

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1.Sistem**

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau tujuan tertentu (Feto Daan Yos;2009:2-3).

Ada beberapa elemen yang membentuk sebuah sistem yaitu tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik serta lingkungan.

##### **1. Tujuan**

Tujuan ini menjadi motivasi yang mengarahkan pada sistem, karena tanpa tujuan yang jelas sistem menjadi tak terarah dan tak terkendali.

##### **2. Masukan (*input*) sistem**

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa hal-hal berwujud maupun yang tidak berwujud. Masukan berwujud adalah bahan mentah, sedangkan yang tidak berwujud adalah informasi.

##### **3. Proses**

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai.

#### 4. Keluaran (*output*)

Keluaran (*output*) merupakan hasil dari pemrosesan sistem dan keluaran dapat menjadi masukan untuk subsistem lain.

#### 5. Batas (*boundary*) sistem

Batas (*boundary*) sistem adalah pemisah antara sistem dan daerah diluar sistem. Batas sistem menentukan konfigurasi, ruang lingkup atau kemampuan sistem

#### 6. Mekanisme pengendalian dan umpan balik

Mekanisme pengendalian diwujudkan dengan menggunakan umpan balik (*feedback*), sedangkan umpan balik ini digunakan untuk mengendalikan masukan maupun proses. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan.

#### 7. Lingkungan

Lingkungan adalah segala sesuatu yang berada diluar sistem. (Yakub;2012:3-4)

## **II.2. Sistem Informasi**

Sistem Informasi dapat dipahami sebagai sekumpulan sub sistem yang saling berhubungan, berkumpul bersama-sama dan membentuk satu kesatuan, saling berinteraksi dan bekerja sama antara bagian satu dengan yang lainnya dengan cara-cara tertentu untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi sebagai dasar bagi pengambilan keputusan yang berguna dan mempunyai nilai nyata yang dapat

dirasakan akibatnya baik pada saat itu juga maupun dimasa mendatang, mendukung kegiatan operasional, manajerial, dan strategis organisasi, dengan memanfaatkan berbagai sumber daya yang ada dan tersedia bagi fungsi tersebut guna mencapai tujuan (Edhy Sutanta;2003 : 19)

### **II.3. Sistem Informasi Akuntansi (SIA)**

Sistem Informasi Akuntansi adalah sistem yang bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan transaksi keuangan. (Putu Mega Selvy Aviana;2012:66).

SIA adalah kumpulan kegiatan-kegiatan dari organisasi yang bertanggung jawab untuk menyediakan informasi keuangan dan informasi yang didapatkan dari transaksi data untuk tujuan pelaporan internal kepada manajer untuk digunakan dalam pengendalian dan perencanaan sekarang dan operasi masa depan serta pelaporan eksternal kepada pemegang saham, pemerintah, dan pihak-pihak luar lainnya. Dari defenisi tersebut dapat diambil kesimpulan bahwa SIA merupakan suatu kegiatan *input*, proses, dan *output* data yang dilakukan oleh perusahaan. Hasil data akhir yang telah di proses SIA bertujuan sebagai pelaporan bagi pihak internal dan eksternal guna melakukan pengendalian terhadap perusahaan tersebut. (Putu Mega Selvy Aviana;2012:66).

SIA (Sistem Informasi Akuntansi) pada umumnya memiliki manfaat dan tujuan sebagai berikut :

1. Mengamankan harta ataupun kekayaan perusahaan. Harta dan kekayaan disini meliputi kas perusahaan, persediaan barang dagangan, termasuk aset tetap 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 atau pemeriksaan
6. Menghasilkan informasi untuk penyusunan dan evaluasi anggaran perusahaan
7. menghasilkan informasi yang diperlukan dalam kegiatan perencanaan dan pengendalian.(Putu Mega Selvy Aviana;2012:66).

#### **II.4. Pajak**

Pajak adalah kewajiban yang melekat kepada setiap warga yang memenuhi syarat yang telah ditetapkan oleh Undang-Undang agar membayar sejumlah uang ke Kas Negara yang bersifat memaksa, dan tidak mendapatkan imbalan secara langsung. Soemirto dikutip oleh Yolina (2009:11), Pajak adalah iuran rakyat kepada Kas Negara berdasarkan Undang-Undang (dapat dipaksakan) dengan tidak mendapat jasa timbal (kontraprestasi) yang langsung dapat ditunjukkan dan yang digunakan untuk membayar pengeluaran umum. Djajadiningratdikutip oleh Resmi (2009: 1), Pajak sebagai suatu kewajiban menyerahkan sebagian kekayaan ke Kas Negara yang disebabkan suatu keadaan, kejadian, dan perbuatan yang memberikan kedudukan tertentu, tetapi bukan sebagai hukuman, menurut peraturan yang ditetapkan pemerintah serta dapat dipaksakan, tetapi tidak ada jasa timbal balik dari Negara secara langsung, untuk

memelihara kesejahteraan secara umum. Dari definisi-definisi pajak diatas dapat disimpulkan bahwa pajak memiliki peranan yang sangat penting dalam penerimaan Negara. Dengan semakin meningkatnya pengetahuan, pemahaman, dan kesadaran Wajib Pajak tentang pajak sangat mendukung kemandirian dalam memenuhi kebutuhan dana untuk kepentingan penyelenggaraan Negara, sehingga pajak memegang peran penting bagi penerimaan Negara. Pajak mempunyai peranan yang sangat penting dalam kehidupan bernegara, khususnya di dalam pelaksanaan pembangunan karena pajak merupakan sumber pendapatan negara untuk membiayai pembangunan (Tryana A.M. Tiraada, 2013).

#### **II.4.1 Fungsi Pajak**

Mardiasmo (2011: 1) mengungkapkan bahwa pajak memiliki fungsi sebagai berikut :

##### **1. Fungsi Anggaran**

Untuk menjalankan tugas-tugas rutin negara dan melaksanakan pembangunan, negara membutuhkan biaya. Biaya ini dapat diperoleh dari penerimaan pajak. Pajak digunakan untuk pembiayaan rutin seperti belanja pegawai, belanja barang, pemeliharaan dan lain sebagainya.

##### **2. Fungsi Mengatur**

Pemerintah bisa mengatur pertumbuhan ekonomi melalui kebijaksanaan pajak. Dengan fungsi mengatur, pajak bisa digunakan sebagai alat untuk mencapai tujuan pemerintah untuk kemajuan negara.

### 3. Fungsi stabilitas

Dengan adanya pajak, pemerintah memiliki dana untuk menjalankan kebijakan yang berhubungan dengan stabilitas harga sehingga inflasi dapat dikendalikan. Hal ini bisa dilakukan antara lain dengan jalan mengatur peredaran uang di masyarakat, pemungutan pajak, penggunaan pajak yang efektif dan efisien (Tryana A.M. Tiraada, 2013).

#### **II.4.2 PPh Pasal 22**

Pajak penghasilan pasal 22 merupakan angsuran atau kredit PPh. Kredit PPh merupakan pembayaran di muka PPh yang akan diperhitungkan atau di kreditkan terhadap PPh terutang pada akhir tahun pajak. Kredit pajak PPh pasal 22 merupakan aktiva lancar dengan nama akun pajak dibayar dimuka. (Suprajadi;2001:7)

Objek PPh pasal 22 meliputi transaksi atas impor barang, penyerahan ke pemerintah atau BUMN/BUMD atas penjualan hasil produksi dalam negeri seperti rokok, otomotif, kertas, baja, semen, transaksi pembelian premium, premix, solar, pelumas, minyak tanah, gas elpiji dari Pertamina oleh penyalur/agen/grosir/dealer, kepada pembeli lainnya (misal: pabrikan), terakhir adalah transaksi pembelian dari BULOG atas gula pasir, tepung terigu yang dilakukan oleh penyalur/grosir/pembeli lainnya. Mengenai tarif PPh pasal 22 bervariasi sesuai dengan yang telah ditentukan. (Suprajadi;2001:7)

Rumus pajak yang digunakan adalah  $\text{PPh} = 0.25\% \times \text{NJOP}$  (Nilai Jual Objek Pajak), sesuai yang ditetapkan oleh Menteri Keuangan mengenai wajib pajak atas penjualan hasil produksi berbahan dasar tanah atau semen.

## II.5. Metode Gross UP

Metode gross-up memberikan tunjangan pajak sebesar pajak penghasilan yang dipotong atas gaji karyawan. Untuk menentukan besarnya tunjangan ada formula rumus yang digunakan agar tetap sesuai yang dimaksud dalam aturan perpajakan yang berlaku. Oleh karena itu, dasar perhitungan rumus metode *gross-up* adalah penghasilan kena pajak (PKP) sebelum ada tunjangan.

Penggunaan metode ini berkaitan dengan penetapan besarnya tunjangan pajak. Dengan penerapan ini, besarnya tunjangan akan ditambahkan kedalam penghasilan karyawan (taxable). Bagi perusahaan, beban yang timbul atas pemberian tunjangan merupakan biaya yang dapat dikurangkan atau bersifat *deductable expenses* dapat mengefisienkan beban pajak perusahaan (Partiwi; 2014:7).

Formula untuk menghitung besarnya tunjangan PPh Pasal 22 dalam Gross-Up Method dibagi kedalam empat lapisan rentang PKP (penghasilan kena pajak) :

- a.  $\text{PKP s/d Rp. 47.500.000} : \text{PKP setahun} \times 5 \div 95,25$
- b.  $\text{PKP} > \text{Rp 47.500.000 s/d Rp.217.500.000} :$   
 $(\text{PKP setahun} - 47.500.000) \times 15 \div 85 + 2.500.000$
- c.  $\text{PKP} > \text{Rp 217.500.000 s/d Rp 405.000.000} :$   
 $(\text{PKP setahun} - 217.500.000) \times 25 \div 75 + 32.500.000$

- d.  $PKP > Rp\ 405.000.000 : (PKP\ setahun - 405.000.000) \times 30 \div 70 + 95.000.000.$  (Partiwi; 2014:7).

## II.6. PHP

PHP (dahulu dikenal sebagai *Personal Home Page*, sekarang PHP: *Hypertext Preprocessor*) yang merupakan program yang dikembangkan secara bersama oleh para programmer dari seluruh dunia yang menekuni dunia opensource. PHP dikembangkan khususnya untuk mengakses dan memanipulasi data yang ada di database server open source seperti MySQL (Viviliana, 2010).

## II.7. MySQL

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user, serta menggunakan perintah standar SQL (*Structured Query Language*) (Viviliana, 2010).

## II.8. Normalisasi

Agar model *database* relasional bisa digunakan secara efektif, maka pengelompokan yang rumit harus disederhanakan sehingga data berlebihan dan relasi yang salah bisa dieliminasi. Proses membuat struktur data yang lebih kecil dan stabil dari sekelompok data yang rumit disebut Normalisasi (Paillin; 2012).

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data / *database*, teknik pengelompokan atribut dari suatu relasi sehingga

membentuk struktur relasi yang baik tanpa redundansi (Paillin,;2012). Tujuan normalisasi adalah mengorganisasikan data kedalam tabel-tabel untuk memenuhi kebutuhan pemakai, menghilangkan kerangkapan data, mengurangi kompleksitas, mempermudah modifikasi data.

#### Proses Normalisasi

- Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu kebeberapa tingkat.
- Apabila tabel yang diuji belum memenuhi persyaratan tertentu maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

#### Tahapan Normalisasi :

1. Bentuk tidak normal : Menghilangkan perulangan grup.

**Tabelll.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 DB	B104	Ati	C
			Akn201	Akuntansi	D310	Lia	B
				Keuangan			
			MKT300	Dasar Pemasaran	B212	Lola	A

( Sumber : J. I. Maanari: 2013)

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.

**TabelII.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.	M135	Manajemen DB	B104	Ati	C
5432	Bakti	Ak.	Akn201	Akuntansi	D310	Lia	B
5432	Bakti	Ak.		Keuangan			
5432	Bakti	Ak.	MKT300	Dasar Pemasaran	B212	Lola	A

( Sumber : J. I. Maanari: 2013)

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.

**TabelIII.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 DB	B104	Ati
Akn201	Akuntansi	D310	Lia
	Keuangan		

MKT300	Dasar Pemasaran	B212	Lola
--------	-----------------	------	------

( Sumber : J. I. Maanari: 2013)

4. Bentuk Normal ketiga (3NF) : Menghilangkan anomali-anomali hasil dari ketergantunganfungsional.

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.

**Tabell.4 : Contoh bentuk normal ketiga (3NF)**

No-Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	Ak.

( Sumber : J. I. Maanari: 2013)

## II.9. *Unified Modelling Language (UML)*

Menurut Yuni Sugiarti (2013:34) *Unified Modelling Language (UML)* adalah sebuah bahasa yang menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak.

UML biasa digunakan untuk :

1. Menggambarkan batasan sistem dan fungsi-fungsi sistem secara umum, dibuat dengan *user* dan *actor*
2. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interactions diagrams*

3. Menggambarkan representasi struktur statik sebuah sistem dalam bentuk *class diagram*
4. Membuat model behavior "yang menggambarkan kebiasaan atau sifat sebuah sistem" dengan *state transition diagrams*.

*UML* merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan system berorientasi objek karena *UML* menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku

*UML* berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui sejumlah elemen grafis yang bias dikombinasikan menjadi diagram. *UML* mempunyai banyak diagram yang dapat mengakomodasi berbagai sudut pandang dari suatu perangkat lunak yang akan dibangun.

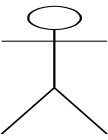


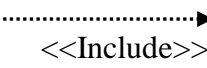
## **II.9.1. Diagram Dasar dalam UML**

### **1. Use Case Diagram**

*Use Case Diagram* atau diagram use case merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat. Diagram use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan pengertian yang cepat, diagram use case digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Terdapat beberapa simbol dalam menggambarkan diagram use case, yaitu use case, aktor dan relasi. Hal yang perlu diingat mengenai diagram use case adalah diagram use bukan menggambarkan

tampilan antar muka (*user interface*), arsitektur dari sistem, kebutuhan nonfungsional, dan tujuan performansi. Sedangkan untuk penamaan *use case* adalah nama didefinisikan sesimple mungkin, dapat dipahami dan menggunakan kata kerja (Yuni Sugiarti : 2013 : 41).Berikut adalah simbol-simbol yang ada pada diagram use case :

**Tabel II.5 : Simbol-simbol pada *use case diagrams***

Simbol	Deskripsi
Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case
 Actor	Orang, proses, atau sistem lain yang berinteraksi sebagai dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
 Asosiasi / Association	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor
 Extend	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu, miri dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, arah panah menunjuk use case yang dituju.
 Include	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini. Ada dua sudut pandang yang cukup besar mengenai include di use case, include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan

( Sumber: Yuni Sugiarti ;2013 : 42)

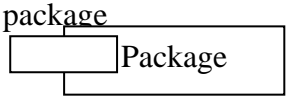
## 2. Class Diagram


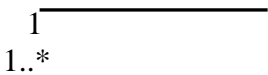



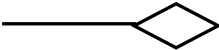
Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, atribut mendeskripsikan properti dengan sebaris teks didalam kotak kelas tersebut. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas mendeskripsikan jenis-jenis objek dalam sistem dan berbagai hubungan statis yang terdapat diantara mereka. Diagram juga menunjukkan properti dan operasi sebuah kelas dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.

Diagram kelas menggambarkan struktur dan deskripsi kelas, package, dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Kelas memiliki tiga area pokok : nama, atribut dan operasi (Yuni Sugiarti : 2013 : 57). Berikut adalah simbol-simbol yang ada pada diagram kelas :

**Tabel II.6 : Simbol-simbol *class diagram***

Simbol	Deskripsi
	Package merupakan sebuah bungkusan dari satu atau lebih kelas

<p>Operasi</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <div style="border: 1px solid black; padding: 2px; width: 80%; margin: 0 auto;">Nama kelas</div> <div style="border: 1px solid black; padding: 2px; width: 80%; margin: 2px auto;">'+Attribut1 '+Attribut1</div> <div style="border: 1px solid black; padding: 2px; width: 80%; margin: 2px auto;">+Operatuin 1()</div> </div>	Kelas pada struktur system
<p>Antarmuka/interface</p> 	Sama dengan konsep interface dalam pemrograman berorientasi objek
<p>Asosiasi / Association</p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Aosiasi berarah/directed asosiasi</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi spesialisasi (umum khusus)
<p>Kebergantungan/Defedency</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas
<p>Agregasi</p> 	Relasi antar kelas dengan makna semua-bagian (whole-part)

( Sumber:Yuni Sugiarti ; 2013 : 59)

### 3. *Sequence Diagram*

*Diagram Sequence* menggambarkan kelakuan/perilaku objek pada use case dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuences maka harus diketahui objek-objek yang terlibat dalam sebuah case beserta metode-metode yang dimiliki kelas yang di instansiasi menjadi objek itu.

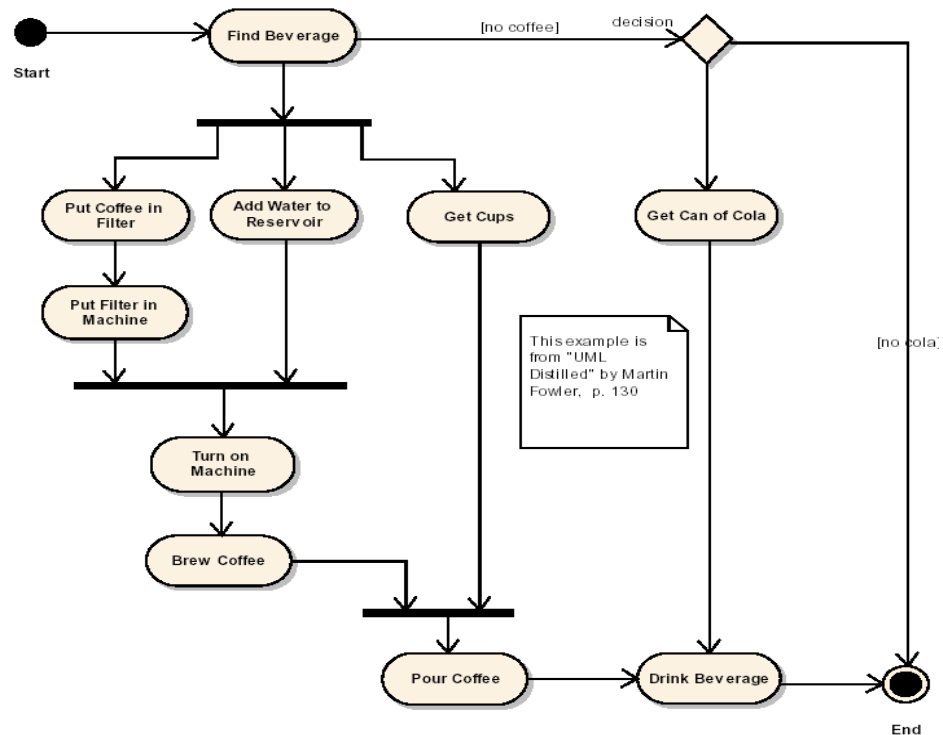
Diagram sekuence memiliki ciri yang berbeda dengan diagram interaksi pada diagram kolaborasi sebagai berikut :

1. Pada diagram sekuence terdapat garis objek.
2. Terdapat pokus kendali (*Focus of Control*) , berupa empat persegi panjang ramping dan tinggi yang menampilkan aksi atau objek secara langsung atau sepanjang sub ordinat (Yuni Sugiarti : 2013 : 69-70).

#### **4. Activity Diagram**

Digram aktivitas atau *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukakn oleh sistem. Diagram aktivitas mendukung perilaku paralel.

*Activity diagrams* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi (Yuni Sugiarti : 2013 : 75).



**Gambar.II.1 : Activity Diagram Tanpa Swimlane**  
(Sumber: Yuni Sugiarti;2013 : 76)

## II.10. Database

*Database* atau basis data adalah sekumpulan data yang memiliki hubungan secara logika dan diatur dengan susunan tertentu serta disimpan dalam media penyimpanan computer. Data itu sendiri adalah representasi dari semua fakta yang ada pada dunia nyata. *Database* sering digunakan untuk menghasilkan informasi tertentu. Misalnya dari data nama siswa dan tanggal lahir siswa Anda bias mendapatkan informasi nama siswa yang berulang tahun pada hari ini. Tentu saja informasi tersebut akan Anda dapatkan dari *software* pemroses *database* dengan cara Anda memberikan perintah dalam bahasa tertentu yaitu SQL (*Structured Query Language*). Paillin, 2012.