

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

2.1.1 Sistem

Sistem adalah suatu kesatuan yang terdiri dari dua komponen atau lebih atau subsistem yang berinteraksi untuk suatu tujuan (Hartono, 2003: 54). Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut.

1. Sistem di klasifikasikan sebagai sistem abstrak (*abstrack system*) dan sistem fisik (*physical system*). Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem fisik merupakan sistem yang ada secara fisik.
2. Sistem di klasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam sedangkan sistem buatan manusia adalah yang di rancang oleh manusia.
3. Sistem di klasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tertentu atau (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat di prediksi sedangkan yang tak tentu sistem yang kondisi masa depannya tidak bisa di prediksi karena mengandung unsur probabilitas.
4. Sistem di klasifikasikan sebagai sistem tertutup dan sistem terbuka. Sistem tertutup adalah sistem yang tidak berhubungan dengan lingkungan luar

sedangkan yang terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

2.1.2 Informasi

Informasi adalah data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut (Kadir, 2003: 31). Sedangkan sumber informasi adalah data. Merupakan kenyataan yang menggambarkan suatu kegiatan-kegiatan dan kesatuan nyata. Kegiatan-kegiatan (*event*) adalah sesuatu yang terjadi pada saat yang tertentu. Kualitas Informasi yaitu.

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bisa menyesatkan yang berarti juga informasi harus mencerminkan maksudnya dengan jelas.
2. Tepat waktu, berarti informasi yang datang pada penerima tidak boleh terlambat, karena informasi yang telah usang tidak akan mempunyai arti lagi.
3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya dan relevansi informasi tiap-tiap orang berbeda-beda.

2.1.3 Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Hartono, 2003: 11).

Sistem informasi terdiri dari *input*, proses, dan *output*. Pada proses terdapat hubungan timbal balik dengan 2 elemen, yaitu kontrol kinerja sistem dan

sumber-sumber penyimpanan data, baik berupa karakter-karakter huruf maupun berupa numerik. Saat ini data bisa berupa suara atau *audio* maupun gambar atau *video*. Data ini diproses dengan metode-metode tertentu dan akan menghasilkan *output* yang berupa informasi. Informasi yang dihasilkan dapat berupa laporan atau *report* maupun solusi dari proses yang telah dijalankan.

2.2 Pengendalian Persediaan

Menurut Herjanto (2008: 237) Pengendalian persediaan dapat diartikan sebagai serangkaian kebijakan yang dilakukan untuk menentukan persediaan yang harus dijaga, menentukan kapan pemesanan suatu barang harus dilakukan dan berapa kuantitas yang harus dipesan.

Pengendalian persediaan membutuhkan ketepatan, apabila jumlah persediaan terlalu besar maka akan timbul dana menganggur yang besar, meningkatnya biaya penyimpanan, dan resiko kerusakan barang yang lebih besar. Namun, jika persediaan terlalu sedikit akan mengakibatkan resiko terjadinya kekurangan persediaan (*stockout*) karena seringkali bahan/barang tidak bisa didatangkan secara mendadak dan sebesar yang dibutuhkan, yang mengakibatkan terhentinya proses produksi, tertundanya penjualan, bahkan hilangnya pelanggan.

Kondisi ideal dari pengendalian persediaan barang adalah tercapainya keseimbangan antara melayani kebutuhan bahan/barang dengan tepat dan dengan biaya yang rendah.

2.3 Konsep Persediaan

Persediaan adalah bahan atau barang yang disimpan yang akan digunakan untuk memenuhi tujuan tertentu, misalnya untuk digunakan dalam

proses produksi atau perakitan, untuk dijual kembali, atau untuk suku cadang dari suatu peralatan atau mesin. Persediaan dapat berupa bahan mentah, bahan pembantu, bahan dalam proses, barang jadi maupun, ataupun suku cadang. Bisa dikatakan tidak ada perusahaan yang beroperasi tanpa persediaan, meskipun sebenarnya persediaan hanyalah suatu sumber dana yang menganggur, karena sebelum persediaan digunakan berarti dana yang terikat didalamnya tidak dapat digunakan untuk keperluan lain. Begitu pentingnya persediaan ini sehingga para akuntan memasukkannya dalam neraca sebagai salah satu pos aktiva lancar (Herjanto, 2008: 237). Adapun fungsi penting yang terdapat pada persediaan dalam memenuhi kebutuhan perusahaan adalah sebagai berikut.

1. Menghilangkan resiko keterlambatan pengiriman bahan baku atau barang yang harus dibutuhkan perusahaan
2. Menghilangkan resiko jika material yang dipesan tidak baik sehingga harus dikembalikan
3. Menghilangkan resiko terhadap kenaikan harga barang atau inflasi
4. Untuk menyimpan bahan baku yang dihasilkan secara musiman sehingga perusahaan tidak akan kesulitan jika bahan itu tidak tersedia dipasaran
5. Mendapatkan keuntungan dari pembelian berdasarkan diskon kuantitas
6. Memberikan pelayanan kepada pelanggan dengan tersedianya barang yang diperlukan.

2.3.1 Jenis Persediaan

Selain dari persediaan yang dilakukan dalam bentuk bahan mentah, bahan pembantu, barang setengah jadi, dan barang jadi. Menurut Herjanto (2008: 238) persediaan juga dapat dikelompokkan ke dalam empat jenis, yaitu.

1. *Fluctuation stock*, merupakan persediaan yang dimaksudkan untuk menjaga terjadinya fluktuasi permintaan yang tidak diperkirakan sebelumnya, dan untuk mengatasi bila terjadi kesalahan/penyimpangan dalam prakiraan penjualan, waktu produksi, atau pengiriman barang.
2. *Anticipation stock*, merupakan persediaan untuk menghadapi permintaan yang dapat diramalkan, misalnya pada musim permintaan tinggi, tetapi kapasitas produksi pada saat itu tidak mampu memenuhi permintaan. Persediaan ini juga dimaksudkan untuk menjaga kemungkinan sukarnya diperoleh bahan baku sehingga tidak mengakibatkan terhentinya produksi.
3. *Lot size inventory*, merupakan persediaan yang diadakan dalam jumlah yang lebih besar dari pada kebutuhan pada saat itu. Persediaan dilakukan untuk mendapatkan keuntungan dari harga barang (berupa diskon) karena membeli dalam jumlah yang besar, atau untuk mendapatkan penghematan dari biaya pengangkutan per unit yang lebih rendah.
4. *Pipeline inventory*, merupakan persediaan yang dalam proses pengiriman dari tempat asal ke tempat dimana barang itu akan digunakan. Misalnya, barang yang dikirim dari pabrik menuju tempat penjualan, yang dapat memakan waktu beberapa hari atau minggu.

2.4 Metode *Economic Order Quantity* (EOQ)

Kuantitas pemesanan ekonomis (*economic order quantity*) atau biasa disingkat EOQ merupakan metode yang dapat memberikan gambaran mengenai pemesanan persediaan dengan meminimumkan biaya pemesanan, penyimpanan, dan kuantitas barang yang akan dipesan. Metode ini merupakan salah satu model

klasik yang paling banyak dikenal dalam teknik pengendalian persediaan diperkenalkan oleh FW Harris pada tahun 1914 (Herjanto, 2008: 245).

Dengan menggunakan perhitungan EOQ, maka persediaan yang ada di dalam gudang tidak terlalu banyak, tapi juga tidak akan terlalu sedikit, sehingga aktivitas perusahaan tidak akan terganggu karenanya (Hansen dan Mowen, 2005: 472). Beberapa asumsi yang mendukung model ini adalah sebagai berikut.

1. Barang yang dipesan dan disimpan hanya satu macam.
2. Kebutuhan/permintaan barang diketahui dan konstan.
3. Biaya pemesanan dan biaya penyimpanan diketahui dan konstan.
4. Barang yang dipesan dan diterima dalam satu kelompok (*batch*).
5. Harga barang tetap dan tidak tergantung dari jumlah yang dibeli.
6. Waktu tenggang (*lead time*) diketahui dan konstan.

Tetapi dalam kenyataannya asumsi-asumsi di atas tidak dapat dipenuhi semuanya, karena kondisi dan keadaan yang terkadang bisa terjadi tiba-tiba. Oleh karena itu metode EOQ mengalami pengembangan yang disesuaikan dengan kondisi dan keadaan dari perusahaan itu sendiri.

Secara umum metode EOQ dapat dirumuskan sebagai berikut.

$$Q = \sqrt{\frac{2DS}{H}} \dots\dots\dots(2.1)$$

Dimana:

D = Jumlah kebutuhan

S = Biaya pemesanan

H = Biaya penyimpanan

Untuk mengetahui jumlah kebutuhan dapat ditentukan dari jumlah permintaan barang dari pelanggan yang terpenuhi dan yang tidak terpenuhi per

periode. Biaya pemesanan (S) adalah biaya yang dikeluarkan setiap melakukan pemesanan barang. Biaya penyimpanan (H) dapat ditentukan dari nilai penyimpanan (% terhadap nilai barang) dikalikan harga barang. Besarnya biaya penyimpanan biasanya berkisar antara 12 sampai 40 persen dari harga barang per-unit (Handoko, 1984: 337).

2.5 Persediaan Pengaman (*Safety Stock*)

Menurut Wignjosoebroto (2003: 286) Persediaan pengaman (*safety stock*) adalah persediaan tambahan yang diadakan untukantisipasi atau menjaga kemungkinan terjadinya fluktuasi, ketidak-pastian, dan diluar kendali sistem industri yang berkaitan dengan tingkat kebutuhan / permintaan, laju produksi, waktu yang dibutuhkan untuk penggantian, dan hal-hal lain. selain itu persediaan pengaman ini juga digunakan untuk menanggulangi terjadinya keterlambatan datangnya bahan baku.

Persediaan pengaman dapat ditentukan dalam jumlah tertentu, atau berdasarkan dari presentase dari kebutuhan selama menunggu barang (waktu tenggang). Cara lain menentukan persediaan pengaman adalah dengan pendekatan tingkat pelayanan (*service level*). Tingkat pelayanan dapat didefinisikan dengan kemungkinan permintaan tidak akan melebihi persediaan selama waktu tenggang.

2.6 Waktu Tunggu (*Lead Time*)

Untuk memesan suatu barang sampai barang itu datang diperlukan jangka waktu yang bervariasi dari beberapa jam sampai beberapa bulan. Perbedaan waktu antara saat memesan sampai saat barang datang dikenal dengan istilah waktu tenggang (*lead time*). Waktu tenggang sangat dipengaruhi oleh

ketersediaan dari barang itu sendiri dan jarak lokasi antara pembeli dan pemasok berada (Herjanto, 2008: 258).

2.7 Titik Pemesanan Ulang (*Reorder Point*)

Menurut Wignjosoebroto (2003: 393) Titik pemesanan ulang (*reorder point*) atau biasa disingkat dengan RoP adalah titik saat pemesanan barang harus dilakukan sebanyak yang dibutuhkan, banyaknya persediaan yang ada digudang, atau banyaknya persediaan yang direncanakan untuk mengantisipasi kebutuhan produksi dimasa mendatang. Pada saat persediaan sudah mencapai suatu tingkat tertentu (*reorder level*), maka harus dilakukan pemesanan supaya kedatangan atau penerimaan barang yang dipesan tepat pada waktunya karena adanya *lead time*, yaitu waktu antara pemesanan dilakukan dengan barang diterima dan *safety stock* atau persediaan pengaman. Jika RoP ditetapkan terlalu rendah, persediaan akan habis sebelum persediaan pengganti diterima sehingga permintaan pelanggan terganggu. Jika RoP terlalu tinggi maka persediaan baru datang namun persediaan digudang masih banyak maka akan mengakibatkan pemborosan biaya investasi yang berlebihan.

Titik pemesanan ulang biasanya ditetapkan dengan cara menambahkan penggunaan selama waktu tenggang dengan persediaan pengaman. Secara umum dapat digambarkan dalam bentuk rumus sebagai berikut.

$$\text{RoP} = d \times L + \text{SS} \dots \dots \dots (2.2)$$

Di mana:

RoP = Titik pemesanan ulang (*reorder point*)

d = Tingkat kebutuhan per unit waktu

L = Waktu tenggang.

SS = Persediaan pengaman (*safety stock*)

2.8 Siklus Hidup Pengembangan Sistem

Sistem informasi yang baik adalah sistem yang dapat dengan mudah untuk dikembangkan sesuai dengan kondisi dan perkembangan dimana sistem tersebut diaplikasikan. Salah satu konsep pengembangan sistem adalah dengan menggunakan *System Development Life Cycle (SDLC)*.

Menurut Kendall (2006: 11) Siklus hidup pengembangan sistem adalah pendekatan melalui beberapa tahap untuk menganalisa dan merancang sistem dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik.

Menurut Herlambang dan Tanuwijaya (2005: 180) tahap-tahap dalam siklus hidup pengembangan sistem meliputi.

1. Tahap perencanaan (*Planning*)

Langkah pertama yang dilakukan pada pengembangan sistem informasi adalah merencanakan apa yang akan dibuat saat akan melakukan pengembangan. Pada tahap ini, detail dari penelitian dirumuskan dengan seksama agar pengembangan sistem yang dibangun tidak keluar dari apa yang telah diharapkan sebelumnya dan mempelajari konsep-konsep serta teori-teori yang mendukung. Selanjutnya adalah menetapkan tujuan dari penelitian, hal ini dimaksudkan agar batasan-batasan dan fungsi dari pengembangan dapat diketahui dengan jelas.

2. Tahap Analisis (*Analisis*)

Setelah tahap perencanaan diselesaikan, maka langkah selanjutnya adalah melakukan analisis terhadap perencanaan yang dilakukan. Analisis dilakukan

untuk menentukan, apakah akan merancang sistem yang baru atau memperbaharui sistem yang sudah ada.

3. Tahap Rancangan (*Design*)

Pada tahap ini, sebagian besar kegiatan yang berorientasi ke komputer dilaksanakan. Rencana pembuatan program dilaksanakan dan juga testing programnya. Latihan bagi para pemakai sistem dimulai, apabila pemakai sistem telah puas melihat hasil testing yang dilakukan maka akan dilanjutkan untuk tahap selanjutnya.

4. Tahap Penerapan (*Implementation*)

Keunggulan kompetitif dapat dicapai jika suatu sistem dapat menggunakan secara maksimal terhadap sumber daya fisik dan sumber daya konseptual. Pada tahap penerapan ini bertujuan untuk menginformasikan kepada pihak yang terlibat tentang keputusan menerapkan sistem baru.

5. Tahap Penggunaan (*Use*)

Tahap terakhir yang terdiri dari 3 (tiga) langkah penting, yaitu.

- a. Menggunakan sistem, untuk mencapai tujuan yang diidentifikasi pada tahap perencanaan.
- b. Audit sistem, kegiatan pengkajian setelah penerapan sistem apakah sudah memenuhi kriteria kinerja.
- c. Memelihara sistem, yang bertujuan untuk memperbaiki kesalahan, menjaga kemuktahiran sistem, dan meningkatkan kinerja sistem.

Dengan menggunakan konsep siklus hidup, sistem harus dievaluasi secara terus menerus untuk menetapkan apakah sistem informasi tersebut masih

layak diaplikasikan, jika tidak, sistem informasi tersebut akan diganti dengan yang baru dan dimulai dari perencanaan kembali.

2.9 Flow Chart

Flow chart adalah teknik analisis yang dipergunakan untuk mendeskripsikan beberapa aspek dari sistem informasi secara jelas, ringkas dan logis. *Flow chart* menggunakan serangkaian simbol standar yang menggambarkan prosedur yang digunakan oleh perusahaan dan arus data melalui sistem (Romney dan Steinbart, 2003: 163). Ada dua macam *flow chart* yang menggambarkan proses dengan menggunakan komputer, yaitu:

1. *Flow chart system*


Flow chart system adalah suatu gambar yang menunjukkan hubungan antara *input*, pemrosesan dan *output* dari suatu sistem informasi.


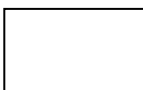

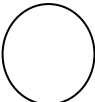





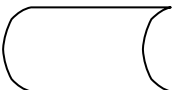
2. *Flow chart program*

Flow chart program adalah bagan yang memperlihatkan urutan proses secara logis yang dilaksanakan oleh komputer dalam menjalankan sebuah program. *Flow chart program* ini merupakan langkah awal pembuatan program. Dengan adanya *flow chart program* maka urutan proses di program menjadi lebih jelas.

Flow chart disusun dengan simbol. Simbol-simbol yang digunakan dalam *flow chart* dapat dilihat pada Tabel 2.1.

Tabel 2.1 Simbol-simbol *Flow Chart*

Simbol	Nama	Keterangan
	Terminal	Simbol untuk menunjukkan awal, akhir dan pemberhentian dalam suatu proses atau program

Simbol	Nama	Keterangan
	Proses manual	Simbol yang menunjukkan pengolahan yang dilakukan secara manual
	Proses dengan komputer	Simbol untuk menunjukkan pengolahan yang dilakukan dengan komputer
	Arah dokumen atau proses	Simbol menunjukkan arah aliran dokumen atau proses
	<i>On-page connector</i>	Simbol untuk menggambarkan proses pada halaman yang sama
	<i>Off-page connector</i>	Simbol untuk menggambarkan proses pada halaman yang berbeda
	Keputusan	Simbol untuk kondisi yang menggambarkan langkah pengambilan keputusan
	<i>On-line keying</i>	Simbol yang menyatakan <i>input</i> dengan menggunakan mesin yang mempunyai keyboard
	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk tulisan tangan atau output dicetak dengan komputer
	<i>Input/output</i>	Simbol yang menyatakan proses <i>input</i> dan output di dalam bagan alir program
	Penyimpanan online	Simbol yang menggambarkan data yang disimpan dalam <i>file on-line</i> yang dapat diakses secara langsung

2.10 Data Flow Diagram (DFD)

Menurut Kristanto (2004: 12), *Data Flow Diagram* (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersebut disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.

Data Flow Diagram merupakan suatu metode pengembangan sistem yang terstruktur (*structure analysis and design*). Penggunaan notasi dalam *data flow diagram* sangat membantu untuk memahami suatu sistem pada semua tingkat kompleksitas. Pada tahap analisis, penggunaan notasi ini dapat membantu dalam berkomunikasi dengan pemakai sistem untuk memahami sistem secara logika.

Adapun beberapa simbol yang sering dipakai dalam DFD terdiri dari:

1. Simbol *entity*, digunakan sebagai sumber dari *input* sistem atau tujuan dari *output* sistem.



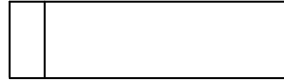
Gambar 2.1 Simbol *Entity*

2. Simbol proses dimana sering digunakan untuk melakukan perubahan terhadap *input* yang masuk sehingga menghasilkan data dari perubahan *input* yang diolah.



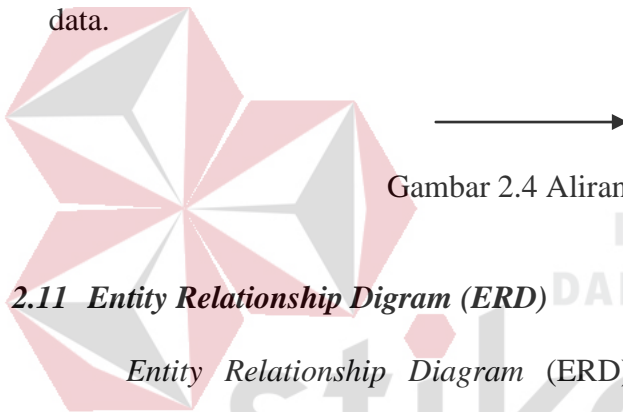
Gambar 2.2 Simbol *Process*

3. Simbol data *store* atau penyimpanan data, sering digunakan sebagai simpanan dari data yang dapat berupa suatu *file* atau basis data.



Gambar 2.3 Simbol *Data Store*

4. Simbol yang menggambarkan aliran data, yang sering digunakan untuk menghubungkan antara proses dengan proses, proses dengan sumber proses, dan proses dengan tujuan. Sedangkan anak panahnya menunjukkan arah aliran data.



Gambar 2.4 Aliran Data

2.11 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) adalah gambaran pada sistem dimana di dalamnya terdapat hubungan antara *entity* beserta relasinya. Model ER adalah persepsi terhadap dunia nyata sebagai terdiri objek-objek dasar yang disebut entitas dan keterhubungan (*relationship*) antar entitas-entitas itu. Konsep paling dasar di model ER adalah entitas, relationship dan atribut.

Model ER penting dalam perancangan basis data. Model ER menyediakan konsep-konsep berguna yang mementingkan bergerak dari deskripsi-deskripsi informal apa yang diinginkan pemakai terhadap basisdata menuju deskripsi-deskripsi lebih rinci dan presisi yang dapat diimplementasikan DBMS.

Menurut Marlinda (2004:28), *Attribute* adalah kolom di sebuah relasi.

Macam-macam *attribute* yaitu :

1. *Simple Attribute*

Attribute ini merupakan *attribute* yang unik dan tidak dimiliki oleh *attribute* lainnya, misalnya *entity* mahasiswa yang *attribute*-nya NIM.

2. *Composite Attribute*

Composite attribute adalah *attribute* yang memiliki dua nilai harga, misalnya nama besar (nama keluarga) dan nama kecil (nama asli).

3. *Single Value Attribute*

Attribute yang hanya memiliki satu nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya umur (tanggal lahir).

4. *Multi Value Attribute*

Multi value attribute adalah *attribute* yang banyak memiliki nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya pendidikan (SD, SMP, SMA).

5. *Null Value Attribute*

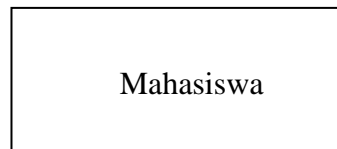
Null value attribute adalah *attribute* yang tidak memiliki nilai harga, misalnya *entity* tukang becak dengan *attribute*-nya pendidikan (tanpa memiliki ijazah).

Sedangkan relasi adalah hubungan antar *entity* yang berfungsi sebagai hubungan yang mewujudkan pemetaan antar *entity*.

Komponen utama pembentuk ER Model adalah:

1. *Entity*

Entity adalah segala hal nyata maupun abstrak yang berhubungan dengan masukan dan keluaran data. Contoh: Mahasiswa, matakuliah, dan sebagainya.

Gambar 2.5 Simbol *Entity*

2. *Attribute*

Attribute adalah identifikasi dari suatu entitas atau *entity*. Contoh: *Entity* mahasiswa mempunyai attribute NIM, Nama, dan seterusnya.

3. *Relation*

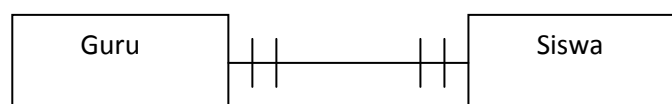
Relation yang berisi komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari 'dunia' yang kita tinjau, dapat digambarkan dengan lebih sistematis menggunakan *Diagram Entity Relation* (Diagram ER).

Macam-macam relasi itu sendiri antara lain:

a. *One to One* (1:1)

Relasi dari *entity* satu dengan *entity* dua adalah satu berbanding satu.

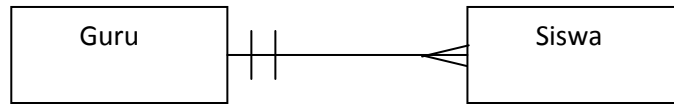
Contoh: Pada pelajaran privat, satu guru mengajar satu siswa dan satu siswa hanya diajar oleh satu guru.

Gambar 2.6 Relasi *One to One*

b. *One to Many* (1: m)

Relasi antara *entity* yang pertama dengan *entity* yang kedua adalah satu berbanding banyak atau dapat pula dibalik, banyak berbanding satu.

Contoh: Pada sekolah, satu guru mengajar banyak siswa dan banyak siswa diajar oleh satu guru.



Gambar 2.7 Relasi *One to Many*

c. *Many to Many* (m : m)

Relasi antara *entity* yang satu dengan *entity* yang kedua adalah banyak berbanding banyak. Contoh: Pada perkuliahan, satu dosen mengajar banyak mahasiswa dan satu mahasiswa diajar oleh banyak dosen pula.



Gambar 2.8 Relasi *Many to Many*

Entity Relationship Diagram ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, mudah dimengerti pemakai dan mudah disajikan oleh perancang *database*. Untuk itu *Entity Relationship Diagram* dibagi menjadi dua jenis model, yaitu:

1. *Conceptual Data Model (CDM)*

Conceptual Data Model (CDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

2. *Physical Data Model (PDM)*

Physical Data Model (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisik.

2.12 Konsep Dasar Basis Data

2.12.1 Database

Menurut Marlinda (2004), *database* adalah suatu susunan atau kumpulan data operasional lengkap dari suatu organisasi atau perusahaan yang diorganisir atau dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.

Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai), masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data).

2.12.2 Sistem Basis Data

Menurut Marlinda (2004), sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Pada sebuah sistem basis data terdapat empat komponen pokok, yaitu:

1. Data

Data didalam sebuah basis data dapat disimpan secara terintegrasi (*integrated*) dan data dapat dipakai secara bersama-sama (*shared*).

2. *Hardware* (Perangkat Keras)

Terdiri dari semua peralatan komputer yang digunakan untuk pengelolaan sistem basis data.

3. *Software* (Perangkat Lunak)

Berfungsi sebagai perantara (*interface*) antara pemakai dengan data fisik pada basis data.

4. *User* atau pemakai

2.13 *Tools* Pemrograman

Dalam pengembangan suatu Aplikasi, tentunya membutuhkan suatu *tool* atau alat berupa bahasa pemrograman. Salah satu *tool* dalam bahasa pemrograman yang sekarang dipakai adalah keluarga Microsoft Visual Studio 2013 yang menggunakan teknologi .NET

2.13.1 Visual Studio 2013

Microsoft Visual Studio 2013 merupakan sebuah IDE (*Integrated Development Environment*) yang dikembangkan oleh microsoft. IDE ini mencakup semua bahasa pemrograman berbasis .NET framework yang dikembangkan oleh microsoft. Keunggulan Microsoft Visual Studio 2013 ini antara lain adalah *support* untuk windows 8, editor baru dengan WPF (*Windows Presentation Foundation*), dan banyak peningkatan fitur lainnya.

2.13.2 SQL Server

Microsoft SQL Server merupakan produk *Relational Database Management System* (RDBMS) yang dibuat oleh Microsoft. Orang sering menyebutnya dengan SQL Server saja. Microsoft SQL Server juga mendukung

SQL sebagai bahasa untuk memproses *query* ke dalam *database*. Microsoft SQL Server banyak digunakan pada dunia bisnis, pendidikan atau juga pemerintahan sebagai solusi database atau penyimpanan data. Pada tahun 2008 Microsoft mengeluarkan SQL Server 2008 yang merupakan versi yang banyak digunakan. Berikut ini adalah beberapa fitur yang dari sekian banyak fitur yang ada pada SQL Server 2008:

1. *XML Support*. Dengan fitur ini, Anda bisa menyimpan dokumen XML dalam suatu tabel, meng-*query* data ke dalam format XML melalui Transact-SQL dan lain sebagainya.
2. *Multi-Instance Support*. Fitur ini memungkinkan untuk menjalankan beberapa *database engine* SQL Server pada mesin yang sama.
3. *Data Warehousing and Business Intelligence (BI) Improvements*. SQL Server dilengkapi dengan fungsi-fungsi untuk keperluan *Business Intelligence* melalui *Analysis Services*. Selain itu, SQL Server 2008 juga ditambah dengan *tools* untuk keperluan *data mining*.
4. *Performance and Scalability Improvements*. SQL Server menerapkan *distributed partitioned views* yang memungkinkan untuk membagi *workload* ke beberapa server sekaligus. Peningkatan lainnya juga dicapai di sisi DBCC, *indexed view*, dan *index reorganization*.
5. *Query Analyzer Improvements*. Fitur yang dihadirkan antara lain: *integrated debugger*, *object browser*, dan *fasilitas object search*.
6. *DTS Enhancement*. Fasilitas ini sekarang sudah mampu untuk memperhatikan *primary key* dan *foreign key constraints*. Ini berguna pada saat migrasi tabel dari RDBMS lain.

7. *Transact-SQL Enhancements*. Salah satu peningkatan disini adalah T-SQL sudah mendukung UDF (*User-Definable Function*). Ini memungkinkan Anda untuk menyimpan rutin-rutin ke dalam *database engine*.

2.14 *Blackbox Testing*

Menurut Romeo (2003), *Blackbox Testing* merupakan pendekatan komplementer dari teknik *whitebox*, karena pengujian *blackbox* diharapkan mampu mengungkapkan kelas kesalahan yang lebih luas dibandingkan teknik *whitebox*. Pengujian *blackbox* berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program. Pengujian *blackbox* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak.

Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *blackbox* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan.

Pengujian *blackbox* berusaha menemukan kesalahan dalam kategori :

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

2.15 Skala *Likert*

Skala *likert* digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok tentang kejadian atau gejala sosial yang telah ditetapkan secara spesifik oleh peneliti yang selanjutnya disebut sebagai variabel penelitian di dalam suatu kuesioner. Dengan menggunakan skala *likert*, maka variabel yang akan diukur dapat dijabarkan menjadi dimensi, sub variabel dan kemudian indikator yang dapat diukur. Akhirnya indikator-indikator yang terukur ini dapat dijadikan titik tolak untuk membuat item instrument yang berupa pertanyaan atau pernyataan yang perlu dijawab oleh responden. Setiap jawaban dihubungkan dengan bentuk pernyataan atau dukungan sikap yang diungkapkan dengan kata-kata sebagai berikut (Ridwan, 2005):

Pernyataan Positif

Sangat setuju (SS) = 5

Setuju (S) = 4

Netral (N) = 3

Tidak Setuju (TS) = 2

Sangat Tidak Setuju (STS) = 1

Pernyataan Negatif

Sangat Setuju (SS) = 1

Setuju (S) = 2

Netral (N) = 3

Tidak Setuju (TS) = 4

Sangat Tidak Setuju (STS) = 5