

**ANALISIS POLA PENYIMPANGAN PERILAKU REMAJA DI KOTA  
PADANG DENGAN METODE FP-GROWTH**

**LAPORAN PENELITIAN**



**OLEH :**

**M.HAFIZH, M.KOM**

**RIMA LIANA GEMA**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PUTRA INDONESIA "YPTK"**

**PADANG**

## ABSTRAK

Perancangan sistem simulasi menentukan pendapatan keripik maco dengan menggunakan metode *monte carlo* pada *home-industry* Istiqomah bertujuan untuk memprediksi penjualan setiap produk barang untuk bulan berikutnya dan mencari pendapatan kotor disetiap barang yang laku terjual. Sistem simulasi penjualan dirancang untuk memprediksi barang dan mencari pendapatan kotor disetiap jenis barang pada *home-industry* Istiqomah berbasis web yang dibuat dengan menggunakan *PHP* dan *MySQL* sebagai database. Dengan sistem ini diharapkan dapat digunakan oleh *home-industry* untuk mensimulasikan penjualan barang pada toko, dan dapat mempermudah bagi pemilik *home-industry* dalam memperoleh informasi tentang kemungkinan yang akan terjadi terhadap penjualan produk keripik maco.

Kata Kunci : Produk, Monte Carlo, *PHP* & *MySQL*

## ABSTRACT

The design of the simulation system determines the number of sales of furniture items by using the monte carlo method at *home-industry* Istiqomah aims to predict the sale of each product for the next month and look for the gross income in each sold item. The sales simulation system is designed to predict goods and search for dirty gains in each type of goods on a web-based store created using PHP and MySQL as the database. With this system is expected to be used by the *home-industry* to simulate the sale of goods on the store, and can make it easier for *home-industry* owners to obtain information about the possibility that will happen to the sale of keripik maco products.

Keywords: Products, Monte Carlo, PHP & MySQL

## KATA PENGANTAR

Puji syukur penulis sampaikan kehadiran Allah SWT yang telah memberikan syafat dan, hidayah serta karunia-nya sehingga penulis dapat menyelesaikan Tugas Akhir (Skripsi) dan dapat pula menyelesaikan laporan ini. Shalawat dan salam penulis kirimkan kepada junjungan alam, kekasih Allah SWT Nabi Muhammad SAW yang telah menjadi tolak ukur, pedoman, dan bimbingan bagi kehidupan manusia dari kebodohan dan kegelapan menuju cahaya terang dengan kebenaran ilmu, iman, dan amal seperti saat ini.

Laporan Skripsi ini diajukan untuk memenuhi persyaratan akademis program studi Strata-1, pada Universitas Putra Indonesia YPTK Padang, Jurusan Teknik Informatika pada Fakultas Ilmu Komputer. Adapun judul penelitian adalah: **“PEMODELAN SIMULASI DALAM MENENTUKAN PENDAPATAN KERIPIK MACO BADARAI ISTIQOMAH MENGGUNAKAN METODE MONTE CARLO BERBASIS WEB”**

Dalam menyelesaikan Penelitian ini kami mengucapkan terima kasih kepada :

1. **Bapak H.Herman Nawas**, selaku Ketua Yayasan Universitas Putra Indonesia YPTK Padang.
2. **Bapak Dr. Sarjon Defit, S.Kom, M.Sc**, selaku Rektor Universitas Putra Indonesia YPTK Padang
3. **Bapak Dr. Julius Santony, S.Kom, M.Kom**, selaku Dekan Fakultas Ilmu Komputer Universitas Putra Indonesia YPTK Padang
4. **Ibu Rini Sovia, S.Kom, M.Kom**, selaku Ketua Program Studi Teknik Informatika Universitas Putra Indonesia YPTK Padang
5. **Bapak Eka Praja Wiyata Mandala, S.kom, M.kom** selaku Sekretaris I Program Studi Teknik Informatika Universitas Putra Indonesia “YPTK” Padang.

6. **Bapak Randy Permana, S.Kom, M.kom** selaku Sekretaris II program studi Teknik Informatika Universitas Putra Indonesia “YPTK” Padang.
7. **Ibu Rima Liana Gema, S.Kom, M.Kom**, selaku Dosen Pembimbing I yang telah memberikan banyak bimbingan dan masukan dalam penyusunan atau penulisan penelitian ini
8. **Bapak M. Hafizh, S.Kom, M.Kom**, selaku Dosen Pembimbing II yang telah memberikan banyak bimbingan dan masukan dalam penyusunan atau penulisan penelitian ini
9. Seluruh staf dosen, Karyawan dan Karyawati di lingkungan Universitas PutraIndonesia YPTK Padang yang telah banyak mendidik dan mengajarkan kami dalam berbagai ilmu.
10. Bapak dan Ibu dari *home-industry* Istiqomah yang telah bersedia memberikan data-data yang dibutuhkan oleh penulis, sehingga penulis dapat menyelesaikan penelitian ini.
11. Dan kepada teman-teman yang telah banyak membantu, memberikan semangat dan motivasi sehingga Skripsi ini dapat diselesaikan.

Akhir kata, semoga skripsi ini dapat memberikan kontribusi terhadap masyarakat dan khususnya terhadap Universitas Putra Indonesia YPTK Padang. Penulis mengucapkan terima kasih dan mudah-mudahan laporan skripsi ini berguna dan dapat menambah wawasan bagi para pembaca.

Padang, 3 Juli 2019

Tim Peneliti

## DAFTAR ISI

### BAB I PENDAHULUAN

1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	2
1.3. Hipotesa .....	3
1.4. Batasan Masalah .....	3...
1.5. Tujuan Penelitian .....	4
1.6. Manfaat Penelitian .....	4

### BAB II LANDASAN TEORI

2.1. Rekayasa Perangkat Lunak.....	6
2.1.1. Tujuan Reakayasa Perangkat Lunak.. .....	7
2.1.2. Proses Rekayasa Perangkat Lunak.....	7
2.1.3 Model Rekayasa Perangkat Lunak .....	8
2.1.4 Model Waterfall .....	9
2.1.5 Model Prototipe.....	11
2.1.6 Model RAD.....	12
2.1.7 Model Iteratif .....	14
2.1.8 Model Spiral.....	16
2.2. <i>Unified Modelling Language</i> ( UML ).....	17
2.2.1. Sejarah UML.....	17
2.2.2. Diagram UML.....	18
2.2.3. Usecase Diagram.....	20
2.2.4. Class Diagram .....	22
2.2.5. Activity Diagram.....	24
2.2.6. Sequence Diagram .....	26

2.2.7. Collaboration Diagram .....	28
2.2.8. Deployment Diagram .....	29
2.2.9. State Machine Diagram .....	30
2.3. Tentang Pemodelan Dan Simulasi .....	31
2.3.1. Tujuan Pemodelan Dan Simulasi .....	32
2.3.2. Istilah Dalam Simulasi .....	33
2.3.3. Pengertian Model .....	34
2.3.4. Jenis-jenis Model .....	35
2.3.5. Definisi Simulasi .....	36
2.3.6. Metode <i>Monte Carlo</i> .....	38
2.3.7. Simulasi Monte Carlo.....	38
2.3.8. Teknik Monte Carlo .....	39
2.3.9. Random Number .....	41
2.4. Pemograman Web.....	44
2.4.1. PHP.....	44
2.4.2. Sejarah PHP.....	45
2.4.3. Kelebihan PHP .....	46
2.4.4. Kekurangan PHP.....	46
2.5. Konsep Database.....	47
2.5.1. Pengertian Database .....	47
2.5.2. <i>MySQL</i> .....	47

### **BAB III METODOLOGI PENELITIAN**

3.1. Kerangka penelitian .....	49
3.2. Tahapan Penelitian .....	50
3.2.1. Penelitian Pendahuluan.....	50

3.2.2. Pengumpulan Data.....	50
3.2.2.1 Waktu Penelitian.....	50
3.2.2.2 Tempat Penelitian .....	51
3.2.2.3 Metode Penelitian .....	51
3.2.3. Analisa.....	53
3.2.3.1 Analisa Data.....	53
3.2.3.2 Analisa Proses.....	53
3.2.3.3 Analisa Sistem .....	53
3.2.4. Perancangan.....	54
3.2.4.1 Perancangan Model.....	54
3.2.5. Implementasi .....	56
3.2.6. Pengujian Sistem .....	57

## **BAB IV ANALISA DAN PERANCANGAN**

4.1. Analisa.....	58
4.1.1. Analisa Data.....	58
4.1.2. Pengumpulan Data .....	58
4.1.3. Analisa Proses .....	59
4.1.4 Analisa Sistem.....	75
4.2. Perancangan Model.....	76
4.2.1 Use Case Diagram.....	76
4.2.2 Class Diagram .....	83
4.2.3 Sequence Diagram .....	88
4.2.4 Collaboration Diagram.....	92
4.2.5 Statechart Diagram.....	96
4.2.6 Activity Diagram.....	97
4.2.7 Deployment Diagram.....	98



4.3. Perancangan Interface.....	99
4.3.1. Tampilan Menu Login.....	99
4.3.2. Tampilan Setelah Login .....	99
4.3.3. Tampilan Input Barang .....	100
4.3.4. Tampilan Input Random .....	100
4.3.5. Tampilan Input Penjualan.....	100
4.3.6. Tampilan Hasil Perhitungan.....	101
4.3.7. Tampilan Grafik.....	101

## **BAB V IMPLEMENTASI DAN PENGUJIAN**

5.1. Implementasi.....	101
5.1.1. Implementasi Sistem .....	101
5.1.2. Instalasi XAMPP .....	102
5.1.3. Implementasi Basis Data (SQL) dengan XAMPP .....	104
5.2. Pengujian .....	109
5.2.1. Pengujian Sistem... ..	109
5.2.2. Pengujian Interface .....	111

## **BAB VI PENUTUP**

6.1. Kesimpulan .....	117
6.2. Keterbatasan Sistem.....	117
6.3. Saran .....	118

## **DAFTAR PUSTAKA**

## **DAFTAR TABEL**

Tabel 2.1 Simbol Simbol Use Case Diagram .....	21
--	----

Tabel 2.2 Skenario Usecase.....	22
Tabel 2.3 Simbol Simbol Class Diagram .....	23
Tabel 2.4 Simbol Simbol Activity Diagram.....	25
Tabel 2.5 Simbol Simbol Sequence Diagram .....	27
Tabel 2.6 Simbol Simbol Communication Diagram .....	28
Tabel 2.7 Simbol Simbol Deployment Diagram .....	29
Tabel 2.8 Simbol Simbol Statemachine Diagram .....	31
Tabel 3.1 Penelitian.....	51
Tabel 4.1 Penjualan Di Toko Hnh Mart Unand .....	60
Tabel 4.2 Penjualan Di Toko Citra Sungai Galang .....	60
Tabel 4.3 Penjualan Di Toko Mahkota Air Tawar .....	61
Tabel 4.4 Penjualan Di Toko Daft Mart.....	61
Tabel 4.5 Penjualan Di Toko Mahkota Tabiang .....	61
Tabel 4.6 Penjualan Di Toko Yessie Tabing.....	62
Tabel 4.7 Penjualan Di Toko Shirley .....	62
Tabel 4.8 Penjualan Di Toko Silungkang Abu Nawas.....	62
Tabel 4.9 Distribusi Kemungkinan Dan Kumulatif Di Toko Hnh Mart Unand.....	63
Tabel 4.10 Distribusi Kemungkinan Dan Kumulatif Di Toko Citra Sungai Galang .....	63
Tabel 4.11 Distribusi Kemungkinan Dan Kumulatif Di Toko Mahkota Air Tawar .....	64

Tabel 4.12 Distribusi Kemungkinan Dan Kumulatif Di Toko Daf Mart .....	64
Tabel 4.13 Distribusi Kemungkinan Dan Kumulatif Di Toko Mahkota Tabiang.....	64
Tabel 4.14 Distribusi Kemungkinan Dan Kumulatif Di Toko Yessie Tabiang .....	65
Tabel 4.15 Distribusi Kemungkinan Dan Kumulatif Di Toko Shirley.....	65
Tabel 4.16 Distribusi Kemungkinan Dan Kumulatif Di Toko Silungkang Abu Nawas	65
Tabel 4.17 Batasan Di Toko Hnh Mart Unand.....	66
Tabel 4.18 Batasan Di Toko Citra Sungai Galang .....	66
Tabel 4.19 Batasan Di Toko Mahkota Air Tawar .....	67
Tabel 4.20 Batasan Di Toko Daf Mart .....	67
Tabel 4.21 Batasan Di Toko Mahkota Tabiang.....	67
Tabel 4.22 Batasan Di Toko Yessie Tabing .....	68
Tabel 4.23 Batasan Di Toko Shirley .....	68
Tabel 4.24 Batasan Di Toko Silungkang Abu Nawas .....	68
Tabel 4.25 Percobaan Gross Income Di Toko Hnh Mart Unand .....	70
Tabel 4.26 Percobaan Gross Income Di Toko Citra Sungai Galang .....	71
Tabel 4.27 Percobaan Gross Income Di Toko Mahkota Air Tawar .....	71
Tabel 4.28 Percobaan Gross Income Di Toko Daf Mart.....	72
Tabel 4.29 Percobaan Gross Income Di Toko Mahkota Tabiang .....	72
Tabel 4.30 Percobaan Gross Income Di Toko Yessie Tabiang.....	73

Tabel 4.31 Percobaan Groos Income Di Toko Shirley .....	73
Tabel 4.32 Percobaan Gross Income Di Toko Silungkang Abu Nawas .....	74
Tabel 4.33 Pendapatan Kotor .....	75
Tabel 4.34 Aktor Dari Sistem Aplikasi .....	77
Tabel 4.35 Scenario Use Case Login .....	77
Tabel 4.36 Input Barang .....	78
Tabel 4.37 Input Random .....	79
Tabel 4.38 Input Penjualan .....	80
Tabel 4.39 Input Hasil Perhitungan .....	81
Tabel 4.40 Tabel Grafik .....	82
Tabel 4.41 Keterangan Class Diagram .....	84
Tabel 4.42 Tabel Admin .....	85
Tabel 4.43 Tabel Barang .....	86
Tabel 4.44 Tabel Grafik .....	86
Tabel 4.45 Tabel Penjualan .....	87
Tabel 4.46 Tabel Proses .....	87
Tabel 4.47 Tabel Random .....	88
Tabel 5.1 Hasil Fungsionalitas Aplikasi Simulasi Dengan Metode Monte Carlo .....	109

## DAFTAR GAMBAR

Gambar 2.1 Tahapan Umum Rekayasa Perangkat Lunak .....	7
Gambar 2.2 Ilustrasi Model Waterfall .....	9
Gambar 2.3 Ilustrasi Model Prototipe .....	11
Gambar 2.4 Ilustrasi Model Rad.....	13
Gambar 2.5 Ilustrasi Model Iteratif .....	15
Gambar 2.6 Ilustrasi Model Spiral .....	16
Gambar 2.7 Pembagian Kategori.....	19
Gambar 3.1 Kerangka Penelitian .....	49
Gambar 4.1 Data Mentah Penjualan.....	59
Gambar 4.2 Data Harga Jual.....	59
Gambar 4.3 Use Case Diagram .....	82
Gambar 4.4 Classs Diagram .....	83
Gambar 4.5 Sequence Diagram Home Admin .....	88
Gambar 4.6 Sequence Diagram Barang .....	89
Gambar 4.7 Sequence Diagram Penjualan .....	89
Gambar 4.8 Sequence Diagram Hasil Perhitungan .....	90
Gambar 4.9 Sequence Diagram Grafik.....	91
Gambar 4.10 Collaboration Diagram Input Random .....	91
Gambar 4.11 Collaboration Diagram Home Admin .....	92
Gambar 4.12 Collaboration Diagram Barang.....	93
Gambar 4.13 Collaboration Diagram Penjualan.....	93
Gambar 4.14 Collaboration Diagram Hasil Perhitungan.....	94
Gambar 4.15 Collaboration Diagram Grafik.....	94
Gambar 4.16 Collaboration Diagram Input Random .....	95

Gambar 4.17 Statechart Input Barang .....	95
Gambar 4.18 Statechart Diagram Penjualan .....	96
Gambar 4.19 Statechart Diagram Input Random .....	96
Gambar 4.20 Activity Diagram .....	97
Gambar 4.21 Deployment Diagram .....	97
Gambar 4.22 Tampilan Menu Login .....	98
Gambar 4.23 Tampilan Setelah Login .....	98
Gambar 4.24 Tampilan Input Barang .....	99
Gambar 4.25 Tampilan Input Random .....	99
Gambar 4.26 Input Penjualan .....	100
Gambar 4.27 Tampil Hasil Perhitungan .....	100
Gambar 4.28 Tampilan Grafik .....	100
Gambar 5.1 Tampilan Setup Wizard .....	102
Gambar 5.2 Tampilan Piih Penyimpanan .....	102
Gambar 5.3 Tampilan Service Selection .....	103
Gambar 5.4 Tampilan Proses Instalasi .....	103
Gambar 5.5 Tampilan Instalasi Success .....	104
Gambar 5.6 Tampilan Hasil Instalasi .....	104
Gambar 5.7 Tampilan XAMPP Panel .....	105
Gambar 5.8 Pengetikan Pada Browser .....	105
Gambar 5.9 Pembuatan Database Baru .....	105
Gambar 5.10 Tabel Admin .....	106
Gambar 5.11 Tabel Barang .....	106
Gambar 5.12 Tabel Grafik .....	107
Gambar 5.13 Tabel Input .....	107

Gambar 5.14 Tabel Proses.....	108
Gambar 5.15 Tabel Random.....	108
Gambar 5.16 Halaman Login .....	112
Gambar 5.17 Halaman Home .....	112
Gambar 5.18 Halaman Input Barang .....	113
Gambar 5.19 Halaman Input Random.....	113
Gambar 5.20 Halaman Input Penjualan.....	114
Gambar 5.21 Halaman Hasil Perhitungan .....	115
Gambar 5.22 Hasil Perhitungan Simulasi.....	115
Gambar 5.23 Halaman Grafik .....	116
Gambar 5.24 Halaman Logout .....	116

## BAB I

### PENDAHULUAN

#### 1.1. Latar belakang

Kemajuan teknologi komputer yang pesat dapat membantu kehidupan manusia bahkan didalam bidang-bidang diluar disiplin ilmu komputer. Perkembangan dan kemajuan ilmu pengetahuan menyebabkan terjadinya perubahan-perubahan dalam segala sektor usaha dan penjualan dimana perubahan-perubahan ini terjadi sejalan dengan perkembangan dan kemajuan dari ilmu pengetahuan itu sendiri.

Seiring dengan maraknya usaha *home-industry* dan permintaan konsumen akan makanan ringan yang terus meningkat, munculah persaingan ketat dalam bisnis makanan ini. Setiap kompetitor saling bersaing dalam penciptaan kualitas produk seperti bentuk, ukuran, rasa, kemasan dan harga. Tidak hanya kualitas produk saja yang berpengaruh besar dalam penjualan. Kemasan juga mempunyai peran yang sangat penting dalam pemasaran produk. Dalam penjualan produk makanan, tentu saja diperlukan sebuah kemasan sebagai wadah guna melindungi produk makanan yang ada di dalamnya agar tidak rusak, terutama bila produk makanan tersebut adalah produk keripik yang mudah remuk. Agar pendapatan sebanding (laba/rugi) setiap minggu dan bulannya maka harus ada perhitungan yang bagus untuk meningkatkan kinerja dan kualitas produk.

Simulasi penjualan merupakan sebuah perkiraan tentang penjualan (*state-estimate*) tetapi merupakan tindakan penyesuaian kesempatan yakni permintaan aktual dan potensial dengan usaha-usaha pemasaran yang diperlukan, agar tujuan dapat diperoleh suatu manfaat dari padanya. Simulasi sendiri merupakan imitasi atau tiruan dari aktivitas atau proses sebuah sistem.



Metode *Monte Carlo* merupakan tipe simulasi probabilistik untuk mencari penyelesaian masalah dengan sampling dari proses random. Simulasi Monte Carlo mengizinkan untuk seorang manajer menentukan beberapa kebijakan yang menyangkut kondisi organisasi. Selain itu, metode simulasi *Monte Carlo* merupakan sebuah teknik simulasi yang menggunakan unsur acak disaat terdapat peluang. Dasar simulasi *Monte Carlo* adalah percobaan pada unsur peluang atau bersifat probabilistik dengan menggunakan pengambilan sampel secara acak.

Dalam penelitian ini Pemodelan dan Simulasi Monte Carlo digunakan untuk membantu menentukan pendapatan dan persentasi penjualan produk keripik maco sehingga dapat digunakan sebagai pertimbangan dalam membuat strategi promosi pasar untuk penjualan yang efektif. Oleh karena itu saya mengangkat judul penelitian tentang **“PEMODELAN DAN SIMULASI DALAM MENENTUKAN PENDAPATAN KERIPIK MACO BADARAI ISTIQOMAH MENGGUNAKAN METODE MONTE CARLO BERBASIS WEB”**

## **1.2 Perumusan masalah**

Berdasarkan latar belakang yang telah diuraikan di atas, masalah yang akan dibahas dalam penelitian ini adalah:

1. Bagaimana aplikasi menggunakan metode Monte Carlo yang dirancang dapat membantu dalam mengetahui keuntungan produk Keripik Maco Badarai Istiqomah?
2. Bagaimana aplikasi menggunakan metode Monte Carlo dapat membantu *home-industry* Istiqomah dalam membandingkan pendapatan disetiap periodenya?

3. Bagaimana aplikasi menggunakan metode Monte Carlo dapat membantu *home-industry* Istiqomah mengetahui tingkat minat konsumen terhadap jenis produk keripik maco?

### 1.3 Hipotesa

Berdasarkan permasalahan yang telah disebutkan diatas, maka didapat hipotesa sebagai jawaban sementara yang ada, yaitu :

1. Diharapkan metode Monte Carlo dapat diterapkan pada pemodelan dan simulasi dalam pencarian keuntungan berdasarkan laporan penjualan yang ada pada *home-industry* Istiqomah”.
2. Dengan pemodelan dan simulasi diharapkan dapat membangun aplikasi dalam membandingkan pendapatan pada *home-industry* Istiqomah disetiap periodenya.
3. Dengan pemodelan dan simulasi dalam menentukan pendapatan keripik maco badarai menggunakan metode monte carlo berbasis web dapat diharapkan mengetahui tingkat minat konsumen pada *home-industry* Istiqomah.

### 1.4 Batasan Masalah

Dari uraian latar belakang dan perumusan masalah di atas, maka dibuat batasan masalah sebagai berikut :

1. Aplikasi keuntungan Keripik Maco Badarai yang dirancang untuk *home-industry* Istiqomah ini dapat mengetahui pendapatan bersih untuk produk Keripik Maco Badarai.
2. Pemilik *home-industry* Istiqomah dapat membandingkan pendapatan dari setiap periodenya dari produk Rakik Maco Badarai.

3. Perancangan aplikasi yang dibuat yaitu berbasis web dengan PHP dan MySQL untuk penelitian ini akan dilakukan perbandingan antara bulan Agustus 2018 sampai Februari 2019 sehingga perancangan aplikasi ini akan memberikan informasi yaitu jumlah keuntungan yang dicapai setiap minggunya dan selama 6 bulan.

### 1.5 Tujuan Penelitian

Berdasarkan latar belakang yang telah dijelaskan, maka tujuan dari penelitian ini adalah:

1. Pemilik dapat mengetahui pendapatan yang didapat dari periode-periode sebelumnya dari setiap item barang.
2. Dapat membandingkan pendapatan yang didapatkan oleh *home-industry* Istiqomah dalam penjualan keripik maco badarai.
3. Dapat mengetahui tingkat minat pembeli terhadap barang di-setiap minggu atau setiap bulan dan setiap tahunnya.
4. Dapat membantu dalam menentukan pendapatan *home-industry* keripik maco badarai Istiqomah menggunakan metode monte carlo.

### 1.6 Manfaat Penelitian

Adapun manfaat yang diberikan dalam penelitian ini adalah :

#### A. *Home-industry* Istiqomah

1. Dapat memudahkan pemilik *home-industry* Istiqomah untuk mengetahui pendapatan yang dihasilkan untuk produk Keripik Maco Badarai.
2. Dapat mengetahui minat beli konsumen terhadap masing-masing produk Keripik Maco Badarai.

#### B. Peneliti

Meningkatkan inovasi dan kreatifitas dalam menciptakan sebuah karya yang mengimplementasikan ilmu teknologi informasi dan komunikasi.

## **BAB II**

## LANDASAN TEORI

### 2.1 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak (RPL) lebih fokus pada praktik pengembangan pelanggan (*customer*). Adapun ilmu komputer lebih fokus pada teori dan konsep dasar perangkat komputer. Perangkat lunak yang baik adalah perangkat yang fokus pada pengguna atau pelanggan (Rosa & M. Shalahudin, 2014). Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut :

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainability*).
2. Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*dependability dan robust*).
3. Efisien dari segi sumber daya dan pengguna.
4. Kemampuan untuk dipakai sesuai dengan kebutuhan (*usability*).

Dari kriteria diatas maka perangkat lunak yang baik adalah perangkat lunak yang dapat memenuhi kebutuhan pelanggan (*customer*) atau pemakai perangkat lunak (*user*) atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak.

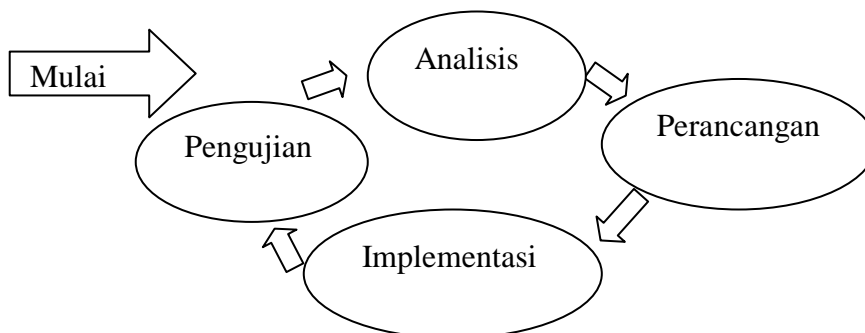
#### 2.1.1 Tujuan Rekayasa Perangkat Lunak

Bidang rekayasa akan selalu berusaha menghasilkan *output* yang kinerjanya tinggi, biaya rendah dan waktu penyelesaian yang tepat. Secara lebih khusus kita dapat menyatakan tujuan RPL adalah :

1. Memperoleh biaya produksi perangkat lunak yang rendah.
2. Menghasilkan perangkat lunak yang kinerjanya tinggi, handal dan tepat waktu.
3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis *platform*.
4. Menghasilkan perangkat lunak yang biaya perawatannya rendah.

### 2.1.2 Proses Rekayasa Perangkat Lunak

Proses rekayasa perangkat lunak dilakukan selama pembangunan perangkat lunak. Proses-proses yang dilakukan dalam rekayasa perangkat lunak secara garis besar adalah sebagai berikut pada gambar 2.1 :



*Sumber : Rosa A.S M.Salahuddin (2014)*

### Gambar 2.1 Tahapan Umum Rekayasa Perangkat Lunak

Proses-proses pada gambar di atas dapat dilakukan berulang kali sampai perangkat lunak memenuhi kebutuhan pelanggan atau user. Untuk membangun perangkat lunak yang benar-benar baik maka diperlukan tahapan-tahapan rekayasa perangkat lunak. Perangkat lunak yang dibangun tanpa melalui tahapan rekayasa perangkat lunak yang baik maka akan seperti membuat baju tanpa pola dan rencana model baju yang baik. Dalam membuat sebuah baju, maka dibutuhkan adanya perencanaan model baju dan pembuatan pola baju

yang baik. Setelah pola baju terbentuk maka kain mulai dipotong sesuai pola, baru kemudian dijahit. Jika baju dibuat tanpa pola, maka yang terjadi adalah baju hasil tambal sulam.

Seperti halnya pembuatan perangkat lunak tanpa melalui proses rekayasa perangkat lunak, maka hanya akan menghasilkan perangkat lunak “tambal sulam”, begitu kurang maka akan ditambal. Tentu saja hasil perangkat lunak “tambal sulam” tidak akan bagus. Rekayasa perangkat lunak bukan seperti sulap yang bisa diselesaikan dalam sekejap, ada beberapa proses yang harus di lalui agar menghasilkan perangkat lunak yang berkualitas (Rosa A.S dan M. Shalahuddin, 2014).

### **2.1.3 Model Rekayasa Perangkat Lunak**

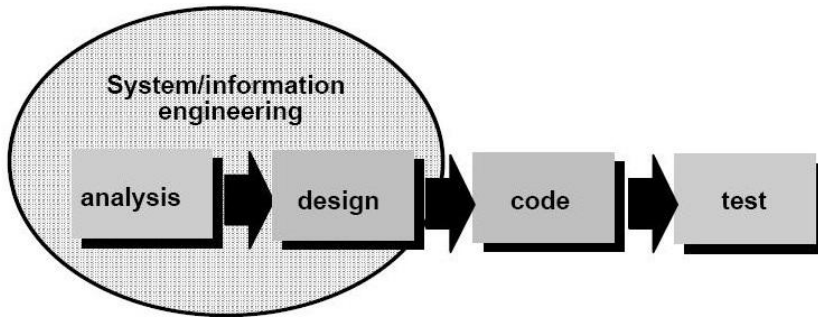
Pada rekayasa perangkat lunak, banyak model yang telah dikembangkan untuk membantu proses pengembangan perangkat lunak. Model-model ini pada umumnya mengacu pada model pengembangan sistem yang disebut *System Development Live Cycle* (SDLC).

SDLC atau Software Development Life Cycle atau sering disebut juga System Development Life Cycle adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (Rosa A.S dan M. Shalahuddin, 2014).

SDLC memiliki model-model dalam penerapan tahapan prosesnya. Model-model tersebut diantaranya:

### **2.1.4 Model *Waterfall***

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar 2.2 model air terjun:



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

**Gambar 2.2 Ilustrasi Model Waterfall**

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif demi memesifikasikan kebutuhan perangkat lunak supaya dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang terfokus pada desain perancangan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur coding.

c. Pembuatan kode program



Desain harus di translasikan kedalam program perangkat lunak, hasil dari tahap ini adalah program komputer yang harus sesuai dengan desain yang telah dibuat pada tahap sebelumnya.

d. Pengujian

Pengujian fokus pada perangkat lunak dari segi logis dan fungsional serta memastikan bahwa semua bagian sudah diuji.

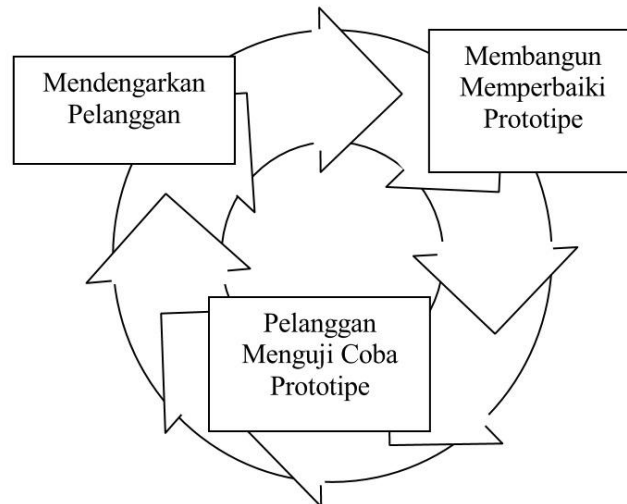
e. Pendukung dan pemeliharaan

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru (A.S & M.Shalahudin, 2014).

### **2.1.5 Model Prototype**

Model Prototype merupakan suatu paradigma baru dalam metode pengembangan perangkat lunak dimana metode ini tidak hanya sekedar evolusi dalam dunia pengembangan perangkat lunak, tetapi juga merevolusi metode pengembangan perangkat lunak yang lama yaitu sistem sekuensial yang biasa dikenal dengan nama SDLC atau Waterfall Development Model di jelaskan pada gambar 2.3.



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

### **Gambar 2.3 Ilustrasi Model Prototipe**

*Mock-up* adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi, atau keperluan lain. Sebuah *Mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesuai dengan keinginan pelanggan (*costumer*) atau *user*. Model prototipe cocok digunakan untuk menggali spesifikasi kebutuhan pelanggan secara lebih detail tetapi beresiko tinggi terhadap membengkaknya

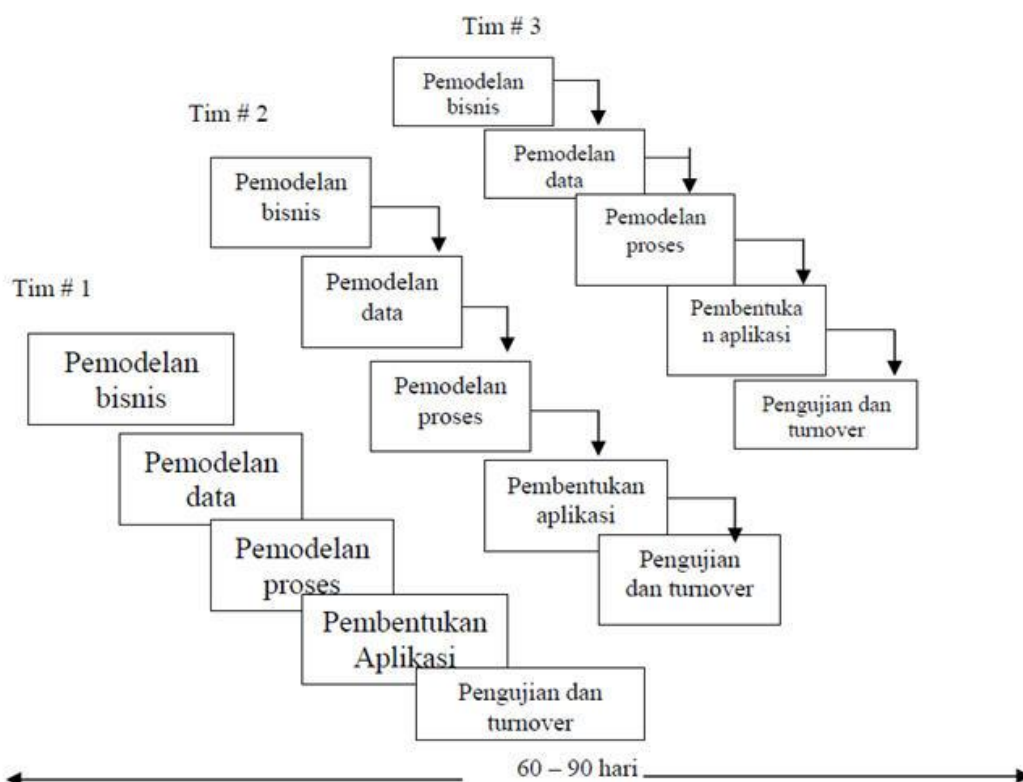
biaya dan waktu proyek (Rosa & M.Shalahudin, 2014).

#### **2.1.6 Model Rapid Application Development (RAD)**

*Rapid Application Development* (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek. Model RAD adalah adaptasi dari model air terjun versi kecepatan tinggi dengan

menggunakan model air terjun untuk pengembangan setiap komponen perangkat lunak (Rosa & M.Shalahudin, 2014).

Metode RAD merupakan pendekatan berorientasi objek untuk menghasilkan sebuah sistem dengan sasaran utama mempersingkat waktu pengerjaan aplikasi dan proses agar sesegera mungkin memberdayakan sistem perangkat lunak tersebut secara tepat dan cepat. Untuk perancangan suatu sistem informasi yang normal seandainya membutuhkan waktu minimal 180 hari, maka dengan menerapkan metode RAD hanya membutuhkan waktu 30-90 hari untuk menyelesaikan sistem perangkat lunak tersebut. dapat dijelaskan pada gambar 2.4.



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

**Gambar 2.4 Metode RAD (Rapid Application Development)**

a. Pemodelan Bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa yang terkait proses bisnis, informasi apa saja yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yang terkait informasi itu.

b. Pemodelan Data

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasi dengan data-data yang lain.

c. Pemodelan Proses

Mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.

d. Pembuatan Aplikasi

Mengimplementasikan pemodelan proses dan data menjadi program. Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

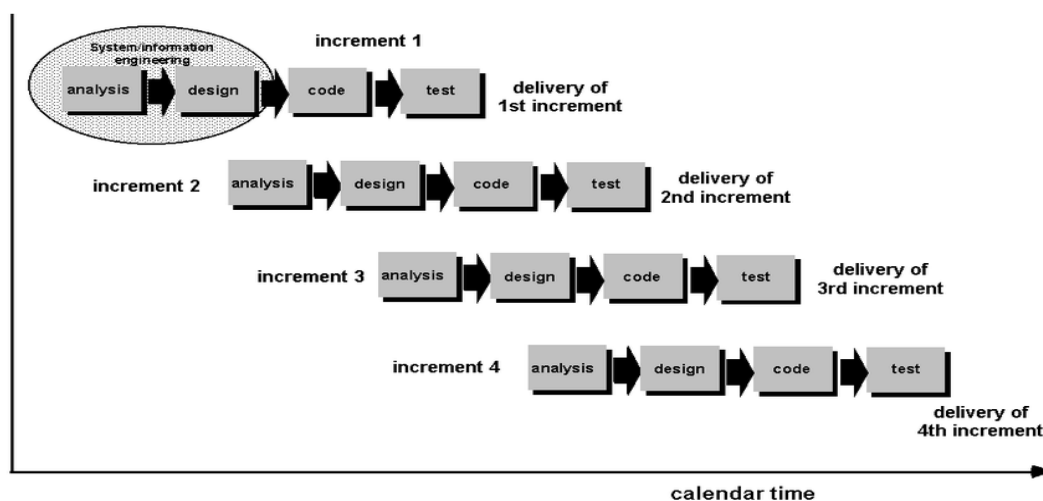
e. Pengujian dan Pergantian

Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya.

### **2.1.7 Model Iteratif**

Model iteratif merupakan model yang diciptakan untuk mengatasi beberapa kekurangan model *waterfall*. Model ini bisa dikatakan sebagai mini *waterfall*. Di

manasetiap subbagian sekuensial dari pengembangan program, di dalamnya disusun dengan mengikuti fase analisis, desain, implementasi, dan pengujian. Hasil pengujian dipakai sebagai dasar dalam mengerjakan subbagian selanjutnya. Model iteratif merupakan gabungan dari model *waterfal* dan model prototipe, model ini cocok digunakan pengembang dengan *turnover* staf yang tinggi (Rosa & M.Shalahudin, 2014). Berikut adalah gambar 2.5 dari model inkremental:



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

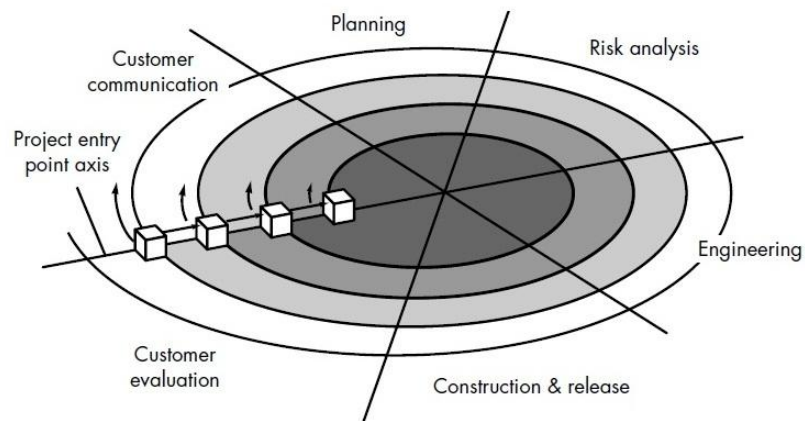
**Gambar 2.5 Ilustrasi Model Iteratif**

Model inkremental dibuat untuk mengatasi kelemahan dari model air terjun yang tidak mengakomodasikan iterasi, dan mengatasi kelemahan dari metode prototipe yang memiliki proses terlalu pendek dan setiap iteratif prosesnya tidak selalu menghasilkan produk.

### 2.1.8 Model *Spiral*

Model *spiral* (*spiral model*) memasangkan iteratif pada model prototipe dengan kontrol dan aspek sistematis yang diambil dari model air terjun. Model spiral menyediakan pengembangan dengan cara cepat dengan perangkat lunak yang memiliki versi yang terus bertambah fungsinya (*increment*). Model spiral cocok digunakan untuk

mengembangkan aplikasi dengan skala besar tetapi target waktu dan biaya tidak terlalu meningkat (Rosa & M.Shalahudin, 2014). Berikut adalah gambar 2.6 dari model spiral:



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

**Gambar 2.6 Ilustrasi Model Spiral**

Model spiral dilakukan searah jarum jam dimulai dari sumbu proyek. Sumbu proyek dapat digunakan sebagai awal iterasi ataupun evaluasi dari iterasi yang sudah dilakukan. Pada gambar diatas setiap wilayah berputar dengan urutan kerja tertentu.

## 2.2 Unified Modeling Language (UML)

Pada perkembangan teknik pemrograman berorientasi objek, munculah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

Unified Modeling Language(UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teksteks pendukung (Andrianof, 2017).

UML hanya berfungsi untuk melakukan pemodelan. UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

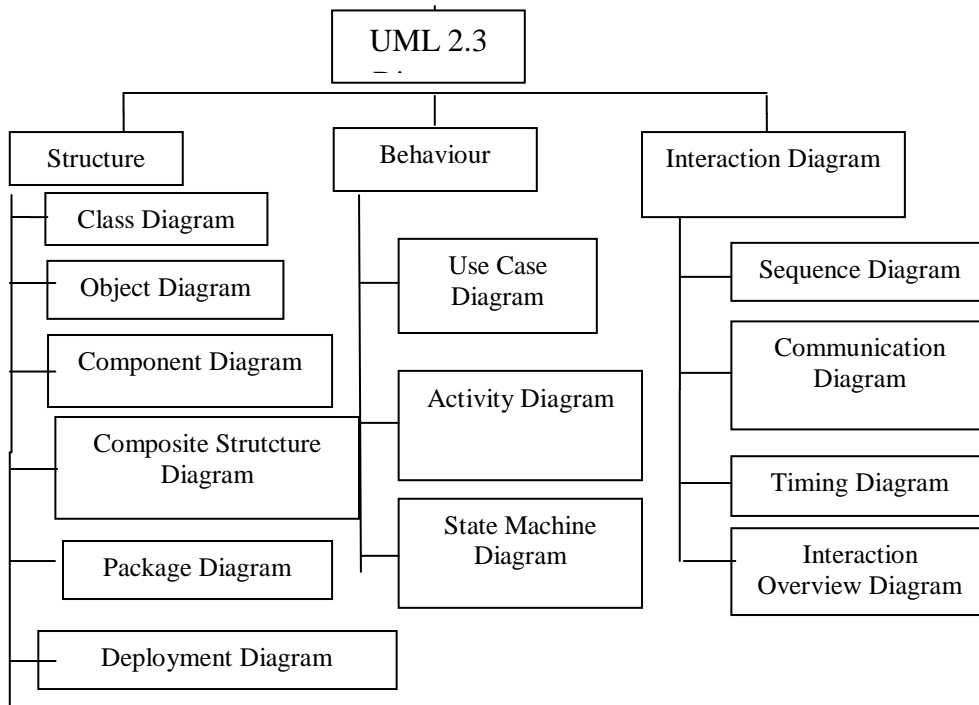
### **2.2.1 Sejarah UML**

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembang bahasa pemrograman berorientasi objek selanjutnya.

Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Usaha penyatuan ini banyak mengambil dari metodologi-metodologi yang berkembang saat itu. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep seperti konsep *Object Modelling Technique (OMT)* dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities, Collaboratos (CRC)* dari Rebecca Wirfs-Brock (1990), konsep pemikiran Ivar Jacobson, dan beberapa konsep lainnya dimana James R. Rumbaigh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama *Rational Software Corporation* menghasilkan bahasa yang disebut dengan *Unified Modelling Language (UML)* (Rosa & M.Shalahudin, 2014) .

### **2.2.2 Diagram Unified Modelling Language (UML)**

Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar 2.7 dibawah.



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

**Gambar 2.7 Pembagian Kategori**

Berikut ini penjelasan singkat dari pembagian kategori tersebut :

- a. *Structure diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- b. *Behavoiur diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.



c. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.



### 2.2.3 Use Case Diagram


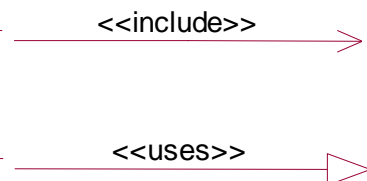
*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Menggambarkan sekelompok Use cases dan aktor yang disertai dengan hubungan diantaranya. Diagram Use case ini menjelaskan dan menerangkan kebutuhan atau requirement yang diinginkan user,(Andrianof, 2017). Use case atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat(Saiful & Ambarita, 2017).

Diagram aktivitas atau activitydiagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak yang di jelaskan pada tabel 2.1 dibawah.

**Tabel 2.1 : Use case diagram**

Simbol	Deskripsi
Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit

 NewUseCase	atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor / actor   Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol yang dibuat buat aktor belum tentu merupakan orang; biasanya dinyatakan dengan menggunakan kata benda di awal frase nama aktor.
Asosiasi / association  - <u>asosiasi</u>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor
Ekstensi / extend  Extensi / extend →	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan

<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>	<p><b>Sumber :</b> <b>Rosa A.S dan M. Shalahun (2014)</b></p>
<p>Menggunakan / <i>include / uses</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p>	

Setiap *use case* dilengkapi dengan skenario. Skenario *use case* adalah alur jalannya proses *use case* dari sisi aktor dan sistem. Berikut adalah tabel format tabel 2.2 di bawah ini.

**Tabel 2.2 : Skenario *use case***

Aksi Aktor	Reaksi Sistem
Skenario Normal	
Skenario Alternatif	

Skenario *use case* dibuat per *use case* terkecil, misalkan untuk generalisasi maka skenario yang dibuat adalah *use case* yang lebih khusus. Skenario normal adalah skenario yang berjalan dengan normal tanpa terjadi kesalahan atau *error*. Sedangkan skenario alternatif adalah skenario bila sistem tidak berjalan normal, atau mengalami *error*.

Skenario normal dan skenario alternatif dapat lebih dari satu. Alur dari skenario inilah yang nantinya menjadi dasar pembuatan diagram

sequen.

#### 2.2.4 *Class Diagram*

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (Andrianof, 2017).

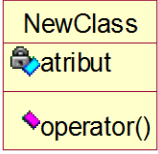






Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- a. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas sesuai dengan rancangan di dalam diagram kelas agar antara dokumentasi dan perangkat lunak sinkron. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai kebutuhan sistem, sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas didalam program perangkat lunak sesuai dengan perancangan diagram kelas. Berikut simbol-simbol yang ada pada tabel 2.3 di bawah ini.

**Tabel 2.3 : *Class diagram***

Simbol	Deskripsi
--------	-----------

<p>Kelas</p> 	<p>Kelas pada struktur system</p>
<p>Antarmuka / <i>interface</i></p>  <p>Interface</p>	<p>Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek</p>
<p>Asosiasi / <i>association</i></p> 	<p>Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i></p>
<p>Asosiasi berarah / <i>directed association</i></p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga dengan disertai <i>multiplicity</i></p>
<p>Generalisasi</p> 	<p>Relasi antar kelas dengan makna generalisasi spesialisasi (umum-khusus)</p>
<p>Kebergantungan / <i>dependency</i></p> 	<p>Relasi antar kelas dengan makna kebergantungan antar kelas</p>
<p>Agregasi / <i>aggregation</i></p> 	<p>Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)</p>

*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

### **2.2.5 Activity Diagram**


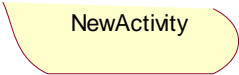



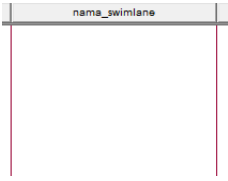
Activity Diagram menggambarkan workflow (aliran Kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktifitas yang dapat dilakukan oleh sistem (Rosa & M.Shalahudin, 2014).

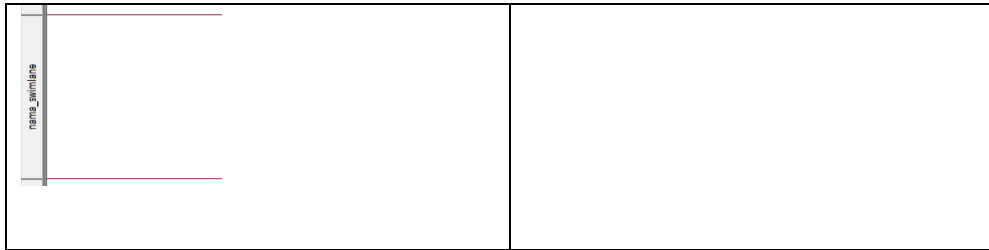
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis dimana setiap uruan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktifitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan diperangkat lunak.

**Tabel 2.4 : Activity diagram**

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane   Atau	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi



*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

### 2.2.6 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirim dan diterima antar objek (Andrianof, 2017).





Diagram sekuen menggambarkan kelakuan objek pada *use case* dan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat sekuen diagram juga dibutuhkan untuk melihat skenario yang ada pada *use case*.




Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak. Berikut adalah simbol-simbol pada tabel 2.5 di bawah ini:

**Tabel 2.5: Sequence diagram**

Simbol	Deskripsi
Aktor	Orang, proses, atau sistem lain yang



 <p>Aktor</p> <p>Atau</p> <div style="border: 1px solid black; background-color: yellow; padding: 2px; display: inline-block;">nama aktor</div> <p>Tanpa waktu aktif</p> <p>Garis hidup / <i>lifeline</i></p>	<p>berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor</p>
<p>Objek</p> <div style="border: 1px solid black; background-color: yellow; padding: 2px; display: inline-block;">nama objek : nama kelas</div>	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>
<p>Pesan tipe create</p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>

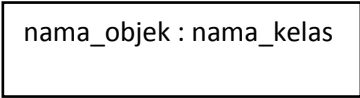

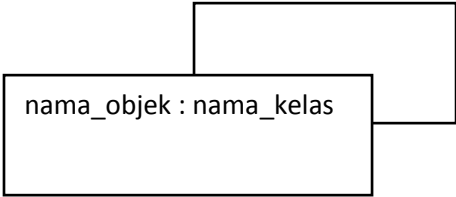
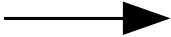
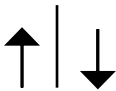
<p>Pesan tipe send</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah kepada objek yang dikirim</p>
<p>Pesan tipe return</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah kepada objek yang menerima kembalian</p>
<p>Pesan tipe destroy</p> 	<p>Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah padda objek yang diakhiri, sebaiknya jika ada create maka ada destroy</p>

*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

### **2.2.7 Collaboration Diagram**

Diagram komunikasi menggambarkan interaksi antar objek/bagian dalam bentuk urutan pengiriman pesan. Berikut adalah simbol-simbol yang ada pada tabel 2.6 di bawah ini.

**Tabel 2.6 Simbol pada *Communications Diagram***

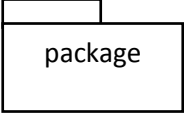


Simbol	Deskripsi
<p>Objek</p> 	Objek yang melakukan interaksi pesan
<p>Link</p> 	<p>Relasi antar-objek yang menghubungkan objek satu dengan yang lainnya atau dengan dirinya sendiri</p> 
<p>Arah pesan / <i>stimulus</i></p> 	<p>Arah pesan yang terjadi, jika pada suatu <i>link</i> ada dua arah pesan yang berbeda maka arah juga digambarkan dua arah pada dua sisi <i>link</i>.</p> 

Sumber : Rosa A.S dan M. Shalahuddin (2014)

### 2.2.8 Deployment Diagram

*Deployment* diagram memberikan gambaran dari arsitektur fisik perangkat lunak, perangkat keras, dan artefak dari sistem. Berikut simbol-simbol yang ada pada tabel 2.7 diagram *deployment*:

Tabel 2.7 Simbol pada *Deployment Diagram*

Simbol	Deskripsi
<p><i>Package</i></p>  <p>package</p>	<p><i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i>.</p>
<p><i>Node</i></p>	<p>Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah di definisikan sebelumnya pada diagram komponen.</p>
<p>Kebergantungan/ <i>dependency</i></p> 	<p>Ketergantungan antar <i>node</i>, arah panah mengarah pada <i>node</i> yang dipakai.</p>
<p><i>Link</i></p> 	<p>Relasi antar <i>node</i>.</p>


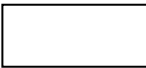
Sumber : Rosa A.S dan M. Shalahuddin (2014)

### 2.2.9 State Machine Diagram

Berikut ini komponen-komponen dasar yang ada dalam *state machine diagram* di jelaskan pada tabel 2.8 :

Tabel 2.8 Simbol pada *State Machine Diagram*

SIMBOL	KETERANGAN
--------	------------

<i>Start/Status Awal</i> <i>(Initial State)</i> ●	Keadaan awal pada saat system mulai hidup.
<i>End / Status Akhir</i> <i>(Final State)</i> ⊙	Keadaan akhir dari daur hidup suatu system.
<i>Event</i> 	Kegiatan yang menyebabkan berubahnya status mesin.
<i>State</i> 	Sistem pada waktu tertentu. <i>State</i> dapat berubah jika ada <i>event</i> tertentu yang memicu perubahan tersebut.

*Sumber : Rosa A.S dan M. Shalahuddin (2014)*

### 2.3 Tentang Pemodelan Dan Simulasi

Pemodelan dan simulasi didasari oleh pengetahuan komputer, matematika serta probabilitas dan statistik, walaupun tak dapat disangkal bahwa prosesnya sering masih dilakukan sebagai proses yang instingtif. Karena untuk membantu suatu model dari sistem dan melakukan simulasi tidak ada rumus-rumus yang eksak, maka yang dapat diberikan hanyalah petunjuk-petunjuk secara garis besar dalam bidang tersebut. Suatu model dan simulasi yang spesifik harus dilakukan terhadap suatu sistem yang bersangkutan, tetapi suatu analogi dapat diambil terhadap model yang telah ada sebagai acuan.

#### 2.3.1 Tujuan Pemodelan Dan Simulasi

Dalam pandangan sistem, pemodelan dan simulasi dapat digunakan untuk tujuan berikut:

1. Studi perilaku sistem kompleks, yaitu sistem dimana suatu solusi analitik tidak dapat dilakukan.

2. Membandingkan alternatif rancangan untuk suatu sistem yang tidak atau belum ada.
3. Studi pengaruh perubahan terhadap sistem yang ada dengan tanpa merubah sistem.
4. Memperkuat atau memverifikasi suatu solusi analitik.

Beberapa keuntungan simulasi sebagai berikut :

1. Simulasi mengizinkan keluwesan besar dalam pemodelan sistem kompleks, sehingga model simulasi dapat sangat *valid*.
2. Mudah membandingkan berbagai alternatif.
3. Kendali kondisi eksperimental.
4. Dapat mempelajari sistem dengan bingkai waktu yang sangat panjang.

Simulasi umumnya menghasilkan volume besar keluaran sehingga perlu untuk meringkaskan, sesuai analisis statistic.

### **2.3.2 Istilah Dalam Simulasi**

Pengertian sistem adalah sehimpunan unsur yang melakukan sesuatu kegiatan atau menyusun skema atau tata cara melakukan sesuatu kegiatan pengolahan (pemrosesan) untuk mencapai sesuatu atau beberapa tujuan, dan hal ini dilakukan dengan cara mengolah data dan/atau energi dan/atau barang (benda) di dalam jangka waktu tertentu guna menghasilkan informasi dan/atau energi dan/atau barang (benda).

Kejadian (*event*) adalah peristiwa sesaat yang dapat mengubah variabel keadaan sistem. Biasanya kejadian terjadi dengan selang-selang waktu acak. Suatu kejadian dapat mengubah harga dari atribut, mengubah entitas, atau memulai/mengakhiri suatu aktifitas.

Model adalah representasi dari dari suatu obyek, benda atau ide-ide dalam bentuk yang lain dengan entitasnya. Model berisi informasi tentang suatu sistem yang dibuat dengan tujuan untuk mempelajari perilaku sistem yang sebenarnya.

Variabel sistem adalah besaran yang nilainya berubah setiap saat baik secara kontinu maupun diskret dalam sistem. Elemen dari suatu sistem diwakili oleh atributnya, dan pengertian tentang variabel dalam pemodelan adalah suatu atribut tertentu yang bernilai tidak tetap, baik atribut elemen sistem, maupun atribut lingkungan sistem.

Keadaan (*state*) suatu sistem dinamis adalah himpunan minimal variabel-variabel sedemikian rupa sehingga pengetahuan tentang variabel tersebut pada waktu tertentu bersamaan dengan pengetahuan tentang masukan (*input*) secara lengkap akan menentukan perilaku (*behaviour*) sistem pada waktu tersebut. Jadi keadaan (*state*) suatu sistem adalah kumpulan variabel-variabel dan nilainya diperlukan untuk mensifatkan suatu sistem pada waktu tertentu.

Parameter sistem adalah besaran yang nilainya akan memberi sifat (karakteristik) tertentu dari suatu sistem. Nilai parameter dapat tetap (*time invariant*) berupa konstanta atau berubah dengan waktu (*time variant*).

Simulasi adalah proses implementasi model menjadi program komputer (*software*) atau rangkaian elektronik dan mengeksekusi *software* tersebut sedemikian rupa sehingga perilakunya menirukan atau menyerupai sistem nyata (realitas) tertentu untuk tujuan

mempelajari perilaku (*behaviour*) sistem, pelatihan (*training*) atau permainan (*gamin*) yang melibatkan sistem nyata (realitas).

### 2.3.3 Pengertian Model

Model adalah representasi dari suatu objek, benda atau ide-ide dalam bentuk yang lain dengan entitasnya.

Tujuan dari studi pemodelan adalah menentukan informasi (variabel dan parameter) yang dianggap penting untuk dikumpulkan, sehingga tidak ada model yang unik. Satu sistem dapat mempunyai berbagai model, tergantung dari sudut pandang dan kepentingan si pembuat model. Sebaliknya, model yang sama dapat dipakai untuk berbagai sistem, jika perilaku dari sistem-sistem tersebut sama.

### 2.3.4 Jenis-Jenis Model

Model dapat direpresentasikan dalam berbagai cara, tergantung tujuan dari studi. Beberapa jenis model dapat dikelompokkan sebagai berikut :

- a. Model Skala : adalah model yang dibuat biasanya dengan memeperkecil skala dari aslinya. Misalnya : model mobil, model pesawat terbang untuk uji terowongan angin (*wind tunnel*).
- b. Model Piktorial (Visual Grafis) : model yang dibuat dengan menggambar rancangan yang sebenarnya belum ada. Misalnya : designer menggambar model baju, arsitek menggambar rumah.



- c. Model Verbal : model yang penjelasannya dengan kata-kata. Misalnya: proses inflasi tergantung dari beberapa faktor ekonomi makro, dijelaskan dengan kata-kata baru dibuat diagram skematis.
- d. Model Skematis : model yang melukiskan unsur-unsur sistem dalam bentuk skema, petak-petak dan arus barang atau informasi. Model diagram dapat berupa diagram, seperti: diagram blok, DFD (*Data Flow Diagram*), CD (*Context Diagram*), Petri Net, Flowchart, PERT (*Program Evaluation and Review Technique*). Model dalam bentuk skema statik, seperti: tabel, bagan Gantt (*Gantt Chart*).
- e. Model Simbolik (Matematika) : model dalam bentuk persamaan matematika, seperti: persamaan diferensial, persamaan diferens, persamaan aljabar, persamaan logika, dan lain-lain.
- f. Model Komputer : model dalam bentuk program komputer (*source code*) yang ditulis menggunakan bahasa komputer tertentu, seperti: C, Pascal, Ada, dan lain-lain.

### **2.3.5 Definisi Simulasi**

Simulasi adalah suatu metode yang penting karena keunggulan dalam memperbaiki kinerja suatu sistem tanpa mengganggu kelancaran kerja sistem nyata (real system). Menurut (Riyanto, 2017) Simulasi adalah suatu metode yang penting karena keunggulan dalam memperbaiki kinerja suatu sistem tanpa mengganggu kelancaran kerja sistem nyata (real system). Sebuah model simulasi dikembangkan untuk mempelajari kerja sistem yang berkembang dari waktu ke waktu. Teknik simulasi diskrit adalah salah satu model simulasi pengoperasian sistem sebagai urutan peristiwa diskrit. Simulasi adalah

proses implementasi model menjadi program komputer (*software*) atau rangkaian elektronik dan mengeksekusi *software* tersebut sedemikian rupa sehingga perilakunya menirukan atau menyerupai sistem nyata (realitas) tertentu untuk tujuan mempelajari perilaku (*behaviour*) sistem, pelatihan (*training*), atau permainan (*gaming*). Jadi, simulasi adalah proses merancang model dari suatu sistem yang sebenarnya, mengadakan percobaan-percobaan terhadap model tersebut dan mengevaluasi hasil percobaan tersebut.

Simulasi merupakan suatu metode eksperimental dan terpakai untuk: menjelaskan perilaku sistem, membangun teori atau hipotesis yang mempertanggungjawabkan perilaku dari sistem yang diamati, memakai teori-teori untuk meramalkan perilaku sistem yang akan datang, yaitu pengaruh yang akan dihasilkan oleh perubahan-perubahan variabel dan parameter sistem atau perubahan operasinya.

Berdasarkan teknik atau metodologi simulasi, maka ada beberapa jenis simulasi, antara lain :

1. Simulasi *Monte Carlo*, tidak elemen waktu, digunakan untuk mengevaluasi ekspresi non-probabilistik (misalnya integral) menggunakan metode probabilistik, melibatkan banyak persoalan matematika.
2. Simulasi kemudi-jejak, secara ekstensif digunakan misalnya pada evaluasi unjuk kerja sistem komputer, misalnya algoritma penghalaman (*paging*).
3. Simulasi kejadian diskret, adalah suatu simulasi yang menggunakan model sistem kejadian diskret (*discrete event*), misalnya pada studi unjuk kerja sistem komputer digital, studi sistem antrian bank, dan lain-lain.

4. Simulasi dinamis kontinu, menggunakan model keadaan perubahan kontinu terhadap waktu, misalnya pada studi proses reaksi kimia, gerakan dinamis suatu kendaraan (*vehicle*) baik darat, laut ataupun darat.

### **2.3.6 Metode Monte Carlo**

Metode Monte Carlo merupakan metode analisis numerik yang melibatkan pengambilan sampel eksperimen bilangan acak. Salah satu model simulasi yang paling populer pada pengendalian persediaan adalah simulasi Monte Carlo. Model simulasi Monte Carlo merupakan bentuk simulasi probabilistik dimana solusi dari suatu masalah diberikan berdasarkan proses Randomisasi (acak). Proses acak ini melibatkan suatu distribusi probabilitas dari variabel-variabel data yang dikumpulkan berdasarkan data masa lalu maupun distribusi probabilitas teoritis. Bilangan acak digunakan untuk menjelaskan kejadian acak dan secara berurutan mengikuti perubahan-perubahan yang terjadi dalam proses simulasi (Putra & Hadi, 2016).

Karena algoritma ini memerlukan pengulangan (repetisi) dan perhitungan yang amat kompleks, metode Monte Carlo pada umumnya dilakukan menggunakan komputer, dan memakai berbagai teknik simulasi komputer. Algoritma Monte Carlo adalah metode Monte Carlo numerik yang digunakan untuk menemukan solusi problem matematis (yang dapat terdiri dari banyak variabel) yang sudah dipecahkan, misalnya dengan kalkulus integral, atau metode numerik lainnya.

### **2.3.7 Simulasi Monte Carlo**

Simulasi merupakan salah satu bentuk pemodelan terhadap sistem nyata untuk memudahkan mempelajari sistem kompleks. Fasilitas atau proses tersebut disebut dengan sistem, yang mana di dalam keilmuan digunakan untuk membuat asumsi-asumsi

bagaimana sistem tersebut bekerja, dimana asumsi-asumsi tersebut biasanya berbentuk hubungan matematik atau logika yang akan membentuk model yang digunakan untuk mendapatkan pemahaman bagaimana perilaku hubungan dari sistem tersebut (Lambertus, Jairus, & KallaYoseph, 2015).

Simulasi Monte Carlo dikategorikan sebagai metode pencuplikan (*sampling*) sebab masukan dibangkitkan secara acak dari distribusi probabilitas untuk menyimulasikan proses pencuplikan dari populasi sebenarnya. Kelebihan lain dari simulasi Monte Carlo adalah waktu eksekusi program lebih baik disemua iterasi (Lambertus et al., 2015).

### **2.3.8 Teknik Monte Carlo**

Teknik Monte Carlo merupakan pendekatan khusus yang sangat berguna untuk mensimulasikan situasi yang mengandung risiko sehingga diperoleh jawaban-jawaban perkiraan yang tidak dapat diperoleh dari penelitian-penelitian secara fisik atau dari penggunaan analisis matematika. Proses Monte Carlo dalam memilih angka acak berdasarkan distribusi probabilitas bertujuan untuk menentukan variabel acak melalui uji sampel dari distribusi probabilitas. Teknik ini dapat dikerjakan dengan alat bantu yaitu perangkat lunak dari komputer berupa lembar kerja (*Spreadsheet*) yang diprogram untuk membangkitkan bilangan random sesuai dengan yang dibutuhkan. Memunculkan nilai bilangan acak seperti : tabel bilangan acak, prosedur atau subrutin di dalam program.

Teknik simulasi Monte Carlo terbagi atas lima langkah sederhana yaitu sebagai berikut :

1. Menetapkan sebuah distribusi probabilitas bagi variabel penting. Ide dasar simulasi *Monte Carlo* adalah untuk membangkitkan nilai untuk variabel pada

model yang sedang diuji. Dalam sistem dunia nyata, sebagian besar variabel memiliki probabilitas alami

2. Membuat distribusi probabilitas kumulatif bagi setiap variabel. Untuk mengubah distribusi probabilitas biasa menjadi sebuah distribusi kumulatif (*cumulative probability distribution*) merupakan pekerjaan yang mudah.
3. Menetapkan sebuah interval angka acak bagi setiap variabel. Setelah distribusi probabilitas kumulatif bagi setiap variabel yang digunakan dalam simulasi sudah diterapkan, maka diberikan serangkaian angka yang mewakili setiap nilai atau output yang mungkin, angka ini disebut sebagai interval angka acak (*random-number interval*).
4. Membangkitkan angka acak yang sedang diteliti melibatkan banyak percobaan simulasi, maka digunakan program komputer untuk membangkitkan angka acak. Jika simulasi dilakukan dengan perhitungan tangan, angka acak dapat diambil dari sebuah tabel angka acak.
5. Membangkitkan serangkaian percobaan. Hasil dari eksperimen dapat disimulasikan secara sederhana dengan memilih angka acak dari tabel angka acak.

### **2.3.9 Random Number**

Bilangan acak atau bilangan random adalah suatu bilangan yang tidak dapat diprediksi kemunculannya disebut dengan bilangan acak atau random. Ada beberapa algoritma pembangkit bilangan *random* atau *pseudo random number* yang digunakan seperti LCG (*Linear Congruential Generators*), MRNG (*Multiplicative Random Number Generator*). LCG merupakan metode yang digunakan untuk membangkitkan bilangan

random dengan distribusi *uniform*. MRNG adalah metode pembangkitan bilangan random berupa bilangan-bilangan prima. Ketika membangkitkan *pseudo-random numbers* terdapat kesalahan tertentu seperti :

1. Angka yang dihasilkan tidak berdistribusi seragam.
2. Angka yang dihasilkan adalah *discrete-valued* sebagai ganti nilai kontinyu.
3. Rata-rata angka yang dihasilkan bernilai adalah terlalu tinggi atau terlalu rendah.
4. Tidak independen, misalnya masih terdapat nilai auto korelasi antar angka-angka yang dihasilkan.

Suatu bilangan random harus *replicable*. Pembangkitan bilangan random memiliki beberapa metode, salah satunya yaitu *Conruential Method (LCM)*. LCM menghasilkan suatu urutan bilangan bulat  $x_1, x_2, \dots$ , antara 0 dan  $m-1$  menurut hubungan berulang berikut ini :

$$X_{i+1} = (a.X_i + C) \bmod m, \text{ dengan } i = 0, 1, 2, \dots$$

Dimana :

a : Konstanta perkalian

$X_i$  : Nilai awal yang ditentukan

C : Kenaikan

Mod : Modulus

Jika  $C \neq 0$ , maka dikatakan sebagai *Mixed Congruential Method*. Ketika  $C = 0$ , maka dikatakan sebagai *Multiplicative Congruential Method*. Bilangan random antara 0 dan 1 dihasilkan oleh :  $R_i = X_i / m, I = 0,1,2..$

Contoh Soal sebagai berikut, Carilah 3 bilangan acak menggunakan metode mixed congruent, dengan nilai awal  $Z_0=12357, a=19, c=237, m=128$ . Gunakan tingkat ketelitian 4 digit di belakang koma.

Bilangan acak 1

$$\begin{aligned} Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\ &= (19 \cdot 12357 + 237) \bmod 128 \\ &= 235020 \bmod 128 \\ &= 12 \text{ (Random Number)} \end{aligned}$$

Random Variate :

$$\begin{aligned} R_1 &= Z_{i+1} / m \\ &= 12 / 128 \\ &= \mathbf{0.0938} \end{aligned}$$

Bilangan acak 2

$$\begin{aligned}
 Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\
 &= (19 \cdot 12 + 237) \bmod 128 \\
 &= 465 \bmod 128 \\
 &= 81 \text{ (Random Number)}
 \end{aligned}$$

Random Variate :

$$\begin{aligned}
 R_2 &= Z_{i+1} / m \\
 &= 81 / 128 \\
 &= \mathbf{0.6328}
 \end{aligned}$$

Bilangan acak 3

$$\begin{aligned}
 Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\
 &= (19 \cdot 81 + 237) \bmod 128 \\
 &= 1776 \bmod 128 \\
 &= 112 \text{ (Random Number)}
 \end{aligned}$$

$$\begin{aligned}
 R_3 &= Z_{i+1} / m \\
 &= 112 / 128 \\
 &= \mathbf{0.875}
 \end{aligned}$$

Yang terpenting adalah bahwa pembangkitan bilangan random menghasilkan R1, R2 ... mendekati keseragaman dan independen. Pertimbangan harus pula diberikan kepada



*maximum density* dan *maximum period*. *Maximum density* berarti bahwa nilai-nilai dihasilkan oleh  $R_i$ ,  $i=1,2 \dots$  tidak memberikan gap yang cukup besar  $[0,1]$ . Ini dapat dicapai dengan pembuatan  $m$  yang besar. Untuk menghindari *cycling* (perulangan angka-angka yang dihasilkan) *generator* perlu mempunyai *possible periode* yang cukup besar. Ini biasa dicapai oleh pemilihan  $a,c,m$ , dan  $X_0$  yang sesuai.

## 2.4 Pemograman Web

Tahap-tahap berikut ini merupakan pengertian teori dasar yang berhubungan dengan kasus yang akan dianalisis yaitu sebagai berikut :

### 2.4.1 PHP

PHP *Hypertext Preprocessor* atau disingkat dengan PHP adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan PHP harus menggunakan web server.

PHP juga dapat diintegrasikan dengan HTML, JavaScript, JQuery, Ajax. Namun, pada umumnya PHP lebih banyak digunakan bersamaan dengan *file* bertipe HTML. Dengan menggunakan PHP Anda bisa membuat *website powerful* yang dinamis dengan disertai manajemen *database*-nya. Selain itu juga penggunaan PHP yang sebagian besar dapat jalan di banyak *platform*, menjadi salah satu alasan kenapa Anda harus menguasai PHP untuk menjadi *web development* hebat.

Xampp merupakan paket server web PHP dan database MySQL, yang paling populer di kalangan pengembangan web menggunakan PHP dan MySQL sebagai Databasenya (Saiful & Ambarita, 2017).

### 2.4.2 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (*Situs personal*). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (*FI*), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilis kode sumber ini menjadi sumber terbuka, maka banyak pemrograman yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis 2.0 ini, *interpreter* PHP sudah diimplementasikan dalam program C. Di dalam rilis ini juga ikut disertakan modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Kemudian pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang *interpreter* PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis *interpreter* baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang ***PHP: Hypertext Preprocessing***.

Pada pertengahan tahun 1999, Zend merilis *interpreter* PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi *web* kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi. PHP adalah Salah satu aplikasi yang digunakan untuk memudahkan dalam melakukan pengelolaan database *Mysql* ( Buana, 2014).

### 2.4.3 Kelebihan PHP

PHP berbasis *server side scripting*. PHP sendiri dapat melakukan tugas-tugas yang dilakukan dengan mekanisme CGI seperti mengambil, mengumpulkan data dari *database*, meng-*generate* halaman dinamis, atau bahkan menerima dan mengirim *cookie*. Dan yang menjadi keutamaan PHP itu sendiri adalah PHP bisa digunakan di berbagai *operating system*, diantaranya Linux, Unix, Windows, Mac OsX, RISC OS, dan *operating system* lainnya.

#### **2.4.4 Kekurangan PHP**

Komunitas yang banyak tentu membawa kompetensi yang ketat. Para web *developer* yang menguasai PHP tiap hari semakin bertambah. Mudah dibajak karena sifat PHP yang merupakan *interpreter*, *source code* dari aplikasi PHP dapat dengan mudah dimodifikasi dan diubah fungsinya. Terkesan kurang *prestigious*, yakni mudah dipelajari oleh programmer pemula membuat bahasa pemrograman PHP terkesan kurang *prestisius* jika dibandingkan dengan pemrograman web lain yang terkesan lebih sulit untuk digunakan.

### **2.5 Konsep Database**

Adapun pengertian dan penjelasan tentang konsep database akan diuraikan dalam sub-sub berikut ini.

#### **2.5.1 Pengertian Database**

Database adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap database mempunyai API tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya. Kumpulan data (elementer) yang secara logic berkaitan dalam mempresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi dalam system tertentu (N. P. Ade, 2016).

### 2.5.2 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Database management system) atau Relational Database Management System (RDBMS) DBMS yang Multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Putra & Hadi, 2016).

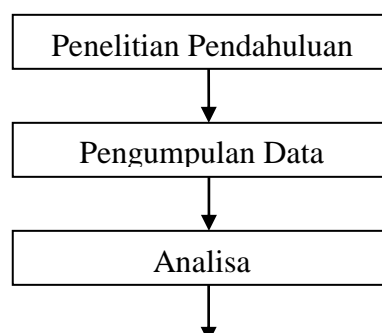
Kelebihan dari MySQL adalah gratis, handal, selalu di-*update* dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala. MySQL juga menjadi DBMS yang sering dibundling dengan *web server* sehingga proses instalasinya jadi lebih mudah.

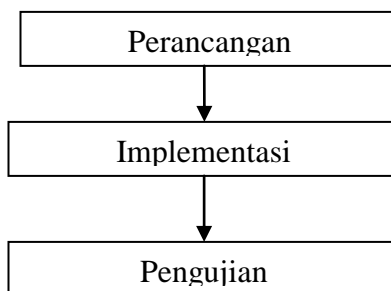
## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Kerangka Penelitian

Berikut merupakan kerangka penelitian yang akan dilakukan:





**Gambar 3.1 Kerangka Penelitian**

Gambar 3.1 merupakan kerangka penelitian yang dilakukan oleh peneliti, dimulai dari pendahuluan, pengumpulan data, analisa dan perancangan serta implementasi dan pengujian sistem secara bertahap dan berurutan. Adapun penjelasan lebih rinci dari tahapan-tahapan kerangka penelitian tersebut akan dibahas pada sub bab 3.2 di bawah ini.

## **3.2 Tahapan Penelitian**

Adapun tahapan - tahapan dalam penelitian ini adalah sebagai berikut:

### **3.2.1 Penelitian Pendahuluan**

Dalam aktifitas produksi pada *home-industry* keripik maco badarai Istiqomah ini, seringkali terjadi kesenjangan angka pendapatan setiap periodenya. Hal ini menyebabkan ketidakstabilan produksi seperti permintaan yang datang secara mendadak maupun stock barang yang terbatas yang menyebabkan aktifitas produksi tidak berjalan dengan baik.

Dengan menggunakan simulasi monte carlo mampu memberikan informasi untuk memprediksi pendapatan di *home-industry* keripik maco badarai Istiqomah, dimana produksi keripik maco pada *home-industry* keripik maco badarai Istiqomah menjadi lebih





Adapun Perangkat keras (*hardware*) yang dipakai dalam penelitian ini, yaitu:

1. Merk : ASUS X450EA
2. Processor : AMD E1-2500 APU with Radeon(TM)
3. RAM : 2 Giga Byte
4. Hardisk : 512 Giga Byte
5. Flash disk : SanDisk 32 Gigabyte

Sedangkan Perangkat lunak (*software*) yang dipakai dalam penelitian ini, yaitu:

1. Sistem Operasi *Windows 7*.
2. Microsoft Office 2010
3. *XAMPP v3.2.1*
4. *MySQL* version 4.4.14
5. Rational Rose
6. Notepad++ v6.8.3
7. Mozilla Firefox.

### **3.2.3 Analisa**

Untuk melakukan penelitian, penulis melakukan analisa terhadap data-data yang telah dikumpulkan serta menganalisa sistem yang akan dijalankan sebagai solusi dari perumusan masalah yang didapat. Setelah melakukan analisa, penulis juga melakukan perancangan terhadap sistem yang akan dibangun.

#### **3.2.3.1 Analisa Data**

Analisa data dibutuhkan agar penulis dapat memahami secara keseluruhan mengenai objek penelitian serta pemahaman secara teoritis maupun praktis mengenai teknik dan



metode yang akan digunakan dalam penyelesaian masalah-masalah yang telah dirumuskan pada *home-industry* Istiqomah.

### **3.2.3.2 Analisa Proses**

Setelah analisa data selanjutnya yaitu analisa proses, analisa proses dilakukan dengan simulasi menggunakan metode monte carlo. Analisa proses dibutuhkan agar penulis dapat menentukan tahapan demi tahapan pada perancangan sistem dan aplikasi yang akan digunakan.

### **3.2.3.3 Analisa Sistem**

Analisa sistem juga dibutuhkan agar penulis dapat mendeskripsikan alur kerja sistem berdasarkan kebutuhan penggunanya. Dengan adanya analisa sistem diharapkan dapat menciptakan suasana *user friendly* terhadap sistem yang akan dibangun.

## **3.2.4 Perancangan**

Setelah melakukan analisa terhadap data dan sistem, penulis juga melakukan perancangan terhadap sistem yang akan dibangun berdasarkan analisa sistem yang telah dilakukan.

### **3.2.4.1 Perancangan Model**

Adapun dalam melakukan perancangan sistem, penulis menggunakan *Unified Modeling Language (UML)* sebagai *tools* dalam mendeskripsikan alur kerja sistem.

Adapun penjelasan alur kerja sistem tersebut dideskripsikan dengan diagram-diagram *UML* diantaranya sebagai berikut :

### 1. *Usecase Diagram*

Diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor, dan digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi serta siapa saja yang berhak menggunakan fungsi-fungsi itu. Pada perancangan simulasi metode monte carlo di *home-industry Istiqomah* ini menggunakan satu aktor yang bisa mengoperasikan aplikasi yaitu manager. Adapun fasilitas yang diberikan untuk manager yaitu dapat mengoperasikan keseluruhan dari aplikasi tersebut.

### 2. *Class Diagram*

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel yang dimiliki oleh suatu kelas, dan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Pada perancangan aplikasi simulasi penjualan ini memiliki beberapa tabel sebagai penyimpanan data-data yang dibutuhkan dalam pengoperasiannya. Diagram ini akan memperlihatkan dengan jelas struktur dari setiap tabel yang ada serta menunjukkan hubungan antar tabel.

### 3. *Statechart Diagram*

*State Chart Diagram* akan menggambarkan cara memodelkan berbagai *state* (keadaan) keberadaan objek, digunakan untuk memodelkan lebih dinamis perilaku dari sistem.

### 4. *Activity Diagram*

*Activity Diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

- a. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.
- b. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).
- c. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

#### 5. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

#### 6. *Colaboration Diagram*

*Collaboration diagram* adalah perluasan dari objek diagram. Objek diagram menunjukkan objek-objek dan hubungannya satu dengan yang lain. *Collaboration diagram* menunjukkan *message-message* objek yang dikirimkan satu sama lain. Antara *collaboration diagram* dan *sequence diagram* bisa saling mengisi. Dengan demikian pada *collaboration diagram* kita bisa tambahkan nomor urut pada label sebuah *message* untuk menunjukkan urutan informasi. Titik dua (:) perlu digunakan untuk memisahkan nomor dengan *message*.

### 7. *Deployment Diagram*

Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik.

### 3.2.5 Implementasi

Tahap perancangan aplikasi setelah tahap analisis selesai. Implementasi ke dalam aplikasi merupakan tahap memasukkan data-data yang telah diolah dan menerapkan model yang terbaik sehingga data tersebut siap untuk dioperasikan. Dalam hal ini penulis merancang dan membangun sistem pemodelan dan simulasi berbasis web. Untuk mengimplementasikan suatu sistem yang telah dirancang, maka diperlukan sebuah alat bantu komputer untuk mengoperasikan komputer itu sendiri yang memerlukan tiga buah komponen pendukung seperti *hardware*, *software*, dan *brainware*. Adapun fungsi dari masing-masing komponen dapat dijelaskan sebagai berikut :

#### 1. Perangkat Keras (*Hardware*)

*Hardware* yang digunakan untuk merancang atau menjalankan program aplikasi yang telah dibuat dalam satu unit komputer yang lengkap dengan CPU, *harddisk* sebagai media penyimpanan data yang akan diinstallkan pada komputer.

#### 2. Perangkat Lunak (*Software*)

Untuk menjalankan program aplikasi yang dirancang harus menggunakan beberapa *software* pendukung. *Software-software* pendukung yang harus dimiliki adalah mowes portable dan adobe dreamweaver yang berfungsi untuk merancang desain program aplikasi tersebut.

### 3. Manusia (*Brainware*)

*Brainware* merupakan operator yang berfungsi untuk mengoperasikan atau menjalankan sebuah sistem.

#### 3.2.6 Pengujian Sistem

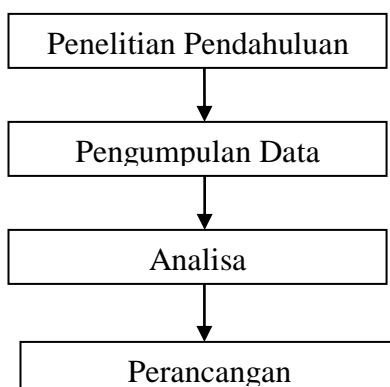
Setelah tahap implementasi selesai, penulis juga melakukan pengujian yang bertujuan untuk melakukan koreksi terhadap sistem yang telah dibangun berdasarkan tingkat efisiensi sistem sebagai solusi dalam pemecahan masalah-masalah yang telah dirumuskan. Pengujian aplikasi merupakan tahap akhir dalam melakukan *testing*, guna untuk mengetahui kesalahan dalam sistem. Pengujian dilakukan dengan melihat apakah aplikasi tersebut sudah berjalan dengan benar dan sesuai dengan perancangan yang dilakukan.

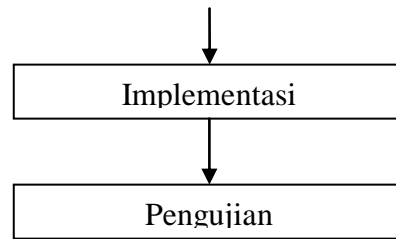
## BAB III

### METODOLOGI PENELITIAN

#### 3.3 Kerangka Penelitian

Berikut merupakan kerangka penelitian yang akan dilakukan:





**Gambar 3.1 Kerangka Penelitian**

Gambar 3.1 merupakan kerangka penelitian yang dilakukan oleh peneliti, dimulai dari pendahuluan, pengumpulan data, analisa dan perancangan serta implementasi dan pengujian sistem secara bertahap dan berurutan. Adapun penjelasan lebih rinci dari tahapan-tahapan kerangka penelitian tersebut akan dibahas pada sub bab 3.2 di bawah ini.

### **3.4 Tahapan Penelitian**

Adapun tahapan - tahapan dalam penelitian ini adalah sebagai berikut:

#### **3.4.1 Penelitian Pendahuluan**

Dalam aktifitas produksi pada *home-industry* keripik maco badarai Istiqomah ini, seringkali terjadi kesenjangan angka pendapatan setiap periodenya. Hal ini menyebabkan ketidakstabilan produksi seperti permintaan yang datang secara mendadak maupun stock barang yang terbatas yang menyebabkan aktifitas produksi tidak berjalan dengan baik.

Dengan menggunakan simulasi monte carlo mampu memberikan informasi untuk memprediksi pendapatan di *home-industry* keripik maco badarai Istiqomah, dimana produksi keripik maco pada *home-industry* keripik maco badarai Istiqomah menjadi lebih baik dengan adanya pemecahan masalah dengan *simulasi* metode monte carlo ini, sehingga







10. Flash disk : SanDisk 32 Gigabyte

Sedangkan Perangkat lunak (*software*) yang dipakai dalam penelitian ini, yaitu:

8. Sistem Operasi *Windows 7*.
9. Microsoft Office 2010
10. *XAMPP v3.2.1*
11. *MySQL version 4.4.14*
12. Rational Rose
13. Notepad++ v6.8.3
14. Mozilla Firefox.

### **3.2.3 Analisa**

Untuk melakukan penelitian, penulis melakukan analisa terhadap data-data yang telah dikumpulkan serta menganalisa sistem yang akan dijalankan sebagai solusi dari perumusan masalah yang didapat. Setelah melakukan analisa, penulis juga melakukan perancangan terhadap sistem yang akan dibangun.

#### **3.2.5.1 Analisa Data**

Analisa data dibutuhkan agar penulis dapat memahami secara keseluruhan mengenai objek penelitian serta pemahaman secara teoritis maupun praktis mengenai teknik dan metode yang akan digunakan dalam penyelesaian masalah-masalah yang telah dirumuskan pada *home-industry Istiqomah*.

#### **3.2.5.2 Analisa Proses**

Setelah analisa data selanjutnya yaitu analisa proses, analisa proses dilakukan dengan simulasi menggunakan metode monte carlo. Analisa proses dibutuhkan agar penulis dapat menentukan tahapan demi tahapan pada perancangan sistem dan aplikasi yang akan digunakan.

### **3.2.5.3 Analisa Sistem**

Analisa sistem juga dibutuhkan agar penulis dapat mendeskripsikan alur kerja sistem berdasarkan kebutuhan penggunanya. Dengan adanya analisa sistem diharapkan dapat menciptakan suasana *user friendly* terhadap sistem yang akan dibangun.

## **3.2.6 Perancangan**

Setelah melakukan analisa terhadap data dan sistem, penulis juga melakukan perancangan terhadap sistem yang akan dibangun berdasarkan analisa sistem yang telah dilakukan.

### **3.2.6.1 Perancangan Model**

Adapun dalam melakukan perancangan sistem, penulis menggunakan *Unified Modeling Language (UML)* sebagai *tools* dalam mendeskripsikan alur kerja sistem. Adapun penjelasan alur kerja sistem tersebut dideskripsikan dengan diagram-diagram *UML* diantaranya sebagai berikut :

#### **8. Usecase Diagram**

Diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu

atau lebih aktor, dan digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi serta siapa saja yang berhak menggunakan fungsi-fungsi itu. Pada perancangan simulasi metode monte carlo di *home-industry* Istiqomah ini menggunakan satu aktor yang bisa mengoperasikan aplikasi yaitu manager. Adapun fasilitas yang diberikan untuk manager yaitu dapat mengoperasikan keseluruhan dari aplikasi tersebut.

#### 9. *Class Diagram*

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel yang dimiliki oleh suatu kelas, dan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Pada perancangan aplikasi simulasi penjualan ini memiliki beberapa tabel sebagai penyimpanan data-data yang dibutuhkan dalam pengoperasiannya. Diagram ini akan memperlihatkan dengan jelas struktur dari setiap tabel yang ada serta menunjukkan hubungan antar tabel.

#### 10. *Statechart Diagram*

*State Chart Diagram* akan menggambarkan cara memodelkan berbagai *state* (keadaan) keberadaan objek, digunakan untuk memodelkan lebih dinamis perilaku dari sistem.

#### 11. *Activity Diagram*

*Activity Diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

- d. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

- e. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).
- f. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

## 12. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

## 13. *Colaboration Diagram*

*Collaboration diagram* adalah perluasan dari objek diagram. Objek diagram menunjukkan objek-objek dan hubungannya satu dengan yang lain. *Collaboration diagram* menunjukkan *message-message* objek yang dikirimkan satu sama lain. Antara *collaboration diagram* dan *sequence diagram* bisa saling mengisi. Dengan demikian pada *collaboration diagram* kita bisa tambahkan nomor urut pada label sebuah *message* untuk menunjukkan urutan informasi. Titik dua (:) perlu digunakan untuk memisahkan nomor dengan *message*.

## 14. *Deployment Diagram*

Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan

jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik.

### 3.2.7 Implementasi

Tahap perancangan aplikasi setelah tahap analisis selesai. Implementasi ke dalam aplikasi merupakan tahap memasukkan data-data yang telah diolah dan menerapkan model yang terbaik sehingga data tersebut siap untuk dioperasikan. Dalam hal ini penulis merancang dan membangun sistem pemodelan dan simulasi berbasis web. Untuk mengimplementasikan suatu sistem yang telah dirancang, maka diperlukan sebuah alat bantu komputer untuk mengoperasikan komputer itu sendiri yang memerlukan tiga buah komponen pendukung seperti *hardware*, *software*, dan *brainware*. Adapun fungsi dari masing-masing komponen dapat dijelaskan sebagai berikut :

#### 4. Perangkat Keras (*Hardware*)

*Hardware* yang digunakan untuk merancang atau menjalankan program aplikasi yang telah dibuat dalam satu unit komputer yang lengkap dengan CPU, *harddisk* sebagai media penyimpanan data yang akan diinstallkan pada komputer.

#### 5. Perangkat Lunak (*Software*)

Untuk menjalankan program aplikasi yang dirancang harus menggunakan beberapa *software* pendukung. *Software-software* pendukung yang harus dimiliki adalah mowes portable dan adobe dreamweaver yang berfungsi untuk merancang desain program aplikasi tersebut.

#### 6. Manusia (*Brainware*)

*Brainware* merupakan operator yang berfungsi untuk mengoperasikan atau menjalankan sebuah sistem.

### 3.2.6 Pengujian Sistem

Setelah tahap implementasi selesai, penulis juga melakukan pengujian yang bertujuan untuk melakukan koreksi terhadap sistem yang telah dibangun berdasarkan tingkat efisiensi sistem sebagai solusi dalam pemecahan masalah-masalah yang telah dirumuskan. Pengujian aplikasi merupakan tahap akhir dalam melakukan *testing*, guna untuk mengetahui kesalahan dalam sistem. Pengujian dilakukan dengan melihat apakah aplikasi tersebut sudah berjalan dengan benar dan sesuai dengan perancangan yang dilakukan.

## **BAB IV**

### **ANALISA DAN PERANCANGAN**

#### **4.1 Analisa**

Untuk melakukan penelitian, penulis melakukan analisa terhadap data-data yang telah dikumpulkan serta menganalisa sistem yang akan dijalankan sebagai solusi dari perumusan masalah yang didapat. Setelah melakukan analisa, penulis juga melakukan perancangan terhadap sistem yang akan dibangun.

#### **4.1.1 Analisa Data**

Tahap analisa data merupakan tahap yang paling penting dalam pengembangan sebuah sistem, karena pada tahap inilah nanti akan dilakukan evaluasi kinerja, identifikasi terhadap masalah yang ada, rancangan sistem dan langkah-langkah yang dibutuhkan untuk perancangan yang diinginkan sampai pada analisis yang diharapkan.

Pada *home-industry* Istiqomah menjual berbagai macam jenis Keripik maco, sehingga penulis mengambil beberapa sampel jenis barang yang banyak diminati oleh konsumen.

#### **4.1.2 Pengumpulan Data**

Pengumpulan data dilakukan secara langsung observasi ke lapangan. Dalam hal penyelesaian dengan metode *monte carlo* yang akan dilakukan adalah menentukan penjualan beberapa barang yang banyak diminati oleh konsumen selama 6 bulan dan dilakukan pengolahan data pada tiap bulannya di *home-industry* Istiqomah sehingga dapat menghasilkan prediksi penjualan barang untu kedepannya. Pengumpulan data dari bulan Agustus sampai Januari yang sudah disalin di Microsoft Excel seperti pada gambar 4.1 dan gambar 4.2 di bawah ini.

No	Nama Toko	Agustus		September		Oktober		November		Desember		January		Jumlah	
		Stock	Permintaan	Stock	Permintaan	Stock	Permintaan	Stock	Permintaan	Stock	Permintaan	Stock	Permintaan	Stock	Permintaan
1	HNH Mart Unand	20	25	22	25	20	25	25	30	20	25	25	30	132	160
2	Citra Sungai Galang	15	20	20	25	17	20	20	25	20	23	15	20	107	133
3	Zahra Unand	15	17	10	20	20	21	15	17	15	16	20	25	95	116
4	Aciak Mart Siteba	15	18	10	17	20	27	20	25	15	18	15	20	95	125
5	Nella Cake Siteba	10	15	15	20	15	11	10	15	15	20	10	15	75	96
6	Mahkota Air Tawar	35	40	25	30	30	35	35	40	35	37	20	25	180	207
7	Dayu Mart	15	21	15	23	15	20	20	23	15	17	15	20	95	124
8	Daf Mart	25	30	20	22	15	18	20	21	16	18	15	20	111	129
9	Mahkota Tabiang	40	45	35	36	40	42	45	48	35	37	30	35	225	243
10	Yessie Tabing	20	22	20	21	15	17	20	24	22	25	20	25	117	134
11	Mahkota A.yani	15	17	15	17	20	22	15	17	17	20	15	20	97	113
12	HNH Lubuk Lintah	15	20	10	12	15	18	20	21	16	20	15	18	91	109
13	Citra Kuranji	15	20	10	15	15	20	20	23	15	17	15	20	90	115
14	Shirley	75	80	80	85	77	80	80	85	85	86	80	81	477	497
15	Citra Thamrin	20	23	15	20	15	20	15	16	10	11	15	18	90	108
16	Ch Nipah	15	16	15	18	13	15	15	18	15	18	10	15	83	100
17	Silungkang Abu Nawas	30	32	25	27	30	33	35	40	40	42	35	38	195	212
18	Ummi Auffa	15	17	10	13	15	20	20	21	15	16	10	15	85	102
19	Citra Andalas	15	20	15	18	13	15	15	17	10	12	15	17	83	99
20	Procom Lubuk Lintah	15	20	10	12	10	15	15	18	15	18	15	16	80	99

**Gambar4.1 Data Mentah Penjualan**

No	Nama Barang	Harga Jual
1	Rakik Maco Badarai	Rp. 15.000
2	Peyek Maco	Rp. 15.000
3	Peyek Kacang Kedelai	Rp. 15.000
4	Peyek Kacang	Rp. 15.000
5	Rakik Maco Koin	Rp. 15.000
6	Rakik Kacang Koin	Rp. 15.000

**Gambar4.2 Data Harga Jual**

#### 4.1.3 Analisa Proses

Analisa proses ini yang dicontohkan hanya yang ada di 8 toko, yang dapat dilihat diatas yang sudah diblok dengan blok bewarna abu-abu, dan menggunakan 6 bilangan random yang akan diolah dengan menggunakan metode monte carlo.

Simulasi *monte carlo* memiliki sifat dasar stokastik yang artinya metode ini berdasarkan pada penggunaan angka-angka yang bersifat acak dan kemungkinan untuk mengidentifikasi sebuah masalah, metode ini sebelumnya digunakan untuk menyelesaikan masalah kuantitatif dengan proses fisik, seperti pelemparan dadu atau pengocokan kartu untuk menurunkan sampel.

Adapun penyelesaian metode *monte carlo* adalah sebagai berikut :



- a. Menentukan tabel distribusi

Dalam hal ini *home-industry* Istiqomah menghitung perkiraan penjualan beberapa barang menurut pola distribusi sebagai berikut :

**Tabel 4.1 Penjualan Di Toko HNH Mart Unand**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	20	Rp. 300.000
2	September	22	Rp. 330.000
3	Oktober	20	Rp. 300.000
4	November	25	Rp. 375.000
5	Desember	20	Rp. 300.000
6	Januari	25	Rp. 375.000

**Tabel 4.2 Penjualan Di Toko Citra Sungai Galang**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	15	Rp. 225.000
2	September	20	Rp. 300.000

3	Oktober	17	Rp. 255.000
4	November	20	Rp. 300.000
5	Desember	20	Rp. 300.000
6	Januari	15	Rp. 225.000

**Tabel 4.3 Penjualan Di Toko Mahkota Air Tawar**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	35	Rp. 525.000
2	September	25	Rp. 375.000
3	Oktober	30	Rp. 450.000
4	November	35	Rp. 525.000
5	Desember	35	Rp. 525.000
6	Januari	20	Rp. 300.000

**Tabel 4.4 Penjualan Di Toko Daft Mart**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
----	--------------------------------	---------------------	----------------------

1	Agustus	25	Rp. 375.000
2	September	20	Rp. 300.000
3	Oktober	15	Rp. 225.000
4	November	20	Rp. 300.000
5	Desember	16	Rp. 240.000
6	Januari	15	Rp. 225.000

**Tabel 4.5 Penjualan Di Toko Mahkota Tabing**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	40	Rp. 600.000
2	September	35	Rp. 525.000
3	Oktober	40	Rp. 600.000
4	November	45	Rp. 675.000
5	Desember	35	Rp. 525.000
6	Januari	30	Rp. 450.000

**Tabel 4.6 Penjualan Di Toko Yessie Tabing**

No	Penjualan	Jumlah	Harga jual
----	-----------	--------	------------

	Barang (Bulan)	Penjualan	barang
1	Agustus	20	Rp. 300.000
2	September	20	Rp. 300.000
3	Oktober	15	Rp. 225.000
4	November	20	Rp. 300.000
5	Desember	22	Rp. 330.000
6	Januari	20	Rp. 300.000

**Tabel 4.7 Penjualan Di Toko Shirley**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	75	Rp. 1.125.000
2	September	80	Rp. 1.200.000
3	Oktober	77	Rp. 1.155.000
4	November	80	Rp. 1.200.000
5	Desember	85	Rp. 1.275.000
6	Januari	80	Rp. 1.200.000

**Tabel 4.8 Penjualan Di Toko Silungkang Abu Nawas**

No	Penjualan Barang (Bulan)	Jumlah Penjualan	Harga jual barang
1	Agustus	30	Rp. 450.000
2	September	25	Rp. 375.000
3	Oktober	30	Rp. 450.000
4	November	35	Rp. 525.000
5	Desember	40	Rp. 600.000
6	Januari	35	Rp. 525.000

- b. Menentukan distribusi kemungkinan dan kumulatif Pada distribusi penjualan barang dapat ditentukan distribusi kemungkinan dan distribusi kumulatif. Dibawah ini adalah distribusi kemungkinan dan distribusi kumulatif dari data yang diolah.

**Tabel 4.9 Distribusi Kemungkinan Dan Kumulatif Di Toko HNH Mart Unand**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	20	$20/132 = 0,1515$	0,1515
2	September	22	$22/132 = 0,1667$	0,3182
3	Oktober	20	$20/132 = 0,1515$	0,4697
4	November	25	$25/132 = 0,1894$	0,6591

5	Desember	20	$20/132 = 0,1515$	0,8106
6	Januari	25	$25/132 = 0,1894$	1

**Tabel 4.10 Distribusi Kemungkinan Dan Kumulatif di Toko Citra Sungai Galang**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	15	$15/107 = 0,1402$	0,1402
2	September	20	$20/107 = 0,1869$	0,3271
3	Oktober	17	$17/107 = 0,1589$	0,486
4	November	20	$20/107 = 0,1869$	0,6729
5	Desember	20	$20/107 = 0,1869$	0,8598
6	Januari	15	$15/107 = 0,1402$	1

**Tabel 4.11 Distribusi Kemungkinan Dan Kumulatif Di Toko Mahkota Air Tawar**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	35	$35/180= 0,1944$	0,1944
2	September	25	$25/180= 0,1389$	0,3333
3	Oktober	30	$30/180= 0,1667$	0,5
4	November	35	$35/180= 0,1944$	0,6944
5	Desember	35	$35/180= 0,1944$	0,8889
6	Januari	20	$20/180= 0,1111$	1

**Tabel 4.12 Distribusi Kemungkinan Dan Kumulatif Di Toko Daf Mart**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	25	$25/111= 0,2252$	0,2252
2	September	20	$20/111= 0,1802$	0,4054
3	Oktober	15	$15/111= 0,1351$	0,5405
4	November	20	$20/111= 0,1802$	0,7207
5	Desember	16	$16/111= 0,1441$	0,8649
6	Januari	15	$15/111= 0,1351$	1

**Tabel 4.13 Distribusi Kemungkinan Dan Kumulatif Di Toko Mahkota Tabiang**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	40	$40/225 = 0,1778$	0,1778
2	September	35	$35/225 = 0,1556$	0,3333
3	Oktober	40	$40/225 = 0,1778$	0,5111
4	November	45	$45/225 = 0,2$	0,7111
5	Desember	35	$35/225 = 0,1556$	0,8667
6	Januari	30	$30/225 = 0,1333$	1

**Tabel 4.14 Distribusi Kemungkinan Dan Kumulatif Di Toko Yessie Tabing**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	20	$20/117 = 0,1709$	0,1709
2	September	20	$20/117 = 0,1709$	0,3419
3	Oktober	15	$15/117 = 0,1282$	0,4701
4	November	20	$20/117 = 0,1709$	0,641
5	Desember	22	$22/117 = 0,188$	0,8291



6	Januari	20	$20/117 = 0,1709$	1
---	---------	----	-------------------	---

**Tabel 4.15 Distribusi Kemungkinan Dan Kumulatif Di Toko Shirley**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	75	$75/477 = 0,1572$	0,1572
2	September	80	$80/477 = 0,1677$	0,3249
3	Oktober	77	$77/477 = 0,1614$	0,4864
4	November	80	$80/477 = 0,1677$	0,6541
5	Desember	85	$85/477 = 0,1782$	0,8323
6	Januari	80	$80/477 = 0,1677$	1

**Tabel 4.14 Distribusi Kemungkinan Dan Kumulatif Di Toko Silungkang Abu Nawas**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif
1	Agustus	30	$30/195 = 0,1538$	0,1538
2	September	25	$25/195 = 0,1282$	0,2821
3	Oktober	30	$30/195 = 0,1538$	0,4359
4	November	35	$35/195 = 0,1795$	0,6154

5	Desember	40	$40/195 = 0,2051$	0,8205
6	Januari	35	$35/195 = 0,1795$	1

- c. Menentukan batasan dari data yang telah diolah diatas. Bilangan acak yang digunakan adalah bilangan acak 4 digit dikarenakan distribusi probabilitas penjualan dan probabilitas kumulatif memiliki 4 digit dibelakang koma.

**Tabel 4.17 Batasan Di Toko HNH Mart Unand**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batas Ri
1	Agustus	20	$20/132 = 0,1515$	0,1515	0-0,1515
2	September	22	$22/132 = 0,1667$	0,3182	0,1516-0,3182
3	Oktober	20	$20/132 = 0,1515$	0,4697	0,3183-0,4697
4	November	25	$25/132 = 0,1894$	0,6591	0,4698-0,6591
5	Desember	20	$20/132 = 0,1515$	0,8106	0,6592-0,8106
6	Januari	25	$25/132 = 0,1894$	1	0,8107-0,9997

**Tabel 4.18 Batasan Di Toko Citra Sungai Galang**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
----	-------	----------------	------------------------	----------------------	------------

1	Agustus	15	$15/107 = 0,1402$	0,1402	0-0,1402
2	September	20	$20/107 = 0,1869$	0,3271	0,1403-0,3271
3	Oktober	17	$17/107 = 0,1589$	0,486	0,3272-0,486
4	November	20	$20/107 = 0,1869$	0,6729	0,4861-0,6729
5	Desember	20	$20/107 = 0,1869$	0,8598	0,673-0,8598
6	Januari	15	$15/107 = 0,1402$	1	0,8599-1

**Tabel 4.19 Batasan Di Toko Mahkota Air Tawar**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
1	Agustus	35	$35/180 = 0,1944$	0,1944	0-0,1944
2	September	25	$25/180 = 0,1389$	0,3333	0,1945-0,3333
3	Oktober	30	$30/180 = 0,1667$	0,5	0,3334-0,5

4	November	35	$35/180= 0,1944$	0,6944	0,5001-0,6944
5	Desember	35	$35/180= 0,1944$	0,8889	0,6945-0,8889
6	Januari	20	$20/180= 0,1111$	1	0,889-1

**Tabel 4.20 Batasan Di Toko Daf Mart**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
1	Agustus	25	$25/111= 0,2252$	0,2252	0-0,2252
2	September	20	$20/111= 0,1802$	0,4054	0,2253-0,4054
3	Oktober	15	$15/111= 0,1351$	0,5405	0,4055-0,5405
4	November	20	$20/111= 0,1802$	0,7207	0,5406-0,7207
5	Desember	16	$16/111= 0,1441$	0,8649	0,7208-0,8649
6	Januari	15	$15/111= 0,1351$	1	0,865-1

**Tabel 4.21 Batasan Di Toko Mahkota Tabiang**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
1	Agustus	40	$40/225= 0,1778$	0,1778	0-0,1778
2	September	35	$35/225= 0,1556$	0,3333	0,1779-0,3333

3	Oktober	40	$40/225 = 0,1778$	0,5111	0,3334-0,5111
4	November	45	$45/225 = 0,2$	0,7111	0,5112-0,7111
5	Desember	35	$35/225 = 0,1556$	0,8667	0,7112-0,8667
6	Januari	30	$30/225 = 0,1333$	1	0,8668-1

**Tabel 4.22 Batasan Di Toko Yessie Tabing**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
1	Agustus	20	$20/117 = 0,1709$	0,1709	0-0,1709
2	September	20	$20/117 = 0,1709$	0,3419	0,1710-0,3419
3	Oktober	15	$15/117 = 0,1282$	0,4701	0,342-0,4701
4	November	20	$20/117 = 0,1709$	0,641	0,4702-0,641
5	Desember	22	$22/117 = 0,188$	0,8291	0,6411-0,8291
6	Januari	20	$20/117 = 0,1709$	1	0,8292-1

**Tabel 4.23 Batasan Di Toko Shirley**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
----	-------	----------------	------------------------	----------------------	------------

1	Agustus	75	$75/477 = 0,1572$	0,1572	0-0,1572
2	September	80	$80/477 = 0,1677$	0,3249	0,1573-0,3249
3	Oktober	77	$77/477 = 0,1614$	0,4864	0,325-0,4864
4	November	80	$80/477 = 0,1677$	0,6541	0,4864-0,6541
5	Desember	85	$85/477 = 0,1782$	0,8323	0,6542-0,8323
6	Januari	80	$80/477 = 0,1677$	1	0,8324-1

**Tabel 4.24 Batasan Di Toko Silungkang Abu Nawas**

No	Bulan	Jumlah terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batasan Ri
1	Agustus	30	$30/195 = 0,1538$	0,1538	0-0,1538
2	September	25	$25/195 = 0,1282$	0,2821	0,1539-0,2821
3	Oktober	30	$30/195 = 0,1538$	0,4359	0,2822-0,4359
4	November	35	$35/195 = 0,1795$	0,6154	0,436-0,6154
5	Desember	40	$40/195 = 0,2051$	0,8205	0,6155-0,8205
6	Januari	35	$35/195 = 0,1795$	1	0,8205-1

- d. Menentukan angka random Penarikan *random number* dilakukan dengan rumus LCM, sehingga didapatkan berapa banyak penjualan barang dalam tiap bulannya berturut-turut. Penarikan angka random untuk simulasi ini adalah sebagai berikut:

$$X_{i+1} = (a \cdot X_i + c) \bmod m$$

Dimana :

$X_i$  : bilangan awal yang ditentukan

$a$  : konstanta perkalian

$c$  : konstanta penambahan

$\bmod$  : modulus

$m$  : batasan nilai bilangan acak

$$A=19, Z_0=12357, c=237, m=128$$

$$Z_i = (a \times Z_i + c) \bmod m$$

$$Z_1 = (19 \times 12357 + 237) \bmod 128 \qquad R_1 = 12 / 128$$

$$= 235.020 \bmod 128 \qquad = 0,0938$$

$$= 12$$

$$Z_2 = (19 \times 12 + 237) \bmod 128 \qquad R_2 = 81 / 128$$

$$= 465 \bmod 128 \qquad = 0,6328$$

$$= 81$$

$$Z_3 = (19 \times 81 + 237) \bmod 128 \qquad R_3 = 112 / 128$$

$$= 1776 \bmod 128 \qquad = 0,8750$$

$$= 112$$

$$Z_4 = (19 \times 112 + 237) \bmod 128 \qquad R_4 = 61 / 128$$

$$= 2365 \text{ mod } 128$$

$$= 0,4766$$

$$= 61$$

$$Z_5 = (19 \times 61 + 237) \text{ mod } 128$$

$$R_5 = 116 / 128$$

$$= 1396 \text{ mod } 128$$

$$= 0,9063$$

$$= 116$$

$$Z_6 = (19 \times 116 + 237) \text{ mod } 128$$

$$R_6 = 9 / 128$$

$$= 2441 \text{ mod } 128$$

$$= 0,0703$$

$$= 9$$

- e. Membuat simulasi dan rangkaian percobaan dari hasil pengambilan random number kemudian dapat disusun suatu tabel tiap bulan penjualan barang pada *home-industry* Istiqomah. Untuk menentukan tabel hasil simulasi dapat disesuaikan pada tabel interval atau batasan.

**Tabel 4.25 Tabel Percobaan Di Toko HNH Mart Unand**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	20	Rp. 300.000



September	0,6328	25	Rp. 375.000
Oktober	0,8750	25	Rp. 375.000
Novemer	0,4766	25	Rp. 375.000
Desember	0,9063	25	Rp. 375.000
Januari	0,0703	20	Rp. 300.000
Jumlah			Rp. 2.100.000
Prediksi : jumlah terjual / 6		23	

**Tabel 4.26 Tabel Percobaan Di Toko Citra Sungai Galang**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	15	Rp. 225.000
September	0,6328	20	Rp. 300.000
Oktober	0,8750	15	Rp. 225.000

Novemer	0,4766	17	Rp. 255.000
Desember	0,9063	15	Rp. 225.000
Januari	0,0703	15	Rp. 225.000
Jumlah			Rp. 1.455.000
Prediksi : jumlah terjual / 6		16	

**Tabel 4.27 Tabel Percobaan Di Toko Mahkota Air Tawar**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	35	Rp. 525.000
September	0,6328	35	Rp. 525.000
Oktober	0,8750	35	Rp. 525.000
Novemer	0,4766	30	Rp. 450.000
Desember	0,9063	20	Rp. 300.000
Januari	0,0703	35	Rp. 525.000
Jumlah			Rp. 2.850.000
Prediksi : jumlah terjual / 6		32	

**Tabel 4.28 Tabel Percobaan Di Toko Daf Mart**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	25	Rp. 375.000
September	0,6328	20	Rp. 300.000
Oktober	0,8750	15	Rp. 225.000
Novemer	0,4766	15	Rp. 225.000
Desember	0,9063	15	Rp. 225.000
Januari	0,0703	25	Rp. 375.000
Jumlah			Rp. 1.725.000
Prediksi : jumlah terjual / 6		19	

**Tabel 4.29 Tabel Percobaan Di Toko Mahkota Tabiang**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	40	Rp. 600.000
September	0,6328	45	Rp. 675.000
Oktober	0,8750	30	Rp. 450.000
Novemer	0,4766	40	Rp. 600.000
Desember	0,9063	30	Rp. 450.000
Januari	0,0703	40	Rp. 600.000
Jumlah			Rp. 3.375.000
Prediksi : jumlah terjual / 6		38	

**Tabel 4.30 Tabel Percobaan Di Toko Yessie Tabiang**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga
-------	------------	----------------	----------------------------

			jual
Agustus	0,0930	20	Rp. 300.000
September	0,6328	20	Rp. 300.000
Oktober	0,8750	20	Rp. 300.000
Novemer	0,4766	20	Rp. 300.000
Desember	0,9063	20	Rp. 300.000
Januari	0,0703	20	Rp. 300.000
Jumlah			Rp. 1.800.000
Prediksi : jumlah terjual / 6		20	

**Tabel 4.31 Tabel Percobaan Di Toko Shirley**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	75	Rp. 1.125.000
September	0,6328	80	Rp. 1.200.000
Oktober	0,8750	80	Rp. 1.200.000
Novemer	0,4766	77	Rp. 1.155.000

Desember	0,9063	80	Rp. 1.200.000
Januari	0,0703	75	Rp. 1.125.000
Jumlah			Rp. 7.005.000
Prediksi : jumlah terjual / 6		78	

**Tabel 4.32 Tabel Percobaan Di Toko Silungkang Abu Nawas**

Bulan	Angka acak	Jumlah terjual	Gi= jumlah terjual x Harga jual
Agustus	0,0930	30	Rp. 450.000
September	0,6328	40	Rp. 600.000
Oktober	0,8750	35	Rp. 525.000
Novemer	0,4766	35	Rp. 525.000
Desember	0,9063	35	Rp. 525.000

Januari	0,0703	30	Rp. 450.000
Jumlah			Rp. 3.075.000
Prediksi : jumlah terjual / 6		34	

- f. Setelah membuat simulasi jumlah penjualan yang didapat dari serangkaian percobaan dari hasil pengambilan random number kemudian dapat disusun suatu tabel tiap pendapatan kotor penjualan (simulasi) pada setiap toko di *home-industry* Istiqomah dapat dilihat pada tabel 4.33 di bawah ini.

**Tabel 4.33 Tabel Pendapatan**

NAMA TOKO	TOTAL	
	JUMLAH PENJUALAN	Gross Income

HNH Mart Unand	132	Rp. 2.100.000
Citra Sungai Galang	107	Rp. 1.455.000
Mahkota Air Tawar	180	Rp. 2.850.000
Daf Mart	111	Rp. 1.725.000
Mahkota Tabing	225	Rp. 3.375.000
Yessie Tabing	117	Rp. 1.800.000
Shirley	447	Rp. 7.005.000
Silungkang Abu Nawas	195	Rp. 3.075.000
TOTAL	1.514	Rp. 23.385.000

#### 4.1.4 Analisa Sistem

Analisis sistem merupakan suatu penguraian dari sebuah sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi, serta kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Analisa ini dilakukan untuk mengetahui apa saja yang dibutuhkan dalam perancangan sistem, sehingga sistem yang dibangun menjadi efektif dan efisien dalam implementasi nantinya.

Sistem pada dasarnya merupakan suatu susunan teratur dari sekumpulan kegiatan-kegiatan yang saling berhubungan antara satu dengan yang lainnya, dan prosedur-prosedur yang dilaksanakan saling berkaitan sehingga memudahkan untuk melakukan kegiatan utama dari suatu instansi atau organisasi.



## 4.2 Perancangan Model

Adapun dalam melakukan perancangan sistem, penulis menggunakan *Unified Modelling Language* (UML) sebagai *tools* dalam mendeskripsikan alur kerja sistem. Adapun penjelasan alur kerja sistem tersebut akan dideskripsikan dengan diagram UML diantaranya sebagai berikut:

### 4.2.1 Use Case Diagram

*Use Case* berisi abstraksi dari interaksi antara sistem dan aktor. *Use Case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use Case Diagram* menggambarkan bagaimana proses-proses yang dilakukan oleh aktor terhadap sebuah sistem. *Use Case* berisi apa yang dilakukan oleh sistem atau apa yang terjadi pada sistem, bukan bagaimana sistem melakukan. *Use case* diagram sendiri berguna untuk membantu dalam tiga bagian :

1. *Determining feature use case* dibuat dan dirancang sebagai awal untuk merancang sebuah sistem.
2. Komunikasi dengan *client*. Notasi yang sederhana membuat *use case* dapat digunakan dengan mudah oleh pengguna yang baru menggunakan diagram ini.
3. Perancangan kasus. Pengumpulan skenario untuk *use case* dilakukan perancangan kasus untuk uji skenario.

Tabel 4.34 Aktor Dari Sistem Aplikasi

No	Aktor	Keterangan
1	Admin (Pemilik toko)	Aktor yang bertugas untuk memonitoring dan mengelola semua system

Tabel 4.35 Skenario *use case login*

1. From Login Admin / Pemilik	
Identifikasi	
Aksi aktor	Reaksi Sistem
Nama	Login
Tujuan	Masuk kedalam sistem sebagai admin
Deskripsi	Proses login admin merupakan proses autentifikasi untuk menggunakan sistem sebagai admin atau pemilik
Alat	Admin
Usecase Yang Berkaitan	
Skenario Utama (Proses Berhasil)	
Mengisi from login	Mengautentifikasi data login dengan data admin atau pemilik pada database
	Bila cocok akan menampilkan halaman menu

	utama
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan data login tidak benar
<b>Mengisi Kembali Form Login</b>	Mengautentifikasi data login dengan data admin atau pemilik pada database
	Bila cocok akan menampilkan halaman menu utama
<b>Kondisi Akhir</b>	Admin dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai admin

**Tabel 4.36 Input Barang**

2. From Input Barang	
Identifikasi	
Aksi Aktor	Reaksi Sistem
<b>Nama</b>	Input Barang
<b>Tujuan</b>	Masuk kedalam sistem sebagai admin atau pemilik
<b>Deskripsi</b>	Proses Input Barang merupakan proses untuk menginputkan kode barang,nama barang,jumlah penjualan dan bulan input barang.
<b>Alat</b>	Admin atau pemilik

<b>Usecase Yang Berkaitan</b>	
<b>Skenario Utama (Proses Berhasil)</b>	
<b>Klik Tambah Informasi</b>	Admin atau pemilik dapat menambahkan barang
<b>Klik Hapus Informasi</b>	Admin atau pemilik dapat menghapus barang
<b>Klik Edit</b>	Admin atau Produk dapat mengedit data barang yang sudah ada
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan tambah barang gagal
<b>Klik Tambah Barang</b>	Admin atau Pemilik dapat menambahkan barang baru kembali
<b>Klik Edit Barang</b>	Admin atau Pemilik dapat mengedit kembali barang yang sudah ada
<b>Klik Hapus Barang</b>	Admin atau Pemilik dapat menghapus kembali barang yang sudah ada

**Tabel 4.37 Input Random**

<b>3. From Input Random</b>	
<b>Identifikasi</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
<b>Nama</b>	Input Random
<b>Tujuan</b>	Masuk kedalam sistem sebagai admin
<b>Deskripsi</b>	Proses Input Random merupakan proses untuk menginputkan nilai untuk mencari angka
<b>Alat</b>	Admin
<b>Usecase Yang Berkaitan</b>	
<b>Skenario Utama (Proses Berhasil)</b>	
<b>Mengisi Nilai a</b>	Admin harus menginputkan nilai a untuk mencari random
<b>Mengisi Nilai Zi</b>	Admin harus menginputkan nilai Zi untuk mencari random
<b>Mengisi Nilai c</b>	Admin harus menginputkan nilai c untuk mencari random
<b>Mengisi Nilai m</b>	Admin harus menginputkan nilai m untuk mencari random
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan input random gagal
<b>Mengisi Nilai a</b>	Admin harus menginputkan nilai a untuk mencari random
<b>Mengisi Nilai Zi</b>	Admin harus menginputkan nilai Zi untuk mencari random

<b>Mengisi Nilai c</b>	Admin harus menginputkan nilai c untuk mencari random
<b>Mengisi Nilai m</b>	Admin harus menginputkan nilai m untuk mencari random
<b>Kondisi Akhir</b>	Admin dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai admin

**Tabel 4.38 Input Penjualan**

<b>4. Input Penjualan</b>	
<b>Identifikasi</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
<b>Nama</b>	Input penjualan
<b>Tujuan</b>	Masuk kedalam sistem sebagai admin
<b>Deskripsi</b>	Proses Input Random merupakan proses input menginputkan jumlah penjualan barang pada setiap bulannya
<b>Alat</b>	Admin atau pemilik
<b>Usecase Yang Berkaitan</b>	
<b>Skenario Utama (Proses Berhasil)</b>	

<b>Klik Pilih Barang</b>	Admin atau pemilik dapat memilih barang yang akan di inputkan
<b>Klik Pilih Bulan</b>	Admin atau pemilik dapat memilih bulan penginputan barang
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan tambahan input penjualan gagal
<b>Klik Pilih Barang</b>	Admin atau pemilik dapat memilih barang yang akan diinputkan
<b>Klik Pilih Bulan</b>	Admin atau pemilik dapat memilih bulan penginputan barang
<b>Kondisi Akhir</b>	Admin dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai admin atau pemilik

Tabel 4.39 Hasil Perhitungan

<b>5. Hasil Perhitungan</b>	
<b>Identifikasi</b>	
<b>Aksi aktor</b>	<b>Reaksi Sistem</b>
<b>Nama</b>	Hasil Perhitungan
<b>Tujuan</b>	Masuk kedalam sistem sebagai admin

<b>Deskripsi</b>	Admin dapat melihat hasil perhitungan dari data yang telah diinputkan, berupa distribusi kumulatif, distribusi kemungkinan dan interval dari data tersebut
<b>Alat</b>	Admin
<b>Usecase Yang Berkaitan</b>	
<b>Skenario Utama (Proses Berhasil)</b>	
<b>Memilih Barang</b>	Admin dapat memilih nama barang yang akan ditampilkan hasil perhitungannya
<b>Cetak</b>	Admin dapat mencetak hasil laporan penjualan
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan hasil perhitungan gagal
<b>Memilih Barang</b>	Admin dapat memilih nama barang yang akan ditampilkan hasil perhitungannya
<b>Cetak</b>	Admin dapat mencetak hasil laporan penjualan
<b>Kondisi akhir</b>	Admin dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai admin

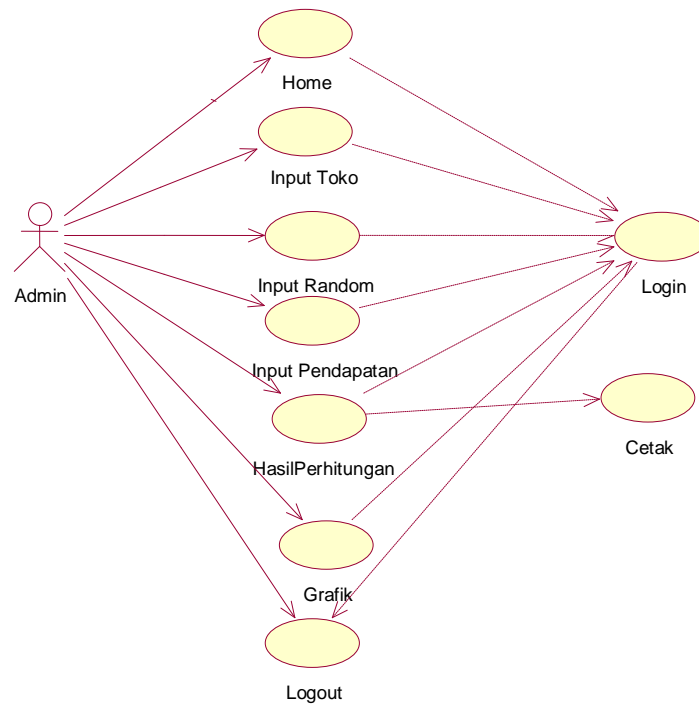
Tabel 4.40 Grafik

## 6. Grafik



<b>Identifikasi</b>	
<b>Aksi aktor</b>	<b>Reaksi Sistem</b>
<b>Nama</b>	Grafik
<b>Tujuan</b>	Masuk kedalam sistem sebagai admin
<b>Deskripsi</b>	Admin dapat melihat grafik penjualan dan prediksi penjualan
<b>Alat</b>	Admin
<b>Usecase Yang Berkaitan</b>	
<b>Skenario Utama (Proses Berhasil)</b>	
<b>Klik Grafik</b>	Admin dapat melihat grafik penjualan
<b>Skenario Alternatif (Proses gagal)</b>	
	Menampilkan pesan grafik gagal
<b>Klik Grafik</b>	Admin dapat melihat grafik penjualan
<b>Kondisi akhir</b>	Admin dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai admin

melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use case* merupakan instruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata user. Dapat dilihat pada gambar 4.3 di bawah ini.



**Gambar 4.3 Use Case Diagram**

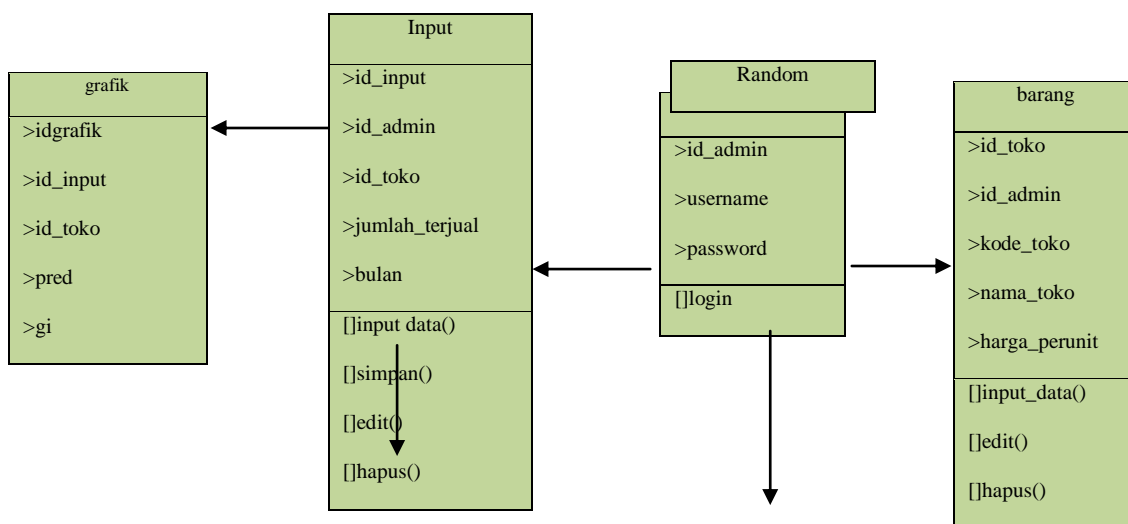
#### 4.2.2 Class Diagram

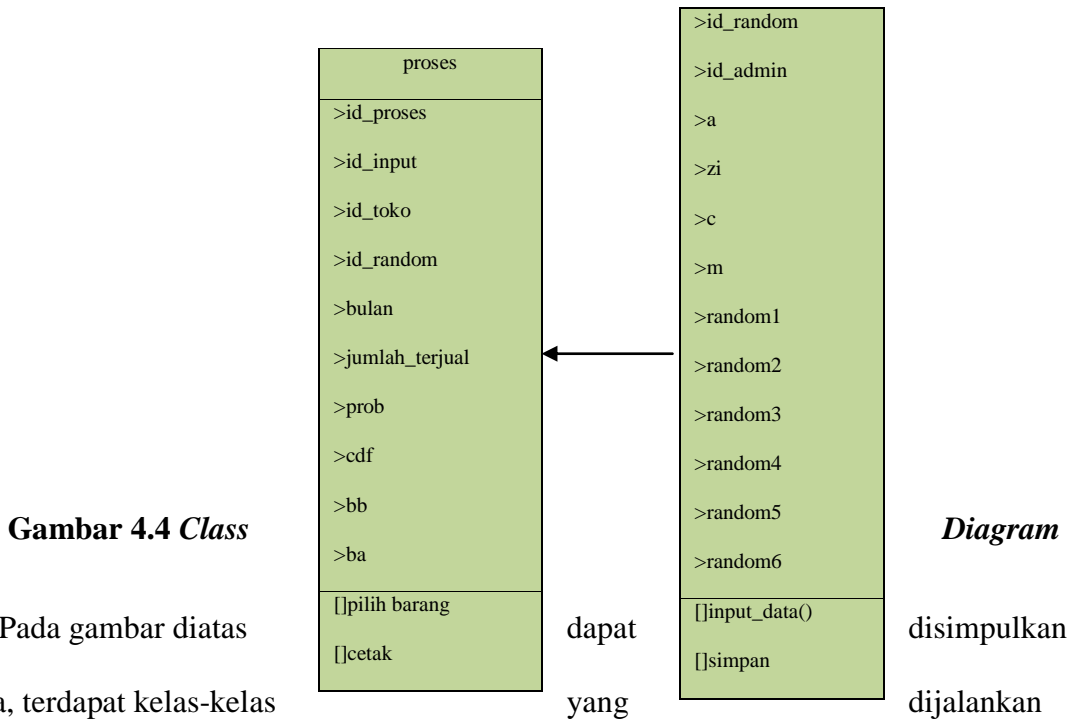
*Class Diagram* adalah kumpulan objek-objek yang mempunyai struktur umum, behavior umum, relasi umum, dan semantik atau kata yang umum. *Class Diagram* menggambarkan struktur dari suatu sistem yang disajikan dalam bentuk *class* beserta atribut-atribut dan hubungan antar *class*. Umumnya *class diagram* dari suatu sistem akan menggambarkan juga bagaimana struktur database yang dibutuhkan untuk membangun *point of sale*, yang dapat dilihat pada tabel 4.41 di bawah ini :

**Tabel 4.41 Keterangan Class Diagram**

No	Nama Class	Keterangan
1	Admin	Aktor yang bertugas untuk mengelola dan memonitoring system
2	Toko	Berisi hasil dari toko yang telah diinputkan
3	Random	Berisi random yang telah diinputkan pada system
4	Grafik	Berisi hasil dari prediksi
5	Input	Berisi data yang akan diinputkan ke dalam system
6	Proses	Berisi data yang diproses

Class dalam notasi UML digambarkan dengan kotak, yang membagi beberapa yaitu : *class*, *atribut* dan *operation* pada gambar 4.4 di bawah ini.





1. Tabel admin

Digunakan admin sebagai login yang berfungsi untuk mengentri, menedit dan menghapus data ( mengelola sistem ). Berikut ini adalah atribut-atribut yang ada pada tabel 4.42 di bawah ini.

**Tabel 4.42 Tabel Admin**

No	Field	Type	Keterangan
1	Id_admin	Int (11)	<i>Primary key</i>
2	Username	Varchar (20)	

3	Password	Varchar(20)	
---	----------	-------------	--

## 2. Tabel Barang

Tabel data yang digunakan untuk menginputkan kode barang, nama barang, penjualan dan bulan input barang. Adapun atribut-atribut yang terdapat pada tabel barang di deklarasikan dalam bentuk tabel 4.43 sebagai berikut :

**Tabel 4.43 Tabel Barang**

No	Field	Type	Keterangan
1	Id_toko	Int (11)	<i>Primary key</i>
2	Id_admin	Varchar (11)	
3	kode_toko	Varchar (3)	
4	Nama_toko	Int (20)	
5	Harga_perunit	Int(11)	

## 3. Tabel Grafik

Tabel grafik digunakan untuk menampilkan hasil laju prediksi pada tiap produk.

Berikut ini adalah atribut-atribut yang terdapat pada tabel 4.44 di bawah ini :

**Tabel 4.44 Tabel Grafik**

No	Field	Type	Keterangan
1	Idgrafik	Int (15)	<i>Primary key</i>
2	Id_input	Int (15)	
3	Id_toko	Int (15)	
4	Pred	Int (15)	
5	Gi	Double	

#### 4. Tabel Input

Tabel input digunakan untuk menampilkan hasil dari data yang telah kita inputkan sebelumnya pada tabel 4.45 dibawah ini.

**Tabel 4.45 Tabel Input**

No	Field	Type	Keterangan
1	Id_input	Int (15)	<i>Primary key</i>

2	Id_admin	Int (15)	
3	Id_toko	Int (15)	
4	Jumlah_terjual	Int (15)	
5	Bulan	Varchar (10)	

### 5. Tabel Proses

Tabel proses menampilkan hasil dari data yang telah diolah. Pada tabel ini akan muncul hasil perhitungan dari pengolahan data kita sebelumnya pada tabel 4.46 dibawah ini.

**Tabel 4.46 Tabel Proses**

No	Field	Type	Keterangan
1	Idproses	Int (20)	<i>Primary key</i>
2	Id_input	Int (15)	
3	Id_toko	Int (15)	
4	Id_random	Int(11)	
5	Bulan	Varchar (10)	
6	Jumlah_terjual	Int (20)	
7	Prob	Double	

8	Cdf	Double	
9	Bb	Double	
10	Ba	Double	

#### 6. Tabel Random

Tabel random akan menampilkan angka yang telah di inputkan untuk mencari random dan juga akan menampilkan random yang telah didapatkan pada tabel 4.47 di bawah ini.

**Tabel 4.47 Tabel Random**

No	Field	Type	Keterangan
1	id_random	Int (20)	<i>Primary key</i>
2	id_admin	Int (15)	
3	A	Int (5)	
4	Zi	Int (5)	
5	C	Int (5)	
6	M	Int (5)	



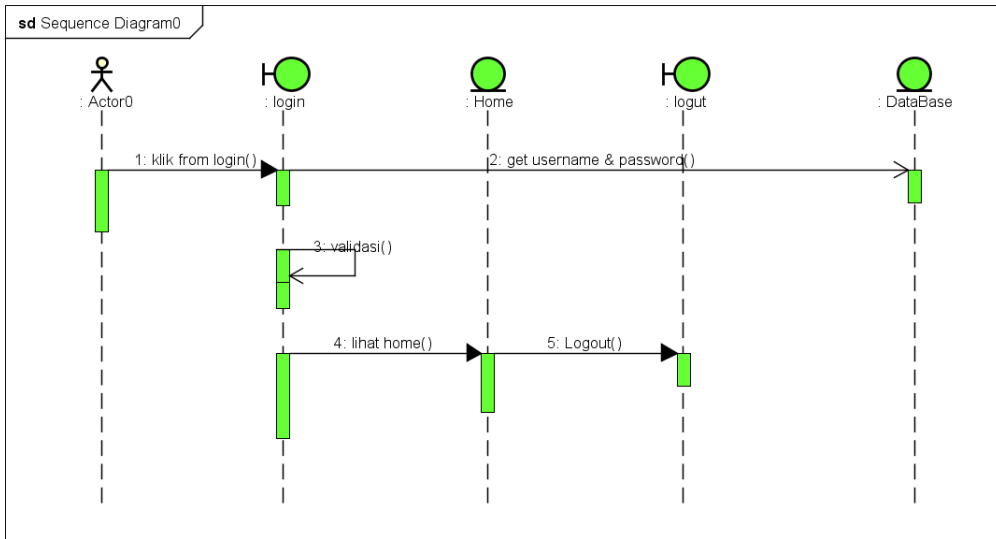
7	Random1	Double	
8	Random2	Double	
9	Random3	Double	
10	Random4	Double	
11	Random5	Double	
12	Random6	Double	

### 4.2.3 Sequence Diagram

*Sequence diagram* adalah suatu penyajian perilaku yang tersusun sebagai rangkaian langkah-langkah percontohan dari waktu ke waktu. *Sequence diagram* digunakan untuk menggambarkan arus pekerjaan.

#### 1. Sequence diagram Home admin

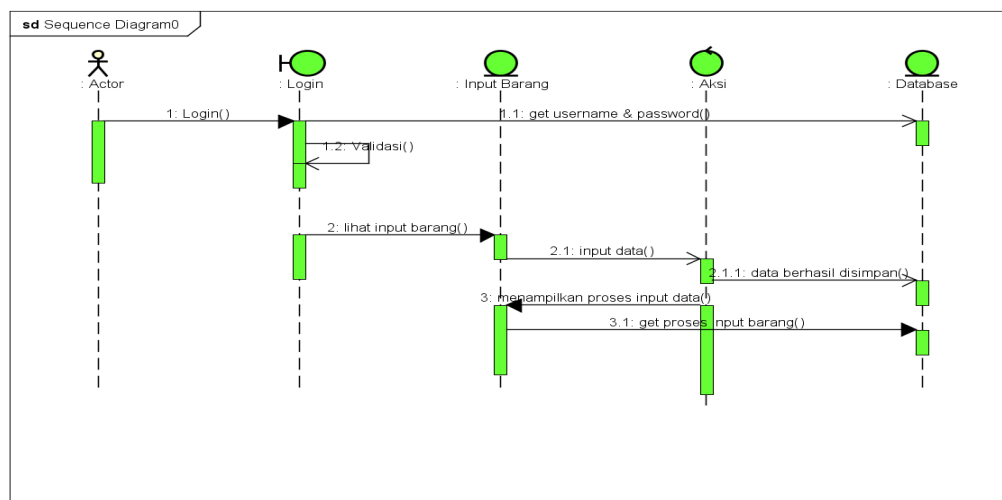
*Sequence diagram* tampilan home admin mendeklarasikan admin yang telah login dan sistem akan menampilkan halaman utama sistem. Adapun alur kerja dan interaksi disaat tampilan home dijalankan tampak seperti gambar 4.5 di bawah ini.



**Gambar 4.5 Sequence Diagram Home Admin**

2. Sequence Diagram Input Barang

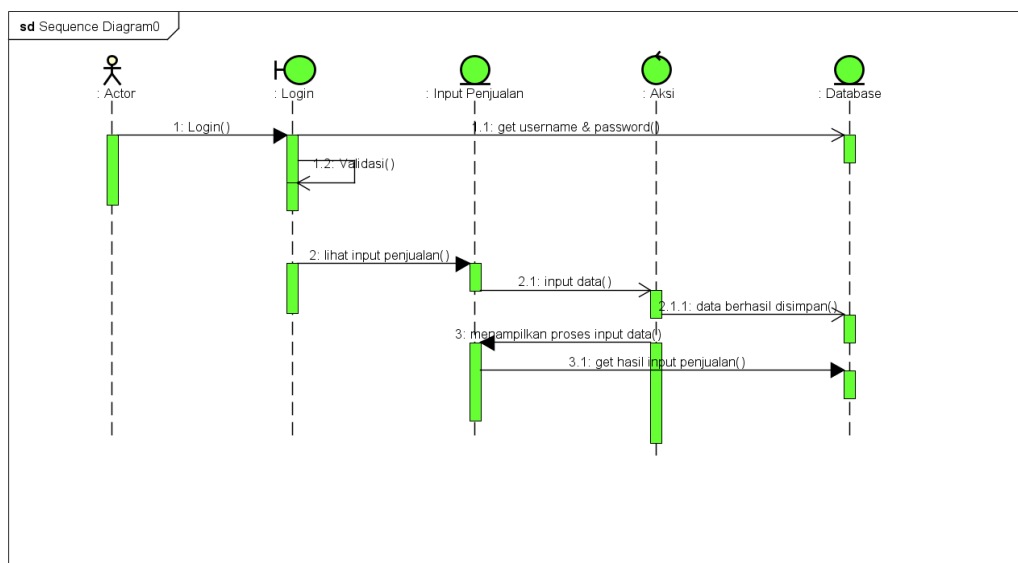
Sequence diagram yang menggambarkan proses input barang yang dilakukan oleh admin sistem. Adapun alur kerja dan interaksi saat admin melakukan aksi input data tampak seperti pada gambar 4.6 berikut :



**Gambar 4.6 Sequence Diagram Input Barang**

3. Sequence Diagram Input Penjualan

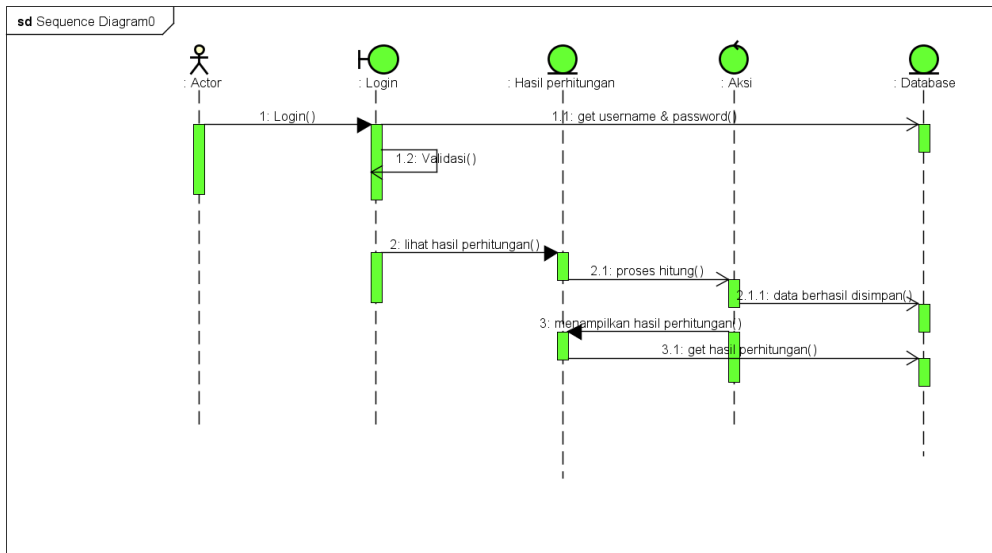
Sequence diagram input penjualan mendeklarasikan alur kerja admin pada input penjualan. Alur kerja dan interaksi saat admin melakukan input penjualan pada gambar 4.7 sebagai berikut :



**Gambar 4.7 Sequence Diagram Input Penjualan**

#### 4. Sequence Diagram Hasil Perhitungan

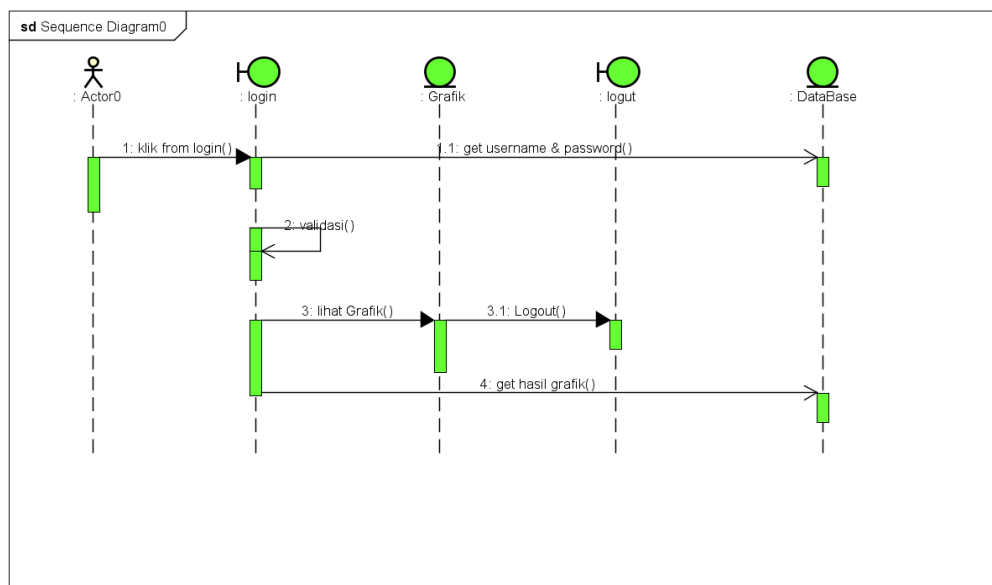
Sequence diagram hasil perhitungan mendeklarasikan alur kerja dan interaksi antara aktor dengan sistem dalam menampilkan hasil perhitungan. Adapun alur kerja dan interaksi saat admin melakukan aksi hasil perhitungan pada gambar 4.8 sebagai berikut :



**Gambar 4.8 Sequence Diagram Hasil Perhitungan**

5. Sequence Diagram Grafik

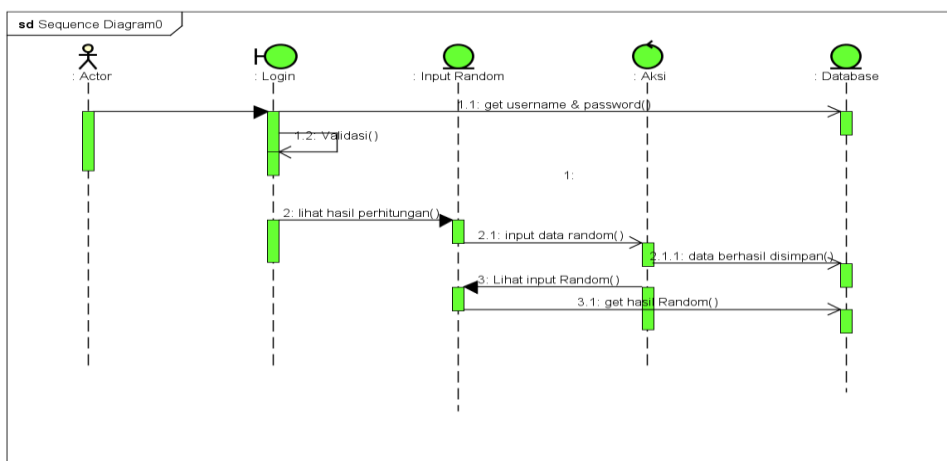
Sequence diagram grafik mendeklarasikan alur kerja dan interaksi antara admin dan sistem. Adapun alur kerja dan interaksi saat admin melakukan aksi melihat grafik tampak seperti gambar 4.9 di bawah ini :



**Gambar 4.9 Sequence Diagram Grafik**

6. Sequence Diagram Random

Sequence diagram grafik mendeklarasikan alur kerja dan interaksi antara admin dan sistem. Adapun alur kerja dan interaksi saat admin melakukan aksi melihat grafik tampak seperti gambar 4.10 di bawah ini :



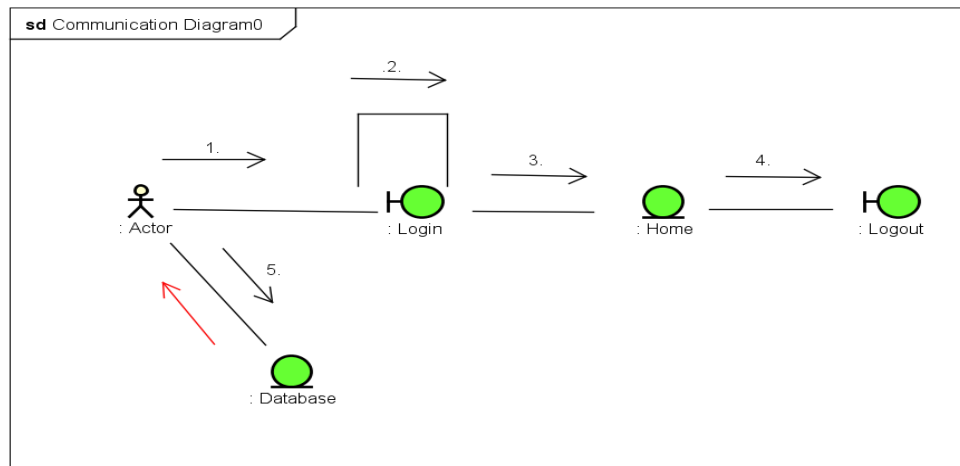
**Gambar 4.10 Sequence Diagram Input Random**

#### 4.2.4 Collaboration Diagram

*Collaboration diagram* menunjukkan *physical view* dari suatu sistem yang akan dibangun. *Collaboration diagram* menekankan pada urutan pesan antara objek sistem.

##### 1. Collaboration Diagram Home Admin

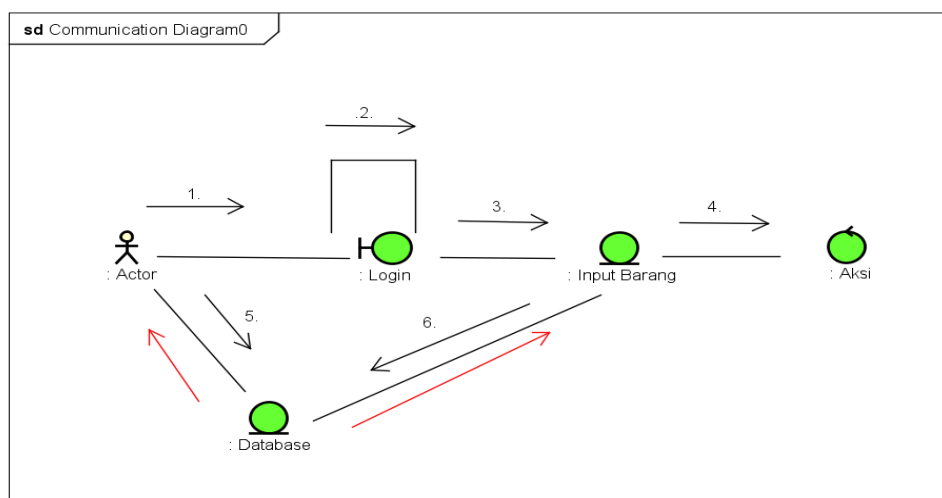
Collaboration diagram menampilkan home admin dan alur kerja serta hubungan antar objek dalam melakukan aktifitas admin. Seperti gambar 4.11 di bawah ini.



**Gambar 4.11 Collaboration Diagram Home Admin**

## 2. Collaboration diagram input barang

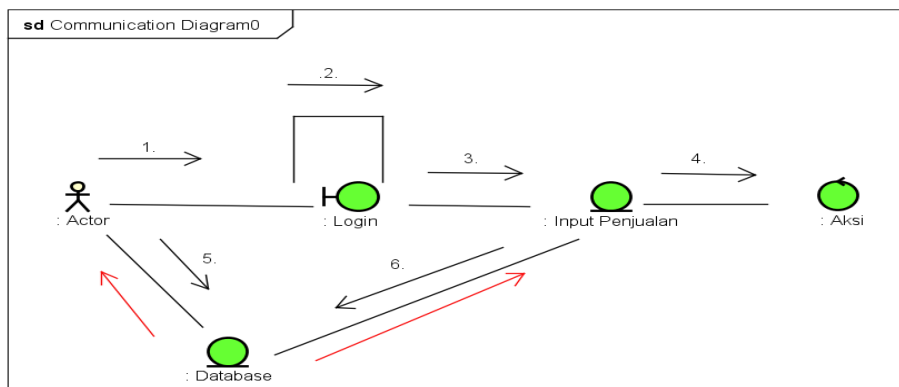
Collaboration diagram input barang mendefinisikan alur kerja dan hubungan antar objek dalam melakukan aktifitas input yang dilakukan oleh admin dan menyimpan hasil input tersebut. Seperti gambar 4.12 di bawah ini.



**cGambar 4.12 Collaboration Diagram Barang**

## 3. Collaboration Diagram Input Penjualan

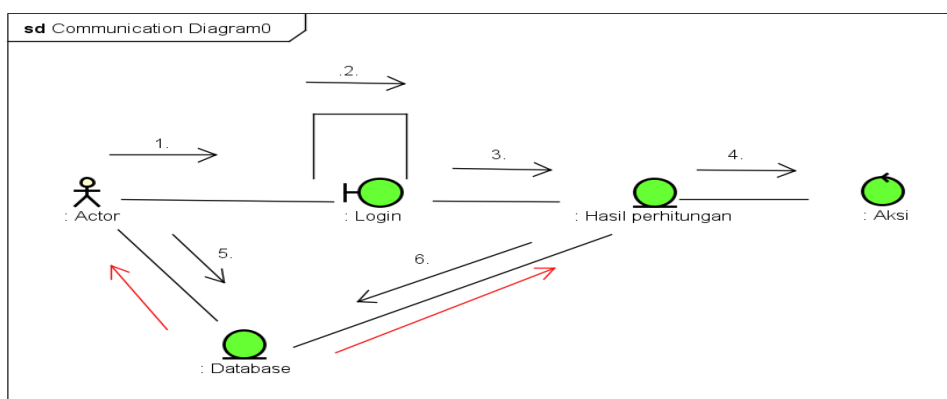
Collaboration diagram input penjualan mendeklarasikan alur kerja dan hubungan antar objek seperti tampak pada gambar 4.13 di bawah ini.



**Gambar 4.13 Collaboration Diagram Input Penjualan**

#### 4. Collaboration diagram hasil perhitungan

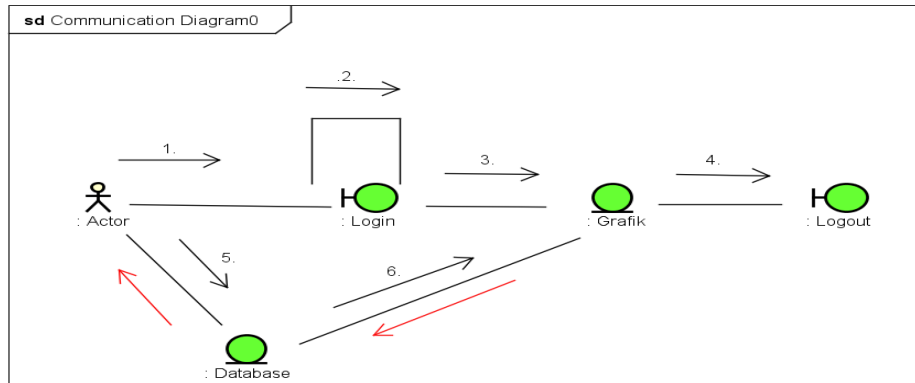
Collaboration diagram menampilkan hasil perhitungan mendeklarasikan alur kerja dan hubungan antar objek dalam melakukan aktifitas melihat hasil perhitungan. Seperti tampak pada gambar 4.14 di bawah ini.



**Gambar 4.14 Collaboration Diagram Hasil Perhitungan**

## 5. Collaboration Diagram Grafik

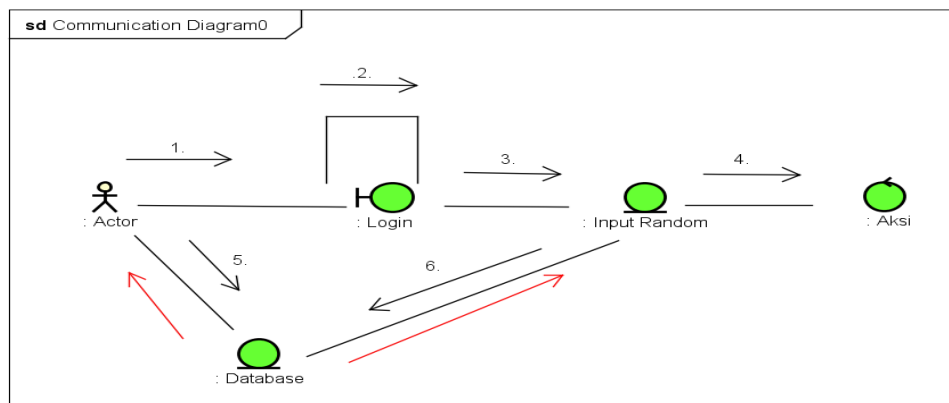
Collaboration diagram grafik menampilkan hasil dari prediksi yang berasal dari data yang telah diinputkan. Seperti pada gambar 4.15 di bawah ini.



**Gambar 4.15 Collaboration Diagram Grafik**

## 6. Collaboration diagram input random

Collaboration diagram input random menampilkan hasil dari random yang berasal dari data yang telah diinputkan. Seperti pada gambar 4.16 di bawah ini.



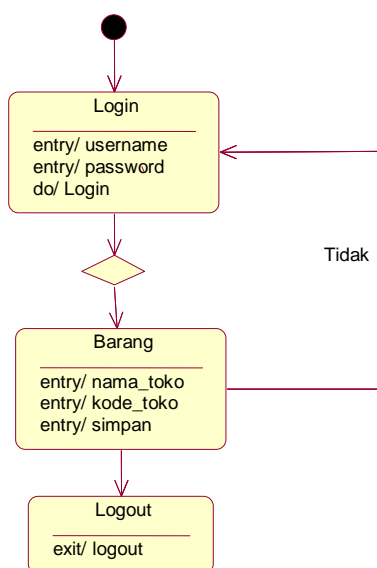
**Gambar 4.16 Collaboration Diagram Input Random**



#### 4.2.5 Statechart Diagram

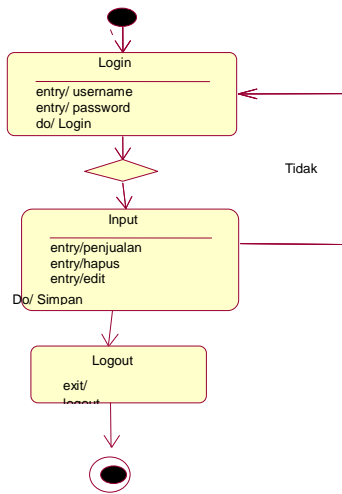
Statechart diagram menggambarkan transisi dan perubahan keadaan ( dari satu state ke state lainnya ) suatu objek pada sistem. Pada umumnya statechart menggambarkan class tertentu.

##### 1. Statechart Diagram Input Barang



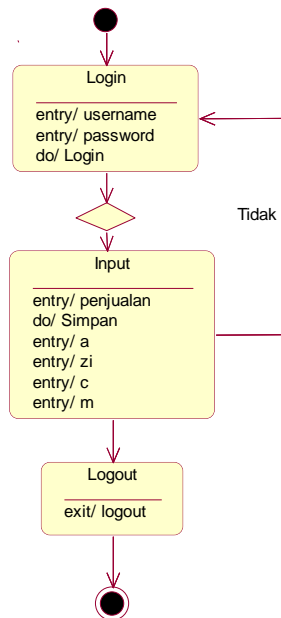
**Gambar 4.17 Statechart Input Barang**

##### 2. Statechart Diagram Input Penjualan



**Gambar 4.18 Statechart Diagram Penjualan**

### 3. Statechart Diagram Input Random



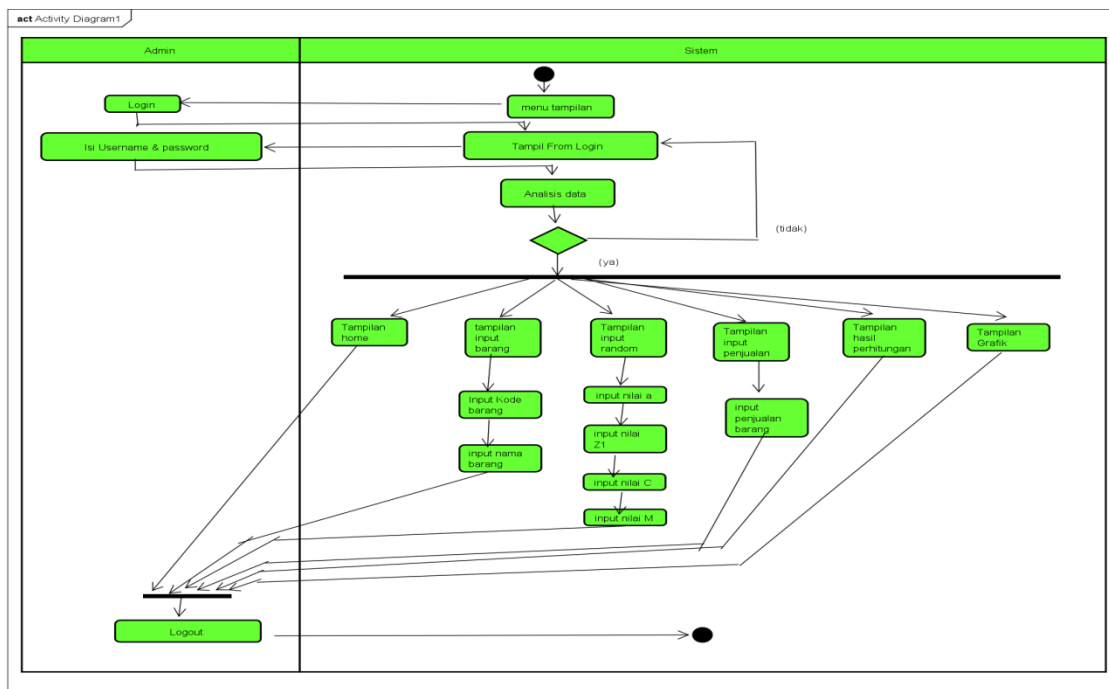
**Gambar 4.19 Statechart Diagram Input Random**

#### 4.2.6 Activity Diagram

*Activity* diagram digunakan untuk menampilkan tindakan dan sebagian besar transisi yang dipicu oleh penyelesaian tindakan yang berasal dari sumber.

1. *Activity diagram* pada Admin

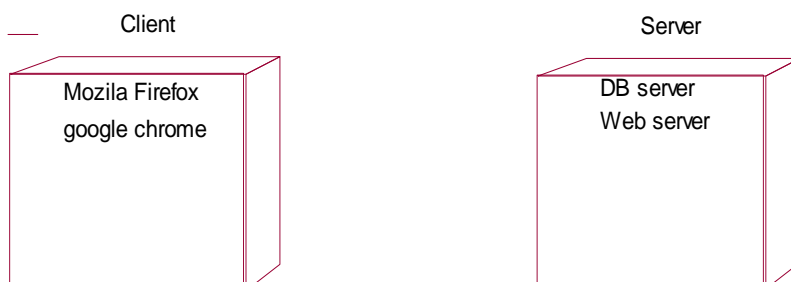
*Activity diagram* pada admin ini merupakan tindakan yang dilakukan oleh admin terhadap sistem.



**Gambar 4.20 Activity Diagram**

**4.2.7 Deployment Diagram**

*Deployment diagram* menggambarkan secara lengkap bagaimana komponen di *deploy* dalam infrastruktur system, dimana komponen akan terletak , bagaimana kemampuan jaringan pada kondisi tertentu, spesifikasi *server*, dan hal-hal lain yang bersifat fisik. Hubungan antar node (misalnya TCP/IP) *requirement* dapat juga didefinisikan dalam gambar 4.21 di bawah ini.



### Gambar 4.21 Deployment Diagram

#### 4.3 Perancangan Interface

Perancangan interface pada aplikasi simulasi penjualan berbasis web ini bertujuan agar dapat memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi serta menampilkan penjelasan sistem dan memberikan panduan pemakaian sistem secara menyeluruh sehingga pengguna (*user*) mengerti dan dapat menggunakan sistem tersebut dengan baik.

##### 4.3.1 Tampilan Menu Login

Pada tampilan ini admin harus login terlebih dahulu, untuk dapat mengakses system.

The image shows a login form with a double border. Inside, there are two text input fields, each labeled 'Varchar 10', stacked vertically. Below the second input field is a button labeled 'Login'.

**Gambar 4.22 Tampilan Menu Login**

##### 4.3.2 Tampilan Setelah Login

Setelah berhasil login, tampilan selanjutnya seperti gambar 4.23 di bawah ini :

The image shows a user interface after login. At the top, there is a header with the word 'Header' centered above it. Below the header is a horizontal navigation bar with the following links: Home, Input Barang, Input Random, Input Penjualan, Hasil Perhitungan, Grafik, and Logout. Below the navigation bar is a large rectangular box containing the text 'SELAMAT DATANG ADMIN' centered.

**Gambar 4.23 Tampilan Setelah Login**

### 4.3.3 Tampilan Input Barang

Pada tampilan input barang, admin dapat menginputkan kode barang dan nama barang penginputan data pada gambar 4.24 di bawah ini .

The image shows a rectangular form with a border. At the top center is a box labeled 'SUB MENU'. Below it are two input fields: the first is labeled 'Int 3' and the second is labeled 'Varchar 20'. At the bottom center is a button labeled 'SIMPAN'.

**Gambar 4.24 Tampilan Input Barang**

### 4.3.4 Tampilan Input Random

Pada tampilan ini admin dapat menginputkan nilai a, zi, c dan m untuk mencari random yang akan digunakan pada proses gambar 4.25 di bawah ini.

The image shows a rectangular form with a border. At the top center is a box labeled 'SUB MENU'. Below it are four input fields, each labeled 'Int 11'. At the bottom center is a button labeled 'SIMPAN'.

**Gambar 4.25 Tampilan Input Random**

### 4.3.5 Tampilan Input Penjualan

Pada tampilan ini admin dapat memilih barang dan bulan serta menginputkan jumlah penjualan per bulan pada gambar 4.26 di bawah ini.

The diagram shows a vertical stack of five rectangular boxes within a larger container. From top to bottom, the boxes are labeled: 'SUB MENU', 'Varchar 20', 'Varchar 11', 'Int 3', and 'SIMPAN'.

**Gambar 4.26 Input Penjualan**

### 4.3.6 Tampilan Hasil Perhitungan

Pada tampilan ini, sistem akan menampilkan hasil dari data yang telah diinputkan sebelumnya, hasilnya yaitu akan tampil banyaknya penjualan, distribusi kemungkinan, distribusi komulatif dan interval dari data tersebut pada gambar 4.27 di bawah ini.

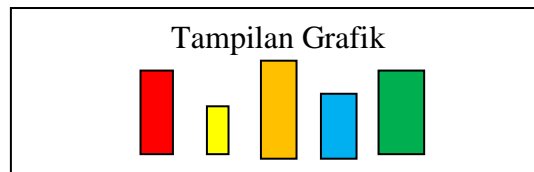
The diagram shows a vertical stack of two rectangular boxes within a larger container. The top box is labeled 'SUB MENU' and the bottom box is labeled 'Tabel Hasil Perhitungan'.

**Gambar 4.27 Tampilan Hasil Perhitungan**

### 4.3.7 Tampilan Grafik

Pada tampilan ini, sistem akan menampilkan grafik dari data yang telah diinputkan sebelumnya pada gambar 4.28 di bawah ini.

The diagram shows a single rectangular box labeled 'SUB MENU' within a larger container.



**Gambar 4.28 Tampilan Grafik**

## **BAB V**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **5.1 Implementasi**

Implementasi merupakan suatu tahapan penerapan sistem menjadi sebuah solusi atas permasalahan yang diteiti. Adapun implementasi sistem prmodelan dan simulasi metode *monte carlo* untuk mengetahui pendapat kotor dari setiap produk jenis produk furniture yang laku terjual sehingga menghasilkan prediksi untuk persiapan bulan berikutnya pada *home-industry* Istiqomah yang akan dibahas pada sub-sub bab berikut :

### 5.1.1 Implementasi Sistem

Implementasi sistem adalah proses penerapan rancangan program yang telah dibuat pada bab-bab sebelumnya. Hasilnya dari tahapan implementasi ini adalah suatu sistem untuk mengetahui pendapat kotor dari setiap jenis produk yang laku terjual pada *home-industry* Istiqomah. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan sistem untuk memprediksi penjualan barang pada *home-industry* Istiqomah yang sesuai dengan tujuan yang diharapkan atau belum. Implementasi sistem simulasi penjualan untuk menentukan pendapatan keripik maco badarai Istiqomah berbasis web dengan menggunakan metode *monte carlo*.

### 5.1.2 Instalasi XAMPP

XAMPP merupakan web server yang mudah digunakan dan dapat melayani tampilan halaman web yang dinamis. Dalam implementasi sistem ini, kita menggunakan aplikasi XAMPP karena aplikasi web server yang akan digunakan yaitu MySQL sudah *include* didalamnya. Berikut langkah-langkah instalasi XAMPP :

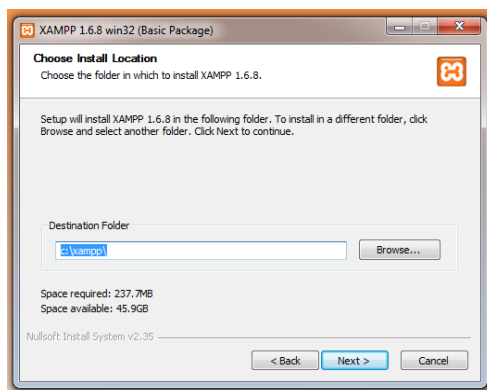
1. Tampilan *setup wizard* XAMPP pada gambar 5.1, klik Next .



**Gambar 5.1** Tampilan *Setup wizard*

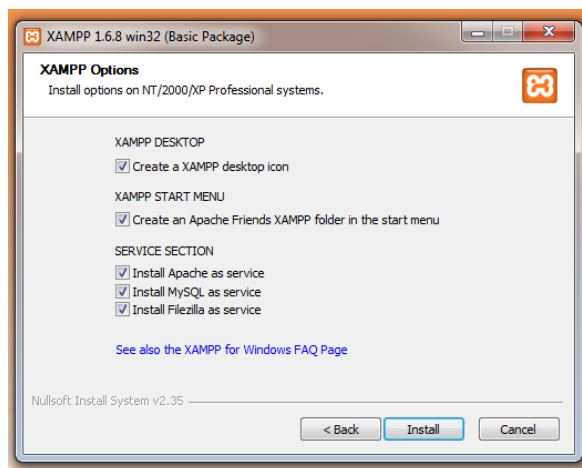


2. Selanjutnya akan muncul menu penyimpanan XAMPP, klik browse untuk memilih lokasi penyimpanan pada gambar 5.2, kemudian klik Next



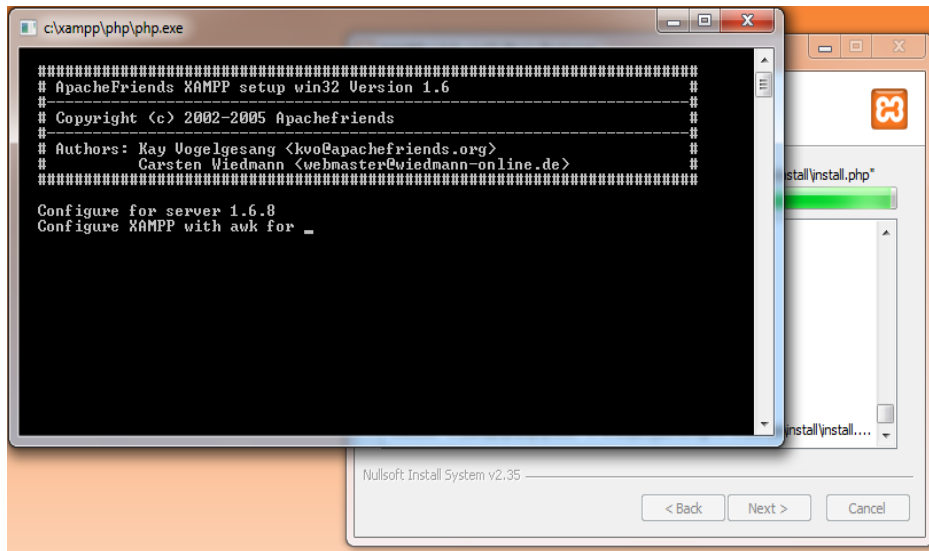
**Gambar 5.2 Tampilan Pilih Penyimpanan**

3. Selanjutnya akan muncul menu *service section* XAMPP, kita bisa memilih layanan yang akan otomatis dijalankan dengan cara mencentang dapat dilihat pada gambar 5.3, setelah itu klik Next.



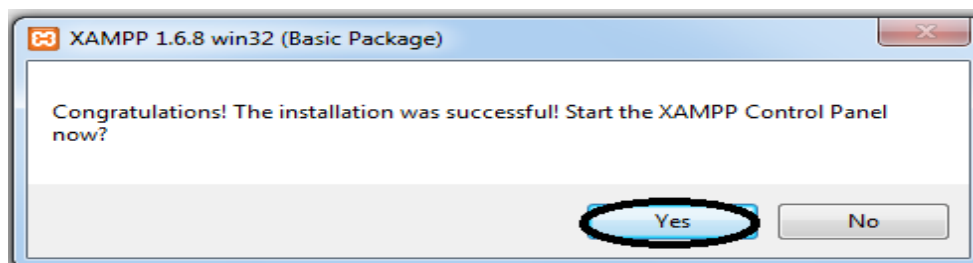
**Gambar 5.3 Tampilan Service Selection**

4. Selanjutnya akan muncul proses instalasi XAMPP, tunggu hingga proses selesai dan klik *finish* dapat dilihat pada gambar 5.4 di bawah ini.



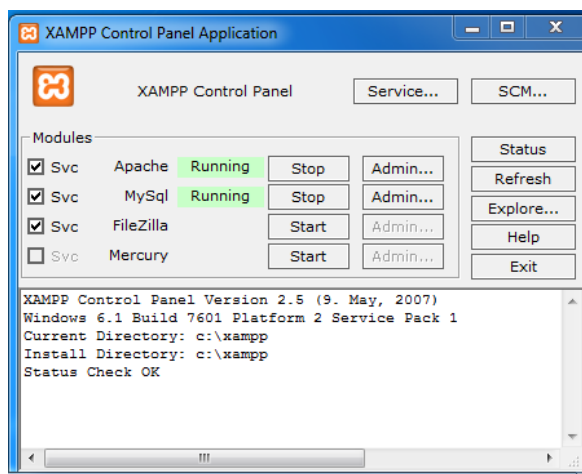
**Gambar 5.4 Tampilan Proses Instalasi**

5. Selanjutnya akan muncul tampilan seperti pada gambar 5.5 dibawah ini:



**Gambar 5.5 Tampilan Instalasi *Success***

6. Setelah selesai, maka tampilan akan terlihat seperti gambar 5.6 di bawah ini :

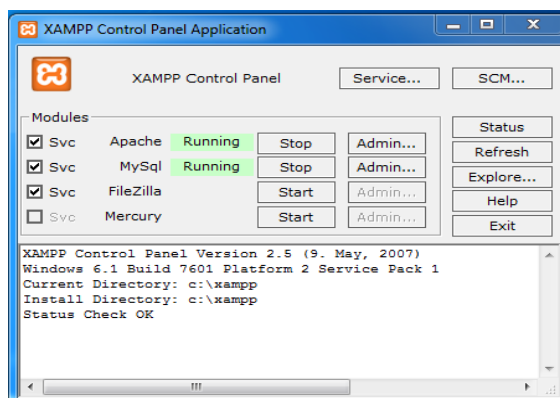


**Gambar 5.6 Tampilan Hasil Instalasi**

### 5.1.3 Implementasi Basis Data ( SQL) dengan Xampp

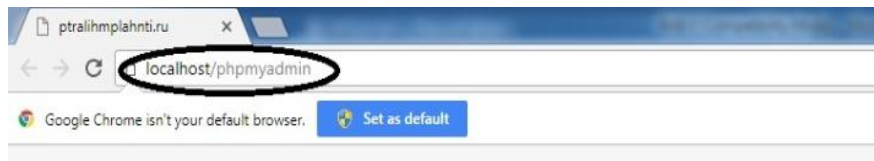
Xampp merupakan sebuah paket instalasi untuk PHP, *apache* dan MySQL yang biasa digunakan dan sebagai aplikasi pendukung dalam pembuatan sebuah *website*. Dan dibawah ini langkah langkah sebagai beriku:

1. Jalankan Xampp untuk mulai menggunakannya, dapat dilihat pada gambar 5.7 di bawah ini.



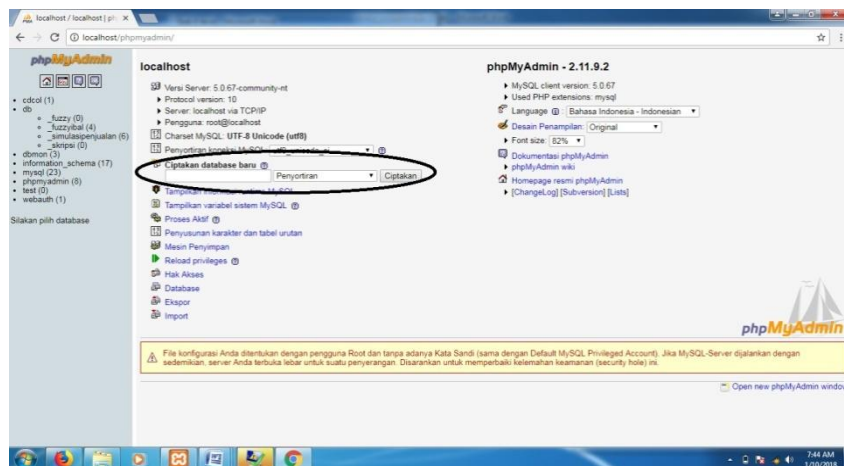
**Gambar 5.7 Tampilan XAMPP Panel**

2. Buka browser dengan mengetikkan : localhost/phpmyadmin, dapat dilihat pada gambar 5.8 di bawah ini.



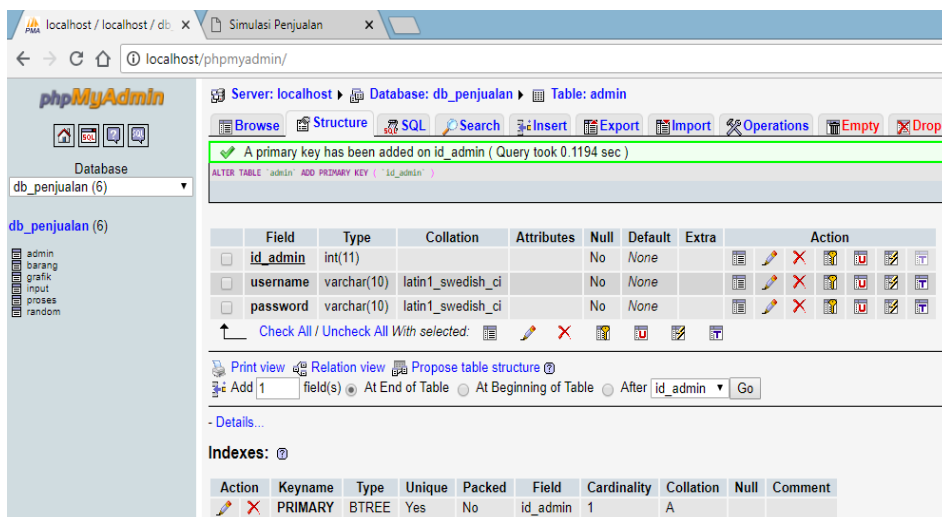
**Gambar 5.8 Pengetikan Pada Browser**

3. Maka selanjutnya tampil halaman phpmyadmin, dan silahkan buat database terlebih dahulu dapat dilihat pada gambar 5.9 di bawah ini.



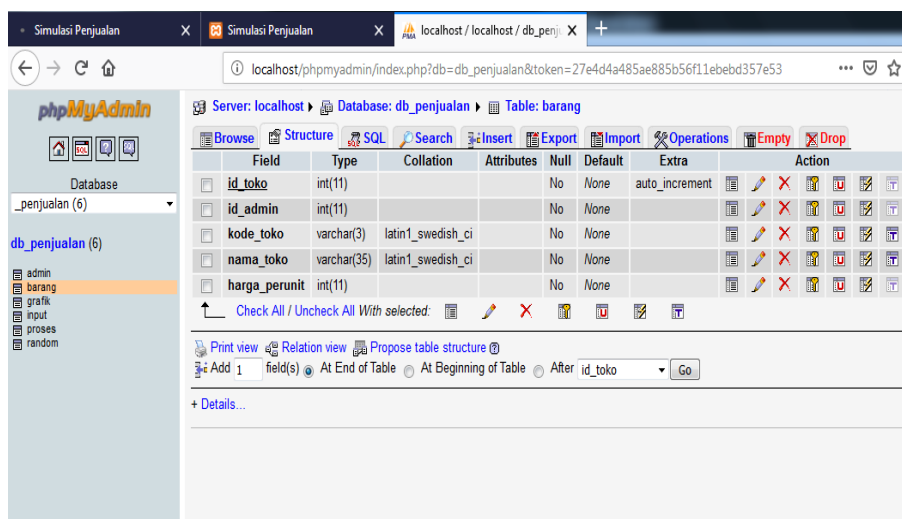
**Gambar 5.9 Pembuatan Database Baru**

4. Setelah membuat database, selanjutnya buat tabel admin yang akan digunakan untuk menyimpan identitas administrator untuk login pada sistem, masukkan field beserta tipe data yang digunakan, seperti gambar 5.10 di bawah ini.



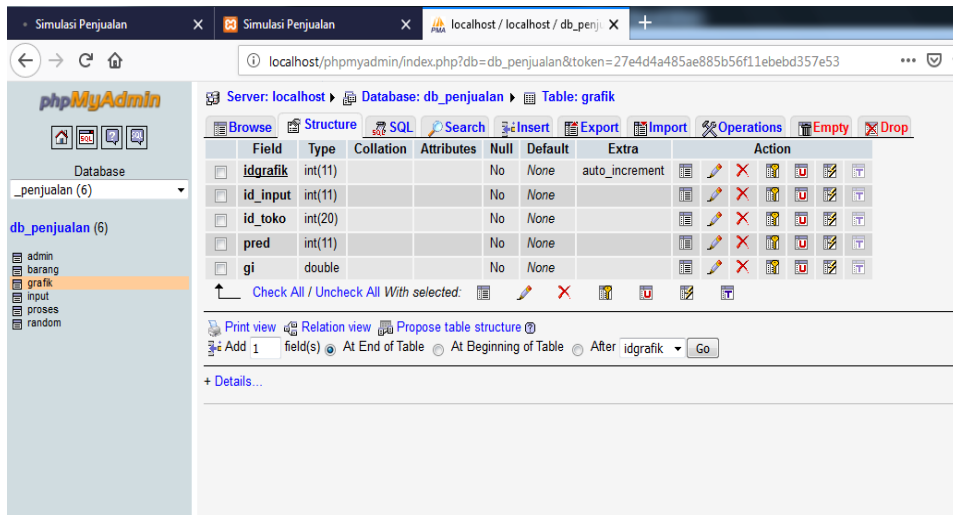
**Gambar 5.10 Tabel Admin**

5. Selanjutnya buat tabel barang, untuk menyimpan hasil dari barang yang telah diinputkan pada sistem.



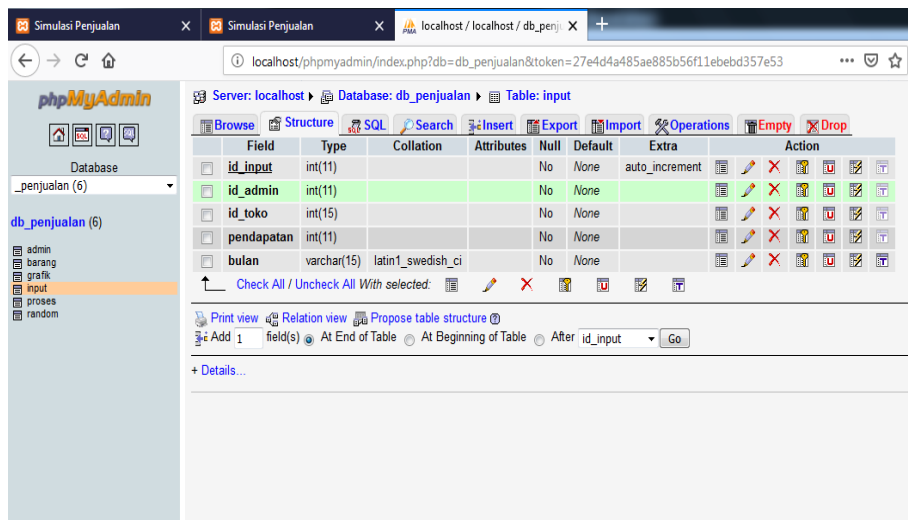
**Gambar 5.11 Tabel Barang**

6. Selanjutnya buat tabel grafik, untuk menampilkan hasil dari data yang telah diinputkan dapat dilihat pada gambar 5.12 di bawah ini.



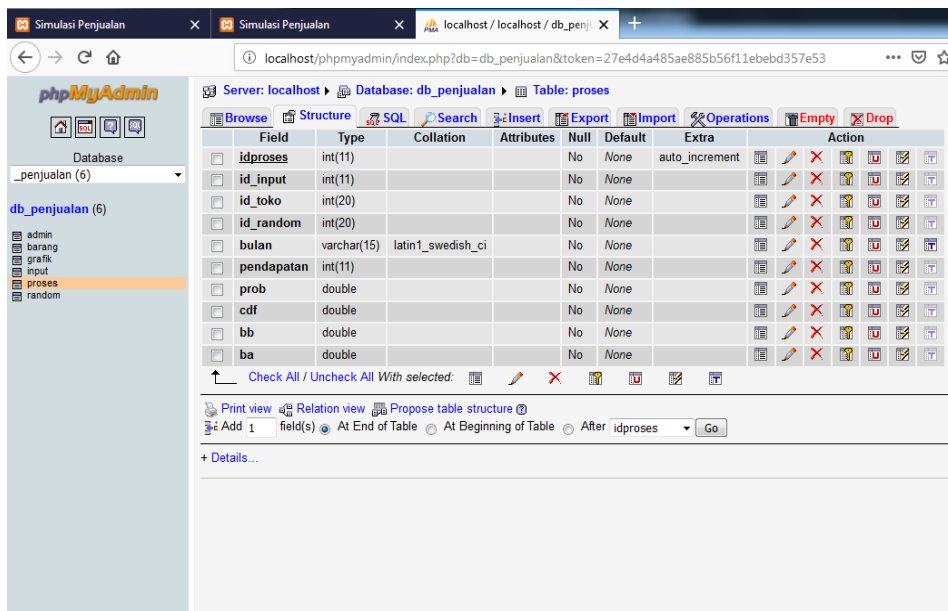
**Gambar 5.12 Tabel Grafik**

7. Selanjutnya buat tabel input, untuk menampilkan data penjualan yang telah diinputkan pada sistem seperti gambar 5.13 di bawah ini.



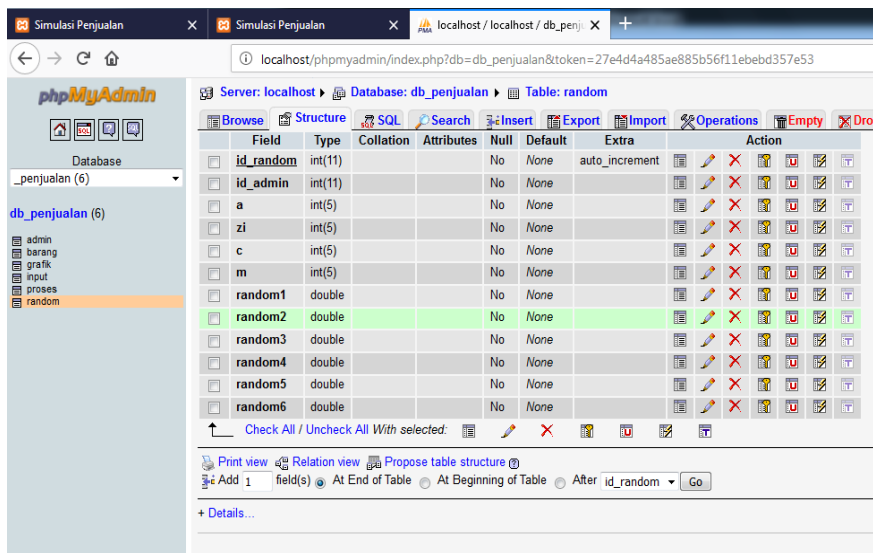
**Gambar 5.13 Tabel Input**

8. Selanjutnya buat tabel proses, untuk menampilkan hasil dari proses yang telah dijalankan pada sistem dapat dilihat pada gambar 5.14 di bawah ini.



Gambar 5.14 Tabel Proses

- Selanjutnya buat tabel random, untuk menampilkan hasil dari random yang telah diinputkan pada sistem dapat dilihat pada gambar 5.15 di bawah ini.



Gambar 5.15 Tabel Random

## 5.2 Pengujian

Pengujian merupakan tahapan verifikasi dan evaluasi hasil dari implementasi tersebut. Berikut dapat juga dilihat di lihat pada hostingan <https://simulasi.metodemontecarlo.000webhostapp.com/awal.html>. Adapaun pengujian yang dilakukan pada sistem pemodelan dan simulasi pada *home-industry* Istiqomah adalah :

### 5.2.1 Pengujian Sistem

Pengujian sistem pemodelan dan simulasi dengan menggunakan metode *monte carlo* menggambarkan kinerja dari aplikasi tersebut, apakah dapat memenuhi kebutuhan yang diinginkan oleh pemilik toko. Tahapan pengujian ini berisikan hasil dari eksekusi program yang telah dirancang. Pengujian sistem bertujuan menemukan kesalahan yang mungkin terjadi didalam sistem dan untuk mengetahui apakah sistem yang telah dibuat sesuai dengan tujuan yang telah ditetapkan sebelumnya.

**Tabel 5.1 Hasil Fungsionalitas Aplikasi Simulasi dengan Metode *Monte Carlo***

NO	Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login	Jika username dan password yang dimasukkan benar, maka akan tampil halaman beranda dan menu aplikasi simulasi, jika salah maka akan kembali lagi ke form login	Sesuai dengan yang diharapkan	Valid
2	Klik Menu "Input	Akan menampilkan input kode barang dan nama	Sesuai dengan	Valid



	Barang”	barang kemudian klik simpan untuk menyimpan data yang telah diinputkan. Selanjutnya data yang telah diinputkan akan muncul di tabel bawahnya.	yang diharapkan	
3	Klik Menu “Input Random”	Akan menampilkan input untuk menginputkan nilai yang akan digunakan untuk random, setelah nilai diinputkan, maka nilai random akan muncul di tabel bawahnya.	Sesuai dengan yang diharapkan	Valid
4	Klik Menu “Input Penjualan”	Akan menampilkan pilih barang yang telah diinputkan sebelumnya dan juga akan memilih bulan, dan menginputkan penjualan pada setiap barangnya, kemudian setelah disimpan, data akan muncul pada tabel dibawahnya.	Sesuai dengan yang diharapkan	Valid

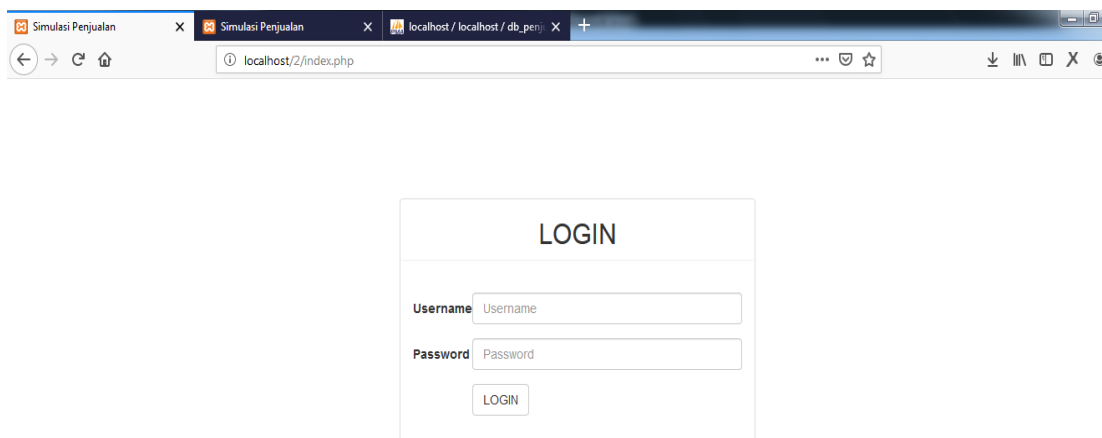
5	Klik Menu “Hasil Perhitungan”	Akan menampilkan pilih barang dan id random yang akan di proses, satu jenis barang hanya dapat menggunakan satu id random, namun satu id random dapat digunakan untuk banyak barang untuk diproses, selanjutnya hasil proses akan muncul di tabel kanan.	Sesuai dengan yang diharapkan	Valid
6	Klik Menu“Grafik”	Akan menampilkan grafik dari data yang telah diproses sebelumnya	Sesuai dengan yang diharapkan	Valid

### 5.2.2 Pengujian Interface

Pengujian dan implementasi sistem bertujuan untuk mengidentifikasi apakah sistem yang dirancang telah sesuai dengan apa yang diinginkan dan sesuai dengan kebutuhan perusahaan. Dengan dilakukannya implementasi dan pengujian, peneliti dapat mengidentifikasi kualitas dari sistem yang dibangun tersebut. Tampilan website yaitu merupakan sub bab yang menjelaskan tentang proses dimulainya sampai program ini selesai dieksekusi *point-point* pada sub bab ini akan menjelaskan tentang bagaimana sebuah halaman dijalankan dan apa saja fungsi yang terdapat dari halaman tersebut.

## 1. Halaman Utama Website

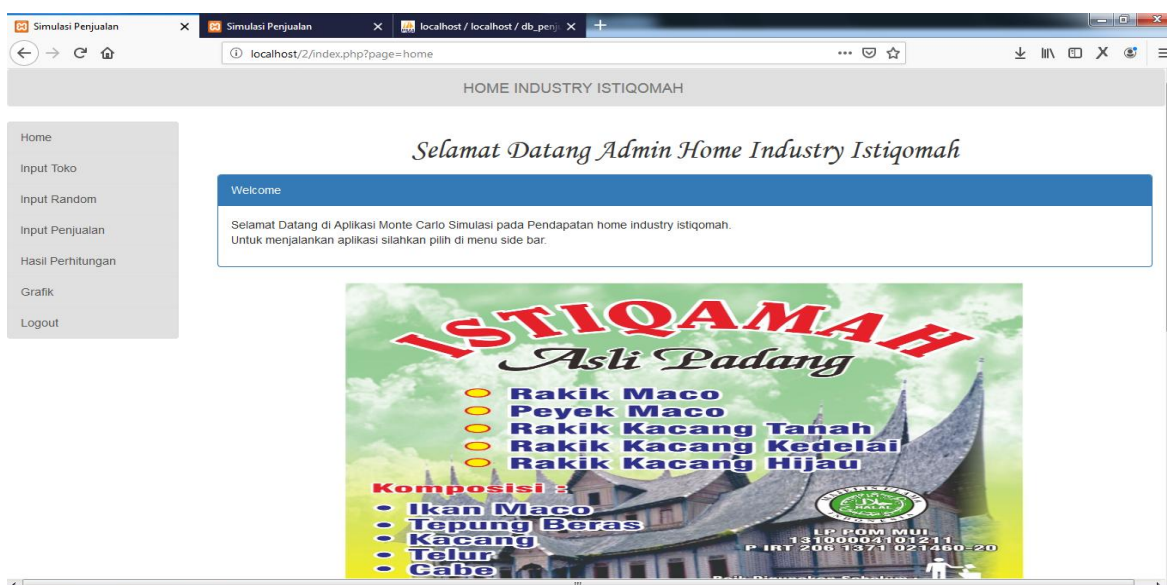
Merupakan tampilan awal website dijalankan, pada halaman utama ini terdapat form *login* admin tampak seperti gambar 5.16 berikut.



**Gambar 5.16 Halaman Login**

## 2. Halaman Home

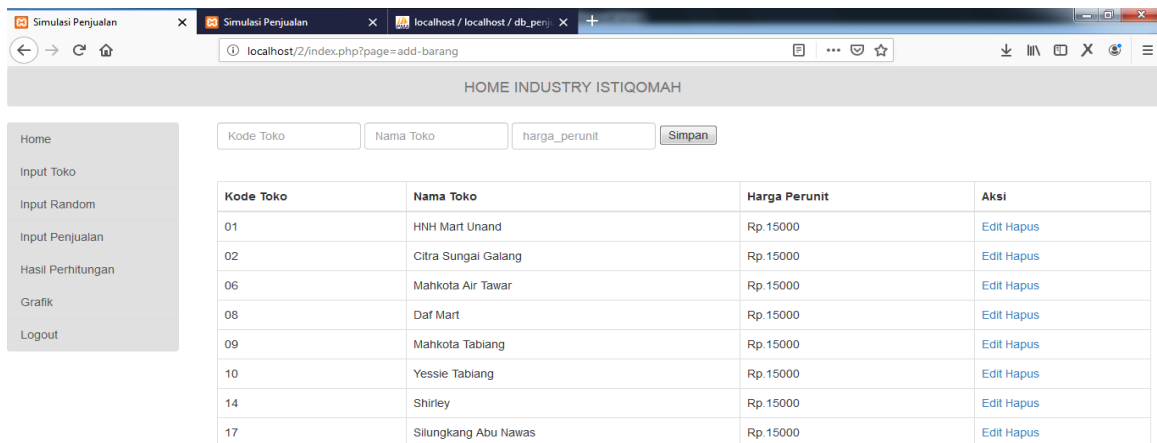
Setelah admin berhasil login, maka tampilan selanjutnya seperti gambar 5.17 di bawah ini.



**Gambar 5.17 Halaman Home**

### 3. Halaman Input Barang

Pada halaman ini, admin dapat menginputkan kode barang dan nama barangnya. Kemudian setelah disimpan, data yang telah diinputkan akan muncul pada table, dapat dilihat pada gambar 5.18 di bawah ini.

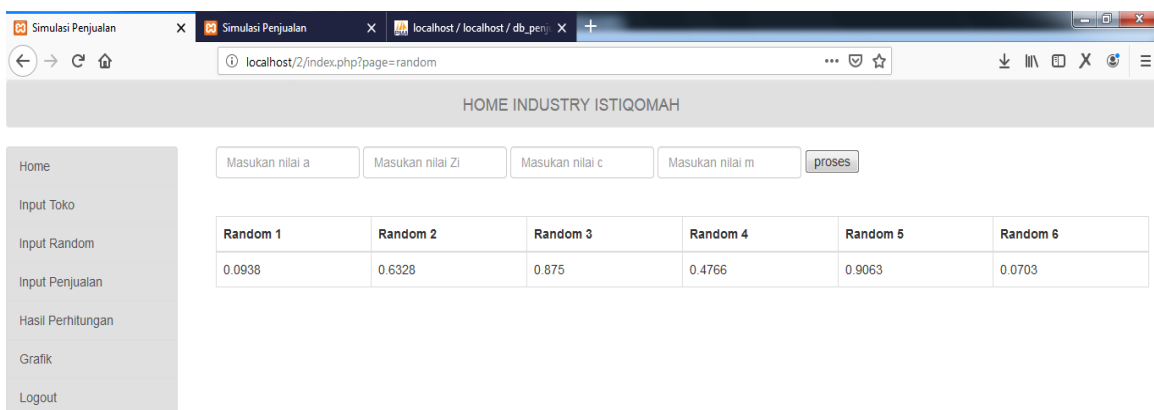


Kode Toko	Nama Toko	Harga Perunit	Aksi
01	HNH Mart Unand	Rp. 15000	<a href="#">Edit Hapus</a>
02	Citra Sungai Galang	Rp. 15000	<a href="#">Edit Hapus</a>
06	Mahkota Air Tawar	Rp. 15000	<a href="#">Edit Hapus</a>
08	Daf Mart	Rp. 15000	<a href="#">Edit Hapus</a>
09	Mahkota Tabiang	Rp. 15000	<a href="#">Edit Hapus</a>
10	Yessie Tabiang	Rp. 15000	<a href="#">Edit Hapus</a>
14	Shirley	Rp. 15000	<a href="#">Edit Hapus</a>
17	Silungkang Abu Nawas	Rp. 15000	<a href="#">Edit Hapus</a>

**Gambar 5.18 Halaman Input Barang**

### 4. Halaman Input Random

Pada halaman ini admin dapat menginputkan nilai  $a$ ,  $z_0$ ,  $c$  dan  $m$  untuk random yang akan digunakan untuk proses data nantinya. Angka yang telah diinputkan akan menghasilkan nilai random yang akan ditampilkan pada table, seperti gambar 5.19 dibawah ini



Random 1	Random 2	Random 3	Random 4	Random 5	Random 6
0.0938	0.6328	0.875	0.4766	0.9063	0.0703

### Gambar 5.19 Halaman Input Random

#### 5. Halaman Input Penjualan

Pada halaman ini admin dapat memilih barang yang telah diinputkan sebelumnya dan juga dapat memilih bulan yang akan diinputkan jumlah penjualannya. Setelah disimpan, hasilnya akan muncul pada tabel dibawahnya, seperti gambar 5.20 di bawah ini.

No	Bulan	Nama Toko	Jumlah Terjual	Aksi
1	Agustus	HNH Mart Unand	20	<a href="#">Edit Hapus</a>
2	September	HNH Mart Unand	22	<a href="#">Edit Hapus</a>
3	Oktober	HNH Mart Unand	20	<a href="#">Edit Hapus</a>
4	November	HNH Mart Unand	25	<a href="#">Edit Hapus</a>
5	Desember	HNH Mart Unand	20	<a href="#">Edit Hapus</a>
6	Januari	HNH Mart Unand	25	<a href="#">Edit Hapus</a>
7	Agustus	Citra Sungai Galang	20	<a href="#">Edit Hapus</a>
8	September	Citra Sungai Galang	20	<a href="#">Edit Hapus</a>
9	Oktober	Citra Sungai Galang	17	<a href="#">Edit Hapus</a>
10	November	Citra Sungai Galang	20	<a href="#">Edit Hapus</a>
11	Desember	Citra Sungai Galang	20	<a href="#">Edit Hapus</a>
12	Januari	Citra Sungai Galang	15	<a href="#">Edit Hapus</a>
13	Agustus	Mahkota Air Tawar	35	<a href="#">Edit Hapus</a>

### Gambar 5.20 Halaman Input Penjualan

#### 6. Halaman Hasil Perhitungan

Pada halaman ini admin dapat memilih barang yang akan di proses dan akan menggunakan id random yang mana, kemudian klik proses. Setelah itu hasil dari perhitungan akan muncul pada tabel di bawahnya. Dan admin dapat mencetak laporan jika diinginkan, dapat dilihat pada gambar 5.21 di bawah ini.

No	Kode Toko	Nama Toko	Bulan	Jumlah Terjual
1	01	HNH Mart Unand	Agustus	20
2	01	HNH Mart Unand	September	22
3	01	HNH Mart Unand	Oktober	20
4	01	HNH Mart Unand	November	25
5	01	HNH Mart Unand	Desember	20
6	01	HNH Mart Unand	Januari	25
Total				132

No	jumlah_terjual	Distribusi Kemungkinan	Distribusi Kumulatif	Batas Ri
1	20	0.1515	0.1515	0 - 0.1515
2	22	0.1667	0.3182	0.1516 - 0.3182
3	20	0.1515	0.4697	0.3183 - 0.4697
4	25	0.1894	0.6591	0.4698 - 0.6591

**Gambar 5.21 Halaman Hasil Perhitungan**

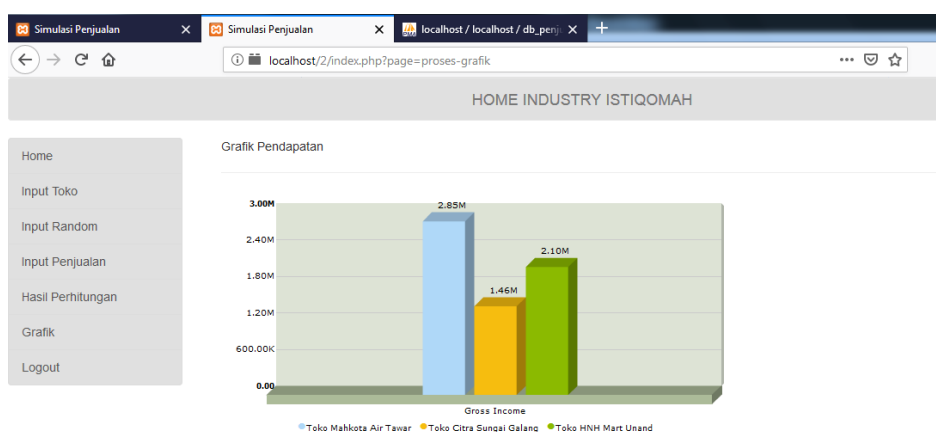
Dan selanjutnya Admin dapat melihat hasil dari proses simulasi, berikut hasil dari perhitungan simulasi yang menunjukkan pendapatan kotor dari jumlah barang yang terjual yang dapat dilihat pada laporan pendapatan pada gambar 5.22 di bawah ini.

Nama Toko	Bulan	Jumlah Terjual
HNH Mart Unand	Agustus	20
HNH Mart Unand	September	22
HNH Mart Unand	Oktober	20
HNH Mart Unand	November	25
HNH Mart Unand	Desember	20
HNH Mart Unand	Januari	25
Total		132
Gross Income		Rp.2100000

**Gambar 5.22 Laporan**

## 7. Halaman Grafik

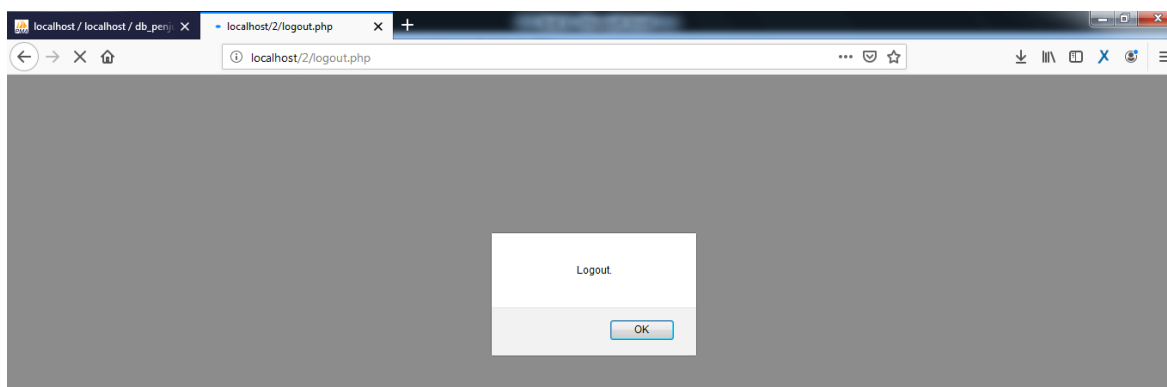
Pada tampilan ini admin dapat melihat grafik dari data yang telah diinputkan dan telah diproses sebelumnya, grafik ini hanya menunjukkan pendapatan kotor dari setiap produk sehingga pemilik toko atau admin dapat melihat produk mana yang lebih tinggi pendapatannya dapat dilihat pada gambar 5.23 di bawah ini.



**Gambar 5.23 Halaman Grafik**

## 8. Halaman Logout

Pada halaman ini admin logout dari sistem dan setelah admin logout maka admin akan tiba di halaman seperti halaman login dapat dilihat pada gambar 5.24 di bawah ini.



**Gambar 5.24 Halaman Logout**

## **BAB VI**

### **PENUTUP**

Berdasarkan hasil pengembangan sistem yang telah penulis lakukan, penulis mencoba membuat suatu kesimpulan dan mengajukan beberapa saran-saran yang berhubungan dengan pembahasan yang telah dikemukakan di bab-bab sebelumnya.

#### **6.1 Kesimpulan**

Berdasarkan dari proses pembangunan sistem yang telah dilakukan oleh peneliti, peneliti menarik kesimpulan bahwa :

1. Dengan menggunakan aplikasi yang dirancang pemilik toko *home-industry* Istiqomah dapat mengetahui keuntungan dengan melihat hasil laporan pendapatan atau barang yg terjual.
2. Dengan adanya sistem penjualan pemilik dapat dengan mudah membandingkan pendapatan pada setiap toko yang menjual produk keripik maco badarai istiqomah melalui hasil perhitungan dan laporan yang terdapat pada aplikasi ini.
3. Dengan menggunakan aplikasi yang dirancang pemilik *home-industry* Istiqomah dapat mengetahui tingkat minat konsumen dengan melihat hasil input data yang didalamnya terdapat jumlah barang yang terjual di setiap periodenya. Dari sana Pemilik *home-industry* dapat meningkat kualitas produk dari setiap jenis produk yang di buat.

#### **6.2 Keterbatasan Sistem**

Setelah melihat alur kerja sistem sistem simulasi penjualan yang dibangun berdasarkan output yang dihasilkan, aplikasi ini memiliki beberapa keterbatasan sistem sebagai berikut :



1. Pada sistem penjualan, bilangan random bersifat tetap dan tidak dapat berubah karena telah ditentukan oleh program.
2. Pada sistem penjualan tidak dapat menentukan price income. Hanya dapat menampilkan gross income dan prediksi kemungkinan jumlah barang yang akan laku di bulan depan.
3. Sistem yang dirancang untuk memprediksi penjualan barang hanya membahas mengenai proses simulasi dengan metode *monte carlo*.

### **6.3 Saran**

Dari penelitian yang telah dilakukan dan sebagai akhir dari penulisan ini, adapun kekurangan dari sistem ini yaitu masih terdapatnya beberapa fitur yang masih sangat sederhana. Adapun saran yang dapat diberikan dengan harapan akan berguna untuk meningkatkan kinerja sistem adalah sebagai berikut :

1. Dari kajian diatas penulis mengharapkan adanya penelitian lebih lanjut, karena ada beberapa fitur yang masih sangat sederhana dan belum sempurna yang belum dibahas karena keterbatasan waktu dan keterbatasan penulis dalam menyusun skripsi ini.
2. Diperlukannya penambahan informasi untuk laporan seperti menampilkan gross income dalam satu tahun serta price income.
3. Diperlukannya maintenance pada aplikasi penjualan agar dapat digunakan untuk jangka waktu yang panjang.

## DAFTAR PUSTAKA

- A.S, R., & M.Shalahudin. (2014). *Rekayasa Perangkat Lunak Pemetaan Wisata Sejarah Dan Budaya Berbasis Web ( Studi Kasus : Kabupaten Lahat ) Gambar 1 . Tahapan Metode Waterfall.* (12), 1–12.
- Andrianof, H. (2017). *PEMBANGUNAN SISTEM INFORMASI SMP NEGERI 2 PADANG BERBASIS WEB DENGAN DIBANTU DATABASE MYSQL.* 4(2), 266–291.
- Lambertus, S. A., Jairus, K. Y., & KallaYoseph. (2015). *SIMULASI PERKIRAAN KEUNTUNGAN PENJUALAN PULSA PADA URAN CELL MENGGUNAKAN METODE MONTE CARLO BERBASIS WEB.* 978–979.
- N. P. Ade. (2016). *Pengelolaan data pendidikan dan pelatihan untuk pegawai bank rakyat indonesia.* 3(2).
- Putra, T. A., & Hadi, A. F. (2016). *Perancangan aplikasi keuntungan produk rokok dengan menggunakan metode monte carlo di toko nabila.* VII(1), 164–188.
- Riyanto, O. A. W. (2017). *Simulasi Model Sistem Kerja Pada Departemen Injection Untuk Meminimasi Waktu Work-In-Process.* *Jurnal Ilmiah Teknik Industri*, 15(1), 69. <https://doi.org/10.23917/jiti.v15i1.1668>
- Rosa, A., & M.Shalahudin. (2014). *REKAYASA PERANGKAT LUNAK.*
- Saiful, S., & Ambarita, A. (2017). *PEMBUATAN APLIKASI WEB PENCARIAN JASA PEMBANTU RUMAH TANGGA (PRT) DIKOTA TERNATE WEB.* *IJIS-Indonesia Journal on Information System*, 2(September 2017), 69–76. <https://doi.org/10.1021/jp5128578>
- Buana, I Komang. S. 2014 . *“Jago Pemograman PHP”*. Yogyakarta: Penerbit Dunia Komputer.
- Ilmi, H.Z & Qoiriah, A. 2016. *“Perancangan Sistem Informasi Pencarian Asisten Rumah Tangga Berbasis Web”*. *Jurnal Manajemen Informatika Use the "Insert Citation" button to add citations to this document.*
- Nasution, Khairun Nizar. 2016 *"Prediksi Penjualan Barang Pada Koperasi PT. Perkebunan Silindak Dengan Menggunakan Metode Monte Carlo."* *Jurnal Riset Komputer.*
- Komang Dharmawan, 2017. *“Penentuan Harga Opsi Beli Tipe Asia Dengan Metode Monte Carlo – Control Variate”*. *Jurnal Matematika.*
- Halifia Hendri, 2015. *“Pemodelan Dan Simulasi Sistem Dengan Metode Kolmogorov-Smirnov Pada Antrian Pendaftaran Ulang Siswa Baru Yang Lulus Psb Online”*. *Jurnal Teknologi Komputer.*
- Betha, Sidik, Ir dkk 2014. *“Pemograman Web Dengan HTML”*. Bandung : InformatikaBandung.

- Prayitno, Agus dan Yulia Safitri, 2015. “*Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis*”. Jakarta Timur : Indonesian Journal on Software Engineering. Vol 1, No 1.
- Minarmi, & Susanti. 2014 .*Sistem Informasi Inventory Obat Pada Rumah Sakit Umum Daerah (Rsud) Padang. Momentum,16 (1)*.
- Sharma, P. 2015. *Discrete-event simulation. International Journal of Scientific & Technology Research*, Vol. 4.
- Sembiring, Alpiamus. 2015. *Perancangan Simulasi Penjualan Barang Dengan Metode Monte Carlo*. Pelita Informatika Budi Darma.
- Mahyoga, I Gusti, Putu, Ngurah, dkk. 2014. *Penentuan Harga Kontrak Opsi Tipe Eropa Menggunakan Metode Qurasi Monte Carlo Dengan Barisan Kuasi-Acak*. E-Jurnal Matematika Vol. 3.
- Ilmi, H.Z & Qoiriah, A. 2016. “*Perancangan Sistem Informasi Pencarian Asisten Rumah Tangga Berbasis Web*”. Jurnal Manajemen Informatika.
- Prasetio, Adhi. 2014. *Buku Sakti Web Master (PHP & MYSQL, HTML & CSS, HTML 5 & CSS3, JavaScript)*. Jakarta: Mediakita.
- Raharjo, Budi 2015. *Belajar otodidak MySQL : Teknik Pembuatan dan pengolahan database*. Bandung : Informatika.
- Iswandy, E, & Novinaldi. 2015. *Pemodelan dan simulasi dalam menentukan jumlah penjualan produk motor dengan metode monte carlo*. Jurnal Teknologi Informasi Dan Pendidikan, (Monte Carlo).