

BAB II

TINJAUAN PUSTAKA

II.1. Peneliti Terkait

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

Menurut Etri Pujiati (2017), yang berjudul “Peramalan Dengan Menggunakan Metode *Double Exponential Smoothing* Dari Brown (Studi Kasus: Indeks Harga Konsumen (IHK) Kota Samarinda)” Dari hasil peramalan yang telah didapatkan, diketahui bahwa IHK Kota Samarinda untuk 3 bulan yang akan datang yaitu dari bulan Januari, Februari dan Maret 2015 mengalami peningkatan setiap bulannya. Hal ini juga dapat dilihat pada Gambar 3. Yang menunjukkan kecenderungan pola data pada garis peramalan mengalami peningkatan untuk 3 bulan yang diramalkan yaitu bulan Januari 2015 sebesar 121,44, bulan Februari 2015 sebesar 123,06, dan bulan Maret 2015 sebesar 124,68.

Menurut Rudi Ariyanto (2017), yang berjudul “ Penerapan Metode *Double Exponential Smoothing* Pada Peramalan Produksi Tanaman Pangan” Produksi pangan adalah salah satu aspek kebutuhan yang sangat penting. Ketersediaan pangan mempengaruhi stabilitas ketahanan pangan. Beberapa komoditas tanaman pangan yang menjadi kebutuhan pokok adalah Padi, Jagung, Kedelai, dan Ubi Kayu, Ubi Jalar, Kacang Tanah dan Kacang Hijau. Jumlah produksi pangan yang tidak menentu menjadi suatu masalah bagi Badan Ketahanan Pangan (BKP) Provinsi Jawa Timur dalam menentukan kebijakan mendatang. Apabila prediksi

tidak diketahui maka dalam pembangunan Ketahanan Pangan dipastikan kurang maksimal dan alhasil Indonesia melakukan impor karena belum bisa memenuhi kebutuhan pangan. Diperlukan suatu peramalan dalam memprediksi produksi tanaman pangan di periode mendatang berdasarkan tahun dengan menggunakan data masa lalu. Sehingga dari pergerakan data masa lalu dapat dianalisa pergerakan *trend*.

Menurut Radiant Victor Imbar (2016) 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.

Menurut Agus Purwanto, (2017), dengan judul “Teknik Peramalan Dengan *Double Exponential Smoothing* Pada Distributor Gula” Distributor gula di Denpasar PT. Larasati, dari hasil analisa terhadap data dan informasi di PT. Larasati terdapat keterbatasan dalam menentukan jumlah stok gula yang harus disediakan oleh perusahaan, sehingga perusahaan menentukan jumlah stok dari permintaan pelanggan saja. Oleh karena itu, dibuatkanlah suatu analisis perhitungan dengan sebuah teknik peramalan menggunakan metode *Double Exponential Smoothing*. Dengan metode ini akan menghasilkan sebuah perhitungan untuk persediaan stok gula di masa yang akan datang. berdasarkan data yang sudah terbentuk di bulan sebelumnya. dapat membantu perusahaan dalam meramalkan atau menentukan jumlah persediaan stok gula dan dalam

proses perhitungan peramalannya diketahui jika semakin banyak jumlah data yang digunakan dalam perhitungan peramalan maka hasil persentase errornya semakin kecil

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 Nendang Kacikal, 2015, yang berjudul “Sistem Peramalan Dan Monitoring Persediaan Obat Di Rspg Cisarua Bogor Dengan Menggunakan Metode Dole Exponential Smoothing Dan Reorder Point” Berdasarkan hasil pengujian, dapat disimpulkan bahwa metode Single Exponential Smoothing cocok untuk digunakan karena rata-rata akurasi peramalan mencapai 93,4%, selain itu sistem yang dibangun membantu Kepala Instalasi Farmasi dalam menentukan jumlah obat yang harus disediakan dan membantu bagian gudang dalam memantau persediaan stok obat.

II.2. Penelitian terkait

II.2.1. Sistem

Definisi sistem adalah “kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama.” Definisi sistem adalah “sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai suatu kesatuan yang dirancang untuk mencapai suatu tujuan. Sistem adalah penggabungan dari bagian-bagian atau komponen-komponen yang terpisah-pisah dan disatukan menjadi satu rangkaian dan menjadi suatu fungsi yang baru (Aris : 2015).

Sistem informasi adalah berupa suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian yang mendukung operasi yang bersifat manajerial dengan kegiatan strategi suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Ratna Sri Hayati : 2015 : 700)

II.2.1.1. Karakteristik Sistem

Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya yaitu:

1. Batasan (*boundary*) adalah Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.

2. Lingkungan (*environment*) adalah Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*input*) adalah Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*output*) adalah Sumber daya atau produk (informasi, laporan, dokumen, tampilan layar Komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*) adalah Kegiatan-kegiatan atau proses dalam sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.
6. Penghubung (*Interface*) adalah Tempat di mana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*) adalah Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama (Aris : 2015).

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 sekumpulan komponen-komponen sistem yang berada didalam suatu ruang lingkup organisasi, saling berinteraksi untuk menghasilkan sebuah informasi yang bertujuan untuk pihak manajemen tertentu dan untuk mencapai tujuan tertentu. Faktor-faktor yang menentukan kehandalan dari suatu sistem informasi atau informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

1. Keunggulan (*usefulness*) yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.
2. Ekonomis yaitu kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.
3. Kehandalan (*Reliability*) yaitu keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan sistem tersebut harus beroperasi secara efektif.

4. Pelayanan (*Customer Service*) yakni suatu sistem memberikan pelayanan yang baik dan efisien kepada para pengguna sistem pada saat berhubungan dengan organisasi.
5. Kapasitas (*Capacity*) yaitu setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.
6. Sederhana dalam kemudahan (*Simplicity*) yaitu sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedur mudah diikuti.
7. Fleksibel (*Flexibility*) yaitu sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.
8. Komponen Sistem Informasi yaitu Istilah dalam komponen sistem informasi adalah blok bangunan (*building block*) yang dapat dibagi 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 ceta laporan.

d. Blok teknologi (*technology block*)

Blok teknologi merupakan penunjang utama dalam berlangsungnya sistem informasi. Yang memiliki beberapa komponen yaitu diantaranya alat memasukan data (*input device*), alat untuk menyimpan dan mengapses data (*storege device*), alat untuk menghasilkan dan mengirimkan keluaran (*output divice*) dan alat untuk membantuk pengendalian sistem secara keseluruhan (*control device*). Teknologi informasi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*humanware* atau *brainare*), 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.2.4. *Double Exponential Smoothing*

Metode *Double Exponential Smoothing* merupakan model linear yang dikemukakan oleh Brown. Dalam metode ini dilakukan proses smoothing dua kali. Dasar pemikiran metode pemulusan eksponensial linear dari Brown adalah serupa dengan rata-rata bergerak linear, karena kedua nilai pemulusan tunggal dan ganda ketinggalan dari data yang sebenarnya jika terdapat unsur trend. Perbedaan antara nilai pemulusan tunggal dan ganda dapat ditambahkan dengan nilai pemulusan tunggal dan disesuaikan untuk trend. Persamaan yang dipakai dalam implementasi pemulusan eksponensial linear satu-parameter. (Cahyarizki Adi Utama :2016)

Adapun rumus dari perhitungan metode *Double Exponential Smoothing* adalah sebagai berikut : (Cahyarizki Adi Utama :2016)

1. Pemulusan Eksponensial Tunggal:

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

2. Pemulusan Eksponensial Ganda:

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

Dimana S'_t adalah nilai pemulusan eksponensial tunggal dan S''_t adalah nilai pemulusan eksponensial ganda

3. Pemulusan Trend:

$$a_t = S'_t + (S'_t - S''_t) = 2 S'_t - S''_t$$

4. Pemulusan Trend:

$$b_t = \frac{a}{1-a} ((S'_t - S''_t))$$

5. Ramalan

$$F_{t+m} = a_t + b_t(m) \quad \dots\dots\dots(5)$$

Dimana m adalah jumlah periode ke muka yang akan diramalkan.

Keterangan :

$$F_{t+m} = \text{Nilai ramalan untuk } m \text{ periode ke muka} \quad \dots\dots\dots(4)$$

M = Jarak periode yang akan diramalkan

X_t = Nilai actual periode ke-t

S'_t = Nilai *Smoothing period* ke-t

a = Konstanta *Smoothing* (1/n)

Studi Kasus Metode *Double Exponential Smoothing*

Berikut ini contoh kasus menggunakan metode *Double Exponential Smoothing*

Tabel II.1. Data Barang Keluar

Bulan	Jumlah Barang Keluar (Unit)
Jan-14	255 Unit
Feb-14	210 Unit
Mar-14	262 Unit
Apr-14	295 Unit
Mei-14	297 Unit
Jun-14	271 Unit
Jul-14	301 Unit
Agust-14	293 Unit
Sep-14	311 Unit
Okt-14	315 Unit
Nop-14	325 Unit
Des-14	342 Unit

Data dari jenis barang tersebut sebanyak 12 periode. Maka akan meramalkan Periode ke-13.

$$S'_t = aX_t + (1 - a) S'_{t-1}$$

$$S'_1 = 255$$

$$S'_2 = (0,3)210 + (1 - 0,3)255 = 241,5$$

$$S'_3 = (0,3)262 + (1 - 0,3)241,5 = 247,65$$

$$S'_4 = (0,3)295 + (1 - 0,3)247,65 = 261,855$$

$$S'_5 = (0,3)297 + (1 - 0,3)261,855 = 272,3985$$

$$S'_6 = (0,3)271 + (1 - 0,3)272,3985 = 271,97895$$

$$S'_7 = (0,3)301 + (1 - 0,3)271,97895 = 280,685265$$

$$S'_8 = (0,3)293 + (1 - 0,3)280,685265 = 284,3796855$$

$$S'_9 = (0,3)311 + (1 - 0,3)284,3796855 = 292,3657799$$

$$S'_{10} = (0,3)315 + (1 - 0,3)292,3657799 = 299,1560459$$

$$S'_{11} = (0,3)325 + (1 - 0,3)299,1560459 = 306,9092321$$

$$S'_{12} = (0,3)342 + (1 - 0,3)306,9092321 = 317,4364625$$

$$S''_t = aS'_t + (1 - a)S''_{t-1}$$

$$S''_1 = 255$$

$$S''_2 = (0,3)241,5 + (1 - 0,3)255 = 250,95$$

$$S''_3 = (0,3)247,65 + (1 - 0,3)250,95 = 249,96$$

$$S''_4 = (0,3)261,855 + (1 - 0,3)249,96 = 253,5285$$

$$S''_5 = (0,3)272,3985 + (1 - 0,3)253,5285 = 259,1895$$

$$S''_6 = (0,3)271,97895 + (1 - 0,3)259,1895 = 263,026335$$

$$S''_7 = (0,3)280,685265 + (1 - 0,3)263,026335 = 268,324014$$

$$S''_8 = (0,3)284,3796855 + (1 - 0,3)268,324014 = 273,1407155$$

$$S''_9 = (0,3)292,3657799 + (1 - 0,3)273,1407155 = 278,9082348$$

$$S''_{10} = (0,3)299,1560459 + (1 - 0,3)278,9082348 = 284,9825781$$

$$S''_{11} = (0,3)306,9092321 + (1 - 0,3)284,9825781 = 291,5605743$$

$$S''_{12} = (0,3)317,4364625 + (1 - 0,3)291,5605743 = 299,3233408$$

$$a_t = S'_t + (S'_t - S''_t) = 2 S'_t - S''_t$$

$$a_1 = 2(255) - 255 = 255$$

$$a_2 = 2(241,5) - 250,95 = 232,05$$

$$a_3 = 2(247,65) - 249,96 = 245,34$$

$$a_4 = 2(261,855) - 253,5285 = 270,1815$$

$$a_5 = 2(272,3985) - 259,1895 = 285,6075$$

$$a_6 = 2(271,97895) - 263,026335 = 280,931565$$

$$a_7 = 2(280,685265) - 268,324014 = 293,046516$$

$$a_8 = 2(284,3796855) - 273,1407155 = 295,6186556$$

$$a_9 = 2(292,3657799) - 278,9082348 = 305,8233249$$

$$a_{10} = 2(299,1560459) - 284,9825781 = 313,3295137$$

$$a_{11} = 2(306,9092321) - 291,5605743 = 322,2578899$$

$$a_{12} = 2(317,4364625) - 299,3233408 = 335,5495842$$

$$b_t = \frac{a}{1-a} ((S'_t - S''_t))$$

$$b_1 = \frac{0,3}{1-0,3} (255 - 255) = 0$$

$$b_2 = \frac{0,3}{1-0,3} (241,5 - 250,95) = -4,05$$

$$b_3 = \frac{0,3}{1-0,3} (247,65 - 249,96) = -0,99$$

$$b_4 = \frac{0,3}{1-0,3} (261,855 - 253,5285) = 3,5685$$

$$b_5 = \frac{0,3}{1-0,3} (272,3985 - 259,1895) = 5,661$$

$$b_6 = \frac{0,3}{1-0,3} (271,97895 - 263,026335) = 3,836835$$

$$b_7 = \frac{0,3}{1-0,3} (280,685265 - 268,324014) = 5,297679$$

$$b_8 = \frac{0,3}{1-0,3} (284,3796855 - 273,1407155) = 4,81670145$$

$$b_9 = \frac{0,3}{1-0,3} (292,3657799 - 278,9082348) = 5,76751932$$

$$b_{10} = \frac{0,3}{1-0,3} (299,1560459 - 284,9825781) = 6,074343338$$

$$b_{11} = \frac{0,3}{1-0,3} (306,9092321 - 291,5605743) = 6,577996206$$

$$b_{12} = \frac{0,3}{1-0,3} (317,4364625 - 299,3233408) = 7,762766453$$

$$F_t + m = a_t + b_t(m)$$

$$F_{12+1} = a_t + b_t(1)$$

$$F_{13} = 335,5495842 + 7,762766453(1) = 343,3123507$$

Hasil peramalan untuk periode ke-13 adalah 343,3123507. (Cahyarizki Adi Utama : 2016)

II.2.5. 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. (Neni Purwati dan Hendra Kurniawan, 2015 : 50).

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 dan Hendra Kurniawan, 2015 : 50).

II.2.6. Normalisasi

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 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.2. 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.3. 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.4. 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.5. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2016

II.2.7. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang

berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Untuk mendapatkannya dapat men-*download* langsung dari *web* resminya. (Randi V Palit, 2015 : 2).

II.2.8. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah *web* dan bisa digunakan pada dokumen HTML.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 dimana aplikasi tersebut yang dibangun oleh PHP pada umumnya akan memberikan hasil pada *web browser*, tetapi proses secara keseluruhan dijalankan di server. (Saipul Anwar, 2016)

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, *Java*, dan *Perl* serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi *server*.

Sederhananya, *server* yang akan menterjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun pengertian lain 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”. (Astria Firman, 2016 : 30).

PHP merupakan Bahasa pemograman yang digunakan untuk membuat website dinamis dan interaktif. Dinamis artinya, website tersebut biasa berubah-ubah tampilan dan kontennya sesuai kondisi tertentu. Sebagai contoh, PHP biasa menampilkan tanggal dan hari saat ini secara berganti-ganti didalam sebuah website. Interaktif artinya, PHP dapat memberi feedback bagi user (misalnya menampilkan hasil pencarian produk). (Jubile Enterprise : 2018 ; 1).

II.2.9. 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 intuk aplikasi multi *user*. *MySQL* juga sering dikenal dengan nama sistem manajemen *database* relasional. Suatu *database* relasional menyimpan data dalam table yang terpisah. Tabel -table tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan Imemperoleh 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, 2016)

II.2.10 UML (*Unified Modeling Language*)

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan Bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi.

Unified Modeling Language (UML) biasa digunakan untuk :

- a. Menggambarkan batasan sistem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.
- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.

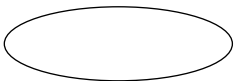
- f. Menyampaikan atau memperluas *functionality* dengan *stereotypes*. (Omni Alfina dan Fitriana Harahap : 2019)

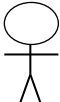

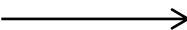
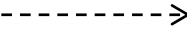

Menurut Gata (2013 : 4) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi 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.6. 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>


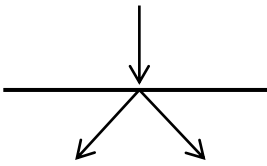
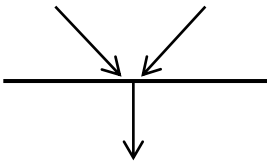
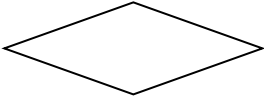

(Sumber : Gata, 2013 : 4)

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.7. 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>

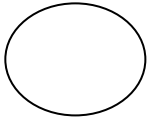
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.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

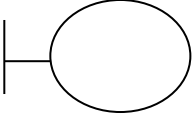
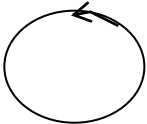

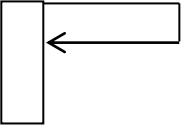


(Sumber : Gata, 2013 : 6)

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.8. 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.9. 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)