

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun Penelitian terkait ini, peneliti akan membandingkan hasil penelitiannya dengan dua jurnal yaitu :

Berdasarkan penelitian Ridho Denanda Putra yang berjudul “Rancang Bangun Aplikasi Peramalan Permintaan Barang Dengan Metode Pemulusan Eksponensial Winter Pada PT. Supramedika Prima” Dihasilkan aplikasi yang dapat meramalkan permintaan barang untuk periode yang akan datang dengan masukan berupa data penjualan barang pada periode sebelumnya, dengan menggunakan Metode Pemulusan Eksponensial Winter yang secara keseluruhan memiliki tingkat persentase kesalahan peramalan sebesar 13,2%. Aplikasi peramalan permintaan barang dibuat dengan berbasis *desktop* untuk penggunaanya. (Ridho Denanda Putra : 2014 : 201).

Menurut Marjuki Dahlan (2012) dengan judul penelitian “Sistem Informasi Peramalan Penjualan Kripik Pisang Menggunakan Metode *Double Exponential Smoothing* Berbasis *Web* Pada Toko Tiga Putra Di Lumajang” dengan tujuan dari penelitian ini yaitu terbangunnya sebuah Sistem Informasi Peramalan Penjualan Kripik Pisang pada Toko Tiga Putra Menggunakan Metode *Double Exponential Smoothing*. Penelitian menghasilkan sebuah aplikasi sistem informasi peramalan penjualan kripik pisang menggunakan metode *Double exponential smoothing*

yang dapat digunakan untuk meramalkan jumlah penjualan kripik pisang yang dijual dimasa sekarang dan yang akan datang.

Menurut Fajar Riska (2014) dengan judul penelitian “Perbandingan Metode Des (Double Exponential Smoothing) Dengan Tes (Triple Exponential Smoothing) Pada Peramalan Penjualan Rokok (Studi Kasus Toko Utama Lumajang)” Toko Utama Daerah Lumajang masih mengalami permasalahan, petugas Toko Utama belum dapat mengetahui nilai peramalan penjualan terbaik dalam setiap periodenya. Sehubungan dengan hal tersebut sangat penting melakukan perbandingan tingkat akurasi peramalan penjualan di Toko Utama guna untuk mengetahui bagaimana membandingkan kedua metode untuk menghasilkan nilai peramalan terbaik disetiap bulannya.

Menurut Ni Ketut Dewi Ari Jayanti (2015) dengan judul “Penerapan Metode Triple Exponential Smoothing pada Sistem Peramalan Penentuan Stok Obat” Untuk memudahkan dalam menggali informasi yang tersimpan dalam data penjualan obat pada periode sebelumnya tersebut maka dibuat sebuah aplikasi yang diimplementasikan dengan bidang ilmu data mining dengan salah satu bagiannya adalah forecasting dengan menggunakan metode smoothing data deret berkala. Metode yang dijadikan dasar pada sistem ini adalah metode triple exponential smoothing dengan pola data deret berkala. Sistem ini membantu manajemen apotek dalam mengelola data penjualan dan menggali informasi untuk menentukan pengadaan stok obat yang tepat dan akurat. Keakuratan maksimal aplikasi ini dalam menentukan perkiraan penjualan obat mencapai 98,15%.

Menurut Radiant Victor Imbar (2013) dengan judul “Aplikasi Peramalan Stok Barang Menggunakan Metode *Double Exponential Smoothing*” Tujuan dari penelitian ini adalah : Membuat Sistem Informasi yang dapat proses pembelian dan penjualan barang secara terkomputerisasi, pengaturan stok barang, harga (satuan, grosir, dan beli) dari setiap barang yang ada dan pengecekan barang yang harus di retur dengan menerapkan aplikasi PHP Dan MySql.

Sedangkan penelitian yang dilakukan oleh penulis bertujuan untuk merancang sebuah sistem informasi prediksi penanaman bibit kelapa sawit pada PT. Palmanco Inti Sawit serta mempermudah perusahaan dalam mengetahui tingkat peramalan penanaman bibit kelapa sawit pada PT. Palmanco Inti Sawit, mempermudah penyampaian laporan perkembangan peramalan penanaman bibit kelapa sawit pada PT. Palmanco Inti Sawit agar meningkatkan kinerja perusahaan dan mempermudah masyarakat dalam mengetahui jumlah peramalan penanaman bibit kelapa sawit pada PT. Palmanco Inti Sawit. Bahasa pemrograman yang digunakan oleh penulis adalah Bahasa pemrograman Visual Studio 2010 dan Dql Server sebagai databasenya.

II.2. Uraian Teoritis

II.2.1. Sistem

Sistem merupakan sekumpulan unsur atau komponen dan prosedur yang harus berhubungan erat (*interrelated*) satu sama lain dan berfungsi secara bersama-sama agar tujuan yang sama (*common purpose*) dapat dicapai. Setiap sistem dibuat untuk mencapai suatu tuju (goal) atau sasaran (*objecties*). Tujuan (*goal*)

meliputi ruang lingkup yang luas, sedangkan sasaran (*objectivites*) lebih dikenal pada sub sistemnya karena meliputi ruang lingkup yang sempit dibanding tujuan. Tujuan maupun sasaran pada prinsipnya adalah sesuatu yang hendak diraih untuk memenuhi kebutuhan (*need*) dan keinginan (*wish*) serta sifatnya senantiasa positif (lebih mengarah pada kebaikan), sehingga alat ukur tercapainya sautu tujuan/ sasaran adalah kepuasan (*satisfaction*) (Mei Hotma Mariati : 2016).

II.2.2. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

1. Akurat (*Accurate*)

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

2. Tepat waktu (*Timelines*)

Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan (*Relevance*)

Berarti informasi tersebut bermanfaat bagi pemakainya (Deppi Linda, 2016 : 62-63).

II.2.3. Sistem Informasi

Sistem informasi adalah sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan. Penggunaan sistem informasi telah banyak diterapkan diberbagai bidang termasuk dalam bisnis. Salah satu tujuan penerapan sistem informasi dalam bidang bisnis agar dapat meningkatkan keuntungan bisnis dengan menggunakan kemampuan yang didapatkan dari sistem informasi. Ada beberapa kemampuan dari sistem informasi yang dapat mendukung dalam bidang bisnis. Kemampuan tersebut seperti pengurangan biaya, mempercepat pekerjaan, dapat meningkatkan kemudahan dalam pengambilan keputusan, dan peningkatan pelayanan terhadap pelanggan. (Alfian Nurlifa : 2017)

II.2.4. Konsep Sistem Informasi

Faktor-faktor yang menentukan kehandalan dari suatu sistem informasi atau informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

a) Keunggulan (*Usefulness*)

Yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

b) Ekonomis

Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.

c) Keandalan (*Reliability*)

Keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan sistem tersebut harus beroperasi secara efektif.

d) Pelayanan (*Customer Service*)

Yakni suatu sistem memberikan pelayanan yang baik dan efisien kepada para pengguna sistem pada saat berhubungan dengan organisasi.

e) Kapasitas (*Capacity*)

Setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.

f) Sederhana dalam kemudahan (*Simplicity*)

Sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedur mudah diikuti.

g) Fleksibel (*Flexibility*)

Sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.

h) Komponen Sistem Informasi

Istilah dalam komponen sistem informasi adalah blok bangunan (building block) yang dapat di bagi menjadi enam blok yaitu :

a. Blok masukan (*Input block*)

Blok *input* merupakan data–data yang masuk ke dalam sistem informasi, yang dapat berupa *document-document* dasar yang dapat diolah menjadi suatu informasi tertentu.

b. Blok model (*Model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan mengolah data *input* untuk menghasilkan suatu informasi yang dibutuhkan.

c. Blok keluaran (*Output block*)

Merupakan informasi yang menghasilkan sekumpulan data yang nantinya akan disimpan berupa data cetak laporan.

d. Blok teknologi (*Technologi block*)

Blok teknologi merupakan penunjang utama dalam berlangsungnya sistem informasi. Yang memiliki beberapa komponen yaitu diantaranya alat memasukkan data (*input device*), alat untuk menyimpan dan mengakses data (*storege device*), alat untuk menghasilkan dan mengirimkan keluaran (*output divice*) dan alat untuk membentuk pengendalian sistem secara keseluruhan (*control device*). Teknologi informasi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*humanware* atau *braiware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data (*Database block*)

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan

digunakan perangkat lunak untuk memanipulasinya. Data perlu di simpan dan perlu di organisasi sedemikian rupa, supaya informasi yang dihasilkan berkualitas.

f. Blok kendali (*Control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah bila terlanjur terjadi. (I Made Budi Adnyana, 2016)

II.2.5. Triple Exponential Smoothing

Metode ini digunakan ketika terdapat unsur trend dan perilaku musiman yang ditunjukkan pada data. *Metode Exponential Smoothing* yang dapat digunakan untuk hampir segala jenis data stasioner atau non –stasioner sepanjang data tersebut tidak mengandung faktor musiman. Tetapi bila mana terdapat data musiman, metode triple dapat dijadikan cara untuk meramalkan data yang mengandung faktor musiman tersebut. (Muhammad Iqbal : 2017)

Berikut adalah persamaan-persamaan yang digunakan untuk melakukan peramalan dengan menggunakan metode *Triple Exponential Smoothing* :

$$S'_t = a X_t + (1 - a) S'_{t-1} \dots\dots\dots(1)$$

$$S''_t = a S'_t + (1 - a) S''_{t-1} \dots\dots\dots(2)$$

$$S'''_t = a S''_t + (1 - a) S'''_{t-1} \dots\dots\dots(3)$$

$$a_t = 3S'_t - 3S''_t + S'''_t \dots\dots\dots(4)$$

$$b_t = a / 2(1 - a)(6 - 5.a)S'_t - (10 - 8.a)S''_t + (4 - 3.a) S'''_t \dots\dots\dots(5)$$

$$c_t = a^2(1 - a)^2 (S''_t - 2S'''_t + S''''_t) \dots\dots\dots(6)$$

$$F_{t+m} = a_t + b_t(1) + \frac{1}{2} c_t(1) \dots\dots\dots(7)$$

Keterangan :

$S t'$ = nilai pemulusan eksponensial tunggal (*Single*)

$S t''$ = nilai pemulusan eksponensial ganda (*Double*)

$S t'''$ = nilai pemulusan eksponensial rangkap tiga (*Triple*)

α_p = parameter pemulusan eksponensial yang besarnya $0 < \alpha_p < 1$

a_t, b_t, c_t = konstanta pemulusan

F_{t+m} = hasil peramalan periode ke depan yang di ramalkan. (Muhammad

Iqbal : 2017)

II.2.6. Visual Studio 2010

Visual studio merupakan suatu perangkat lunak yang dapat digunakan untuk pengembangan berbagai macam aplikasi yang memiliki berbagai macam tipe antara lain aplikasi *desktop* (*Windows Form, CommandLine (Console)*), *Aplikasi Web, Windows Mobile (Paket PC)*. Visual Studio 2010 memiliki lebih dari satu kompiler, SDK (*Software Development Kit*), dan Dokumentasi Tutorial (*MSDN Library*). Kompiler yang dimasukkan kedalam Visual Studio 2010 antara lain Visual Basic, Visual C#, Visual C++, Visual InterDev, Visual J++, Visual F#, dan Visual Source Safe, dan banyak yang lainnya. Dan semua itu sudah terpaket dan diperuntukkan kedalam platform .Net Framework 4.0 atau versi yang lebih tinggi. Visual studio ini dapat digunakan untuk membuat aplikasi yang berbasis desktop yang merupakan platform windows, namun juga dapat dijalankan dalam bentuk Microsoft Intermediate Language diatas .Net Framework. Selain itu Visual

Studio juga dapat digunakan untuk membuat aplikasi yang dapat dijalankan diatas windows mobile yang berjalan diatas .Net Compact Framework. (Rolly Yesputra : 2017 : 1).

Visual Studio 2010 terbagi menjadi beberapa tipe diantaranya :

1. Visual Studio 2010 Express Edition yang bisa digunakan secara gratis tanpa memberikan royalti kepada Microsoft Inc.
2. Visual Studio Standard Edition
3. Visual Studio 2010 Professional Edit
4. Visual Studio 2010 Ultimate Edition (Rolly Yesputra : 2017 : 1).

II.2.7. SQL Server 2008

SQL Server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang *database*. SQL Server adalah DBMS (*Database Management System*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. SQL Server 2008 dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL Server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data. Microsoft merilis SQL Server 2008 dalam beberapa versi yang disesuaikan dengan segment-segment pasar yang dituju. Versi-versi tersebut adalah sebagai berikut. Menurut cara pemrosesan data pada prosesor maka Microsoft mengelompokkan produk ini berdasarkan 2 jenis yaitu :

- a. Versi 32-bit(x86), yang biasanya digunakan untuk komputer dengan singleprosesor (Pentium 4) atau lebih tepatnya prosesor 32 bit dan sistem operasi Windows XP.
- b. Versi 64-bit(x64), yang biasanya digunakan untuk komputer dengan lebih dari satu prosesor (Misalnya Core 2 Duo) dan system operasi 64 bit seperti Windows XP 64, Vista, dan Windows 7. Sedangkan secara keseluruhan terdapat versi-versi seperti berikut ini:
- c. Versi Compact, ini adalah versi “Tipis” dari semua versi yang ada. Versi ini seperti versi desktop pada SQL Server 2000. Versi ini juga digunakan pada handled drvice seperti Pocket PC, PDA, SmartPhone, Tablet PC.
- d. Versi Express, ini adalah versi “Ringan” dari semua versi yang ada (tetapi versi ini berbeda dengan versi compact) dan paling cocok untuk latihan. Express Manager standar, integrasi dengan CLR dan XML. (Agus Tinus Setiawan;2016:54).

II.2.8 UML (*Unified Modeling Language*)


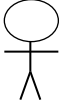

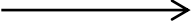
Menurut Gata (2013 : 4) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi 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. 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.1. 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> .
	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> .
	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 mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.




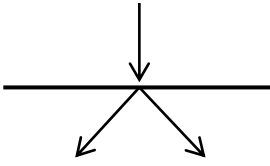
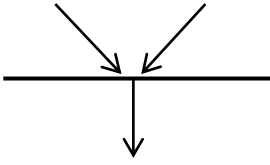
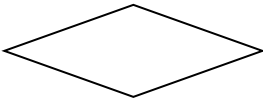
----->	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
<-----	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(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.2. Simbol *Activity Diagram*

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> .

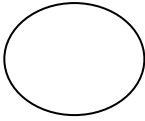
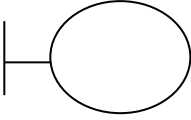
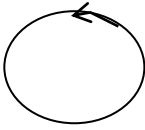
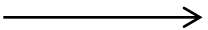
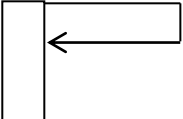

New Swimline	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.
--------------	--

(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.3. Simbol *Sequence Diagram*

Gambar	Keterangan
	<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.
	<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 : 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.4. *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.2.9. Basis Data (*Database*)

Pangkalan data atau basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa

menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi. Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Model yang umum digunakan sekarang adalah model relasional, yang mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel (Neni Purwati, 2015 : 50).

II.2.10. 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.

Proses pernomalan tabel ada beberapa tahap yang harus dilakukan yaitu :

1. *First Normal Form* (1 NF)

Sudah tidak ada repeating group yaitu pengulangan yang terjadi pada beberapa atribut atau kolom dalam sebuah tabel, dan juga setiap atribut harus bernilai tunggal. Atribut *multivalued*, *composite*, *derive* tidak tunggal. Setiap nilai dari atribut hanya mempunyai nilai tunggal.

2. *Second Normal Form* (2 NF)

Untuk menjadikan tabel normal tingkat ke 2 maka sudah 1NF dan setiap atribut yang bukan *primary key* sepenuhnya secara fungsional tergantung pada semua atribut pembentuk *primary key*.

3. *Third Normal Form* (3 NF)

Tabel sudah 2NF dan tidak memiliki *transitive dependencies*, *Transitive Dependencies* adalah ketika atribut yang secara tidak langsung tergantung pada *primary key* dan atribut tersebut juga tergantung pada atribut lain yang bukan *primary key*.

4. *Boyce-codd Normal Form* (BCNF)

Tabel dalam BCNF jika sudah 3NF dan semua *determinants* adalah *candidate keys*. Perbedaan 3NF dan BCNF adalah untuk *functional dependency* $A - B$, 3NF memperbolehkan ketergantungan ada dalam relasi

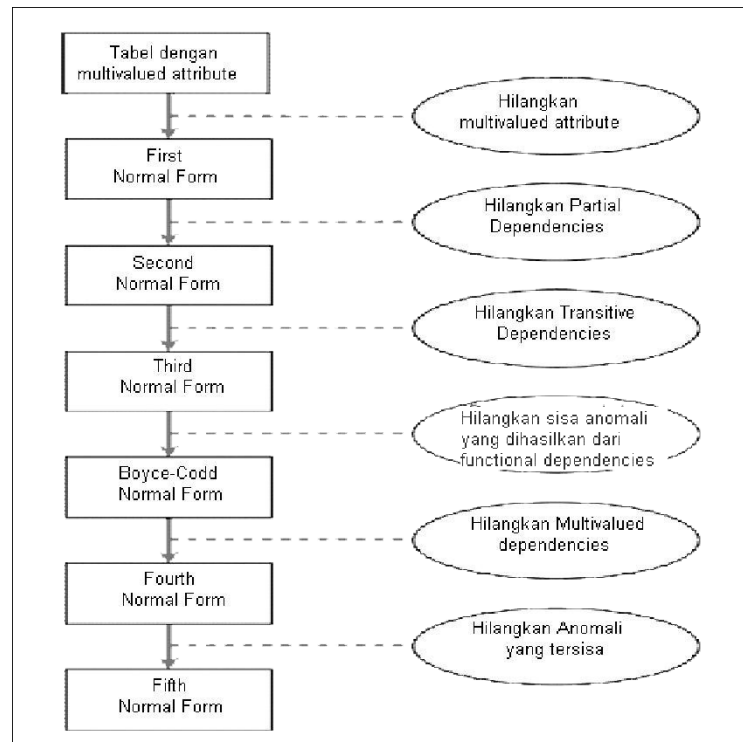
jika B adalah *primary key* dan A bukan merupakan *candidate key*. Sedangkan BCNF menuntut untuk ketergantungan tetap ada dalam relasi, A harus menjadi *candidate key*.

5. *Fourth Normal Form (4 NF)*

Relasi berada pada bentuk normal keempat apabila memenuhi syarat BCNF dan tidak mempunyai multivalued *dependency*.

6. *Fifth Normal Form (5 NF)*

Tabel bentuk normal kelima sering disebut PJNF (*Projection Join Normal Form*), Penyebutan PJNF karena untuk suatu relasi akan berbentuk normal kelima jika tabel tersebut dapat dipecah atau diprogersikan (kemajuan) menjadi beberapa tabel dan dari progersi (kemajuan) dapat disusun kembali (*join*) menjadi tabel yang sama dengan keadaan semula. Jika penyusunan ini tidak mungkin dilakukan dikatakan pada relasi itu terdapat *join dependencies* dan dikatakan bersifat *lossy join*.(Gandung Triyono : 2013 : 19-20)



Gambar II.1. Langkah-Langkah Dalam Pernomalan Tabel

(Sumber: Gandung Triyono; 2013: 20)

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data / database, 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 : 2016)

1. Proses Normalisasi

- a. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu kebeberapa 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.5. 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 : 2016

- 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.6. 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 : 2016

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.7. 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 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.

Tabel II.8. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2016