BAB 2

LANDASAN TEORI

2.1 Teori Umum

Teori umum adalah teori-teori pokok yang digunakan sebagai landasan bagi teori-teori lainnya.

2.1.1 Pengertian Analisis Sistem

Menurut (Whitten, 2007, p160) analisis sistem adalah sebuah teknik pemecahan masalah yang memecahkan sebuah sistem menjadi komponen-komponen untuk bertujuan pembelajaran bagaimana.

2.1.2 Pengertian Perancangan Sistem

Menurut (Whitten, 2007, p160) perancang sistem adalah teknik komplementer pemecahan masalah (yang bekerjasama dengan sistem analisis) yang menyusun kembali komponen-komponen sebuah sistem kembali ke sistem yang utuh dengan harapan menghasilkan sistem yang baik. Teknik ini dapat melibatkan penjumlahan, penghapusan, dan perubahan komponen-komponen terhadap sistem yang sebelumnya.

2.1.3 Internet dan Sejarah singkat

Menurut Williams dan Sawyer (2010, 18), *Internet* merupakan "induk dari semua jaringan" *internet* adalah jantung era informasi. Disebut "induk dari semua jaringan" karena *internet* ("net" atau "jaringan") adalah jaringan komputer di seluruh dunia yang menghubungkan ratusan bahkan ribuan jaringan yang lebih kecil, misalnya jaringan pendidikan, komersial, nirlaba dan militer, bahkan juga individual.

Internet sebagai jaringan yang terhubung dalam *Internet Protocol* (IP) akan membentuk sejumlah jaringan fisik dan jaringan logik, yang selanjutnya disebut *inter-network*. *Internet* sebagai komunitas jaringan komputer juga akan memberikan layanan *WWW* (*World Wide Web)* yang tidak terbatas sehingga tercipta suatu komunitas global dari orang-orang yang saling berbagi informasi yang dapat diakses oleh siapa saja saat dibutuhkan.

Proses pembentukan internet pada awalnya berasal dari sebuah proyek kalangan militer AS yaitu *ARPA* (*Advance Research Projects Agency*). Pada akhir 1969 bersama-sama dengan universitas dan organisasi riset lainnya membangun *ARPAnet* (*Advance Research Project Agency Networks*). *ARPAnet* terus mengalami perkembangan teknik dan *protokol* selama beberapa perioe hingga pada tahun 1981 dimana *ARPAnet*

mengadaptasi *protocol* TCP/IP. *ARPAnet* inilah yang kemudian berkembang menjadi internet yang di kenal sekarang.

Keuntungan memakai internet :

1. Luasnya beragam informasi yang bisa di akses tanpa batas.
2. Kecepatan dalam mendapatkan informasi.
3. Informasi-informasi tersebut dapat di akses tanpa perlu mendatangi tampat sehingga menjadikan biaya yang relatif lebih murah.

Kerugian memakai Internet :

1. Kemudahan mengakses informasi dapat menjadi kelemahan bila *user* tidak memiliki kemampuan dan pengetahuan yang cukup untuk mengevaluasi dan menyaring informasi-informasi yang tersedia pada *internet*
2. Kemudahan publikasi menimbulkan pertanyaan tentang kualitas dan stabilitas dari informasi-informasi yang tersedia tersebut.

2.1.4 Web Browser

Menurut Schneiderman (2010, 2) *World Wide Web (WWW)* adalah sebuah subset dari computer diinternet dan tujuannya untuk mengorganisasikan sumber-sumber secara luas untuk membuat mereka lebih mudah diakses oleh user.

2.1.5 Web Server

Menurut Shelly et al (2010, 4) Halaman *Web* disimpan di server *Web* atau *host*, yang merupakan komputer untuk menyimpan dan mengirim (melayani) halaman *Web* dan files lainnya yang telah direquest. Setiap komputer yang memiliki *web server* dan terhubung ke *Internet* dapat bertindak sebagai *server Web*. Setiap situs *web* disimpan dan dijalankan satu *web server* atau lebih. Sebuah situs *Web* yang besar mungkin tersebar ke beberapa *server* di lokasi geografis yang berbeda.

2.1.6 *Web Database*

Menurut Shelly et al (2010, 5) *Web Database* merupakan sistem penyimpanan data yang dapat diakses oleh bahasa pemrograman *web* tertentu. *Web Database* dapat diakses oleh aplikasi-aplikasi web yang telah berkembang dengan *HTML*,

CGI, IIS (*Internet Information Server*) atau *script* yang bersifat *server side* seperti *ASP, PHP* atau *JSP*.

2.1.7 E-Commerce

menurut Kenneth C. Laudon and J.P Laudon (2010) *"E-commerce refers to the use of the Internet and the Web to transact business. More formally, e-commerce is about digitally enabled commercial transactions between and among organizations and individuals."* Dengan kata lain, *E-Commerce* merupakan penggunaan *Internet* dan *Web* dalam melakukan transaksi bisnis dan *E-commerce* secara digital untuk mempermudah transaksi komersial antar organisasi dan antara organisasi dengan individu. *E-Commerce* menggambarkan proses pembelian, penjualan, pemindahan, pelayanan atau pertukaran produk, jasa atau informasi melalui jaringan komputer termasuk juga *Internet*

2.1.7.1 Komponen-komponen Internet

2.1.7.2 *Electronic Mail (E-mail)*

E-mail merupakan cara pengiriman surat atau pesan secara elektronis. *E-mail* telah mengubah cara-cara pengiriman surat atau pesan secara konvensional menjadi lebih cepat dan tepat karena dengan *e-mail* hanya membutuhkan waktu dalam hitungan detik

untuk mengirim pesan-pesan kepada suatu alamat yang terdaftar pada internet.

Meskipun *e-mail* telah mempengaruhi cara-cara orang mengirimkan surat, dimana awalnya menggunakan kertas kini dilakukan secara elektronis, namun *e-mail* ini belum bisa mengganti cara-cara konvensional dalam surat-menyurat, terutama untuk dokumen penting yang harus disampaikan kepada penerima dan perlu konfirmasi dari si penerima surat, misal melalui pembubuhan tanda tangan. Meskipun begitu, dalam banyak hal, terutama untuk pesan-pesan yang tidak begitu formal, *e-mail* telah mengubah cara seseorang mengirim surat. *E-mail* sangat efektif dan efisian bila digunakan untuk *korespondensi* antar teman, kolega dan lain-lain yang tidak banyak menggunakan formalitas.

2.1.7.3 *WWW (World Wide Web)*

Worrld wide web adalah adalah nama yang diberikan untuk semua bagian internet yang dapat diakses dengan software web browser. Sampai saat ini ada beberapa *software web browser* yang sering digunakan antara lain *microsoft internet explorer, mozilla, chrome* dan *opera*.

World wide web terdiri dari jutaan situs website dan setiap website terdiri banyak halaman *web*. Halaman-halaman web ini

tersebar diseluruh dunia kepada komputer *server* yang terhubung dengan *internet.*

Halaman web adalah halaman yang tampak dari sebuah *software web browser*. Halaman ini seperti *file* dokumen *word-processing*, kecuali didalamnya dapat terkandung teks, gambar bahkan animasi.

Situs Web terdiri dari beberapa halaman web yang saling terkait (*Link)* antara yang satu dengan yang lain dengan satu cara tertentu. (anonim3, 2011).

Pengertian *website* menurut William/sawyer (2007, p65) adalah sebuah lokasi pada computer tertentu di *web* yang memiliki alamat unik atau *URL*.

2.1.7.4 *HTTP (Hypertext Transfer Protocol)*

Menurut Sawyer (2011, 66) *Hypertext Transfer Protocol* (*HTTP*) adalah aturan-aturan komunikasi yang memungkinkan browser untuk terhubung dengan *web server*.

HTTP termasuk *application layer* yang berjalan melalui *TCP/IP* dengan membuat *TCP* ke *port* tertentu (biasanya *port* 80). Sebagaimana internet protokol lainnya, *HTTP* berfungsi sebagai model *client-server*, dimana *browser* bertindak sebagai *client* dan membuat kode permintaan data (*request*) Kemudian *web server*

menjawab *request* tersebut dan menyediakan data yang diinginkan.

2.1.7.5 *Uniform Resource Language (URL)*

Menurut Sawyer (2011, 65) *URL* adalah string karakter yang menunjuk ke sebuah bagian tertentu informasi di mana saja di *web*. Dengan kata lain, *URL* adalah alamat unik dari *website*.

2.1.7.6 *Hyper Text Markup Language(HTML)*

Menurut Shelly et al (2010, 8), Halaman *web* dibuat menggunakan *Hypertext Markup Language (HTML*), yang merupakan bahasa *authoring* yang digunakan untuk membuat dokumen *World Wide Web (www*). *HTML* menggunakan set instruksi khusus yang disebut tag atau markup untuk menentukan struktur dan tata letak dokumen *Web* serta menentukan bagaimana halaman ditampilkan di *browser*.

2.1.7.7 Transmission Control Protocol/Internet Protocol (TCP/IP)

Menurut Sardiman (2008, 1) *TCP/IP* adalah suatu set protokol standar yang digunakan untuk menghubungkan jaringan komputer, mengamati lalu lintas dalam jaringan, mengatur format data yang diizinkan, penanganan kesalahan dan standar komunikasi lainnya. Selama 20 tahun lebih sudah protokol

TCP/IP digunakan dan waktu telah membuktikannya bahwa protokol ini telah teruji dan sangan stabil. *TCP/IP* memiliki beberapa fitur dan keuntungan, seperti :

1. Dukungan dari berbagai vendor.

Seperti telah di sebutkan diatas, *TCP/IP* sangat *compatible* dengan *hardware* dan *software* dari berbagai vendor, sehingga tidak terikat pada satu vendor saja.

2. *Interoperability*

Alasan mengapa *TCP/IP* sangan populer dan banyak di gunakan adalah *TCP/IP* dapat di install dan dipakai di berbagai *platform* sistem operasi.

3. *Flexibility*

TCP/IP merupakan protokol yang sangat fleksibel. Contoh dari *fleksibilitas TCP/IP* adalah seorang *administrator* dapat mementukan dan memindahkan alamat *IP*, menetapkan alamat *IP* ke dalam *host* secara manual ataupun otomatis dan dapat merubah *TCP/IP host* menjadi sebuah nama yang mudah diingat seperti *www.google.com*.

4. *Routability*

TCP/IP dapat beradaptasi dengan baik terhadap proses *routing* data dari satu segmen ke segmen jaringan yang lain atau dari *host* ke *host* yang lain.

2.1.7.8 Model *Client-Server*

Internet beroperasi dengan model *client-server*. Dalam model *client-server* computer *client* mengirimkan *request* informasi yang dibutuhkan kepada *computer server*. *Server* merespons dengan mengolah permintaan dari *client* dan mengirimkan hasil kembali kepada *client* tersebut. Secara umum, model *client-server* ini memiliki *client* yang lebih besar jumlahnya dibandikan dengan *server*. Di dalam internet, *server* dikenal dengan sebutan *web server*. Sedangkan *Client* dikenal sebagai *web browser*.

2.1.7.9 Hypertext Preprocessor (PHP)

Menurut Zaki (2008, 2), *PHP* adalah sebuah bahasa pemrograman *scripting* untuk membuat halaman *web* yang dinamis. Walaupun dikenal sebagai bahasa untuk membuat halaman *web*, tapi *PHP* sebenarnya juga dapat digunakan untuk membuat aplikasi *command line* dan *GUI*. Namun yang dibahas di buku ini adalah penerapan *PHP* untuk membuat halaman *web*

dinamis. Cara kerja *PHP* adalah dengan mnyelipkan diantara kode *HTML (Hypertext Markup Language*).

Website yang dibuat menggunakan *PHP* memerlukan *software* bernama *webserver*, tempat pemrosesan kode *PHP* dilakukan. *Server web* yang memiliki *software PHP* parser akan memproses input berupa kode *PHP* dan menghasilkan *output* berupa halaman *web*. *PHP* bersifat terbuka dan *multiplatform*, karenanya dapat dijalankan dibanyak merk *webserver* (seperti *Apache* dan *IIS*). Saat ini ,pengguna *PHP* sangat banyak, ada lebih dari dua puluh juta *website* yang menggunakan *PHP* yang dijalankan lebih dari satu juta *server.*

2.1.7.10 Sintaks Dasar PHP

Bahasa *PHP* memiliki script sebagai berikut :

- Menggunakan tanda <?..?>
- Menggunakan tanda <?php…?>
- Menggunakan perintah seperti pada Javascript

  Contoh : <script language “php”>…</script>
- Menggunakan tanda <%...%>

Dari seluruh cara tersebut dapat dilihat bahwa tanda penutup harus sama dengan tanda pembukanya. Misalnya jika tanda pembukanya (<?) maka tanda penutupnya adalah (?>). Fungsi dari tanda penutup

ini adalah untuk memberikan tanda bahwa sebuah *statement* atau *script PHP* telah selesai dilengkapi.

2.1.7.11. Web Server Apache

Apache HTTP server adalah *software open source* untuk *HTTP web server* yang dijalankan di *platform* system *Unix-like* seperti *BSD, Linux* dan *UNIX, Microsoft Windows* dan lainnya. *Apache* dikembangkan dan dipelihara oleh komunitas terbuka yang bergabung dengan *Apache Software Foundation*. Mulanya, *Apache* sekedar *open source alternative* untuk *Netscape web server* (kini dikenal sebagai *SUN ONE*). Kemudian berkembang dengan pesat dan melampaui semua jenis *Unix based HTTP server* lainnya dalam hal funsionalitas dan kecepatan. Apache adalah *HTTP server* paling popular di *internet* sejak April 1996 dan menguasai 57% pangsa pada bulan Mei 1999. Pada bulan Agustus 2004 tercata peningkatan menjadi 67%.

Apache piawai dengan fitur *'highly configurable error messages, DBMS-based authentication databases, and content negotiation'*. Didukung beberapa *GUI* memudahkan konfigurasi dan lebih intuitif. *Apache* mendukung fitur yang diimplementasikan sebagai modul terkompilasi mulai dari *server-side programming language support* sampai dengan *authentication schemes*. Sejumlah bahasa pemrograman yang lazim digunakan seperti *Perl, Phyton,* dan *PHP* didukung *Apache* sepenuhnya. Apache umumnya digunakan dalam satu paket popular *LAMP (Linux Apache MySQL* dan *PHP/Perl/Phyton*). Selain itu *apache*

juga dikemas bersama paket-paket seperti *Oracle database* atau *IBM WebSphere application server.* Sementara secara tak langsung juga didukung *Borland* bersama *Kylix* dan *Delphi development tools* mereka.

2.1.7.12. Database

Menurut Connolly et al (2010, 65), *database* adalah suatu kumpulan data yang saling berhubungan secara logis dan penjelasan tentang data yang terhubung tersebut dirancang sedemikian rupa sehingga dapat memberikan informasi yang diperlukan oleh organisasi. Artinya, *database* adalah tempat penyimpanan data yang terbesar dimana bisa digunakan secara simultan atau secara bersamaan oleh banyak departemen dan pengguna lainnya. Di dalam *database,* semua data *item* diintegrasikan dengan menghindarkan duplikasi data. *Database* tidak hanya mengandung data operasional organisasi, tetapi juga deskripsi dari data tersebut. Untuk alasan ini sebuah *database* juga mendefinisikan integrasi *record* dari *database* itu sendiri (*self describing of integrated record*). Deskripsi atau data ini dikenal sebagai *system catalog* (data *dictionary* atau meta-data). Deskripsi ini menetapkan kebebasan dari program aplikasi (*program data independence*). Pendekatan dengan sistem *database* dimana definisi dari data adalah dipisahkan dari program aplikasi.

Database adalah sebuah objek untuk menyimpan informasi terstruktur yang kompleks, serta diorganisir dan disimpan dalam suatu cara yang diizinkan penggunanya sehingga dapat mengambil informasi dengan cepat dan efisien. Informasi dipecah ke dalam bagian-bagian kecil, agar bisa dikelola dengan mudah. *Database* juga merupakan data yang digunakan pada suatu perusahaan.

Menurut Whitten (2007, 548), basis data adalah kumpulan file yang saling terkait. Basis data tidak hanya merupakan kumpulan file. *Record* pada setiap file harus memperbolehkan hubungan-hubungan untuk menyimpan file lain. Keuntungan basis data yaitu :

- Kemampuannya untuk menggunakan data yang sama di banyak aplikasi dan sistem.
- Penyimpanan data dalam format yang fleksibel. Hal ini didefinisikan secara terpisah dari sistem informasi dan program-program aplikasi yang akan menggunakan basis data.
- Teknologi basis data menyediakan skalabilitas superior, dalam arti basis data dan sistem yang menggunakannya dapat ditingkatkan atau dikembangkan untuk memenuhi kebutuhan-kebutuhan perubahan pada sebuah organisasi.
- Kemajuan independensi data yang sangat mengurangi redudansi data, telah mengingkatkan fleksibilitas.

2.1.7.13 *Relational Database*

RDBMS adalah sebuah aplikasi yang dirancang untuk mengatur sebuah database sebagai sekumpulan data yang disimpan secara terstruktur (Conolly et al 2007, 69).

2.1.7.14. MYSQL

Menurut Conolly (2008, 46-47) SQL adalah *Structured Query Language. SQL* ini adalah suatu bahasa permintaan yang terstruktur.*SQL* merupakan bahasa yang berada di database, atau biasa disebut dengan *MySQL*. *MySQL* merupakan database server dari *SQL*. *MySQL* adalah *database management system (DBMS*) dan *SQL* merupakan perintah dalam *DBMS*.

Menurut Abdul Kadir (2009, 15) *MySQL* merupakan *software* yang tergolong *database server* dan bersifat *open source. Open source* menyatakan bahwa software ini dilengkapi oleh *source code* (kode yang dipakai untuk membuat *MySQL*), selain itu bentuk *excutable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi,dan bisa diperoleh dengan cara mengunuduh di *internet* secara gratis. Hal menarik lainnya adalah *MySQL* juga bersifat *multiplatform* yang membuat *MySQL* dapat dijalankan pada berbagai sistem operasi.

Pengaksesan data dalam *database* dapat dilakukan dengan mudah melalui *SQL* (*Structured Query Language*). Data dalam *database* bisa di akses

melaui aplikasi *non-Web* (misalnya dengan *Visual Basic*) maupun aplikasi *web* (misalnya dengan *PHP*).

2.1.7.15.U*nified Modeling Language* (*UML*)

Menurut Whitten (2007, 430) *unified Modeling Language (UML)* adalah sekumpulan pemindahan konvensi yang digunakan untuk menentukan atau menggambarkan sebuahsistem perangkat lunak dalam kaitannya dengan objek.

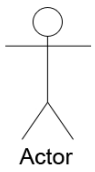
2.1.7.16.*Use Case Diagram*

Menurut Whitten et al (2007, 246), *Use Case Diagram* adalah diagram yang menggambarkan interaksi antara sistem dengan hal-hal eksternal dari sistem dan *user*. Dengan kata lain, secara grafis menjelaskan siapa yang akan mempergunakan sistem dan dengan cara apa *user* diharapkan untuk berinteraksi dengan sistem.

Use Case Diagram berguna untuk menentukan *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem

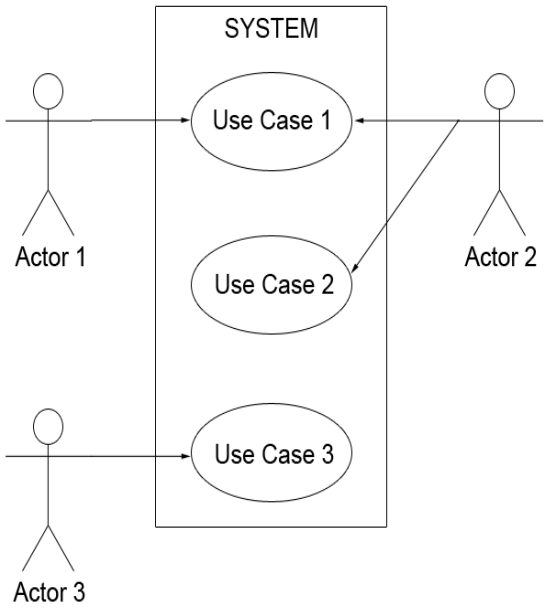
Berikut ini adalah simbol-simbol dalam *Use Case Diagram*:

Tabel 2.1 Tabel Simbol dalam Use Case Diagram

Simbol	**Nama**
	Aktor
	Use Case
	Association

(Sumber: Whitten et al 2007, pp247-248)

Gambar 2.1 Contoh Penggunaan Use Case Diagram



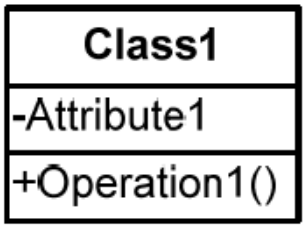
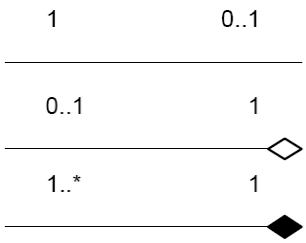
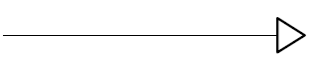
(Sumber: Whitten et al 2007, p246)

2.1.7.17 Class Diagram

Menurut Whitten et al (2007, 382), Class diagram menggambarkan struktur dari sistem. Serta menampilkan *Class Object* yang berada di dalam sistem serta hubungan antara objek tersebut dan objek lainnya.

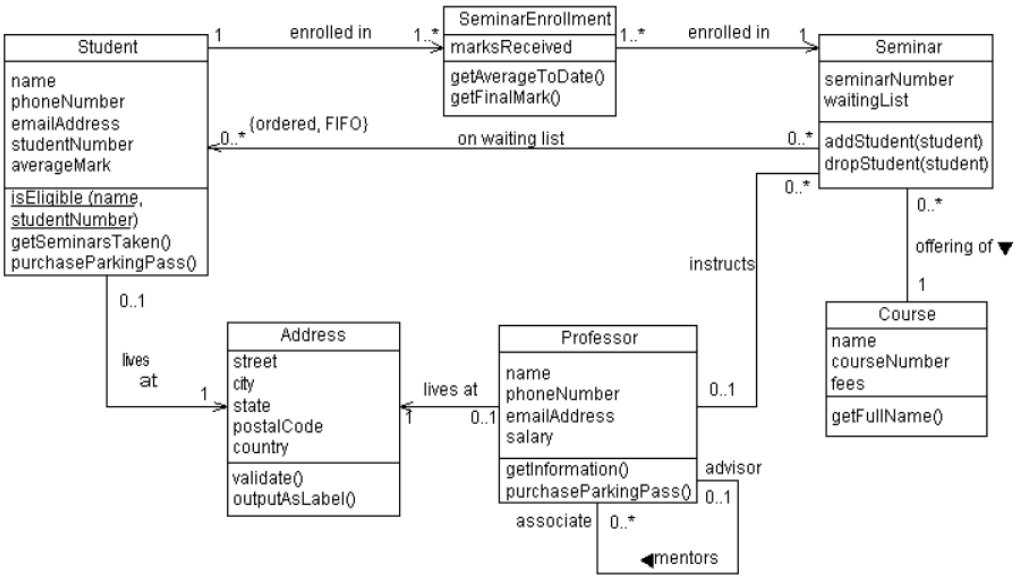
Berikut ini adalah contoh relasi *Class Diagram:*

Tabel 2.2 Relasi pada Class Diagram

Symbol	Penjelesan
	*Class* Sebuah deskripsi dari seperangkat objek yang berbagi atribut, operasi, dan relasi yang sama. *Class* terbagi atas 3 bagian ,yaitu nama *class* pada bagian atas, atribut *class* pada bagian tengah, dan operasi pada bagian bawah.
	Association Merupakan hubungan *structural* antar *class* yang saling berelasi.
	Aggregation Bentuk special dari hubungan asosiasi yang menspesifikasikan semua hubungan antara kumpulan dan sebuah bagian. Agresasi digambarkan dengan wajik tidak berisi.
	*Multiplicity* Menggambarkan jumlah objek yang berpartisipasi dalam hubungan antar class.
	*Generalization* Merupakan sebuah relasi specialisasi / generalisasi dimana suatu kelas dapat lebih general atau lebih spesifik dari kelas lainnya.

Berikut ini adalah contoh penggunaan *Class Diagram*:

Gambar 2.2 Contoh Class Diagram



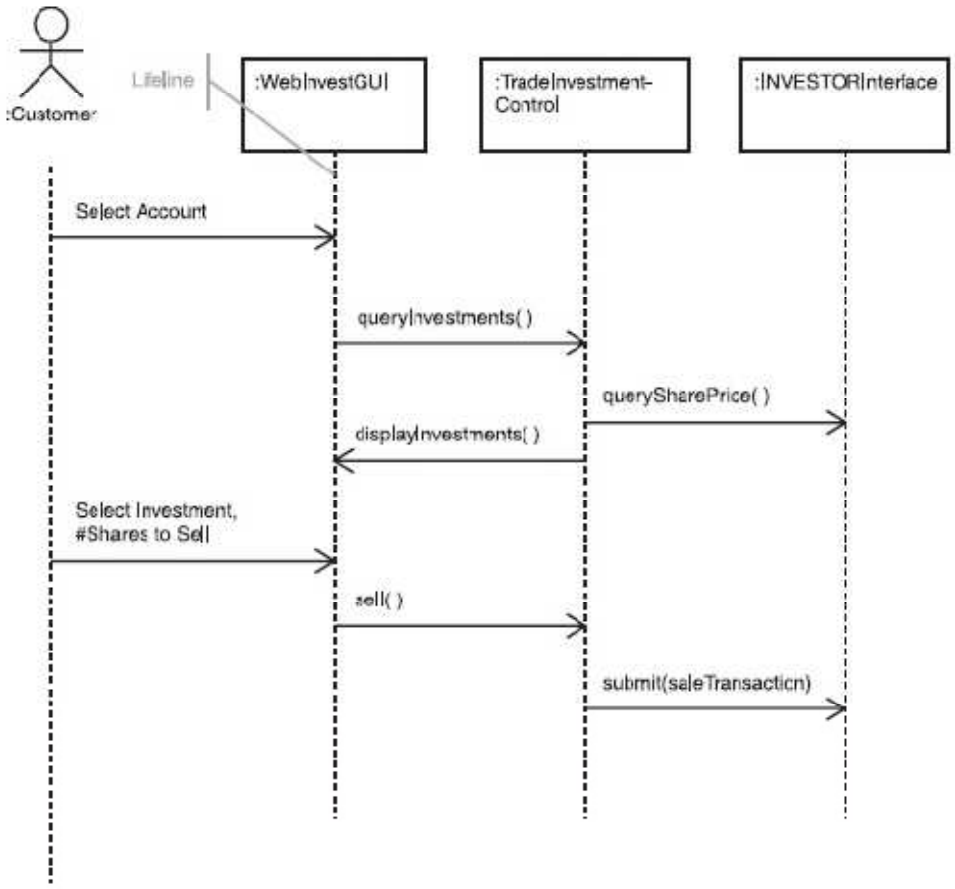
(Sumber: Whitten et al, 2007)

2.1.7.18. Sequence Diagram

Menurut Whitten et al (2007, p382), *Sequence diagram* adalah diagram yang menggambarkan bagaimana objek berinteraksi dengan objek lainnya melalui pesan dalam suatu eksekusi dari sebuah *Use Case* atau sebuah operasi. *Sequence diagram* juga mengilustrasikan bagaimana pesan dikirim dan diterima oleh objek dan terjadi di dalam suatu *sequence.*

Berikut ini adalah contoh penggunaan *Sequence Diagram*:

Gambar 2.3 Contoh Penggunaan Sequence Diagram



(Sumber: Whitten et al, 2007, p382)

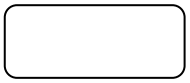
2.1.7.19. Activity Diagram

Menurut Whitten et al (2007, p390), *Activity Diagram* adalah *diagram* yang dipergunakan untuk menggambarkan alur dari proses

bisnis secara grafis, langkah-langkah dari *use case* dan logika dari karakteristik objek.

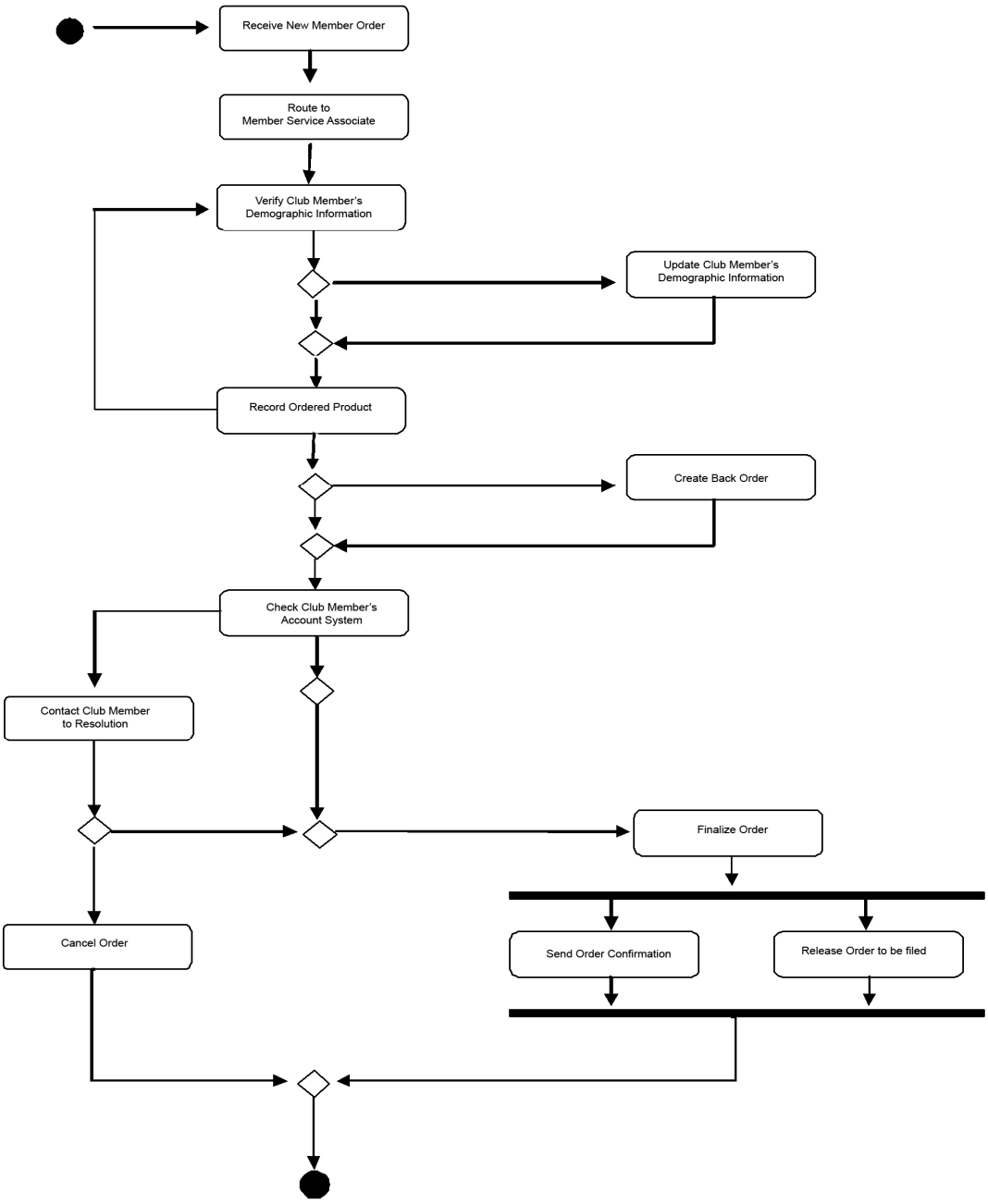
Berikut ini adalah contoh notasi-notasi dalam *activity Diagram*:

Tabel 2.3 Relasi pada Activity Diagram

Notasi	**Fungsi**
	Melambangkan awal dari sebuah proses.
	Melambangkan *activity.*
	Melambangkan *trigger*.
	Melambangkan garis sinkronisasi.
[ ]	Kata-kata di dalam [ ] melambangkan *trigger* yang menjadi hasil dari *decision activity*.
	Melambangkan *decision activity.*
	Melambangkan akhir dari proses.

Berikut ini adalah contoh penggunaan *activity Diagram*:

Gambar 2.4 Contoh Activity Diagram



(Sumber: Whitten et al, p392

2.1.7.20 Interaksi Manusia dan Komputer (IMK)

Menurut Santoso (2009,p2). Ketika seseorang bekerja dengan sebuah komputer secara disadari atau tidak, dia melakukan interaksi dengan komputer dengan menggunakan cara-cara tertentu. Interaksi terjadi ketika pengguna memasukan data yang kemudian akan ditanggapi oleh komputer dengan menampilkan suatu keluaran ke *layer* tampilan atau pencetak.

Menurut (Shneiderman, 2010, p22) Pembuat *interface* bisa dikatakan sebagai pahlawan transformasi di karenakan pekerjaannya yang membuat *personalcomputer*, yang memungkinkan *user* untuk berkomunikaasi dengan komputer dan berkolaborasi dengan cara yang luar biasa.

Menurut Shneiderman (2010, p32) dalam membuat *interface* ada ukuran atau *parameter* yang harus diperhatikan para *designer* interface agar *interface* yang di buatnya dapat dikatakan *user-friendly*. Ukuran *parameter* tersebut adalah waktu untuk belajar yaitu berapa lama waktu yang dibutuhkan *user* untuk mempelajari aksi yang harus dilakukan pada suatu tugas.

2.1.7.21 Delapan Aturan Emas

Delapan aturan emas adalah prinsip-prinsip mendasar untuk desain antarmuka. Peraturan ini diperoleh berdasarkan pengalaman pembuatan desain. Berikut ini adalah Delapan Aturan Emas tersebut menurut Shneiderman (2010, p88) :

1. **Berusaha untuk konsisten**, Konsisten pada aksi-aksi dalam situasi tertentu, konsistensi warna, tampilan, huruf, dan sebagainya
2. **Menyediakan *universal usability***, Menyadari kebutuhan dari berbagai *user* dan merancang *design* yang flexibel serta memfasilitasi perubahan dari isi.
3. **Umpan balik yang infomatif**, Untuk setiap aksi yang dilakukan pengguna terhadap sistem, sistem harus memiliki umpan balik yang sopan dan jelas.
4. **Membuat dialog untuk menghasilkan keadaan akhir**, Urutan-urutan aksi diatur ke dalam grup-grup dengan bagian awal, tengah dan akhir. *Feedback* pada saat akhir dari group aksi tersebut harus memuaskan pengguna.
5. **Memberikan penanganan kesalahan yang sederhana**, Jika pengguna melakukan kesalahan, sistem harus dapat mendeteksinya dan memberikan instruksi sederhana dan membangun untuk perbaikan.
6. **Mengizinkan pembalikan aksi**, Sebanyak mungkin, semua aksi dapat dibalik. Fitur ini mengurangi kekhawatiran karena pengguna tahu kesalahan yang dapat diabaikan.
7. **Mendukung pusat kendali internal**, Pengguna yang berpengalaman menginginkan suatu perasaan bahwa mereka menguasai sistem dan sistem harus merespon keinginan mereka.

8. **Mengurangi beban ingatan jangka pendek**, Untuk mengatasi hal ini dapat dilakukan dengan mengurangi *frekuensi window* dan dengan waktu pelatihan yang cukup.

2.1.7.22 Rekayasa Piranti Lunak (RPL)

Menurut Pressman (2010, p53), rekayasa perangkat lunak adalah pengembagan dan penggunaan prinsip pengembangan suara untuk memperoleh perangkat lunak secara ekonomis yang *reliable* dan bekerja secara efisien pada mesin nyata. Definisi tersebut telah dikembangkan menjadi lebih komprehensif oleh *IEEE (Institute of Electrical and Electronics Enginners*) yaitu Aplikasi dari sebuah pendekatan kuantifiabel, disiplin, dan sistematis kepada pengembangan, operasi, dan pemeliharaan perangkat lunak.

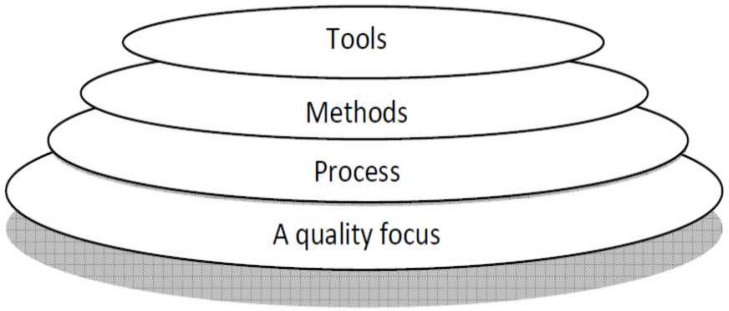
2.1.7.23 Lapisan Rekayasa Piranti Lunak

Secara umum Rekayasa Perangkat Lunak dapat dibagi menjadi empat *layer,* Pressman (2010, p54), antara lain :

1. *Process Model*, adalah fondasi dari RPL yang mendefinisikan sebuah framework untuk sekumpulan *key process* area yang harus dibangun demi keefektifan penyampaian teknologi pengembangan RPL.

2. *Methods*, menyediakan secara teknis bagaimana untuk membangun suatu perangkat lunak.
3. *Tools*, menyediakan dukungan otomatis dan semi otomatis untuk process model dan methods.
4. *Quality focus*, merupakan batu landasan yang menopang *tools, methods,* dan *process* dalam RPL.

Gambar 2.5 Software Engineering Layers



(Sumber: Pressman, 2010)

2.1.7.24 Model Proses Piranti Lunak

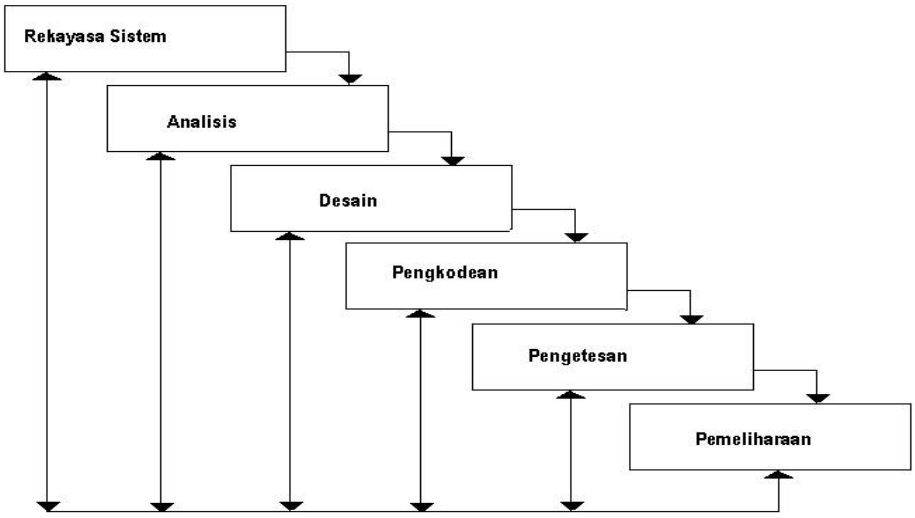
Menurut Pressman (2010, p78), ada beberapa model proses *software* yang umum digunakan, salah satunya adalah Model *Sekuensial Linear*. Model Sekuensial Linear ini juga dikenal dengan nama "*ClassicLife Cycle*" atau "*Waterfall Model*". Model ini melingkupi aktivitas-aktivitas sebagai berikut :

1. **Rekayasa dan Pemodelan Sistem Informasi,** Rekayasa dan pemodelan sistem informasi diperlukan karena perangkat lunak selalu merupakan bagian dari sebuah sistem yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa *subset* dari kebutuhan ke perangkat lunak tersebut. Perangkat lunak harus berhubungan dengan element-element yang lain seperti perangkat lunak, dan *database*.
2. **Analisis kebutuhan perangkat lunak,** Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak. Untuk memahami sifat program yang dibangun, seorang perancang perangkat lunak harus memahami kebutuhan informasi, fungsi-fungsi, unjuk kerja, dan interface yang diperlukan.
3. **Perancangan,** *Design* perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat *attribute* sebuah program yang berbeda : struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (*algoritma).*
4. **Pengkodean,** Design harus diterjemahkan menjadi bentuk yang bisa dibaca atau dimengerti oleh komputer, biasanya dalam bentuk bahasa pemrograman.
5. **Pengujian,** Sekali kode dibuat, pengujian program juga dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan

kesalahan-kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

6. **Pemeliharaan,** Perangkat lunak akan mengalami perubahan setelah disampaikan kepada pelanggan. Perubahan akan terjadi karena kesalahan yang ditentukan, karena perangkat lunak harus disesuaikan untuk mengakomodasikan perubahan-perubahan didalam lingkungan eksternalnya, atau pelanggan membutuhkan perkembangan fungsional atau untuk kerja. Pemeliharaan perangkat lunak mengaplikasi lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.

Gambar 2.6 System Developing Lifecycle



(Sumber: Pressman, 2010)

2.1.5.25 Entity Relationship Diagram (ERD)

Menurut **Whitten** (2007, p381), ERD adalah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang dideskripsikan oleh data tersebut.

Menurut **Connoly dan begg** (2010, p371-395), *Entity Relation Diagram* digunakan untuk menggambarkan struktur *logical database* dalam bentuk diagram ERD, serta menyediakan cara yang sederhana dan mudah untuk memahami bagian berbagai komponen dalam desain database.

ERD memiliki 3 komponen yaitu :

1. *Entity*

   Entity adalah benda yang memiliki identifikasi yang berebeda. Entity dapat digambarkan sebagai persegi yang berisi nama dari entity tersebut.

2. *Relationship*

   Relationship adalah hubungan antara entity. Entity merupakan pengikut dari relationship. Relationship dapat digambarkan dalam bentuk belah ketupat yang mana berisi nama dari relasi tersebut.

   Relationship dapat berupa hubungan *one-to-one*, *one-to-many, atau many-to-many.*

a. Hubungan *one-to-one* (1 atau 1..1)

   Hubungan dimana setiap *entity* yang ada hanya dapat mempunyai maksimal 1 (satu) hubungan dengan *entity* yang lain.

b. Hubungan *one-to-many* (1..*)

   Hubungan dimana setiap *entity* yang dapat mempunyai satu atau lebih dari satu hubungan dengan *entity* yang lain.

c. Hubungan *many-to-many* (*..*)

   Hubungan dimana setiap *entity* dapat mempunyai lebih dari satu relasi dengan *entity* lainnya.

2.1.7.26 *XAMPP*

XAMPP adalah aplikasi *web server* gratis yang *multiplatform* yang terdiri atas apache *HTTP server*, *MySQL database*, dan penerjemah untuk *script* yang ditulis dalam bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* adalah singkatan dari X (untuk sistem operasi yang berbeda), *Apache, MySQL, PHP* dan *Perl.*

Program ini dipublikasikan dibawah lisensi *GNU general public License* dan bersifat gratis, dan sebagai *web server* yang mudah digunakan untuk aktifitas halaman *web* yang dinamis. Saat ini, *XAMPP* dapat digunakan pada sistem operasi *Microsoft Windows*, *Linux*, *Sun solaris* dan *Mac OS X*, dan biasa digunakan untuk proyek pembuatan *website.*

Umumnya, pembuat *XAMPP* hanya berencana sebagai alat untuk pengembangan yang mengijinkan pembuat website dan pembuat program menguji pekerjaan mereka di komputer mereka sendiri tanpa harus terhubung ke *internet*. Dalam pelaksanaannya, *XAMPP* seringkali digunakan sebagai *web* server di *WWW* (*world wide web*), *XAMPP* juga mendukung pembuatan dan perubahan *database di MySQL dan SQLite*.

2.1.8 Definifi Pelayanan dan Jasa

2.1.8.1 Definisi Pelayanan

Menurut Kotler dalam Laksana (2008) pelayanan adalah setiap tindakan atau kegiatan yang dapat ditawarkan oleh suatu pihak kepada

pihak lain, yang pada dasarnya tidak berwujud dan tidak mengakibatkan kepemilikan apapun**.**

2.1.8.2 Definisi Jasa

“ Jasa merupakan aktivitas, manfaat atau kepuasan yang ditawarkan untuk dijual. “(Fandly Tjiptono, 2007:23).

2.1.8.3 Definisi Perancangan dan Implementasi Sistem

Menurut Al Fatta (2007, p24) perancangan sistem diartikan sebagai menjelaskan dengan detail bagaimana bagian-bagian dari sistem informasi diimplementasikan. Implementasi sistem adalah proses pemakaian informasi yang telah dirancang dalam perancangan sistem.

2.1.8.4 Definisi *Tracking*

Tracking adalah suatu proses pencatatan *interval* perjalanan barang dari tempat asal ke tempat tujuan oleh perusahaan pengangkutan. (Rumapea, 2010, p350).

2.1.8.5 Definisi Sistem Tracking

Dari definisi sistem dan definisi *tracking* di atas dapat disimpulkan bahwa sistem *tracking* adalah sekelompok elemen atau unsur yang saling berhubungan dalam mencatat interval perjalanan barang dari tempat asal ke tempat tujuan, sehingga dapat mengubah suatu

masukan yang berupa data – data interval pejalanan suatu barang menjadi suatu keluaran yang berupa informasi interval perjalanan suatu barang.

2.1.8.6 *Order*

(Rumapea, 2010) *order* atau pemesanan penjemputan adalah layanan yang memudahkan *customer* yang diberikan oleh perusahaan apabila *customer* tidak memiliki waktu untuk mengantarkan barang yang akan dikirimkan, dengan melakukan pemesanan agar barang yang akan dikirim dapat dijemput oleh kurir perusahaan. Pemesanan dilakukan dengan mengisi formulir pemesanan secara *online*.

2.1.8.7 Definisi Pengiriman Barang

Menurut Himayati (2008, 149) Pengiriman barang adalah fasilitas transaksi yang digunakan yang digunakan untuk meng input atau menjurnal barang atau jasa yang dikirim atau di juak kepada pelanggan berdasarkan *sales order* (*SO)* yang sudah dikirim sebelumnya.

2.1.8.8 Teori Tentang Pengiriman Barang

Di bawah ini akan menjelaskan secara singkat mengenai beberapa pengertian penting yang berkaitan dengan pengiriman barang, yaitu :

1. Perdagangan internasional dan ekspor impor

Perdagangan internasional adalah suatu kegiatan jual beli guna memperoleh keuntungan (perdagangan) yang dilkukan dengan

melibatkan unsur-unsur dua negara atau lebih (Internasional). Apabila diperluas makna memperoleh keuntungannya tidak hanya memperoleh keuntungan secara finansial tetapi bisa juga memberi keuntungan non finansial seperti untuk kepentingan promosi, persaingan usaha dan keuntungan strategis lainnya. Faktor – faktor / kendala dalam transaksi perdagangan internasional :

1. Letak geografis berjauhan.
2. Batas kenegaraan.
3. Mata uang.
4. Peraturan pemerintah yang terdapat di setiap negara di bidang perdagangan internasional.

Menurut Susilo (2008, 62) ekspor impor adalah suatu kegiatan memasukan /mengeluarkan barang ke atau dari wilayah pabean berdasarkan peraturan yang ditetapkan, intinya ada pada pemasukan atau pengeluaran barang, baik didasari atas transaksi perdagangan atau bukan.

2.1.8.9 *Freight forwarding*

Menurut Kosasih (2009, 10) *Freight forwarding* merupakan sebuah jasa yang sering kali di gunakan oleh berbagai macam perusahaan yang berhubungan dengan impor dan ekspor yang bersifat internasional maupun multi-nasional. *Freight forwarder* hanya berperan sebagai perantara diantara pelanggan dan berbagai jenis transportasi. Mengirim

produk dari sebuah tujuan internasional ke tempat yang lain bisa melibatkan banyak pengangkut sekaligus, serta membutuhkan persyaratan dan legalitas. *Freight forwarding* menangani semua keperluan logistik yang di butuhkan demi pelanggan, yang jika dilakukan oleh pelanggan sendiri akan cukup melelahkan.

Jasa *Freight forwarder* menjamin kalau produk yang di kirim akan sampai di tujuan yang benar sesuai dengan tanggal yang sudah dijanjikan dan dalam kondisi yang baik. *Freight forwarder* menyediakan hubungan dengan berbagai jenis perusahaan transportasi, baik udara, laut ataupun darat, dari kereta hingga ke pelayaran.

Jasa *freight forwarder* yang baik dapat mengurangi rasa was-was dari pelanggan dengan memberikan layanan pengiriman yang berkualitas dengan harga yang kompetitif. Jasa *Freight forwarder* merupakan aset bagi perusahaan apapun yang berhubungan dengan pengiriman barang secara internasional dan sangat membantu terutama dalam penggunaan sumber daya manusia yang tidak dapat dengan sembarangan dipakai dalam prosedur pengiriman barang internasional.

Berikut adalah istilah-istilah yang berkaitan dengan *Freigth forwarding:*

1. *Letter of credit*

   Letter of credit, atau sering disingkat menjadi *L/C,LC,* atau *LOC,* adalah sebuah cara pembayaraninternasional yang memungkinkan eksportir menerima pembayaran tanpa

menunggu berita dari luar negri setelah barang dan berkas dokumrn dikirimkan keluar negeri kepada pemesan.

2. *Bill Of Lading*

   Bill of Lading adalah dokumen yang menunjukan adanya suatu kontrak pengangkutan bareang antara *shipper* sebagai pemilik barang – barang dengan *carrier* sebagai pihak yang menyewakan ruangan kapal untuk pengangkutan barang.

3. *Shipping* adalah kegiatan pengiriman barang yang melibatkan *shipper*, penyedia jasa, *consingee*, dan armada pengangkutan mitra bisnis penyedia jasa pengiriman barang
4. *Shipper* adalah orang yang mengirimkan barang
5. *Consingee* adalah penerima barang dari *shipper* melalui penyedia jasa layanan barang
6. CFS (*container freight station*) adalah tempat untuk pangepakan dan pembongkaran dari pengirim *LCL*. Kadang dikenal sebagai *C/B*.
7. *FCL/FCL* (*Full Container Load/Full Container Load*) adalah muatan dalam satu kontainer dikirim oleh satu perusahaan exportir, untuk 1 perusahaan importir di negra tujuan, ongkos angkutnya biasanya di hitung per kontainer, meskipun kontainer mungkin hanya diisi 10 MT.
8. *FCL/LCL* (*Full Container load/Less Container Load*) Adalah muatan dalam satu kontainer dikirim oleh satu perusahaan

eksportir, untuk beberapa perusahaan importir di negara tujuan yang sama. Ongkos angkutnya biasanya dihitung per kontainer, meskipun kontainer mungkin hanya di isi 10 MT.

9. *LCL/LCL* (*less container load/less container load*) Adalah Muatan dalam satu kontainer dikirim oleh beberapa perusahaan eksportir di negara asal yang sama, untuk beberapa perusahaan inportir di negara tujuan yang sama. Ongkos angkutnya biasanya di hitung per *MT/M3* berdasarkan berat atau volume, tergantung *ukuran/satuan* yang lebih besar.
10. *LCL/FCL* (*less container load/full container load*) Adalah muatan dalam satu kontainer dikirim oleh beberapa perusahaan eksportir, untuk satu perusahaan *inmortir* di negara tujuan. Ongkos angkutnya biasanya di hitung per *MT/M3* berdasarkan berat atau volume, tergantung *ukuran/satuan* yang lebih besar.
11. *Shiper on board (SOB)* adalah sebuah pengesahan yang mengkorfirmasikan bahwa barang sudah di muat.
12. Terminal adalah depot dimana kontainer melakukan proses pemuatan barang maupun pengosongan kontainer.
13. *Waybill* adalah *bill of lading* yang bertindak sebagai tanda terima untuk barang dan bukti dari pembawa barang.
14. *Invoice* adalah surat tagihan jasa pengiriman barang yang dikeluarkan oleh pihak penyedia jasa pengiriman brang kepada *shipper*.

2.1.8.10 Prosedur *Freight Forwarding*

Freight forwarder secara khusus mengatur pergerakan barang dari asal ke tujuan secara internasional melalui tahapan yang terbilang cukup panjang. Bagaimanapun juga, cara yang terbilang panjang ini menjadi masalah, dikarenakan *freight forwarder* memang mengkhususkan diri dibidang per-ijinan serta pajak dan lain sebagainya.

Proses dimulai dari penyiapan dokumen yang menjelaskan tentang produk apakah yang akan dimasukan ke dalam peti kemas *(kargo),* berjenis apakah benda tersebut, akan dibawa dari mana ke mana, dan akan melalui kota dan negara apa saja sebelum sampai di tujuan yang sebenarnya. Tidak lupa persoalan pajak dan ijin yang menjadi beberapa lembar surat, *Bill of Lading* bila metoda transportasi yang akan di gunakan ialah melalui laut. *Shipping Instruction* yang merupakan surat awal daripada pelanggan meminta *freight forwarder* untuk mengirimkan produk dari pelanggan, dan tentu *invoice* sebagai bukti pembayaran yang sah, baik dari *freight forwarder* kepada perusahaan kargo, maupun *freight forwarder* kepada pelanggan.

2.1.8.11 Tinjauan Pustaka

Studi ini menemukan inovasi layanan logistik di bidang logistik yang halal, juga telah mempertimbangkan faktor-faktor antara lain praktik kebersihan yang komprehensif, kualitas dan keamanan dan dengan demikian, penting bagi non-Muslim pelanggan. Penelitian ini memberikan kontribusi bagi kemajuan pengetahuan melalui penerapan konsep halal dalam praktek pelayanan logistik. Kebutuhan untuk menjadi inovatif dalam memulai jasa logistik yang lebih didasarkan pada konsep halal sangat penting dalam memenuhi kebutuhan meningkatnya permintaan oleh pelanggan khususnya umat Islam. Temuan memberikan wawasan kepada para praktisi penting untuk menjadi inovatif dalam menciptakan layanan logistik yang lebih halal untuk memenuhi meningkatnya permintaan produk halal.

. Menggunakan data mengangkat bahan bakar, angkutan udara dan gerakan udara kerajinan, dan asumsi atas beban penumpang rata-rata dan massa penumpang dan angkutan udara, emisi CO2 dari faktor 0,82 kg CO2 per t-km dan 0,69 kg CO2 per t-km untuk pendek-angkut dan jarak jauh perjalanan, masing-masing, dihitung. Jumlah total bahan bakar yang dikonsumsi untuk angkutan udara internasional impor Selandia Baru dan ekspor dihitung menjadi 0,21 dan 0,17 Mt Mt masing-masing, dengan emisi CO2 yang sesuai dari 0,67 dan 0,53 Mt Mt. Faktor-faktor ini dianalisis

dan kemudian diselenggarakan dalam kelompok-kelompok dalam hal kehandalan, lingkungan IFF bisnis, manajemen informasi, layanan dan harga.

Kelompok Keandalan ditemukan untuk menjadi yang paling penting bagi klien IFF. Hal ini diikuti oleh Dinas dan kelompok Harga. Kelompok kepentingan adalah sedikit Informasi Manajemen. Data untuk penelitian ini dikumpulkan dengan menggunakan kuesioner yang diberikan kepada sekitar 200 organisasi. Tingkat pengembalian adalah sekitar 25%. Reliabilitas kuesioner ini (s Cronbach) adalah 0,829, konsisten dengan fakta bahwa hanya 6% dari responden menyarankan alasan lebih lanjut untuk memilih perusahaan forwarding di luar yang disarankan.