

## **BAB II**

### **TINJAUAN PUSTAKA**

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

Analisis Sistem adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka. Analisis sistem merupakan tahapan yang paling awal dari pengembangan sistem yang menjadi fondasi menentukan keberhasilan sistem informasi yang dihasilkan nantinya. (Hanif Al Fatta ; 2007 : 44)

#### **II.2. Knowledge Management**

Sebagai langkah awal untuk memberikan pemahaman dalam manajemen pengetahuan (*knowledge management*), diperlukan peninjauan berbagai literatur. Para pakar dan ahli dalam mengkonsepsikan formulasi definisi satu sama lain berbeda-beda, hal tersebut karena perbedaan sudut pandang dari masing-masing pakar dan ahli.

*Knowledge Management* adalah usaha untuk meningkatkan pengetahuan yang berguna dalam organisasi, diantaranya membiasakan budaya berkomunikasi antar personil, memberikan kesempatan untuk belajar, dan menggalakan saling berbagi *knowledge*. Dimana usaha ini akan menciptakan dan mempertahankan peningkatan nilai dari inti kompetensi bisnis dengan memanfaatkan teknologi informasi yang ada. (Winda & Ken ; 2014 : 682)

Konsep dan definisi manajemen pengetahuan, antara lain dikemukakan oleh Davidson dan Philip Voss (2002), Manajemen pengetahuan sebagai system yang memungkinkan perusahaan menyerap pengetahuan, pengalaman dan kreativitas para stafnya untuk perbaikan perusahaan. Menurut pendapat Betgerson (2003), manajemen pengetahuan merupakan suatu pendekatan yang sistematis untuk mengelola asset intelektual dan informasi lain sehingga memberikan keunggulan bersaing bagi perusahaan. (Ismail Nawawi ; 2012 : 2)

Dalam memperkaya pemahaman, Tannebaum (1998) memberikan definisi dengan berbagai formulasi untuk memberikan pemahaman terhadap manajemen pengetahuan sebagai berikut :

1. Manajemen pengetahuan mencakup pengumpulan, penyusunan, penyimpanan, dan pengaksesan informasi untuk membangun pengetahuan, pemanfaatan dengan tepat teknologi informasi, seperti computer yang dapat mendukung manajemen pengetahuan, namun teknologi informasi tersebut bukanlah manajemen pengetahuan.
2. Manajemen pengetahuan mencakup berbagai pengetahuan (*sharing knowledge*). Tanpa berbagi pengetahuan, upaya manajemen pengetahuan akan gagal culture perusahaan, dinamika dan praktik, seperti system penggajian dapat mempengaruhi berbagai pengetahuan. Kultur dan aspek social dari manajemen pengetahuan merupakan tantangan yang signifikan.
3. Manajemen pengetahuan terkait dengan pengetahuan orang. Pada suatu saat, organisasi membutuhkan orang yang kompeten untuk memahami dan memanfaatkan informasi dengan efektif. Organisasi terkait dengan individu

untuk melakukan inovasi dan memberi petunjuk kepada organisasi. Organisasi juga terkait dengan persoalan keahlian yang menyediakan *input* untuk menerapkan manajemen pengetahuan. Oleh karena itu, organisasi mesti mempertimbangkan bagaimana menarik, mengembangkan, dan mempertahankan pengetahuan anggota sebagai bagian dari *domain* manajemen pengetahuan.

4. Manajemen pengetahuan terkait dengan peningkatan efektivitas organisasi. Kita berkonsentrasi dengan manajemen pengetahuan karena dipercaya bahwa manajemen pengetahuan dapat memberikan kontribusi kepada vitalitas dan kesuksesan perusahaan. Upaya untuk mengukur modal intelektual dan untuk menilai efektivitas manajemen pengetahuan harus dapat membantu kita memahami secara luas pengelolaan pengetahuan yang telah dilakukan. (Ismail Nawawi ; 2012 : 2)

Selain mengusulkan satu consensus mengenai pengertian manajemen pengetahuan, Tannebaum juga memberikan penjelasan mengenai karakteristik berbagai aktivitas manajemen pengetahuan. Manajemen pengetahuan, menurut Tannebaum, paling tidak terdiri atas berikut ini.

1. Pengembangan *database* organisasi mengenai pelanggan, masalah yang bersifat umum dan serta pemecahannya.
2. Mengenali para ahli internal, memperjelas apa yang mereka ketahui, dan mengembangkan kamus yang menjelaskan sumber daya internal kunci dan mengenali bagaimana menemukannya.

3. Mendapatkan dan menangkap pengetahuan dari para ahli tersebut untuk disebar ke yang lain.
4. Mendesain struktur pengetahuan yang membantu mengelola informasi dalam suatu cara yang dapat diakses dan siap untuk diaplikasikan.
5. Menciptakan forum bagi orang-orang yang ada di dalam perusahaan untuk berbagi pengalaman dan ide, baik dalam bentuk tatap muka, berkomunikasi melalui *internet, website, chatting room, e-mail*, dan lain-lain.
6. Memanfaatkan *groupware* sehingga memungkinkan berbagai macam orang di lokasi yang berbeda dapat berkomunikasi untuk menyelesaikan masalah secara bersama-sama dan mencatat informasi di dalam suatu *domain* pengetahuan yang telah dipilih.
7. Bertindak untuk mengenali, mempertahankan talenta orang-orang yang memiliki pengetahuan yang diperlukan di dalam bidang kegiatan utama bisnis.
8. Mendesain pelatihan dan aktifitas pengembangan lainnya untuk menilai dan mengembangkan pengetahuan internal.
9. Menerapkan praktik penghargaan pengakuan dan promosi yang mendorong berlangsungnya kegiatan berbagi informasi antar anggota maupun antar unit dalam organisasi.
10. Membantu pekerjaan serta menyediakan alat-alat yang mendukung kinerja sehingga memungkinkan setiap orang menilai dan menerapkan pengetahuan apabila diperlukan.
11. Memaknai *database* pelanggan, produk, transaksi, atau hasil dengan mengenali kecenderungan dan menggali informasi sebanyak mungkin.

12. Mengukur modal intelektual di dalam upaya mengelola pengetahuan yang lebih baik.

13. Menangkap dan menganalisis informasi yang terkait dengan perhatian pelanggan, pilihan-pilihan, dan kebutuhan dari lapangan, *front line* atau personil bagian pelayanan didorong untuk mampu memahami dengan lebih baik terhadap kecenderungan pelanggan. (Ismail Nawawi ; 2012 : 2-3)

Di pihak lain, ada yang mengkonsepsikan dengan formulasi definisi dikaitkan dengan komponen krisis bahwa manajemen pengetahuan (*knowledge management*) yang sukses tidak hanya karena komputerisasi yang impresif, tetapi sebaiknya ditinjau dari ketiga komponen yang kritis berikut.

1. Alur *knowledge* yang benar dan sumber yang dilimpahkan ke organisasi / institusi.
2. Teknologi tepat yang disimpan dan dapat mengomunikasikan *knowledge* tersebut.
3. Budaya tempat kerja yang benar, sehingga karyawan termotivasi untuk memanfaatkan *knowledge*. (Ismail Nawawi ; 2012 : 2)

Oleh karena itu, manajemen pengetahuan (*knowledge management*) akan sukses apabila terjadi interaksi di antara komponennya dan tidak terjadi tumpang tindih (*overlap*) dari ketiga komponen tadi. Meskipun demikian, *knowledge management* memberikan kesempatan pada organisasi tersebut untuk :

1. Menangkap dan menganalisa informasi organisasi dan diaplikasikan secara strategis dalam bentuk *warehousing* dan *dataming*, system pendukung

keputusan (*Decision System Support/DSS*), serta system informasi eksekutif (EIS).

2. Menciptakan proses untuk akses informasi ke seluruh dunia melalui intranet, *groupware*, dan sistem pendukung keputusan kelompok (Group DSS) agar karyawan mendapat informasi secara tepat, informative dan inovatif, menjadikan kekuatan pendorong dari *knowledge* yang terakumulasi dari pengalaman masa lalu seluruh organisasi
3. Membangun dan menyelesaikan proyek dengan meningkatkan kecepatan, ketangkasan, dan keselamatan. (Ismail Nawawi ; 2012 : 4)

### **II.3. Impor**

Impor dapat diartikan sebagai pembelian barang dan jasa dari luar negeri ke dalam negeri dengan perjanjian kerjasama antara 2 negara atau lebih. Impor juga bisa dikatakan sebagai perdagangan dengan cara memasukkan barang dari luar negeri ke wilayah Indonesia dengan memenuhi ketentuan yang berlaku (Hutabarat, 1996:403). Impor adalah proses transportasi barang atau komoditas dari suatu negara ke negara lain secara legal, umumnya dalam proses perdagangan. Proses impor umumnya adalah tindakan memasukan barang atau komoditas dari negara lain ke dalam negeri. Impor barang secara besar umumnya membutuhkan campur tangan dari bea cukai di negara pengirim maupun penerima. Impor adalah bagian penting dari perdagangan internasional. Kegiatan impor dilakukan untuk memenuhi kebutuhan rakyat. Produk impor merupakan barang-barang yang tidak dapat dihasilkan atau negara yang sudah dapat

dihasilkan,tetapi tidak dapat mencukupi kebutuhan rakyat (Jurnal EMBA Vol.1 No.4 Desember 2013, Hal. 1406).

Kegiatan importasi, harus diikuti oleh pemenuhan kewajiban kepabeanan, seperti penyampaian pemberitahuan pabean, pembayaran bea masuk dan pajak dalam rangka impor dan kelengkapan dokumen - dokumen yang diperlukan dalam penelitian dokumen dan pemeriksaan fisik atas barang. Kegiatan importasi ini tidak terlepas dari ketentuan - ketentuan dalam perdagangan, angka pengenal importer (API) yaitu, tanda pengenal sebagai importir yang harus dimiliki setiap perusahaan yang melakukan perdagangan impor (Ali Purwito & Indriani : 2015 : 10)

#### **II.4. PHP**

PHP pertama kali ditemukan pada 1995 oleh seorang Software Developer bernama Rasmus Lerdorf. Ide awal PHP adalah ketika itu Rasmus ingin mengetahui jumlah pengunjung yang membaca resume onlinenya. Script yang dikembangkan baru dapat melakukan 2 pekerjaan, yakni merekam informasi visitor, dan menampilkan jumlah pengunjung dari suatu website. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia web saat ini. Kemudian, dari situ banyak orang dimilisi mendiskusikan script buatan Rasmus Lerdorf, hingga akhirnya Rasmus mulai membuat sebuah tool/script, bernama Personal Home Page (PHP). (Loka Dwiartara ; 2010 : 4)

Kebutuhan PHP sebagai tool yang serba guna membuat Lerdorf melanjutkan untuk mengembangkan PHP hingga menjadi suatu bahasa tersendiri yang mungkin dapat mengkonversikan data yang di inputan melalui Form HTML

menjadi suatu variable, yang dapat dimanfaatkan oleh sistem lainnya. (Loka Dwiartara ; 2010 : 4)

PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML.

Kode PHP mempunyai ciri – ciri khusus, yaitu :

- a. Hanya dapat dijalankan menggunakan web server, misal : Apache.
- b. Kode PHP diletakkan dan dijalankan di web server.
- c. Kode PHP dapat digunakan untuk mengakses database, seperti : MySQL, PostgreSQL, Oracle, dan lain – lain.
- d. Merupakan Software yang bersifat *open source*.
- e. Gratis untuk di-*download* dan digunakan.
- f. Memiliki sifat *multiplatform*, artinya dapat dijalankan menggunakan sistem operasi apapun, seperti : Linux, Unix, Windows, dan lain – lain.

(Diar Puji Oktavian ; 2010 : 31)

Sedangkan karakteristik script PHP dapat diuraikan sebagai berikut :

- a. File PHP disimpan dengan extensi filenya yaitu : \*.php3, \*.php4, \*.php
- b. Script PHP biasanya diawali dengan tag ‘<?’ atau ‘<?php’ dan ditutup dengan tag ‘?>’
- c. File PHP dapat menginduk atau disisipkan pada bahasa script lainnya atau dapat berdiri sendiri. Contoh skrip PHP yang disisipkan pada HTML :

```

<html>

<head>

<title>Contoh</title>

</head>

<body>

<?php

echo "Hai, Aku adalah skrip PHP!";

?> </body> </html> (Komunitas eLearning IlmuKomputer.com :
2003-2007 : 5 )

```

#### II.4.1 AppServ

Merupakan sebuah *web server* yang berguna untuk menampung *database*. *Appserv* sangat *simple*, efisien untuk web pemula dan ukurannya tidak terlalu besar. Fungsi *Appserv* adalah sebagai *server* yang berdiri sendiri (*localhost*) dan sebagai penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP. *AppServ* terdiri dari dua *software* utama yaitu, *Apache* dan MySQL. Tujuannya adalah mempermudah dalam membuat *database*.

AppServ mempunyai konfigurasi khusus saat penginstallan, Untuk apache menggunakan `httpd.conf`, konfigurasi MySQL menggunakan `my.ini`, dan konfigurasi PHP menggunakan `php.ini`. AppServ memberi garansi bahwa semua isi AppServ dapat berjalan stabil seperti Official Apache, PHP, MySQL Release. (Komunitas eLearning IlmuKomputer.com : 2003-3007 : 18)

#### II.5. Basis Data (*database*)

Menurut James Martin (2009), Basis data adalah suatu kumpulan data

terhubung yang disimpan secara bersama-sama pada suatu media tanpa mengatap satu sama lain atau tidak perlu suatu kerangkaan data dengan cara-cara tertentu sehingga mudah untuk digunakan dan ditampilkan kembali, dapat digunakan untuk satu atau lebih program aplikasi secara optimal, data dapat disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya, serta disimpan sedemikian rupa sehingga penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol. (Kusrini , 2007:140)

Adapun elemen - elemen sistem manajemen basis data adalah sebagai berikut:

1. *Database*

*Database* adalah kumpulan dari item data yang saling berhubungan satu sama lain, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer, dan harus menggunakan *software* untuk melakukan manipulasi tertentu

2. *File*

*File* adalah kumpulan *record* sejenis yang mempunyai panjang elemen dan atribut yang sama, namun *value*-nya berbeda. *Database* dibentuk dari kumpulan *file*.

3. *Record*

*Record* adalah kumpulan elemen yang saling berkaitan yang menginformasikan tentang satu entitas secara lengkap. Satu *record* mewakili satu data atau informasi.

4. *Field*

*Field* adalah bagian tertentu dari data dalam *record* yang mewakili satu entitas. Misalnya *file* anggota dapat dilihat dari *field*-nya. Seperti kode anggota, nama dan lain - lain.

#### 5. *Data Value*

*Data Value* adalah data actual atau informasi yang disampaikan pada setiap data elemen atau *field* data.

#### 6. *Entity*

*Entity* adalah objek riil yang dapat dibedakan satu sama lain dan tidak saling bergantung. Misal, pada bidang sirkulasi, entitasnya adalah anggota dan buku.

#### 7. *Query*

*Query* merupakan perintah yang dirancang untuk memanggil kelompok *record* tertentu dari satu file atau lebih untuk melakukan operasi pada *file*.

#### 8. *View*

*View* adalah data yang terdiri atas sejumlah record yang diproses urutan penampilan. (Kusrini ,2007:142-143)

### **II.6. MySQL**

MySQL adalah Database. Database sendiri merupakan suatu jalan untuk menyimpan berbagai informasi dengan membaginya berdasarkan kategori-kategori tertentu. Dimana informasi – informasi tersebut saling berkaitan, satu dengan yang lainnya. (Loka Dwiartara ; 2010 : 6)

MySQL (*My Structure Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS dari

sekian banyak DBMS seperti *Oracle*, *MS SQL*, *Postagree SQL* dan lain – lain. MySQL merupakan DBMS yang multi-user yang bersifat gratis dibawah lisensi GNU *General Public Licence* (GPL). Tidak seperti *Apache* yang merupakan perangkat lunak yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing – masing. MySQL dimiliki dan disponsori oleh sebuah perusahaan Swedia, yaitu MySQLAB. MySQLAB memegang hak cipta kode sumbernya. Seperti yang telah disebutkan sebelumnya MySQL bersifat gratis atau *open source* sehingga kita bisa menggunakannya secara gratis (Anhar ; 2010 : 21).

MySQL memiliki beberapa kelebihan, antara lain :

1. MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaa tanpa mengalami masalah.
2. MySQL memiliki kecepatan yang bagus dalam menangani query sederhana.
3. MySQL memiliki operator dan fungsi secara penuh dan mendukung perintah *Select* dan *Where* dalam perintah query.
4. MySQL memiliki keamanan yang bagus karena beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perijinan yang mendetail serta sandi terenkripsi.
5. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta kurang lebih 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

6. MySQL dapat melakukan koneksi dengan client menggunakan protocol TCP/IP, Unix socket (UNIX), atau Named Pipes (NT).
7. MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari 20 bahasa.
8. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
9. MySQL didistribusikan secara *open source*, di bawah lisensi GPL sehingga dapat digunakan secara gratis. (MADCOMS : 2011 : 140)

### **II.6.1. PHPMyAdmin**

PhpMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL melalui Jejaring Jagat Jembar (World Wide Web). phpMyAdmin mendukung berbagai operasi MySQL, diantaranya (mengelola basis data, tabel-tabel, bidang (fields), relasi (relations), indeks, pengguna (users), perijinan (permissions), dan lain-lain).

Saat ini banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola basis data dalam MySQL, salah satunya adalah phpMyAdmin. Dengan phpMyAdmin, seseorang dapat membuat database, membuat tabel, mengisi data, dan lain-lain dengan mudah, tanpa harus menghafal baris perintahnya. PhpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada di komputer. Untuk membukanya, buka browser lalu ketikkan alamat <http://localhost/phpmyadmin>, maka akan muncul halaman phpMyAdmin. Di situ

nantinya seseorang bisa membuat (create) basis data baru, dan mengelolanya. (Komunitas eLearning IlmuKomputer.com : 2003-2007 : 15)

## **II.7. Dreamweaver dan Web**

Macromedia adalah aplikasi standar industry yang digunakan untuk membuat halaman web. Kemasyhuran dreamweaver tidak terlepas dari pesatnya kemajuan dunia internet sekarang ini. Internet memungkinkan semua penduduk dunia saling terhubung secara instan melalui komputer yang terhubung ke jaringan internet. (Ali kabar:2005:1)

Dreamweaver adalah sebuah HTML editor professional untuk mendesain web secara visual dan mengelola situs atau halaman web. Dreamweaver merupakan software utama yang digunakan oleh Web Designer maupun Web Programmer dalam mengembangkan suatu situs web, karena Dreamweaver mempunyai ruang kerja, fasilitas dan kemampuan yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs web. (MADCOMS : 2011 : 13)

Halaman web merupakan salah satu wahana yang digunakan untuk berkomunikasi di dunia internet. Halaman web menjadi terkenal karena halaman web ditulis menggunakan format teks murni (*plain text*) yang konsekuensi ukurannya akan sangat kecil sehingga sangat efisien digunakan di jaringan internet. Halaman web ditulis menggunakan format dasar HTML yang singkatannya adalah Hypertext Markup Language. Selain HTML, masih ada beberapa bahasa lain seperti PHP, ASPX, CGI, JSP, dan lain sebagainya.

## II.8. UML (*Unified Modelling Language*)

UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Adi Nugroho, 2010 : 6-7).

**Tabel II.1. View dan Diagram dalam UML**

Major Area	View	Diagrams	Main Concepts
Structural	Static view	Class diagram	Class, association, generalization, dependency, realization, interface.
	Use case view	Use case diagram	Use case, actor, association, extend, include, use case generalization
	Implementation view	Component diagram	Component, interface, dependency, realization
	Deployment view	Deployment diagram	Node, component, dependency, location
Dynamic	State machine	Statechart diagram	State, event, transition, action
	Activity view	Activity diagram	State, activity, completion transition, fork, join
	Interaction view	Sequence diagram	Interaction, object, message, activation
		Collaboration	Collaboration,

		diagram	interaction, collaboration role, message
Model management	Model management view	Class diagram	Package, subsystem, model
Extensibility	All	All	Constraint, stereotype, tagged values

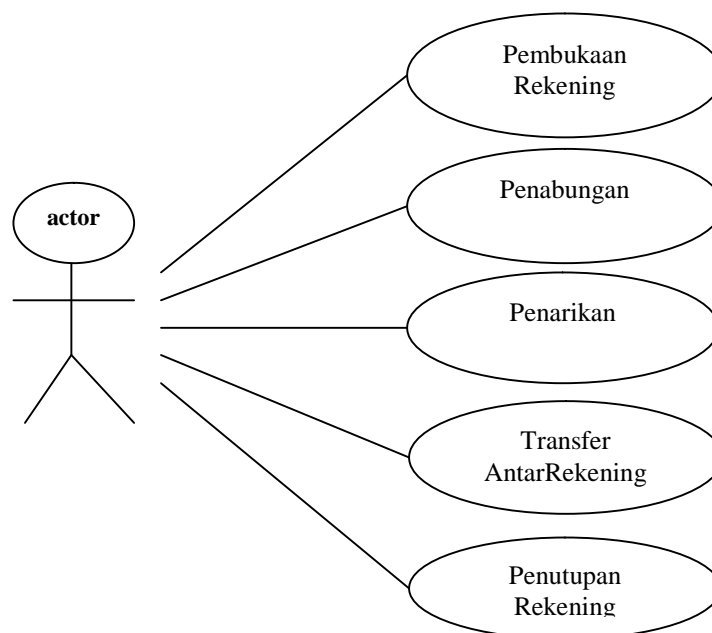
(Sumber : Adi Nugroho ; 2010 : 9-10)

### II.8.1. Use Case Diagram

*Use Case* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. Use case bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut *scenario*. Setiap *scenario* mendeskripsikan urutan kejadian. Setiap urutan diinisialisasikan oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan demikian secara singkat bisa dikatakan use case adalah serangkaian *scenario* yang digabungkan bersama-sama oleh tujuan umum pengguna.

Dalam konteks UML, tahap konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh penggunanya. Selanjutnya, *use case diagram* tidak hanya sangat penting pada saat analisis, tetapi juga sangat penting dalam tahap perancangan (*design*), untuk mencari kelas-kelas yang terlibat dalam aplikasi, dan untuk melakukan pengujian (*testing*).

Saat akan mengembangkan *use case diagram*, hal yang pertama kali harus dilakukan adalah mengenali *actor* untuk sistem yang sedang dikembangkan. Dalam hal ini, ada beberapa karakteristik untuk para *actor*, yaitu *actor* yang ada di luar sistem yang sedang dikembangkan dan *actor* yang berinteraksi dengan sistem yang sedang dikembangkan. (Adi Nugroho ; 2009 : 7 )



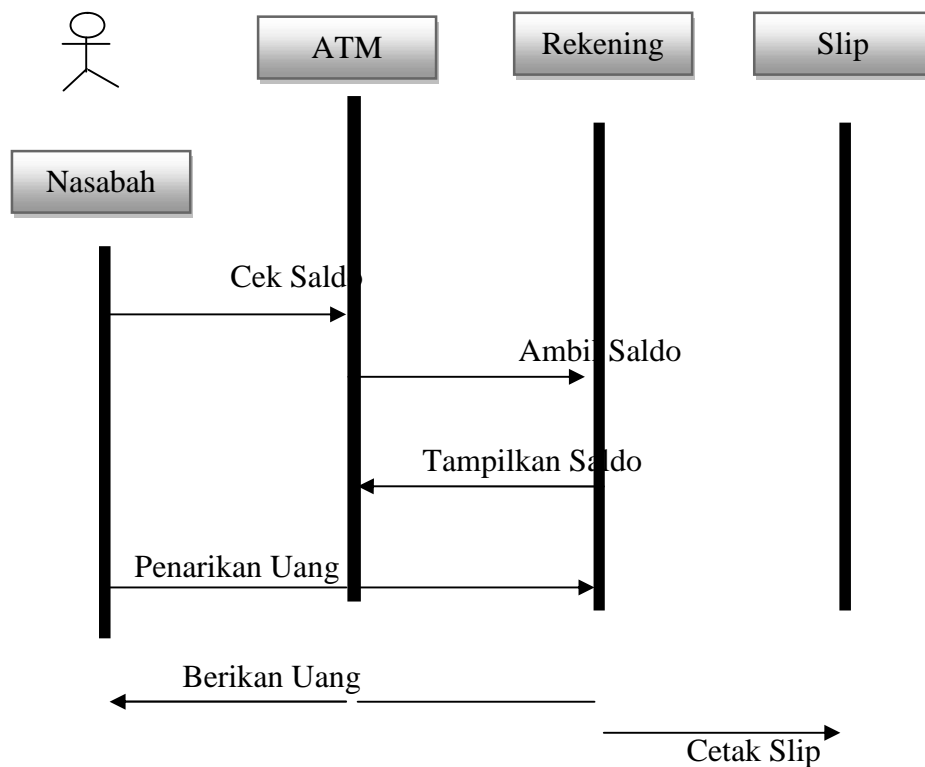
**Gambar II.1. Contoh Use Case Diagram**  
(Sumber : Adi Nugroho ; 2009 : 8)

### II.8.2. Sequence Diagram

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

*Sequence diagram* juga menampilkan interaksi antar suatu kelas dengan kelas yang lainnya, bagaimana suatu *message* (pesan) dikirimkan dari suatu kelas

ke kelas yang lainnya, dengan penekanan lebih pada urutan kejadian menurut waktu. Keunggulan dari *Sequence diagram* memperlihatkan dengan baik urutan interaksi yang terjadi antara suatu kelas dengan kelas lainnya, tetapi mengabaikan pengorganisasiannya. (Adi Nugroho ; 2009 : 101)



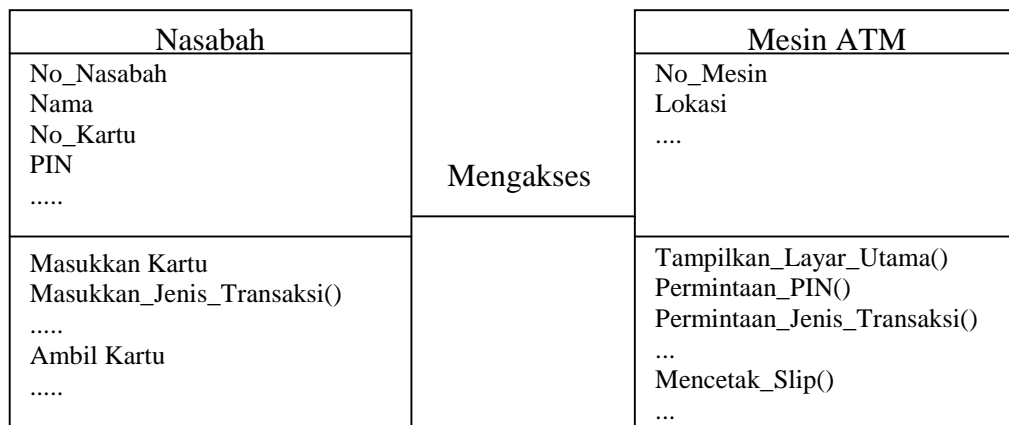
**Gambar II.2. Sequence Diagram**  
(Sumber : Adi Nugroho ; 2009 : 102)

### II.8.3. Class Diagram

*Class* didefinisikan sebagai kumpulan/himpunan objek yang memiliki kesamaan dalam atribut/properti, perilaku (operasi), serta cara berhubungan dengan objek lain. (Adi Nugroho ; 2009 : 18)

Selain itu, kita juga mendefenisikan objek sebagai konsep, abstraksi dari sesuatu dengan batas nyata, sehingga kita dapat menggambarkan secara sistematis. Pemahaman objek memiliki dua fungsi, yaitu :

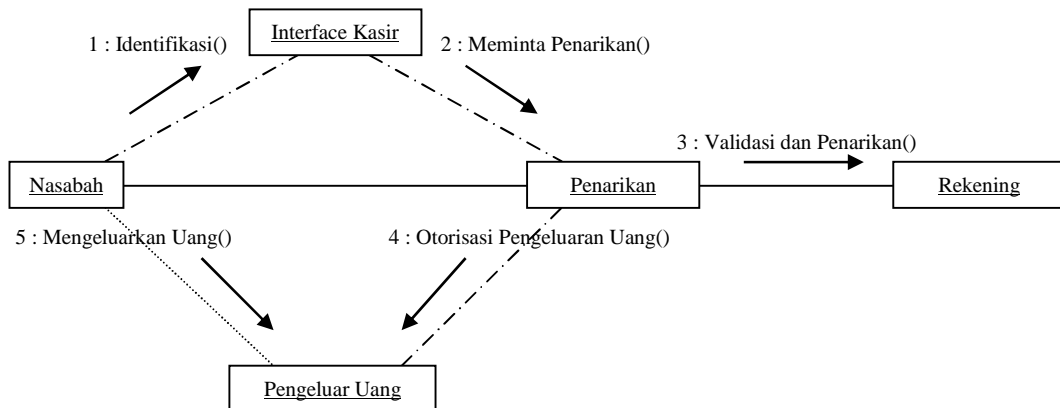
- a. Memudahkan untuk mempelajari secara seksama hal-hal yang ada di dunia nyata.
- b. Menyediakan suatu dasar yang kuat dalam implementasi ke dalam sistem terkomputerisasi. (Adi Nugroho ; 2009 : 17)



**Gambar II.3. Contoh Class Diagram**  
(Sumber : Adi Nugroho ; 2009 : 39)

#### II.8.4. Collaboration Diagram

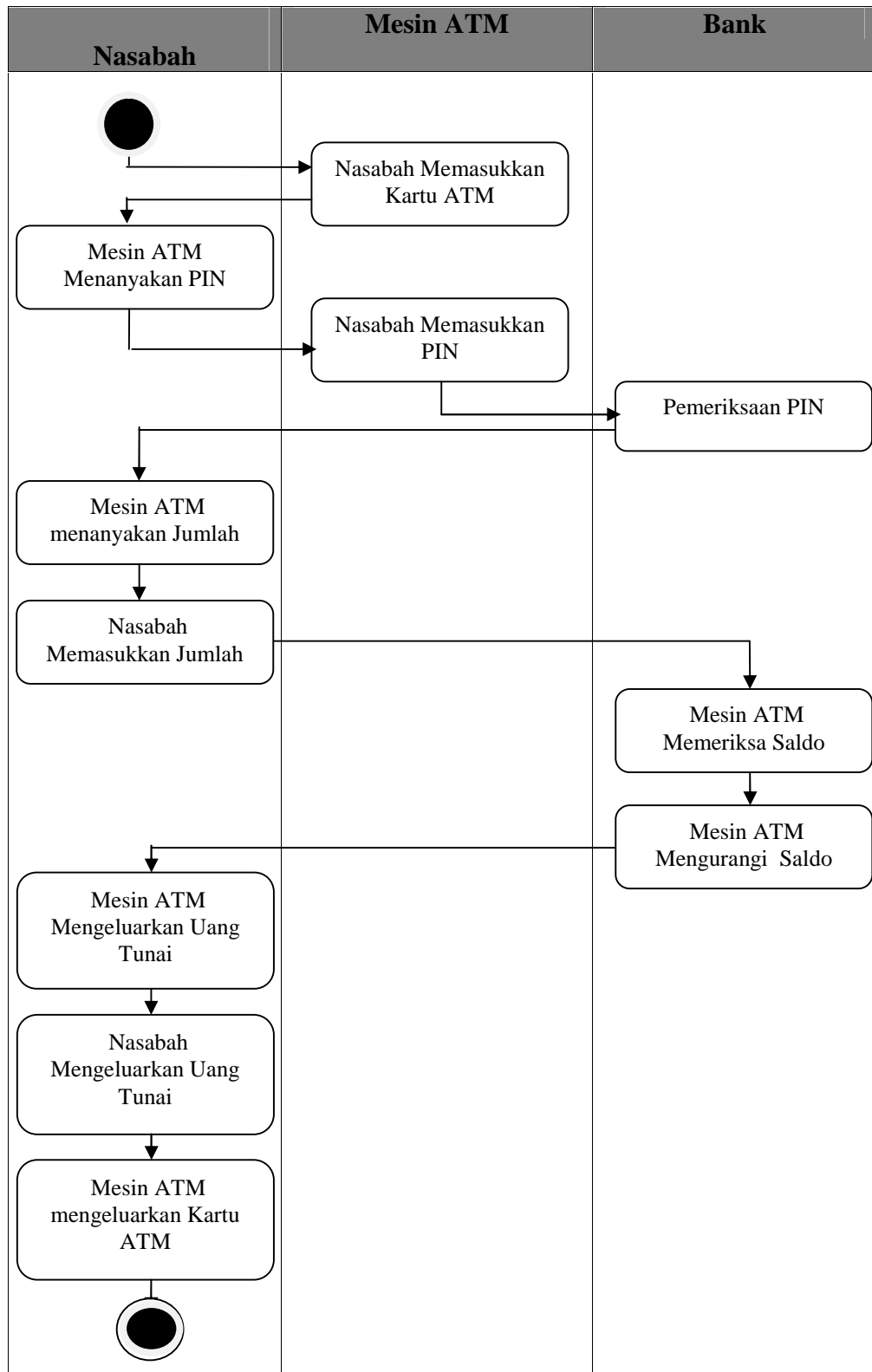
Collaboration diagram pada dasarnya merupakan diagram kelas yang memuat peran-peran pengklasifikasi dan peran-peran asosiasi, alih-alih hanya menampilkan pengklasifikasi-pengklafikasi serta asosiasi-asosiasi. Peran pengklasifikasi dan peran asosiasi mendeskripsikan konfigurasi objek-objek dan tautan-tautan yang mungkin terjadi saat suatu *instance* kolaborasi dieksekusi. (Adi Nugroho ; 2010 : 44)



**Gambar II.4. Contoh Collaboration Diagram**  
(Sumber : Adi Nugroho ; 2010 : 44)

### II.8.5. Activity View

Diagram aktivitas (activity diagram) sesungguhnya merupakan bentuk khusus dari state machine yang bertujuan memodelkan komputasi-komputasi dan aliran-aliran kerja yang terjadi dalam sistem/perangkat lunak yang sedang dikembangkan. Biasanya, suatu diagram aktivitas mengasumsikan komputasi-komputasi dilaksanakan tanpa adanya interupsi-interupsi eksternal berbasis event terjadi padanya (Adi Nugroho, 2010 : 62).



**Gambar II.5. Contoh Activity Diagram**  
*(Sumber : Adi Nugroho ; 2009 : 11)*

## **II.9. Normalisasi**

Normalisasi merupakan cara pendekatan dalam membangun desain logika basis data relasional yang tidak secara langsung berkaitan dengan model data, tetapi dengan menerapkan sejumlah aturan dan kriteria standar untuk menghasilkan struktur table yang normal. (Kusrini : 2008 : 40)

Dalam perspektif normalisasi sebuah database dikatakan baik jika setiap table yang membentuk basis data sudah berada dalam keadaan normal. suatu tabel dikatakan normal, jika :

- a. Jika ada dekomposisi/penguraian tabel, maka dekomposisinya dijamin aman (lossless-join decomposition)
- b. Terpeliharanya ketergantungan functional pada saat perubahan data (dependency preservation)
- c. Tidak melanggar Boyce Code Normal Form (BCNF), jika tidak bisa minimal tidak melanggar bentuk normalisasi ketiga. (Kusrini : 2008 : 40)

### **II.9.1. Bentuk – Bentuk Normalisasi**

#### **a. Bentuk Tidak Normal**

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja tidak lengkap dan terduplikasi. data dikumpulkan apa adanya sesuai keadaannya. (Kusrini : 2008 : 41)

#### **b. Bentuk Normal Pertama (1NF/First Normal Form)**

Suatu tabel disebut 1NF jika :

- Tidak ada baris yang duplikat dalam tabel tersebut
- Masing – masing cell bernilai tunggal. (Kusrini : 2008 : 41)

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

Bentuk normal kedua terpenuhi jika pada sebuah tabel semua atribut yang tidak termasuk dalam primary key memiliki ketergantungan fungsional pada primary key secara utuh. sebuah tabel dikatakan tidak memenuhi 2NF, jika ketergantungannya hanya bersifat parsial (hanya tergantung pada sebagian dari primary key). (Kusrini : 2008 : 41)

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

Bentuk normal ketiga terpenuhi jika untuk setiap ketergantungan fungsional dengan notasi  $X \rightarrow A$ , dimana A mewakili semua atribut tunggal di dalam tabel yang tidak ada dalam X, maka :

- X haruslah superkey pada tabel tersebut.
- Atau A merupakan bagian dari primary key pada tabel tersebut.

(Kusrini : 2008 : 42)

**e. Bentuk Normal Keempat Dan Kelima**

Penerapan aturan normalisasi sampai bentuk normal ketiga sudah memadai untuk menghasilkan tabel berkualitas baik. Namun demikian, terdapat pula bentuk normal keempat (4NF) dan kelima (5NF). Bentuk normal keempat berkaitan dengan sifat ketergantungan banyak nilai (multiple dependency) pada suatu tabel yang merupakan pengembangan dari ketergantungan fungsional. Adapun bentuk normal kelima merupakan nama lain dari Project Join Normal Form (PJNF). (Kusrini : 2008 : 43)

**f. Bentuk Normal Boyce - Codd (BCNF/ Boyce - Codd Normal Form)**

Bentuk normal ini terpenuhi jika :

- Memenuhi 1NF
- Relasi harus bergantung fungsi pada atribut superkey. (Kusrini : 2008 : 43)

Contoh kasus :

Faktur Pembelian Barang				
PT. Ayo Bersama Jl. Senopati 12 yk				
Kode Suplier : G01			Tanggal : 07/10/04	
Nama Suplier : Gober Nustra			Nomor : 004	
Kode	Nama Brg	Qty	harga	Jumlah
A01	AC Model 1	10	1000000	10000000
A02	AC Model 2	5	900000	4500000
			Total Faktur	14500000
Jatuh Tempo 15/10/04				

Faktur Pembelian Barang				
PT. Ayo Bersama Jl. Senopati 12 yk				
Kode Suplier : A03			Tanggal : 07/10/04	
Nama Suplier : Angkasa			Nomor : 006	
Kode	Nama Brg	Qty	harga	Jumlah
B01	Kursi	5	1000000	500000
A03	Meja	7	250000	3250000
			Total Faktur	3750000
Jatuh Tempo 17/10/04				

**Gambar II.6 Contoh Normalisasi Faktur Pembelian Barang**  
(Sumber : Kusrini : 2008 : 44)

Contoh bentuk tidak ternormalisasi

**Tabel II.2. Tabel Bentuk Tidak Ternormalisasi**

Kd. Fak	Kode Sup	Nama Sup	Kode Brg	Nama Brg	Tgl	....
004	G01	Gobel nustra	A01, A02	AC Model 1, Ac Model 2	07/10/04	...
006	A03	Angkasa	B01, A03	Meja, Kursi	09/10/05	...

...	J.tempo	Qty	Hrg	Jumlah	Tot
004	G01	Gobel nustra	A01, A02	AC Model 1, Ac Model 2	07/10/04
006	A03	Angkasa	B01, A03	Meja, Kursi	09/10/05

*(Sumber : Kusrini : 2008 : 45)*

Bentuk Normal I

**Tabel II.3. Tabel Bentuk Normal I (1NF)**

Kd. Fak	Kode Sup	Nama Sup	Kode Brg	Nama Brg	....
004	G01	Gobel nustra	A01	AC Model 1	...
004	G01	Gobel nustra	A01	AC Model 2	...
006	A03	Angkasa	B01	Meja	
006	A03	Angkasa	A03	Kursi	

....	tgl	J. Tempo	Qty	Hrg
...	07/10/04	15/10/04	10	1000000
...	07/10/04	15/10/04	5	900000
	09/10/05	17/10/04	5	100000
	09/10/05	17/10/04	7	250000

*(Sumber : Kusrini : 2008 : 45)*

**Bentuk 2NF :**

KdFaktur : Tgl, JTempo  
 KodeSup, NamaSup  
 KdFaktur, Kodebrg : NamaBrg, Qty, Harga (Kusrini : 2008 : 46)

**Bentuk BCNF :**

KdFaktur : Tgl, JTempo, KodeSup  
 KdSup : NamaSup  
 KdFaktur, Kodebrg : Qty, Harga  
 KodeBrg : NamaBrg (Kusrini : 2008 : 46)

**II.10. Entity Relationship Diagram**

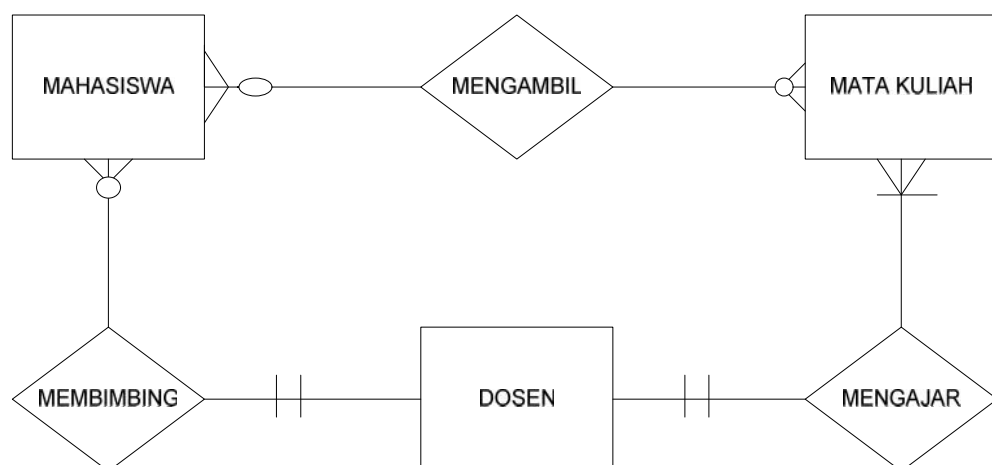
*Entity Relationship Diagram* (ERD) digunakan untuk mengidentifikasi data yang akan diambil, disimpan, dan dipanggil kembali (*retrieve*) untuk keperluan-keperluan tertentu dalam mendukung kegiatan yang dilakukan oleh organisasi. ERD juga digunakan untuk mengidentifikasi asal data yang dibutuhkan dan dilaporkan.

ERD (model data) merupakan alat yang digunakan dalam analisis untuk menggambarkan kebutuhan data dan asumsi – asumsi dalam sistem yang akan dibangun/dikembangkan secara terstruktur dari atas ke bawah. Model data ini juga diatur pada tahapan SDLC dalam mendesain database. Pembuatan ERD

membutuhkan pemahaman terhadap sistem dan komponen – komponen yang menyusunnya.

Untuk mempermudah dalam perancangan database, maka digunakan *Entity Relationship Diagram* (ERD). ERD diutamakan untuk permodelan dari desain konseptual. *Entity Relationship Diagram* menggambarkan struktur dan keterkaitan tabel – tabel data yang menyusun database secara detail. ERD merupakan representasi data sebagai entitas, atribut, dan relasi. (Marimin, dkk : 2006 : 111)

Contoh sebuah model E-R dapat ditunjukkan pada gambar di bawah ini :



**Gambar II.7. Contoh Model E-R**  
*Sumber : (Abdul Kadir; 2009:30)*

Secara garis besar model E-R di atas menerangkan hubungan antara :

- Mahasiswa dan mata kuliah yang diambil,
- Dosen dan mahasiswa yang dibimbing,
- Dosen dan mata kuliah yang diajar.

Dalam hal ini MAHASISWA, MATAKULIAH, dan DOSEN menyatakan tipe entitas dan mengambil, membimbing, dan mengajar menyatakan hubungan.

## II.11. Kamus data

Kamus data (KD) atau data *dictionary* (DD) atau disebut juga dengan istilah sistem data dictionary adalah catalog fakta tentang data dan kebutuhan - kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan KD, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. KD dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis, maupun pada tahap perancangan sistem. Pada tahap analisis, KD dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Pada tahap perancangan sistem, KD digunakan untuk merancang input, merancang laporan - laporan dan database. KD dibuat berdasarkan arus data yang ada di DAD. Arus data di DAD sifatnya adalah global, hanya ditunjukkan nama arus datanya saja. (Jogiyanto:2005:725)

Contoh Kasus :

### ORDER SLIP

Nomor :

Nama Pelanggan :

Alamat :

Tanggal Order :

Untuk dikirimkan pada Tgl :

Alamat Pengiriman :

TOP

HEADER

Nama Barang	No Barang	Jumlah Barang	Harga Satuan	Jumlah	
				Total	

FOOTER

Discount =

Pajak =

Total Akhir =

Jakarta, .....

Kamus Data dari SLIP ORDER :

ORDER = TOP HEADER + ISI + FOOTER \*Slip order valid\*

TOP HEADER = NOMOR + NAMA\_LANGGANAN + ALAMAT +

TGL\_ORDER + UNTUK\_DIKIRIM\_PADA\_TGL +

ALAMAT\_PENGIRIMAN

NOMOR=\*nomor order terdiri dari 8 digit\*

NAMA\_LANGGANAN=(Title) + Nama\_Depan + Nama\_Belakang

ISI = 1 {NO + NAMA\_BARANG + NOMOR\_BARANG +

JUMLAH\_BARANG + HARGA\_SATUAN + JUMLAH } 10

NO = \*nomor unit dari barang yang diorder\*

\*maksimum 10 macam setiap order formulir\*

NOMOR\_BARANG=\*10 digit\*

HARGA\_SATUAN=\*dalam rupiah\*

FOOTER = TOTAL + DISCOUNT + PAJAK + TOTAL\_AKHIR

TOTAL = \*Total dari jumlah yang diorder\*

DISCOUNT=\*Jumlah potongan untuk pembelian > 3 macam\*

PAJAK=\*Jumlah pajak penjualan yang dibayar oleh langganan\*

\*Total dikali prosentase pajak penjualan\*

TOTAL\_AKHIR=\*Jumlah yang harus dibayar langganan\*

\*Total ditambah pajak dikurang discount\*