

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang dilakukan oleh Akmal Nasution (2018) dengan judul “Forecasting Produksi Karet Menggunakan Metode Weighted Moving Average” Proses perhitungan peramalan secara manual menggunakan metode Weighted Moving Average tentunya harus menguasai keahlian khusus terutama dibidang matematis. Untuk itu peneliti bermaksud membangun sebuah rancangan aplikasi peramalan sehingga dapat digunakan oleh semua orang dengan mudah dan cepat. Teknik peramalan ini dapat diterapkan diberbagai bidang pekerjaan, termasuk dalam peramalan produksi karet. Dengan memperoleh data peramalan produksi karet dimasa mendatang tentunya dapat memberikan gambaran untuk langkah-langkah kerja kedepannya, sehingga dapat meningkatkan produktivitas kerja perusahaan.

Berdasarkan penelitian Dewi Putu (2018) dengan judul “Sistem Informasi Peramalan Persediaan Barang Menggunakan Metode *Weighted Moving Average*” Terdapat kendala yang dialami oleh perusahaan antara lain permintaan dengan persediaan tidak seimbang dan pencatatan persediaan barang juga masih manual dengan tulisan tangan. Penentuan persediaan barang itu sendiri masih dilakukan secara manual dengan tulisan tangan jadi untuk menentukan berapa jumlah

persediaan barang yang akan disediakan perusahaan harus terlebih dahulu membandingkan jumlah barang yang keluar dengan perbandingan data sebelumnya juga data pengeluaran barang yang baru terjadi. Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk merancang dan membangun Sistem Informasi yang dapat membantu dalam pengolahan data barang dan peramalan persediaan barang untuk bulan berikutnya.

Berdasarkan penelitian yang dilakukan oleh Alfian Nurlifa (2017) dengan judul “Sistem Peramalan Jumlah Penjualan Menggunakan Metode *Moving Average* Pada Rumah Jilbab Zaky” Bentuk laporan yang masih manual membuat pemilik menjadi kesulitan dalam melihat perkembangan usaha, akibatnya pemilik dapat melakukan kesalahan dalam pengambilan keputusan terutama untuk penambahan atau pengurangan stok barang. Tujuan dari penelitian ini adalah untuk memanfaatkan informasi yang ada semaksimal mungkin dengan menggunakan sistem informasi manajemen. Dalam sistem informasi manajemen ini akan mengelola data untuk dijadikan informasi yang lebih mudah dilihat seperti menggunakan grafik. Sistem informasi manajemen ini disertai peramalan untuk membantu pengguna terutama manajer dalam pengambilan keputusan.

Berdasarkan penelitian yang dilakukan oleh Eko Sulistiono (2016) dengan judul “Sistem Prediksi Penjualan Dengan Single Moving Everage Pada Distro Mega Busana I” Transaksi penjualan mencapai seratus kali dalam sehari dengan sekali transaksi dapat berisi lima macam perlengkapan fashion bahkan lebih, sehingga sering mengalami kekurangan stok barang. Hal ini dikarenakan Distro Mega Busana II tidak memiliki sebuah sistem yang dapat memperkirakan jumlah

persediaan yang harus dilakukan. Sistem Informasi Prediksi Penjualan menggunakan Metode Single Moving Average digunakan untuk menyelesaikan permasalahan tersebut. Sistem informasi ini menghasilkan prediksi yang dapat memprediksi jumlah penjualan pada periode yang akan datang dan jumlah ini dapat digunakan sebagai acuan dalam pemesanan barang pada supplier. Implementasi Sistem informasi prediksi penjualan dapat digunakan untuk melakukan prediksi penjualan per bulan dan dapat digunakan untuk melakukan pemesanan barang

Berdasarkan penelitian yang dilakukan oleh Junaidi (2014) dengan judul “Analisis Hubungan Deret Waktu untuk Peramalan” 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 memang tidak akan pernah tepat 100%, karena masa depan mengandung masalah ketidakpastian. Namun demikian, dengan pemilihan metode yang tepat, kita membuat peramalan dengan tingkat kesalahan yang kecil atau memberikan perkiraan yang sebaik mungkin terhadap keadaan masa yang akan datang. Ramalan bisa dilakukan secara kualitatif maupun kuantitatif.

Sedangkan berdasarkan penelitian yang dilakukan oleh penulis Susilawati, 2019, dengan judul “Sistem Informasi Peramalan Produksi Mie Sedap Menggunakan Metode *Moving Average*” Sistem informasi peramalan ini dapat memprediksi jumlah penjualan produk mie sedap dengan tujuan untuk mengambil

keputusan dan menerapkan prioritas berapa banyak jumlah produk yang akan diproduksi pada bulan atau tahun berikutnya.

II.2. Uraian Teoritis

II.2.1. Sistem Informasi

Sistem informasi adalah suatu kumpulan dari komponen – komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi. Sistem informasi dapat juga didefinisikan sebagai suatu sistem yang menerima sumber data sebagai *input* dan mengolahnya menjadi produk informasi sebagai *output*. Sistem informasi merupakan suatu sistem yang terdiri dari beberapa subsistem atau komponen *hardware*, *software* dan *brainware*, data dan prosedur untuk menjalankan *input*, proses, *output*, penyimpanan dan pengontrolan yang mengubah sumber data menjadi informasi. Atau dapat juga didefinisikan sebagai suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan – laporan yang diperlukan (Marimin ; 2012 : 18).

II.2.2. Peramalan

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*. (Junaidi : 2014).

II.2.3. Metode *Simple Moving Average*

Metode *Simple Moving Average* atau rata-rata bergerak adalah salah satu metode peramalan bisnis yang sederhana dan sering digunakan untuk memperkirakan kondisi pada masa yang akan datang dengan menggunakan data-data masa lalu (data-data historis). Metode peramalan yang menghitung rata-rata suatu nilai runtut waktu dan kemudian digunakan untuk memperkirakan nilai pada periode selanjutnya. *Moving Average* atau rata-rata bergerak diperoleh melalui penjumlahan dan pencarian nilai rata-rata dari sejumlah periode tertentu, kemudian menghilangkan nilai terlamanya dan menambah nilai baru. Metode *Moving Average* ini lebih baik digunakan untuk menghitung data yang bersifat

stabil atau data yang tidak berfluktuasi dengan tajam (data yang perubahan naik dan turunnya sangat drastis). Hal ini dikarenakan data pada setiap periode diberikan bobot yang sama sehingga tidak dapat mewakili periode-periode tertentu yang bersifat khusus ataupun data periode terakhir yang biasanya dinilai sebagai data yang terbaik dalam menggambarkan kondisi terkini.

Keterangan rumus pada Simple Moving Average :

$$\text{MA} = \Sigma X / \text{Jumlah Periode} \dots\dots\dots(1)$$

Keterangan :

MA = Moving Average

ΣX = Keseluruhan penjumlahan dari semua data periode waktu yang diperhitungkan.

Jumlah Periode = Jumlah periode rata-rata bergerak.

Atau dapat ditulis dengan :

$$\text{MA} = (n_1 + n_2 + n_3 + \dots) / n \dots\dots\dots(2)$$

Keterangan :

MA = Moving Average

n_1 = data periode pertama

n_2 = data periode kedua

n_3 = data periode ketiga dan seterusnya

n = Jumlah periode rata-rata bergerak

Metode rata-rata bergerak atau yang biasa disebut moving average adalah metode peramalan dimana dalam menghitung peramalan di masa depan didapat

dari penjumlahan data-data lampau dan kemudian dibagi dengan jumlah data yang ada. Jumlah data-data yang harus diambil untuk perhitungan rata-rata bergerak (moving average) berjumlah bebas, namun tentunya harus disesuaikan dengan ketersediaan data-data lampau yang ada.

Data yang digunakan pun harus berurut dari tahun terbaru sampai tahun yang paling lama, karena data terbaru yang dianggap jauh lebih akurat dibanding data-data yang lama.

Biasanya dalam perhitungan rata-rata bergerak (moving average), data yang diambil adalah data 3 periode sebelumnya. Karena data 3 tahun yang lalu, dianggap paling relevan dengan keadaan yang akan datang. Namun, tetap saja, jumlah data yang diambil adalah bebas sesuai dengan selera masing masing.

Rumus peramalan dengan metode rata-rata bergerak (moving average) adalah :

$$\text{Moving Average (Ft)} = \frac{\sum \text{jumlah data dan periode sebelumnya}}{n} \\ = \frac{\sum A_{t-1} + A_{t-2} + A_{t-3} + \dots + A_{t-n}}{n} \quad \dots\dots\dots(3)$$

Keterangan :

Ft = Peramalan untuk periode yang akan datang

n = Jumlah periode peramalan moving average

At~1 = Data aktual satu periode sebelum peramalan

At~2 = Data aktual dua periode sebelum peramalan

At~n = Data aktual satu n sebelum peramalan

Jumlah ke-n harus disesuaikan dengan persoalan yang diminta. Jika menggunakan moving average 2 tahunan, maka otomatis jumlah n dan data aktual akan berjumlah 3 pula.

Rata-rata bergerak tunggal (*Moving average*) untuk periode t adalah nilai rata-rata untuk n jumlah data terbaru. Dengan munculnya data baru, maka nilai rata-rata yang baru dapat dihitung dengan menghilangkan data yang terlama dan menambahkan data yang terbaru. *Moving average* ini digunakan untuk memprediksi nilai pada periode berikutnya. Model ini sangat cocok digunakan pada data yang stasioner atau data yang konstant terhadap variansi, tetapi tidak dapat bekerja dengan data yang mengandung unsur trend atau musiman. Rata-rata bergerak pada orde 1 akan menggunakan data terakhir (F_t), dan menggunakannya untuk memprediksi data pada periode selanjutnya. Metode ini sering digunakan pada data kuartalan atau bulanan untuk membantu mengamati komponen-komponen suatu runtun waktu. Semakin besar orde rata-rata bergerak, semakin besar pula pengaruh pemulusan (smoothing).

Studi kasus :

Perusahaan melakukan penjualan beras 10 kg dalam kurun waktu 4 bulan yaitu Januari 2018 – April 2018 sebagai berikut ;

Tabel II.1. Penjualan Beras Januari – April 2018

Bulan	Penjualan Beras	Prediksi Penjualan
Januari	120	-
Februari	150	-
Maret	200	-
April	221	
Mei	?	
Juni	?	

Perhitungan peramalan penjualan beras 10 kg dengan menggunakan moving average dan tingkat periode rata-rata bergerak adalah 4 bulan :

Moving Average (MA) Bulan Mei 2018 :

$$\text{Moving Average (MA) Mei} = 120+150+200+221/4$$

$$\text{Moving Average (MA) Mei} = 691/4$$

$$\text{Moving Average (MA) Mei} = 172.75$$

$$\text{Moving Average (MA) Mei} = 173$$

Dengan demikian peramalan penjualan beras 10kg pada bulan Mei adalah 173 Karung. Perhitungan nilai peramalan pada bulan Juni dengan nilai actual Mei 200 karung dan dapat dilihat pada Tabel II.2. berikut ini :

Tabel II.2. Penjualan Beras Januari – Mei 2018

Bulan	Penjualan Beras	Prediksi Penjualan
Januari	120	-
Februari	150	-
Maret	200	-
April	221	-
Mei	200	173
Juni	?	

Moving Average (MA) Bulan Juni 2018 :

$$\text{Moving Average (MA) Juni} = 150+200+221+200/4$$

$$\text{Moving Average (MA) Juni} = 771/4$$

$$\text{Moving Average (MA) Juni} = 192.75$$

$$\text{Moving Average (MA) Juni} = 193$$

Dengan demikian peramalan penjualan beras 10kg pada bulan Juni 2018 adalah 193 Karung. Maka nilai prediksi penjualan beras 10 kg adalah sebagai berikut :

Tabel II.3. Penjualan Beras Januari – Mei 2018

Bulan	Penjualan Beras	Prediksi Penjualan
Januari	120	-
Februari	150	-
Maret	200	-
April	221	-
Mei	200	173
Juni	?	193

II.2.4. Basis Data (*Database*)

Secara sederhana *database* (basis data atau pangkalan data) dapat diungkapkan sebagai suatu pengorganisasian data dengan bantuan komputer yang memungkinkan data dapat diakses dengan mudah dan cepat. Pengertian akses dapat mencakup pemerolehan data maupun pemanipulasian data seperti menambah serta menghapus data. Dengan memanfaatkan komputer, data dapat disimpan dalam media pengingat yang disebut *hard disk*. Dengan menggunakan media ini, keperluan kertas untuk menyimpan data dapat dikurangi. Selain itu, data menjadi lebih cepat untuk diakses terutama jika dikemas dalam bentuk *database*. (Mujilan : 2012:23)

Pengaplikasian *database* dapat kita lihat dan rasakan dalam keseharian kita. *Database* ini menjadi penting untuk mengelola data dari berbagai kegiatan. Misalnya, kita bisa menggunakan mesin ATM (anjungan tunai mandiri/ *automatic teller machine*) bank karena bank telah mempunyai *database* tentang nasabah dan rekening nasabah. (Mujilan : 2012:23)

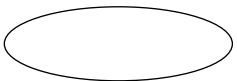
II.2.5. UML (*Unified Modeling Language*)

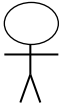

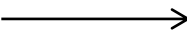
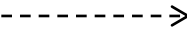
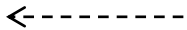
Menurut Gata (2013 : 4) Hasil pemodelan pada OOAD (*Object Oriented Analysis And Design*) terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi 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. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*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.4. Simbol *Use Case*

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .




	<p>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 <i>use case</i>, tetapi tidak memiliki control terhadap <i>use case</i>.</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i>, merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i>, merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.</p>

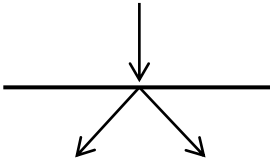
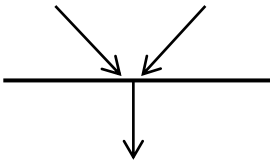
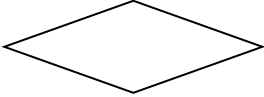

(Sumber : Gata, 2013 : 4)

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.5. Simbol *Activity Diagram*

Gambar	Keterangan
	<p><i>Start point</i>, diletakkan pada pojok kiri atas dan merupakan awal aktifitas.</p>
	<p><i>End point</i>, akhir aktifitas.</p>
	<p><i>Activites</i>, menggambarkan suatu proses/kegiatan bisnis.</p>

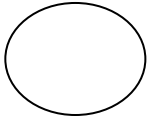
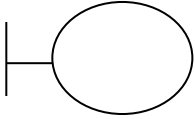
	<p><i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.</p>
	<p><i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.</p>
	<p><i>Decision Points</i>, menggambarkan pilihan untuk pengambilan keputusan, <i>true</i>, <i>false</i>.</p>
	<p><i>Swimlane</i>, pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.</p>

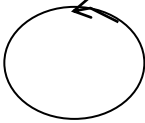
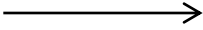
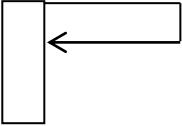


(Sumber : Gata, 2013 : 6)

3. Diagram Urutan (*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.6. Simbol *Sequence Diagram*

Gambar	Keterangan
	<p><i>EntityClass</i>, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.</p>
	<p><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.</p>

	<p><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.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Gata, 2013 : 7)

4. *Class Diagram* (Diagram Kelas)

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.7. Multiplicity 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 : Gata, 2013 : 9)

II.2.6. PHP (*Hypertext Preprocessor*)

PHP sendiri sebenarnya merupakan singkatan dari “*Hypertext Preprocessor*”, yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancang web yang dinamis dan dapat bekerja secara otomatis. PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web *browser* menjadi kode HTML. (Achmad Solihin, 2013)

II.2.7. MySQL

MySQL merupakan RDBMS (atau *server database*) yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak *user*”. (Rahardjo Adisasmita : 2011 : 21), MySQL memiliki beberapa keistimewaan, antara lain :

1. **Portabilitas.** MySQL dapat berjalan stabil pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga*, dan masih banyak lagi.
2. **Perangkat Lunak Sumber Terbuka.** MySQL didistribusikan sebagai perangkat lunak sumber terbuka, dibawah lisensi GPL sehingga dapat digunakan secara gratis.
3. **Multi-user.** MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. **Performance tuning**, MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. **Ragam tipe data.** MySQL memiliki ragam tipe data yang sangat kaya, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
6. **Perintah dan Fungsi.** MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
7. **Keamanan.** MySQL memiliki beberapa lapisan keamanan seperti *level subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. **Skalabilitas dan Pembatasan.** MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya. (Ratna Dwi Utami ; 2016)

II.2.8. Normalisasi

Normalisasi merupakan parameter digunakan untuk menghindari duplikasi terhadap tabel dalam basis data dan juga merupakan proses sebuah tabel yang masih memiliki beberapa anomali atau ketidak wajaran sehingga menghasilkan tabel yang lebih sederhana dan struktur yang bagus, yaitu sebuah tabel yang tidak memiliki data *redundancy* dan memungkinkan *user* untuk melakukan *insert*, *delete*, dan *update* pada baris (*recod*) tanpa menyebabkan inkonsistensi data. Tujuannya untuk menghindari beberapa anomali

1. *Insertion Anomaly* adalah proses melakukan penambahan *recod* baru akan tetapi mempengaruhi *user* untuk terjadinya duplikasi data.
2. *Deletion Anomaly* adalah proses melakukan penghapusan *record* akan tetapi akan menyebabkan hilangnya data yang akan dibutuhkan pada *record* lain.
3. *Modification Anomaly* adalah proses merubah data pada sebuah *record* mempengaruhi perubahan pada *record* lain karena adanya duplikasi.

Adapun bentuk – bentuk normalisasi adalah sebagai berikut :

1. **Bentuk tidak normal**

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja tidak lengkap dan terduplikasi. Data dikumpulkan apa adanya sesuai keadaanya.

Tabel II.9 Tabel Tidak Normal

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 : 2016

2. Bentuk normal tahap pertama (1st Normal Form)

Definisi :

Sebuah table disebut 1NF jika :

- a. Tidak ada baris yang duplikat dalam tabel tersebut.
- b. Masing-masing *cell* bernilai tunggal

Catatan: Permintaan yang menyatakan tidak ada baris yang duplikat dalam sebuah tabel berarti tabel tersebut memiliki sebuah kunci, meskipun kunci tersebut dibuat dari kombinasi lebih dari satu kolom atau bahkan kunci tersebut merupakan kombinasi dari semua kolom.

Tabel II.10. Tabel Normal 1

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 : 2016

3. Bentuk normal tahap kedua (2nd normal form)

Bentuk normal kedua (2NF) terpenuhi jika pada sebuah tabel semua atribut yang tidak termasuk dalam *primary key* memiliki ketergantungan fungsional pada *primary key* secara utuh.

Tabel II.11. 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 : 2016

4. Bentuk normal tahap ketiga (3rd normal form)

Sebuah tabel dikatakan memenuhi bentuk normal ketiga (3NF), jika untuk setiap ketergantungan fungsional dengan notasi $X \rightarrow A$, dimana A mewakili semua atribut tunggal di dalam tabel yang tidak ada di dalam X, maka :

- a. X haruslah *superkey* pada tabel tersebut.
- b. Atau A merupakan bagian dari *primary key* pada tabel tersebut.

Tabel II.12. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2016