

## BAB II LANDASAN TEORI

### 2.1 Konsep Sistem Informasi

#### 2.1.1 Sistem

Sistem adalah sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan bersama dengan *input* serta menghasilkan *output* dalam proses transformasi yang teratur [2]. Sistem adalah kelompok dari dua atau lebih komponen atau subsistem yang saling berhubungan yang berfungsi dengan tujuan yang sama [3].

Jadi, dengan kata lain istilah sistem mengandung arti himpunan bagian atau komponen yang saling berhubungan secara teratur dan merupakan satu kesatuan untuk mencapai tujuan.

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, seperti [3]:

1. Komponen-Komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa subsistem atau bagian dari sistem.

2. Batasan Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas suatu sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara, sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak, maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung (*Interface*)

Penghubung merupakan media penghubung antara satu subsistem dengan yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran dari satu sistem akan menjadi masukan untuk subsistem yang lainnya melalui penghubung.

5. Masukan (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*Maintenance Input*) dan masukan sinyal. *Maintenance Input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapat keluaran.

6. Keluaran (*Output*)

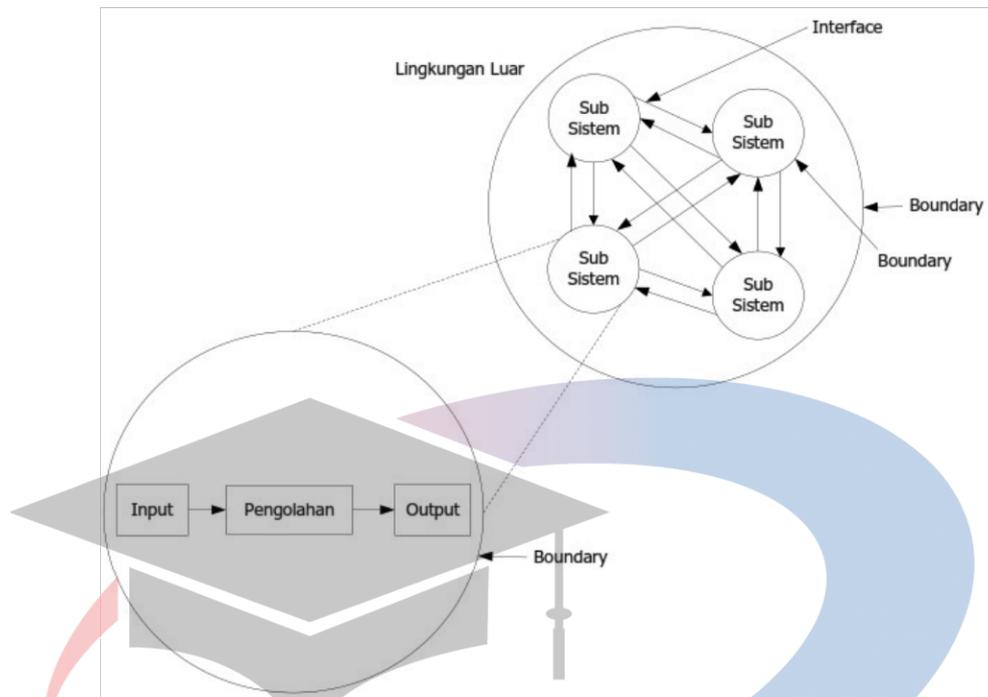
Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem. Keluaran biasanya berbentuk laporan-laporan yang formatnya disesuaikan dengan kebutuhan pemakai laporan tersebut.

7. Pengolah (*Process*)

Suatu sistem mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran. Pengolah dapat berupa suatu urutan-urutan tertentu yang akan mengubah data menjadi bentuk lainnya yang dapat dimanfaatkan. Dalam proses tersebut mungkin digunakan rumus-rumus tertentu, kondisi-kondisi tertentu dalam melakukan seleksi data, atau standar-standar yang akan menjadikan data menjadi informasi yang sesuai dengan standar yang diinginkan.

8. Sasaran (*Objectives*)

Suatu sistem pasti mempunyai tujuan dan sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



Gambar 2.1 Karakteristik Sistem

### 2.1.2 Informasi

Informasi adalah data yang telah diproses atau diorganisasi ulang menjadi bentuk yang berarti [4]. Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerima dan bermanfaat dalam pengambilan keputusan saat ini dan masa mendatang [5].

Berdasarkan pengertian informasi di atas, maka disimpulkan bahwa informasi adalah hasil pemrosesan dari sistem informasi dan mempunyai manfaat pada masa sekarang maupun masa yang akan datang.

Informasi dapat dikelompokkan menjadi 3 (tiga) bagian, yaitu [5]:

1. Informasi Strategis

Informasi ini digunakan untuk mengambil keputusan jangka panjang, seperti informasi eksternal, rencana perluasan perusahaan, dan lainnya.

2. Informasi Taktis

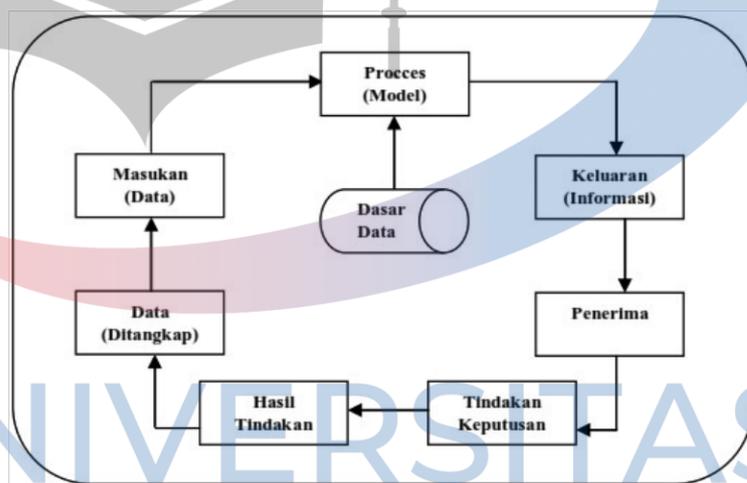
Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.

### 3. Informasi Teknis

Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stok, retur penjualan, dan laporan kas harian.

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data tersebut harus melalui beberapa pengolahan dengan menggunakan suatu model tertentu agar dihasilkan suatu informasi yang berguna [6].

Siklus informasi menggambarkan pengolahan data menjadi informasi dan pemakaian informasi untuk pengambilan keputusan, hingga akhirnya dari tindakan hasil pengambilan keputusan tersebut dihasilkan data kembali. Data tersebut akan ditangkap sebagai *input* [6].



Gambar 2.2 Siklus Informasi

### 2.1.3 Sistem Informasi

Sistem informasi dapat didefinisikan sebagai orang, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan sebagai *output* informasi yang diperlukan untuk mendukung sebuah organisasi [4].

Sistem informasi adalah suatu sistem yang melibatkan manusia dalam pengumpulan data, serta proses pemeliharaan dan pengolahan data tersebut sebagai informasi, dan teknologi untuk menopang dan meningkatkan pengoperasian bisnis sekaligus memberikan solusi terhadap permasalahan dan memberikan keputusan yang diperlukan manajemen dan pengguna [7].

Dari pengertian sebelumnya dapat disimpulkan bahwa sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja) atau sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan.

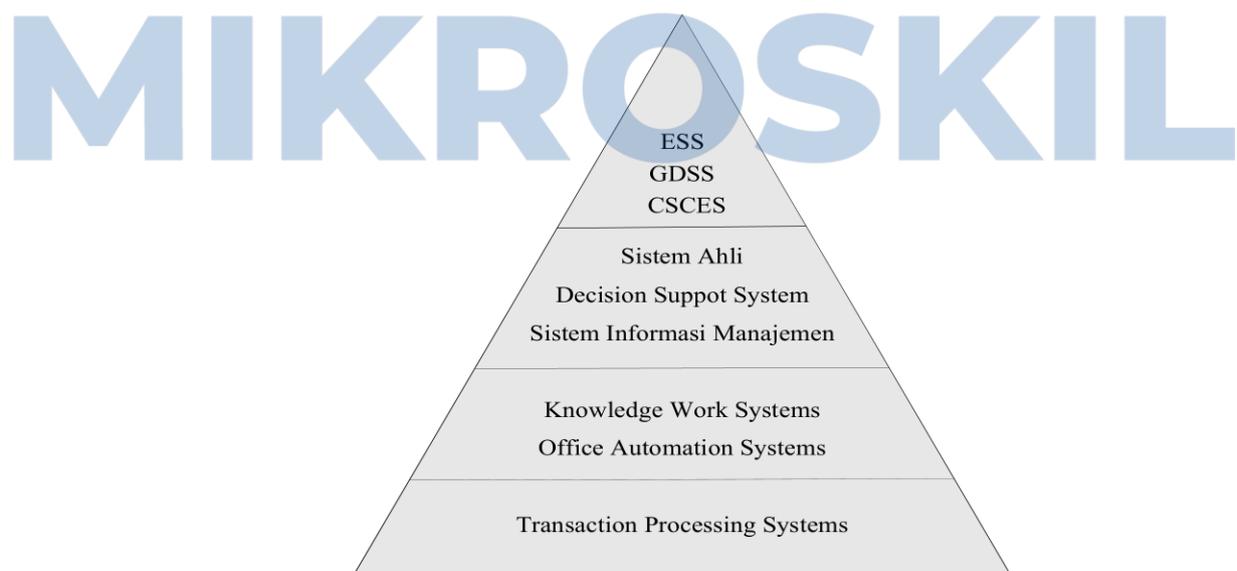
Sistem informasi dapat dikelompokkan menjadi beberapa bagian, yaitu [7]:

1. *Transaction Processing Systems* (TPS), yaitu sistem informasi yang terkomputerisasi yang dikembangkan untuk memperoleh data dalam jumlah besar untuk transaksi bisnis rutin, seperti daftar gaji dan inventarisasi.
2. *Office Automation System* (OAS), yaitu *Knowledge Work System* (KWS) yang bekerja pada level *knowledge*. OAS mendukung pekerja data, yang biasanya tidak menciptakan pengetahuan baru, melainkan hanya menganalisis informasi sedemikian rupa untuk mentransformasikan data atau memanipulasikannya dengan cara-cara tertentu sebelum menyebarkannya secara keseluruhan dengan organisasi dan kadang-kadang di luar organisasi. KWS mendukung para pekerja profesional seperti ilmuwan, insinyur, dan dokter dengan membantu menciptakan pengetahuan baru dan memungkinkan mereka mengkontribusikannya ke organisasi atau masyarakat.
3. Sistem Informasi Manajemen (SIM), yang mendukung spektrum tugas-tugas organisasional yang lebih luas dari TPS, termasuk analisis keputusan dan pembuatan keputusan. SIM menghasilkan informasi yang digunakan untuk membuat keputusan, dan juga dapat membantu menyatukan beberapa fungsi informasi bisnis yang sudah terkomputerisasi (basis data).
4. *Decision Support System* (DSS), yang hampir sama dengan SIM karena menggunakan basis data sebagai sumber data. DSS bermula dari SIM karena menekankan pada fungsi mendukung pembuat keputusan di seluruh tahap-tahapnya, meskipun keputusan aktual tetap wewenang eksklusif pembuat keputusan.
5. Sistem Ahli (ES) dan Kecerdasan Buatan (AI), dimana AI yang dimaksudkan untuk mengembangkan mesin-mesin yang berfungsi secara cerdas. Sedangkan, sistem ahli menggunakan pendekatan-pendekatan pemikiran AI untuk menyelesaikan masalah serta memberikannya lewat pengguna bisnis. Sistem ahli (disebut juga *knowledge-based systems*) secara efektif menangkap dan

menggunakan pengetahuan seorang ahli untuk menyelesaikan masalah yang dialami dalam suatu organisasi. Berbeda dengan DSS, ES meninggalkan keputusan terakhir bagi pembuat keputusan, sedangkan sistem ahli menyelesaikan solusi terbaik terhadap suatu masalah khusus. Komponen dasar sistem ahli adalah *knowledge-based*, yakni suatu mesin inferensi yang menghubungkan pengguna dengan sistem melalui pengolahan pertanyaan lewat bahasa terstruktur dan antarmuka pengguna.

6. *Group Decision Support System (GDSS)* dan *Computer-Support Collaborative Work System (CSCW)*, dimana GDSS dimaksudkan untuk membawa kelompok bersama-sama menyelesaikan masalah dengan memberi bantuan dalam bentuk pendapat, kuesioner, konsultasi, dan skenario. Kadang-kadang GDSS disebut dengan CSCW yang mencakup pendukung perangkat lunak yang disebut dengan “*groupware*” untuk kolaborasi tim melalui komputer yang terhubung dengan jaringan.
7. *Executive Support System (ESS)* yang tergantung pada informasi yang dihasilkan TPS dan SIM, serta ESS membantu eksekutif mengatur interaksinya dengan lingkungan eksternal dengan menyediakan grafik-grafik dan pendukung komunikasi di tempat-tempat yang bisa diakses seperti kantor.

Ilustrasi pengelompokan sistem informasi dapat dilihat pada gambar berikut ini [7].



Gambar 2.3 Pengelompokan Jenis Sistem Informasi

## 2.2 Proses Pengembangan Basis Data

Dalam organisasi, banyak pengembangan basis data dimulai dengan pemodelan perusahaan, yang didirikan dari jangkauan dan isi umum basis data organisasi. Tujuannya adalah untuk menciptakan gambaran keseluruhan atau penjelasan dari basis data organisasi, bukan desain untuk basis data tertentu. Basis data tertentu menyediakan data untuk satu atau sistem informasi lebih lanjut, sedangkan model data perusahaan, yang banyak mencakup basis data, menggambarkan ruang lingkup data yang dikelola oleh organisasi. Dalam pemodelan data perusahaan, dengan meninjau sistem saat ini, menganalisis sifat dari area bisnis harus didukung, menggambarkan data yang dibutuhkan pada tingkat yang sangat tinggi dari abstraksi, dan merencanakan satu atau proyek pengembangan basis data yang lebih [8].

Selain seperti penggambaran grafis dari jenis entitas, model data perusahaan secara menyeluruh juga akan mencakup deskripsi berorientasi bisnis dari setiap jenis entitas dan ringkasan dari berbagai pernyataan tentang bagaimana bisnis beroperasi, disebut aturan bisnis, yang mengatur validitas data. Hubungan antara objek bisnis (fungsi bisnis, unit, aplikasi, dan sebagainya) dan data sering ditangkap menggunakan matriks dan melengkapi informasi yang ditangkap dalam model data perusahaan [8].

Tabel 2.1 Contoh Matriks *Business Function-to-Data Entity*

<i>Data Entity Types</i>	<i>Customers</i>	<i>Product</i>	<i>Raw Material</i>	<i>Order</i>	<i>Work Center</i>	<i>Work Order</i>	<i>Invoice</i>	<i>Equipment</i>	<i>Employee</i>
<i>Business Planning</i>	X	X						X	X
<i>Product Development</i>		X	X		X			X	
<i>Materials Management</i>		X	X	X	X	X		X	
<i>Order Fulfillment</i>	X	X	X	X	X	X	X	X	X
<i>Order Shipment</i>	X	X		X	X		X		X
<i>Sales Summarization</i>	X	X		X			X		X
<i>Production Operations</i>		X	X	X	X	X		X	X
<i>Finance and Accounting</i>	X	X	X	X	X		X	X	X

X = data entity is used within business function

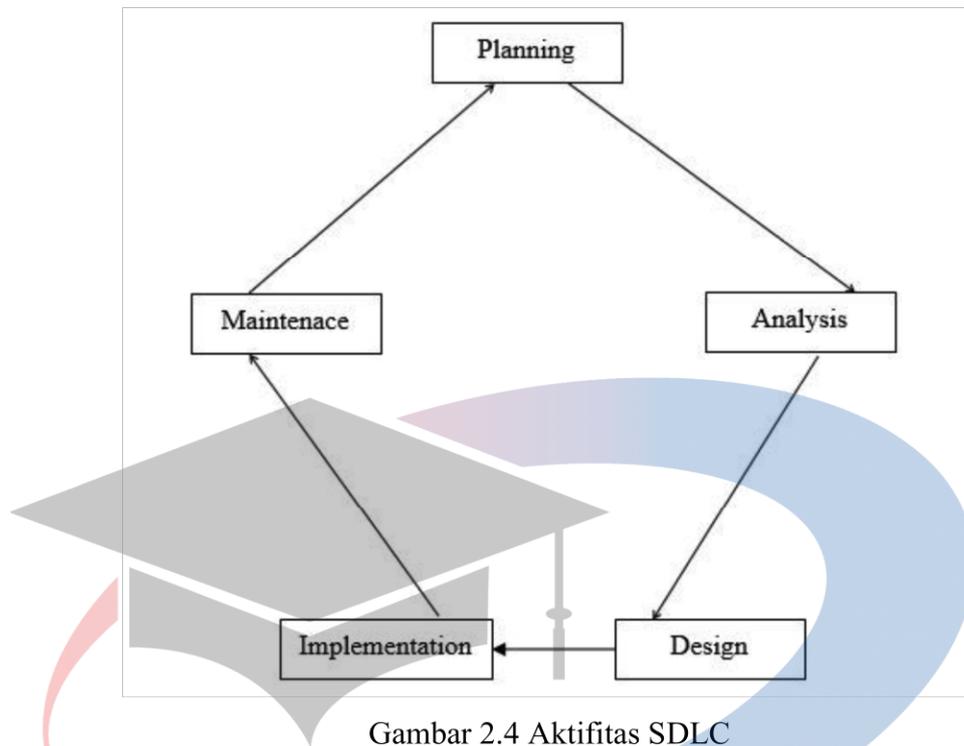
Pemodelan data perusahaan sebagai komponen dari pendekatan *top-down* untuk perencanaan dan pengembangan sistem informasi merupakan salah satu sumber

proyek basis data baru untuk memenuhi tujuan strategis organisasi, seperti peningkatan dukungan pelanggan, produksi, dan manajemen persediaan yang lebih baik, atau lebih akurat dalam peramalan penjualan. Banyak proses basis data yang meningkat, namun terlebih dalam model *bottom-up*. Dalam hal ini, proyek yang diminta oleh pengguna sistem informasi yang membutuhkan informasi tertentu untuk melakukan pekerjaan mereka, atau dengan sifat profesional lainnya, yang menentukan kebutuhan untuk meningkatkan pengelolaan data dalam organisasi [4].

Sifat proyek pengembangan basis data *bottom-up* biasanya berfokus pada penciptaan satu basis data. Beberapa proyek basis data hanya berkonsentrasi pada mendefinisikan, merancang, dan menerapkan basis data sebagai dasar untuk pengembangan sistem informasi selanjutnya. Dalam beberapa kasus, bagaimana fungsi pengelolaan sistem informasi yang terkait dikembangkan bersama-sama sebagai bagian dari proyek pengembangan sistem informasi yang komprehensif [4].

Siklus hidup pengembangan sistem (SDLC) adalah sistem informasi tim lengkap profesional, termasuk basis data yang diikuti organisasi untuk menentukan, mengembangkan, memelihara, dan mengganti sistem. Berbagai langkah dalam SDLC dan gambar yang terkait proses untuk menjadi melingkar, menyampaikan sifat berulang dari proyek pengembangan sistem. Waktu mereka dapat dilakukan secara paralel, dan mungkin untuk saat mengambil keputusan perlu dipertimbangkan kembali. Beberapa percaya melalui proses pembangunan untuk siklus pada tingkat yang lebih rinci pada setiap tahapan sebagai persyaratan yang lebih konkret [8].

Adapun gambar aktivitas SDLC dan pengertiannya sebagai berikut [8].



Gambar 2.4 Aktifitas SDLC

### 1. Perencanaan (*Planning*)-Pemodelan Perusahaan

Pemodelan proses pengembangan basis data dimulai dengan ulasan dari komponen-komponen pemodelan perusahaan yang dikembangkan selama proses perencanaan sistem. Selama langkah ini, analis meninjau basis data saat ini dan sistem informasi, menganalisis sifat dari area bisnis yang menjadi subjek dari proyek pembangunan, dan menjelaskan secara umum data yang diperlukan untuk setiap pembentukan sistem, dipertimbangkan untuk pengembangan. Mereka menentukan data apa yang sudah tersedia di basis data yang ada dan apa data baru yang harus ditambahkan ke proyek baru yang diusulkan. Proyek yang dipilih pindah ke tahap berikutnya berdasarkan nilai proyeksi setiap proyek untuk organisasi.

### 2. Perencanaan (*Planning*)-Pemodelan Data Konseptual

Untuk proyek sistem informasi yang diinisiasi, persyaratan data keseluruhan dari sistem informasi yang diusulkan harus dianalisis. Hal ini dilakukan dalam dua tahap. Pertama, selama fase perencanaan, analis mengembangkan diagram serta dokumentasi lainnya untuk menguraikan ruang lingkup data yang terlibat dalam proyek pembangunan khusus ini tanpa pertimbangan basis data yang sudah ada. Hanya kategori tingkat tinggi data (entitas) dan hubungan utama termasuk pada

saat ini. Langkah dalam SDLC ini sangat penting untuk meningkatkan peluang keberhasilan pengembangan kebutuhan spesifikasi organisasi, semakin dekat model konseptual harus datang untuk memenuhi kebutuhan organisasi dan daur ulang harus diperlukan.

### 3. Analisis (*Analysis*)-Pemodelan Data Konseptual

Selama fase SDLC, analisis menghasilkan model data rinci yang mengidentifikasi semua data organisasi yang harus dikelola untuk sistem informasi ini. Setiap data atribut didefinisikan, semua kategori data yang tercantum, setiap hubungan bisnis antara entitas data diwakili dan setiap aturan yang mendikte integritas data ditentukan. Hal ini juga diperiksa selama fase analisis model data konseptual dan diperiksa konsistensinya dengan jenis lain dari model yang dikembangkan untuk menjelaskan dimensi lain dari target sistem informasi, seperti langkah-langkah pengolahan, aturan untuk penanganan data dan waktu kejadian. Namun, model data konseptual awal ini, karena kegiatan selanjutnya dari SDLC mungkin menemukan elemen yang hilang atau kesalahan ketika merancang khusus transaksi, laporan, pameran, dan pertanyaan. Dengan pengalaman, pengembang basis data memperoleh model mentah dari fungsi bisnis yang umum, seperti penjualan atau merekam keuangan, tetapi harus tetap waspada untuk pengecualian yang diikuti oleh sebuah organisasi. *Output* dari fase pemodelan konseptual adalah skema konseptual.

### 4. Desain (*Design*)-Desain Basis Data Logis

Desain basis data logis pendekatan pengembangan ada dua perspektif. Pertama, skema konseptual harus diubah menjadi skema logis, yang menggambarkan teknologi manajemen data yang akan digunakan untuk mengimplementasikan basis data. Misalnya, jika teknologi akan digunakan, maka data konseptual ditransformasikan dan meliputi tabel, kolom, baris, kunci primer, dan kunci asing, serta bagaimana melakukan proses penting ini dalam representasi disebut skema logis.

### 5. Implementasi (*Implementation*)-Implementasi Basis Data

Di tahapan implementasi basis data, desainer menulis, menguji, dan menginstalasi program, membuat, atau memodifikasi basis data. Perancang mungkin melakukan hal ini dengan menggunakan bahasa pemrograman standar atau dalam bahasa

pengolahan basis data (misalnya SQL), atau menggunakan tujuan khusus bahasa nonprosedural untuk menghasilkan laporan, mungkin termasuk grafik, juga selama pelaksanaan, desainer akan menyelesaikan semua dokumentasi basis data, melatih pengguna, dan menempatkan prosedur ke tempat untuk dukungan yang berkelanjutan dari sistem informasi pengguna. Langkah terakhir adalah membuat data dari sumber informasi yang ada (*file* dan basis data dari aplikasi *legacy* ditambah data baru yang diperlukan sekarang), memuat *file* dan basis data yang ada ke dalam format netral (seperti biner atau teks *file*), dan kemudian memuat data ini ke dalam basis data baru. Akhirnya, basis data dan aplikasi terkait dimasukkan ke dalam produksi untuk pemeliharaan data dan pengambilan oleh pengguna yang sebenarnya. Selama produksi, basis data harus berkala didukung dan pulih dalam kasus pencemaran atau perusakan.

#### 6. Pemeliharaan (*Maintenance*)-Pemeliharaan Basis Data

Basis data berevolusi selama perawatan. Pada tahap ini dilakukan penambahan desain, menghapus atau mengubah karakteristik pada struktur di basis data dalam hal menemukan kondisi perubahan bisnis, serta memperbaiki kegagalan pada desain basis data, atau memperbaiki kecepatan memproses pada aplikasi basis data. Perancang mungkin juga butuh membangun ulang sebuah basis data jika terkontaminasi atau rusak oleh sebuah program atau kerusakan semua sistem ini adalah langkah panjang dalam pengembangan basis data, karena ini adalah akhir dari basis data dan aplikasi asosiasinya. Saat evaluasi basis data akan dilihat sebagai mempersingkat proses pengembangan basis data dalam model data konseptual, logikal, dan fisikal rancangan basis data, serta pelaksanaan basis data untuk menyepakati dengan perubahan yang telah diusulkan.

### 2.3 Basis Data

Basis data tidak hanya merupakan kumpulan *file*. Lebih dari itu, basis data adalah pusat sumber data yang caranya dipakai oleh banyak pemakai untuk berbagai aplikasi. Inti dari basis data adalah *Database Management System (DBMS)*, yang membolehkan pembuatan, modifikasi, dan pembaharuan data, mendapatkan kembali data, dan membangkitkan laporan. DBMS adalah perantara *user* dengan basis data. Untuk dapat berinteraksi dengan DBMS dapat memakai bahasa basis data yang sudah

ditentukan oleh perusahaan DBMS. Bahasa basis data umumnya terdiri dari berbagai macam instruksi yang diformulasikan sehingga instruksi tersebut dapat diproses oleh DBMS [9].

Perintah atau instruksi tersebut umumnya ditentukan oleh *user*. Adapun bahasa yang digunakan dibagi ke dalam 2 (dua) macam, yaitu [9]:

1. DDL (*Data Definition Language*)

Pertama adalah bahasa DDL atau kepanjangannya *Data Definition Language*, yaitu dipakai untuk menggambarkan desain dari basis data secara menyeluruh. DDL dapat dipakai untuk membuat tabel baru, memuat indeks, maupun mengubah tabel. Hasil dari kompilasi DDL akan disimpan di kamus data.

2. DML (*Data Manipulation Language*)

Kedua adalah DML atau kepanjangannya *Data Manipulation Language*, yaitu dipakai untuk memanipulasi dan mengambil data pada suatu basis data, misalnya seperti penambahan data yang baru ke dalam suatu basis data, menghapus data pada suatu basis data, dan mengubah data pada suatu basis data.

Adapun tujuan basis data yang efektif yaitu [9]:

1. Memastikan bahwa data dapat dipakai di antara pemakai untuk berbagai aplikasi.
2. Memelihara data baik keakuratan maupun kekonsistennannya.
3. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan dengan cepat.
4. Membolehkan basis data untuk berkembang dan kebutuhan pemakai untuk berkembang.
5. Membolehkan pemakai untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik.

Ketika pemakai memerlukan data khusus, basis data yang dirancang dengan baik (*well designed*) memenuhi lebih dahulu kebutuhan data yang demikian atau mungkin telah digunakan aplikasi yang lain. Akibatnya, data memiliki kesempatan tersedia yang lebih baik dalam basis data daripada dalam sistem *file* yang konvensional. Basis data yang dirancang dengan baik juga lebih fleksibel daripada *file* terpisah. Oleh karena itu, basis data dapat berkembang, seperti pada perubahan kebutuhan pemakai dan aplikasinya [9].

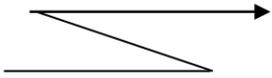
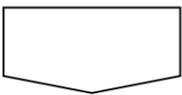
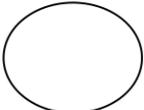
## 2.4 Flow Of Document (FOD)

Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Adapun pedoman untuk pengembangannya adalah [10]:

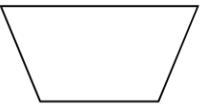
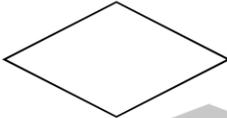
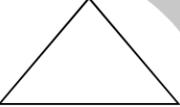
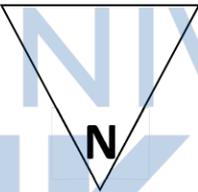
1. Sebaiknya digambarkan dari atas ke bawah dan dimulai dari bagian kiri suatu halaman.
2. Kejadiannya harus ditunjukkan dengan jelas.
3. Ditunjukkan dengan jelas dimulai dan berakhirnya suatu kegiatan.
4. Masing-masing kegiatan sebaiknya menggunakan suatu kata yang mewakili suatu pekerjaan.
5. Kejadiannya sudah dalam urutan yang benar.
6. Kegiatan yang terpotong dan akan disambung ditunjukkan dengan jelas oleh simbol penghubung.
7. Menggunakan simbol-simbol yang standar.

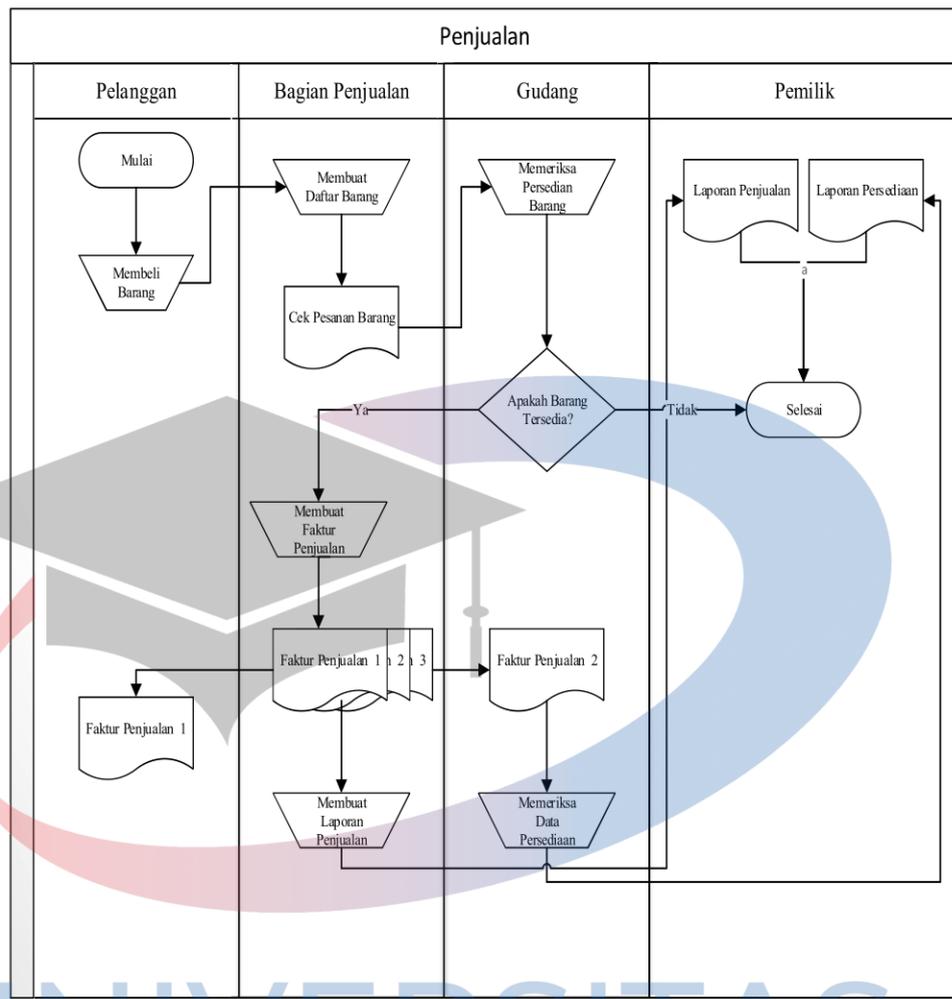
Bagan alir dokumen (*document flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) merupakan bagan alir yang menunjukkan arus dari laporan dan formulir, termasuk tembusan-tembusannya. Adapun simbol-simbolnya dapat dilihat pada tabel berikut ini [10].

Tabel 2.2 Simbol-Simbol *Flow Of Document*

Simbol	Keterangan
	Menghubungkan antara simbol yang satu dengan simbol lainnya
	<i>Communication link</i> : Pengiriman data dari satu lokasi ke lokasi lainnya melalui jalur komunikasi
	Keluar/masuk prosedur atau proses dalam lembar/halaman yang lain
	Keluar/masuk prosedur atau proses dalam lembar/halaman yang sama

Gambar berikut ini adalah contoh penggambaran bagan alir dokumen.

	Menunjukkan pengolahan yang tidak dilakukan oleh komputer
	Menunjukkan pengolahan yang dilakukan komputer
	Kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi
	Permulaan atau akhir dari suatu program
	Menunjukkan bahwa data di dalam simbol ini akan disimpan secara sementara
	Menunjukkan bahwa data di dalam simbol ini akan disimpan permanen
	<i>Input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas
	<i>File</i> dokumen secara manual disimpan dan ditarik kembali, huruf yang ditulis di dalam simbol menunjukkan urutan pengurutan <i>file</i> secara N = numeris, A = Alfabetis, D = Berdasarkan tanggal



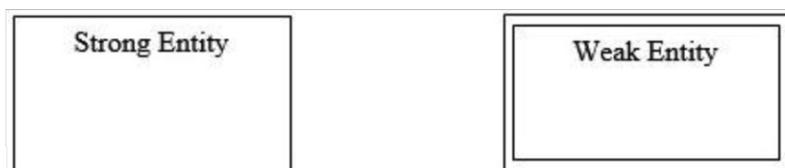
Gambar 2.5 Contoh *Flow of Document*

## 2.5 Diagram ER

Penggambaran diagram ER harus memiliki notasi-notasi berikut [4]:

### 1. Entitas (*Entity*)

Merupakan kelas dari orang, tempat, objek, peristiwa, atau konsep yang perlu ditangkap dan disimpan datanya. Nama dari entitas ditulis dengan kata benda. Entitas dapat dikategorikan menjadi entitas kuat (*Strong Entity*) dan entitas lemah (*Weak Entity*).



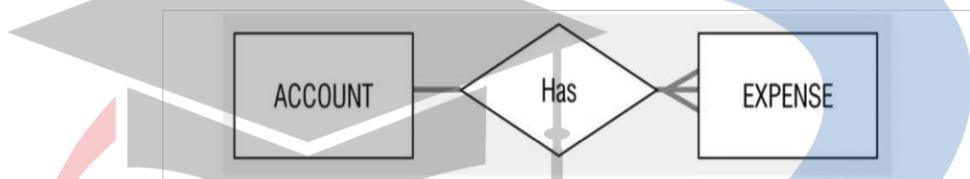
Gambar 2.6 Simbol Dasar Entitas

Tabel berikut ini menjelaskan perbedaan antara entitas kuat dan entitas lemah.

Tabel 2.3 Perbedaan Entitas Kuat dan Entitas Lemah

Entitas Kuat	Entitas Lemah
Tidak bergantung pada tipe entitas yang lain	Bergantung pada <i>strong entities</i>
Memiliki pengenalan ( <i>identifier</i> ) yang unik	Tidak memiliki pengenalan ( <i>identifier</i> ) yang unik
Digambarkan dalam bentuk segi empat	Digambarkan dalam bentuk segi empat ganda

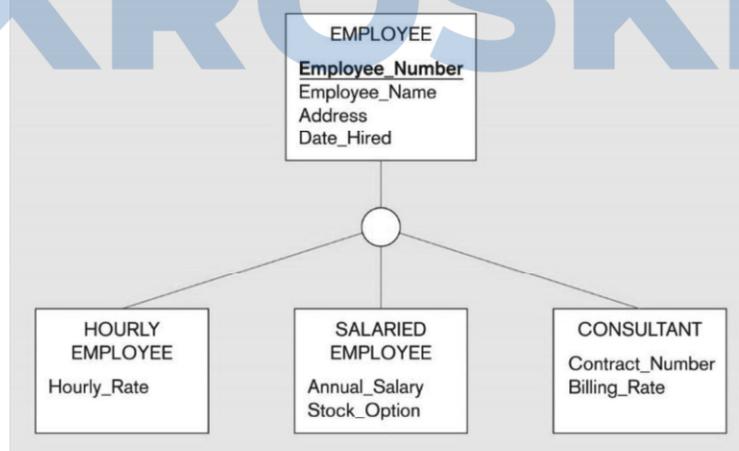
Adapun contoh dari entitas adalah sebagai berikut:



Gambar 2.7 Contoh Entitas

## 2. Atribut (*Attribute*)

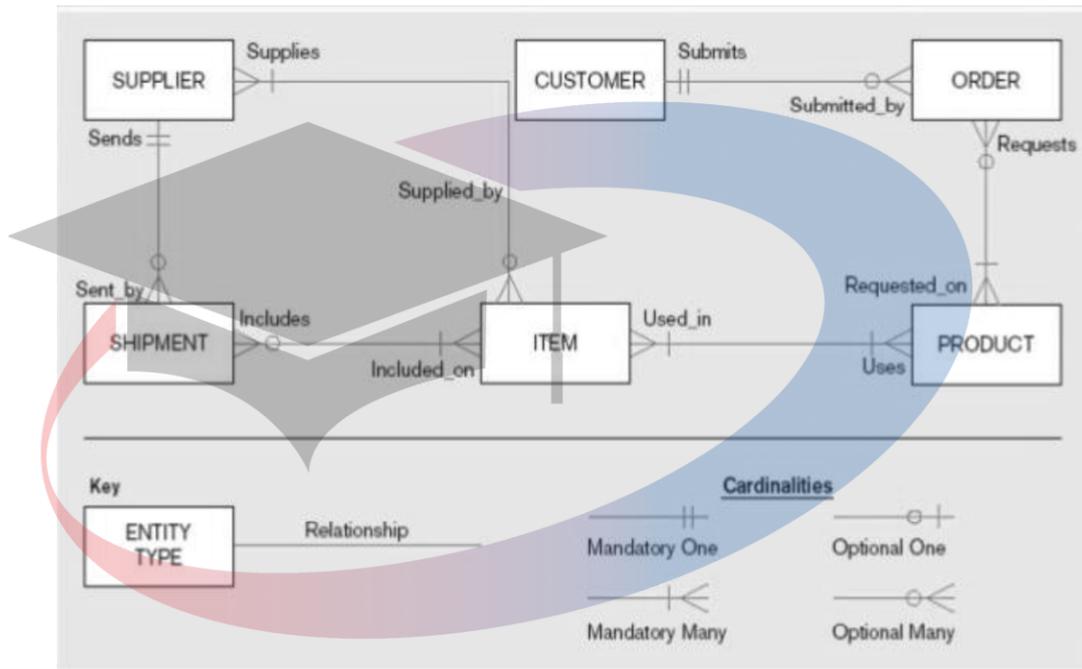
Atribut adalah properti atau karakteristik deskriptif dari suatu entitas. Seorang mahasiswa memiliki atribut-atribut seperti: *last name*, *first name*, *middle name*, *address*, dan lain-lain. Ini semua dapat menjadi atribut dari entitas data, bergantung pada persyaratan data yang dimiliki. Notasi untuk atribut sangat bervariasi. Atribut komposit adalah atribut yang merupakan gabungan beberapa atribut. Atribut kunci dapat memiliki lebih dari 1 atribut di dalamnya yang disebut dengan atribut komposit. Berikut adalah contoh gambar atribut.



Gambar 2.8 Contoh Atribut

### 3. Hubungan Antar Entitas (*Relationship*)

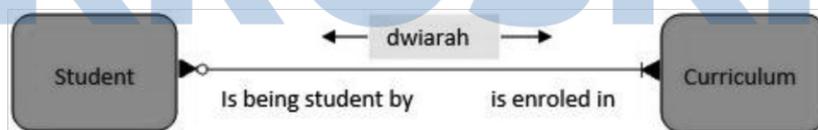
Keterhubungan (*relationship*) adalah asosiasi bisnis yang terdapat di antara satu atau beberapa entitas. Keterhubungan dapat menyajikan peristiwa yang menghubungkan entitas ataupun hanya suatu pertalian yang ada antara entitas. Berikut adalah contoh keterhubungan.



Gambar 2.9 Contoh Keterhubungan (*Relationship*)

### 4. Kardinalitas

Merupakan jumlah minimum dan maksimum kehadiran satu entitas terkait dengan kehadiran tunggal dari entitas lainnya. Karena semua keterhubungan selalu bersifat dwiarah, kardinalitas harus didefinisikan untuk kedua arah.



Gambar 2.10 Contoh Kardinalitas (1)

Sebagai contoh pada gambar di atas, harus dibaca bagaimana kardinalitas dari:

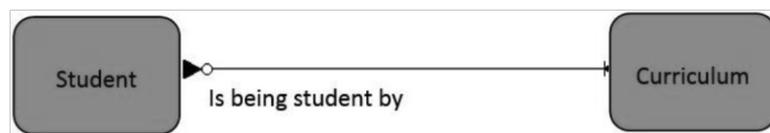
a. *Student* ke *Curriculum*



Gambar 2.11 Contoh Kardinalitas (2)

Gambar di atas menunjukkan bahwa setiap *student* terdaftar untuk mengikuti minimal 1 kurikulum dan maksimal banyak kurikulum. Tanda minimal 1 (ditandai dengan tanda “|”) dan maksimal banyak (ditandai dengan tanda “<”).

b. *Curriculum ke Student*



Gambar 2.12 Contoh Kardinalitas (3)

Hubungan kardinalitas harus dibaca timbal balik (dwi arah) karena bisa saja jumlah minimum dan maksimumnya berbeda dari arah berlawanan. Sebagai contoh, gambar di atas memperlihatkan bagaimana kardinalitas antara entitas kurikulum dengan *student* sebagai berikut. Setiap kurikulum dapat dipelajari minimal nol mahasiswa dan maksimalnya banyak. Maksudnya, sebuah kurikulum bisa saja tidak ada mahasiswa yang mempelajari.

5. Derajat relasi

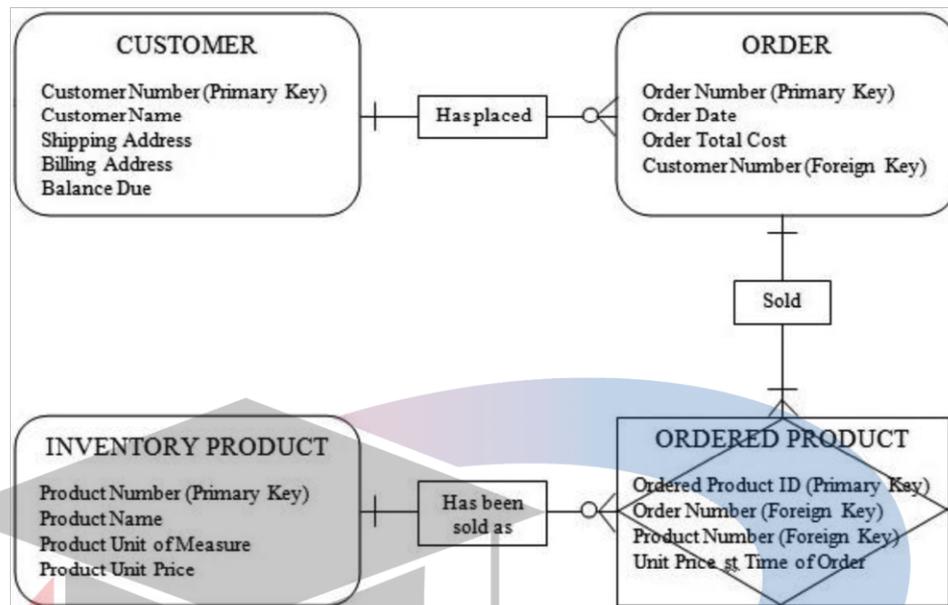
Derajat adalah jumlah entitas yang berpartisipasi dalam suatu keterhubungan yang dapat dikelompokkan sebagai berikut:

- a. Keterhubungan biner – keterhubungan antara dua entitas.
- b. Keterhubungan rekursif – keterhubungan antara contoh entitas yang berbeda dalam entitas yang sama (antar 1 entitas disebut keterhubungan *unary*).
- c. Keterhubungan ternary – keterhubungan antara tiga entitas atau lebih.

Derajat sesungguhnya menyajikan aturan bisnis. Kesalahan dalam mengidentifikasi atau mendokumentasikan derajat bisa menyebabkan sistem tidak memenuhi persyaratan bisnis dengan tepat.

Pemodelan data merupakan proses mengorganisasikan dan mendokumentasikan data sistem. Yang dimaksud data sistem adalah data yang harus ditangkap, diolah, dan disimpan sistem. Diagram *Entity Relationship* (ER) dapat digunakan sebagai teknik pemodelan data. Diagram ini secara garis besar menunjukkan berbagai entitas data (*entity*) yang ada dan berbagai keterhubungan (*relationship*) di antara entitas-entitas tersebut [4].

Berikut ini adalah contoh sederhana diagram ER [4].



Gambar 2.13 Contoh Diagram ER

Gambar di atas adalah contoh sebuah diagram ER sederhana. Diagram di atas menjelaskan fakta-fakta berikut [4]:

1. Sistem perlu menyimpan data tentang pelanggan, pesanan, dan produk.
2. Nilai dari nomor pelanggan secara unik mengidentifikasi tepat satu orang pelanggan. Begitu pula dengan nomor pesanan dan nomor produk.
3. Sistem perlu mengetahui nama, alamat pengiriman, alamat penagihan, dan saldo utang untuk setiap pelanggan. Sistem perlu mengetahui tanggal dan total biaya untuk setiap pesanan. Sistem perlu mengetahui nama, satuan, dan harga satuan untuk tiap produk.
4. Pelanggan bisa saja belum pernah melakukan pesanan, bisa saja hanya pernah memesan berkali-kali.
5. Pesanan dilakukan hanya oleh satu pelanggan. Nomor pelanggan yang dicatat pada pesanan menunjukkan pelanggan bersangkutan yang memesan.
6. Pesanan bisa mengandung satu atau beberapa produk yang dipesan.
7. Produk dapat tercatat dalam nol, satu, atau beberapa produk yang dipesan.
8. Produk yang dipesan menunjukkan satu produk dalam satu pesanan. Nomor pesanan pada produk yang dipesan menunjukkan pesannya, sedangkan nomor produk pada produk yang dipesan menunjukkan produk itu sendiri. Nomor