BAB II

LANDASAN TEORI

2.1 Freight Forwarder

Menurut (Steven, 2014) *Freight Forwarder* adalah sebuah perusahaan yang bergerak dalam bidang keagenan yang mengurusi pengiriman dan penerimaan barang expor dan impor maupun pengiriman dalam negri. *Freight Forwarder* ini bisa dikatakan sebagai *agent shipping agent / carrier.* Secara keseluruhan,freight forwarding berfungsi sebagai Jasa pengiriman. Usaha jasa pengurusan transportasi (*freight forwading*) adalah kegiatan usaha yang ditujukan mengurus semua kegiatan yang diperlukan bagi terlaksananya pengiriman dan penerimaan barang melalui transportasi darat, laut atau udara yang dapat mencakup kegiatan penerimaan, penyimpanan, pengepakan, pengukuran, penimbangan, biaya-biaya lainnya berkenaan dengan pengiriman barang- barang tersebut sampai dengan diterimanya oleh yang berhak menerimanya.

Dari ke-tiga golongan tersebut, masing-masing adalah :

a. International *Freight Forwarder* (Klasifikasi A)

b. Domestik/*Regional Forwarder* (Klasifikasi B)

2.1.1 Kontainer

Menurut (Sylva, jakarta.kompasiana.com/transportasi, 2013) kontainer atau yang biasa sering disebut peti kemas merupakan suatu kemasan yang dirancang khusus dengan ukuran tertentu yang dapat dipakai berulangkali, dan bisa untuk menyimpan sekaligus mengangkut muatan yang ada di dalamnya. Filosofi dari kontainer adalah muatan bisa dibungkus dalam peti – peti dengan ukuran yang sama sehingga semua moda transportasi

kargo (kapal laut, truk, kereta api) bisa mengangkutnya secara cepat, aman, dan efisien.

2.1.2 Surat Jalan

Menurut (Dhimar, contohsuratku.com/contoh-surat-jalan-pengiriman-barang. 2013) Surat jalan dapat diartikan sebagai sebuah dokumen yang berupa surat dan memiliki fungsi sebagai surat pengantar terhadap barang yang telah dicantumkan di dalamnya. Surat ini dibuat dan ditujukan untuk penerima atau pembeli barang tersebut, dengan adanya surat ini tentunya akan membuat suatu barang menjadi legal yang biasanya sangat diperlukan ketika suatu barang telah dikeluarkan dari suatu perusahaan dan memasuki jalan raya sampai berada di tangan penerima atau pembeli. Tentunya dengan adanya surat jalan ini penerima barang akan dapat melihat secara jelas informasi barang yang diterimanya.

2.1.3 Pelayanan

Menurut (Lovelock, 2004) pelayanan adalah suatu kegiatan usaha melayani kebutuhan manusia melalui aktivitas orang lain.

2.1.4 Jasa

Menurut (Helmi, Zeithaml 2009) jasa adalah setiap tindakan aau unjuk kerja yang ditawarkan oleh salah satu pihak ke pihak lain yang secara prinsip *intangible* dan tidak menyebabkan perpindahan kepemilikan apapun. Produksinya bisa terkait dan bisa juga tidak terikat pada suatu produk fisik.

2.1.5 Pengiriman yang Baik

Menurut (Andi, belajarekspor.com 2009) ada beberapa hal yang harus dipertimbangkan perusahaan jasa pengiriman barang atau *exportir* untuk melaksanakan pengiriman import yang bisa dijadikan pertimbangan dalam proses pengiriman barang dari *packing*, menetukan tempat penyimpanan, dan menetukan jalur yang dilalui untuk pengiriman sehinnga barang sampai ke tempat tujuan.

2.1.6 *Packing*

Menurut (Andi, www.belajarekspor.com 2009) dalam proses *packing* ini perlu dilihat beberapa hal: jenis barang (pecah belah, cairan, makanan, dan atau jenis lainnya) dan bahan pembungkus barang. Beberapa ekspedisi tidak mau mengangkut barang pecah belah jika *packing* yang kita buat tidak kuat. Untuk pecah belah biasanya dilakukan *packing* tambahan berupa *packing* kayu. Tentu hal ini akan berpengaruh terhadap harga *packing*nya. Perlu diingat dalam proses pengiriman barang akan banyak dilakukan proses *handling*, mulai dari barang diambil di gudang ekspedisi, pada saat dimuat di kendaraan (truk) atau pada saat proses pemuatan di pesawat maupun setelah barang tiba di tujuan. Semakin banyak *handling* yang dilakukan maka semakin besar juga resiko kerusakan barang. Jadi jenis *packing* akan mempengaruhi barang tersebut rusak atau tidak.

2.2 Pemrograman Web

2.2.1 Pemrograman

Pemrograman adalah proses menulis, menguji dan memperbaiki *(debug)*, dan memelihara kode yang membangun sebuah program komputer. Kode ini ditulis dalam berbagai bahasa pemrograman. Tujuan dari pemrograman adalah untuk memuat suatu program yang dapat melakukan suatu perhitungan atau pekerjaan sesui dengan keinginan si pemrogram, diperlukan keterampilan dalam algoritma, logika, bahasa pemrograman, dan banyak kasus, pengetahuan lain seperti matematika.

Pemrograman adalah sebuah seni dalam menggunakan satu atau lebih algoritma yang saling berhubungan dengan menggunakan sebuah bahasa pemrograman tertentu sehingga menjadi sebuah program komputer. Bahasa pemrograman yang berbeda mendukung gaya pemrograman yang berbeda pula. Gaya pemrograman ini biasa disebut paradigm pemrograman.

2.2.2 Pendekatan Terstrukstur

Pendekatan Terstruktur adalah suatu proses untuk mengimplementasikan urutan langkah untuk menyelesaikan suatu masalah dalam bentuk program. Pendekatan terstruktur merupakan pendekatan formal untuk memecahkan masalah-masalah dalam aktivitas bisnis menjadi bagian-bagian kecil yang dapat diatur dan berhubungan untuk kemudian dapat disatukan kembali menjadi satu kesatuan yang dapat dipergunakan untuk memecahkan masalah. Pendekatan terstruktur dalam pengembangan system informasi adalah proses yang berorientasi kepada teknik yang digunakan untuk merancang dan menulis program secara jelas dan konsisten.

Prinsip dari pendekatan terstruktur adalah jika suatu proses telah sampai pada suatu langkah tertentu, maka proses selanjutnya tidak boleh mengeksekusi langkah sebelumnya. Ciri-ciri pendekatan terstruktur, yaitu:

1. Merancang berdasar modul. Modularisasi adalah proses yang membagi suatu system menjadi beberapa modul yang dapat beroperasi secara independen.
2. Bekerja dengan pendekatan *top-down*. Dimulai dari tingkat atas (secara global) kemudian diuraikan sampai ke tingkat modul (rinci).
3. Dilakukan secara iterasi.
4. Kegiatan dilakukan secara parallel.

2.2.3 Pemrograman Web

Menurut pemrograman web adalah proses menulis dan memelihara kode untuk membangun sebuah program komputer berbasis *client-server* yang kebanyakan berjalan di browser (missal: *web page*, aplikasi enterprise dan lain-lain.

2.2.4 Aplikasi Web

Menurut (Kadir, 2009) kepopuleran internet di seluruh dunia mendorong aplikasi web semakin diminati. Dengan menggunakan aplikasi web, anda hanya perlu menempatkan aplikasi dalam sebuah server dan dengan sendirinya aplikasi tersebut dapat diakses dariman pun, sepanjang pemakai dapat mengakses *web server-nya.* Aplikasi web yang paling dasar ditulis dengan menggunakan HTM. Sebagai mana diketahui, HTML *(HyperText Markup Language)* adalah bahasa standar untuk membuat halaman-halaman web.

2.2.5 Hypertext markup language

Menurut (Castro, 2010) *Hypertext Markup Language (HTML)* adalah bahasa yang digunakan untuk membuat suatu situs web atau *homepage.* Setiap dokumen dalam web ditulis dalam format *HTML*. Semua format dokumen, *hyperlink* yang dapat di klik, gambar, dokumen multimedia, *form* yang dapat diisi dan sebagainya didasarkan pada *HTML*.

2.2.6 Cascade Style Sheet

Cascading Style Sheet (CSS) adalah suatu bahasa *style sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. Penggunaan yang paling umum dari *CSS* adalah untuk memformat halaman web yang ditulis dengan *HTML* dan *XHTML*. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen *XML* termasuk *SVG* dan *XUL*. Spesifikasi *CSS* diatur oleh *World Wide Web Consortium (W3C).*

CSS digunakan oleh penulis maupun pembaca halaman web untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. *CSS* digunakan terutama untuk memisahkan antara isi dokumen (yang ditulis dengan *HTML* atau bahasa *markup* lainnya) dengan presentasi dokumen (yang ditulis dengan *CSS*). Pemisahan ini dapat meningkatkan akesesibilitas, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada struktur isi.

CSS memungkinkan halaman yang sama untuk ditampilkan dengan cara yang berbeda untuk metode presentasi yang berbeda, seperti melalui layar, cetak, suara (sewaktu dibacakan oleh *browser* basis-suara atau pembaca layar), dan juga alat

pembaca Braille. Halaman *HTML* atau *XML* yang sama juga dapat ditampilkan secara berbeda, baik dari segi gaya tampilan atau skema warna dengan menggunakan *CSS*.

2.2.7 Java Script

Javascript merupakan sebuah bahasa *script* (bahasa yang kodenya ditulis menggunakan teks biasa) yang ditempelkan pada kode HTML dan diproses di sisi klien.Dengan adanya bahasa ini, kemampuan dokumen HTML menjadi semakin luas. Sebagai contoh, dengan menggunakan *Javascript* dimungkinkan untuk memvalidasi masukan-masukan pada formulir sebelum formulir dikirimkan ke *server*. Selain itu, dengan menggunakan *Javascript* juga memungkinkan untuk mengimplementasikan tugas yang bersifat interaktif tanpa berhubungan dengan *server*.

*2.2.8 PHP (***Hypertext Prepocessor***)*

Hypertext Preprocessor (PHP) merupakan *script* untuk pemograman *script* web *server side*, *script* yang membuat dokumen *HTML* secara *on the fly*, dokumen *HTML* yang dihasilkan dari suatu aplikasi bukan dokumen *HTML* yang dibuat dengan menggunakan *editor* teks atau *editor HTML*.

2.3 Basis Data

Perkembangan sistem basis data semakin tahun semakin berkembang dengan tujuan untuk mengelola data dan informasi agar dapat memenuhi tuntutan kebutuhan masyarakat akan data spasial yang semakin meningkat melalui kegiatan peningkatan pendayagunaan basis data untuk menunjang berbagai kegiatan yang dibutuhkan manusia. Dari berbagai kegiatan ini memberikan suatu kemudahan

kepada pengguna data melalui pemanfaatan sistem penelusuran dan pemrosesan data.

2.3.1 Definisi Basis Data

Database adalah koleksi atau kumpulan data yang mekanis, terbagi / *shared*, terdefinisi secara normal dan dikontrol terpusat pada organisasi. (Everest. Gordon C, 2005). Sedangkan perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar disebut *Database Management Sistem (DBMS) .DBMS* dapat menjadi alternatif penggunaan secara khusus untuk aplikasi.

2.3.2 MySQL

MySQL adalah sebuah sebuah sistem manajemen basis data relasi *(Relational Database Management Sistem)* yang bersifat "terbuka" *(open source)*. MySQL adalah suatu perangkat lunak *database* relasi *(Relational Database Management System* atau RDBMS), seperti halnya ORACLE, Postgresql, MS SQL, dan sebagainya. MySQL AB menyebut produknya sebagai *database open source* terpopuler di dunia.

Berdasarkan riset dinyatakan bahwa bahwa di *platform web*, dan baik untuk kategori *open source* maupun umum, MySQL adalah *database* yang paling banyak dipakai. Menurut perusahaan pengembangnya, MySQL telah terpasang di sekitar 3 juta komputer. Puluhan hingga ratusan ribu situs mengandalkan MySQL bekerja siang malam memompa data bagi para pengunjungnya.

2.3.3 Normalisasi

Menurut (Janner Simarmata & Iman Peryudi, 2005) ”Normalisasi adalah teknik perancangan yang banyak digunakan sebagai pemandu dalam merancang basisdata relasional”. Pada dasarnya, normalisasi adalah proses dua langkah yang meletakkan data dalam bentuk tabulasi dengan menghilangkan kelompok berulang lalu menghilangkan data yang terduplikasi dari tabel relasional.

2.4 Alat Bantu Sistem

2.4.1 *Flowmap*

Bagan alir atau *flow map* adalah bagan yang menunjukan alir didalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan dokumentasi. Bagan alir dokumen atau sering disebut juga *Flow Map* merupakan bagan alir yang menunjukan arus dari aliran formulir termasuk tembusan-tembusannya.

2.4.2 Diagram Konteks

Diagram konteks menggambarkan aplikasi dalam satu lingkaran dan hubungan dengan entitas luar. Dimana lingkaran tersebut menggambarkan keseluruhan proses dalam aplikasi. Dalam penggambaran ini, sistem dianggap sebagai sebuah objek yang tidak dijelaskan secara rinci, karena yang ditekankan adalah interaksi sistem dengan lingkungan yang mengaksesnya.

2.4.3 *Data Flow Diagram*

Menurut (Jalinas, jalinas.staff.gunadarma.ac.id 2006) *data flow fiagram* adalah diagram yang berfungsi untuk menggambarkan aliran data dalam sistem, sumber dan tujuan data, proses yang mengolah data tersebut, dan tempat penyimpanan datanya. Representasi jaringan dari sistem yang menggambarkan sistem berdasarkan komponen-komponennya berdasarkan antarmuka antara komponen-komponen tersebut

2.4.4 Kamus Data

Kamus data atau sistem data *dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Pada perancangan sistem kamus dapat digunakan untuk merancang input, laporan-laporan dan database. Kamus data dibuat berdasarkan arus data yang ada di DFD. Kamus data berfungsi untuk membantu pelaku sistem untuk mengerti aplikasi secara detail dan mengorganisasikan semua elemen data yang digunakan dalam sistem secara presisi sehingga pemakai dan penganalisa sistem mempunyai dasar pengertian yang sama tentang masukkan, keluaran, penyimpanan dan proses.

2.4.5 *Entitiy Relationship Diagram* (ERD)

Entity Relationship Diagram merupakan jaringan yang menggunakan susunan data yang disimpann dari system secara abstrak. Diagram *Entitiy Relationaship* ini ditemukan oleh Chen tahun 1976. Tujuan dari *Entity Relationship* adalah untuk menunjukkan objek data dan *relationship* yang ada pada objek tersebut. Disamping itu Model ER ini merupakan salah satu alat untuk perancangan dalam basis data.

Komponen (Simbol) ERD

1. *Entity*

   Adalah suatu objek yang dapat dibedakan atau dapat diidentifikasikan secara unik dengan objek lainnya, dimana semua informasi yang berkaitan dengannya dikumpulkan. Kumpulan dari *entity* yang sejenis dinamakan *Entity Set*.

2. *Relationship*

   Adalah hubungan yang terjadi antara satu *entity* dengan *entity* lainnya. *Relationship* tidak mempunyai keberadaan fisik atau konseptual kecuali yang sejenis dinamakan dengan *Relationship* Diagram.

3. Atribut

   Adalah karakteristik dari *entity* atau *relationship* yang menyediakan penjelasan detail tentang *entity* atau *relationship* tersebut.

Cardinality Rasio

Yaitu menjelaskan batasan pada jumlah *entity* yang berhubungan melalui suatu *relationship*.

Jenis-jenis *Cardinality* Rasio :

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

   Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding satu berbanding satu.

   Contoh :



Gambar 2.1 *Cardinality* Rasio *One to One*

2. *One to Many* (1 : M)

   Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding satu berbanding banyak.

   Contoh :

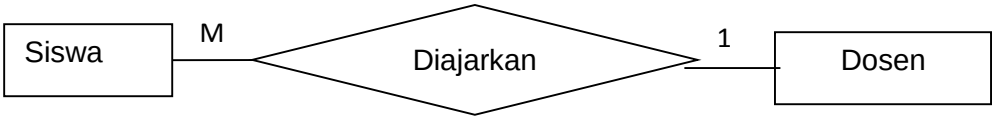


Gambar 2.2 *Cardinality* Rasio *One to Many*

3. *Many to One* (M : 1)

   Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding banyak berbanding satu.

   Contoh :



Gambar 2.3 *Cardinality* Rasio *Many to One*

4. *Many to Many* (M : M)

   Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding banyak berbanding banyak.

   Contoh :



Gambar 2.4 *Cardinality* Rasio *Many to Many*

Langkah – Langkah Membuat ERD:

1. Mengidentifikasikan dan menetapkan seluruh himpunan entitas yang akan terlibat.
2. Menentukan atribut-atribut *key* dari masing-masing himpunan entitas.
3. Mengidentifikasikan dan menetapkan seluruh himpunan relasi diantara himpunan entitas yang ada beserta *foreign key*-nya.
4. Menentukan derajat dan *cardinality* rasio relasi untuk setiap himpunan relasi
5. Melengkapi himpunan relasi dengan atribut-atribut yang bukan kunci (*non-key*).