

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terdahulu**

Selama melakukan penelitian pada PT. 212 Mart mengenai sistem pendukung keputusan menentukan penjualan produk susu terlaris , yaitu bagaimana peneliti memecahkan masalah yang ada pada perusahaan, sehingga peneliti merancang suatu Sistem pendukung keputusan menentukan produk susu terlaris menggunakan metode SMART (Studi Kasus: PT. 212 Mart) . Agar mempermudah perusahaan dalam menentukaan penjualan produk susu terlaris, dan peneliti membanding penelitiannya dengan jurnal-jurnal yang ada yang menggunakan metode SMART.

Sistem pendukung keputusan penentuan supplier bahan bangunan menggunakan metode SMART (*simple multi attribute ranking technique*) pada toko bintang keramik jaya. (Irwan ukkas, dkk : 2017). Sistem Pendukung Keputusan (SPK) adalah sistem yang dapat membantu seseorang dalam mengambil keputusan yang akurat dan tepat sasaran. Banyak permasalahan yang dapat diselesaikan dengan menggunakan SPK, salah satunya adalah Penentuan Supplier Bahan Bangunan dengan menggunakan bantuan metode SMART Pada Toko Bintang Keramik Jaya. Tujuan dari penelitian ini adalah untuk menghasilkan Sistem Pendukung Keputusan Penentuan Supplier Bahan Bangunan Menggunakan Metode SMART (*Simple Multi Attribute Ranking Technique*) Pada Toko Bintang Keramik Jaya dengan harapan seleksi Supplier secara *obyektif*.

Dengan menggunakan bahasa pemrograman *Visual Basic 6.0* dan databasenya menggunakan *Microsoft Access*. Dalam penelitian ini, teknik pengumpulan data yang digunakan adalah studi pustaka, observasi dan wawancara. Hasil dari penelitian ini adalah dibuatnya sistem pendukung keputusan untuk mengetahui Penentuan Supplier yang layak, dipertimbangkan, dan belum terpilih untuk dipilih. Pengguna dapat menginputkan data *Supplier*, data kriteria, data subkriteria. Kemudian sistem akan mencari solusi dengan metode SMART. Setelah keputusan didapatkan, maka sistem akan menampilkan keputusan tersebut.

Menurut penelitian Dwi Novianti, Indah Fitri Astuti dan Dyna Marisa Khairina (2016), dengan judul peneliti Sistem Pendukung Keputusan Berbasis Web Untuk Pemilihan Café Menggunakan Metode Smart. Berdasarkan hasil pengembangan Sistem Pendukung Pengambilan Keputusan Pemilihan Kamera DSLR menggunakan Metode SMART ini, maka dapat disimpulkan bahwa pembangunan sistem sudah sesuai dengan rancangannya dan sudah mampu berjalan dengan baik sistem yang dibuat sudah memberikan informasi *spesifikasi* kamera DSLR dengan detail, yang akan menjadi bahan pertimbangan lanjutan bagi pengguna ketika menentukan pemilihan kamera DSLR.

Sistem pendukung keputusan penentuan lokasi pembuatan batu bata menggunakan metode SMART (Zara Yunizar,2018) Batu bata merupakan salah satu bahan utama dalam pembuatan konstruksi bangunan. Batu bata dikenal dan digunakan masyarakat karena harganya relatif murah, mudah didapat dan tahan lama. Batu bata terbuat dari tanah liat, air dan bahan campuran lain, yang

pembuatannya mengalami proses pembakaran. Proses pembakaran ini menimbulkan pro dan kontra di masyarakat karena menimbulkan polusi udara dan penyakit pernafasan bagi penduduk yang jarak rumahnya relatif dekat dengan lokasi pembuatan batu bata. Salah satu solusi yang dilakukan untuk mengatasi hal tersebut adalah dibuatnya lokasi pembuatan batu bata yang strategis, dengan mempertimbangkan faktor biaya, efisiensi serta aman terhadap masyarakat sekitar. Metode *Simple Multi Attribute Ranking Technique* (SMART) adalah metode pengambilan keputusan multi atribut yang dapat digunakan untuk mendukung pembuat keputusan dalam memilih beberapa alternatif. Dalam prosesnya, metode SMART menggunakan linier adaptif model untuk meramal nilai setiap alternatif. Kriteria yang digunakan dalam penelitian adalah luas lokasi, jarak dengan lokasi pengambilan bahan utama (tanah lempung), jarak dari perumahan penduduk, jarak dari jalan utama, mudah dijangkau, dengan jumlah sample 5 lokasi yaitu lokasi A, lokasi B, lokasi C, lokasi D, dan lokasi E. Sistem pendukung keputusan ini diharapkan dapat membantu pembuat keputusan dalam menentukan lokasi strategis pembuatan batu bata yang tidak hanya menguntungkan bagi pemilikinya, namun juga bagi masyarakat sekitar. Sehingga hasil penelitian menunjukkan bahwa lokasi strategis yang direkomendasikan adalah lokasi D dengan nilai akhir.

Implementasi Metode *Simple Multiple Attribute Rating Technique* (SMART) dalam Pemilihan Hotel di Kota Palu (Hajra Rasmita, dkk : 2017) Penelitian ini bertujuan untuk membantu calon pengunjung hotel yang ada di Kota Palu dalam menentukan pilihan hotel yang pengunjung inginkan sesuai dengan

kriteria. Dalam penelitian ini menggunakan kriteria yang biasanya menjadi penilaian pengunjung dalam memilih hotel. Kriterianya adalah harga, lokasi, fasilitas, pelayanan dan kelas. Metode yang digunakan dalam mengembangkan sistem adalah Extreme Programming dengan empat tahap yaitu planing, design, coding, dan testing. Hasil yang didapatkan dalam penelitian ini adalah sebuah sistem penunjang keputusan menggunakan metode SMART yang dapat digunakan untuk membantu calon pengunjung dalam menentukan hotel prioritas yang sesuai dengan kriteria yang diinginkan.

## **II.2. Landasan Teori**

### **II.2.1. Sistem Pendukung Keputusan**

Sistem Pendukung Keputusan merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Sistem Pendukung Keputusan juga merupakan sistem informasi berbasis *computer* untuk manajemen pengambilan keputusan yang menangani masalah-masalah semi struktur. Dengan pengertian diatas dapat dijelaskan bahwa sistem pendukung keputusan bukan merupakan alat pengambilan keputusan, melainkan merupakan sistem yang membantu pengambil keputusan dengan melengkapi mereka dengan informasi dari data yang telah diolah dengan relevan dan diperlukan untuk membuat keputusan tentang suatu masalah dengan lebih cepat dan akurat. Sehingga sistem ini tidak dimaksudkan untuk menggantikan pengambilan keputusan dalam proses pembuatan keputusan

Sebuah aplikasi berupa Sistem Pendukung Keputusan (*Decision Support System*) mulai dikembangkan pada tahun 1970. *Decision Support Sistem* (DSS) dengan didukung oleh sebuah sistem informasi berbasis komputer dapat membantu seseorang dalam meningkatkan kinerjanya dalam pengambilan keputusan. Untuk memberikan pengertian yang lebih mendalam, akan diuraikan beberapa definisi mengenai SPK yang dikembangkan oleh beberapa ahli, diantaranya oleh Man dan Watson yang memberikan definisi sebagai berikut, SPK merupakan suatu sistem yang interaktif, yang membantu pengambil keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah yang sifatnya semi terstruktur maupun yang tidak terstruktur. (Dwi Novianti, dkk. 2016)

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan berkumpul bersama – sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Sedangkan menurut Raymond Mcleod (Raymond, 2008), sistem pendukung keputusan merupakan sebuah sistem yang menyediakan kemampuan untuk penyelesaian masalah dan komunikasi untuk permasalahan yang bersifat semi-terstruktur. Sistem pengambilan keputusan adalah keputusan-keputusan yang dibuat untuk memecahkan masalah. Pada sistem pengambilan keputusan akan menghasilkan beberapa alternatif penyelesaian masalah, sedangkan keputusan terakhir tetap terletak pada pemakai. Sistem Pendukung Keputusan (SPK) atau DSS (*Decision Support System*) (Steven, 2002) merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk

membantu pengambilan keputusan dalam situasi yang semi-terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat. Menurut Turban (Turban, 2005) sistem pendukung keputusan atau DSS (Decision Support System) merupakan sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semiterstruktur. DSS dimaksudkan untuk menjadi alat bantu bagi para pengambil keputusan untuk memperluas kapabilitas mereka, namun tidak untuk menggantikan penilaian mereka. DSS ditujukan untuk keputusan-keputusan yang memerlukan penilaian atau pada keputusan-keputusan yang sama sekali tidak dapat didukung oleh algoritma. (Halim Agung dan Ricky : 2016 )

### **II.2.2. Penjualan**

Penjualan ialah menjual barang dagang yang menjadi usaha pokok suatu perusahaan dengan dilakukan secara terus menerus dan tepat. penjualan juga adalah salah satu langkah pemasaran dari suatu perusahaan, sehingga perusahaan dapat memperoleh keuntungan yang akibatnya adalah kegiatan operasional perusahaan dapat tetap dijalankan. Berikut pengertian penjualan menurut beberapa ahli: diambil dari bukunya yang merupakan buku manajemen penjualan edisi ketiga, yaitu "Menjual adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang dan jasa yang ditawarkan". Menurut Winardi (2005:26), "Penjualan adalah berkumpulnya seorang pembeli dan penjual dengan tujuan melaksanakan tukar menukar barang dan jasa berdasarkan pertimbangan yang berharga misalnya

pertimbangan uang”. Menurut Thamrin Abdullah dan Francis Tantri,(2016:3) ”Penjualan adalah bagian dari promosi dan promosi adalah salah satu bagian dari keseluruhan sistem pemasaran” ( Anthony Anthony : 2017).

### **II.2.3. Produk**

Produk memiliki arti penting bagi perusahaan karena tanpa adanya produk, perusahaan tidak akan dapat melakukan apapun dari usahanya. Pembeli akan membeli produk kalau merasa cocok, karena itu produk harus disesuaikan dengan keinginan ataupun kebutuhan pembeli agar pemasaran produk berhasil. Dengan kata lain, pembuatan produk lebih baik diorientasikan pada keinginan pasar atau selera konsumen. Menurut Kotler dan Armstrong (2001: 346) adalah ”Segala sesuatu yang dapat ditawarkan ke pasar untuk mendapatkan perhatian, dibeli, digunakan, atau dikonsumsi yang dapat memuaskan keinginan atau kebutuhan”. Saat ini semua produsen memahami begitu pentingnya peranan arti kualitas produk yang unggul untuk memenuhi harapan pelanggan pada semua aspek produk yang dijual ke pasar. Para petinggi perusahaan pun semakin menyadari dan mempercayai adanya keterhubungan langsung antara kualitas produk terhadap keputusan pembelian konsumen yang pada akhirnya akan meningkatkan sasaran pasar (Riyono dan Gigih Erlik Budiharja : 2016).

#### **II.2.4. Susu**

Susu adalah cairan yang dihasilkan dari sekresi kelenjar mammae hewan mamalia yang berfungsi untuk memenuhi kebutuhan gizi . Dalam kebijakan pembangunan nasional, subsektor peternakan mendapat perhatian yang cukup besar dari pemerintah. Tujuan pembangunan subsektor peternakan antara lain mencukupi target gizi sesuai dengan yang dicanangkan dan bersumber dari protein hewani. Target pemerintah untuk konsumsi masyarakat akan protein hewani asal ternak adalah sebesar 4,6 g/kapita/hari. Susu sebagai salah satu produk ternak mempunyai kandungan zat gizi yang lengkap seperti protein (3,5%), lemak (3,9%), laktosa (4,9%), mineral, dan vitamin (0,7%). Sifat zat gizi tersebut mudah dicerna dan diserap secara sempurna (Elisa Putri : 2016).

#### **II.2.5. SMART (*Simple Multi Attribute Ranking Technique*)**

Metode Simple Multi Attribute Rating Technique merupakan suatu metode untuk pengambilan keputusan multi kriteria yang dikembangkan pada tahun 1997 oleh Edward. Metode SMART didasarkan pada teori bahwa setiap alternatif terdiri dari sejumlah kriteria yang memiliki nilai dan setiap kriteria memiliki bobot yang menggambarkan seberapa penting nilai dari bobot tersebut dibandingkan dengan kriteria lain [1]. Metode SMART lebih sering digunakan karena kesederhanaannya dalam merespon kebutuhan pembuat keputusan dan menganalisa respon. SMART menggunakan linear additive model untuk meramal

nilai setiap alternatif dan metode pengambilan keputusannya fleksibel. Metode ini memberikan pemahaman masalah yang tinggi dan dapat diterima oleh pembuat keputusan. Langkah – langkah penyelesaian metode SMART secara umum adalah sebagai berikut :

1. Menentukan masalah
2. Menentukan kriteria yang akan digunakan
3. Menentukan alternatif yang akan digunakan
4. Memberi bobot pada setiap kriteria pada setiap alternatif

Dimana :

$W_{ij}$  = Bobot kriteria pada baris i kolom ke j

$C_{out}$  = nilai record

$C_{min}$  = nilai minimal pada kriteria ke x

$C_{max}$  = nilai maksimal pada kriteria ke x 5.

5. Hitung nilai normalisasi pada setiap kriteria di setiap alternatif

Dimana :

$W_{ij}$  = Bobot kriteria pada baris i kolom ke j

6. Hitung nilai utilities pada setiap alternatif

Dimana :

$w_j$  = nilai pembobotan kriteria ke-j dan k kriteria

$u(a_i)$  = nilai utility kriteria ke-i untuk kriteria ke-i

7. Lakukan perankingan berdasarkan nilai utilities

8. Pilih alternatif dengan nilai utilities terbesar

### II.2.6. Database Management System (DBMS)

*Database Management System* atau disingkat DBMS adalah perangkat lunak atau (*software*) yang berfungsi untuk mengelola *database*. Mulai dari mengelola *databases* sampai dengan proses yang berlaku dalam *database* tersebut, baik berupa *entry*, *edit*, hapus, *query* terhadap data, membuat laporan dan lain sebagainya secara efektif dan efisien, ( Nurlaila Hasyim, dkk; 2014: 2 - 3).

### II.2.7. SQL Server 2008

SQL Server 2008 adalah sebuah terobosan baru dari *Microsoft* dalam bidang *database*. SQL Server adalah 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. *Microsoft* merilis *SQL Server* 2008 dalam beberapa versi yang disesuaikan dengan segment-segment pasar yang dituju. Versi-versi tersebut adalah sebagai berikut. Menurut cara pemrosesan data pada *prosesor* maka *Microsoft* mengelompokkan produk ini berdasarkan 2 jenis yaitu :

- a. Versi 32-bit (x86), yang biasanya digunakan untuk komputer dengan *single* prosesor (Pentium 4) atau lebih tepatnya *prosesor* 32 bit dan sistem operasi Windows XP.
- b. Versi 64-bit (x64), yang biasanya digunakan untuk komputer dengan lebih dari satu *prosesor* (Misalnya Core 2 Duo) dan *system* operasi 64 bit seperti Windows XP 64, Vista, dan Windows 7.
- c. Versi Compact, ini adalah versi “Tipis” dari semua versi yang ada. Versi ini seperti versi desktop pada SQL Server 2000. Versi ini juga digunakan pada *handheld device* seperti Pocket PC, PDA, *SmartPhone*, Tablet PC.
- d. Versi *Express*, ini adalah versi “Ringan” dari semua versi yang ada (tetapi versi ini berbeda dengan versi *compact*) dan paling cocok untuk latihan. *Express Manager standar, integrasi* dengan CLR dan XML, (Agus Tinus Setiawan, dan Rika; 2016:54).

## II.2.8. Microsoft Visual Studio 2010

*Microsoft Visual Studio* merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi *console*, aplikasi Windows, ataupun aplikasi Web. *Visual Studio* mencakup kompiler, *Software Development Kit (SDK)*, *Integrated Development Environment (IDE)*, dan dokumentasi (umumnya berupa *MSDN Library*). Kompiler yang dimasukkan ke dalam paket *Visual Studio* antara lain Visual C++, Visual C#, *Visual Basic*, *Visual Basic .NET*, *Visual Inter Dev*, Visual J++, Visual

J#, *Visual Fox Pro*, dan *Visual Source Safe*. *Microsoft Visual Studio* dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas *Windows*) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu, *Visual Studio* juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*). *Visual Studio* sebelumnya versi *Visual Studio 9.0.21022.08*, atau dikenal dengan sebutan *Microsoft Visual Studio 2008* yang diluncurkan pada 19 November 2007, yang ditujukan untuk platform *Microsoft .NET Framework 3.5*. Versi sebelumnya, *Visual Studio 2005* ditujukan untuk platform *.NET Framework 2.0* dan *3.0*. *Visual Studio 2003* ditujukan untuk *.NET Framework 1.1*, dan *Visual Studio 2002* ditujukan untuk *.NET Framework 1.0*. Versi-versi tersebut di atas kini dikenal dengan sebutan *Visual Studio.NET*, karena memang membutuhkan *Microsoft .NET Framework*. Sementara itu, sebelum muncul *Visual Studio.NET*, terdapat *Microsoft Visual Studio 6.0 (VS1998)*, (Herpendi;2016:1).

### **II.2.9. Normalisasi**

Normalisasi merupakan parameter digunakan untuk menghindari duplikasi terhadap tabel dalam basis data dan juga merupakan proses mendekomposisikan sebuah tabel yang masih memiliki beberapa anomali atau ketidak wajaran sehingga menghasilkan tabel yang lebih sederhana dan struktur yang bagus, yaitu sebuah tabel yang tidak memiliki data *redundancy* dan memungkinkan *user* untuk

melakukan *insert*, *delete*, dan *update* pada baris (*recod*) tanpa menyebabkan inkonsistensi data. Tujuannya untuk menghindari beberapa anomali:

1. *Insertion Anomaly* adalah proses melakukan penambahan *recod* baru akan tetapi mempengaruhi *user* untuk terjadinya duplikasi data.
2. *Deletion Anomaly* adalah proses melakukan penghapusan *record* akan tetapi akan menyebabkan hilangnya data yang akan dibutuhkan pada *record* lain.
3. *Modification Anomaly* adalah proses merubah data pada sebuah *record* mempengaruhi perubahan pada *record* lain karena adanya duplikasi.

Proses pernomalan tabel ada beberapa tahap yang harus dilakukan yaitu :

1. *First Normal Form (1 NF)*

Sudah tidak ada repeating group yaitu pengulangan yang terjadi pada beberapa atribut atau kolom dalam sebuah tabel, dan juga setiap atribut harus bernilai tunggal. Atribut multivalued, composite, derive tidak tunggal. Setiap nilai dari atribut hanya mempunyai nilai tunggal.

2. *Second Normal Form (2 NF)*

Untuk menjadikan tabel normal tingkat ke 2 maka sudah 1NF dan setiap atribut yang bukan primary key sepenuhnya secara fungsional tergantung pada semua atribut pembentuk primary key.

3. *Third Normal Form (3 NF)*

Tabel sudah 2NF dan tidak memiliki transitive dependencies, Transitive Dependencies adalah ketika atribut yang secara tidak langsung tergantung pada primary key dan atribut tersebut juga tergantung pada atribut lain yang bukan primary key.

4. *Boyce-codd Normal Form (BCNF)*

Tabel dalam *BCNF* jika sudah 3NF dan semua determinants adalah candidate keys. Perbedaan 3NF dan *BCNF* adalah untuk functional dependency  $A - B$ , 3NF memperbolehkan ketergantungan ada dalam relasi jika B adalah primary key dan A bukan merupakan candidate key. Sedangkan *BCNF* menuntut untuk ketergantungan tetap ada dalam relasi, A harus menjadi candidate key.

5. *Four Normal Form (4 NF)*

Relasi berada pada bentuk normal keempat apabila memenuhi syarat *BCNF* dan tidak mempunyai multivalued dependency.

6. *Fifth Normal Form (5 NF)*

Tabel bentuk normal kelima sering disebut *PJNF (Projection Join Normal Form)*, Penyebutan *PJNF* karena untuk suatu relasi akan berbentuk normal kelima jika tabel tersebut dapat dipecah atau diproyeksikan menjadi beberapa tabel dan dari proyeksi-proyeksi itu dapat disusun kembali (*join*) menjadi tabel yang sama dengan keadaan semula. Jika penyusunan ini tidak mungkin dilakukan dikatakan pada relasi itu terdapat join dependencies dan dikatakan bersifat *lossy join*.

## II.2.10. UML (*Unified Modeling Language*)

*Unified Modeling Language (UML)*. UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam

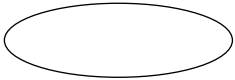
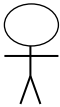
mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Gellysa; 2015 : 93-94).



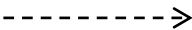

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

### 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, Adapun simbol-simbol yang digunakan usecase diagram dapat dilihat pada tabel II.1 :

**Tabel II.1 Simbol *Use Case***

Gambar	Keterangan
	<p><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>.</p>
	<p>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</p>




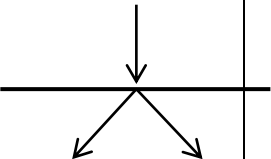
	control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka 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.

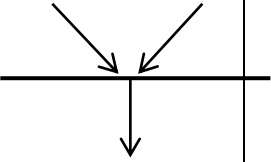
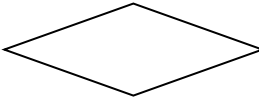

(Sumber : Gellysa; 2015 : 93-94)

## 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*, Adapun simbol-simbol yang digunakan usecase diagram dapat dilihat pada table II.2 :

**Tabel II.1 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.

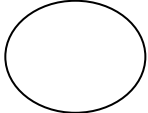
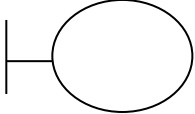
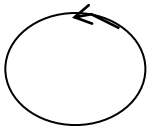

	<p><i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.</p>
	<p><i>Decision Points</i>, menggambarkan pilihan untuk pengambilan keputusan, <i>true</i>, <i>fals</i></p>
	<p><i>Swimlane</i>, pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.</p>

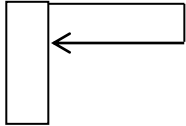


(Sumber : Gellysa; 2015 : 94)

### 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*, Adapun simbol-simbol yang digunakan usecase diagram dapat dilihat pada table II.3 :

**Tabel II.3. Simbol *Sequence Diagram***

Gambar	Keterangan
	<p>Entity Class, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.</p>
	<p>Boundary Class, berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan form cetak</p>
	<p>Control class, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p>Message, simbol mengirim pesan antar class.</p>

	Recursive, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	Activation, activation mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	Lifeline, garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat activation.

(Sumber : Gellysa; 2015 : 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. Adapun simbol-simbol yang digunakan *Multiplicity class diagram* dapat dilihat pada table II.4 :

**Tabel II.4. *Multiplicity Class Diagram***

<b>Multiplicity</b>	<b>Penjelasan</b>
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; 2015 : 95)*