

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

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

Menurut Ruli (2017) yang berjudul “Perbandingan Metode *Holt Exponential Smoothing* dan *Winter Exponential Smoothing* Untuk Peramalan Penjualan Souvenir” Dalam penjualan souvenir, seringkali UD. Fajar Jaya mengalami kekosongan stok souvenir ketika pesanan membludak di waktu-waktu tertentu. Hal ini terjadi karena tidak terdapat analisa dan strategi manajemen stok (tidak dapat memprediksi berapa jumlah optimal souvenir yang harus di sediakan) yang di lakukan oleh eksekutif UD. Fajar Jaya. Untuk mengatasi hal tersebut di atas, maka penulis mengusulkan untuk melakukan peramalan terhadap tingkat penjualan souvenir dengan menggunakan metode *Holt* dan *Winter* yang ada pada pengembangan metode *Exponential Smoothing* (ES). Dari penerapan kedua metode tersebut, kemudian dibuat perbandingan efektifitas metode yang di ukur melalui akurasi data aktual dan hasil peramalan dengan cara mengetahui tingkat kesalahan peramalan.

Menurut Ridho (2018) 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.

Menurut Elisa (2017) dengan judul “Perbandingan Metode Winter Eksponensial Smoothing dan Metode Event Based untuk Menentukan Penjualan Produk Terbaik di Perusahaan X” Oleh sebab itu, dalam Tugas Akhir ini dilakukan peramalan dengan membandingkan dua metode yaitu metode Winter Eksponensial Smoothing dan metode Event Based untuk menentukan penjualan terbaik di perusahaan. Metode event based terdiri dari metode moving average event based dan eksponensial smoothing event based yang perhitungannya menggunakan indeks special event, dan kemudian hasilnya dibandingkan dengan metode winter eksponensial smoothing. Setelah dilakukan peramalan diperoleh hasil bahwa metode winter eksponensial smoothing sesuai untuk produk A dan produk B, yang memiliki tingkat error terkecil.

Menurut I Made (2019) dengan judul “Perbandingan Metode Holt Winter Additive Dan Metode Holt Winter Additive Damped Dalam Peramalan Jumlah Pendaftaran Mahasiswa” Tujuan penelitian ini adalah melakukan perbandingan hasil peramalan dari dua buah metode yang digunakan sehingga diketahui metode mana yang lebih tepat diterapkan untuk meramalkan jumlah pendaftaran mahasiswa baru. Metode Penelitian berupa pengembangan sistem, perhitungan peramalan, perhitungan nilai error, perbandingan nilai error dan pengambilan

kesimpulan. Hasil dari penelitian yaitu dengan kasus peramalan jumlah pendaftaran mahasiswa baru perhitungan nilai MAPE Metode Holt Winter Additive lebih baik dibandingkan dengan Metode Holt Winter Additive Damped. Nilai MAPE tahun 2017 untuk Metode Holt Winter Additive adalah 1,039% sedangkan Holt Winter Additive Damped 1,363%, begitu pula dengan tahun 2018 untuk Metode Holt Winter Additive adalah 0,793% sedangkan Holt Winter Additive Damped 1,214%.

Menurut Dila (2017) dengan judul “Penerapan Metode Holt Winter Dan Seasonal Arima Pada Peramalan Perkembangan Wisatawan Mancanegara Yang Datang Ke Indonesia” Data perkembangan wisatawan mancanegara yang datang ke Indonesia mengandung pola musiman dan pola trend, sehingga metode Holt Winter dan Seasonal ARIMA dapat digunakan untuk meramalkan jumlah wisatawan di masa yang akan datang. Metode peramalan terbaik dilihat dengan cara membandingkan nilai error terkecil. Tingkat error peramalan didapat berdasarkan perhitungan menggunakan *Mean Squared Deviation* (MSD), *Mean Absolute Deviation* (MAD), dan *Mean Absolute Percentage Error*.

II.2. Uraian Teoritis

II.2.1. Sistem

Sistem adalah kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama dan Sistem adalah penggabungan dari bagian-bagian atau komponen-komponen yang terpisah-pisah dan disatukan menjadi satu rangkaian dan menjadi suatu fungsi yang baru (Aris : 2017).

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*) : Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.
2. Lingkungan (*environment*) : Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*input*) : Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*output*) : Sumber daya atau produk (informasi, laporan, dokumen, tampilan layer komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*) : 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*) : Tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*): Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. (Aris : 2017)

II.2.2. Informasi

Informasi adalah data yang diolah menjadi bentuk yang berguna dan menjadi berarti bagi penerimanya. Kegunaan informasi adalah untuk mengurangi ketidak pastian di dalam proses pengambilan keputusan tentang suatu keadaan. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut. (Priyo Sutopo, dkk 2018).

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, 2018 : 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)

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. (I Made Budi Adnyana, 2018)

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 (*Fleksibility*)

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 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 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 dicegah bila terlanjur terjadi. (I Made Budi Adnyana, 2018)

II.2.4. 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).

II.2.5. Winter Exponential Smoothing

Model *Winter Exponential Smoothing (Triple Exponential Smoothing Multiplicative)* ini digunakan untuk meramalkan data dengan pola musiman.

Berbeda dengan model *Holt Double Exponential Smoothing*, model *Triple Exponential Smoothing Multiplicative* menggunakan tiga parameter untuk mencapai nilai peramalan yaitu α , β , dan μ . Langkah untuk melakukan peramalan terdiri dari 3 bagian yaitu proses peramalan (A_t), proses estimasi *trend* (T_t), dan proses estimasi musiman (S_{t+p}). Rumus yang digunakan untuk menghitung peramalan dapat dilihat pada Persamaan II.1. hingga II.4:

$$A_t = \alpha \frac{Y_t}{S_{t-L}} + (1-\alpha)(A_{t-1} + T_{t-1}) \dots \dots \dots (II.1)$$

$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1} \dots \dots \dots (II.2)$$

$$S_t = \mu \frac{Y_t}{A_t} + (1-\mu)S_{t-L} \dots \dots \dots (II.3)$$

Sama halnya dengan model *Holt Double Exponential Smoothing*, nilai A_1 dapat diasumsikan sama dengan nilai data aktual yang pertama yaitu Y_1 . Sedangkan nilai T_1 dapat diasumsikan bernilai 0 (karena nilai *trend* yang diperoleh dari periode yang lalu tidak ada). Dan nilai estimasi musiman awal (S_1) diasumsikan dengan nilai 1 (menghilangkan pengaruh musiman pada data aktual). Rumus untuk menentukan nilai ramalannya adalah sebagai berikut:

$$Y_{t+p} = (A_t + T_{t+p})S_{t-L+p} \dots \dots \dots (II.4)$$

Dengan ketentuan:

μ : Parameter pemulusan untuk estimasi musiman ($0 < \mu < 1$)

S_t : Estimasi musiman ke – t

L : Panjangnya musim. (Ruli Utami : 2017)

Berikut ini contoh kasus menggunakan metode Winter Exponential Smoothing:

Data Penjualan Barang Tulip Anti A dapat dilihat pada Tabel II.1:

Tabel II.1. Data Penjualan Barang Tulip Anti A

2009		2010	
Bulan	Data	Bulan	Data
1	325	13	365
2	375	14	553
3	389	15	402
4	430	16	421
5	320	17	614
6	456	18	422
7	423	19	442
8	530	20	558
9	399	21	450
10	271	22	524
11	715	23	503
12	673	24	362
2011		2012	
Bulan	Data	Bulan	Data
25	634	37	615
26	495	38	575
27	460	39	515
28	378	40	687
29	708	41	608
30	565	42	511
31	441	43	885
32	620	44	518
33	633	45	708
34	705	46	794
35	620	47	909
36	440	48	480

Berikut adalah implementasi perhitungan model peramalan berdasarkan persamaan (1) hingga persamaan (4):

1. Perhitungan Pemulusan Eksponensial

$$A_t = a \frac{Y_t}{S_{t-L}} + (1-a) (A_{t-1} + T_{t-1})$$

$$A_{13} = 0.4 \frac{Y_{13}}{S_{13-12}} + (1-0.4) (A_{13-1} + T_{13-1})$$

$$A_{13} = 0.4 \frac{365}{1} + (0,6) (586.6 + 19.9)$$

$$A_{13} = 146 + 363.9$$

$$A_{13} = 509.9$$

2. Perhitungan Estimasi *Trend*

$$T_t = \beta(A_t - A_{t-1}) + (1-\beta) T_{t-1}$$

$$T_{13} = 0,1 (A_{13} - A_{13-1}) + (1-0.1) T_{13-1}$$

$$T_{13} = 0,1 (509.9 - 586.6) + (0.9)19.9$$

$$T_{13} = (-7.67) + 17.91$$

$$T_{13} = 10.2$$

3. Perhitungan Estimasi Musiman

$$S_t = \gamma \frac{Y_t}{A_t} + (1 - \gamma) S_{t-L}$$

$$S_t = 0.3 \frac{Y_{13}}{A_{13}} + (1-0.13) s_{13-12}$$

$$S_t = 0.3 \frac{365}{509.9} + (0.7) 1$$

$$S_t = 0.214 + 0.7$$

$$S_t = 0.91$$

4. Perhitungan Peramalan pada Periode p di Masa Mendatang

$$\gamma_{t+p} = (A_t + p T_t) S_{t-L+p}$$

$$\gamma_{13+1} = (A_{13} + 1 T_{13}) S_{13-12+1}$$

$$\gamma_{13+1} = (509.9 + 1 10.2) 1$$

$$\gamma_{13+1} = (520.1) 1,03$$

$$\gamma_{13+1} = 533.7$$

Setelah implementasi perhitungan dari model peramalan, selanjutnya adalah menghitung kesalahan atau *error* yang dihasilkan oleh perhitungan tersebut. Berikut adalah perhitungan *error* berdasarkan persamaan (5):

$$e_{13} = \gamma_{13} - Y_{13}$$

$$e_{13} = 365 - 606.5$$

$$e_{13} = -241.53$$

Perhitungan *error* tersebut akan selalu dihitung mulai dari periode setelah panjang musim (periode ke-13) hingga total jumlah data yang digunakan untuk perhitungan peramalan, yang terdapat pada Tabel 2 kolom *Error*. Setelah mendapatkan nilai *error* pada periode yang ditentukan, selanjutnya adalah menghitung nilai MPE. Pada perhitungan nilai MPE, lebih mudah dilakukan apabila mengetahui nilai *Mean Error* (ME) terlebih dahulu. Nilai ME adalah rata-rata dari seluruh nilai *error* pada periode yang ditentukan. Berikut adalah perhitungan MPE berdasarkan persamaan (6):

$$MPE = \frac{\sum_{t=1}^{48} \left(\frac{Y_{13} - Y_{13}}{Y_{13}} \right)}{48}$$

$$MPE = \frac{\sum_{t=1}^{48} \left(\frac{Y_{13} - Y_{13}}{Y_{13}} \right) + \sum_{t=1}^{48} \left(\frac{Y_{14} - Y_{14}}{Y_{14}} \right) + \dots \dots \dots}{48}$$

$$MPE = \frac{\left(\frac{365 - 606.5}{365} \right) + \left(\frac{553 - 333.7}{553} \right) + \dots \dots \dots}{48}$$

$$MPE = -5$$

II.2.6. 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 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. Adapun tabel tidak normal dapat dilihat pada Tabel II.2:

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 : 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. Adapun tabel normal kedua dapat dilihat pada Tabel II.3:

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 : 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. Adapun tabel normal kedua dapat dilihat pada Tabel II.4:

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 : 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), Adapun tabel normal ketiga dapat dilihat pada dapat dilihat pada Tabel II.5:

Tabel II.5. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2018

II.2.7. 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.2.8. 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 : 2018)

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 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)

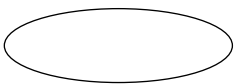
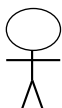
II.2.10. UML (*Unified Modelling Language*)

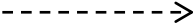
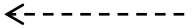
Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga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

1. *Diagram Use Case (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, dapat dilihat pada Tabel II.6:

Tabel II.6. 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 mengindikasikan aliran data.
	Asosiasi antara aktor dan use case yang menggunakan panah terbuka untuk mengindikasikan 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.

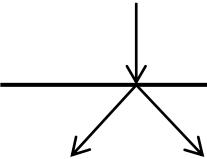
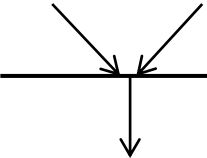
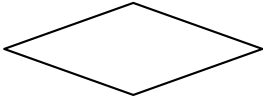
(Sumber : Ade Handini : 2018)

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*, dapat dilihat pada Tabel II.7:

Tabel II.7. 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>Activities</i> , menggambarkan suatu proses/kegiatan bisnis.

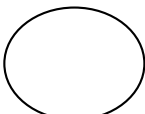
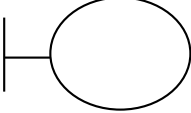
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk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

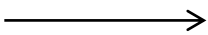
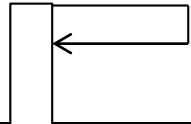
(Sumber : Ade Handini : 2018)

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*, dapat dilihat pada Tabel II.8:

Tabel II.8. 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 dapat dilihat pada Tabel II.9:

Tabel II.9. 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)