

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu :

1. Penelitian yang dilakukan oleh M. Hadi Prayitno, 2017, “Sistem Informasi Eksekutif Pemasaran Dengan Metode *Drill Down*”. Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa penyampaian informasi pemasaran dapat lebih mudah dipahami oleh pihak eksekutif sehingga pengambilan keputusan menjadi lebih cepat. Sistem Informasi Eksekutif Pemasaran yang dibuat mampu memberikan informasi yang dibutuhkan oleh pihak eksekutif dalam memantau perkembangan pemasaran secara cepat dan tepat. Pihak eksekutif dapat mengakses system informasi eksekutif pemasaran berbasis *web* dimanapun mereka berada dan lebih praktis.
2. Penelitian yang dilakukan oleh Priyo Sutopo, Dedi Cahyadi dan Zainal Arifin, 2017, “Sistem Informasi Eksekutif Sebaran Penjualan Kendaraan Bermotor 2 Di Kalimantan Timur Berbasis *Web*”. Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa sistem informasi eksekutif dapat digunakan sebagai alat bantu dalam penyampaian informasi penjualan yang dapat dipergunakan dalam pengambilan keputusan untuk penjualan kendaraan bermotor.

3. Penelitian yang dilakukan oleh M. Tirta Mulia dkk, 2018, “Pengembangan Sistem Informasi Eksekutif dengan Dukungan *Web Service* Studi Kasus : Fakultas Teknik Unpas”. Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa menggunakan *web service* sebagai *feeder* data memberikan kemudahan dalam sisi pengembangan yang didapat dari pemisahan *front end* dan *back end* aplikasi yang bisa memilih prioritas informasi yang diinginkan.

II.2. Landasan Teori

II.2.1. Data

Menurut Siregar (2010:9) dalam kamus bahasa Inggris-Indonesia, data diterjemahkan sebagai istilah yang berasal dari kata “datum” yang berarti fakta atau bahan-bahan keterangan. Data merupakan deskripsi dari sesuatu dan kejadian yang kita hadapi. Data adalah fakta yang jelas lingkup, tempat, dan waktunya. (Priyo sutopo, dkk : 2016)

Data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya suatu pengolahan. Data bisa berwujud suatu keadaan, gambar, suara, huruf, angka, matematika, bahasa ataupun simbol-simbol lainnya yang bisa kita gunakan sebagai bahan untuk melihat lingkungan, obyek, kejadian ataupun suatu konsep. (Rachel : 2016).

II.2.2. Tipe Data

Tipe Data adalah suatu bentuk penggolongan jenis data berdasarkan kategori data, ukuran, dan kegunaan data yang dapat di tampung. Oleh sebuah variable dalam media bahasa pemrograman. (Muhammad Hari Ramadhan : 2015)

II.2.3. Sistem

Secara umum sistem adalah kumpulan dari beberapa bagian tertentu yang saling berhubungan secara harmonis untuk mencapai suatu tujuan tertentu. (Priyo : 2016)

Sistem adalah sekumpulan elemen – elemen yang sering berinteraksi untuk mencapai suatu tujuan tertentu (Muhammad Hari Ramadhan : 2015).

II.2.3.1. Elemen – Elemen Sistem

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu *subsistem* atau bagian-bagian dari sistem.

2. Batasan Sistem

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem.

4. Penghubung Sistem

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

5. Masukan Sistem

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

6. Keluaran Sistem

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan atau sasaran. Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. (Priyo : 2016)

II.2.4. Informasi

Dalam Jurnal Priyo Sutopo, dkk, 2016 menyatakan Informasi adalah data yang diolah menjadi bentuk yang berguna dan menjadi berarti bagi penerimanya. Kegunaan informasi adalah untuk mengurangi ketidak pastian di dalam proses

pengambilan keputusan tentang suatu keadaan. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut. Kualitas informasi sangat dipengaruhi atau ditentukan oleh beberapa hal yaitu :

1. Relevan (*Relevancy*)
2. Akurat (*Accurancy*)
3. Tepat waktu (*Time liness*)
4. Ekonomis (*Economy*)
5. Efisien (*Efficiency*)
6. Ketersediaan (*Availability*)
7. Dapat dipercaya (*Reliability*)
8. Konsisten (Priyo Sutopo,ddk : 2016)

II.2.5. Sistem Informasi

Sistem Informasi adalah suatu sistem didalam satu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan untuk pihak luar tertentu. (Tata Sutabri : 2012)

II.2.6. Sistem Informasi Eksekutif

Sistem Informsi Eksekutif adalah sebuah sistem informasi yang mendukung perencanaan dan penilaian kebutuhan manajer eksekutif. Sistem

informasi eksekutif disesuaikan kebutuhan informasi unik para eksekutif yang merencanakan bisnis dan menilai performa terhadap rencana tersebut. (Nur Andriyan : 2015)

Sistem Informasi Eksekutif merupakan sistem untuk mendukung keputusan yang membantu eksekutif tingkat atas menganalisa, membandingkan informasi yang penting sehingga bisa memonitoring dan mengidentifikasi peluang serta masalah. (Aji : 2017)

II.2.7. Produksi

Produksi adalah kegiatan untuk menciptakan atau menambah kegunaan barang atau jasa. (Herlin Herawati : 2016)

Produksi adalah suatu kegiatan yang mengubah input menjadi output. (Effrida Manalu : 2017)

II.2.8. Sistem Produksi

Sistem Produksi adalah kegiatan untuk menciptakan atau menambah barang atau jasa. (Herlin dan Dewi : 2018)

Sistem Produksi adalah suatu proses dimana terdapat kegiatan pengolahan bahan mentah (*input*), dengan serangkaian tahapan-tahapan untuk menghasilkan produk (*output*), yang lebih bernilai maknanya. (Citra : 2018).

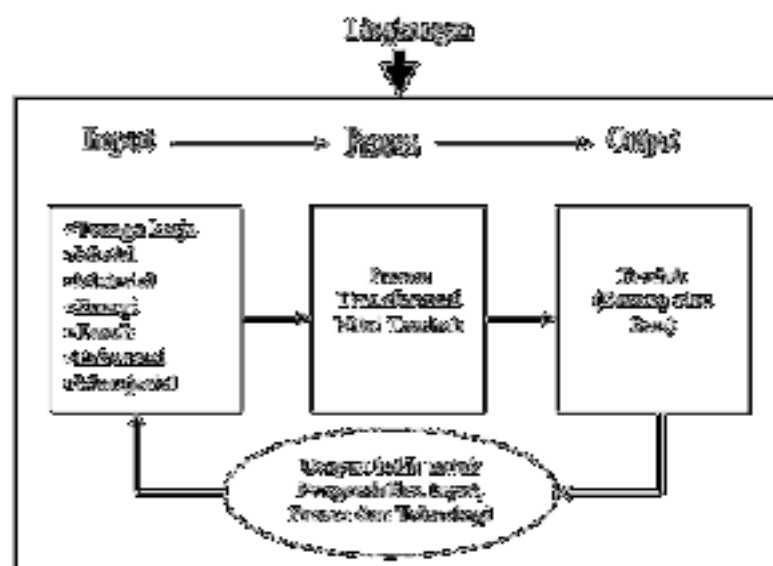
II.2.8.1. Karakteristik Sistem Produksi

Sistem produksi memiliki beberapa karakteristik sebagai berikut :

1. Mempunyai komponen-komponen atau elemen-elemen yang saling berkaitan satu sama lain dan membentuk satu kesatuan yang utuh. Hal ini berkaitan dengan komponen structural yang membangun sistem produksi itu.
2. Mempunyai tujuan yang mendasari keberadaannya, yaitu menghasilkan produk (barang dan/ atau jasa) berkualitas yang dapat dijual dengan harga kompetitif di pasar.
3. Mempunyai aktivitas berupa proses transformasi nilai tambah input menjadi output secara efektif dan efisien.
4. Mempunyai mekanisme yang mengendalikan pengoperasiannya, berupa optimalisasi pengalokasian sumber-sumber daya. (Citra : 2018).

II.2.8.2. Skema Sistem Produksi

Secara skematis sederhana, sistem produksi dapat digambarkan seperti dalam gambar dibawah ini :



Gambar II.1. Skema Sistem Produksi

(Sumber : Citra : 2018)

II.2.9. Metode *Drill Down*

Drill Down adalah teknik yang dapat memberikan informasi dan dapat diakses secara hierarkis, mulai dari yang bersifat umum atau global kemudian dipecah hingga ke level yang lebih detail. Informasi yang ditampilkan berupa grafik maupun tabel. (M.Hadi Prayitno : 2017)

Drill Down merupakan teknik analisa untuk menggeneralisasi atau menspesifikasi informasi, semakin keatas maka informasi makin ringkas dan semakin kebawah maka informasi semakin rinci. (Aminurlah dan Abdul : 2017)

II.2.10. Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah dari pengguna aplikasi tersebut yang berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Nunung, dkk : 2017)

Aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut. (M. Noor dan Ayu : 2017)

II.2.11. *Visual Basic*

Visual Basic yaitu *.NET Framework*, bahasa yang bekerja dengan *.NET Framework*, dan *Visual Basic IDE (Integrated Development Environment)*. *Visual Basic 2010* adalah inkarnasi dari bahasa *Visual Basic* yang sangat populer dan

telah dilengkapi dengan fitur serta fungsi yang setara dengan bahasa tingkat tinggi lainnya seperti C++. (Cristopher : 2014).

II.2.12. SQL Server

Microsoft *SQL Server* adalah sebuah sistem manajemen basis data (DBMS) yang dibuat oleh perusahaan *Microsoft* untuk ikut bersaing dengan aplikasi DBMS, seperti *Oracle* dan IBM. Bahasa query yang digunakan adalah *Transact-SQL* yang berasal dari implementasi SQL standar ANSI/ISO. *SQL Server* sering digunakan pada skala bisnis kecil sampai menengah, tetapi lambat laun berkembang pada skala besar. (Anton : 2017).

II.2.13. Basis Data

Basis data merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lain. Basis data atau *database* merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi berfungsi sebagai basis penyedia informasi bagi pemakainya, Sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan dengan yang lainnya dan untuk membuatnya tersedia beberapa aplikasi yang bermacam-macam dalam suatu sistem organisasi. (Priyo, dkk : 2016)

II.2.14. Database

Database merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi berfungsi sebagai basis penyedia informasi bagi pemakainya. (Priyo, dkk : 2016)

II.2.15. Normalisasi

Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar *ambiguity* bisa dihilangkan. Normalisasi merupakan sebuah teknik dalam *logical* desain sebuah basis data relasional yang mengelompokkan atribut dari suatu tabel sehingga membentuk struktur tabel yang normal. (Dwi Puspitasari, et al : 2016)

II.2.15.1. Tujuan Normalisasi

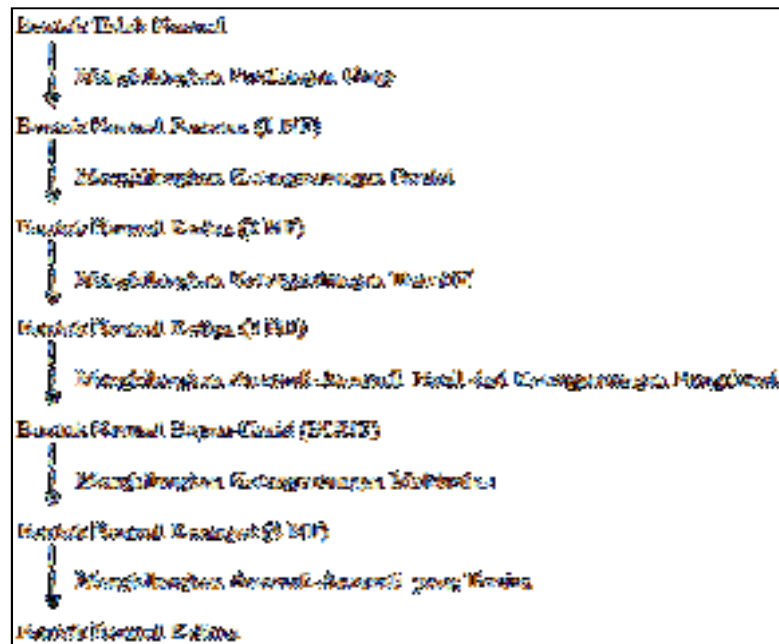
Tujuan dari normalisasi adalah untuk :

1. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh base relations(*file*).
2. Untuk mengurangi kompleksitas
3. Untuk mempermudah pemodifikasian data

Adapun proses normalisasi sebagai berikut:

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat kemudian.
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal. Untuk melakukan proses tersebut dibutuhkan beberapa tahapan. Tahapan dalam normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik. (Dwi Puspitasari, et al : 2016).

Urutan tahapan normalisasi dapat dilihat pada gambar II.1 sebagai berikut :



Gambar II.2. Tahapan Normalisasi
(Sumber : Dwi Puspitasari, et al : 2016)

II.2.15.2. Aturan-Aturan Normalisasi

Adapun aturan dalam normalisasi adalah suatu tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sbb:

1. Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*). Artinya, setelah tabel tersebut diuraikan / didekomposisi menjadi tabel-tabel baru, tabel-tabel baru tersebut bisa menghasilkan tabel semula dengan sama persis.
2. Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*).
3. Tidak melanggar *Boyce-Codd Normal Form* (BCNF) Jika kriteria ketiga (BCNF) tidak dapat terpenuhi, maka paling tidak tabel tersebut tidak melanggar Bentuk Normal tahap ketiga (3rd Normal Form / 3NF). Pada

penelitian ini formula yang dibuat sampai memenuhi bentuk normal ke 3 yaitu 3NF. (Dwi Puspitasari, et al : 2016)

II.2.16.UML (*Unified Modeling Language*)

Menurut Windu Gata, Grace dalam jurnal (Ade : 2016), *Unified Modeling Language* (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.


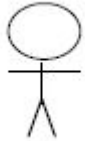


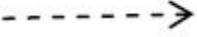
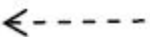
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi siapa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. (Ade Hendini : 2016)

Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel II.1. Simbol-simbol *Use Case Diagram*

Gambar	Nama	Keterangan
	<i>Use Case</i>	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja.
	<i>Actor</i>	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 kontrol terhadap <i>use case</i> .
	Asosiasi Aktif	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 Pasif	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila <i>actor</i> berinteraksi secara pasif dengan sistem.
	<i>Include</i>	<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 sebuah fungsi program
	<i>Extend</i>	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Ade Hendini : 2016)




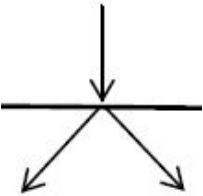
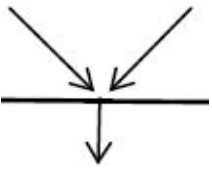
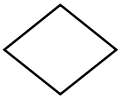

2. *Activity Diagram*

Activity diagram adalah diagram yang menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah

bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktivitas-aktivitas tersebut bergantung satu sama lain. *Activity Diagram* menggambarkan *Work Flow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. (Ade Hendini : 2016)

Simbol -simbol yang digunakan dalam *activity* diagram yaitu:

Tabel II.2. Simbol-simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Start Point</i>	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End Point</i>	<i>End point</i> , akhir aktifitas.
	<i>Activities</i>	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i>	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu.
	<i>Join</i>	<i>Join</i> (Penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i>	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan <i>true</i> , <i>false</i> .
	<i>Swimlane</i>	<i>Swimlane</i> , Pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

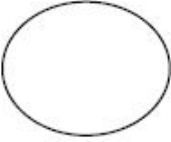
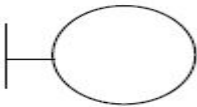

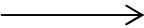
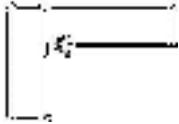
(Sumber : Ade Hendini : 2016)



3. *Sequence Diagram*

Sequence Diagram merupakan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima objek. (Ade Hendini : 2016).

Simbol-simbol yang digunakan dalam *sequence diagram* yaitu :

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

Gambar	Nama	Keterangan
	<i>Entity Class</i>	<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>	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan form cetak.
	<i>Control Class</i>	<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 objek.
	<i>Message</i>	<i>Message</i> , Simbol pengiriman antar pesan.
	<i>Recursive</i>	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirm untuk dirinya sendiri.

	<i>Activation</i>	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktiivtas sebuah operasi.
	<i>Lifeline</i>	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Hendini : 2016)

4. *Class Diagram*

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap 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. (Ade Hendini : 2016)

Class diagram secara khas meliputi: kelas (*class*), Relasi, *Associations*, *Generalization* dan *Aggregations*, 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. (Ade Hendini : 2016)

Tabel II.4. *Multiplicity Class Diagram*

<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau satu atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal

n..n	Batasan antara, Contoh 2..4 mempunyai arti minimal 2 maksimum 4
------	---

(Sumber : Ade Hendini : 2016)