

Sistem Order Barang Menggunakan Metode Unified Modelling Language (UML)

TUGAS PRAKTIKUM

Disusun oleh:

Dian Anggraini	3311811024
Aslam Mubaroch	3311811032
Samuel Ebenezer	3311801049

Disusun untuk memenuhi salah satu syarat kelulusan
mata kuliah IF 3 A Karyawan
Rekayasa Perangkat Lunak II



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
BATAM
2019**

HALAMAN PENGESAHAN

**Sistem Order Barang Menggunakan
Metode Unified Modelling Language (UML)**

Disusun oleh:

Dian Anggraini	3311811024
Aslam Mubaroch	3311811032
Samuel Ebenezer	3311801049

Batam, 25 Oktober 2019

Disetujui dan disahkan oleh:

Dosen pengajar,

Rina Yulius

NIK/NIP. _____

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311811024

Nama : Dian Anggraini

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri
Batam menyatakan bahwa Tugas Praktikum dengan judul:

Sistem Order Barang Menggunakan Metode Unified Modelling Language (UML)

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuandata
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijinpemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 25 Oktober 2019

Dian Anggraini
NIM. 3311811024

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311811032

Nama : Aslam Mubaroch

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri
Batam menyatakan bahwa Tugas Praktikum dengan judul:

Sistem Order Barang Menggunakan Metode Unified Modelling Language (UML)

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuandata
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijinpemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 25 Oktober 2019

Aslam Mubaroch
NIM. 3311811032

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311801049

Nama : Samuel Ebenezer

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri

Batam menyatakan bahwa Tugas Praktikum dengan judul:

Sistem Order Barang Menggunakan Metode Unified Modelling Language (UML)

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya oranglain
2. tidak melakukan pemalsuandata
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijinpemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Praktikum ini.

Batam, 25 Oktober 2019

Samuel Ebenezer
NIM. 3311801049

ABSTRACT

Sistem Order Barang PT. ABC Menggunakan Metode Unified Modelling Language (UML)

Perusahaan yang maju dan berkembang memerlukan suatu sistem informasi yang cepat tepat dan akurat untuk mendukung kegiatan dalam usahanya . Sumber daya manusia dan teknologi informasi merupakan faktor terpenting dalam menghasilkan suatu sistem informasi. Untuk itulah dibuat Sistem Order Barang pada PT. ABC Perusahaan yang bergerak pada penjualan barang dan jasa, Sistem yang ada pada PT. ABC Masih menggunakan sistem secara manual atau sederhana, Mulai dari pengadaan barang, data barang masuk dan juga data barang keluar, serta pembuatan laporan persediaan. Yang memungkinkan pada saat proses berlangsung terdapat kesalahan pada data pemesanan barang, pengadaan barang hingga pencatatan pada laporan persediaan barang. Kurang akuratnya laporan persediaan serta data barang dan pengadaanya membuat permasalahan pada perusahaan. Perancangan sistem informasi ini yang di buat penulis dapat membuat perusahaan menjadi terbantu dengan efektif dan efisien dalam menunjang kelancarnya perusahaan ini. Pembuatan Sistem Order Barang menggunakan bahasa pemodelan UML (Unified Modelling Language).

Kata Kunci : Perancangan Sistem Informasi, Sistem Order Baran

Bab I

Pendahuluan

1.1 Latar Belakang

Perdagangan tidak lagi harus memajang barang-barang yang hendak ditawarkan kepada konsumen didalam lemari dan juga dalam sebuah toko. Semakin berkembangnya zaman, perdagangan bisa terjadi dimana saja dan kapan saja. Contohnya dengan adanya perdagangan dunia elektronik bisa menjadi alternative berdagang yang bisa meningkatkan pendapatan dan keuntungan serta bisa memangkas biaya-biaya operasional lainnya. Konsumen juga bisa mendapatkan efisiensi waktu dan lebih fleksibel dalam berbelanja dan bertransaksi. Perdagangan secara elektronik ini memberikan rasa aman secara fisik karena konsumen tidak perlu lagi mendatangi langsung perusahaan atau toko untuk berbelanja dan ini memungkinkan konsumen bertransaksi secara aman sebab mungkin sebagian daerah sangat rawan jika berkendara dan membawa uang tunai dalam jumlah yang cukup banyak. Perdagangan elektronik akan memacu orang-orang gaptek (gagap teknologi) untuk mempelajari teknologi sistem informasi ini demi kepentingan mereka sendiri, sehingga hal ini dapat meningkatkan jumlah kualitas sumber daya manusia.

PT. ABC merupakan perusahaan yang bergerak pada penjualan barang dan jasa, sistem proses pemesanan barang yang ada pada PT. ABC masih menggunakan sistem secara manual atau sederhana. Adanya sistem pemesanan barang melalui jalur online telah terbukti banyak membantu, baik itu untuk para penyedia jasa dan barang maupun bagi para konsumen dan costumer. Oleh karena itu, untuk mengefisienkan system pemesanan dan penjualan barang di PT. ABC ini dibutuhkan suatu sistem terkomputerisasi yang mendukung untuk pemesanan dan penjualan barang secara elektronik (*e-commerce*).

Meskipun demikian, proses pemesanan tetap dapat berlangsung dengan baik dan jelas layaknya system pemesanan barang secara manual. Perancangan system penjualan ini pun diharapkan dapat membantu pihak Perusahaan untuk melakukan promosi dan penjualan barang hingga jangkauan wilayah yang lebih luas.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah di atas, maka dapat dirumuskan pokok permasalahannya adalah sebagai berikut:

1. Bagaimana cara kerja aplikasi order barang tersebut;
2. Bagaimana cara aplikasi order barang menawarkan kemudahan bagi pengguna; dan
3. Apa yang menjadi keunggulan aplikasi tersebut dibandingkan dengan aplikasi serupa.

1.3 Batasan Masalah

Aplikasi Sistem Order Barang ini dibatasi oleh beberapa hal , yaitu :

1. Sistem aplikasi hanya berjalan dengan berbasis berupa website;
2. Administrator dapat menambah dan menghapus list barang yang dapat dibeli pengguna; dan
3. User hanya dapat berinteraksi dengan aplikasi secara quest tanpa akun.

1.4 Maksud dan Tujuan

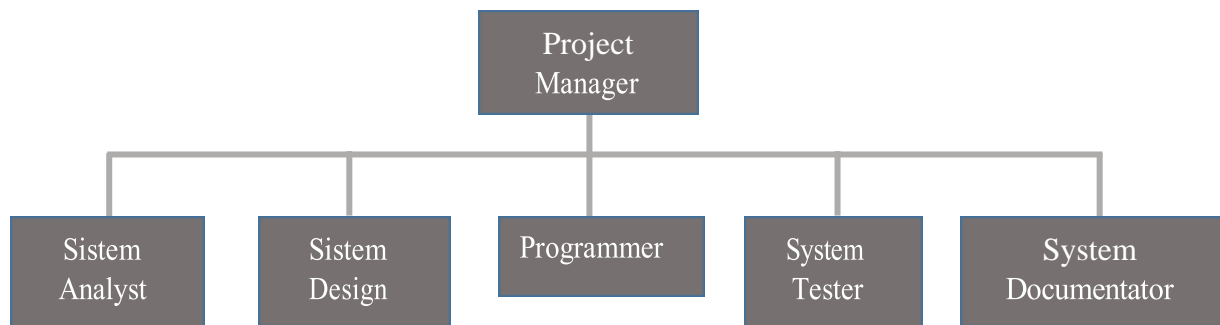
Aplikasi Order Barang ini memiliki Maksud dan Tujuan, yaitu :

1. Memperkenalkan kepada customer cara berbelanja yang nyaman.

2. Memberikan kemudahan kepada penjual untuk memperoleh untung yang lebih besar.

1.5 Struktur Organisasi

a. Struktur Organisasi



b. Peran dan Tanggung jawab

FUNGSI	NAMA	PERAN
Project Manager		<ul style="list-style-type: none"> - Menjadwalkan pelaksanaan dan manajemen proyek. - Memantau kinerja proyek pelaksanaan dari analisis sampai implementasi. - Membuat dokumen SDPLN yang mendefinisikan rencana proyek.
System Analyst		<ul style="list-style-type: none"> - Menganalisa proses bisnis dalam koperasi. - Mendefinisikan prosedur yang ada dalam sistem. - Membuat dokumen flow, sistem flow.

		- Membuat dokumen SRS yang mendefinisikan spesifikasi kebutuhan perangkat lunak.
System Design		<ul style="list-style-type: none"> - Membuat rancangan sistem dengan HIPO dan DFD. - Membuat basis data dan ERD (Entity Relational Diagram). - Membuat dokumen SAD yang mendefinisikan arsitektur sistem.
Programmer		Membuat aplikasi yang telah dirancang dan direncanakan.
System Testing		Membuat test plan untuk implementasi sistem.
System Documentation		<ul style="list-style-type: none"> - Control keselarasan dan kelengkapan seluruh dokumen. - Membuat user guide software sistem informasi.

BAB II

Landasan Teori

2.1 Pengertian UML

UML adalah salah standar bahasa yang banyak di gunakan di dunia industry untuk mendefinisikan requirement, membuat analisis dan desain , serta menggambaran arsitektur dalam pemograman berorientasi objek. UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu developer ke developer lainnya. Tidak hanya antar developer terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya UML.

Tujuan atau fungsi dari penggunaan UML diantaranya:

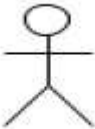
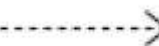
- Dapat memberikan bahasa permodelan visual kepada pengguna dari berbagai macam pemerograman maupun proses rekayasa.
- Dapat menyatukan praktek-praktek terbaik yang ada dalam permodelan.
- Dapat memberikan model yang siap untuk digunakan, merupakan bahasa permodelan visual yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah.


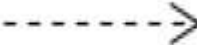





- Dapat berguna sebagai blue print, sebab sangat lengkap dan detail dalam perancangannya yang nantinya akan diketahui informasi yang detail mengenai koding suatu program.
- Dapat memodelkan sistem yang berkonsep berorientasi objek, jadi tidak hanya digunakan untuk memodelkan perangkat lunak (*software*) saja.
- Dapat menciptakan suatu bahasa permodelan yang nantinya dapat dipergunakan oleh manusia maupun oleh mesin.

2.2 Diagram-Diagram UML

2.2.1 Use Case Diagram






Use Case Diagram menggambarkan sejumlah external actors dan hubungannya ke use case yang diberikan oleh sistem. Use case adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari use case symbol namun dapat juga dilakukan dalam activity diagrams. Use case digambarkan hanya yang dilihat dari luar oleh actor (keadaan lingkungan sistem yang dilihat user) dan bukan bagaimana fungsi yang ada di dalam sistem.

No	Nama	Simbol	Fungsi
1	Actor		Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
2	Dependency		hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri(<i>independet</i>)

3	Generalization		hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
4	Include		menspesifikasikan bahwa use case sumber secara eksplisit.
5	Association		Apa yang menghubungkan antara objek satu dengan objek lainnya.
6	System		menspesifikasikan paket yang menampilkan sistem secara terbatas
7	Use Case		deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
8	Collaboration		Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi)
9	Note		Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.


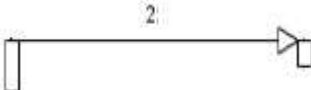
2.2.2 Activity Diagram

Activity Diagram menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

No	Nama	Simbol	Fungsi
1	Activity		memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2	Action		state dari sistem yang mencerminkan eksekusi dari suatu aksi
3	Initial Node		bagaimana objek dibentuk atau diawali
4	Final Node		bagaimana objek dibentuk dan dihancurkan
5	Fork Node		satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran




2.2.3 Sequence Diagram




Sequence Diagram menggambarkan kolaborasi dinamis antara sejumlah object. Kegunaanya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

No	Nama	Simbol	Fungsi
1	Life Line,Objek Entity		antarmuka yang saling berinteraksi
2	Message		spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

2.2.4 Class Diagram

Class Diagram menggambarkan struktur statis class di dalam sistem. Class merepresentasikan sesuatu yang ditangani oleh sistem. Class dapat berhubungan dengan yang lain melalui berbagai cara: associated (terhubung satu sama lain), dependent (satu class tergantung/menggunakan class yang lain), specialied (satu class merupakan spesialisasi dari class lainnya), atau package (grup bersama sebagai satu unit). Sebuah sistem biasanya mempunyai beberapa class diagram.

No	Nama	Simbol	Fungsi
1	Generalization		hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
2	Class		himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
3	Collaboration		deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

4	Realization		operasi yang benar-benar dilakukan oleh suatu objek
5	Dependecy		hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempegaruhi elemen yang bergantung padanya
6	Association		apa yang menghubungkan antara objek

BAB III

ANALISIS DAN PERANCANGAN

3.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja atau layanan apa saja yang nantinya harus disediakan oleh sistem, mencakup bagaimana sistem harus bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu.

Melakukan Input Pendataan Produk yang dijual :

- Administrator memasukkan tentang data-data produk yang dijual meliputi : nama produk, jumlah stok, harga, deskripsi produk ,dll.
- Pengguna dapat memperoleh informasi secara detail mengenai barang yang akan di beli.
- Pengguna dapat mengirim dan menerima pesan dari seller/pemilik toko.
- Pengguna dapat melaporkan kesalahan atau bantuan oleh sistem administrator.

Melakukan Input Pendataan Transaksi Pembelian :

- Sistem dapat melakukan transaksi pembelian.
- Sistem dapat menampilkan barang apa yang akan dibeli oleh pembeli.
- Sistem dapat melakukan perhitungan berapa jumlah barang yang dibeli.
- Sistem dapat melakukan pendataan pembeli.

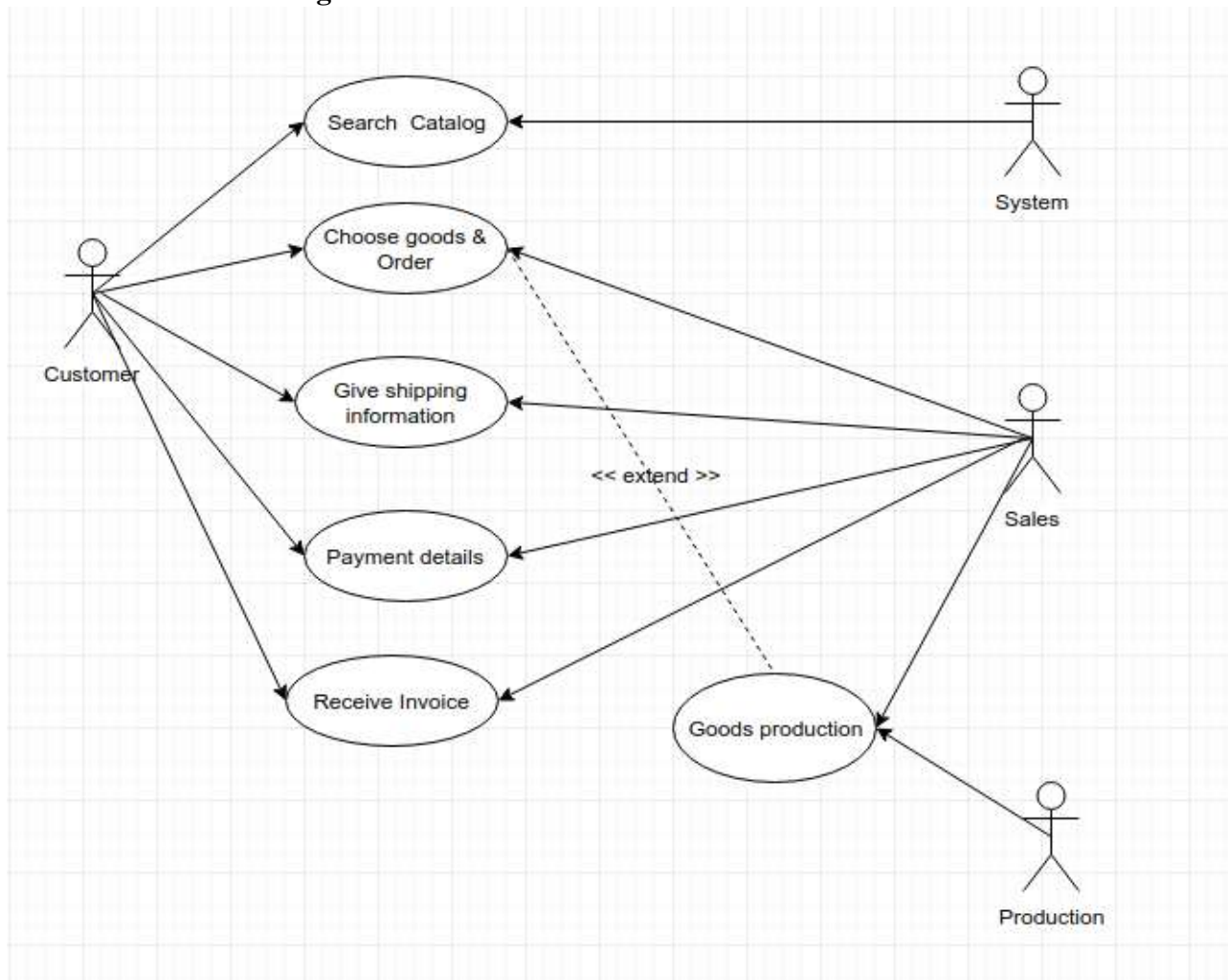
3.2 Kebutuhan Non Fungsional

Kebutuhan nonfungsional merupakan kebutuhan yang berisi properti yang perilaku yang dimiliki oleh sistem. Kebutuhan nonfungsional ini meliputi antara lain:

1. Operasional
Operasional ini menjelaskan secara teknis tentang bagaimana sistem baru akan bekerja.
2. Keamanan (*Security*)
Sistem aplikasi dan database dilengkapi dengan kata sandi (*Password*).
3. Informasi
- Digunakan untuk memantau pengguna dengan hak aksesnya.

- Digunakan untuk menginformasikan pengguna apabila kata sandi (*Password*) yang dimasukkan salah.

3.3 Use Case Diagram



Gambar 3.3 use case diagram sistem aplikasi oder barang

3.4 Skenario use case

3.4.1 Skenario usecase pemesanan barang

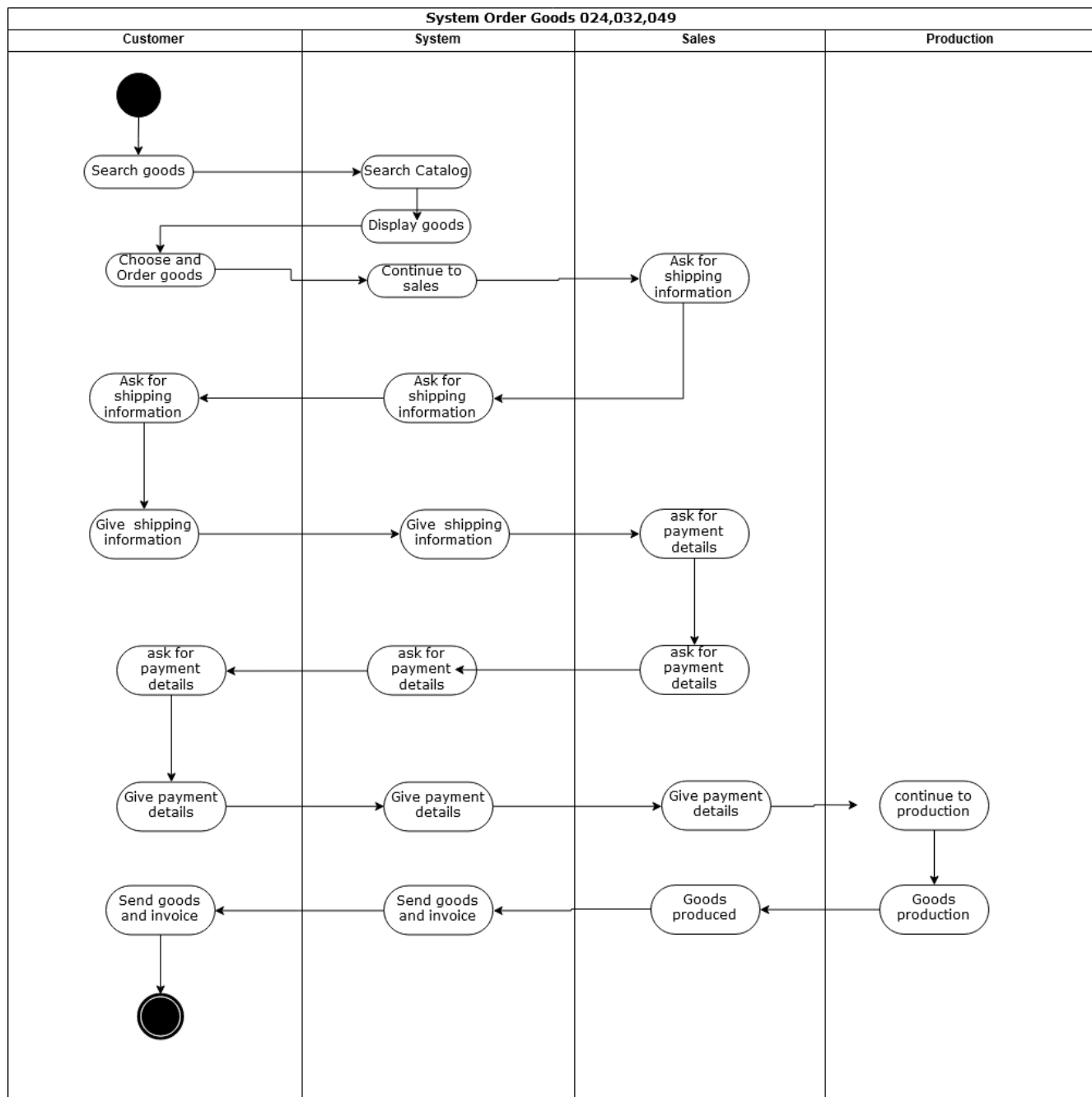
Aksi aktor	Reaksi sistem
1. Customer membuka menu barang	
	2. System menampilkan data barang
3. Customer memilih barang	
4. Customer memasukkan data pemesanan	
5. Customer menekan tombol “simpan”	
	6. System mengecek stok barang
	7. Jika stok barang tersedia, maka data pesanan disimpan dan menampilkan pesan :data pemesanan berhasil disimpan.
	8. Jika stok barang yang dipesan tidak tersedia, maka system akan menampilkan pesan “mohon maaf, stok tidak tersedia”
9. Customer menekan tombol “Pesan Sekarang”	
	10. System mengirim notifikasi pemesanan barang ke sales
	11. System menyimpan data pemesanan dan menampilkan pesan “data pemesanan berhasil disimpan”

3.4.2 Skenario usecase pembayaran

Aksi aktor	Reaksi sistem
1. Customer membuka form rincian biaya	
	2. System menampilkan rincian biaya dari transaksi yang dilakukan
3. Customer memilih metode pembayaran	
	4. System menampilkan metode pembayaran (cash, kredit, transfer bank/indomaret)
5. Customer melakukan pembayaran	
6. Customer mengklik “OK” dari pembayaran yang dilakukan	

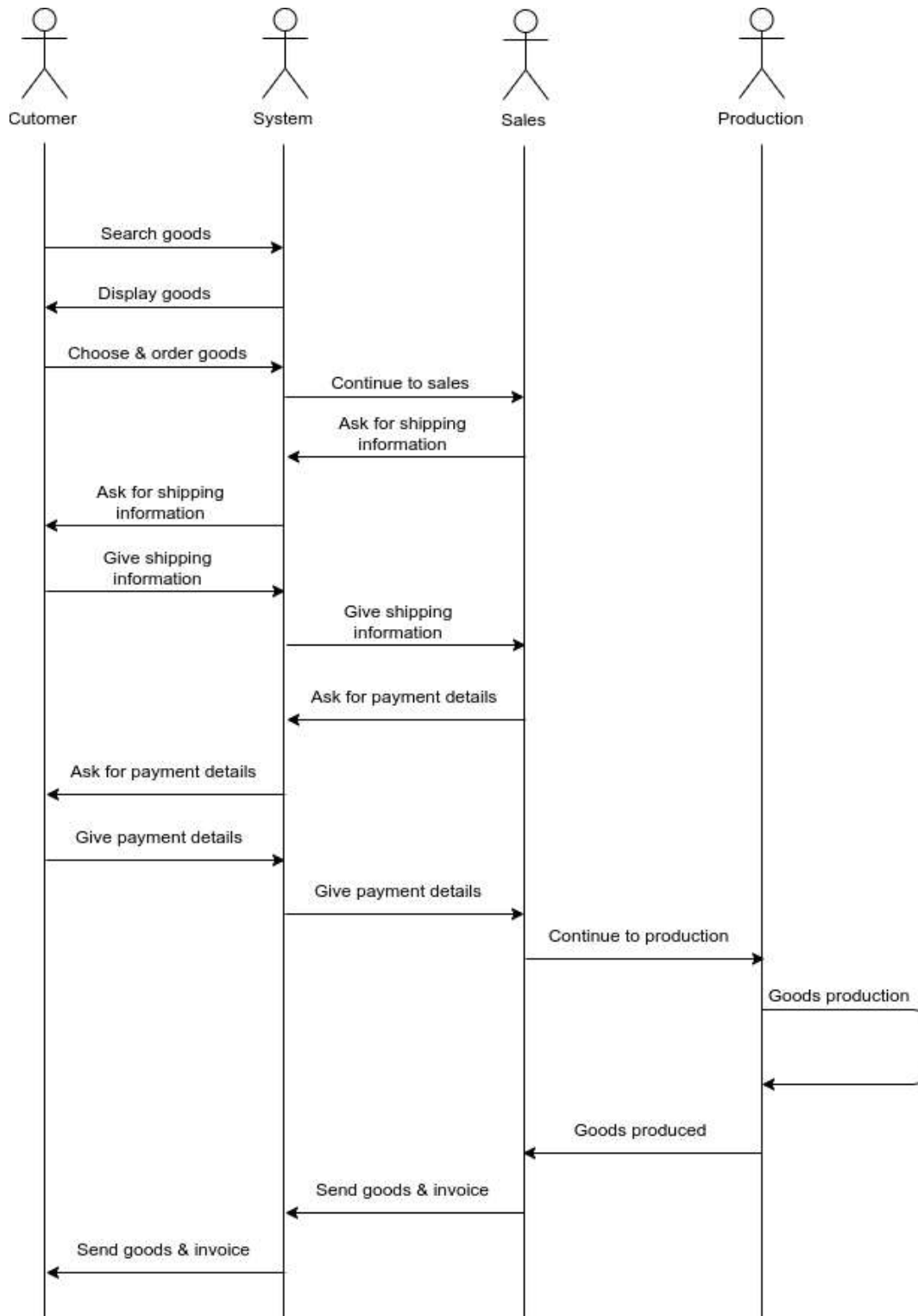
3.5 Activity Diagram

Berikut activity diagram pemesanan barang :



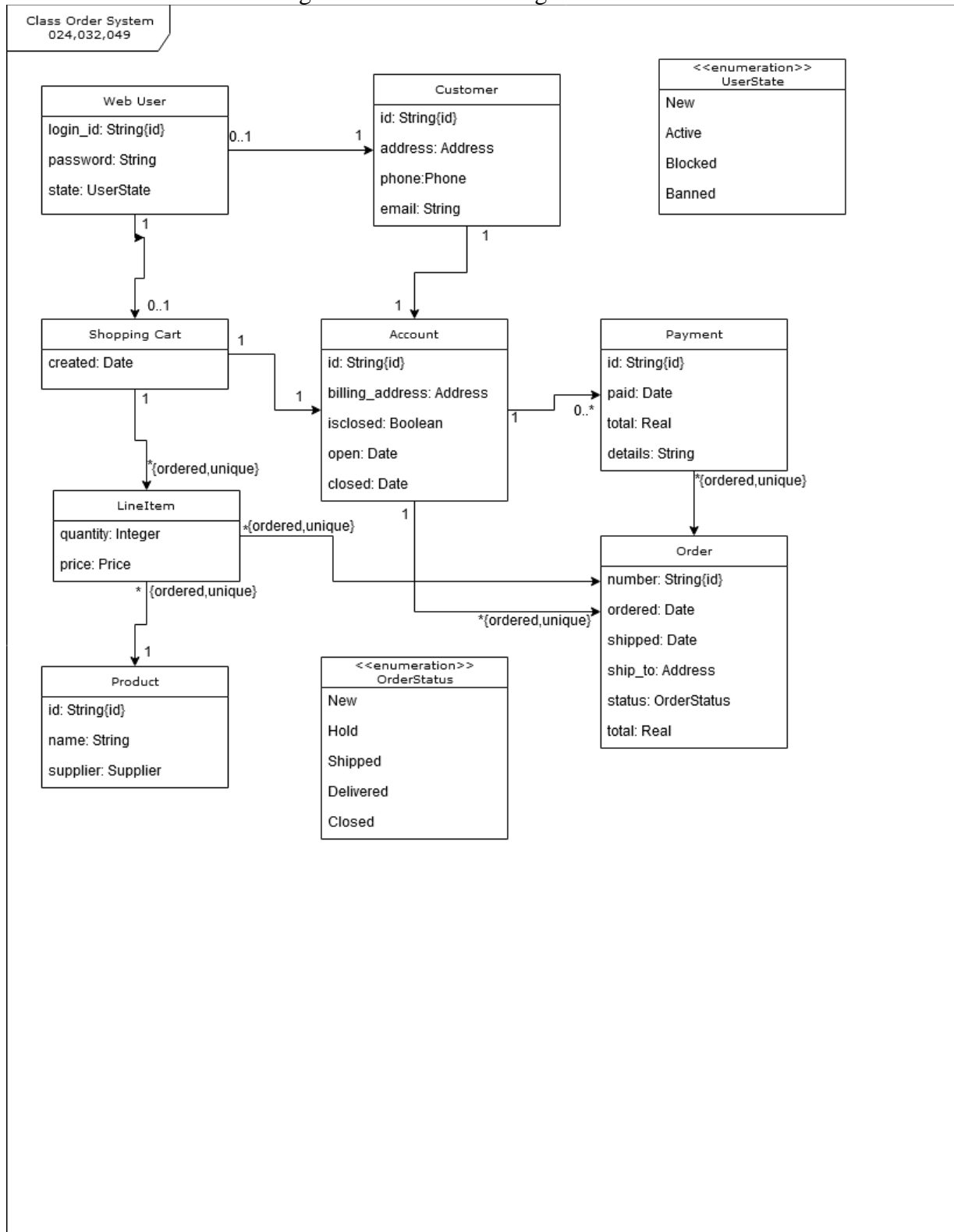
3.6 Sequence Diagram

Berikut sequence diagram perancangan pemesanan barang :



3.7 Class Diagram

Berikut Class Diagram Pemesanan Barang :



Timeline Jadwal

Daftar Kegiatan		Number	Predecessor	Start	Stop	Waktu Pelaksanaan																														
						Januari																														
						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Analisis Sistem	Melakukan wawancara dan rapat koordinasi kepada pihak client	2		2/1/2017	3/1/2017																															
	Membuat Statement of Purpose, Event List dan Context Diagram	3	2	4/1/2017	6/1/2017																															
	Membuat Usecase Diagram	4	3	9/1/2017	9/1/2017																															
	Membuat State Diagram	5	4	10/1/2017	10/1/2017																															
	Membuat Data Flow Diagram	6	5	11/1/2017	11/1/2017																															
	Membuat Kamus Data	7	6	12/1/2017	12/1/2017																															
	Membuat Proses Spesification	8	7	13/1/2017	13/1/2017																															
	Membuat Desain Database	9	8	16/1/2017	16/1/2017																															
	Dokumentasi Kebutuhan dan Desain Sistem (SKPL)	10	3	9/1/2017	18/1/2017																															
Desain Aplikasi	Membuat Desain Menu Aplikasi dan Form Entry	12	8	16/1/2017	17/1/2017																															
	Membuat Desain Report dan Form Report	13	12	18/1/2017	19/1/2017																															
	Dokumentasi Desain Aplikasi (DPPL)	14	8	16/1/2017	23/1/2017																															
Implementasi	Membuat Program Aplikasi	16	14	24/1/2017	13/2/2017																															
	Dokumentasi Program Aplikasi	17	16	24/1/2017	13/2/2017																															
Testing	Membuat List Testing Program	19	17	14/2/2017	15/2/2017																															
	Melakukan Testing Program	20	19	16/2/2017	17/2/2017																															
	Membuat Catatan Perbaikan	21	19	16/2/2017	16/2/2017																															
	Dokumentasi Testing Program	22	17	14/2/2017	17/2/2017																															
Instalasi	Melakukan Setting Infrastruktur dan Software Pendukung	24	18	20/2/2017	24/2/2017																															
	Instalasi Program Aplikasi	25	24	27/2/2017	28/2/2017																															
	Membuat Dokumentasi User Guide	26		1/3/2017	3/3/2017																															
	Melakukan Pelatihan untuk User	27	26	6/3/2017	7/3/2017																															
Maintenance	Pemeliharaan Aplikasi	29	23	8/3/2017	24/3/2017																															

Daftar Kegiatan		Number	Predecessor	Start	Stop	Waktu Pelaksanaan																											
						Februari																											
						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Analisis Sistem	Melakukan wawancara dan rapat koordinasi kepada pihak client	2																															
	Membuat Statement of Purpose, Event List dan Context Diagram	3	2																														
	Membuat Usecase Diagram	4	3																														
	Membuat State Diagram	5	4																														
	Membuat Data Flow Diagram	6	5																														
	Membuat Kamus Data	7	6																														
	Membuat Proses Spesification	8	7																														
	Membuat Desain Database	9	8																														
	Dokumentasi Kebutuhan dan Desain Sistem (SKPL)	10	3																														
Desain Aplikasi	Membuat Desain Menu Aplikasi dan Form Entry	12	8																														
	Membuat Desain Report dan Form Report	13	12																														
	Dokumentasi Desain Aplikasi (DPPL)	14	8																														
Implementasi	Membuat Program Aplikasi	16	14																														
	Dokumentasi Program Aplikasi	17	16																														
Testing	Membuat List Testing Program	19	17																														
	Melakukan Testing Program	20	19																														
	Membuat Catatan Perbaikan	21	19																														
	Dokumentasi Testing Program	22	17																														
Instalasi	Melakukan Setting Infrastruktur dan Software Pendukung	24	18																														
	Instalasi Program Aplikasi	25	24																														
	Membuat Dokumentasi User Guide	26																															
	Melakukan Pelatihan untuk User	27	26																														
Maintenance	Pemeliharaan Aplikasi	29	23																														

Daftar Kegiatan		Number	Predecessor	Start	Stop	Waktu Pelaksanaan																							
						Maret																							
						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Analisis Sistem	Melakukan wawancara dan rapat koordinasi kepada pihak client	2		2/1/2017	3/1/2017																								
	Membuat Statement of Purpose, Event List dan Context Diagram	3	2	4/1/2017	6/1/2017																								
	Membuat Usecase Diagram	4	3	9/1/2017	9/1/2017																								
	Membuat State Diagram	5	4	10/1/2017	10/1/2017																								
	Membuat Data Flow Diagram	6	5	11/1/2017	11/1/2017																								
	Membuat Kamus Data	7	6	12/1/2017	12/1/2017																								
	Membuat Proses Spesification	8	7	13/1/2017	13/1/2017																								
	Membuat Desain Database	9	8	16/1/2017	16/1/2017																								
	Dokumentasi Kebutuhan dan Desain Sistem (SKPL)	10	3	9/1/2017	18/1/2017																								
Desain Aplikasi	Membuat Desain Menu Aplikasi dan Form Entry	12	8	16/1/2017	17/1/2017																								
	Membuat Desain Report dan Form Report	13	12	18/1/2017	19/1/2017																								
	Dokumentasi Desain Aplikasi (DPPL)	14	8	16/1/2017	23/1/2017																								
Implementasi	Membuat Program Aplikasi	16	14	24/1/2017	13/2/2017																								
	Dokumentasi Program Aplikasi	17	16	24/1/2017	13/2/2017																								
Testing	Membuat List Testing Program	19	17	14/2/2017	15/2/2017																								
	Melakukan Testing Program	20	19	16/2/2017	17/2/2017																								
	Membuat Catatan Perbaikan	21	19	16/2/2017	16/2/2017																								
	Dokumentasi Testing Program	22	17	14/2/2017	17/2/2017																								
Instalasi	Melakukan Setting Infrastruktur dan Software Pendukung	24	18	20/2/2017	24/2/2017																								
	Instalasi Program Aplikasi	25	24	27/2/2017	28/2/2017																								
	Membuat Dokumentasi User Guide	26		1/3/2017	3/3/2017																								
	Melakukan Pelatihan untuk User	27	26	6/3/2017	7/3/2017																								
Maintenance	Pemeliharaan Aplikasi	29	23	8/3/2017	24/3/2017																								

Rencana Biaya

Berikut merupakan tabel pemetaan dana pada setiap aktifitas.

DAFTAR	AKTIVITAS	DURASI KERJA (JAM)	TARIF TENAGA KERJA	JUMLAH TENAGA KERJA	BIAYA TENAGA KERJA	TOTAL BIAYA
Analisis Sistem	Melakukan wawancara dan rapat koordinasi kepada pihak client	16	100.000		100.000	
	Membuat Statement of Purpose, Event List dan Context Diagram	16	200.000		200.000	
	Membuat Usecase Diagram	8	-		-	
	Membuat State Diagram	8	-	1	-	5.300.000
	Membuat Data Flow Diagram	8	-		-	
	Membuat Kamus Data	8	-		-	
	Membuat Proses Spesification	8	-		-	
	Membuat Desain Database	8	-		-	
	Dokumentasi Kebutuhan dan Desain Sistem (SKPL)	64	5.000.000		5.000.000	
	Membuat Desain Menu	16	500.000	1	500.000	6.000.000
Desain Aplikasi						

	Aplikasi dan Form Entry					
	Membuat Desain Report dan Form Report	16	500.000		500.000	
	Dokumentasi Desain Aplikasi (DPPL)	48	5.000.000		5.000.000	
Implementasi	Membuat Program Aplikasi	120	20.000.000	2	40.000.000	50.000.000
	Dokumentasi Program Aplikasi	120	5.000.000		10.000.000	
Testing	Membuat List Testing Program	16	2.000.000		2.000.000	
	Melakukan Testing Program	16	5.000.000	1	5.000.000	13.000.000
	Membuat Catatan Perbaikan	16	1.000.000		1.000.000	
	Dokumentasi Testing Program	32	5.000.000		5.000.000	
	Melakukan Setting Infrastruktur dan Software Pendukung	40	2.000.000		2.000.000	
Instalasi	Instalasi Program Aplikasi	16	500.000	1	500.000	9.000.000
	Membuat Dokumentasi User Guide	24	5.000.000		5.000.000	

	Melakukan Pelatihan untuk User	16	1.500.000		1.500.000	
Main-tenance	Pemeliharaan Aplikasi	104	10.000.000	2	20.000.000	20.000.000
Lain-Lain	Sewa Server	-	-	-	1.000.000	1.000.000
Perencanaan Total		744	69.300.000	8	104.300.000	104.300.000

c. Monitoring dan Kontrol Proyek

1. Rencana Manajemen Kebutuhan

Dalam perencanaan proyek ini membutuhkan berbagai macam dokumen-dokumen pendukung untuk kebutuhan sistem.

2. Rencana Kontrol Schedule

Penjelasan dari bagian ini yaitu setiap melakukan progres atau pengerjaan proyek ini masing-masing pekerja meminta tanda tangan sebagai tanda bukti progresnya sudah selesai dan diterima.

3. Rencana Kontrol Budget

Pengontrolan terhadap budget akan dilakukan dengan menggunakan laporan-laporan tentang bagaimana penggunaan dana tersebut oleh setiap bagian tim.

4. Rencana Kontrol Kualitas

Kualitas dari setiap software diperlihatkan dari setiap laporan-laporan yang ada serta pada bagian akhir (feedback dan revisi), bila dirasa kualitas software masih kurang masih dapat dilakukan perbaikan pada software yang ada.

5. Rencana Laporan

Ada beberapa laporan yang hanya didiskusikan dalam internal tim dalam menjalankan proyek. Ada pula laporan-laporan yang harus diketahui oleh perusahaan sebagai bahan evaluasi bersama. Laporan-laporan yang cukup diketahui oleh internal tim adalah draft awal Software Development Plan (SDPLN), draft awal Software Requirements Specification (SRS), draft awal Software Architecture Diagram (SAD), dan draft awal Test Plan. Sedangkan laporan yang harus diberikan pada perusahaan adalah proposal proyek dan Laporan keseluruhan yang berisi SDPLN, SRS, SAD, dan Test Plan.

d. Rencana Manajemen Resiko

Resiko	Kejadian	Teknik Mengurangi Resiko
--------	----------	--------------------------

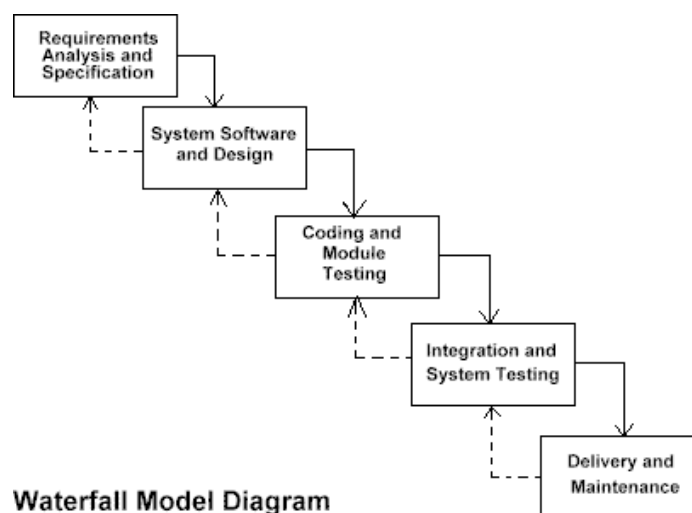
Kegagalan pada personil	<ul style="list-style-type: none"> - Kesalahan dalam coding - Tidak menguasai pada bidangnya - Tim tidak kompak - Terjadi perselisihan dalam tim - Jadwal yang rancu 	<ul style="list-style-type: none"> - Mempekerjakan staf yang handal - Job matching - Membangun tim - Mengadakan pelatihan dan peningkatan karir - Membuat jadwal lebih awal bagi personil utama
Estimasi biaya dan waktu yang tidak realistis	<ul style="list-style-type: none"> - Biaya yang berlebihan - Kekurangan biaya - Kesalahan perhitungan 	<ul style="list-style-type: none"> - Membuat beberapa estimasi - Desain untuk biaya - Meningkatkan pengembangan - Merekam dan menganalisa proyek sebelumnya - Standarisasi metode
Mengembangkan fungsi software yang salah	<ul style="list-style-type: none"> - Kesalahan pemrograman - Kesalahan desain - Analisa kebutuhan yang tidak sesuai 	<ul style="list-style-type: none"> - Evaluasi proyek ditingkatkan - Buat metode spesifikasi yang formal - Survey pengguna - Buat prototype - Buat user manual lebih awal
Mengembangkan antarmuka pengguna yang salah	<ul style="list-style-type: none"> - Pengguna tidak bisa menggunakan aplikasi yang telah di hasilkan 	<ul style="list-style-type: none"> - Membuat prototype - Analisis tugas - Keterlibatan pengguna
Gold plating	-	<ul style="list-style-type: none"> - Mengurangi kebutuhan - Membuat prototype - Analisis biaya manfaat - Desain biaya
Terlambat untuk mengubah kebutuhan	<ul style="list-style-type: none"> - Perubahan mendadak ditengah proyek - Pembatalan kontrak kerja 	<ul style="list-style-type: none"> - Mengubah prosedur kendali - Membatasi perubahan yang terlalu banyak - Meningkatkan prototype - Meningkatkan pengembangan (akibat perubahan)
Kegagalan pada komponen yang disuplai pihak eksternal	<ul style="list-style-type: none"> - Suplai biaya berhenti - Pembatalan kontrak kerja 	<ul style="list-style-type: none"> - Melakukan benchmarking - Inspeksi - Spesifikasi formal - Kontrak perjanjian - Prosedur dan sertifikasi jaminan kualitas
Kegagalan menjalankan tugas eksternal	-	<ul style="list-style-type: none"> - Prosedur jaminan kualitas - Desain / prototype yang kompetitif - Membangun tim - Kontrak insentif

Kegagalan kinerja real-time	<ul style="list-style-type: none"> - Timada yang sakit, sehingga waktu kerja perlu di tambah - Pekerjaan terlalu cepat diselesaikan - Kesalahan estimasi waktu 	<ul style="list-style-type: none"> - Simulasi - Benchmarking - Prototipe - Tuning - Analisis teknis
Pengembangnya terlalu sulit secara teknis	<ul style="list-style-type: none"> - Aplikasi yang di buat sulit di kembangkan lagi 	<ul style="list-style-type: none"> - Analisa teknis - Analisis biaya manfaat - Prototipe - Melatih dan mengembangkan staf

B. Perencanaan Teknis

a. Metode

Metode pengembangan aplikasi pada proyek ini menggunakan Waterfall SDLC. Pendekatan pengembangan aplikasi dimulai pada level sistem dan prosesnya melalui gambar berikut :



Metode ini digunakan karena tergolong mudah jika diterapkan pada proyek dengan kebutuhan user yang stabil (tidak berubah-ubah). Pada akhir setiap tahapan dalam metode ini selalu dihasilkan sebuah dokumen yang akan digunakan sebagai pedoman bagi tahapan selanjutnya. Metode lain yang juga digunakan pada proyek adalah UML. UML merupakan proses rekayasa perangkat lunak. UML menyediakan pendekatan perancangan perangkat lunak yang berorientasi objek melalui gambaran berbagai diagram rancangan perangkat lunak.

b. Infrastruktur

1. Perangkat Keras

Perangkat keras yang dibutuhkan dalam pengembangan sistem dapat dilihat melalui tabel berikut.

No.	Perangkat Keras	
	Nama	Keterangan
1.	Prosesor	Minimal Intel Core i5
2.	Memori	Minimal 2GB DDR3
3.	<i>Hard Drive</i>	Minimal 500GB Serial ATA (7200 RPM)
4.	<i>Network</i>	<i>Gigabit Network</i>
5.	<i>Optical Drive</i>	Tipe DVD RW
6.	<i>Monitor</i>	Minimal 15"
7.	<i>Keyboard</i>	Tipe <i>USB keyboard</i>
8.	<i>Mouse</i>	Tipe <i>USB optical mouse</i>

2. Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam pengembangan sistem dapat dilihat melalui tabel berikut.

No.	Perangkat Lunak	
	Nama	Keterangan
1.	Sistem operasi	Menggunakan sistem operasi windows 7/8/10
2.	Bahasa pemrograman PHP, HTML, MySql	Untuk pembuatan aplikasi sistem
3.	IDE Php Storm	Untuk editor pembuatan aplikasi sistem
4.	MySQL Server	Untuk <i>database server</i>
5.	XAMPP	Untuk <i>web server</i>
6.	Mozilla firefox/ Chrome	Untuk <i>web browser</i>
7.	Microsoft word	Untuk pembuatan <i>form</i>
8.	Microsoft excel	Untuk pembuatan data input

C. Perencanaan Pendukung

a. Rencana Manajemen Konfigurasi

Dalam pelaksanaan proyek semua hal yang terkait dapat dikatakan sebagai item konfigurasi. Semua item konfigurasi harus dikelola sebaik mungkin untuk menciptakan efisiensi dan efektivitas dalam pembangunan perangkat lunak. Item konfigurasi dan file-file dokumentasi akan dibuat dengan penamaan sesuai dengan standart dan berdasarkan versinya. Penentuan versi berdasarkan

urutan hasil perubahan atau evaluasi dari setiap dokumen yang telah dibentuk. Penamaan dari item konfigurasi dan file-file dokumentasi akan disesuaikan dengan dokumen SRS (Software Requirement Specification) dan dokumen SDD (Software Desain Dokumentation). Dokumen versi terakhir yang telah ditinjau oleh manager proyek akan didistribusikan kepada seluruh anggota proyek untuk dijadikan sebagai pedoman dalam pembangunan perangkat lunak.

b. Rencana Evaluasi

Perencanaan evaluasi akan dilakukan dan dikembangkan pada pembangunan perangkat lunak. Evaluasi akan dilakukan pada setiap tahap proses pembangunan perangkat lunak, mulai dari tahap perancangan sampai dengan tahap akhir implementasi perangkat lunak. Setiap hasil evaluasi akan didokumentasikan untuk menunjang Perencanaan Pengembangan Perangkat Lunak.

c. Rencana Dokumentasi

Rencana Dokumentasi dibentuk sesuai dengan standart yang telah disepakati sehingga hasil dokumentasi dapat digunakan sesuai dengan tujuannya. Dokumentasi dilakukan pada setiap aktivitas dalam pembangunan perangkat lunak.

d. Rencana Penjaminan Mutu

Rencana Jaminan Kualitas akan dibentuk sesuai dengan kesepakatan antara pihak pengembang dengan pihak Stakeholder. Rencana jaminan kualitas dapat dikembangkan pada saat berjalannya proses pembangunan perangkat lunak.

e. Rencana Penyelesaian Konflik

Rencana Resolusi Permasalahan merupakan rencana yang harus dibentuk untuk mencegah dan menanggulagi kegagalan pada sistem. Rencana resolusi permasalahan akan dibicarakan secara informal antara pihak pengembang dan pihak Stakeholder.

f. Rencana Tambahan

Rencana tambahan akan dilaksanakan dan dirancang selama proyek dikerjakan sesuai dengan kondisi dan situasi pada saat itu jika ada rencana yang tidak sesuai.

BAB IV

Kesimpulan dan Saran

4.1 Kesimpulan dan Saran

Kesimpulan

UML adalah metode pemodelan (tools/model) secara visual yang memudahkan kita untuk merancang dan membuat software berorientasi objek dan memberikan standar penulisan sebuah system untuk pengembangan sebuah software yang dapat menyampaikan beberapa informasi untuk proses implementasi pengembangan software.\

Saran

Untuk dapat membuat UML yang sesuai kebutuhan diperlukan pemahaman tentang konsep bahasa pemodelan dan tiga elemen utama UML yaitu :

- Benda/Things/Objek
- Hubungan/Relationship
- Bagan atau Diagrams