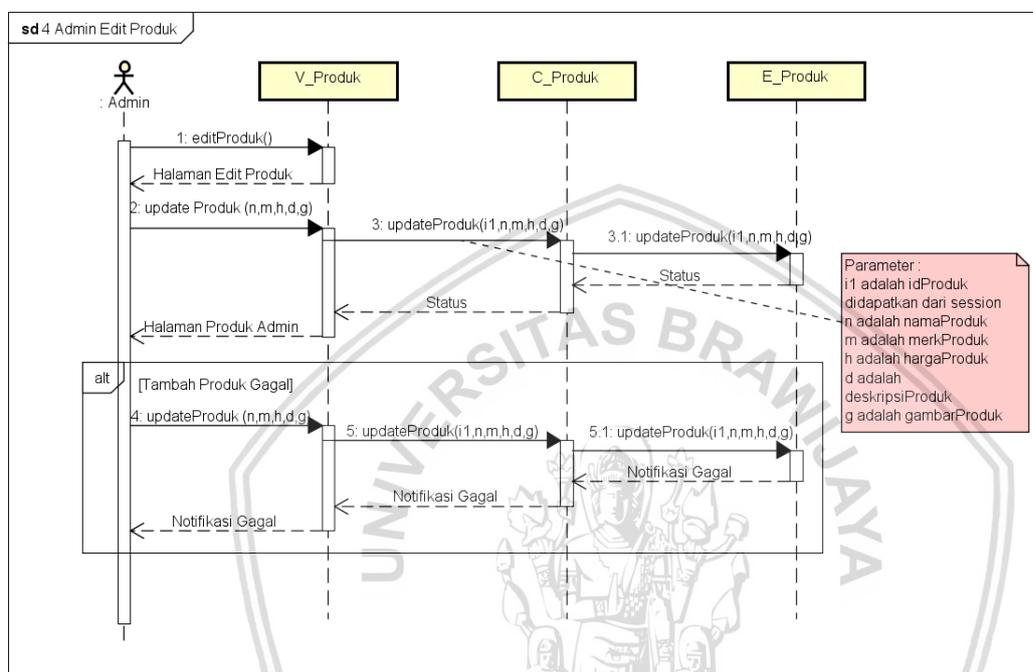
Gambar 0.50 Sequence Diagram untuk Use Case Login : (SIPB-F-02)

Gambar 4.50 adalah Sequence Diagram untuk Use Case Login : (SIPB-F-02). Alur ini diawali dengan pelanggan memilih fungsi login dengan memasukkan email perusahaan dan password. Lalu sistem akan menjalankan operasi `verifyLogin()` dengan parameter sesuai inputan pengguna. Sistem akan memeriksa apakah pengguna sudah terdaftar dalam sistem.



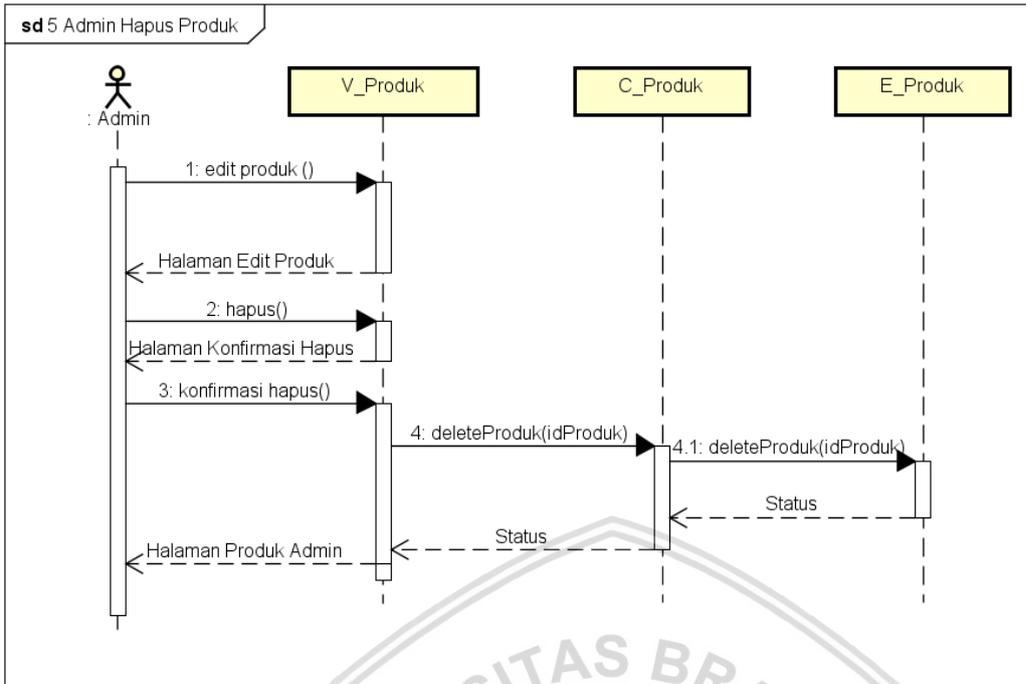
Gambar 0.51 Berikut ini adalah *Sequence Diagram* untuk *Use Case* Tambah Produk : (SIPB-F-03)

Gambar 4.51 adalah *Sequence Diagram* untuk *Use Case* Tambah Produk : (SIPB-F-03). Alur ini diawali dengan admin memilih fungsi tambah produk lalu sistem akan menampilkan form tambah produk. Admin memasukan informasi produk meliputi nama, merk, harga, deskripsi dan gambar produk. Setelah itu sistem akan menjalankan operasi addProduk() dengan parameter yang telah dimasukan oleh admin sebelumnya. Lalu sistem akan kembali ke halaman admin.



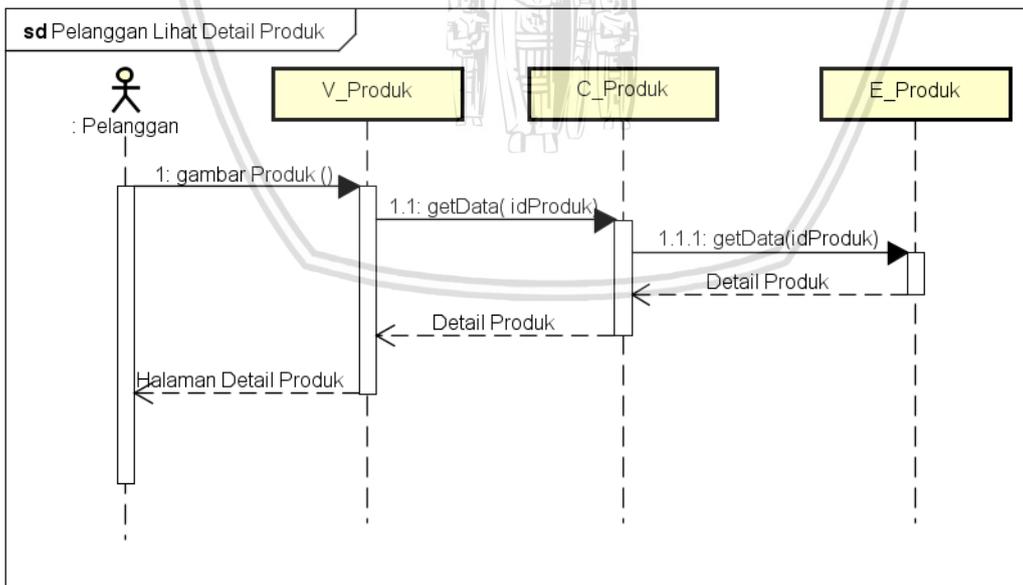
Gambar 0.52 *Sequence Diagram* untuk *Use Case* Edit Produk : (SIPB-F-04)

Gambar 4.52 adalah *Sequence Diagram* untuk *Use Case* Edit Produk : (SIPB-F-04). Alur ini diawali dengan admin memilih fungsi edit produk lalu sistem akan menampilkan halaman edit produk. Lalu admin akan memasukan perubahan informasi mengenai produk dan menyimpan perubahan. Sistem akan menjalankan operasi updateProduk(). Dalam diagram ini terdapat alur alternatif apabila gambar yang dimasukan lebih dari 500KB maka sistem akan menampilkan notifikasi bahwa proses edit produk gagal.



Gambar 0.53 Sequence Diagram untuk Use Case Hapus Produk : (SIPB-F-05)

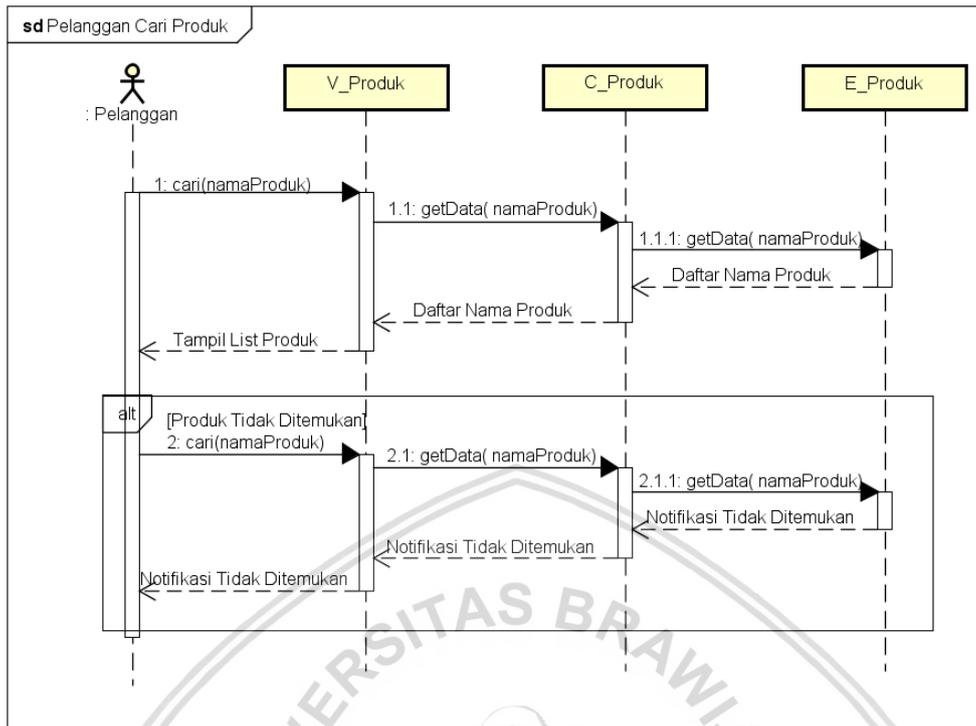
Gambar 4.53 adalah *Sequence Diagram* untuk Use Case Hapus Produk : (SIPB-F-05). Alur ini diawali dengan admin memilih operasi edit produk lalu sistem akan menampilkan halaman edit produk. Admin memilih fungsi hapus, untuk memastikan apakah produk ingin dihapus sistem meanmpilkan halaman konfirmasi hapus. Setelah admin melakukan konfirmasi hapus sistem akan menjalankan operasi deleteProduk().



Gambar 0.54 Sequence Diagram untuk Use Case Lihat informasi Produk : (SIPB-F-06)

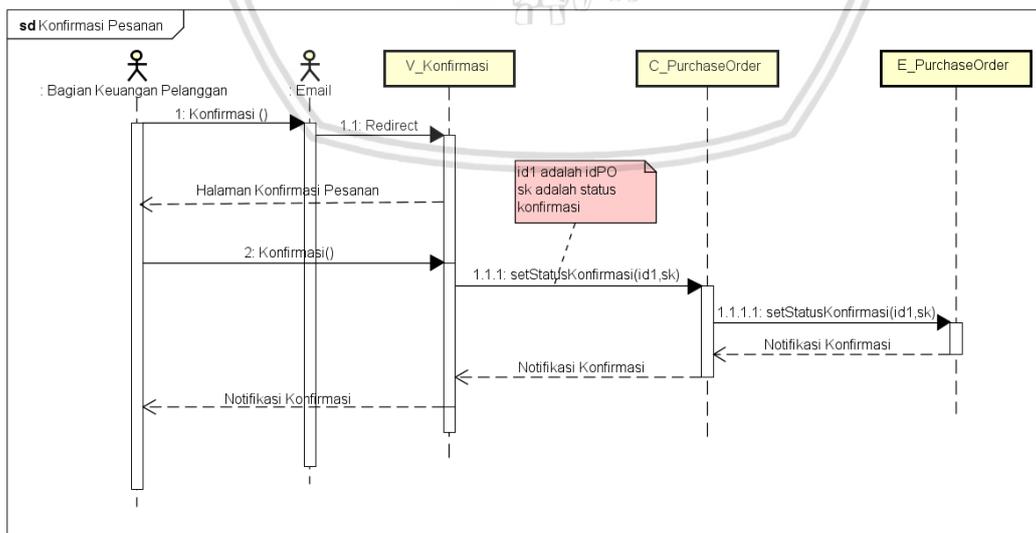
Gambar 4.54 adalah *Sequence Diagram* untuk Use Case Lihat informasi Produk : (SIPB-F-06). Alur ini diawali dengan pelanggan menekan gambar yang tersedia pada halaman produk lalu sistem akan mengambil detail informasi produk dengan menjalankan operasi getData().





Gambar 0.55 Sequence Diagram untuk Use Case Cari Produk : (SIPB-F-07)

Gambar 4.55 adalah *Sequence Diagram* untuk *Use Case Cari Produk : (SIPB-F-07)*. Alur ini diawali dengan pelanggan memasukkan nama produk yang ingin dicari lalu sistem akan menjalankan operasi `getData()` dengan parameter nama produk. Setelah itu sistem akan menampilkan list produk yang sesuai dengan masukan dari pelanggan. Dalam diagram ini terdapat alur alternatif apabila data tidak ditemukan sistem akan menampilkan notifikasi bahwa data yang dicari tidak ada dalam sistem.

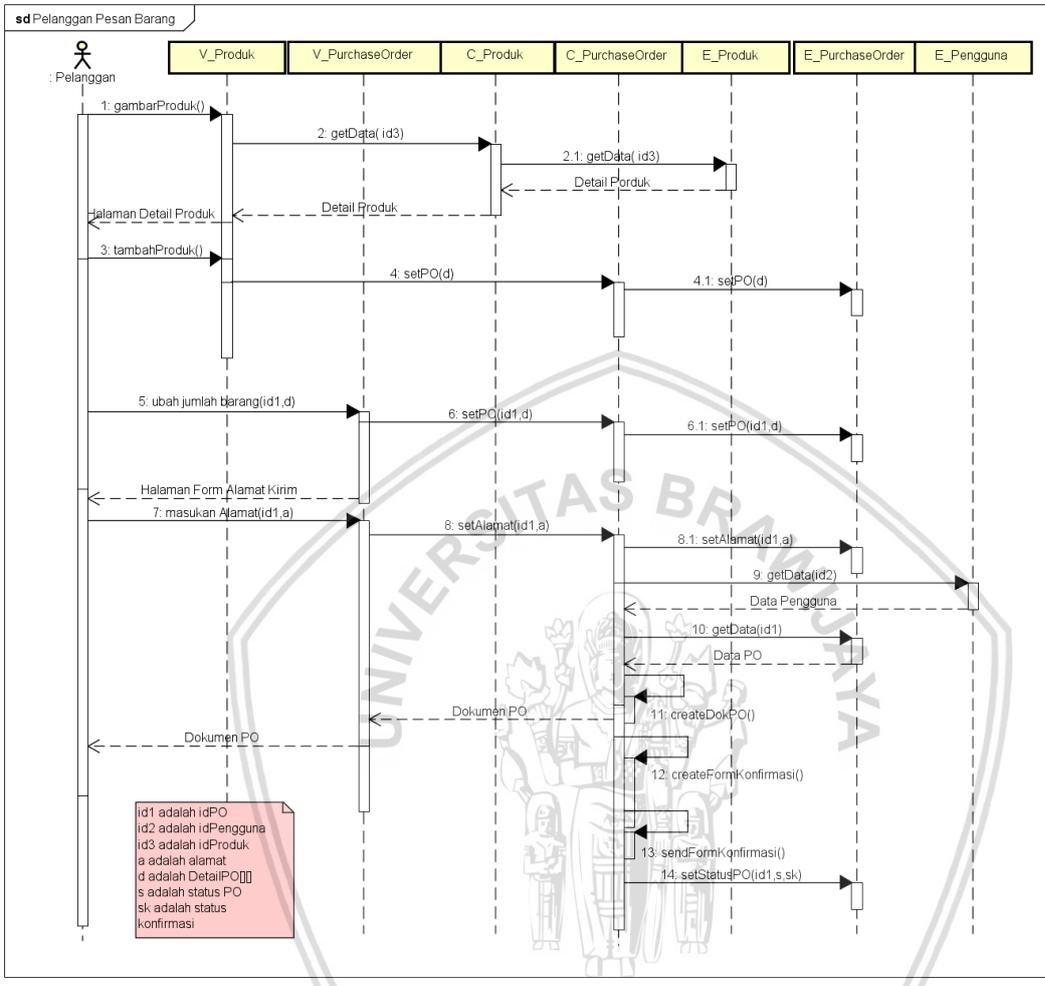


Gambar 0.56 Sequence Diagram untuk Use Case Konfirmasi Pesanan : (SIPB-F-09).

Gambar 4.56 adalah *Sequence Diagram* untuk *Use Case Konfirmasi Pesanan : (SIPB-F-09)*. Alur ini diawali dengan bagian keuangan pelanggan melakukan konfirmasi pemesanan



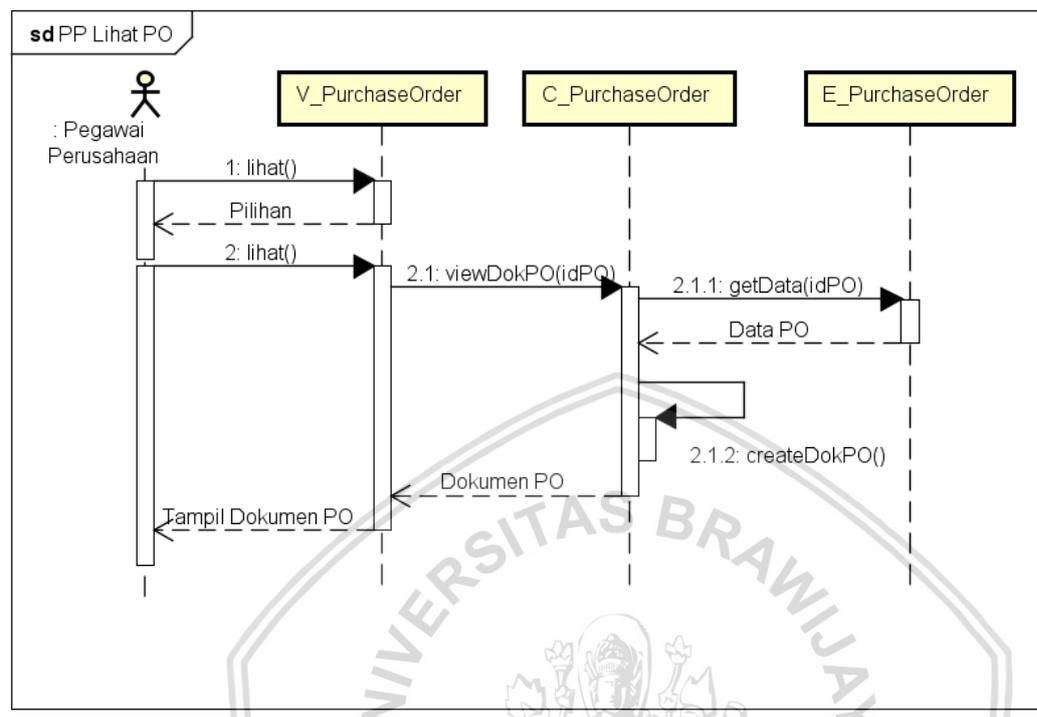
pada halaman email. Lalu email akan melink ke halaman konfirmasi. Setelah itu sistem akan menampilkan notifikasi , lalu bagian keuangan pelanggan memilih fungsi konfirmasi dan sistem merespon dengan menjalankan operasi setStatusKonfirmasi() yang mana merubah status konfirmasi dari “invalid” menjadi “valid”.



Gambar 0.57 Sequence Diagram untuk Use Case Pesan Barang : (SIPB-F-08)

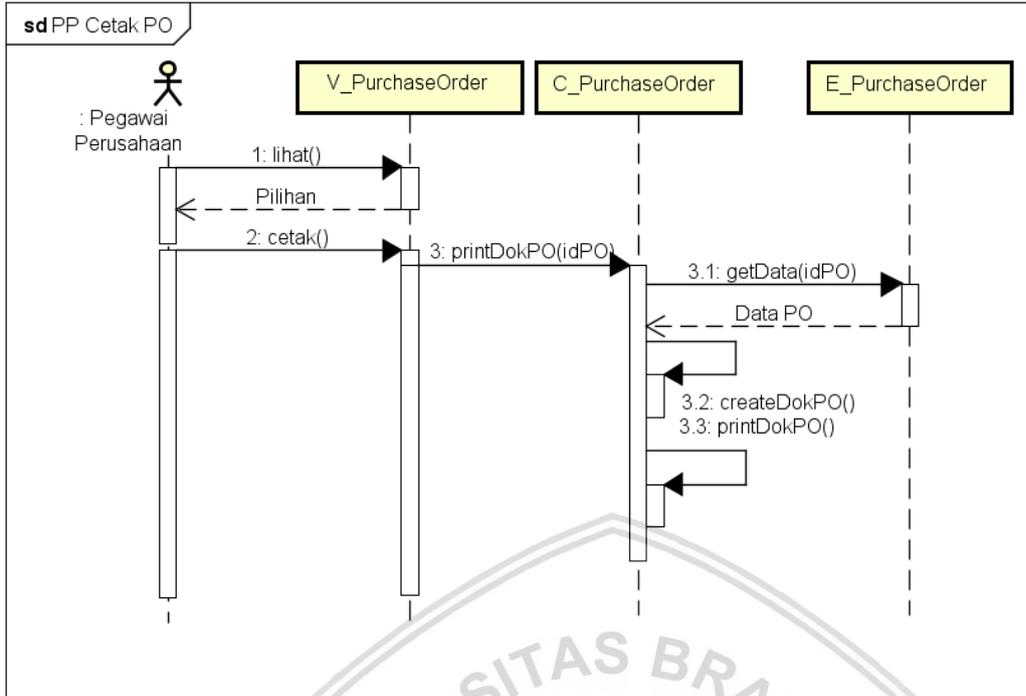
Gambar 4.58 adalah Sequence Diagram untuk Use Case Pesan Barang : (SIPB-F-08). Alur ini diawali dengan pelanggan melakukan pemesanan barang dengan memilih produk mana yang ingin dilihat detailnya. Lalu sistem akan menjalankan operasi getData() untuk mengambil data produk yang diinginkan. Lalu pelanggan memilih fungsi tambah produk dan sistem akan menjalankan operasi setPO() sesuai dengan barang yang dipilih dan diatur dengan jumlah barang 1. Setelah itu pelanggan ingin mengubah jumlah barang yang dibeli . lalu sistem akan menyimpan pesanan dan jumlah tersebut dengan menjalankan fun operasi gsi setPO(). Lalu sistem akan menampilkan halaman form alamat kirim yang digunakan untuk mengetahui barang yang telah dipesan akan dikirim kemana. Setelah itu sistem akan menyimpan alamat pengiriman tersebut dengan menjalankan operasi setAlamat() . Agar memudahkan untuk mendapatkan detail pemesanan maka sistem akan membuat dokumen PO dengan menjalankan operasi getData() pada kelas E_pengguna sehingga didapatkan informasi pengguna yang melakukan pemesanan. Dan menjalankan operasi getData() pada kelas E_PurchaseOrder untuk mengambil detail pemesanan barang. Setelah itu sistem akan membuat dokumen dengan menjalankan operasi createDokPO().

Sistem akan menampilkan dokumen itu dan menjalankan operasi createFormKonfirmasi() untuk membuat form konfirmasi yang akan dikirim ke email bagian keuangan perusahaan.



Gambar 0.58 Sequence Diagram untuk Use Case Lihat PO : (SIPB-F-10)

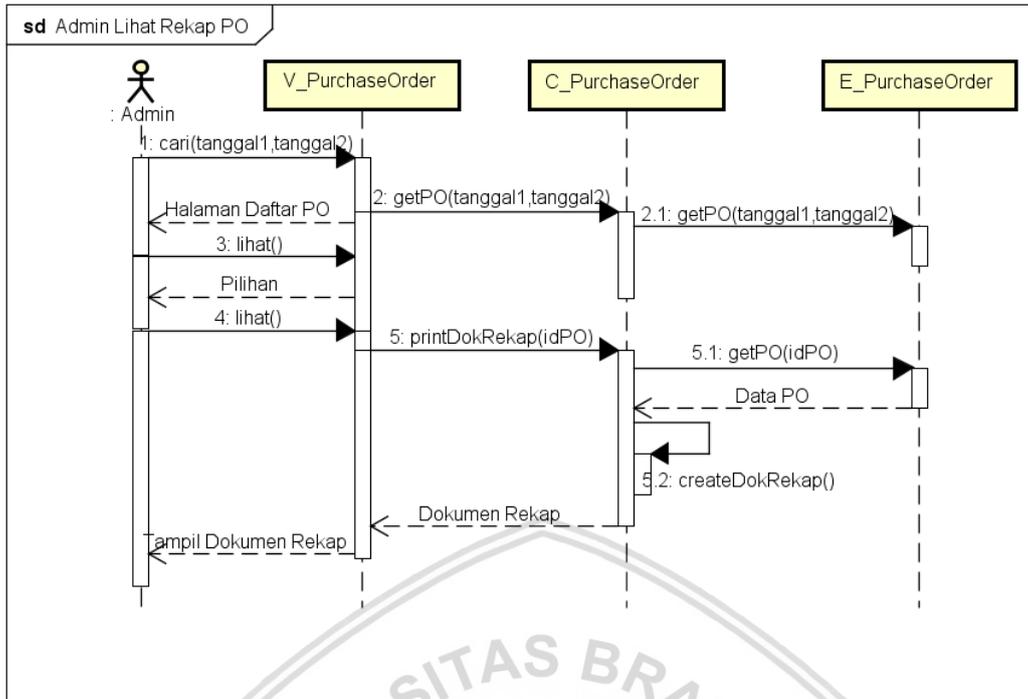
Gambar 4.58 adalah Sequence Diagram untuk Use Case Lihat PO : (SIPB-F-10). Alur ini diawali dengan pegawai perusahaan memilih fungsi lihat lalu sistem akan menampilkan apakah dokumen PO akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih lihat dan sistem akan menjalankan operasi viewDokPO() dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas E_Purchase Order lalu menjalankan operasi createDokPO().



Gambar 0.59 Sequence Diagram untuk Use Case Cetak PO : (SIPB-F-11)

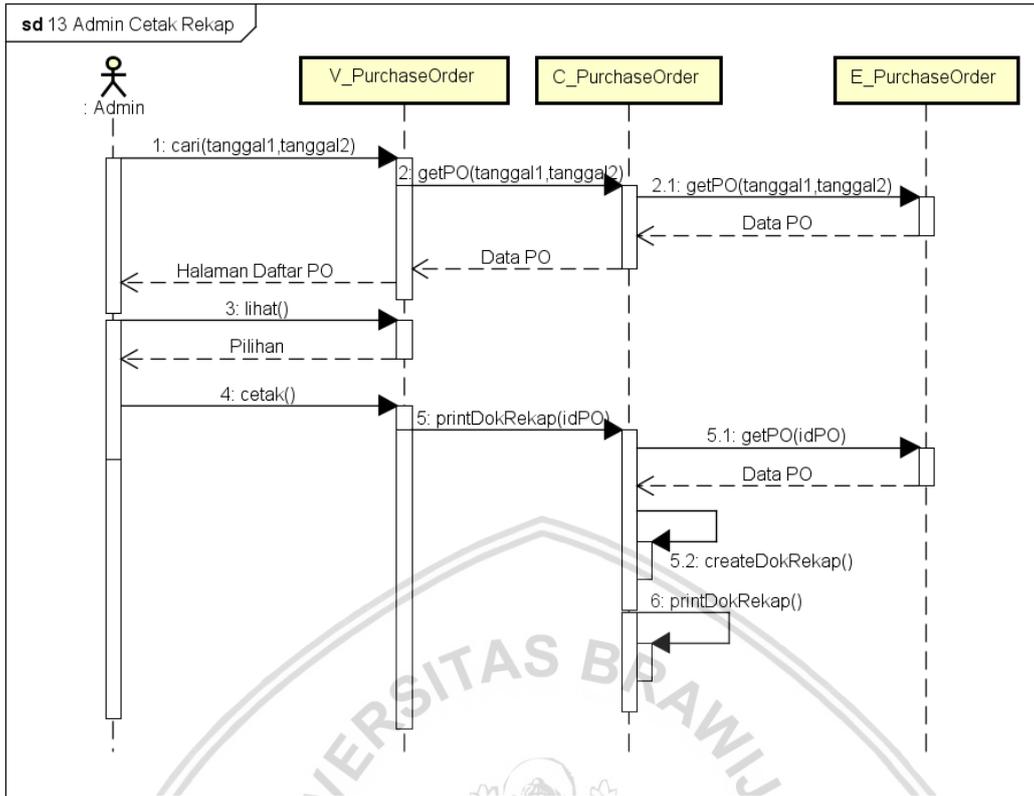
Gambar 4.59 adalah *Sequence Diagram* untuk Use Case Cetak PO : (SIPB-F-11). Alur ini diawali dengan pegawai perusahaan memilih fungsi lihat lalu sistem akan menampilkan pilihan apakah dokumen PO akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih cetak dan sistem akan menjalankan operasi printDokPO() dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas E_Purchase Order lalu menjalankan operasi createDokPO(). Setelah itu sistem akan menjalankan operasi printDokPO().





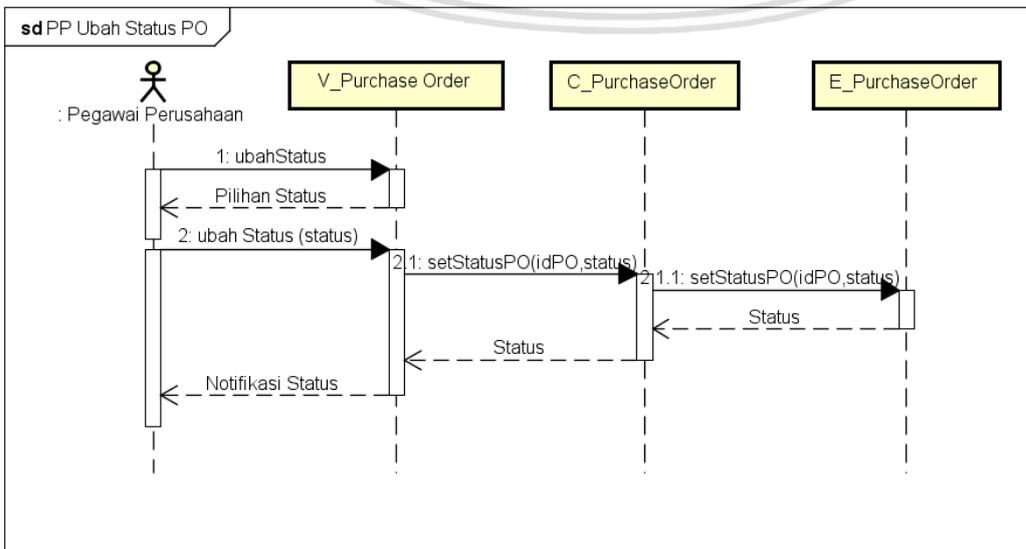
Gambar 0.60 Sequence Diagram untuk Use Case Lihat Rekap PO : (SIPB-F-12)

Gambar 4.60 adalah *Sequence Diagram* untuk *Use Case* Lihat Rekap PO : (SIPB-F-12). Alur ini diawali dengan admin memasukkan rentang waktu pemesanan barang yang diinginkan lalu sistem akan menampilkan daftar pemesanan barang pada rentang waktu tersebut dengan menjalankan operasi `getPO()`. Setelah itu admin memilih fungsi lihat lalu sistem akan menampilkan apakah dokumen PO akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih lihat dan sistem akan menjalankan operasi `viewDokRekap()` dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas `E_Purchase Order` lalu menjalankan operasi `createDokRekap()`.



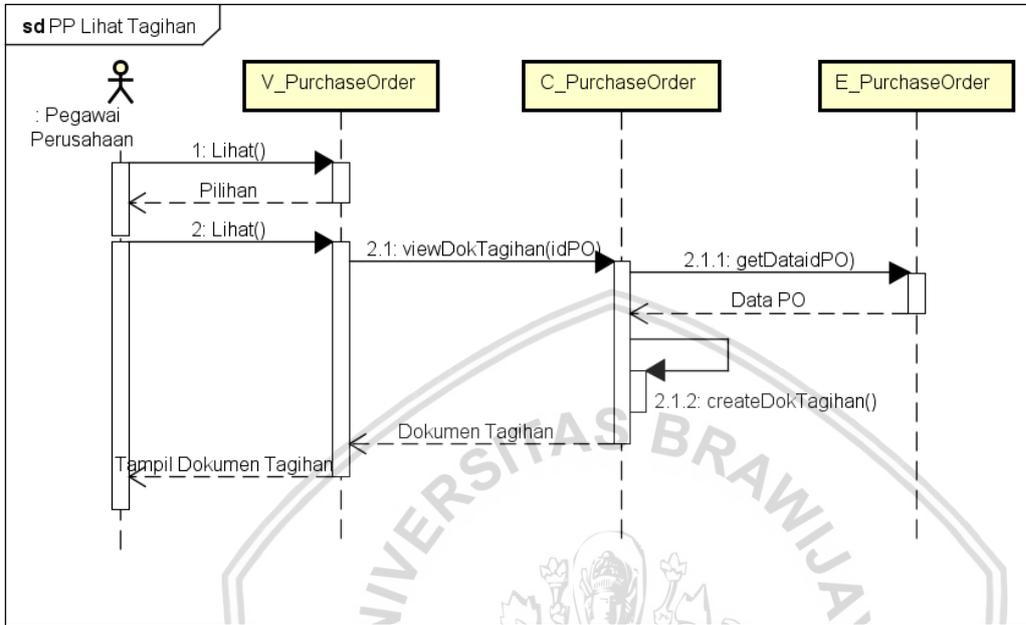
Gambar 0.61 Sequence Diagram untuk Use Case Cetak Rekap PO : (SIPB-F-13)

Gambar 4.61 adalah *Sequence Diagram* untuk *Use Case* Cetak Rekap PO : (SIPB-F-13). Alur ini diawali dengan admin memasukkan rentang waktu pemesanan barang yang diinginkan lalu sistem akan menampilkan daftar pemesanan barang pada rentang waktu tersebut dengan menjalankan operasi `getPO()`. Setelah itu admin memilih fungsi lihat lalu sistem akan menampilkan apakah dokumen PO akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih fungsi cetak dan sistem akan menjalankan operasi `printDokRekap()` dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas `E_Purchase Order` lalu menjalankan operasi `createDokRekap()`. Setelah itu mencetak dokumen dengan menjalankan operasi `printDokRekap()`.



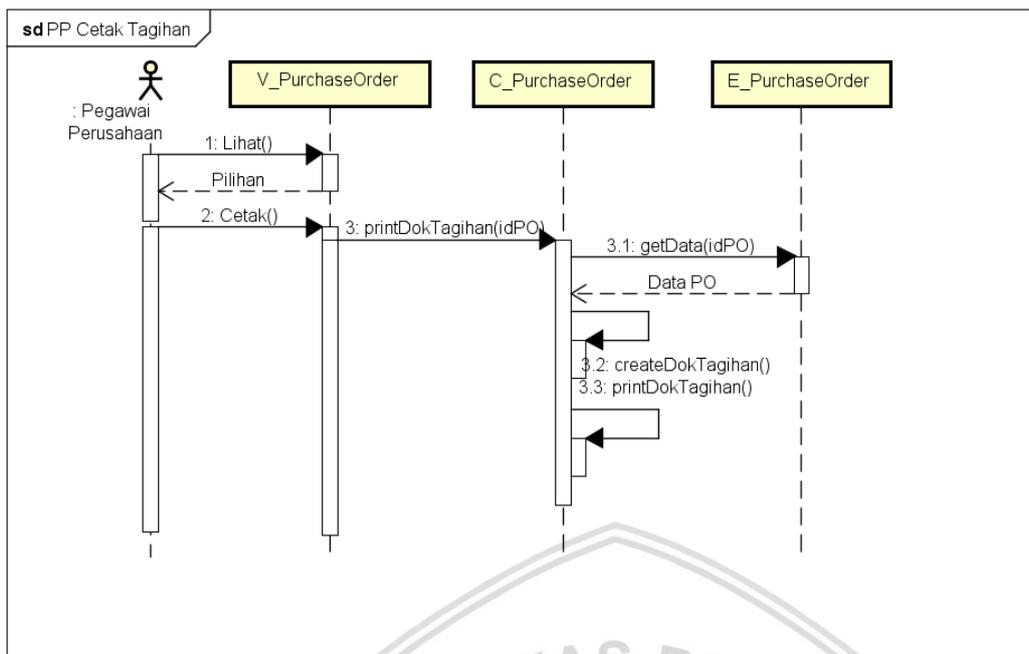
Gambar 0.62 Sequence Diagram untuk Use Case Ubah Status PO : (SIPB-F-14)

Gambar 4.62 adalah *Sequence Diagram* untuk Use Case Ubah Status PO (SIPB-F-14). Alur ini diawali dengan pegawai perusahaan memilih fungsi ubah status dan sistem akan menampilkan macam macam pilihan status. Pegawai perusahaan memilih status lalu sistem akan mengubah status PO pada suatu idPO.



Gambar 0.63 Sequence Diagram untuk Use Case Lihat Tagihan : (SIPB-F-15)

Gambar 4.63 adalah *Sequence Diagram* untuk Use Case Lihat Rekap Penjualan : (SIPB-F-15). Alur ini diawali dengan pegawai perusahaan memilih fungsi lihat lalu sistem akan menampilkan pilihan apakah dokumen tagihan akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih lihat dan sistem akan menjalankan operasi viewDokTagihan() dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas E_Purchase Order lalu menjalankan operasi createDokTagihan().



Gambar 0.64 Sequence Diagram untuk Use Case Cetak Tagihan: (SIPB-F-16)

Gambar 4.64 adalah *Sequence Diagram* untuk *Use Case* Cetak Tagihan : (SIPB-F-16). Alur ini diawali dengan pegawai perusahaan memilih fungsi lihat lalu sistem akan menampilkan pilihan apakah dokumen tagihan akan ditampilkan saja atau dicetak. Pegawai perusahaan akan memilih cetak dan sistem akan menjalankan operasi `printDokTagihan()` dan membuat dokumen dengan mengambil data terlebih dahulu ke kelas `E_Purchase Order` lalu menjalankan operasi `createDokTagihan()`. Setelah itu sistem akan menjalankan fungsi `printDokTagihan()`.

4.5.3 Database Schema

Database Schema adalah sebuah kerangka yang digunakan untuk menyusun *database*. *Database schema* mendapatkan masukan dari hasil entitas yang telah digambarkan sebelumnya di *class diagram*.

Berikut ini adalah nama tabel beserta atribut dari sistem pemesanan barang PT. Trivia Nusantara.

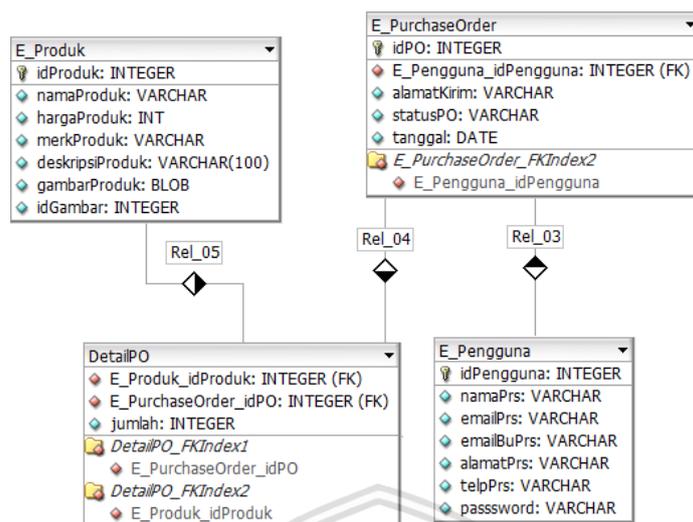
E_Pengguna (`idPengguna:INTEGER`, `namaPrs:VARCHAR`, `emailPrs: VARCHAR`, `emailBuPrs:VARCHAR`, `alamatPrs:VARCHAR`, `telpPrs:VARCHAR`, `passsword:VARCHAR`)

E_Produk (`idProduk:INTEGER`, `namaProduk:VARCHAR`, `hargaProduk:INT`, `merkProduk:VARCHAR`, `deskripsiProduk:VARCHAR(100)`, `gambarProduk:BLOB`, `idGambar:INTEGER`)

E_PurchaseOrder(`idPO:INTEGER`, `E_Produk_idProduk:INTEGER`, `jumlah:INTEGER`, `alamatKirim:VARCHAR`, `statusPO:VARCHAR`, `tanggal:DATE`)

DetailPO(`idPO:INTEGER`, `idProduk:INTEGER`,`jumlah:int`)

Berikut *database schema* dari sistem pemesanan barang PT. Trivia Nusanta terdapat pada gambar di bawah ini .



Gambar 0.65 Database Schema

4.5.4 Penyelesaian User Interface

Penyelesaian desain antar muka untuk sistem informasi pemesanan barang PT. Trivia Nusantara dapat dilihat pada gambar 4.6-4.23. Penyelesaian desain antar muka tidak dituliskan pada sub-bagian ini karena desain antarmuka pada fase *design* sama seperti pada fase *requirements*.

Kesimpulan dari fase *design* terdapat 4 tahap yaitu membuat *class diagram*, membuat *sequence diagram*, membuat *database schema* dan melakukan penyelesaian desain antarmuka. Pada tahap membuat *class diagram*, semua *operation* dan atribut yang sudah diidentifikasi pada fase *analysis* dikelompokkan sesuai kelas kelasnya. Setelah itu digambarkan relasi antar kelasnya, dalam *class diagram* yang digambarkan terdapat 3 *controller* dan 3 *entity*. Setelah menggambarkan *class diagram* kegiatan yang dilakukan setelahnya yaitu membuat *sequence diagram*, masukan yang digunakan untuk membuat *sequence diagram* berasal dari *communication diagram*. Berbeda dengan membuat *communication diagram* apabila pada *communication diagram* tujuan utamanya adalah memperlihatkan hubungan antar *class* yang ada sedangkan, pada *sequence diagram* urutan proses lebih terlihat. Setelah itu dilanjutkan dengan membuat *database schema* yang digunakan untuk melakukan penggambaran *database* yang akan digunakan untuk membangun sistem. Dalam *database schema* didapatkan dari hasil kelas entitas yang sebelumnya sudah digambarkan di *class diagram*. Dalam *database schema* yang dirancang terdapat 4 tabel yaitu E_Produk, E_PurchaseOrder, E_Pengguna dan DetailPO.

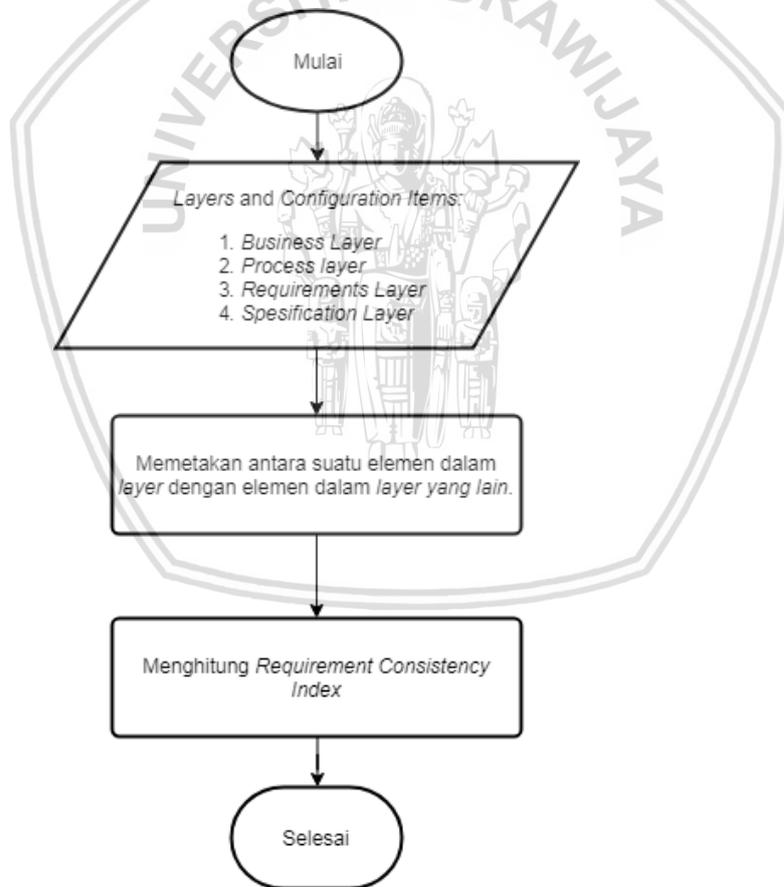
BAB 5 EVALUASI PERANCANGAN

Setelah melakukan analisis kebutuhan dan perancangan sistem, kegiatan selanjutnya adalah evaluasi hasil perancangan sistem informasi pemesanan barang. Evaluasi perancangan dilakukan menggunakan metode *Consistency Analysis: Requirement Configuration Structure* dan *Correctness*.

5.1 *Consistency Analysis: Requirement Configuration Structure*

Framework requirements configuration structure digunakan untuk melakukan evaluasi konsistensi perancangan, terutama berokus pada konsistensi pendefinisian kebutuhan sebelum dilakukannya perancangan sebuah sistem. *Framework* ini mempunyai 4 komponen kerangka kerja yang terdiri dari : *Layers of the Model*, *Requirement Configuration Structure*, *Consistency Analysis Method*, *Requirement Consistency Index*.

Proses yang dilakukan untuk melakukan evaluasi konsistensi terhadap pendefinisian kebutuhan menggunakan *requirements configuration structure* akan dijelaskan pada gambar 5.1



Gambar 0.1 Diagram Alir *Consistency Analysis*

Pada proses evaluasi perancangan menggunakan consistency analysis langkah pertama yang dilakukan adalah mengidentifikasi *item* pada setiap *layer* bisnis, *layer* proses, *layer* kebutuhan dan *layer* spesifikasi. Setelah melakukan identifikasi setiap *item* tersebut proses pemetaan akan dilakukan untuk mengetahui hubungan setiap item pada *layer*. Hasil pemetaan tersebut akan dihitung *Requirement Consistency Index* nya dengan mengetahui jumlah total elemen kebutuhan, jumlah elemen kebutuhan tidak terdefinisi, dan elemen kebutuhan yang konsisten.

5.1.1 Layers and Configuration Items

Framework ini mempunyai 4 jenis *layer* yang harus ditentukan sebelum melakukan evaluasi konsistensi:

1. Business Layer

Masukan untuk *layer* ini diperoleh dari dokumen maupun proses yang berjalan pada sebuah organisasi. Pada penelitian ini, *business use case list* akan menjadi masukan *business layer*.

2. Process Layer

Masukan untuk *process layer* adalah suatu proses atau sub-proses yang harus ada untuk mencapai tujuan organisasi. Pada penelitian ini, masukan dari *layer* ini didapatkan dari hasil wawancara dan analisis *business use case list*.

3. Requirements layer

Masukan untuk *layer* ini adalah yang berisi kunci dari kebutuhan sistem berdasarkan proses dan sub-proses. Pada penelitian ini masukan untuk *layer* ini didapatkan dari dokumen poin 4.2.1 *Project Genesis*.

4. Specification Layer

Masukan untuk *layer* ini adalah yang berisi kunci dari kebutuhan sistem berdasarkan proses dan sub-proses. Pada penelitian ini masukan untuk *layer* ini didapatkan dari dokumen poin 4.3.7 *System Use Case List* yang mana dalam sub bab tersebut telah disebutkan nama spesifikasi atau *use case* yang dibutuhkan dalam sistem.

5.1.2 Requirement Configuration Structure

1. Business Layer

Pada pemodelan bisnis saat ini terdapat 4 proses dalam proses bisnis pemesanan barang. Tabel 5.1 akan menjelaskan masukan dari *business layer*. Setiap masukan akan di representasikan dalam kode BL-01 hingga BL-04.

Tabel 0.1 Masukan Business Layer

No.	Kode Evaluasi	Uraian
1	BL-01	Pelanggan melakukan pemesanan barang

2	BL-02	Petugas melakukan konfirmasi kepada bagian keuangan pelanggan
3	BL-03	Bagian gudang memeriksa stock barang pesanan.
4	BL-04	Kurir mengirim barang dan memberikan surat penagihan

2. Process Layer

Pada pemodelan bisnis saat ini terdapat 3 proses yang menjadi kunci dalam proses bisnis pemesanan barang. Tabel 5.2 akan menjelaskan masukan dari *process layer*. Setiap masukan akan di representasikan dalam kode PL-01 hingga PL-03.

Tabel 0.2 Masukan Process Layer

No.	Kode Evaluasi	Uraian
1	PL-01	Pelanggan melakukan pemesanan barang
2	PL-02	Petugas melakukan konfirmasi kepada bagian keuangan pelanggan
3	PL-03	Kurir mengirim barang dan memberikan surat penagihan

3. Requirements layer

Pada kebutuhan fungsional sistem terdapat beberapa kebutuhan mengenai fungsi dalam sistem yang ingin dibangun. Dalam sistem ini dibutuhkan 7 fungsi yang mana akan menjadi masukan dalam *requirement layer*. Setiap masukan pada *requirement layer* akan di representasikan melalui kode RL-01 hingga RL-07

Tabel 0.3 Masukan Requirements Layer

No.	Kode Evaluasi	Uraian
1	RL-01	Penggunaan User
2	RL-02	Mengelola Informasi Produk
3	RL-03	Mendapatkan Informasi Produk
4	RL-04	Melakukan Pemesanan Barang
5	RL-05	Melakukan Konfirmasi Pemesanan
6	RL-06	Mengelola Pesanan
7	RL-07	Mengelola Tagihan

4. Specification Layer

Dalam penelitian ini terdapat 16 spesifikasi sistem yang akan diuraikan di representasikan melalui kode SL-01 hingga SL-16.

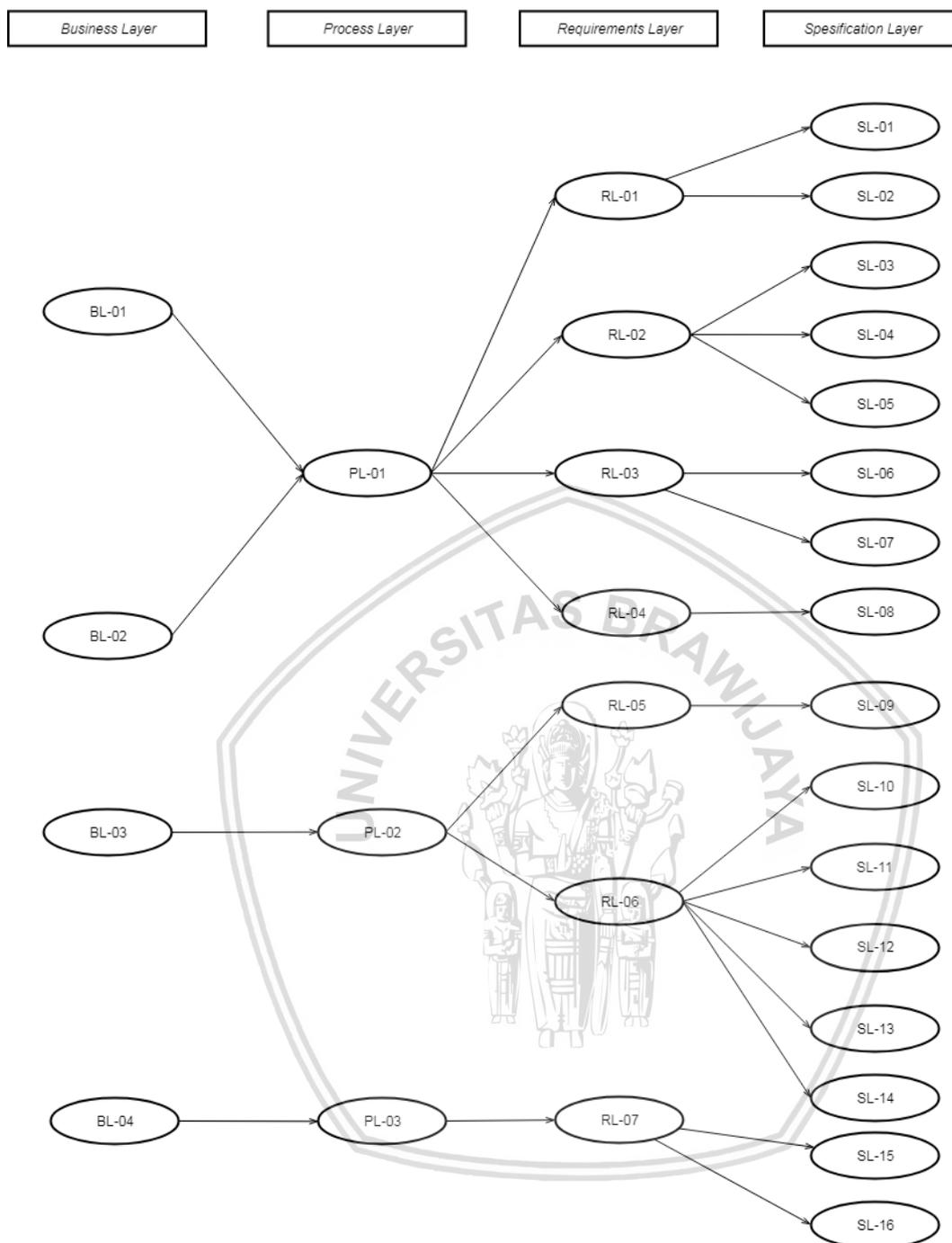
Tabel 0.4 Masukan Specification Layer

No.	Kode Evaluasi	Uraian
1	SL-01	Buat Akun
2	SL-02	Login

No.	Kode Evaluasi	Uraian
3	SL-03	Tambah Produk
4	SL-04	Edit Produk
5	SL-05	Hapus Produk
6	SL-06	Lihat informasi Produk
7	SL-07	Cari Produk
8	SL-08	Pesan Barang
9	SL-09	Konfirmasi Pesanan
10	SL-10	Lihat PO
11	SL-11	Cetak PO
12	SL-12	Lihat Rekap Penjualan
13	SL-13	Cetak Rekap Penjualan
14	SL-14	Status PO
15	SL-15	Lihat Tagihan
16	SL-16	Cetak Tagihan

5.1.3 Consistency Analysis Method

Consistency analysis method menentukan tingkat konsistensi kebutuhan terhadap tujuan dari sistem pemesanan barang dengan melakukan penelusuran pada setiap hubungan yang dimiliki oleh setiap item pada setiap layer. Gambar 5.2 merupakan *requirement configuration structure* yang menggambarkan hubungan dari setiap masukan pada *layer*. Serta menilai konsistensi, ketepatan dan kesesuaian setiap hubungan.



Gambar 0.2 Consistency Analysis Method

Pendefinisian kebutuhan yang konsisten mempunyai indikator bahwa setiap elemen kebutuhan pada *process layer* memiliki hubungan dengan elemen kebutuhan pada *business layer*, elemen kebutuhan pada *requirement layer* memiliki hubungan dengan elemen kebutuhan pada *process layer*, elemen kebutuhan pada *specification layer* memiliki hubungan dengan elemen kebutuhan pada *requirement layer*. Ketika ada elemen kebutuhan pada sebuah layer tidak memiliki dengan elemen pada layer sebelumnya, maka pendefinisian elemen kebutuhan tersebut dapat dikatakan tidak konsisten.

Berdasarkan gambar sebelumnya, dapat disimpulkan bahwa seluruh elemen kebutuhan dari setiap *layer* sudah mencakup keseluruhan *use case* sesuai dengan tujuan

pengguna (*business layer*), dan tidak ada pendefinisian elemen kebutuhan yang tidak berdasar pada tujuan pengguna (*business layer*).

5.1.4 Requirement Consistency Index

Requirement Consistency Index merupakan perhitungan matematis untuk menentukan persentase tingkat konsistensi kebutuhan. Suatu kebutuhan dianggap baik apabila nilai persentasenya tinggi.

Metode analisis konsistensi pada gambar 5.2 menjelaskan bahwa:

- Total kebutuhan (A) yang konsisten berjumlah 31
- Total item kebutuhan (B) berjumlah 31 yang dimasukkan pada nilai B. (4 *business layer*, 3 *process layer*, 7 *requirement layer*, dan 16 *spesification layer*)
- Total kebutuhan yang tidak terdefinisi (C) berjumlah 0

Setiap variable yang telah diidentifikasi akan dimasukkan dalam penilaian pada rumus:

$$RCI = A/(B+C)$$

Dengan rumus tersebut maka di temukan hasil dari nilai RCI sistem pemesanan barang 100 %. Hal ini membuktikan bahwa masing-masing item pada setiap layer bersifat konsisten/kebutuhan terpenuhi.

5.2 Correctness

Evaluasi ini bertujuan untuk memastikan bahwasannya setiap kebutuhan yang dimiliki pada sistem sudah sesuai (*correctness*) terhadap kebutuhan dan perancangan dari PT. Trivia Nusantara. Pada akhir evaluasi akan terlihat apakah fungsi-fungsi tersebut sesuai (*correctness*), hanya sebagian yang sesuai (*partially correctness*) atau tidak sesuai sama sekali (*terminate normally*).

5.2.1 Kebutuhan Sistem

Kebutuhan sistem merupakan hasil dari analisis permasalahan dan kebutuhan yang dilakukan pada fase *requirements* dan didapatkan dari *user story*. Kebutuhan sistem diambil pada Bab 4 sub bab 4.3.7 *System Use Case List* pada penelitian ini. Kebutuhan sistem berjumlah 7 kebutuhan. Kebutuhan sistem dapat dilihat pada tabel dibawah ini.

Tabel 0.5 Kebutuhan Sistem

No.	Kode Kebutuhan	Nama Kebutuhan
1.	(SIPB-R-01)	Penggunaan User
2.	(SIPB-R-02)	Mengelola Informasi Produk
3.	(SIPB-R-03)	Mendapatkan Informasi Produk
4.	(SIPB-R-04)	Melakukan Pemesanan Barang

No.	Kode Kebutuhan	Nama Kebutuhan
5.	(SIPB-R-05)	Melakukan Konfirmasi Pemesanan
6.	(SIPB-R-06)	Mengelola Pesanan
7.	(SIPB-R-07)	Mengelola Tagihan

5.2.2 Fungsi Sistem

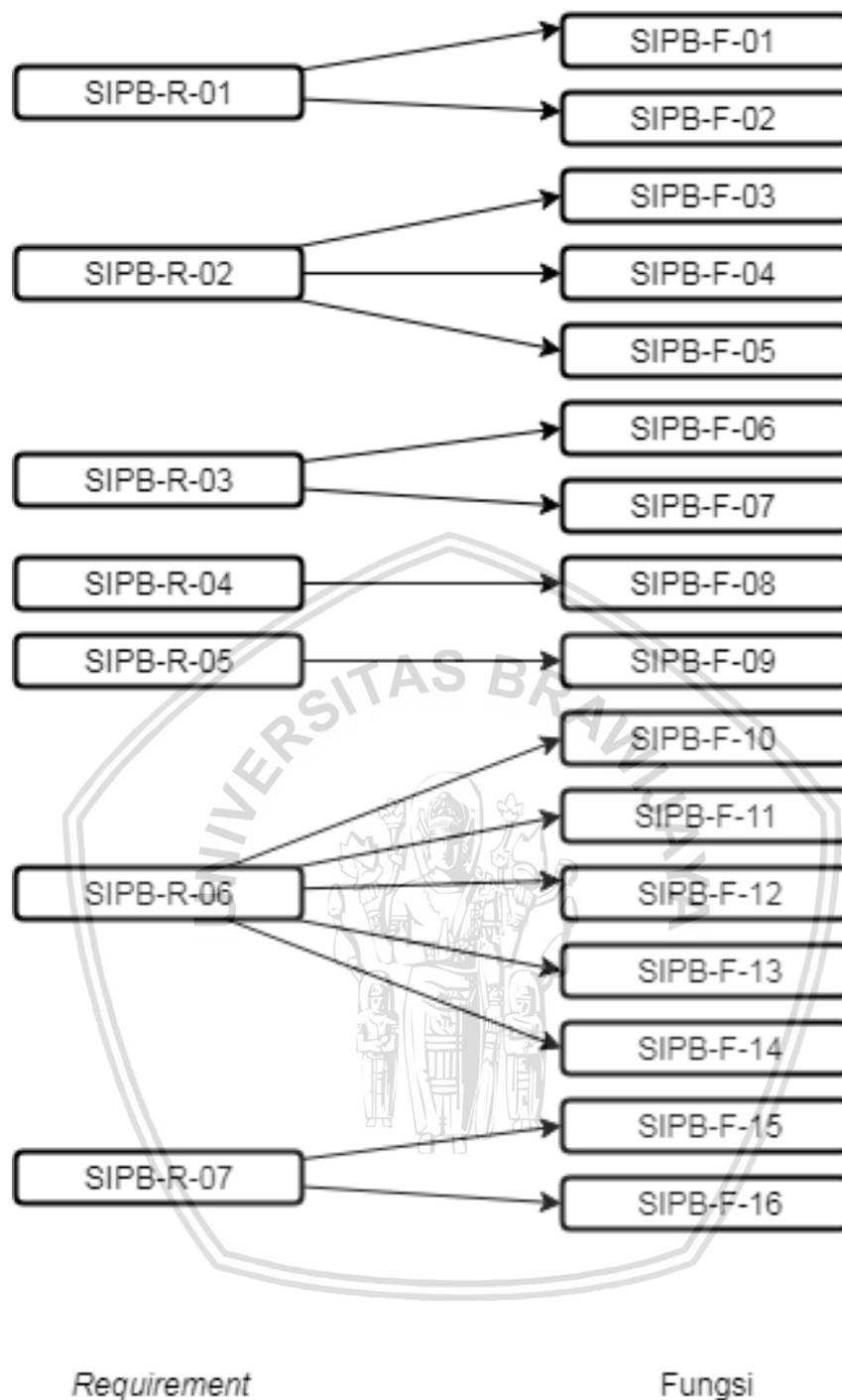
Fungsi sistem merupakan hasil dari analisis permasalahan dan kebutuhan yang dilakukan pada fase *requirements* dan didapatkan dari *user story*. Kebutuhan sistem diambil pada Bab 4 sub bab 4.3.7 *System Use Case List* pada penelitian ini. Kebutuhan sistem berjumlah 16 fungsi sistem.

Tabel 0.6 Fungsi Sistem

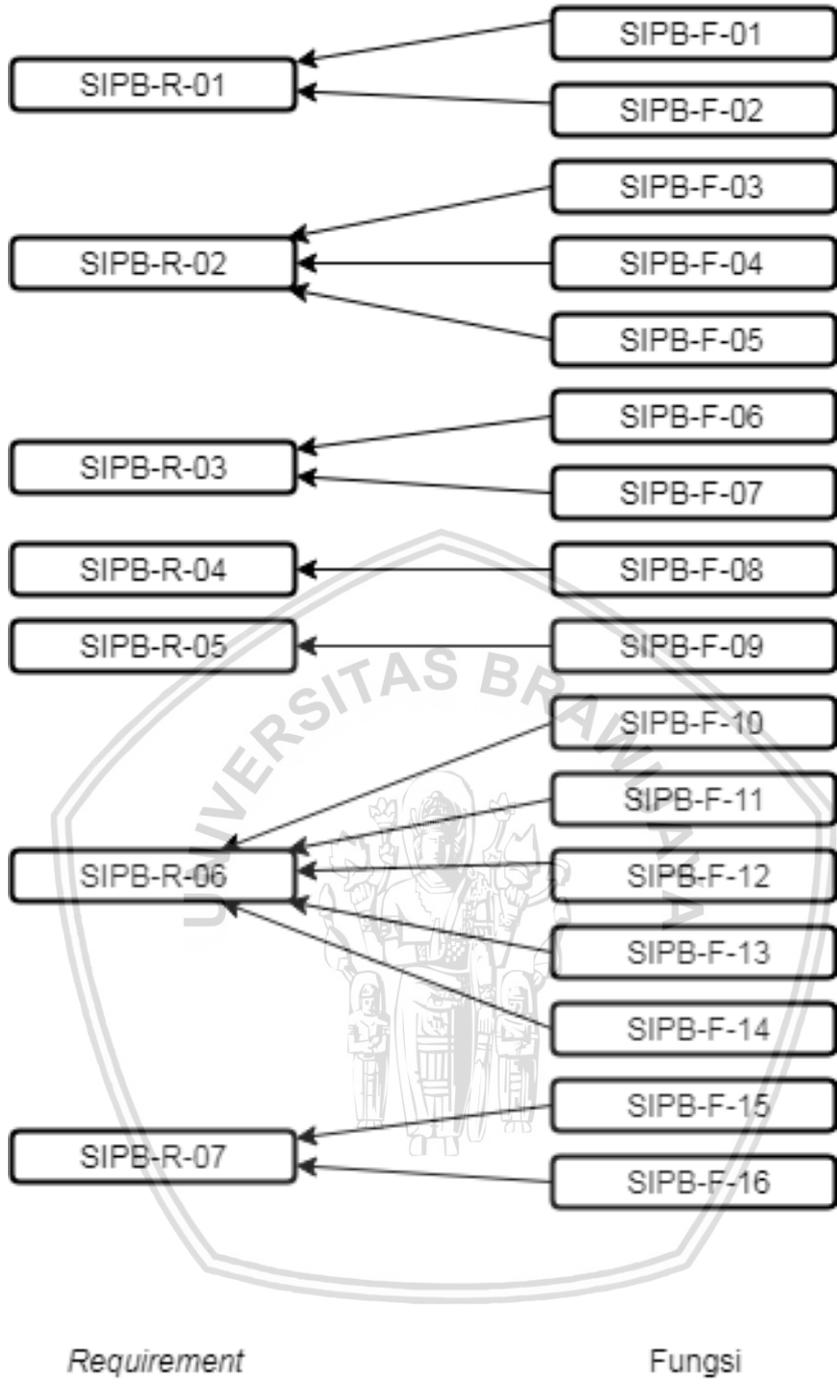
No.	Kode Fungsi	Deskripsi
1.	(SIPB-F-01)	Buat Akun
2.	(SIPB-F-02)	Login
3.	(SIPB-F-03)	Tambah Produk
4.	(SIPB-F-04)	Edit Produk
5.	(SIPB-F-05)	Hapus Produk
6.	(SIPB-F-06)	Lihat informasi Produk
7.	(SIPB-F-07)	Cari Produk
8.	(SIPB-F-08)	Pesan Barang
9.	(SIPB-F-09)	Konfirmasi Pesanan
10.	(SIPB-F-10)	Lihat PO
11.	(SIPB-F-11)	Cetak PO
12.	(SIPB-F-12)	Lihat Rekap Penjualan
13.	(SIPB-F-13)	Cetak Rekap Penjualan
14.	(SIPB-F-14)	Status PO
15.	(SIPB-F-15)	Lihat Tagihan
16.	(SIPB-F-16)	Cetak Tagihan

5.2.3 Correctness: Proposition

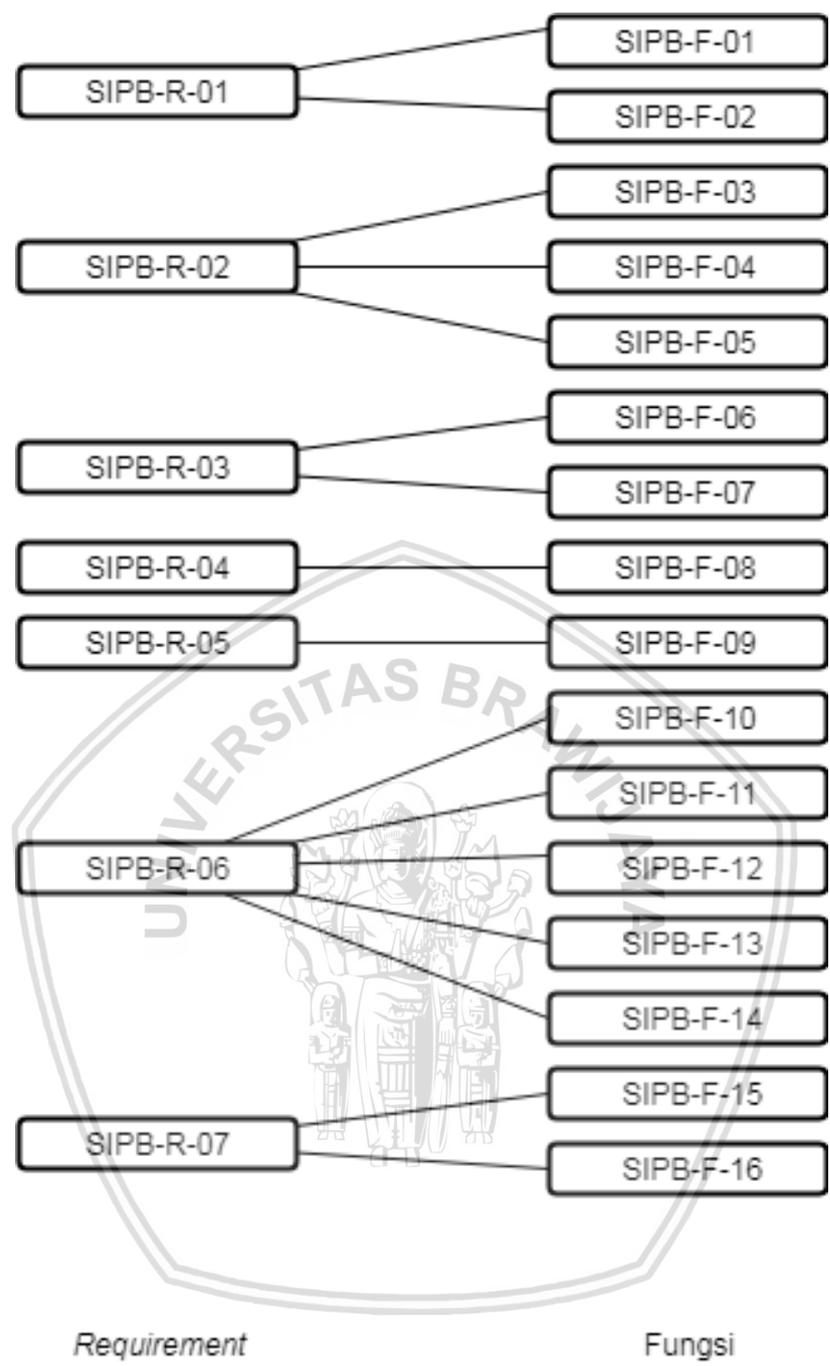
Setelah melakukan pendefinisian kebutuhan sistem dan juga fungsi sistem, maka kebutuhan dan fungsi sistem akan direlasikan untuk mengetahui kebutuhan apa saja yang tidak didefinisikan dalam fungsi sistem. Gambar 5.3- aGambar 5.5 akan menjelaskan relasi antara kebutuhan dan sistem.



Gambar 0.3 Hubungan Kebutuhan Sistem dengan Fungsi Sistem (dom(R))



Gambar 0.4 Gambar Relasi Fungsi Sistem dengan Kebutuhan Sistem (dom(P))



Gambar 0.5 Hasil (R∩P)

Dengan melakukan identifikasi dan merelasikan kebutuhan sistem dengan fungsi sistem maka didapatkan bahwa setiap anggota *requirement*(R) beririsan dengan anggota himpunan fungsi(P) akan menghasilkan nilai $dom(R)$. Dari gambar tersebut disimpulkan bahwa semua kebutuhan (R) didukung oleh semua fungsi sistem (P). Yang menghasilkan nilai $(dom(R \cap P)) = nilai (dom(R))$. Sehingga hasil evaluasi *correctness* yang dilakukan masuk kedalam kategori sesuai (*correctness*).

5.3 Analisis Hasil Evaluasi Kebutuhan

Analisis evaluasi kebutuhan menghasilkan nilai RCI (*Requirement Consistency Index*) sebesar 100% dan uji *correctness* termasuk dalam golongan *correctness*. Evaluasi dengan menggunakan *consistency analysis* dilakukan dengan mengidentifikasi kebutuhan kedalam 4 *layer*, *layer business*, *layer process*, *layer requirement* dan *layer specification*. Selanjutnya menghubungkan setiap *layer* untuk mengetahui hubungan antar *layer*. Nilai 100% diperoleh dalam perhitungan presentase konsistensi.

Sedangkan dalam uji *correctness* mengidentifikasi kesesuaian antara kebutuhan sistem dengan fungsi yang dimiliki sistem yang akan dikembangkan. Dari hasil uji *correctness* diketahui bahwa setiap kebutuhan telah diakomodir oleh setiap spesifikasi yang disediakan sistem pemesanan barang hal tersebut menjelaskan bahwa uji *correctness* menghasilkan kategori sesuai (*correctness*).



BAB 6 KESIMPULAN DAN SARAN

Pada bab ini akan menjelaskan kesimpulan dari kegiatan penelitian yang telah dilakukan dan saran yang dapat diberikan dari penelitian ini untuk perkembangan sistem selanjutnya.

6.1 Kesimpulan

Metodologi Ripple yang merupakan metode *object oriented analysis and design* dapat digunakan sebagai kerangka dalam melakukan analisis untuk mendapatkan kebutuhan yang berguna sebagai dasar pembangunan Sistem Informasi Pemesanan Barang PT. Trivia Nusantara. Langkah-langkah yang dilakukan adalah sebagai berikut :

1. Pada fase *genesis*, dilakukan penggalian kebutuhan pengguna sebagai dasar pembuatan *project genesis*. Pada *project genesis* didapatkan kebutuhan organisasi mengenai sistem yang akan dibangun. Masukan untuk mendapatkan ide tersebut berasal dari *user story* yang telah dipaparkan oleh narasumber. Setelah mendapatkan kebutuhan suatu sistem, tahap selanjutnya adalah membuat *glossary*, yang digunakan untuk menyamakan pemahaman apabila suatu hari akan ada *maintenance* pada sistem. Terdapat 25 istilah beserta relasinya terhadap sistem.
2. Pada fase *requirements*, bertujuan untuk mendapatkan kebutuhan suatu sistem dengan melakukan analisis pada proses bisnis pemesanan barang yang sedang dijalankan pada PT. Trivia Nusantara. Dalam fase ini terbagi menjadi 2 sub-fase yaitu sub-fase *business* dan sub-fase *system*. Pada sub-fase *requirement* ada beberapa tahapan yang pertama menentukan *business actor list* untuk menentukan aktor yang berperan dalam sistem. Setelah itu membuat *business use case list* untuk mengidentifikasi proses bisnis apa saja yang ada dalam proses bisnis pemesanan barang pada PT. Trivia Nusantara. Terdapat 4 proses bisnis yang dituliskan dalam *business use case list*. Lalu pembuatan *business activity diagram* kegiatan yang dilakukan yaitu menggambarkan proses apa saja yang dilakukan dalam melakukan suatu proses bisnis dalam suatu *activity diagram*. Dan pada tahap terakhir membuat *business use case details* menjelaskan tahap-tahap dalam suatu proses bisnis dalam bentuk diskripsi. Terdapat 4 *business use case detail* yang dibuat sesuai dengan jumlah *business use case list*.

Pada sub-fase *system* terdapat beberapa tahapan yang pertama memberikan gambaran *user interfase sketches* sistem yang akan dibangun yang mana menghasilkan 23 *user interfase sketches*. Setelah itu pembuatan *use case list* didasarkan pada kebutuhan fungsional yang didapat dari *project genesis*, *user story* dan analisis proses bisnis, dihasilkan 16 *use case*. Lalu tahap selanjutnya yaitu pembuatan *use case diagram* setelah mendapatkan *use case list* setelah itu menggambarkan peran suatu *use case* dengan aktor. Setelah itu pembuatan *use case detail* yang mana menggambarkan alur interaksi antara sistem dan aktor dalam suatu *use case* dan pembuatan *supplementary requirements* menjelaskan kebutuhan tambahan mengenai sistem yang dibangun terdapat 1 kebutuhan tambahan yaitu *ergonomy*.

3. Pada fase *analysis* dilakukan analisis terhadap hasil pendefinisian kebutuhan fungsional sistem sehingga dihasilkan gambaran perancangan awal untuk Sistem Informasi

Pemesanan Barang PT. Trivia Nusantara. Yang mana terdiri dari pembuatan *analysis class diagram* yang digunakan untuk mendapatkan gambaran awal perancangan kelas yang akan dibangun. Setelah itu pembuatan *analysis attributes list* yaitu menganalisis atribut apa saja yang ada dalam sistem. Lalu tahap selanjutnya yaitu pembuatan *communication diagram* yang berdasarkan pada tahapan sebelumnya yaitu *use case details* dan pembuatan *operation list* yang didapatkan 41 operasi yang digunakan untuk menjalankan suatu *use case* dan telah digambarkan dalam *communication diagram*.

4. Pada fase *design* dilakukan perancangan sistem yang lebih mendalam . Perancangan didasarkan pada hasil yang diperoleh dalam fase *requirements* dan fase *analysis*. Langkah-langkah yang dilakukan untuk melakukan perancangan Sistem Informasi Pemesanan Barang PT. Trivia Nusantara meliputi pembuatan *class diagram* yang mana memperoleh masukan dari hasil analisis *class diagram*, *analisis attribute* dan *communication diagram* sehingga teridentifikasi kelas kelas apa saja yang akan dibangun. Setelah itu pembuatan *sequence diagram* , diagram ini mendapatkan masukan dari *communication diagram* dan pada tahap terakhir yaitu pembuatan *database schema*.
5. Hasil evaluasi sistem menggunakan metode *consistency analysis* terbukti memiliki nilai RCI sebesar 100 % dimana setiap *layer* yang ada pada Sistem Informasi Pemesanan Barang saling berhubungan dan konsisten. Sedangkan hasil evaluasi sistem menggunakan uji *correctness* berupa kategori sesuai (*correctness*) perancangan sistem terhadap kebutuhan. Hal tersebut menjelaskan bahwa setiap kebutuhan yang terdapat pada spesifikasi kebutuhan sudah sesuai atau tepat pada setiap fungsi yang akan dimiliki oleh Sistem Informasi Pemesanan Barang.

6.2 Saran

1. Hasil analisis kebutuhan dan perancangan sistem telah mengakomodir kebutuhan sesuai dengan kebutuhan PT. Trivia Nusantara Surabaya. Dokumen persyaratan dapat digunakan sebagai dasar implementasi sistem.

Dibutuhkan proses pengukuran kualitas perancangan untuk mengetahui seberapa dapat dipahami perancangan sistem pemesanan barang untuk diimplementasikan

DAFTAR PUSTAKA

- Bittner, K., dan Spence, I., 2002. Use case Modeling . [pdf] US: Addison Wesley.
- Burch,J., dan Grudnitski,G., 1986. Information Systems Theory and Practice. New York : John Wiley and Sons.
- Dewi, R. K., 2015. Analisa dan Pemodelan Arsitektur Proses Bisnis Kepegawaian dengan Metode Framework for the Application of System Thingking (Studi kasus bagian kepegawaian UNMER Malang). S1. Universitas Brawijaya
- Jogiyanto, H. M., 2008. Metodologi Penelitian Sistem Informasi. Yogyakarta: Andi Offset.
- Kurniawan, T. A., 2018. Pemodelan Use Case (UML) Evaluasi Terhadap beberapa Kesalahan dalam Praktik. Jurnal Teknologi Informasi dan Ilmu Komputer, p. 10.
- Mili, Ali., Tchier, Farioruz., 2014. Software testing conceps and operations. USA:Wiley
- Nistala,P. Dan Kumari,O.,2013. Validating and Tracking Requirements through a Configuration Structure,[e-journal] 320. Tersedia melalui IEE Digital Library <<http://www.ieee.org>> [diakses 18 Februari 2018].
- Nurhandoko, S.Luluk. Purnama, Bambang., 2012. Sistem Informasi Pengolahan Data Iklan Pada Stasiun Radio PAS FM Pati berbasis Web. FTI UNSA.
- O`Dochtery, M.,2005. Object Oriented Analysis and Design: Understanding System Development with 2.0 Great Britain:Biddles Ltd.
- Pressman, Roger S., 2010. Software Engineering : A Practitioner's Approach. 7th ed. New York:McGraw-Hill.
- Rizky , S., 2011. Konsep Dasar Rekayasa Perangkat Lunak. Jakarta: Prestasi Pusaka.
- Satzinger,J.W.,Jackson, R., dan Burd, S.D., 2007. Systems Analysis and Design in Changing World. Brigham : Young University, University of New Mexico.
- Siahaan, D., 2012. Analisa kebutuhan dalam Rekayasa Perangkat Lunak. Yogyakarta : Andi
- Stair, R. M. & Renolds, G. W.,2010. Principles of Information System. 9th Edition penyunt. Boston : Cengage Learnig.
- Whitten, J. L. dan Bentley, L. D., 2007. Systems Analysis and Design Methods. [e-book] New York: McGraw-Hill. Tersedia di: [Analysis_and_Design_Methods_-Februari 2017\]_7th_Edition?auto=download](https://www.academia.edu/8787830/Whitten_and_Bentley_2007_System_7th_Edition?auto=download) [diakses 13 <https://www.academia.edu/8787830/Whitten_and_Bentley_2007_System_7th_Edition?auto=download>