

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

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

1. 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.
2. 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%.

3. Menurut Radiant Victor Imbar (2014) 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.
4. Menurut Haryanto Tanuwijaya (2014), yang berjudul “Penerapan Metode *Winter’s Exponential Smoothing* Dan *Single Moving Average* Dalam Sistem Informasi Pengadaan Obat Rumah Sakit” Tujuan penelitian ini adalah membandingkan kedua metode dengan mengambil hasil peramalan tingkat kesalahan terkecil untuk menghasilkan informasi jumlah kebutuhan pemakaian obat bulan berikutnya. Penelitian ini dapat memberi kontribusi teori melalui penyediaan literatur tentang penerapan dan perbandingan metode *winter’s exponential smoothing* (WES) dan *single moving average* (SMA). Penelitian ini juga memberi kontribusi bagi kepala departemen pengadaan obat agar dapat mengambil keputusan pembelian obat dari *supplier* dengan tepat.

5. Menurut Ridho Denanda Putra (2014), 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. [2]
6. Menurut penelitian oleh Cahyarizki Adi Utama, 2016, Untuk meminimalkan dan mengatasi permasalahan tersebut perlu adanya suatu system terkomputerisasi yang dapat meramalkan kebutuhan stok barang secara optimal. Dan dapat memenuhi kebutuhan barang pada saat pelayanan. Metode yang digunakan adalah *Double Exponential Smoothing* karena Metode ini tergolong dalam metode time series (runtut waktu) yang mempergunakan data masa lalu untuk memprediksi sesuatu di masa yang akan datang. Data yang akan digunakan juga menunjukkan adanya trend. Sistem ini dapat meramalkan kebutuhan stok barang untuk 1 periode / 1 bulan kedepan agar stok barang optimal. Dalam peramalan menggunakan metode ini nilai alpha (α) sangat berpengaruh terhadap kesalahan / *error* dari peramalan.

II.2. Uraian Teoritis

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 dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai misal, sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar, dan kemudian subsistem-subsistem tersebut akan berinteraksi sedemikian rupa sehingga dapat mencapai satu kesatuan yang terpadu. Menurut (Jogiyanto, 2011) dalam buku *Analisa dan Design Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu (Deppi Linda, 2016 : 62).

II.2.2. 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.2.3. 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.2.4. Sistem Informasi

Sistem informasi adalah suatu kumpulan dari komponen – komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi. System informasi dapat juga didefinisikan sebagai suatu system yang menerima sumber data sebagai *input* dan mengolahnya menjadi produk informasi sebagai *output*. System informasi merupakan suatu system 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 system 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.2.6. Penjualan

Penjualan merupakan sebuah proses dimana kebutuhan pembeli dan kebutuhan penjualan dipenuhi, melalui antar pertukaran informasidan kepentingan. Jadi konsep penjualan adalah cara untuk mempengaruhi konsumen untuk membeli produk yang ditawarkan. Dalam kenyataannya penjualan mempunyai dua sistem yang biasa diterapkan oleh suatu perusahaan dagang yaitu penjualan yang dilakukan dengan cara tunai dan penjualan yang dilakukan secara kredit atau sering disebut cara angsuran. (Ruslan : 2013 ; 39)

II.2.7. Persediaan

Persediaan barang merupakan banyaknya barang yang digunakan untuk memfasilitasi produksi atau untuk memuaskan permintaan konsumen. Persediaan barang meliputi barang yang dibeli dan disimpan untuk dijual kembali. Dalam hal ini, untuk memenuhi persediaan barang sebuah perusahaan dagang melakukan pembelian dan selanjutnya barang yang dibeli akan dijual kembali kepada konsumen. Persediaan berpengaruh terhadap neraca dan laporan Laba/ Rugi dalam neraca perusahaan dagang atau perusahaan Idustri, persediaan seringkali merupakan bagian yang sangat besar dari keseluruhan Aktiva lancar

yang dimiliki perusahaan ini merupakan bukti betapa pentingnya persediaan barang dagangan untuk kegiatan pembelian dan penjualan dalam operasi perusahaan semacam itu. Pencatatan persediaan yang efektif seringkali merupakan kunci keberhasilan operasi perusahaan disamping pimpinan perusahaan berusaha untuk mempertahankan kuantitas dan jenis persediaan yang cukup untuk memenuhi permintaan konsumen, pimpinan juga harus menjaga keseimbangan persediaan agar tidak terlalu tinggi dan juga tidak terlalu rendah. Persediaan yang terlalu kecil akan menimbulkan kekecewaan konsumen sebaliknya persediaan yang terlalu tinggi akan menyebabkan biaya penyimpanan dan pemeliharaan persediaan akan membengkak . (Ruslan ; 2013 : 38).

II.2.5. 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 dataderet 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).

Peramalan (forecasting) merupakan hal yang penting bagi setiap organisasi bisnis dan untuk setiap pengambilan keputusan manajemen yang sangat signifikan. Peramalan menjadi dasar bagi perencanaan jangka panjang perusahaan. Ketepatan hasil peramalan bisnis akan meningkatkan peluang tercapainya investasi yang menguntungkan perusahaan. Orang bisnis melakukan kegiatan untuk mencapai sesuatu pada waktu yang akan datang serta memperhitungkan kondisi yang mungkin terjadi pada waktu itu. Usaha untuk meminimalkan ketidakpastian itu lazim dilakukan dengan metode atau teknik peramalan tertentu. Sedangkan, menurut peramalan adalah kegiatan untuk memperkirakan apa yang akan terjadi pada masa yang akan datang. Ada banyak jenis peramalan. Misalnya Metode Pemulusan, Metode Box-Jenkins maupun Metode Proyeksi Trend dengan Regresi. (Puspa Linda : 2014)

II.2.8. *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:

$$a_t = S'_t + (S'_t - S''_t) = 2 S'_t - S''_t \dots\dots\dots(3)$$

4. Pemulusan Trend:

$$b_t = \frac{a}{1-a} ((S'_t - S''_t)) \dots\dots\dots(4)$$

5. Ramalan

$$F_t + m = a_t + b_t(m) \dots\dots\dots(5)$$

Dimana m adalah jumlah periode ke muka yang akan diramalkan.

Keterangan :

$F_t + m$ = Nilai ramalan untuk m *periode* ke depan

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)

II.2.9. 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 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.1. Contoh bentuk tidak normal (Unnormal)

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

Sumber : Mukhlisulfatih Latief : 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.2. Contoh Bentuk Normal Pertama (1NF)

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

Sumber : Mukhlisulfatih Latief : 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.3. Contoh Bentuk Normal Kedua (2NF)

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

Sumber : Mukhlisulfatih Latief : 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.4. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2016

II.2.10. PHP

PHP merupakan kependekan dan kata *Hypertext Preprocessor*. PHP tergolong sebagai perangkat lunak *open source* yang diatur dalam *aturan general purpose licences (GPL)*. Bahasa pemrograman PHP sangat cocok dikembangkan dalam *line* kegunaan *web*, karena PHP bisa diletakkan pada *script* HTML atau sebaliknya. PHP dikhususkan untuk pengembangan *web* dinamis. Maksudnya, PHP mampu menghasilkan *website* yang secara terus menerus hasilnya bisa berubah-ubah sesuai dengan pola yang diberikan. PHP tergolong juga sebagai bahasa pemrograman yang berbasis *server (server side scripting)*. (Hikmah, 2 ; 2015).

II.2.11. MYSQL

MySQL merupakan RDBMS (atau *server database*) yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat diakses oleh banyak *user*". 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 ; 2013)

II.2.12. *UML (Unified Modelling Language)*

Menurut Windu Gata (2013) 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.


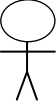

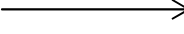
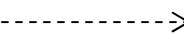
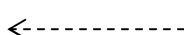
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. *UML* saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Gellysa Urva dan Helmi Fauzi Siregar, 2015, Hal : 93).

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 dapat dilihat pada tabel II.4 dibawah ini:

Tabel II.4. Simbol *Use Case*




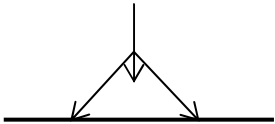
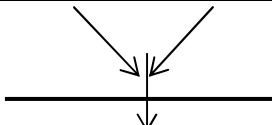
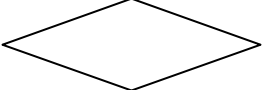
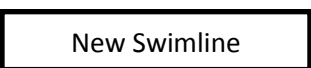
Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang 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 : Gellysa Urva dan Helmi Fauzi Siregar; 2015, Hal : 94)

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.5 dibawah ini:

Tabel II.5. Simbol *Activity Diagram*

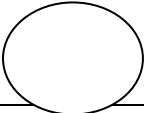
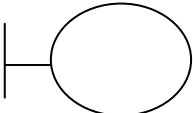
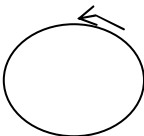
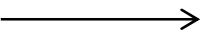
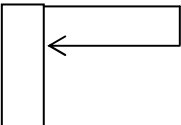


Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau 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> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Gellysa Urva dan Helmi Fauzi Siregar; 2015, Hal : 94)

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* dapat dilihat pada tabel II.6 dibawah ini :

Tabel II.6. 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 : Gellysa Urva dan Helmi Fauzi Siregar; 2015, Hal : 95)

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 yang dapat dilihat pada tabel II.7 dibawah ini :

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 : Gellysa Urva dan Helmi Fauzi Siregar; 2015, Hal : 95)