

BAB II

LANDASAN TEORI

II.1. Sistem

Sistem merupakan sekumpulan elemen – elemen yang saling terintegrasi serta melaksanakan fungsinya masing – masing untuk mencapai tujuan yang telah ditetapkan.

II.1.1. Karakteristik Sistem

Karakteristik sistem terdiri dari :

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen – komponen sistem atau elemen – elemen sistem dapat berupa suatu subsistem atau bagian – bagian dari sistem.

2. Batasan Sistem

Batasan merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber – sumber daya mengalir dari satu subsistem ke subsistem lainnya.

5. Masukan Sistem

Masukkan sistem adalah energi yang di masukkan ke dalam sistem. Masukkan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem

Keluaran sistem adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem

Suatu sistem dapat mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. (Sulindawati dan Fathoni : Vol.9 No.2 : 2010)

II.2. Informasi

informasi dapat didefinisikan sebagai hasil pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan kejadian-kejadian sebenarnya yang digunakan untuk pengambilan keputusan Menurut Hendi Haryadi (2009)

II.3. Sistem informasi

Sistem informasi adalah kumpulan perangkat keras dan perangkat lunak yang di rancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna. Bondar dan Hopwood (1993). Menurut Jogiyanto (2005:11) 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 harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

2. Ekonomis (*Economic*)

Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat.

3. Kehandalan (*Reliability*)

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*)

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

6. Sederhana dalam kemudahan (*Simplicity*)

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

7. Fleksibel (*Flexibility*)

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

II.3.1. Komponen Sistem Informasi

Menurut Jogiyanto (2005:12) Istilah dalam komponen sistem informasi adalah blok bangunan (*buildingblock*) yang dapat di bagi menjadi enam blok, yaitu

1. Blok masukan (*input block*)

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

2. 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.

3. Blok keluaran (*output block*)

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

4. Blok teknologi (*technology block*)

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

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

6. Blok kendali (control block)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah bila terlanjur terjadi kesalahan-kesalahan kerusakan dalam penggunaan sistem.

II.4. Sistem informasi akuntansi

Sistem informasi akuntansi merupakan kumpulan sumber daya seperti manusia dan peralatan yang dirancang untuk mengubah data keuangan dan data lainnya ke dalam informasi. Informasi tersebut dikomunikasikan kepada para pembuat keputusan. Sistem informasi akuntansi memalukan hal tersebut entah dengan sistem manual atau melalui sistem terkomputerisasi George H. Bodnar , (William S.Howoop ; 2006 : 98-99)

II.5. Sistem informasi Keuangan

Sistem informasi yang menyediakan informasi pada keuangan (departemen/bagian keuangan) yang menyangkut keuangan perusahaan. Misalnya ringkasan arus kas (*cash flow*) dan informasi pembayaran.(Abdul kadir ; 2003 :97)

II.6. Laporan Keuangan

Menurut Kamaludin dan Indriani. R (2011) Laporan keuangan adalah hasil akhir dari suatu proses pencatatan yang merupakan suatu ringkasan dari transaksi keuangan yang terjadi selama tahun buku yang bersangkutan.” Laporan

keuangan adalah laporan yang diterbitkan oleh suatu perusahaan berupa neraca, laporan laba rugi dan laporan perubahan modal yang merupakan catatan transaksi perusahaan dan perkembangan perusahaan selama periode tertentu.

II.6.1. Komponen Laporan Keuangan

Menurut Brigham dan Houston (2009) laporan keuangan terdiri dari empat komponen, yaitu:

1. Laporan Laba Rugi
2. Laporan Laba Ditahan

II.6.2. Jenis-jenis Analisis Rasio Keuangan

Menurut Kasmir (2010) jenis-jenis rasio keuangan yang biasa digunakan dalam analisis laporan keuangan antara lain

1. Rasio Likuiditas (*Liquidity Ratio*) Rasio analisa tentang kemampuan perusahaan untuk menyelesaikan kewajiban hutang jangka pendeknya.

Adapun yang tergabung dalam rasio ini adalah :

- a. Rasio Lancar (*Current Ratio*) Merupakan rasio yang digunakan untuk mengukur kemampuan perusahaan dalam membayar kewajiban jangka pendeknya
- b. Rasio Cepat (*Quick Ratio*) Merupakan rasio yang digunakan untuk mengukur kemampuan perusahaan dalam membayar kewajiban jangka pendeknya.

- c. Rasio Kas (*Cash Ratio*) Merupakan rasio yang digunakan untuk mengukur kemampuan perusahaan dalam membayar kewajiban jangka pendek..
 - d. Rasio Perputaran Kas (*Cash Turn Over Ratio*) Rasio perputaran kas berfungsi untuk mengukur tingkat kecukupan modal kerja perusahaan yang dibutuhkan untuk membayar tagihan dan membiayai penjualan.
2. Rasio Solvabilitas (*Leverage Ratio*) Rasio solvabilitas adalah rasio yang mengukur sejauh mana kemampuan perusahaan untuk membayar semua utang-utangnya, Adapun rasio-rasio yang tergabung dalam Rasio Solvabilitas, yaitu:.
- a. Rasio Hutang terhadap Total Aktiva (*Debt to Asset Ratio*) merupakan perbandingan antara hutang lancar dan hutang jangka panjang dan jumlah seluruh aktiva diketahui
 - b. Rasio Hutang terhadap Ekuitas (*Debt to Equity Ratio*) merupakan perbandingan antara hutang – hutang dan ekuitas.
 - c. *Long Term Debt to Equity Ratio* rasio ini digunakan untuk mengukur bagian dari modal sendiri yang dijadikan jaminan untuk hutang jangka panjang.
 - d. Rasio Kemampuan Membayar Bunga (*Times Interes Earned*) rasio ini dapat digunakan untuk mengetahui seberapa besar kemampuan perusahaan dalam membayar beban bunga.

3. Rasio Aktivitas (*Activity Ratio*) Rasio yang digunakan untuk mengukur efektivitas perusahaan dalam memanfaatkan sumber daya atau aktivitya.

Yang tergolong dalam rasio ini adalah:

- a. Perputaran Piutang (*Receivable Turn Over*) Rasio ini digunakan untuk mengukur berapa lama penagihan piutang selama satu periode.
 - b. Perputaran Persediaan (*Inventory Turn Over*) rasio ini digunakan untuk mengukur berapa kali dana yang ditanam dalam persediaan berputar dalam satu periode.
 - c. Perputaran Modal Kerja (*Working Capital Turn Over*) Rasio yang digunakan untuk mengukur atau menilai keefektifan modal kerja perusahaan.
 - d. Perputaran Aktiva Tetap (*Fixed Assets Turn Over*) Rasio ini digunakan untuk mengukur berapa kali dana yang ditanamkan dalam aktiva tetap berputar dalam satu periode.
 - e. Perputaran Aktiva (*Total Assets Turn Over*) *Total asset turn over* merupakan rasio yang digunakan untuk mengukur perputaran semua aktiva yang dimiliki perusahaan.
4. Rasio Profitabilitas (*Profitability Ratio*) Rasio profitabilitas adalah rasio untuk menilai kemampuan perusahaan dalam mencari keuntungan. Jenis-jenis rasio ini adalah:
- a. *Net Profit Margin* merupakan rasio yang digunakan untuk mengukur margin laba bersih setelah pajak.

- b. Hasil Pengembalian Investasi (*Return on Investment*) Rasio yang menunjukkan hasil atas jumlah yang digunakan dalam perusahaan.
- c. *Return on Equity* Rasio yang digunakan untuk mengukur laba bersih setelah pajak dengan *equity* perusahaan

II.7 Microsoft Visual Basic 2010

Visual basic adalah *visual basic* yang direkayasa kembalikan untuk digunakan pada *platform net* sehingga aplikasi yang dibuat menggunakan *visual basic* dapat berjalan pada sistem computer apapun dan dapat mengambil data server dengan tipe apa pun asalkan terinstal.(Edhy Sutanta ;2011 : 5)

II.8. MYSQL

MySql adalah perangkat lunak basis data server yang terkenal dan bersifat open source dengan dukungan driver yang luas dari berbagai vendor. *MySql* adalah seakuntansi implementasi dari system manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. (Supardi ; 2007).

II.9. Database

Database adalah sekumpulan table-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap table yang ada. Satu database

menunjukkan satu kumpulan data yang dipakai didalam suatu lingkup perusahaan atau instansi (asri anda dan fadlisyah ;2008 :1)

II.10 Normalisasi

Menurut Martin (1975) Normalisasi diartikan sebagai suatu teknik yang menstrukturkan/ mendekomposisi data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpangan-penyimpangan (*anomalies*) yang terjadi akibat adanya kerangkapan data dalam relasi dan in-efisiensi pengolahan

Proses normalisasi menghasilkan relasi yang optimal, yaitu martin (1975)

1. Memiliki struktur *record* yang konsisten secara logic;
2. Memiliki struktur *record* yang mudah untuk dimengerti;
3. Memiliki struktur *record* yang sederhana dalam pemeliharaan;
4. Memiliki struktur *record* yang mudah ditampilkan kembali untuk memenuhi kebutuhan pengguna;

Secara beurut-urut masing-masing level normal tersebut dibahas berikut ini, dimulai dari bentuk tidak normal. martin (1975)

1. Relasi bentuk tidak normal (*un normalized form/UNF*)

Relasi –relasi yang dirancang tanpa mengindahkan batasan dalam defenisi basis data dan karakteristik RDBM menghasilkan relasi UNF. Bentuk ini harus dihindari dalam perancangan relasi dalam basis data. Relasi UNF mempunyai kriteria sebagai berikut.

- a. Jika relasi mempunyai bentuk *non flat file* (dapat terjadi akibat data disimpan sesuai dengan kedatangan, tidak memiliki struktur tertentu, terjadi duplikasi atau tidak lengkap);
 - b. Jika relasi memuat set atribut berulang (*non single value*);
 - c. Jika relasi membuat atribut non *atomic value*.
2. Relasi bentuk normal pertama (*first norm form/1NF*)

Relasi disebut sebagai 1NF jika memenuhi kriteria sebagai berikut.

- a. Jika seluruh atribut dalam relasi bernilai atomic (*atomic value*)
- b. Jika seluruh atribut dalam relasi bernilai tunggal (*single value*)
- c. Jika relasi tidak muat set atribut berulang;
- d. Jika semua *record* mempunyai sejumlah atribut yang sama.

Permasalahan dalam 1NF adalah sebagai berikut.

- a. Tidak dapat menyisipkan informasi parsial;
- b. Terhapusnya informasi ketika menghapus sebuah *record*;
- c. Pembaruan atribut nonkunci mengakibatkan sejumlah record harus diperbaharui.

Mengubah relasi UNF menjadi bentuk 1NF dapat dijadikan dengan cara sebagai berikut.

- a. Melengkapi nilai- nilai dalam atribut;
- b. Mengubah struktur relasi.

3. Bentuk normal kedua (*second norm form/2NF*)

Relasi disebut sebagai 2NF jika memenuhi kriteria sebagai berikut.

- a. Jika memenuhi kriteria 1NF;
- b. Jika semua atribut nonkunci FD pada PK.

Permasalahan dalam 2NF adalah sebagai berikut;

- a. Kerangkapan data (*data redundancy*);
- b. Pebaruan yang tidak benar dapat menimbulkan inkonsistensi data (*data inconsistency*);
- c. Proses pembaruan data tidak efisien;
- d. Penyimpangan pada saat penyisipan, penghapusan, dan pembaruan.

Kriteria tersebut mengindikasikan bahwa di antara atribut dalam 2NF masih mungkin mengalami TDF. Selain itu relasi 2NF menuntut telah didefinisikan atribut PK dalam relasi. Mengubah relasi 1NF menjadi bentuk 2NF dapat dilakukan dengan mengubah struktur relasi dengan cara ;

- a. Identifikasikan FD relasi 1NF (jika perlu digambarkan diagram ketergantungan datanya);
- b. Berdasarkan informasi tersebut, dekomposisi relasi 1NF menjadi relasi-relasi baru sesuai FD-nya. Jika menggunakan diagram maka simpul-simpul yang berada pada puncak diagram ketergantungan data bertindak sebagai PK pada relasi baru.

4. Bentuk normal ketiga (*third norm from/3NF*)

Suatu relasi disebut sebagai 3NF jika memenuhi kriteria sebagai berikut.

- a. Jika memenuhi kriteria 2NF ;

- b. Jika setiap atribut nonkunci tidak TDF (*non transitive dependency*) terhadap PK

Permasalahan dalam 3NF adalah keberadaan penentuan yang tidak merupakan bagian dari PK menghasilkan duplikasi rinci data pada atribut yang berfungsi sebagai FK (duplikasi berbeda dengan kerangkapan data).

Mengubah relasi 2NF menjadi bentuk 3NF dapat dilakukan dengan mengubah struktur relasi dengan cara;

- a. Identifikasi TDF relasi 2NF (jika perlu digambarkan diagram ketergantungan datanya);
- b. Berdasarkan informasi tersebut dekomposisi relasi 2NF menjadi relasi-relasi baru sesuai TDF-nya jika menggunakan diagram maka simpul - simpul yang berada pada puncak diagram ketergantungan data bertindak sebagai PK pada relasi baru.

Misal;

Terhadap relasi R dengan sifat sebagai berikut.

- a. $R=(A,B,C)$ dengan $PK = A$
- b. FD: $R.B \rightarrow R.C$

Maka relasi R perlu didekomposisi menjadi relasi-relasi R1 dan R2 yaitu;

- a. $R1 = (B,C)$
- b. $R2 = (A,B)$, FK; B references R1

5. Bentuk normal *boyce-codd*(*boyce-codd norm from/BCNF*)

Bentuk normal BCNF dikemukakan oleh R.F Boyce dan E.F. Codd. Suatu relasi disebut sebagai BCNF jika memenuhi kriteria sebagai berikut.

- a. Jika memenuhi kriteria 3NF ;
 - b. Jika semua atribut penentu (determinan) merupakan CK.
6. Bentuk normal keempat (*forth norm from/4NF*)
- Relasi disebut sebagai 4NF jika memenuhi kriteria sebagai berikut.
- a. Jika memenuhi kriteria BCNF;
 - b. Jika setiap atribut didalamnya tidak mengalami ketergantungan pada banyak nilai. Atau dengan kalimat lain, bahwa semua atribut yang mengalami ketergantungan pada banyak nilai adalah bergantung secara fungsional (*functionally dependency*)
7. Bentuk normal kelima (*fifth norm from/5NF*)
- Suatu relasi memeuhi kriteria 5NF jika kerelasian antaradata dalam relasi tersebut tidak dapat direkonstruksi dari struktur relasi yang sederhana.
8. Bentuk normal kunci domain (*domain key norm from/DKNF*)
- Suatu relasi disebut sebagai DKNF jika setiap batasan dapat disimpulkan secara sederhana dengan mengetahui sekumpulan nama atribut dan domainnya selama menggunakan sekumpulan atribut pada kuncinya bentuk DKNF ini dikemukakan oleh R.Fagin pada 1981 dan bersifat sangat spesifik, artinya tidak semua relasi dapat mencapai level ini.

II.11 *Unified Modeling Language (UML)*

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. (Windu Gata : 2013)

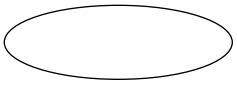
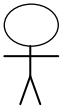
II.11.1. Diagram-Diagram UML



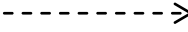
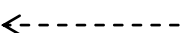
1. Diagram Use Case (Use Case Diagram)

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, yaitu :

Tabel II.1. 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.



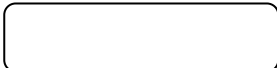
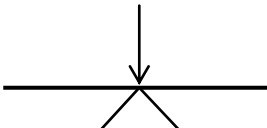
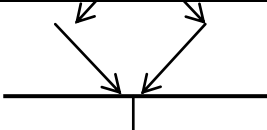
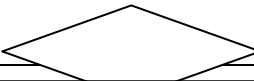
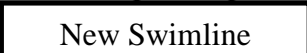
	Asosiasi antara aktor dan use case, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan use case yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi 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 : (Windu Gata : 2013)

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 Diagram Aktivitas

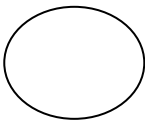
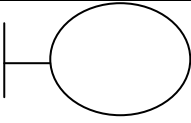
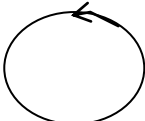
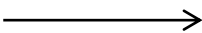
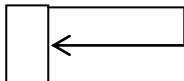
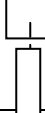
Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau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

Sumber : (Windu Gata : 2014)

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*, yaitu :

Tabel II.3. 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 : (Windu Gata : 2013)

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. 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 : (Windu Gata : 2013)