# 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 matakuliah IF312 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 Praktikumini.

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 Praktikumini.

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 Praktikumini.

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 mengunakan 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 kelancaranya perusahaan ini.Pembuatan Sistem Order Barang menggunakan bahasa pemodelan UML (Unified Modelling Language).

Kata Kunci : Perancangan Sistem Informasi, Sistem Order Barang

## DAFTAR ISI

[HALAMAN JUDUL 1](#_Toc24471481)

[HALAMAN PENGESAHAN 2](#_Toc24471481)

[HALAMAN PERNYATAAN 3](#_Toc24471482)

[HALAMAN PERNYATAAN 4](#_Toc24471483)

[HALAMAN PERNYATAAN 5](#_Toc24471484)

[Abstract 6](#_Toc24471485)

[Daftar Isi 7](#_Toc24471486)

[BAB I 8](#_Toc24471487)

[PENDAHULUAN 8](#_Toc24471488)

[1.1 Latar Belakang 8](#_Toc24471489)

[1.2 RumusanMasalah 9](#_Toc24471490)

[1.3 Batasan Masalah 9](#_Toc24471491)

[1.4 Maksud danTujuan 9](#_Toc24471492)

[BAB II 10](#_Toc24471495)

[LANDASAN TEORI 10](#_Toc24471496)

[2.1 Pengertian UML 10](#_Toc24471497)

[2.2 Jenis-jenis diagram UML 11](#_Toc24471498)

[2.2.1 Use case 11](#_Toc24471499)

[2.2.2 Activity Diagram 12](#_Toc24471500)

[2.2.3 Sequence diagram 12](#_Toc24471501)

[2.2.4 Class diagram 13](#_Toc24471501)

[BAB III 14](#_Toc24471502)

[ANALISIS DAN PERANCANGAN 14](#_Toc24471503)

[3.1 Kebutuhan Fungsional 14](#_Toc24471505)

[3.2 Kebutuhan NonFungsional 14](#_Toc24471506)

[3.3 Use case 15](#_Toc24471509)

[3.4 Skenario use case 16](#_Toc24471510)

[3.4.1 Skenario usecase pemesanan barang 16](#_Toc24471511)

[3.4.2 skenario usecase pembayaran 16](#_Toc24471512)

[3.5 Activity diagram 17](#_Toc24471513)

[3.6 Sequence diagram 18](#_Toc24471514)

[3.7 Class diagram 19](#_Toc24471514)

BAB IV…………………………………………………………………………………………20

KESIMPULAN DAN SARAN…………………………………………………………………20

4.1 Kesimpulan dan Saran……………………………………………………………………...20

## Bab I

## Pendahuluan

## 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 perdangangan 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 mengunakan 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 mendukunguntuk pemesanan dan penjualan barang secara elektronik (*e-commerce*).

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

## RumusanMasalah

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.

## 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.

## 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 Str uktur Organisasi

* 1. Str uktur Organisasi



Project

Manager

Sistem

Analyst

Sistem

Design

Programmer

System

Tester

System

Documentator

* 1. Peran dan Tanggung jawab

|  |  |
| --- | --- |
| **Nama** | **Peran dan Tanggung jawab** |
| **Project Manager** | * **Menjadwalkan pelaksanaan dan manajemen proyek.** * **Memantau kinerja proyek pelaksanaan dari analisis sampai implementasi.** |
| **System Analyst** | * **Mendefinisikan prosedur yang ada dalam sistem.** * **Membuat dokumen flow, sistem flow.** * **Membuat dokumen yang mendefinisikan spesifikasi kebutuhan perangkat lunak.** |
| **System Design** | * Membuat rancangan sistem * Membuat dokumen yang   mendefinisikan arsitektur sistem. |
| **Programmer** | * **Membuat aplikasi yang telah dirancang dan direncanakan.** |
| **System Testing** | * **Membuat test plan untuk implementasi sistem.** |
| **System Documentation** | * **Control kelengkapan seluruh dokumen.** * **Membuat user guide software sistem informasi.** |

|  |  |  |
| --- | --- | --- |
|  |  | - |
|  |  | * dengan HIPO dan DFD. * Membuat basis data dan ERD (Entity Relational Diagram). |
| Pr og r a mmer  System Testi ng |  |  |
|  |  |
| System  Documenta ti on |  |  |

## 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 mengambaran 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 lainya. 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.

## Diagram-Diagram UML]

* + 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 *use case* |
| 2 | Dependecy |  | hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan memperngaruhi elemen yang bergantung padanya elemen yang tidak mandiri(*independet*) |
| 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 prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi) |
| 9 | Note |  | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

* + 1. **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 |

* + 1. **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 |

* + 1. **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), specialed (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

* 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.

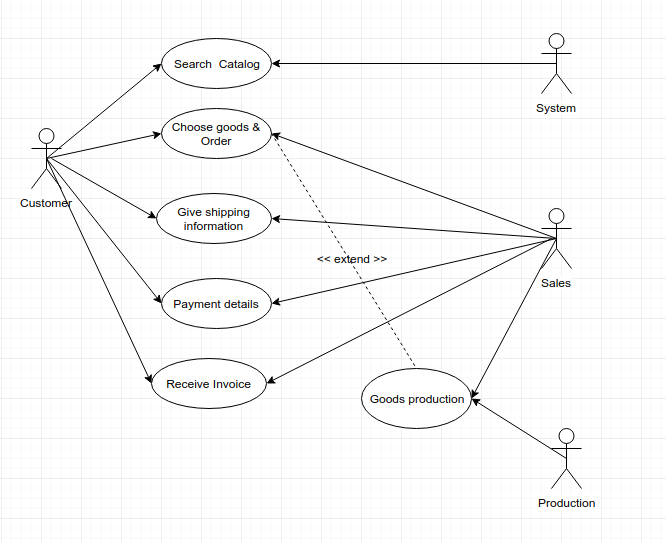
Melakukan Input Pendataan Transaksi Pembelian :

* Sistem dapat melakukan transaksi pembelian.
* Sistem dapat manampilkan barang apa yang akan dibeli oleh pembeli.
* Sistem dapat melakukan perhitungan berapa jumlah barang yang dibeli.
* Sistem dapat melakukan pendataan pembeli.
  1. **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 baruakan 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

## Skenario use case

### **Skenario usecase pemesanan barang**

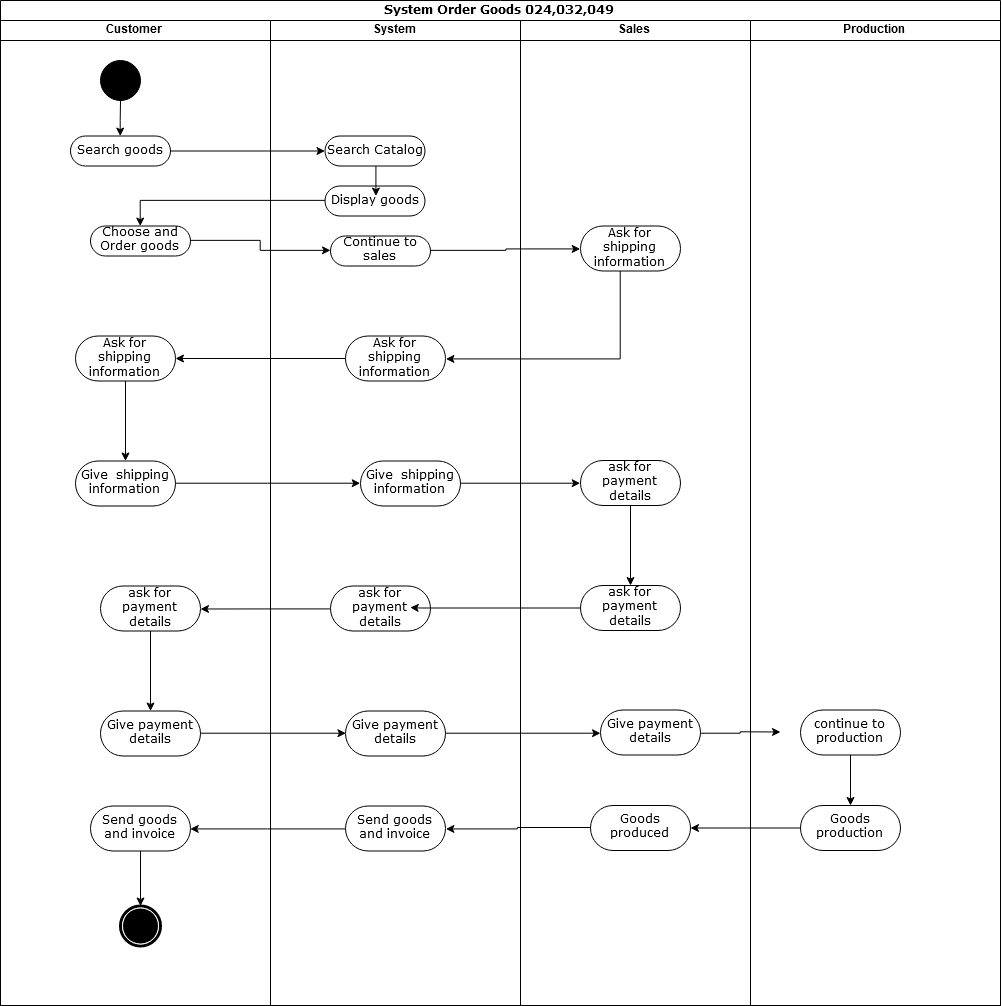
|  |  |
| --- | --- |
| **Aksi aktor** | **Reaksi sistem** |
| 1. Customer membuka menu barang |  |
|  | 1. System menampilkan data barang |
| 1. Customer memilih barang |  |
| 1. Customer memasukkan data pemesanan |  |
| 1. Customer menekan tombol “simpan” |  |
|  | 1. System menegecek stok barang |
|  | 1. Jika stok barang tersedia, maka data pesanan disimpan dan menampilkan pesan :data pemesanan berhasil disimpan. |
|  | 1. Jika stok barang yang dipesan tidak tersedia, maka system akan menampilkan pesan “mohon maaf, stok tidak tersedia” |
| 1. Customer menekan tombol “Pesan Sekarang” |  |
|  | 1. System mengirim notifikasi pemesanan barang ke sales |
|  | 1. System menyimpan data pemesanan dan menampilkan pesan “data pemesanan berhasil disimpan” |

### **Skenario usecase pembayaran**

|  |  |
| --- | --- |
| **Aksi aktor** | **Reaksi sistem** |
| 1. Customer membuka form rincian biaya |  |
|  | 1. System menampilkan rincian biaya dari transaksi yang dilakukan |
| 1. Customer memilih metode pembayaran |  |
|  | 1. System menampilkan metode pambayaran (cash, kredit, transfer bank/indomaret) |
| 1. Customer melakukan pembayaran |  |
| 1. Customer mengklik “OK” dari pembayaran yang dilakukan |  |
|  | 1. System menyimpan data pembayaran |

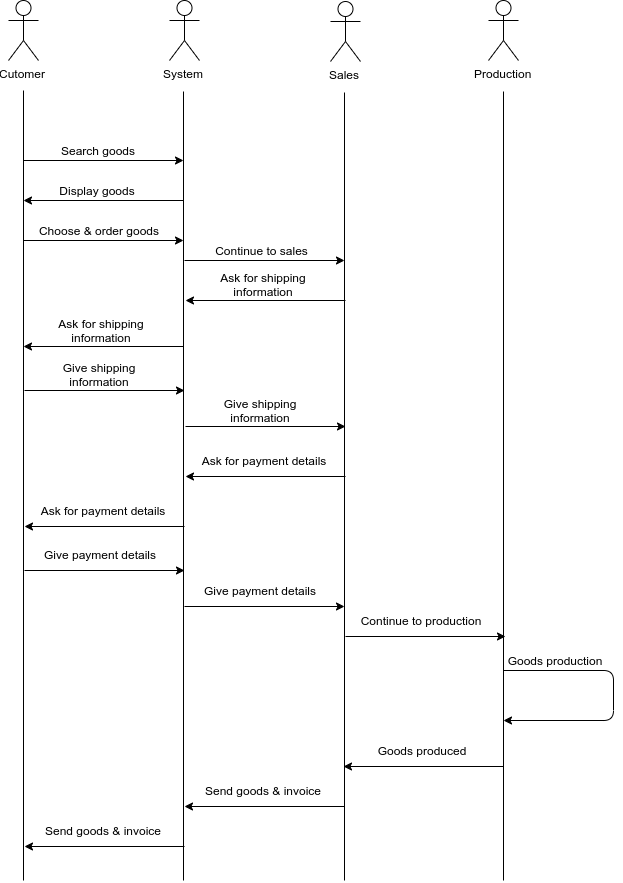
* 1. **Activity Diagram**

Berikut activity diagram pemesanan barang :



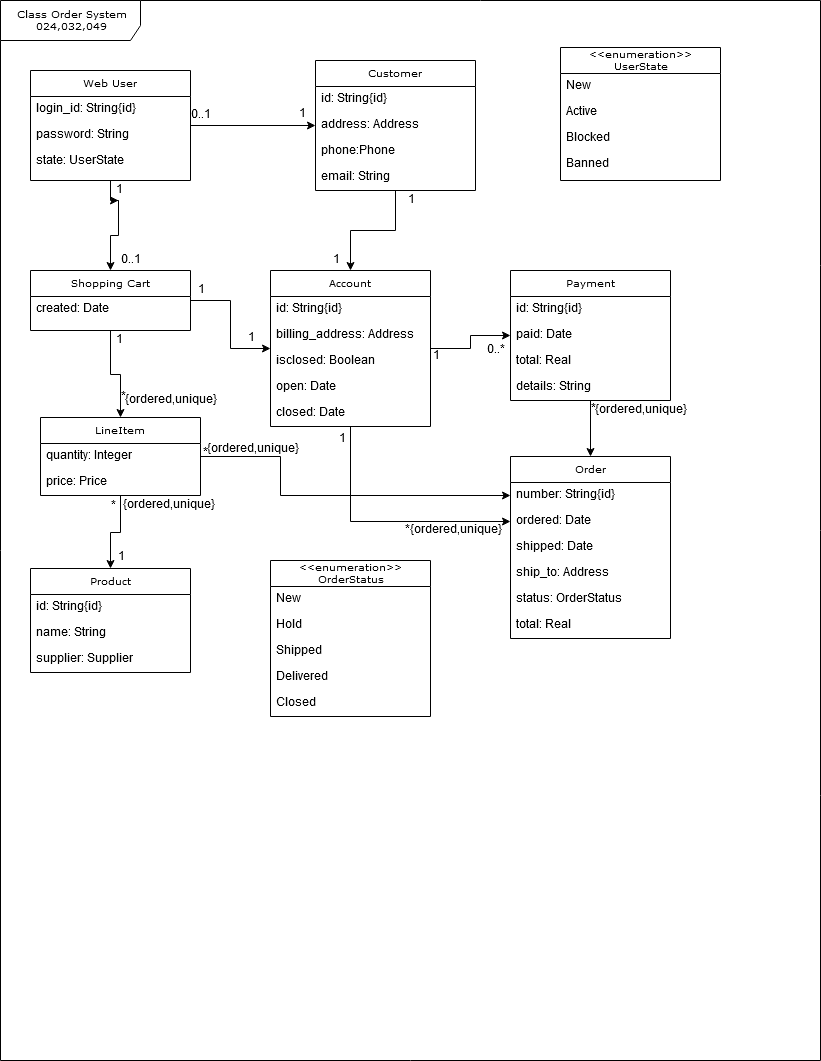
* 1. **Sequence Diagram**

Berikut sequence diagram perancangan pemesanan barang :



* 1. **Class Diagram**

Berikut Class Diagram Pemesanan Barang :



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Daftar Kegiatan** | | Waktu Pelaksanaan | | | | | | | | | | | |
| September | | | | Oktober | | | | November | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| **Analisis Sistem** | **Melakukan wawancara dan rapat koordinasi kepada pihak client** |  |  | **✓** |  |  |  |  |  |  |  |  |  |
| Membuat Usecase Diagram |  |  |  | **✓** |  |  |  |  |  |  |  |  |
| Membuat Activity Diagram |  |  |  | **✓** |  |  |  |  |  |  |  |  |
| Membuat Sequence Diagram |  |  |  | **✓** |  |  |  |  |  |  |  |  |
| Membuat Class Diagram |  |  |  | **✓** |  |  |  |  |  |  |  |  |
| **Desain Aplikasi** | Membuat Desain Menu Aplikasi |  |  |  |  | **✓** |  |  |  |  |  |  |  |
| Dokumentasi Desain Aplikasi |  |  |  |  | **✓** |  |  |  |  |  |  |  |
| **Implementasi** | Membuat Program Aplikasi |  |  |  |  |  | **✓** |  |  |  |  |  |  |
| Dokumentasi Program Aplikasi |  |  |  |  |  |  | **✓** |  |  |  |  |  |
| **Testing** | Melakukan Testing Program |  |  |  |  |  |  |  | **✓** |  |  |  |  |
| Membuat Catatan Perbaikan |  |  |  |  |  |  |  | **✓** |  |  |  |  |
| Dokumentasi Testing Program |  |  |  |  |  |  |  | **✓** |  |  |  |  |
| **Instalasi** | Melakukan Setting Infrastruktur dan  Software Pendukung |  |  |  |  |  |  |  |  | **✓** |  |  |  |
| Instalasi Program Aplikasi |  |  |  |  |  |  |  |  |  | **✓** |  |  |
| Membuat Dokumentasi User Guide |  |  |  |  |  |  |  |  |  | **✓** |  |  |
| Melakukan Pelatihan untuk User |  |  |  |  |  |  |  |  |  |  | **✓** |  |
| **Maintenance** | Pemeliharaan Aplikasi |  |  |  |  |  |  |  |  |  |  |  | **✓** |

# Schedule Time

**Rencana Biaya**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DAFTAR** | **AKTIVITAS** | **JUMLAH**  **TENAGA KERJA** | **BIAYA**  **TENAGA KERJA** | **TOTAL BIAYA** |
|  | Melakukan wawancara  dan rapat koordinasi  kepada pihak  client | 2 | 200.000 | 400.000 |
|  |
|  |
|  |
|  |
|  | Membuat  Usecase  Diagram | 2 | 2.500.000 | 5.000.000 |
|  |
|  |
| **Analisis**  **Sistem** | Membuat  Activity Diagram |
|  | Membuat |
|  | Sequence |
|  | Diagram |
|  | Membuat |
|  | Class Diagram |
| **Desain**  **Aplikasi** | Membuat  Desain Menu | 2 | 2.500.000 | 5.000.000 |
| Dokumentasi Desain Aplikasi | 2 | 2.000.000 | 4.000.000 |
| **Implementasi** | Membuat Program Aplikasi | 2 | 3.500.000 | 7.000.000 |
| Dokumentasi Program Aplikasi | 2 | 2.000.000 | 4.000.000 |
| **Testing** | Melakukan Testing Program | 2 | 2.000.000 | 4.000.000 |
| **Instalasi** | Melakukan Setting Infrastruktur dan Software Pendukung | 3 | 2.000.000 | 6.000.000 |
| Instalasi Program Aplikasi | 2 | 500.000 | 1.000.000 |
| Melakukan Pelatihan Untuk User | 2 | 1.500.000 | 3.000.000 |
| **Maintenance** | Pemeliharaan Aplikasi | 2 | 10.000.000 | 20.000.000 |
| **Lain-lain** | Sewa Server | - | 2.000.000 | 2.000.000 |
| **TOTAL** | | | **30.700.000** | **61.400.000** |

* 1. **Monitoring dan Kontrol Proyek**
     1. **Rencana Managemen Kebutuhan**

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

* + 1. **Rencana Kontrol Schedule**

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

* + 1. **Rencana Kontrol Budget**

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

* + 1. **Rencana Kontrol Kualitas**

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

* + 1. **Rencana Laporan**

**Ada beberapa laporan yang hanya didiskusikan dalam internal tim dalam menjalankan proyek. Adapula 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 Spesification(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 yangberisi SDPLN, SRS, SAD, dan Test Plan.**

* 1. **RencanaManajemenResiko**

|  |  |  |
| --- | --- | --- |
| **Resiko** | **Kejadian** | **Teknik Mengurangi Resiko** |
| **Kegagalan pada personil** | * **Kesalahan dalam coding** * **Tidak menguasai pada bidangnya** * **Timtidak kompak- Terjadi perselisihan dalamtim** * **Jadwal yang rancu** | * **Memperkerjakan 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** | * **Biaya yang**   **berlebihan**   * **Kekurangan biaya** * **Kesalahan perhitungan** | * **Membuat beberapa estimasi** * **Desain untuk biaya** * **Meningkatkan pengembangan** * **Merekam dan menganalisa proyek sebelumnya** * **Standarisasi metode** |
| **Mengembangkan fungsi software**  **yang salah** | * **Kesalahan pemrograman** * **Kesalahan desain** * **Analisa kebutuhan yang tidak sesuai** | * **Evaluasi proyek ditingkatkan** * **Buat metode spesifikasi yang formal** * **Survey pengguna** * **Buat prototype** * **Buat user manual lebih awal** |
| **Mengembangkan**  **Antarmuka** | * **Pengguna tidak bias menggunakan aplikasi yang telah**   **di hasilkan** | * **Membuat prototype** * **Analisis tugas** |
| **Pengguna yang salah** | **- Keterlibatan pengguna** |
| **Kegagalan**  **Menjalankan tugas eksternal** | **-** | * **Prosedur jaminan kualitas** * **Desain/prototype yang kompetitif** * **Membangun tim** * **Kontrak insentif** |

# BAB IV

## Kesimpulan dan Saran

* 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 eleman utama UML yaitu :

* + - * Benda/Things/Objek
      * Hubungan/Relationship
      * Bagan atauDiagrams