

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Sistem**

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau *variable* yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif. Yang dimaksud dengan unsur atau komponen pembentuk organisasi disini bukan hanya bagian-bagian yang tampak secara fisik, tetapi juga hal-hal yang mungkin bersifat abstrak atau konseptual seperti misi, pekerjaan, kegiatan, kelompok informal, dan lain sebagainya. Selain itu suatu sistem tidak bisa lepas dari lingkungan sekitarnya maka umpan balik (*feed back*) dapat berasal dari lingkungan sistem yang dimaksud. (Tata Sutabri : 2012 : 10).

#### **II.2. Konsep Dasar Informasi**

##### **II.2.1. Informasi**

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi dan lain sebagainya. Informasi ibarat darah yang mengalir dalam suatu organisasi sehingga informasi ini sangat penting didalam suatu organisasi.

Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil dan akhirnya mati. (Tata Sutabri : 2012 : 29).

Kualitas informasi tergantung dari 3 hal yaitu :

1. Akurat (*Accurate*)

Informasi harus bebas dari kesalahan – kesalahan dan tidak menyesatkan.

Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

2. 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.

3. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk orang satu dengan yang lain berbeda, misalnya informasi sebab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan dan akan lebih relevan bila ditunjukkan ahli teknik perusahaan. (Yakub: 2012 : 9).

## **II.2.2. Sistem Informasi**

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar dengan laporan-laporan

yang diperlukan (Tata Sutabri : 2012 : 46). Sistem informasi terdiri dari komponen – komponen yang disebut blok bangunan (*building block*) yaitu :

1. Blok masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi, Input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen – dokumen dasar.

2. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang di inginkan.

3. Blok keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi (*technology block*)

Teknologi merupakan “tool box” dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, teknologi terdiri dari 3 bagian utama, yaitu teknisi (*brainware*), Perangkat lunak (*software*), perangkat keras (*hardware*).

5. Blok basis data (*database block*)

Basis data atau database merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (*control block*)

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan – kecurangan, kegagalan – kegagalan. (Tata Sutabri : 2012 : 47-48).

### **II.2.3. Sistem Informasi Geografis**

Istilah sistem informasi geografis merupakan gabungan dari tiga unsur pokok yaitu sistem, informasi dan geografis. Dengan demikian pengertian terhadap ketiga unsur pokok ini akan sangat membantu dalam memahami SIG. Dengan melihat unsur-unsur pokoknya, maka sudah jelas bahwa SIG juga merupakan salah satu tipe sistem informasi seperti yang telah dibahas, tetapi dengan tambahan unsur geografis. Istilah geografis merupakan bagian dari spasial (keruangan), kedua istilah ini sering digunakan secara bergantian atau bahkan tertukar satu sama lainnya hingga muncullah istilah yang ketiga yaitu geospasial. Ketiga istilah ini mengandung pengertian yang kurang lebih serupa didalam konteks SIG. Penggunaan kata geografis mengandung pengertian suatu persoalan atau hal mengenai wilayah di permukaan bumi, baik permukaan dua dimensi atau tiga dimensi. Jadi SIG juga merupakan sejenis perangkat lunak, perangkat keras (manusia, prosedur, basis data, dan fasilitas jaringan komunikasi) yang dapat

digunakan untuk memfasilitasi proses pemasukan, penyimpanan, manipulasi, menampilkan dan keluaran data/informasi geografis berikut atribut atribut terkait. (Eddy Prahasta :2009: 109-110). Jika beberapa definisi yang disebutkan diatas diperhatikan dengan teliti maka SIG dapat diuraikan menjadi beberapa sub-sistem sebagai berikut :

1. Data input :sub-sistem ini bertugas untuk mengumpulkan, mempersiapkan dan menyimpan data spasial dan atributnya dari berbagai sumber. Sub-sistem ini pula yang bertanggung jawab dalam mengonversikan atau mentransformasikan format-format dari aslinya kedalam format yang dapat digunakan oleh perangkat SIG yang bersangkutan.
2. Data *output* : sub-sistem ini bertugas untuk menampilkan atau menghasilkan keluaran (termasuk mengekspornya ke format yang dikehendaki) seluruh atau sebagian basis data (spasial) baik dalam bentuk softcopy maupun hardcopy seperti halnya tabel, grafik, report, peta dan lain sebagainya.
3. Data *management* : sub-sistem ini mengorganisasikan baik data spasial maupun tabel-tabel atribut terkait kedalam sebuah sistem basis data sedemikian rupa hingga mudah dipanggil kembali atau *diretrieve* (di *load* ke memori), di *update* dan di *edit*.
4. Data *manipulation* dan *analysis* : sus-sistem ini menentukan informasi-informasi yang dapat dihasilkan oleh SIG. Selain itu, sub-sistem ini juga melakukan manipulasi (evaluasi dan penggunaan fungsi-fungsi dan operator matematis dan logika) dan pemodelan data untuk menghasilkan informasi yang diharapkan. (Eddy Prahasta :2009: 118).

### II.3. PHP

PHP singkatan dari PHP *Hypertext Preprocessor* yang digunakan sebagai bahasa script server-side dalam pengembangan web yang disisipkan pada dokumen HTML. Hasilnyalah yang dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk aplikasi web dinamis. Artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, anda bisa menampilkan isi database ke halaman web. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (Active Server Page), Cold fusion, ataupun Perl. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara command line. Artinya, skrip PHP dapat dijalankan tanpa melibatkan web server maupun browser.

Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi tool yang disebut personal home page. Paker inilah yang menjadi cikal bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI versi 2. Pada versi inilah pemrograman dapat menempelkan kode terstruktur didalam tag HTML. Yang menarik, kode PHP juga bisa berkomunikasi dengan database dan melakukan perhitungan yang kompleks sambil jalan. Pada saat ini, PHP cukup populer sebagai peranti pemrograman web, terutama dilingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada server-server yang berbasis UNIX, Windows dan Macintosh.

Pada awalnya, PHP dirancang untuk diintegrasikan dengan web server apache. Namun, belakangan PHP juga dapat bekerja dengan web server seperti

PWS (Personal Web Server), IIS ( Internet Information Server) dan Xitami. Untuk mencoba PHP, anda tidak perlu menggunakan komputer berkelas server. Hanya dengan sebuah komputer biasa, anda bisa mempelajari dan mempraktikkan PHP. PHP bersifat bebas dipakai, anda tidak perlu membayar apapun untuk menggunakan perangkat lunak ini alias free. Anda dapat mendownloadnya melalui situs <http://www.php.net>. PHP tersedia dalam bentuk kode biner maupun kode sumber yang lengkap. (Abdul Kadir: 2012: 3)

Tipe data merupakan jenis data, setiap data pasti memiliki jenis data tertentu. Jenis data berguna untuk menghindari ekspresi yang tipe datanya bercampur. Sebagai contoh `'PHP' + 4.0` (jenis data *string PHP* mau ditambah dengan tipe data *numerik 4.0*). PHP mengenal 3 kelompok jenis data yaitu :

1. Integer (bilangan bulat)

Tipe data integer merupakan tipe data yang direpresentasikan berbentuk angka dan tidak memakai tanda titik desimal. Contoh : `$angka=2;`

2. Tipe data Double (bilangan pecahan)

Tipe data double merupakan tipe data yang direpresentasikan memakai tanda titik desimal. Contoh : `$angka=20.50;`

3. Tipe data String (teks)

Tipe data string merupakan tipe data yang direpresentasikan diapit tanda petik tunggal ( `' '` ) atau petik ganda ( `" "` ). Contoh : `$kantor="CV.Noornet";`  
( Yuniar Supardi :2010 : 17-18).

#### II.4. *ArcView*

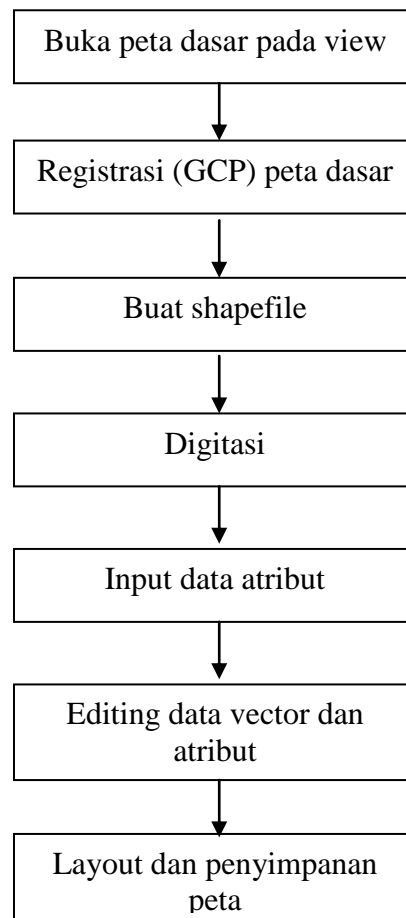
*ArcView* adalah salah satu *software* pengolah Sistem Informasi Geografik (SIG/GIS). Sistem Informasi Geografik sendiri merupakan suatu sistem yang dirancang untuk menyimpan, memanipulasi, menganalisis, dan menyajikan informasi geografi. Mungkin anda sudah kenal kenal dengan yang namanya peta. Perlu diketahui bahwa peta juga bisa disebut *SIG* atau istilahnya *SIG* Konvensional. Terdapat beberapa perbedaan antara peta di atas kertas (peta analog) dan *SIG* yang berbasis komputer. Perbedaannya adalah bahwa peta menampilkan data secara grafis tanpa melibatkan basis data. Sedangkan *SIG* adalah suatu sistem yang melibatkan peta dan basis data. Dengan kata lain peta adalah bagian dari *SIG*. Sedangkan pada *ArcView* anda dapat melakukan beberapa hal yang peta biasa tidak dapat melakukannya.

*Shapefile* adalah *file* yang menyimpan data *vector* dalam *arcview*. *Shapefile* inilah yang kemudian diolah dan dianalisis dalam berbagai pekerjaan spasial dengan *arcview*. Saat ditampilkan dalam lembar *view*, *shapefile* masih sebagai sebuah *theme*. Data *vector* yang tidak disimpan dalam sebuah *shapefile* hanya akan menjadi objek grafik dalam lembar *view*. Objek grafik tersebut akan hilang jika dimasukkan dalam *shapefile*. Oleh karena itu, sebelum proses *digitasi* dilakukan, terlebih dahulu harus dibentuk sebuah *shapefile* kosong untuk wadah data *vector* yang telah terdigitasi. (Eko Budiyanto :2010 : 1).

*Digitasi* merupakan proses pembentukan data *vector*. Dalam sistem informasi geografis dan pemetaan digital, data *vector* banyak digunakan sebagai dasar analisis dan berbagai proses. *Digitasi* yang dilakukan tanpa disimpan pada

sebuah *theme* (file *.shp*), maka akan menjadi bentuk grafis. Data grafis ini juga merupakan data *vector* tetapi tidak memiliki basis data *atributal*. Dalam pembuatan peta *digital*, data grafis harus disimpan di dalam sebuah *shapefile* (file *.shp*). Oleh karena itu, proses *digitasi* didahului dengan pembuatan sebuah *shapefile* kosong. (Eko Budiyanto :2010 : 19).

Peta dasar adalah peta yang akan digunakan sebagai referensi pembuatan peta. Peta dasar ini dapat berupa data citra satelit, data foto udara, atau data *raster* peta hasil proses penyiaman (*scanning*). Proses updating peta sering menggunakan data penginderaan jauh seperti citra satelit resolusi tinggi atau foto udara sebagai peta dasarnya. Sementara itu proses reproduksi peta biasanya menggunakan data *raster* peta tertentu yang disiam. Teknik *digitasi* peta dasar pada masa lalu sering menggunakan alat bantu yang bernama *digitizer*. Dalam perkembangannya, proses *digitasi* peta dasar kemudian dilakukan secara *onscreen*. Teknik *digitasi onscreen* ini adalah teknik digitasi peta dasar secara langsung pada layar computer. Akurasi teknik digitasi peta dasar *onscreen relative* lebih tinggi karena adanya kemampuan *zoom* pada setiap perangkat lunak yang digunakan untuk *digitasi*. (Eko Budiyanto :2010 : 25). Secara garis besar, pembuatan peta *digital* dilakukan dalam rangkaian proses berikut :



**Gambar II.1. Proses Pembuatan Peta Digital**  
Sumber : (Eko Budiyanto ; 2010 : 26)

## II.5. Alat Bantu Pengembangan Sistem

Alat bantu pengembangan sistem yang digunakan dalam penulisan skripsi ini adalah sebagai berikut :

### II.5.1 UML (Unified Modelling Language)

*Unified Modelling Language* (UML) merupakan alat komunikasi yang konsisten dalam mensupport para pengembang sistem saat ini. Sebagai perancang sistem, mau tidak mau pasti akan menjumpai *UML*, baik kita sendiri yang membuat atau sekedar membaca diagram *UML* buatan orang lain. (Prabowo Pudjo Widodo ; 2011 : 7). Sebelum ada *UML*, para pengembang bahasa pemrograman

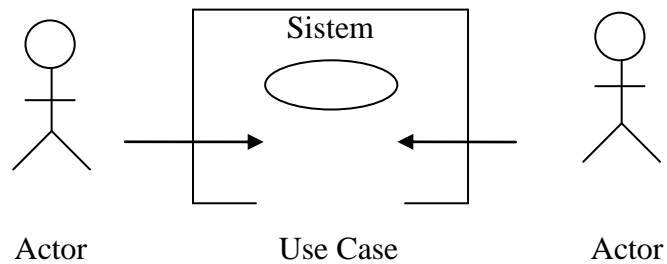
berorientasi objek sulit untuk berkomunikasi satu sama lain. Ada kira-kira 50 jenis notasi dan grafik yang menggambarkan bahasa pemrograman berorientasi objek pada waktu itu. Para pengguna notasi yang berlainan ini saling berebut pengaruh agar notasi yang mereka gunakan jadi notasi standar. (Prabowo Pudjo Widodo ; 2011 : 8).

Sejak tahun 1997, divisi *Revision Task Force (RTF)* milik *OMG* beberapa kali merevisi *UML*. Revisi dimaksudkan untuk memperkuat konsistensi notasi, meningkatkan kekompakan antara *user* dan pengembang perangkat lunak. Akan tetapi *UML* terpaksa mengikuti perkembangan *software-software* berbasis objek yang ada. Setelah dilakukan perubahan secara sistematis, akhirnya dihasilkan *UML 2.0* pada tahun 2003. (Prabowo Pudjo Widodo ; 2011 : 9).

## **II.5.2 Diagram – diagram Pada Metode UML**

### *1. Use Case Diagram*

*Use case* adalah alat bantu terbaik guna menstimulasikan pengguna potensial untuk mengatakan tentang suatu sistem dari sudut pandangnya. Tidak selalu mudah bagi pengguna untuk menyatakan bagaimana mereka bermaksud menggunakan sebuah sistem. Ide dasarnya adalah bagaimana melibatkan penggunaan sistem di fase – fase awal analisis dan perancangan sistem. Diagram use case menunjukkan 3 aspek dari sistem yaitu actor , use case dan sistem / sub sistem boundary. Actor mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan use case . Gambar II.4 mengilustrasikan actor , use case dan boundary.

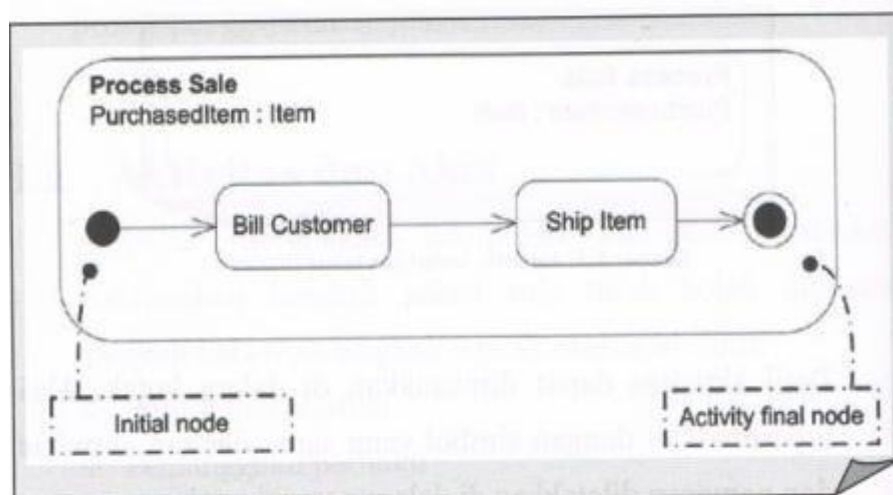


**Gambar II.2 Use Case Model**

Sumber : (Prabowo Pudjo Widodo ; 2011 : 17)

## 2. Activity Diagram

*Activity diagram* adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Berikut gambar dari sederhana dari *activity diagram*.

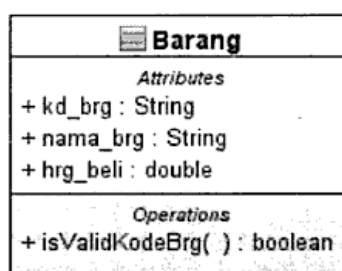


**Gambar II.3 Contoh Activity Diagram Sederhana**

Sumber : (Prabowo Pudjo Widodo ; 2011 : 143)

### 3. *Class Diagram*

Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif. (Prabowo Pudjo widodo ; 2011 :10)



**Gambar II.4 Contoh Class Diagram**  
Sumber : (Prabowo Pudjo Widodo ; 2011:41)

### 4. *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh obyek dan pesan yang diletakkan diantara obyek – obyek di dalam *use case*.

Komponen utama *sequence diagram* terdiri atas obyek yang dituliskan dengan kotak segiempat bernama Message diwakili olehn gariks dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*

## II.6. Sistem Basis Data

Basis adalah markas, gudang, tempat bersarang tau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (dosen, mahasiswa, pembeli, pelanggan), barang, hewan, peristiwa,

konsep, keadaan dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Basis data adalah kumpulan data yang saling berhubungan yang disimpan/diorganisasi secara bersama, dalam bentuk sedemikian rupa, dan tanpa redundansi (pengulangan) yang tidak perlu supaya dapat dimanfaatkan kembali dengan cepat dan mudah untuk memenuhi berbagai kebutuhan. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 3).

Komponen-komponen pada sebuah sistem basis data antara lain:

1. Data
2. Perangkat keras
3. Sistem operasi
4. Basis data
5. DBMS (*Database Management System*)
6. Pemakai
7. Aplikasi lain. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 4).

DBMS merupakan perangkat lunak yang dirancang untuk dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah yang besar. DBMS juga dirancang untuk dapat melakukan manipulasi data secara lebih mudah. DBMS merupakan antarmuka pengguna basis data (baik merupakan pengguna langsung maupun aplikasi) dengan data yang tersimpan. Penyimpanan data oleh DBMS disesuaikan dengan bentuk model datanya, beberapa contoh DBMS adalah *PostgreSQL*, *MYSQL*, *DB2*, *Oracle*, *SQLserver* dan lain lain. Sebelum ada DBMS, data pada umumnya disimpan dalam bentuk *flat file*, yaitu *file teks* yang ada pada sistem operasi. Sampai sekarang pun masih banyak aplikasi yang

menyimpan dalam bentuk *flat file* secara langsung. Contoh yang sederhana adalah misalnya dalam sistem operasi linux terdapat *file password* yang digunakan untuk menyimpan nama penggunanya. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 5).

## II.7. MySQL

*MySQL* adalah suatu perangkat lunak database relasi (*relational database management system* atau RDBMS). Seperti halnya *Oracle*, *MS SQL*, dan sebagainya. Jangan disalah artikan *MySQL* dengan *SQL*, *SQL* sendiri adalah suatu sintaks perintah-perintah tertentu atau bahasa pemrograman yang digunakan untuk mengelola suatu *database*. Jadi *MySQL* dan *SQL* adalah dua makhluk yang berbeda. Mudah-mudahan *MySQL* adalah *softwarena* dan *SQL* adalah bahasa perintahnya. *Software database* mulai bermunculan seiring dengan bertambahnya kebutuhan akan *database server*. Salah satu dari pendatang baru dalam dunia *database* adalah *MySQL*, sebuah *server/klien database SQL* yang berasal dari Skandinavia. *MySQL* terdiri atas *server SQL*, klien program untuk mengakses *server*, *tools* untuk administrasi dan *interface* program untuk menulis program sendiri.

Pengembangan *MySQL* sendiri dimulai pada tahun 1979 dengan *tools database UNIREG* yang dibuat oleh Michael “Monty” Widenius untuk perusahaan TcX di Swedia. Kemudian pada tahun 1994, TcX mulai mencari *server SQL* untuk mengembangkan aplikasi *web*. Mereka menguji beberapa *server* komersial namun semuanya masih terlalu lambat untuk tabel-tabel TcX yang besar

Tahun 1995 David Axmark dari Detro HB berusaha menekan TcX untuk merelease *MySql* di internet. Ia juga membuat dokumentasi *MySql* yang di *build* untuk *GNU configure utility*. *MySql* 3.11.1 dipublikasikan didunia pada tahun 1996 dan didistribusikan untuk *Linux* dan *Solaris*. Sekarang ini *MySql* bekerja untuk banyak *platform* serta tersedia *source codenya*. *MySql* memiliki banyak hal yang bisa ditawarkan yaitu :

1. Kecepatan

Banyak ahli berpendapat *MySql* merupakan *server* tercepat.

2. Kemudahan Penggunaan

*MySql* punya performa tinggi namun merupakan *database* yang *simple* sehingga mudah *disetup* dan di konfigurasi.

3. Harga

*MySql* cenderung gratis untuk penggunaan tertentu.

4. Mendukung *query language*

*MySql* mengerti bahasa *SQL* yang merupakan sistem *database modern*. Anda juga dapat mengakses *MySql* lewat *protocol ODBC* buatan *Microsoft*.

5. Kapabilitas

Banyak klien dapat mengakses *server* dalam satu waktu. Mereka dapat menggunakan banyak *database* secara simultan.

6. Konektifitas dan *securitas*

*Database MySql* dapat diakses dari semua tempat dinternet dengan hak akses tertentu.

## 7. Pertabilitas

*MySql* dapat berjalan dalam banyak varian *UNIX* dengan baik, sebaik seperti saat berjalan di *system non-UNIX*.

## 8. Distribusi yang terbuka

*MySql* mudah didapatkan dan memiliki *source code* yang boleh disebarluaskan sehingga bisa dikembangkan lebih lanjut. (Jurnal Dwi Apri Setyorini:2009:2-4).

## II.8. Konsep DataBase

*Database* adalah kumpulan beragam elemen informasi yang akan digunakan demi tujuan klasifikasi. Terdapat tiga tingkat arsitektur yang terkait dengan *database* dan sistem manajemen *database* yaitu tingkat konseptual, tingkat logika, dan tingkat fisik.

Dalam konsep perancangan *database* terdapat pendekatan yang perlu diperhatikan yaitu sebagai berikut :

### II.8.1 Normalisasi

Normalisasi merupakan proses pengelompokan elemen data menjadi tabel yang menunjukkan entitas sekaligus relasinya. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 40).

Tahap normalisasi terdiri dari beberapa bentuk :

#### 1. Bentuk Tidak Normal (*Unnormalized Form*)

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti suatu format tertentu, dapat saja data tidak lengkap atau

terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya.

2. Bentuk Normal Kesatu ( 1NF/ *First Normal Form* )

Bentuk normal kesatu mempunyai ciri : setiap data dibentuk dalam *file file* (*file* datar/rata), data dibentuk dalam satu *record* demi *record* dan nilai dari *field* berupa "atomic value". Tidak ada set atribut yang berulang atau atribut bernilai ganda (*multivalued*). Tiap *field* hanya satu pengertian, bukan merupakan kumpulan kata yang mempunyai arti mendua, hanya satu arti saja dan juga bukan pecahan kata sehingga artinya lain. Atom adalah zat terkecil yang masih memiliki sifat induknya, bila dipecah lagi, maka ia tidak memiliki sifat induknya.

3. Bentuk Normal Kedua (2NF/ *Second Normal Form* )

Bentuk normal kedua memiliki syarat : bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci haruslah bergantung fungsi pada kunci utama/*primary key* sehingga untuk membentuk normal kedua haruslah sudah ditentukan kunci *field*. Kunci *field* haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya. Tahap normalisasi terdiri dari beberapa bentuk:

4. Bentuk Normal Ketiga ( 3NF/ *Third Normal Form* )

Untuk menjadi bentuk normal ketiga, relasi haruslah dalam bentuk normal kedua dan semua atribut dalam primer tidak punya hubungan yang transif. Dengan kata lain, setiap atribut bukan kunci haruslah bergantung hanya pada *primary key* dan pada *primary key* secara menyeluruh. Contoh pada bentuk kedua di atas termasuk juga bentuk normal ketiga seluruh atribut yang ada

disitu bergantung penuh pada kunci primernya.

5. Bentuk Normal *Boyce Codd* ( BCNF/ *Boyce Codd Normal Form* )

*Boyce Codd Normal Form* mempunyai paksaan yang lebih kuat dari pada bentuk normal ketiga. Untuk menjadi BCNF, relasi harus dalam bentuk normal kesatu dan setiap atribut harus bergantung fungsi pada atribut *superkey*. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 40).

## II.8.2 Kamus Data

Kamus data ikut berperan dalam perancangan dan pembanguna SI karena peralatan berfungsi untuk :

1. Menjelaskan arti aliran data dan penyimpan dalam penggambaran dala data flow diagram.
2. Mendeskripsikan komposisi paket data yang bergerak melalui aliran, misalnya data alamat diurai menjadi nama jalan,nomor,kota,negara dan kode pos.
3. Menjelaskan spesifikasi nilai dan satuan yang relevan terhadap data yang mengalir dalam sistem tersebut.

**Tabel II.1 Simbol-simbol dalam kamus data**

Simbol	Uraian
=	Terdiri atas,mendefinisikan,diuraikan menjadi,artinya  Contoh : Nama=Sebutan + Nama1+Nama2 + Gelar1 + Gelar2
+	Dan
( )	Optional (pilihan boleh ada atau boleh tidak)

	Contoh : Alamat = Alamat rumah + (Alamat surat)
{ }	Pengulangan Contoh : Nama1 = {Karakter _valid}
[ ]	Memilih salah satu dari sejumlah alternatif,seleksi. Contoh : Sebutan =[ Bapak Ibu Yang Mulia]
**	Komentar Contoh : *Seminar yang akan diikuti*
	Pemisah sejumlah alternatif pilihan antara simbol [ ]

Sumber :( Budi Sutedjo Dharma Oetomo ; 2006 : 118-119)

## II.9. E-R Diagram (ERD)

*ERD* merupakan notasi grafis dalam pemodelan data *konseptual* yang mendeskripsikan hubungan antar penyimpanan. *ERD* juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lainnya dalam dunia nyata. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 18)

*ERD* menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar dua data. Pada dasarnya ada 3 macam simbol yang digunakan, yaitu :

### 1. Entity

*Entity* adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat. Sebagai contoh adalah barang, pemasok, pekerja dan lain-lain.

Seandainya adalah A maka barang A adalah isi dari barang, sedangkan jika B adalah seorang pelanggan maka B adalah isi dari pelanggan. Karena itu harus dibedakan antara entitas sebagai bentuk umum dari deskripsi tertentu dan isi entitas seperti A dan B dalam contoh diatas. Entitas dapat digambarkan dalam bentuk persegi empat.

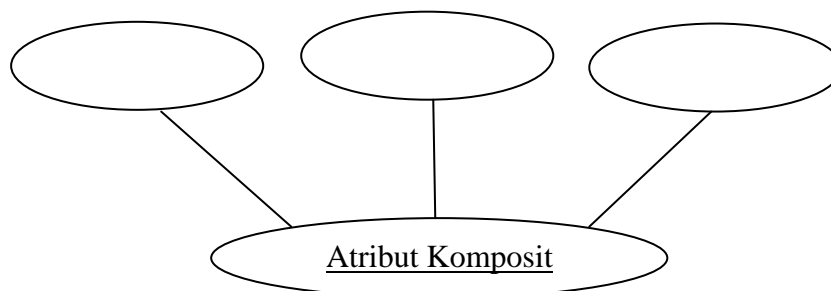


**Gambar II.6. Entitas**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 19)

## 2. *Atribut*

*Entitas* mempunyai elemen yang disebut *atribut* dan berfungsi mendeskripsikan karakter *entitas*, misalnya atribut nama barang dari *entitas* barang. Setiap *ERD* bisa berisi lebih dari satu *atribut*. Penggambaran atribut dalam *ERD* sering hanya digambar berupa elips untuk semua jenis atribut dan dengan garis bawah untuk atribut sebagai kunci.

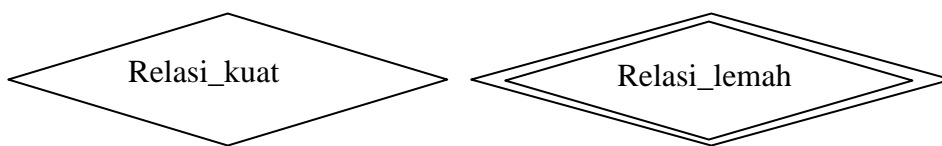


**Gambar II.7 Atribut**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 23)

### 3. Hubungan / *relationship*

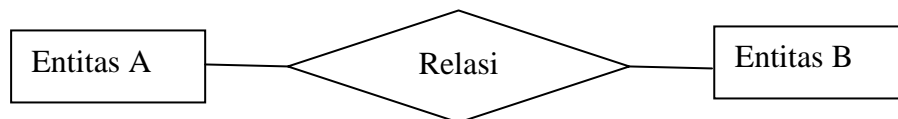
Belah ketupat merupakan penggambaran hubungan (relasi) antarentitas atau sering disebut kerelasian. Ada dua macam penggambaran relasi yaitu relasi kuat dan relasi lemah. Relasi kuat biasanya menghubungkan antarentitas kuat, sedangkan relasi lemah untuk menghubungkan antara entitas kuat dengan entitas lemah. *Relationship* digambarkan sebagai berikut :



**Gambar II.8. Relationship**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 24)

Untuk menghubungkan entitas-kerelasian-entitas digunakan garis lurus seperti gambar berikut ini :

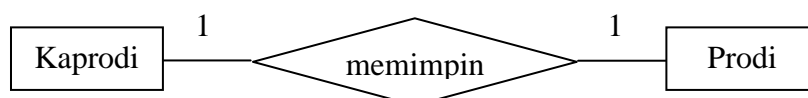


**Gambar II.9. Kerelasian antarentitas**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 24)

Jenis-jenis hubungan:

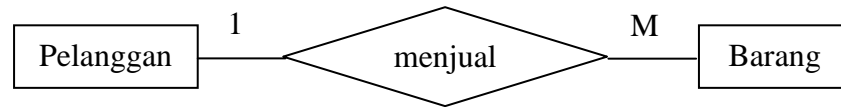
- a. Satu ke satu, misalnya suatu perusahaan mempunyai aturan satu supir hanya boleh menangani satu kendaraan karena alasan tertentu.



**Gambar II.10. Relational 1 to 1**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 24)

- b. Satu ke banyak atau banyak ke satu, misalnya suatu perusahaan selalu berasumsi bahwa satu pelanggan dapat membeli banyak barang



**Gambar II.11. Relational 1 to Many**

Sumber : (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 25)

## II.10. Web Server

Web server adalah sebuah aplikasi server yang melayani permintaan HTTP atau HTTPS dari browser yang mengirimkannya kembali dalam bentuk halaman-halaman web. Halaman-halaman web yang dikirim web server biasanya berupa file-file HTML dan CSS yang nantinya akan di parsing dan ditata oleh browser sehingga menjadi halaman-halaman web yang bagus dan mudah dibaca.