

BAB II

TINJAUAN PUSTAKA

II.1. Sistem Informasi

II.1.1. Sistem

Menurut (McLeod, 2004), “sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan.”

Sedangkan menurut (Jogiyanto, 1999), “terdapat dua kelompok pendekatan sistem di dalam mendefinisikan sistem yaitu pendekatan pada prosedur, dan pendekatan pada komponen-komponen atau elemen-elemen”

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu (Yakub ; 2012 : 1).

II.1.2. Informasi

Menurut (McLeod, 2004), “Informasi (*information*) adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya.”

Informasi juga disebut data yang diproses atau data yang memiliki arti. Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan. Para pembuat keputusan memahami bahwa informasi menjadi faktor kritis dalam menentukan kesuksesan atau kegagalan dalam suatu bidang usaha. Sistem apapun tanpa ada informasi tidak akan berguna, karena sistem tersebut akan mengalami kemacetan

dan akhirnya berhenti. Informasi dapat berupa data mentah, data tersusun, kapasitas sebuah saluran informasi, dan sebagainya (Yakub ; 2012 : 8).

II.1.3. Sistem Informasi

Menurut (O'Brian, 2005), “sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi”.

Menurut (Jogiyanto, 1999), “sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.”

Sistem informasi merupakan sebuah susunan yang terdiri dari beberapa komponen atau elemen. Komponen sistem informasi disebut dengan istilah blok bangunan (*building block*). Komponen sistem informasi tersebut terdiri dari blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*) (Yakub ; 2012 : 20).

II.2. Sistem Informasi Akuntansi

Ada beragam defenisi dari para pakar mengenai Sisem Informasi Akuntansi, diantaranya adalah :

1. Sistem Informasi Akuntansi adalah suatu komponen organisasi yang mengumpulkan, menggolongkan, mengolah, menganalisa dan komunikasikan informasi keuangan yang relevan untuk pengambilan keputusan kepada pihak-pihak luar seperti inspeksi pajak, *investor* dan *kreditur* serta pihak-pihak dalam terutama manajemen (Baridwan ; 1996 : 4)
2. Sistem Informasi Akuntansi adalah kumpulan sumber daya seperti manusia dan peralatan yang diatur untuk mengubah data menjadi informasi (Bodnard dan Hopwood ; 2000 : 23)
3. Sistem Informasi Akuntansi adalah sistem yang bertujuan untuk mengumpulkan data serta melaporkan informasi yang berkaitan dengan transaksi keuangan (Anastasia Diana dan Lilis Setiawati ; 2008 : 4).

Namun penulis juga dapat mengambil kesimpulan mengenai pengertian sendiri mengenai Sistem Informasi Akuntansi, jadi intinya Sistem informasi Akuntansi adalah sebuah sistem yang terdiri atas formulir, catatan dan laporan yang dikoordinasikan sedemikian rupa untuk menyediakan informasi keuangan yang dibutuhkan oleh manajemen guna memudahkan pengelolaan perusahaan. Beberapa unsur yang turut mendukung Sistem Informasi Akuntansi adalah formulir, catatan yang terdiri dari jurnal, buku besar dan buku pembantu serta laporan.

Lingkup Sistem Informasi Akuntansi dapat dijelaskan dari manfaat yang didapat dari informasi akuntansi. Manfaat atau tujuan Sistem Informasi Akuntansi tersebut adalah sebagai berikut :

1. Mengamankan harta/ kekayaan perusahaan

2. Menghasilkan beragam informasi untuk pengambilan keputusan
3. Menghasilkan informasi untuk pihak *eksternal*
4. Menghasilkan informasi untuk penilaian kinerja karyawan atau divisi
5. Menyediakan data masa lalu untuk kepentingan audit (pemeriksaan)
6. Menghasilkan informasi untuk penyusunan dan evaluasi anggaran perusahaan
7. Menghasilkan informasi yang diperlukan dalam kegiatan perencanaan dan pengendalian (Anastasia Diana dan Lilis Setiawati ; 2008 : 5-7).

II.2.1 Akuntansi

Akuntansi merupakan proses mengidentifikasi, mengukur, mencatat dan mengkomunikasikan peristiwa-peristiwa ekonomi dari suatu organisasi (bisnis maupun nonbisnis) kepada pihak-pihak yang berkepentingan dengan informasi bisnis tersebut (pengguna informasi). Pada dasarnya fokus utama dari akuntansi adalah transaksi bisnis (Anastasia Diana dan Lilis Setiawati : 2011 : 14).

Akuntansi sering diistilahkan sebagai bahasa bisnis, dengan kata lain bahasa yang digunakan antar pelaku bisnis untuk berkomunikasi (Johar Arifin dan Busono Adi Wicaksono ; 2006 : 9).

II.3 Piutang

Piutang usaha terjadi karena transaksi penjualan barang atau jasa secara kredit. Penjualan kredit menyebabkan adanya piutang maupun tagihan, transaksi kredit melibatkan paling sedikit dua pihak yang menjual barang atau jasa dan memperoleh piutang, dan pihak lain adalah *debitur* yaitu pihak yang melakukan

pembelian dan menjadikan hutang (Johar Arifin dan Busono Adi Wicaksono ; 2006 : 240).

Dalam prektik bisnis terdapat beberapa jenis piutang, diantaranya sebagai berikut :

1. Piutang Usaha

Piutang usaha terjadi karena transaksi penjualan barang atau jasa secara kredit pada umumnya berjangka waktu kurang dari satu tahun sehingga dalam neraca dicantumkan dalam kelompok aktiva lancar

2. Piutang Wesel

Piutang wesel terjadi karena transaksi penjualan secara kredit maupun transaksi penjualan uang, ada perbedaan antara piutang usaha dengan piutang wesel, dalam piutang wesel *debitur* membuat janji tertulis kepada *kreditur* untuk membayar sejumlah uang dalam surat tersebut untuk jangka waktu tertentu

3. Piutang Lain-lain

Piutang lain-lain merupakan tagihan perusahaan kepada pihak lain yang tidak termasuk dalam piutang usaha dan piutang karyawan. Termasuk dalam kelompok ini adalah piutang karyawan perusahaan, piutang terhadap cabang-cabang perusahaan, dan piutang kepada direksi (Johar Arifin dan Busono Adi Wicaksono ; 2006 : 241-243).

II.4 *Management Fee*

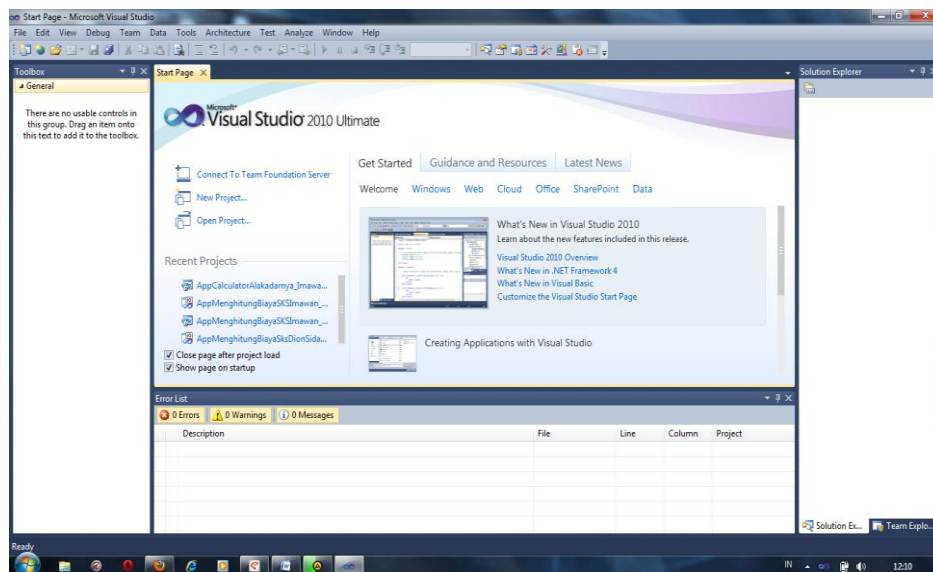
Management fee adalah merupakan uang jasa yang diterima oleh pihak PT. HRD Mandiri selaku penyedia tenaga kerja dari pihak perusahaan pengguna jasa tenaga kerja berdasarkan kesepakatan yang telah dibuat antara kedua belah pihak. Uang *management fee* ini dihitung berdasarkan jumlah karyawan dan jumlah kehadiran (absensi) dari setiap karyawan yang ditempatkan pada suatu perusahaan pengguna jasa tenaga kerja, dimana sistem pembayarannya dilakukan dalam waktu yang telah ditentukan dan disepakati bersama.

II.5 *Visual Basic.Net (Visual Basic 2010)*

Visual Basic 2010 merupakan salah satu bagian dari produk pemrograman terbaru yang dikeluarkan oleh *Microsoft*, yaitu *Microsoft Visual Studio 2010*. *Visual Studio* merupakan produk pemrograman andalan dari *Microsoft Corporation*, dimana di dalamnya berisi beberapa jenis *IDE* pemrograman seperti *Visual Basic*, *Visual C++*, *Visual Web Developer*, *Visual C#*, dan *Visual F#*. Semua *IDE* pemrograman tersebut sudah mendukung penuh implementasi *.Net Framework* terbaru, yaitu *.Net Framework 4.0* yang merupakan pengembangan dari *.Net Framework 3.5*. Adapun *database* standar yang disertakan adalah *Microsoft SQL Server 2008 express*.

Bahasa *Visual Basic 2010* sendiri awalnya berasal dari bahasa pemrograman yang sangat populer di kalangan *programmer* komputer, yaitu bahasa *BASIC*, yang oleh *Microsoft* diadaptasi dalam program *Microsoft Quick BASIC*. Seiring dengan berkembangnya teknologi komputerisasi dan desain, *Microsoft*

mengeluarkan produk yang dinamakan *Microsoft Visual Studio* dengan *Visual Basic* didalamnya. Saat ini versi *Microsoft Visual Studio* yang beredar adalah versi 10 yang populer dengan nama *Microsoft Visual Studio 2010*, yang didalamnya termasuk *Microsoft Visual Basic 2010* (Wahana Komputer ; 2011 : 2-3).



Gambar II.1. Microsoft Visual Basic 2010
Sumber : Wahana Komputer ; 2011 : 3

II.6 Basis Data

Basis data (*Database*) merupakan kumpulan data yang saling berhubungan (punya relasi). Relasi biasanya ditunjukkan dengan kunci (*key*) dari tiap *file* yang ada. Dalam satu *file* terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu *record* terdiri dari *field-field* yang saling berhubungan dan menunjukkan dalam satu pengertian yang lengkap dalam satu *record* (Yakub ; 2012 : 51-52).

Database adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (*controlled redudancy*) dengan cara tertentu sehingga mudah digunakan atau ditampilkan kembali; dapat digunakan oleh satu atau lebih program aplikasi secara optimal; data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya; data disimpan sedemikian rupa sehingga penambahan, pengambilan, dan modifikasi dapat dilakukan dengan mudah dan terkontrol (Tata Sutabri ; 2005 :161).

II.6.1 ERD (Entity Relationship Diagram)

Tentang *ERD* Ladjamudin menyatakan :

Entity Relationship Diagram (*ERD*) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.

Terdapat beberapa notasi untuk model data yang digunakan:

1. Entitas

Entitas adalah kelompok orang, tempat obyek kejadian atau konsep tentang apa yang kita perlukan untuk men-*capture* dan menyimpan data.

2. Atribut

Atribut adalah sifat atau karakteristik deskriptif suatu entitas. Penentuan atau pemilihan atribut-atribut yang relevan bagi sebuah entitas merupakan hal yang penting dalam pembentukan model data.

3. Relasi

Relasi adalah hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

4. Kardinalitas

Kardinalitas relasi menunjukkan jumlah maksimum tupel yang dapat berelasi dengan entitas pada entitas yang lain. Dari sejumlah kemungkinan banyaknya hubungan antar entitas tersebut, kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari entitas yang satu ke entitas yang lain dan begitu juga sebaliknya. Terdapat 3 macam kardinalitas relasi yaitu (Ladjamudin, 2005) :

- a. *One to one*
- b. *One to many* atau *many to one*
- c. *Many to many*

II.6.2 Kamus Data

Mengenai kamus data menurut Jogiyanto (2005) :

Kamus data adalah katalog fakta tentang data dan kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. kamus data dapat juga digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. (Jogiyanto, 2005)

Kamus data harus memuat hal-hal berikut ini (Jogiyanto, 2005) :

1. Nama arus data
2. Alias
3. Bentuk data
4. Arus data

5. Penjelasan
6. Periode
7. Volume
8. Struktur data

II.6.3 Normalisasi

Normalisasi adalah suatu teknik yang menstrukturkan data dalam cara tertentu untuk membantu mengurangi atau mencegah timbulnya masalah yang berhubungan dengan pengolahan data dalam *database*. Proses normalisasi menghasilkan struktur *record* yang konsisten secara logis yang mudah dimengerti dan sederhana dalam pemeliharaannya. Beberapa *level* normalisasi dapat dijelaskan dan kriteria yang mendefinisikan *level* pada normalisasi adalah bentuk normal (*norm form*). Proses normalisasi merupakan proses pengelompokan data elemen menjadi tabel yang menunjukkan *entity* dan relasinya.

II.7 SQL Server

SQL Server adalah sebuah *DBMS (Database Management System)* yang dibuat oleh *Microsoft* untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti *IBM* dan *Oracle*. *SQL Server 2008* dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa *SQL Server 2008* membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data (Wahana Komputer ; 2010 : 2).

Menurut cara pemrosesan data pada *processor* maka *Microsoft* mengelompokkan produk ini berdasarkan 2 jenis yaitu :

1. Versi 32-bit(x86), yang biasanya digunakan untuk komputer dengan *single processor* (Pentium IV) atau lebih tepatnya *processor* 32 bit dan sistem operasi *Windows XP*
2. Versi 64-bit(x64), yang biasanya digunakan untuk komputer dengan lebih dari satu *processor* (Misalnya: *Core 2 Duo*) dan sistem operasi 64 bit seperti *Windows XP 64, Vista, dan Windows 7* (Wahana Komputer ; 2010 : 3).

II.8 UML

UML singkatan dari *Unified Modelling Language* yang berarti bahasa pemodelan standar. (Chonoles, 2003 : bab 1) mengatakan sebagai bahasa, berarti *UML* memiliki sintaks dan semantik. Ketika kita membuat model menggunakan konsep *UML* ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan lainnya harus memiliki standart yang ada. *UML* bukan hanya sekedar diagram, tetapi juga menceritakan konteksnya. Ketika pelanggan memesan sesuatu dari sistem, bagaimana transaksinya? Bagaimana sistem mengatasi *error* yang terjadi? Bagaimana keamanan terhadap sistem yang kita buat? Dan sebagainya dapat dijawab dengan *UML* (Prabowo Pudjo Widodo dan Herlawati : 2011 : 6).

UML diaplikasikan dengan maksud tertentu, biasanya antara lain untuk :

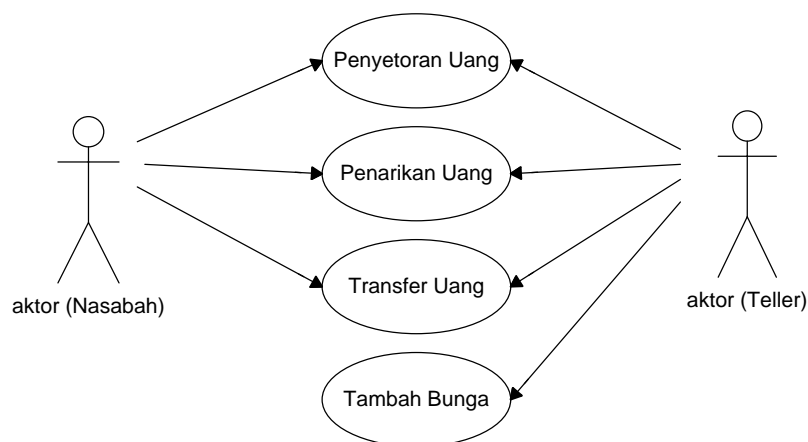
1. Merancang perangkat lunak
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis

3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

II.8.1 Use Case Diagram

Diagram *Use case* bersifat statis. Diagram ini memperlihatkan himpunan use-case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna (Prabowo Pudjo Widodo dan Herlawati : 2011 : 10).

Gambar di bawah ini merupakan salah satu contoh bentuk diagram *use case* :



Gambar II.2. Contoh Diagram Use Case

Sumber : Prabowo Pudjo Widodo dan Herlawati : 2011 : 17

Diagram *use case* bersama dengan narasi *use case* dan skenario mendefinisikan tujuan suatu sistem atau pengklasifikasi lain seperti *enterprise*, subsistem atau komponen. Konsep ini diperkenalkan oleh Ivar Jacobson bersama organisasinya dalam bentuk metodologi yang mereka namakan *Object-Oriented*

Software Engineering (OOSE). Tujuan dibentuknya metode ini adalah agar dihasilkan fokus yang baik pada pengembangan dan tujuan utama tanpa terpengaruh oleh implementasi praktis.

Elemen *use case* terdiri dari :

1. Diagram *use case*, disertai dengan narasi dan skenario
2. Aktor (*actor*), mendefinisikan entitas di luar sistem yang memakai sistem
3. Asosiasi (*assosiation*), mengindikasikan aktor mana yang berinteraksi dengan *use case* dalam suatu sistem
4. <<*include*>> dan <<*extend*>>, merupakan indikator yang menggambarkan jenis relasi dan interaksi antar *use case*
5. Generalisasi (*generalization*), menggambarkan hubungan turunan antara *use case* atau antar aktor.

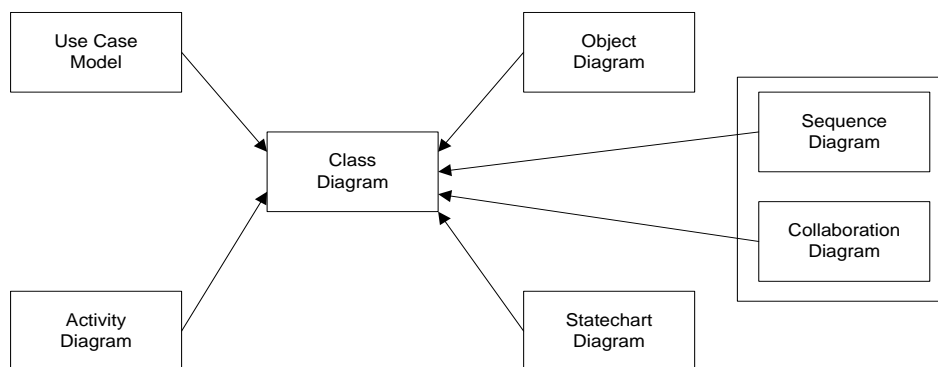
Use case mengekspresikan apa yang *user* harapkan terhadap sistem. Narasi *use case* menjelaskan secara detail bagaimana *user* berinteraksi dengan sistem saat mengakses *use case*. Skenario memecah penjelasan narasi untuk menyediakan penjelasan detail terhadap segala kemungkinan yang terjadi pada *use case*, apa yang terjadi dan apa respon sistem (Prabowo Pudjo Widodo dan Herlawati : 2011 : 34-36).

II.8.2 Class Diagram

Diagram kelas (*class diagram*) bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek.

Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif (Prabowo Pudjo Widodo dan Herlawati : 2011 : 10).

Kemampuan menghasilkan kode program yang dimiliki diagram kelas menyebabkan diagram ini memiliki hubungan yang khas dengan diagram UML lainnya. Pada gambar II.2 terlihat bahwa diagram yang lain memberi masukan kepada diagram kelas. Diagram kelas yang baik menghasilkan suatu rancangan sistem atau program yang mendekati kenyataan.



Gambar II.3. Hubungan diagram kelas dengan diagram UML Lainnya

Sumber : Prabowo Pudjo Widodo dan Herlawati : 2011 : 38

Kelas digambarkan dengan kotak yang terdiri dari sekat-sekat berturut-turut dari atas ke bawah untuk nama, atribut dan operasi. Dalam satu paket, nama kelas harus unik (tidak boleh ada kesamaan nama). Bila namanya sama tapi beda paket dengan kelas lain, harus disebutkan dalam format nama paket :: nama kelas (Prabowo Pudjo Widodo dan Herlawati : 2011 : 76).

II.8.3. Activity Diagram

Diagram Aktivitas (*Activity diagram*) bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek (Prabowo Pudjo Widodo dan Herlawati : 2011 : 11).

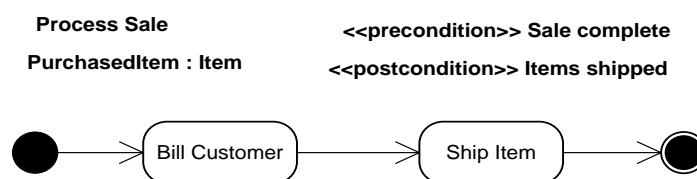
Diagram aktivitas lebih memfokuskan diri pada eksekusi dan alur sistem dari pada bagaimana sistem itu dirakit. Diagram ini tidak hanya memodelkan *software* melainkan memodelkan model bisnis juga. Diagram aktivitas menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi. Ketika digunakan dalam pemodelan *software*, diagram aktivitas merepresentasikan pemanggilan suatu fungsi tertentu misalnya *call*. Sedangkan bila digunakan dalam pemodelan bisnis, diagram ini menggambarkan aktivitas yang dipicu oleh kejadian-kejadian di luar seperti pemesanan atau kejadian-kejadian internal misalnya proses penggajian tiap jumat sore.

Aktivitas merupakan kumpulan aksi-aksi. Aksi-aksi melakukan langkah sekali saja tidak boleh dipecah menjadi beberapa langkah lagi. Contoh aksi yaitu :

- a. Fungsi matematika
- b. Pemanggilan perilaku
- c. Pemrosesan data

Ketika kita menggunakan diagram aktivitas untuk memodelkan perilaku suatu *classifier*, *classifier* dikatakan kontek dari aktivitas. Aktivitas dapat mengakses atribut dan operasi *classifier*, tiap objek yang terhubung dan parameter-parameter

jika aktivitas memiliki hubungan dengan perilaku. Ketika digunakan untuk model proses bisnis, informasi itu biasanya disebut *process-relevant* data. Aktivitas diharapkan dapat digunakan ulang dalam suatu aplikasi, sedangkan aksi biasanya *specific* dan digunakan hanya untuk aktivitas tertentu (Prabowo Pudjo Widodo dan Herlawati : 2011 : 143-145).



Gambar II.4. Diagram aktivitas dengan kondisi mula dan akhir
Sumber : Prabowo Pudjo Widodo dan Herlawati : 2011 : 147

II.8.4 *Sequence Diagram*

Diagram urutan (*Sequence Diagram*) adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu (Prabowo Pudjo Widodo dan Herlawati : 2011 : 11).

Sequence Diagram digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian untuk menghasilkan *output* tertentu. *Sequence Diagram* diawali dari apa yang memicu aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara *internal* dan *output* apa yang dihasilkan.