

## **BAB II**

### **TINJAUAN PUSTAKA**

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

##### **II.1.1. Sistem**

Sistem adalah prosedur logis dan rasional untuk merancang suatu rangkaian komponen yang berhubungan satu dengan yang lainnya dengan maksud untuk berfungsi sebagai suatu kesatuan dalam usaha mencapai suatu tujuan yang telah ditentukan. (*Ika Nur Indah ; 2013 : 125*)

Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen. Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu. (*Yeremia Yuliawan ; 2013 : 85*)

##### **II.1.2. Informasi**

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. (*Ika Nur Indah ; 2013 : 125*)

Pengertian Informasi menurut Rizan Machmud (2013 : 410) adalah sebagai berikut : “Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berguna bagi pengguna yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi”.

##### **II.1.3. Sistem Informasi**

Menurut Laudon, K. C dan Laudon, J.P didalam buku yang berjudul “Pengenalan Sistem Informasi” (Kadir, A, 1992). Sistem informasi adalah

kombinasi antara prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. Sistem informasi dapat didefinisikan sebagai sekumpulan komponen yang saling berhubungan, mengumpulkan, memproses, menyimpan dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi. Sistem informasi menerima masukan data dan instruksi, mengolah data tersebut sesuai instruksi, dan mengeluarkan hasilnya. Untuk penerapan pengolahan informasi dapat dianalisis menjadi masukan, penyimpanan, pengolahan dan keluaran. (*Dessi Tri Santi ; 2014 : 7-8*)

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Namun, secara sederhana sistem informasi dapat didefinisikan sebagai suatu sistem yang bertujuan untuk menyimpan, memproses dan mengomunikasikan informasi. (*Purnomo Budi Santoso ; 2013 ; 61*)

## **II.2. Penjualan**

Penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencana-rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba. Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga dapat mengetahui hasil produk yang

dihasikan. Penjualan adalah suatu transfer hak atas benda-benda. Dari penjelasan tersebut dalam memindahkan atau mentransfer barang dan jasa diperlukan orang-orang yang bekerja dibidang penjualan seperti pelaksanaan dagang, agen, wakil pelayanan dan wakil pemasaran. (*Dessi Tri Santi ; 2014 : 8*)

Penjualan adalah suatu sistem keseluruhan dari kegiatan usaha yang ditujukan untuk merencanakan, menentukan harga, mempromosikan dan mendistribusikan barang, jasa, ide kepada pasar sasaran agar dapat mencapai tujuan organisasi. Menurut Marwan (1991), penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencanarencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba. (*Rara Sri Artati Rejeki ; 2011 : 152*)

Menurut Siegel. Joel G dan Shim. Joe K (2009:67) dalam bukunya Moh. Kurdi yang berjudul “kamus Istilah Akuntansi” Penjualan adalah Penerimaan yang diperoleh dari pengiriman barang dagangan atau dari penyerahan pelayanan dalam bursa sebagai barang pertimbangan. Pertimbangan ini dapat dalam bentuk tunai peralatan kas atau harta lainnya. Pendapatan dapat diperoleh pada saat penjualan, karena terjadi pertukaran, harga jual dapat ditetapkan dan bebannya diketahui. (*Dessi Tri Santi ; 2014 : 8*)

### **II.3. Sistem Informasi Akuntansi**

Sistem informasi akuntansi adalah sistem yang mengumpulkan dan memproses transaksi-transaksi data dan menyampaikan informasi keuangan kepada pihak-pihak tertentu. Sistem informasi akuntansi adalah sistem berbasis komputer yang dirancang untuk mentransformasi data akuntansi menjadi

informasi. (*Merystika Kabuhung ; 2013 : 340*)

Sistem Informasi Akuntansi (SIA) adalah sumber daya manusia dan modal dalam organisasi yang bertanggung jawab untuk persiapan informasi keuangan dan informasi yang diperoleh dari mengumpulkan dan memproses berbagai transaksi perusahaan. Dalam hubungan dengan sistem informasi akuntansi, memberikan pernyataan tentang transaksi keuangan dan transaksi non keuangan. Transaksi keuangan adalah sebuah peristiwa ekonomi yang mempengaruhi aktiva dan ekuitas suatu organisasi, direfleksikan dalam akun-akunnya, dan diukur dalam satuan moneter. Transaksi non-keuangan: termasuk dalam semua peristiwa yang diproses oleh sistem informasi organisasi yang tidak memenuhi defenisi sempit dari transaksi keuangan. Sistem informasi akuntansi tidak hanya mengolah data keuangan saja, data non keuangan juga diikut sertakan karena dapat pengambilan keputusan tidak hanya informasi keuangan saja yang diperlukan, informasi non keuangan tentang suatu kondisi dan keadaan juga dapat dipergunakan sebagai pertimbangan dalam pengambilan keputusan. (*Vendy Michael Kandouw ; 2013 : 435*)

### **II.3.1. Komponen-Komponen Sistem Informasi Akuntansi**

Sistem informasi akuntansi terdiri dari lima komponen, yaitu sebagai berikut (*Merystika Kabuhung ; 2013 : 340-341*).

1. Orang-orang yang mengoperasikan sistem tersebut dan melaksanakan berbagai fungsi.
2. Prosedur-prosedur, baik manual maupun yang terotomatisasi, yang dilibatkan dalam mengumpulkan, memproses, dan menyimpan data tentang

aktivitas-aktivitas organisasi.

3. Data tentang proses-proses bisnis organisasi.
4. Software yang dipakai untuk memproses data organisasi.
5. Infrastruktur teknologi informasi, termasuk komputer, peralatan pendukung (peripheral device), dan peralatan untuk komunikasi jaringan.

### **II.3.2. Unsur-Unsur Sistem Informasi Akuntansi**

Unsur-unsur sistem informasi akuntansi menurut (*Merystika Kabuhung ; 2013 : 341*) adalah sebagai berikut.

#### **1. Sumber Daya manusia**

Sistem informasi akuntansi membutuhkan sumber daya untuk dapat berfungsi. Sumber daya dapat diklasifikasikan sebagai alat, data, bahan pendukung, sumber daya manusia dan dana.

#### **2. Peralatan**

Peralatan merupakan unsur sistem informasi akuntansi yang berperan dalam mempercepat pengolahan data, meningkatkan ketelitian kalkulasi atau perhitungan dan kerapihan bentuk informasi.

#### **3. Formulir**

Formulir merupakan unsur pokok yang digunakan untuk mencatat semua transaksi yang terjadi. Formulir sering disebut dengan istilah dokumen.

#### **4. Catatan**

Catatan terdiri dari beberapa bagian, yaitu sebagai berikut.

##### **a. Jurnal**

Merupakan catatan akuntansi yang pertama digunakan untuk mencatat,

mengklasifikasi dan meringkas data keuangan dan data yang lainnya.

b. Buku besar

Terdiri dari rekening-rekening yang digunakan untuk meringkas data keuangan yang telah dicatat sebelumnya kedalam jurnal.

5. Prosedur

Prosedur merupakan urutan atau langkah-langkah untuk menjalankan suatu pekerjaan, tugas atau kegiatan.

6. Laporan

Hasil akhir dari system informasi akuntansi adalah laporan keuangan dan laporan manajemen.

#### **II.4. Prinsip-Prinsip Sistem Informasi Akuntansi**

Sistem informasi akuntansi yang efektif dan efisien didasarkan pada beberapa prinsip dasar. Prinsip prinsip dasar tersebut yaitu sebagai berikut (*Merystika Kabuhung ; 2013 : 341*) :

1. Keefektifan biaya. Sistem informasi akuntansi harus efektif biaya.
2. Tingkat kegunaan. Agar berguna, informasi harus dapat dimengerti, relevan, dapat diandalkan, tepat waktu, dan akurat.
3. Fleksibilitas. Sistem harus cukup fleksibel dalam memenuhi perubahan permintaan informasi yang dibutuhkan.

#### **II.5. *Macromedia Dreamweaver 8***

*Macromedia Dreamweaver 8*, atau yang biasa disebut “*Dreamweaver 8*”, adalah sebuah perangkat lunak aplikasi untuk mendesain dan membuat halaman

*web*, anda tidak perlu lagi mengetik kode – kode *HTML* atau kode – kode lainnya. Selain *HTML*, *Dreamweaver 8* juga mendukung *CSS*, *JavaScript*, *PHP*, *ASP*, dan bahasa pemrograman lainnya untuk membuat *web*. *Dreamweaver 8* adalah versi terbaru dari keluarga *Dreamweaver*. Versi pertamanya sendiri diluncurkan sekitar tahun 1994 oleh Macromedia Inc. Dalam versi terbaru ini, banyak sekali fasilitas baru yang ditambahkan. *Dreamweaver 8* juga menyediakan beberapa *template* halaman *web* baru, termasuk fasilitas *Starter Pages* (Arief Ramadhan;2007:2).

## **II.6. Database**

*Database* merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan digunakan diperangkat lunak untuk memanipulasinya. *Database* merupakan salah satu komponen yang sangat penting dalam sistem informasi, karena merupakan basis sistem dalam menyediakan informasi bagi para pemakai.(*Indra Warman ; 2012 : 45*)

### **II.6.1. Jenis-Jenis Database**

#### **1. Database Hirarki**

Yaitu suatu data yang tersusun dengan bentuk hirarki pohon. Susunan yang seperti ini terdiri dari beberapa unsur komponen yang saling mempengaruhi dan tidak dapat dipisahkan, jenis *database* ini merupakan hubungan satu komponen dengan banyak komponen. (*Indra Warman ; 2012 : 46*)

## 2. *Database* Relasi

Adalah suatu data yang disusun dalam bentuk tabel yang terdiri dari dua definisi dan tersusun secara terstruktur. Bentuk susunan dua dimensi ini terdiri dari beberapa kolom dan *record* yang tersusun berbentuk baris dari kiri kekanan. Data- data yang susunannya berbentuk baris adalah susunan yang menurun kebawah. Dimana pada setiap baris berisikan data- data yang saling berkaitan satu sama lainnya. Artinya setiap pemasukan data yang tersimpan pada *field* merupakan kesatuan dalam bentuk satu baris.( *Indra Warman ; 2012 : 46*)

### **II.6.2. Prinsip-Prinsip *Database***

Sistem database manajemen dibentuk untuk mengurangi masalah-masalah dalam organisasi. Misalnya data/informasi tidak tersedia atau saling tumpang tindih. Berikut prinsip manajemen database menurut *Anis nurhanafi* (2013 : 3) adalah:

#### 1. Ketersediaan

Data mudah diakses oleh suatu program dan pemakai ( *user* ) dimanapun dan kapanpun diperlukan.

#### 2. Pemakaian bersama

Struktur data disusun sedemikian hingga dapat digunakan oleh beberapa pemakai bersama-sama untuk mengurangi redundansi data.

#### 3. Pengembangan

*Databases* dapat dikembangkan sesuai dengan perkembangan kebutuhan pemakai. *Databases* dapat dimodifikasi untuk pengembangan selanjutnya dan dapat beradaptasi dengan lingkungan.

#### 4. Kesatuan

*Databases* dibentuk dalam satu kesatuan untuk memudahkan pengontrolannya (pemeliharaan dan pengawasan) mudah dilakukan.

### II.7. MySql

*MySQL Server* 2000 adalah suatu Perangkat lunak *Relational Database Management system* ( RDBMS ) yang handal. Didesain untuk mendukung proses transaksi yang besar (seperti *order entri* yang *online*, inventori, akuntansi atau manufaktur). *MySQL Server* akan secara otomatis menginstal enam *database* utama, yaitu *master*, *model*, *tempdb*, *pubs*, *Northwind*, dan *Msdb*. (Anis nurhanafi ; 2013 : 5)

### II.8. UML (*Unified Modelling Language*)

*Unified Modelling Language* (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, *Java*, C# atau *VB.NET*. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. Seperti bahasa-bahasa lainnya, UML

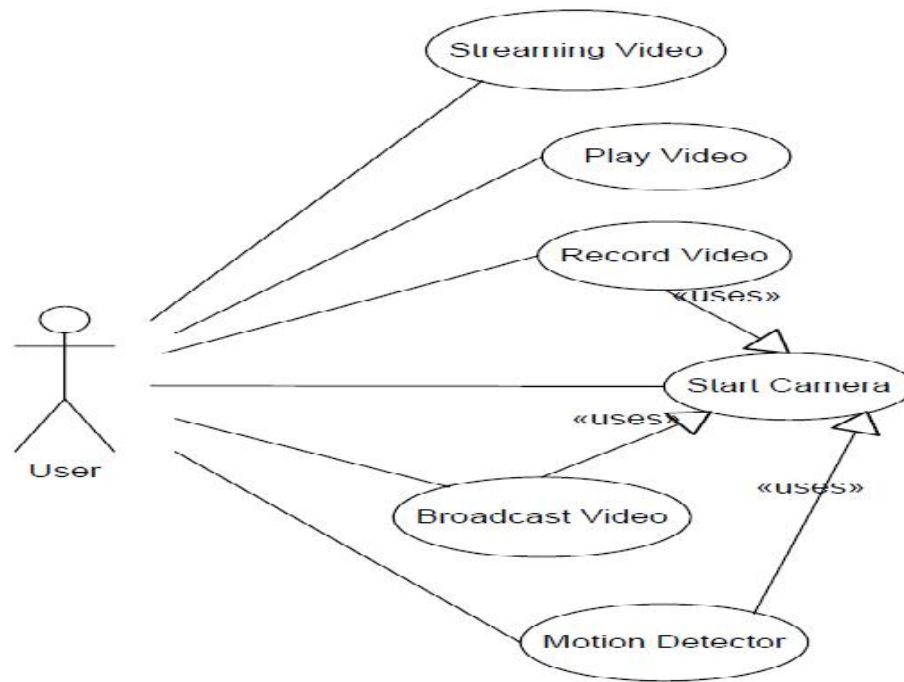
mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: *Grady Booch OOD (Object-Oriented Design)*, *Jim Rumbaugh OMT (Object Modeling Technique)*, dan *Ivar Jacobson OOSE (Object-Oriented Software Engineering)*. Sejarah UML sendiri cukup panjang. Sampai era tahun 1990 seperti kita ketahui puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: *metodologi booch, metodologi coad, metodologi OOSE, metodologi OMT, metodologi shlaer-mellor, metodologi wirfs-brock*, dsb. Masa itu terkenal dengan masa perang metodologi (*method war*) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan group/perusahaan lain yang menggunakan metodologi yang berlainan. Dimulai pada bulan Oktober 1994 *Booch, Rumbaugh dan Jacobson*, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan memelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 direlease *draft* pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh *Object Management Group (OMG – <http://www.omg.org>)*. Tahun 1997 UML versi 1.1 muncul, dan saat ini versi terbaru adalah versi 1.5 yang dirilis bulan Maret 2003. *Booch, Rumbaugh dan Jacobson* menyusun tiga buku serial tentang UML pada tahun 1999. Sejak saat itulah UML telah menjelma

menjadi standar bahasa pemodelan untuk aplikasi berorientasi objek. (Yuni Sugiarti ; 2013 : 33)

Dalam pembuatan skripsi ini penulis menggunakan diagram *Use Case* yang terdapat di dalam UML. Adapun maksud dari *Use Case Diagram* diterangkan dibawah ini.

#### 1. *Use Case Diagram*

*Use case diagram* adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar dan menjelaskan sistem secara fungsional yang terlihat pengguna. Yang ditekankan dalam *use case* adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. *Use case* mempresentasikan sebuah interaksi antara aktor dan sistem. *Use case* dapat digunakan selama proses analisis untuk menangkap persyaratan sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, *use case diagram* berperan untuk mendapatkan perilaku sistem saat diimplementasikan. Komponen pembentuk diagram *use case* yang pertama adalah *actor*. *Actor* mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan inputan pada sistem, hanya menerima informasi dari sistem atau keduanya. *Actor* hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*. *Actor* digambarkan dengan *stick man*. Selain itu komponen pembentuk lainnya adalah *use case*. *Use case* adalah gambaran fungsionalitas dari sistem sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. (Rian Aldy Hidayat ; 2012 : 253)







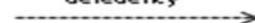
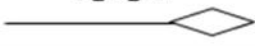


**Gambar II.1. Use Case Diagram**

*Sumber : (Rian Aldy Hidayat ; 2012 : 253)*

## 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. Berikut adalah simbol-simbol pada diagram kelas :

Simbol	Deskripsi
	Package merupakan sebuah bungkus dari satu atau lebih kelas
	Kelas pada struktur sistem
	sama dengan konsep interface dalam pemrograman berorientasi objek
	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
	relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
	relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
	relasi antar kelas dengan makna kebergantungan antar kelas
	relasi antar kelas dengan makna semua-bagian (whole-part)

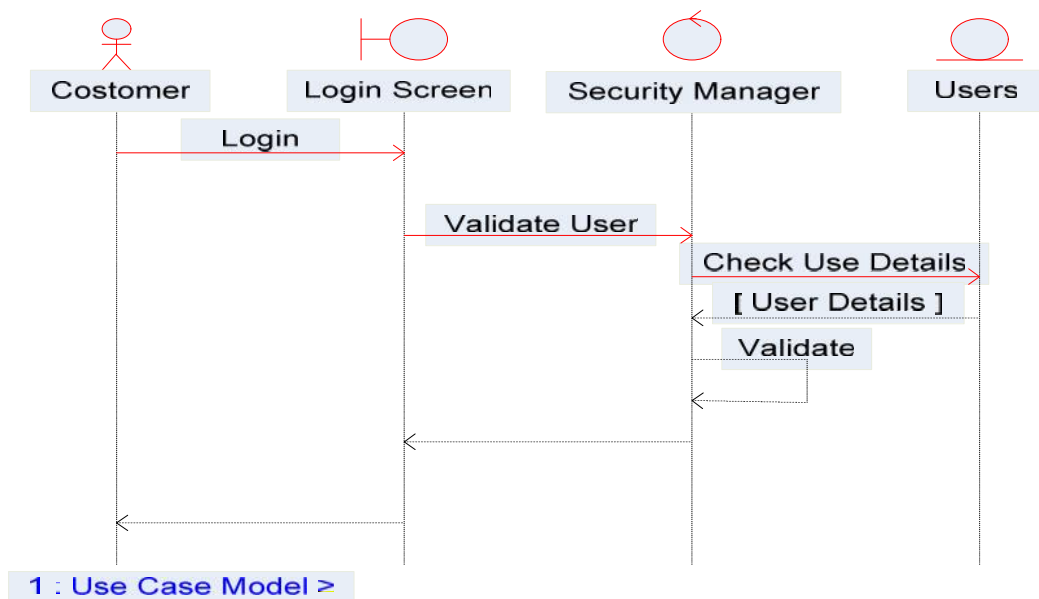
**Gambar II.2. Class Diagram**

Sumber : (Yuni Sugiarti ; 2013 : 59)

### 3. Sequence Diagram

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

Banyaknya diagram *sequence* yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram *sequence* sehingga semakin banyak *use case* yang didefinisikan maka diagram *sequence* yang harus dibuat juga semakin banyak.



**Gambar II.3. Contoh Sequence Diagram**

*Sumber : (Yuni Sugiarti ; 2013 : 63)*

#### 4. Activity Diagram

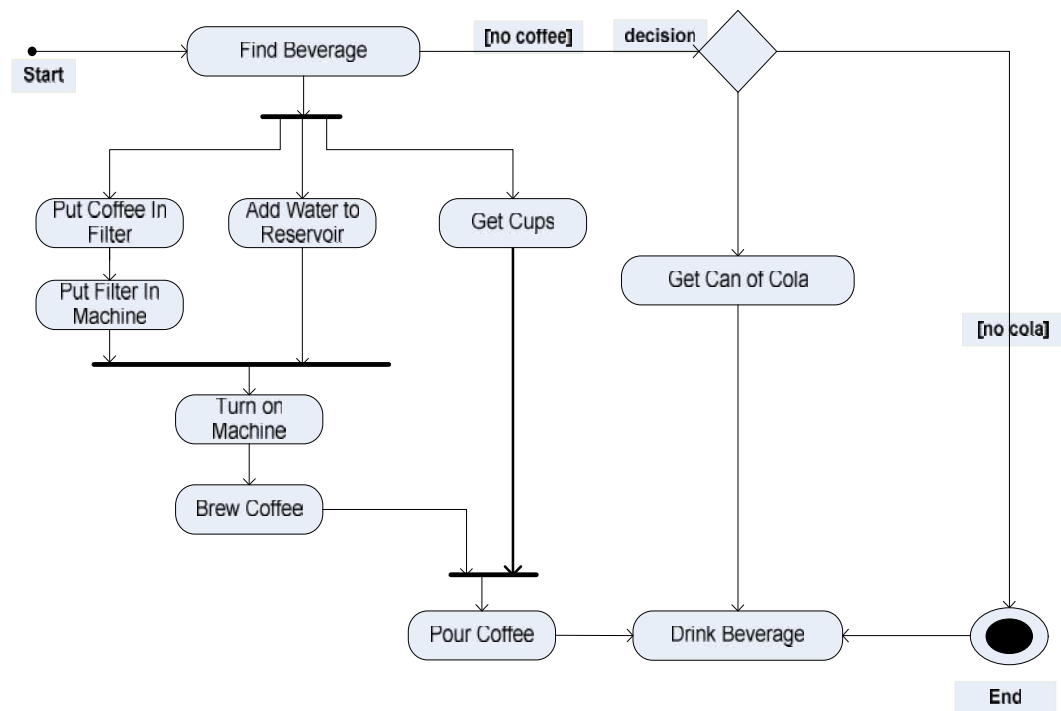
*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

*Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal.

*Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.



**Gambar II.4. Activify Diagram**

*Sumber : (Yuni Sugiarti ; 2013 : 76)*