

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Sistem**

Menurut Tata Sutabri (2010 : 10-11), “sebuah sistem terdiri atas bagian-bagian atau komponen yang terpadu untuk satu tujuan”. Model dasar dari suatu sistem adalah adanya masukan, pengolahan dan keluaran. Akan tetapi, sistem ini dapat dikembangkan hingga menyetakan media penyimpanan. Sistem dapat terbuka dan tertutup akan tetapi sistem informasi biasanya adalah sistem terbuka. Artinya sistem tersebut dapat menerima beberapa masukan dari lingkungan luarnya.

##### **2.1.1 Definisi Sistem**

Sistem merupakan kumpulan dari komponen - komponen yang saling berkaitan untuk mencapai tujuan tertentu (menurut Soendoro Herlambang dan Haryanto Tanuwijaya, 2005:116).

Dari pengertian beberapa diatas dapat disimpulkan bahwa Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi.

Pengolahan data akan menjadi suatu informasi bila diolah melalui sebuah model, bahwa pada awalnya data dimasukkan kedalam model untuk diproses kemudian setelah diproses data tersebut akan menghasilkan sebuah informasi yang bermanfaat bagi penerima sebagai dasar dalam membuat keputusan. Dari keputusan tersebut akan menghasilkan kejadian-kejadian tertentu yang akan digunakan kembali sebagai data yang nantinya akan dimasukkan kedalam model (proses).

### 2.1.2 Karakteristik Sistem

Menurut Tata Sutabri ( 2012 : 8 ) sistem mempunyai karakteristik atau sifat – sifat tertentu, yaitu:

- a. **Komponen Sistem ( *component* )**  
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan
- b. **Batasan sistem ( *Boundary* )**  
Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya.
- c. **Lingkungan Luar Sistem ( *Environment* )**  
Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem.
- d. **Penghubung Sistem ( *Interface* )**  
Media yang menghubungkan sistem dengan sub sistem yang lain.
- e. **Masukan Sistem ( *Input* )**  
Energi yang dimasukkan kedalam sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal ( *signal input* ).
- f. **Sistem ( *output* )**  
Hasil dari entity yang diolah dan diklasifikasikan menjadi keluaran yang berguna.
- g. **Sistem ( *Process* )**  
Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.
- h. **Sasaran Sistem ( *objective* )**  
Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministik*.

## **2.2 Konsep Dasar Informasi**

Informasi adalah sebuah istilah yang tepat dalam pemakaian umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi dan lain sebagainya. Informasi ibarat darah yang mengalir didalam tubuh suatu organisasi sehingga informasi ini sangat penting didalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati.

Informasi sangat penting didalam suatu organisasi, bagi manajer informasi yang diterimanya akan sangat bermanfaat untuk membantu dalam hal pengambilan keputusan. Informasi dapat didefinisikan sebagai berikut: Menurut Jogiyanto (2005:8) "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya". Sumber informasi adalah data. Data adalah sekumpulan kejadian yang diangkat dari suatu kenyataan yang belum dievaluasi. Data dapat terbentuk dari karakter yang berupa sistem, angka maupun simbol-simbol khusus atau gabungan dari keduanya yang menggambarkan suatu ide, objek, kondisi atau situasi tertentu.

Kualitas suatu informasi tergantung dari tiga hal, yaitu : informasi harus akurat (accurate), tepat waktu (timelines), dan relevan (relevance). Penjelasan tentang kualitas informasi tersebut akan dipaparkan dibawah ini :

a. **Akurat (accurate)**

Informasi harus bebas dari kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena biasanya dari sumber informasi sampai penerima informasi ada kemungkinan terjadi gangguan (noise) yang dapat mengubah atau merusak informasi tersebut.

b. **Tepat Waktu (timeline)**

Informasi yang datang pada si penerima tidak boleh terlambat.

Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan landasan dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini, mahalnya informasi disebabkan karena harus cepatnya informasi tersebut dikirim atau didapat sehingga diperlukan teknologi mutakhir untuk mendapat, mengolah dan mengirimkannya.

c. **Relevan (relevance)**

Informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk orang satu dengan yang lain berbeda, misalnya informasi sebab musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan apabila ditunjukkan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang relevan tetapi akan sangat relevan untuk seorang akuntan perusahaan.

### **2.2.1 Definisi Informasi**

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang (Menurut Gordon B. Davis, 2002:28).

### **2.2.2 Pengertian Sistem Informasi**

Sebuah sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut. Selain itu data juga memegang peranan yang penting dalam sistem informasi.

Data yang akan dimasukkan adalah sebuah sistem informasi dapat berupa formulir-formulir, prosedur-prosedur dan bentuk data lainnya. Selain itu sistem informasi dapat didefinisikan menjadi :

- a. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dari organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
- b. Sekumpulan prosedur informasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan untuk mengendalikan informasi.
- c. Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat managerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan - laporan. Dalam sistem informasi dibutuhkan beberapa komponen yang fungsinya sangat bermanfaat didalam sistem informasi. Komponen-komponen sistem informasi tersebut adalah sebagai berikut : input, proses, output, teknologi, basis data dan kendali.

### **2.3 Analisa Sistem Berorientasi Objek**

*Unified Modeling Language (UML)* adalah sebuah “bahasa” yang telah menjadi standar dalam industry 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.

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*). Tujuan utama UML diantaranya adalah untuk :

- a. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
- b. Memberikan bahasa pemodelan yang bebas dari berbagai bahasapemrograman dan proses rekayasa.
- c. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.

### **2.3.1 Pengertian Analisa Berorientasi Objek**

Menurut Ariesto hadi Sutopo (2002:242): “ Analisa sistem adalah proses menentukan kebutuhan sistem apa yang dilakukan sistem untuk memenuhi kebutuhan klien, bukanlah bagaimana sistem tersebut diimplementasikan”.

Analisa sistem adalah suatu proses untuk mengidentifikasi dan mengevaluasi permasalahan – permasalahan, penyebab – penyebab masalah, mengidentifikasi kebutuhan – kebutuhan sistem, dan memahami secara keseluruhan tentang sistem yang yang akan dikembangkan.

Keberhasilan dari tahapan analisa adalah memahami kebutuhan – kebutuhan sistem dan membuat konsep sistem baru yang menggambarkan apa yang harus dilakukan sistem guna memenuhi kebutuhan – kebutuhan sistem. Tujuan utama dari analisa berorientasi objek adalah memodelkan sistem yang nyata dengan penekanan apa yang harus dilakukan sistem.

### 2.3.2 Alat Bantu Analisis Sistem Berorientasi Objek

#### a. Activity diagram

Activity diagram adalah teknik untuk mendeskripsikan logika procedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram mempunyai pesan seperti halnya flowchart, akan tetapi perbedaannya dengan flowchart adalah activity diagram bisa mendukung perilaku paralel sedangkan flowchart tidak bisa. Simbol- simbol yang digunakan pada saat pembuatan activity diagram adalah sebagai berikut :

- 1) *Start Point*, diletakkan pada pojok kiri atas dan merupakan awal aktifitas, (Munawar 2005:109)



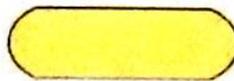
Simbol *Start Point*

- 2) *End Point*, akhir aktifitas, (Munawar 2005:109)



Simbol *End Point*

- 3) *Activity*, menggambarkan suatu proses / kegiatan bisnis (Munawar 2005:109)

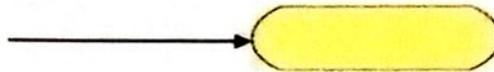


Simbol *Activity*

Jenis jenis *activities*, yaitu :

a) *Black hole activities*

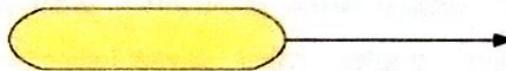
Ada masukan dan tidak ada keluaran, biasanya digunakan jika dikehendaki ada satu atau lebih transisi.



Simbol *blackhole activities*

b) *Miracle activities*

Tidak ada masukan dan ada keluaran, biasanya dipakai pada waktu start point dan dikehendaki ada 1 atau lebih transisi.



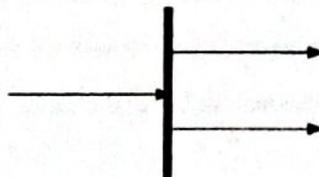
Simbol *Miracle Activities*

c) *Paralel Activities*

Suatu activity yang berjalan secara bersamaan terdiri dari :

(1) *Fork* (Percabangan)

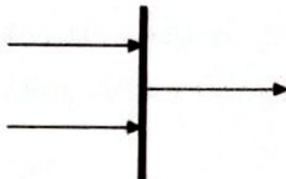
*Fork* digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu (Munawar 2005:110)



Simbol *Fork* (Percabangan)

(2) *Join* (Penggabungan)

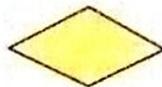
*Join* (Penggabungan) / *Rake* menunjukkan adanya rekomposisi. (Munawar 2005:110). Yaitu mempunyai 1 atau lebih transisi masuk dan hanya 1 transisi keluar dan fork harus berhubungan dengan join.



Simbol *Join* (Penggabungan)

(3) *Decision Point*

*Decision* digambarkan dengan lambang wajik atau belah ketupat. Mempunyai transisi (sebuah garis dari atau ke *decision point*). Setiap transisi yang ada harus mempunyai *guard* (kunci). Tidak ada keterangan (pernyataan) pada tengah belah ketupat seperti pada flowchart.



Simbol *Decision Point*

(4) *Guard* (kunci)

*Guard* (kunci) adalah kondisi benar sewaktu melewati transisi. Digambarkan dengan diletakkan diantara tanda [ ]. Tanda *otherwise guard* untuk menangkap suatu kondisi yang belum terdeteksi. Setiap transisi dari atau ke *decision point* harus mempunyai *guard* yang harus konsisten dan lengkap serta tidak *overlap*.

(5) *Swimlane*

*Swimlane* merupakan sebuah cara untuk mengelompokkan *activity*

berdasarkan actor. Actor bisa ditulis nama actor ataupun sekaligus dalam lambing actor. *Swimlane* digambarkan secara vertical, walaupun kadang kadang digambarkan secara horizontal.

(6) *Swimarea*

Ketika sebuah activity diagram mempunyai banyak *swimlane*, perlu dipikirkan dengan pendekatan *swimarea*. *Swimarea* mengelompokkan activity berdasarkan kegiatan didalam use case.

b. Analisa Dokumen Keluaran

Analisa keluaran adalah analisa mengenai dokumen – dokumen keluaran yang dihasilkan dari sebuah sistem.

c. Analisa Dokumen Masukan

Analisa dokumen masukan adalah bagian dari pengumpulan informasi tentang sistem yang sedang berjalan. Tujuan analisa masukan adalah memahami prosedur berjalan.

d. Use Case Diagram

*Use case diagram* menggambarkan sebuah *fungsi* yang diharapkan dari sebuah sistem dan bagaimana sistem berinteraksi dengan dunia luar yang ditekankan dalam use case diagram adalah “apa” yang diperbuat sistem, dan bukan “bagaimana” sistem itu melakukannya. Sebuah use case mempersentasikan sebuah interaksi antara actor dengan sistem use case diagram juga menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (actor). Use case merupakan sebuah pekerjaan tertentu, misalnya log in kesistem, meng-*create* sebuah daftar belanja dan sebagainya.

Secara umum *use case* diagram terdiri dari :

1) Aktor (*actor*)

*Actor* adalah sebuah peran yang bisa dimainkan oleh pengguna dalaminteraksinya dengan sistem, untuk mengidentifikasi actor harus ditentukan pembagian kerja tugas – tugas yang berkaitan dengan peran konteks target sistem actor dilukiskan dengan peran yang mereka mainkan dalam use case. Seperti staff penjualan ,pelanggan,dll.



Simbol Actor

2) Use case

*Use case* menggambarkan perilaku, termasuk didalamnya interaksi antaraactor dengan sistem. Use case dibuat berdasarkan keperluan actor, merupakan “apa” yang dikerjakan sistem bukan “bagaimana” sistem mengerjakannya. Setiap use case harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksi nya dengan actor. Nama use case boleh terdiri dari beberapa kata dan tidak boleh ada dua use case yang memiliki nama yang sama.



Simbol Use Case

3) Asosiasi (*Asociation*)

Asosiasi menggambarkan aliran data atau informasi asosiasi atau relasi juga digunakan untuk menggambarkan bagaimana actor terlibat dalam use case. Relasi (*relationship*) digambarkan sebagai bentuk garis antara dua symbol use case diagram.



Simbol *Asociation*

Ada empat jenis relasi / asosiasi yang dapat timbul pada use case diagram yaitu :

a) Asosiasi antara actor dan use case.

Ujung panah pada association antara actor dan use case mengindikasikan siapa/apa yang meminta interaksi dan bukannya mengindikasikan aliran data. Sebaiknya gunakan garis tanpa panah untuk association antara actor dan use case, association antara actor dan use case yang menggunakan panah untuk mengindikasikan bila actor berinteraksi secara pasif dengan system.

---

Simbol Asosiasi antara actor dan use case

b) Asosiasi antara use case

(1) *Include*

*Include* menggambarkan suatu use case termasuk didalam use case lain(diharuskan). Contohnya adalah pemanggilan sebuah fungsi program. Digambarkan dengan garis lurus berpanah dengan tulisan.

`<<include>>`

(2) *Extend*

*Extend* digunakan ketika hendak menggambarkan variasi pada kondisi perilaku normal dan menggunakan lebih banyak control form dan mendeklarasikan extension pada use case utama atau dengan kata lain adalah perluasan dari use case lain jika syarat atau kondisi terpenuhi. Digambarkan dengan garis lurus berpanah dengan tulisan.

`<<extend>`

(3) *Generalization / Inheritance* antar use case (Pewarisan kerja).

*Generalization / Inheritance* antar use case dipakai ketika ada sebuah perlakuan khusus (single condition) dan merupakan pola hubungan base-parent use case. Digambarkan dengan *generalization / inheritance* antar use case secara vertical dengan *inheriting use case* dibawah *base / parent use case*

(4) *Generalization / Inheritance* antar actor (Pewarisan sifat actor).

*Generalization / Inheritance* antar actor dibuat ketika ada sebuah actor terbentuk dan mempunyai atribut dan metode yang sama dengan actor yang sudah ada.

e. Deskripsi Use Case Diagram

Deskripsi Use Case Diagram adalah abstraksi dari interaksi antar sistem dan actor. Oleh karena itu sangat penting untuk memilih abstraksi yang cocok. Sebagai contoh, saat pelanggan menelpon restoran untuk melakukan booking, dia akan berbicara kepada karyawan restoran yang akan mencatat booking tersebut ke sistem. Untuk melakukan hal tersebut, karyawan akan menjalankan peran sebagai resepsionis meskipun pekerjaan tersebut mungkin bukan pekerjaan formalnya. Pada situasi ini, karyawan merupakan instance dari actor resepsionis dan interaksi diantara karyawan dengan sistem adalah instance dari use case. Use case dibuat berdasarkan keperluan actor. Use case harus merupakan 'apa' yang dikerjakan software aplikasi, bukan 'bagaimana' software aplikasi mengerjakannya. Setiap use case harus diberi nama yang menyatakan apa hal yang ingin dicapai dari hasil interaksinya dengan actor. Nama use case boleh terdiri dari beberapa kata dan tidak boleh ada dua use case yang memiliki nama yang sama.

## 2.4 Perancangan Sistem Berorientasi Objek

Perancangan berorientasi objek merupakan tahap lanjutan setelah analisa berorientasi objek, perancangan berorientasi objek adalah suatu pendekatan yang digunakan untuk menspesifikasi kebutuhan – kebutuhan sistem dengan menkolaborasikan objek – objek, atribut – atribut dan method – method yang ada (Whitten 2008:686).

Tujuan perancangan sistem itu untuk memahami kebutuhan kepada pemakai sistem ( user ) dan memberikan gambaran yang jelas serta rancang bangun yang lengkap.

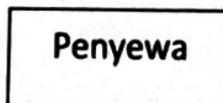
Tahap – tahap yang dilakukan pada perancangan berorientasi objek, adalah sebagai berikut :

### a. *Entity Relationship Diagram (ERD)*

*ERD* adalah sebuah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam hal entitas dan relasi yang digambarkan oleh data tersebut (Whitten 2004:295). Elemen – elemen ERD yaitu sebagai berikut :

#### 1) *Entity (Entitas)*

Sesuatu (obyek) yang ada didalam sistem. Entitas merupakan kata benda yang dikelompokkan menjadi empat jenis nama, yaitu : orang, benda, lokasi dan kejadian. Entitas disimbolkan dengan persegi panjang.

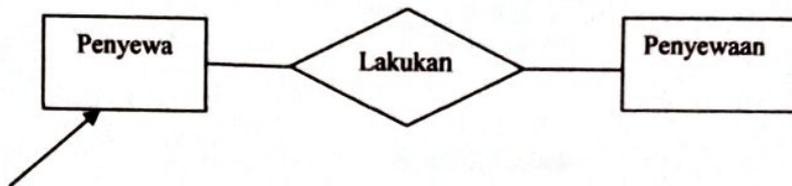


Simbol Entitas

Jenis – jenis entitas antara lain :

a) *Strong Entity Set*

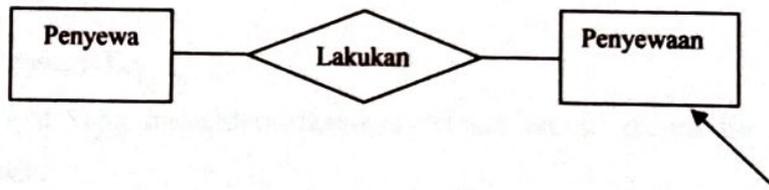
*Strong entity set* yaitu entity set satu atau banyak atributnya digunakan oleh entitas lain.



Strong Entity

b) *Weak Entity Set*

*Weak Entity Set* yaitu entity set yang tidak memiliki atribut yang dapat dijadikan kunci, sehingga membutuhkan atribut dari entitas lain.



Week Entity

## 2) *Relationship* ( Hubungan atau relasi)

Simbol *Relationship* pada ERD digambarkan dengan *diamond* atau *decision*. Jika satu entitas dihubungkan dengan *Relationship*, maka digambarkan dengan garis lurus. Kumpulan dari *Relationship* yang sejenis disebut *Relationship set*.



Simbol Relasi

## 3) *Attribute* (atribut)

Suatu deskripsi karakteristik dari entitas (Whitten 2004:296). Atribut juga merupakan karakteristik dari *relationship*, maksudnya sesuatu yang menjelaskan apa yang sebenarnya yang dimaksud dengan entitas maupun *relationship*. Atribut disimbolkan dengan sebuah *ellips*. Dari setiap atribut entitas terdapat satu atribut yang dijadikan sebuah *key* (kunci). Beberapa jenis *key*, yaitu :

### a) *Primary Key*

Field yang mengidentifikasi sebuah record dalam file dan bersifat unik.

### b) *Secondary Key*

Field yang mengidentifikasi sebuah record dalam file dan tidak bersifat unik.

### c) *Candidate key*

Beberapa field yang dapat dijadikan calon *primary key*.

### d) *Alternative Key*

Field dari *candidate key* yang tidak terpilih jadi *primary key*.

### e) *Composite Key*

Beberapa field yang digabungkan untuk membentuk *primary key*.

f) *Foreign Key*

Field yang bukan key, tetapi merupakan key pada file lain.

4) *Cardinality (Kardinaliti)*

Ada 3 kemungkinan hubungan yang ada yaitu :



a) *One To One*

Artinya tingkat hubungan dimana satu kejadian pada entitas yang pertama hanya mempunyai satu hubungan dengan satu kejadian pada entitas kedua, demikian juga sebaliknya.



b) *One To Many*

Artinya tingkat hubungan dimana satu kejadian pada entitas pertama mempunyai banyak hubungan dengan kejadian pada entitas kedua, demikian juga sebaliknya.



c) *Many To Many*

Artinya tingkat hubungan dimana tiap kejadian sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya.



**b. Logical Record Structure (LRS)**

Sebuah model sistem yang digambarkan dengan sebuah Diagram-ER akan mengikuti pola aturan pemodelan tertentu. Dalam kaitannya dengan konversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan – aturan berikut ini :

- 1) Setiap entitas akan diubah ke bentuk kotak.
- 2) Sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada diagram-ER 1:M (relasi bersatu dengan cardinality M) atau tingkast hubungan 1:1 (relasi bersatu dengan cardinality yang paling membutuhkan referensi),sebuah relasi dipisah dalam sebuah kotak tersendiri (menjadi entitas baru) jika tingkat hubungannya M:M (many to many) dan memiliki *foreign key* sebagai primary key yang diambil dari kedua entitas yang sebelumnya saling berhubungan.

**c. Tabel**

Tabel adalah koleksi objek yang terdiri dari sekumpulan elemen yang diorganisasi secara kontinyu, artinya memori yang dialokasi antara satu elemen dengan elemen yang lainnya mempunyai address yang berurutan. Pada table, pengertian perlu dipahami adalah :

- 1) Keseluruhan tabel (sebagai koleksi) adalah container yang menampung seluruh elemen.
- 2) Indek table,yang menunjukkan address dari sebuah elemen.
- 3) Element tabel yang dapat dipacu melalui indeknya,bertipe tertentu yang sudah terdefinisi.

Seluruh elemen tabel bertipe “sama”. Dengan catatan: beberapa bahasa pemrograman memungkinkan pendefinisian table dengan elemen genetik, tapi pada saat diinstansiasi harus dengan tipe yang sama.

#### **d. Spesifikasi Basis Data**

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lain dan tersimpan diluar computer serta digunakan perangkat lunak(*software*) tertentu untuk memanipulasinya. Sedangkan sistem berbasis data adalah suatu sistem penyusunan dan pengelolaan *record-record* dengan menggunakan computer dengan tujuan untuk menyimpan atau merekam serta melihat data operasional lengkap pada sebuah organisasi, sehingga mampu menyediakan informasi yang diperlukan untuk kepentingan proses pengambilan keputusan.

### **2.5 Sistem Informasi Persewaan Mobil**

Dalam kegiatan suatu perusahaan, baik itu perusahaan kecil menengah maupun perusahaan besar. Pada umumnya memerlukan sistem informasi penyewaan mobil yang efisien dan efektif. Penyewaan mobil merupakan salah satu cara untuk mendukung proses bisnis dari suatu perusahaan khususnya perusahaan yang bergerak di bidang jasa transportasi. Secara klasik penyewaan mobil merupakan proses permintaan (*requisition*), pengelompokkan (*classifying*), order pembelian (*purchase order*), penerimaan (*receiving*), dan pelaporan (*reporting*) dari kegiatan penyewaan mobil.