

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Sebagai bukti penelitian yang akan dibuat, maka penelitian akan dibandingkan terhadap penelitian sejenis yang pernah dilakukan. Adapun Penelitian sebelumnya yang penulis angkat yaitu :

Menurut Muhammad Ihksan (2021) Penelitian ini bertujuan memprediksi rata rata pendapatan penjualan, sehingga menjadi rekomendasi untuk digunakan dalam membuat strategi manajemen. Data yang diolah dalam penelitian ini adalah data penjualan dari tahun 2017 sampai 2019 yang bersumber dari Radja Minas. Data ini akan di olah dengan menggunakan Metode Monte Carlo. Hasil dari pengujian yang telah di lakukan memiliki tingkat akurasi sebesar 92,66%. Tingkat akurasi yang tinggi dari hasil pengolahan data prediksi, maka penelitian ini sangat tepat dan cocok digunakan untuk mengoptimalkan pendapatan penjualan. Sehingga penelitian ini menjadi rekomendasi untuk digunakan dalam membuat strategi manajemen di Radja Minas pada masa yang akan datang.

Menurut Bias Yulisa Geni (2021) Metode Simulasi Monte Carlo adalah metode komputasi dalam menghasilkan kemungkinan-kemungkinan dengan membangkitkan bilangan secara acak berdasarkan pengambilan sampel data. Prediksi pencarian keuntungan akan dilakukan setiap tahun. Selain untuk memprediksi keuntungan penjualan, data penjualan juga digunakan untuk memprediksi jumlah produk yang terjual setiap tahun. Berdasarkan hasil

perhitungan yang telah dilakukan, didapatkan tingkat akurasi dari seluruh penjualan produk wardah adalah 77,4 %.

Menurut Efani Desi (2021) Dalam proses pelayanan akademik, melayani mahasiswa sangatlah penting. Pelayanan harus dilakukan dengan sangat cepat. Agar proses pelayanan tersebut dapat dilakukan dengan cepat, dibutuhkan sebuah simulasi dalam proses antrian pelayanannya. Simulasi Monte Carlo merupakan simulasi antrian yang sangat cocok untuk diterapkan dalam proses pelayanan ini. Karena simulasi Monte Carlo ini perhitungannya menggunakan teori matematika untuk penyelesaian proses setiap antrian pelayanan mahasiswa. Sehingga proses pelayanan antrianya dapat dihitung lebih akurat dan minim dari kesalahan.

Menurut Radiyan Rahim (2020) Salah satu pendekatan yang dapat dilakukan untuk memperkirakan jumlah penjualan adalah dengan menggunakan cara simulasi. Penelitian ini menggunakan metode Monte Carlo dalam mengelola data dan menganalisa persediaan barang atau menentukan jumlah barang yang akan terjual pada periode berikutnya di toko Mardenis dengan sampling dari proses bilangan acak (Additive Random Number). Pengolahan data menggunakan empat sample data berdasarkan data history.

Berdasarkan penelitian dari Bias Yulisa Geni (2019) Prediksi pendapatan akan dilakukan setiap tahun. Selain untuk memprediksi pendapatan, data penjualan tersebut juga digunakan untuk memprediksi permintaan produk cat setiap tahun. Untuk melakukan prediksi terhadap penjualan produk cat dengan sangat baik. Berdasarkan hasil pengujian yang telah dilakukan didapatkan bahwa sistem yang digunakan untuk memprediksi pendapatan penjualan produk cat dengan rata-rata

akurasi sebesar 89%. Dengan tingkat akurasi yang cukup tinggi, penerapan metode monte carlo dianggap dapat melakukan prediksi pendapatan dan permintaan masing-masing produk cat setiap tahunnya. Sehingga, akan memudahkan pihak pimpinan untuk memilih strategi bisnis yang tepat untuk meningkatkan pendapatan penjualan produk cat.

Menurut Tri Kustanti Rahayu (2019) Penelitian ini bertujuan untuk melakukan simulasi untuk memprediksi keuntungan berdasarkan data penjualan pada Toko Kue Aneka Rasa menggunakan metode monte carlo. Beberapa tahapan yang dilakukan yaitu pertama menghitung frekuensi penjualan, kemudian menghitung distribusi probabilitas dan probabilitas kumulatif. Kemudian menghitung rentang nilai. Setelah itu melakukan simulasi menggunakan sejumlah variabel acak. Hasilnya adalah sejumlah kemungkinan keuntungan yang dapat diperoleh company.

II.2. Uraian Teoritis

II.2.1. Perancangan

Perancangan Sistem adalah salah satu tahap dari metodologi pengembangan sistem, dan merupakan salah satu bagian yang sangat penting di dalam itu. Tahap ini dilakukan oleh sistem designer yang melakukan interaksi dengan pengguna sistem, hasil desain dievaluasi oleh user dari sudut pandang kepentingan pemakai untuk kemudian diimplementasikan kembali oleh sistem designer. Berikut ini adalah beberapa prinsip yang perlu dipahami dalam melakukan desain sebuah sistem informasi :

1. Proses desain merupakan langkah lanjutan dari analisis data. Jadi desain harus dapat ditelusuri sampai ketinggian analisis.
2. Desain sebuah sistem harus meminimalkan kesenjangan intelektual.
3. Desain harus mengungkap keseragaman dan integrasi antar sub sistem yang kuat.
4. Desain harus berorientasi ke kondisi sekarang dan masa depan.
5. Desain harus mempertimbangkan konsep penanganan kesalahan disaat pengoperasian sistem.
6. Desain harus dinilai kualitasnya pada saat desain dibuat.
7. Desain harus dikaji lebih lanjut sehingga dapat meminimalkan kesalahan-kesalahan konseptual. (Saipul Anwar : 2018)

II.2.2. Aplikasi

Pengertian aplikasi adalah suatu bagian dari perangkat lunak yang dapat digunakan untuk menyelesaikan masalah-masalah yang khusus yang dihadapi *user* dengan menggunakan kemampuan komputer. (Fergiawan Listianto, dkk : 2017)

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Ari Setiaji, dkk : 2020).

II.2.3. Peramalan

Peramalan adalah proses menaksirkan/ memperkirakan sesuatu di masa yang akan datang yang berdasarkan pada data yang ada di masa lalu yang kemudian dianalisis secara ilmiah dengan memakai metode statistika dengan tujuan supaya memperbaiki peristiwa yang akan terjadi di waktu yang akan datang. Peramalan (*forecasting*) merupakan alat bantu yang penting dalam perencanaan yang efektif dan efisien khususnya dalam bidang ekonomi. Dalam organisasi modern mengetahui keadaan yang akan datang tidak saja penting untuk melihat yang baik atau buruk tetapi juga bertujuan untuk melakukan persiapan peramalan. Tujuan diadakannya peramalan atau *forecasting* adalah untuk meminimalisasi resiko serta faktor ketidakpastian. Dengan adanya hasil peramalan, diharapkan tindakan atau keputusan dari suatu perusahaan atau organisasi dapat memberi dampak lebih baik pada jangka yang akan datang (Yanuar Adi Kurniawan:2018).

Ramalan (*forecast*) merupakan dugaan atau perkiraan mengenai terjadinya suatu kejadian atau peristiwa di waktu yang akan datang. Ramalan ini sangat berguna dalam berbagai bidang kehidupan, terutama dalam rangka perencanaan untuk mengantisipasi berbagai keadaan yang terjadi pada masa yang akan datang. Ramalan bisa dilakukan secara kualitatif maupun kuantitatif. Terkait dengan ramalan kuantitatif, metode peramalannya pada dasarnya dapat dibedakan atas:

1. Metode peramalan melalui analisis suatu variabel yang akan diperkirakan dengan variabel waktu, yang dikenal dengan metode hubungan deret waktu. Data yang digunakan adalah data deret waktu (*time series*).

2. Metode peramalan melalui analisis pola hubungan antara variabel yang akan diperkirakan dengan variabel-variabel lain yang mempengaruhinya (waktu dan/serta bukan waktu). Metode ini sering disebut metode hubungan sebab akibat (*causal method*). Data yang digunakan dapat berupa data time series maupun data *cross section*.

Prediksi (*Forecasting*) adalah suatu teknik analisa perhitungan yang dilakukan dengan pendekatan kualitatif maupun kuantitatif untuk memperkirakan kejadian dimasa depan dengan menggunakan referensi data-data di masa lalu untuk meminimumkan pengaruh ketidakpastian. Metode ini diterapkan dalam bagian proses perencanaan produksi maka pihak perusahaan akan lebih terbantu dalam penjadwalan produksi, karena metode ini dapat memberikan output terbaik sehingga diharapkan resiko kesalahan yang disebabkan oleh kesalahan perencanaan dapat ditekan seminimal mungkin (Ria Anggraini Walangadi : 87 : 2019).

II.2.4 PHP

PHP adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah web dan bias digunakan pada HTML. PHP merupakan singkatan dari “PHP : *Hypertext Preprocessor*”, dan merupakan bahasa yang disertakan dalam dokumen HTML, sekaligus bekerja di sisi server (*server-side HTML-embedded scripting*). Artinya sintaks dan perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada

halaman HTML biasa, sehingga script-nya tak tampak disisi client. PHP dirancang untuk dapat bekerja sama dengan database *server* dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses database menjadi begitu mudah. Tujuan dari bahasa *scripting* ini adalah untuk membuat aplikasi di mana aplikasi tersebut yang dibangun oleh PHP pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan di *server* (Muhammad Susilo, dkk : 2018)

II.2.5 MySQL

MySQL merupakan suatu jenis database server yang sangat terkenal. MySQL termasuk jenis RDBMS (*Relational Database Manajement System*). MySQL mendukung bahasa pemrograman PH, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (*Relational Database Management System*) server. RDBMS adalah program yang memungkinkan pengguna database untuk membuat, mengelola, dan menggunakan data pada suatu model relational. Dengan demikian, tabel-tabel yang ada pada database memiliki relasi antara satu tabel dengan tabel lainnya. Beberapa keunggulan dari MySQL yaitu :

- a. Cepat, handal dan mudah dalam penggunaannya. MySQL lebih epat tiga sampai empat kali dari pada database server komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.

- b. Didukung oleh berbagai bahasa *Database Server MySQL* dapat memberikan pesan *Error* dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
- c. Mampu membuat tabel berukuran sangat besar. Ukuran maksimal dari setiap tabel yang dapat dibuat dengan *MySQL* adalah 4 GB sampai dengan ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.
- d. Lebih murah *MySQL* bersifat open source dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan Windows Platform. Melekatnya integrasi PHP dengan *MySQL*. Keterikatan antara PHP dengan *MySQL* yang sama-sama Software Open-Source sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan database server lainnya. Modul *MySQL* di PHP telah dibuat Built-in sehingga tidak memerlukan konfigurasi tambahan pada File konfigurasi Php. (Reza Hermiati : 2021).

II.2.7. 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 setiap waktu dari variabel acak dan secara berurutan mengikuti perubahan-perubahan yang terjadi

dalam proses simulasi. Penggunaan metode *Monte Carlo* memerlukan sejumlah besar bilangan acak, dan hal tersebut semakin mudah dengan perkembangan pembangkit bilangan acak, yang jauh lebih cepat dan praktis dibanding dengan metode sebelumnya. Simulasi monte carlo adalah suatu metode untuk mengevaluasi secara berulang suatu model *deterministik* menggunakan himpunan bilangan acak sebagai masukan. Metode ini sering digunakan bila model adalah kompleks, *non linier*, atau melibatkan banyak parameter tertentu yang saling berhubungan. (Radian Rahim : 2020)

Beberapa jenis model dapat dikelompokkan sebagai berikut (Pemodelan dan Simulasi Sistem Teori Aplikasi :

- a. Model skala: model yang dibuat biasanya memperkecil 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 dapat berupa diagram, seperti: diagram blok DFD (*Data Flow Diagram*), CD (*Context Diagram*), Petri Net, *Flowchart*, PERT (*Program Evaluation dan Review*

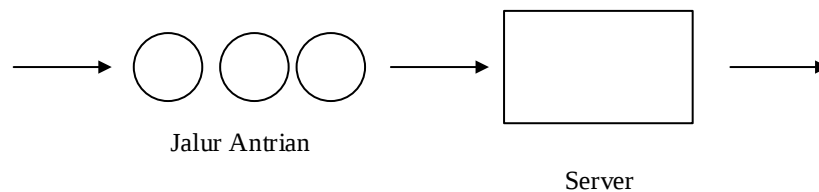
Technique). Model dalam bentuk skema statik, seperti: tabel, bagan Gantt (*Gantt chart*).

- e. Model simbolik (matematika): model dalam bentuk persamaan matematika seperti: persamaan *differential*, persamaan deferens, persamaan aljabar, persamaan logika, dan lain-lain.
- f. Model komputer: model dalam bentuk program komputer (*source code*) yang ditulis menggunakan bahasa komputer tertentu. (Radiyan Rahim : 2020)

Ada 3 struktur dasar model antrian yang umum terjadi dalam sebuah antrian:

1. *Single Channel Single Phase*

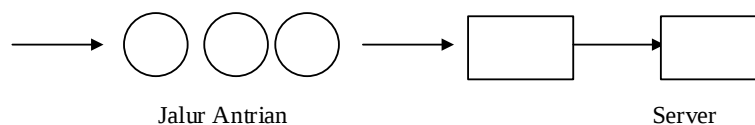
Menunjukkan hanya terdapat satu jalur masuk sistem pelayanan dan hanya terdapat satu



Gambar II.1. *Single Channel Single Phase*

2. *Single Channel Multi Phase*

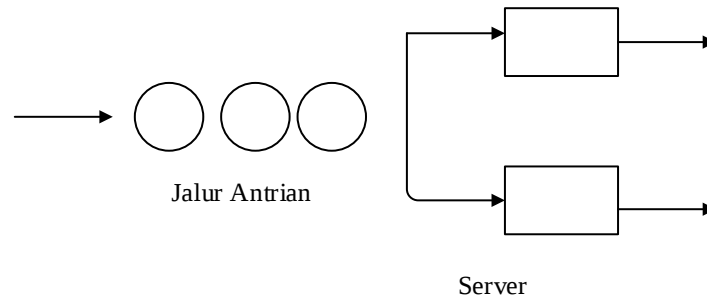
Menunjukkan hanya terdapat satu jalur masuk sistem pelayanan dan terdapat dua atau lebih fasilitas pelayanan secara seri dalam jalur tersebut:



Gambar II.2. *Single Channel Multi Phase*

3. *Multi Channel Single Phase*

Menunjukkan terdapat dua atau lebih jalur masuk sistem pelayanan dan hanya terdapat satu fasilitas pelayanan dalam setiap jalurnya.



Gambar II.3. Multi Channel Multi Phase

II.2.5. Metode Monte Carlo

Metode *Monte Carlo* adalah kelas algoritma komputasi yang menggunakan sampel acak dalam menghasilkan pemecahan masalah. Metode *Monte Carlo* menggunakan angka acak dan statistik *probabilitas* dalam menyelesaikan masalah.

Berikut ini adalah langkah dan rumus :

1. Mengelompokkan Data.

Melakukan pengelompokan data yang akan digunakan.

2. Menetapkan Interval Angka Random (Angka Acak).

Angka acak dibangkitkan dengan *Mixed Congruent Method* dengan menetapkan nilai Z_i , a c dan m .

3. Membangkitkan Angka Random (Angka Acak).

Untuk simulasi ini bilangan acak dibangkitkan terdapat 12 bilangan acak.

Dimana $Z_{i+1} = (a \cdot Z_i + C) \bmod M \dots \dots \dots (3)$

Keterangan :

Z_i = Bilangan awal (bilangan bulat ≥ 0 , $Z_0 < m$);

A = Konstanta Pengali ($a < m$);

C = Konstanta Pergeseran ($c < m$);

Mod = Konstanta Modulus ($m > 0$).

4. Simulasi *Monte Carlo* Produksi Bata Merah.

Pengujian untuk menentukan hasil dari *monte carlo* dengan membandingkan data real dengan data sebelumnya dengan cara menyesuaikan angka acak yang telah didapat sebelumnya.

5. Hasil Simulasi. Hasil dari simulasi dari *monte carlo* dapat dilihat tingkat akurasi antara data real dan data yang telah diprediksi dengan menentukan persentasi dari perbandingannya. (Radiyan Rahim : 2020)

II.2.5. *Random Number*

Bilangan acak atau bilangan random adalah Suatu bilangan yang tidak dapat diprediksi kemunculannya disebut dengan bilangan acak/random. Untuk membangkitkan bilangan random pada komputer maka diperlukan sebuah IDE compiler. IDE ini berfungsi untuk menerjemahkan program yang telah dibuat ke dalam bahasa mesin, sehingga komputer dapat menjalankan program tersebut. (Radiyan Rahim : 2020)

II.2.8 Normalisasi

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data, teknik pengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi

yang baik tanpa redundansi. Tujuan normalisasi adalah mengorganisasikan data kedalam tabel-tabel untuk memenuhi kebutuhan pemakai, menghilangkan kerangkapan data, mengurangi kompleksitas, mempermudah modifikasi data. (Mukhlisulfatih Latief : 2018)

1. Proses Normalisasi

- a. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu keberapa tingkat.
- b. Apabila tabel yang diuji belum memenuhi persyaratan tertentu maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

2. Tahapan Normalisasi :

- 1) Bentuk tidak normal : Menghilangkan perulangan grup.

Tabel II.1. Contoh bentuk tidak normal (Unnormal)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
			M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
			Akn201	Akuntansi	D310	Lia	B
			MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2018

- 2) Bentuk Normal pertama (1NF) : Menghilangkan ketergantungan sebagian.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kesatu bila setiap data bersifat atomik yaitu setiap irisan baris dan kolom hanya mempunyai satu nilai data.

Tabel II.2. Contoh Bentuk Normal Pertama (1NF)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
2683	Welli	MI	M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
5432	Bakti	AK	Akn201	Akuntansi	D310	Lia	B
5432	Bakti	AK	MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2018

3) Bentuk Normal kedua (2NF) : Menghilangkan ketergantungan transitif.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kedua bila relasi tersebut sudah memenuhi bentuk normal kesatu dan atribut yang bukan *key* sudah tergantung penuh terhadap *key*-nya.

Tabel II.3. Contoh Bentuk Normal Kedua (2NF)

Kode-MK	Nama-MK	Kode Dosen	Nama Dosen
M1350	Manajemen DB	B104	Ati
M1465	Analisis Perc. Sistem	B317	Dita
M1350	Manajemen DV	B104	Ati
Akn201	Akuntansi	D310	Lia
MKT300	Dasar Pemasaran	B212	Lola

Sumber : Mukhlisulfatih Latief : 2018

4) Bentuk Normal ketiga (3NF) : Menghilangkan anomali-anomali hasil dari ketergantungan fungsional. Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal ketiga bila relasi tersebut sudah memenuhi bentuk normal kedua dan atribut yang bukan *key* tidak tergantung transitif terhadap *key*-nya. (Mukhlisulfatih Latief : 2018)

Tabel II.4. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2018

II.2.9. Basis Data (*Database*)

Basis data merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lain. Basis data atau *database* merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi berfungsi sebagai basis penyedia informasi bagi pemakainya, Sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan dengan yang lainnya dan untuk membuatnya tersedia beberapa aplikasi yang bermacam-macam dalam suatu sistem organisasi. Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan. (Priyo Sutopo,dkk 2018).

II.2.14. MySQL

MySQL (My Structure Query Language) merupakan sebuah program pembuat database yang bersifat *open source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *windows* maupun *linux*. *MySQL* juga merupakan sebuah perangkat lunak sistem manajemen basis data SQL yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multi user*. *MySQL* juga sering dikenal dengan nama sistem manajemen *database* relasional. Suatu *database relasional* menyimpan data dalam *table* yang terpisah. *Tabel* tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan *user* memperoleh kombinasi data dari beberapa *table* dalam suatu permintaan. Untuk administrasi *database*, seperti pembuatan *database*, pembuatan *tabel*, dan sebagainya dapat digunakan aplikasi berbasis *web* seperti *phpMyAdmin* dengan aplikasi XAMPP. (Saipul Anwar : 2018)

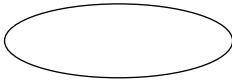
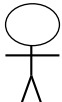
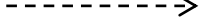
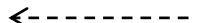
II.2.15 UML (Unified Modelling Language)

Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language (UML)*. UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Ade Handini : 2018)

1. Diagram Use Case (Use Case Diagram)

Use case diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu :

Tabel II.5. Simbol *Use Case* Diagram

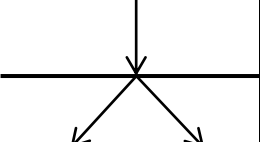
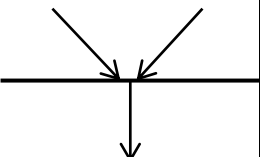
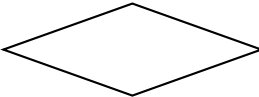
Gambar	Keterangan
	Use case menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama use case.
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan use case, tetapi tidak memiliki control terhadap use case.
	Asosiasi antara aktor dan use case, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan use case yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam use case lain (<i>required</i>) atau pemanggilan use case oleh use case lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari use case lain jika kondisi atau syarat terpenuhi.

(Sumber : Ade Handini : 2018)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

Tabel II.6. Simbol Diagram Aktivitas

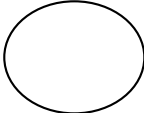
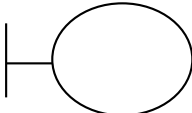
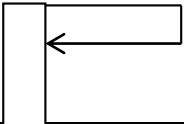
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

(Sumber : Ade Handini : 2018)

3. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu :

Tabel II.7. Simbol Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Handini : 2018)

4. Class Diagram (Diagram Kelas)

Class Diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan

dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

Tabel II.8. Simbol Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Handini : 2018)