**BAB II**

**LANDASAN TEORI**

* 1. **Tinjauan Pustaka**

Pada bagian ini dipaparkan teori-teori serta sumber pustaka sebagai landasan penulisan skripsi ini. Teori-teori ini diambil dari buku literatur dan dari internet. Teori yang dibahas meliputi teori tentang konsep dasar sistem, *tools*, bahasa pemrograman, sumber-sumber yang berkaitan dengan masalah yang dibahas di skripsi ini.

# Konsep Dasar Sistem Informasi

Menurut Tata (2012:38) “Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan setrategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang di perlukan oleh pihak tertentu”.

Konsep dasar sistem informasi terbagi menjadi beberapa bagian diantarannya sebagai berikut :

1. Komponen dan tipe sistem informasi

Sistem informasi terdiri dari komponen-komponen yang di sebut dengan istilah blok bangunan (*building block*) yang terdiri dari :

1. Blok Masukan (*input block*)

Input mewakili data yang masuk ke dalam sistem informasi. Yang di maksud dengan input di sini termasuk metode dan media untuk menangkap data yang akan di masukan, yang dapat berupa dokumen-dokumen dasar.

1. 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 susdah tertentu untuk menghasilkan keluaran yang diinginkan.

1. 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.

1. Blok Teknologi (*technologi block*)

Teknologi merupakan tool box dalam system informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan. Teknologi terdiri dari 3 bagian yaitu teknisi, perangkat lunak dan perangkat keras.

1. Blok Basis Data (*database block*)

Basis data (*database*) kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan perangkat lunak di gunakan untuk memanipulasinya

1. Blok kendali (*control block*) Banyak hal yang dapat merusak sistem informasi seperti bencana alam, api, temperatur dan debu, kecurangan kegagalan pada sistem itu sendiri, ketidakefisienan dan lain sebagainya.
2. Perencanaan Sistem Informasi

Perencanaan sistem informasi, terjemahan dari *Information System Planning* (ISP), menjelaskan bagaimana menerapkan tentang sistem pengetahuan tentang sistem informasi ke dalam organisasi agar dapat terus maju dan eksis bila organisasi berkembang sesuai dengan teknologi dan teori organisasi *modern*.

1. Pengolahan Sistem Informasi

Pengolahan sistem informasi merupakan bagian yang tak terpisahkan dari studi manajemen, sebagaimana halnya pengelolaan ketenagaan, keuangan, organisasi, tata laksana dan lain sebagainya. Barang kali dapat diasumsikan bahwa pengelolaan sistem informasi merupakan faktor kunci dari keberhasilan dan terlaksananya manajemen.

1. Pengendalian Sistem Informasi

Pengendalian Sistem Informasi merupakan bagian yang tak dapat di pisahkan dari pengelolaan sistem informasi bahkan memegang kunci yang sangat penting karena mengamati setiap tahapan dalam proses pengelolaan informasi.

1. Penilaian Sistem Informasi

Fungsi utama dari penilaian informasi adalah menyediakan informasi sebagai bahan pertimbangan utuk membuat keputusan. Siapa pun dapat membuat keputusan setelah dia memperoleh informasi yang merupakan hasil penilaian sesuai dengan tugas dan fungsinya, seperti pemimpin, pelaksana, penyuluh, pelatih, tenaga, dan lain sebagainya.

* + 1. **Konsep Dasar Sistem Inventori**

Sistem inventori adalah sistem yang mengelola suatu kegiatan transaksi-transaksi keluar masuknya barang. Manfaat dari sistem inventori adalah :

1. Menjaga agar persediaan di gudang selalu mencukupi
2. Meminimumkan biaya pemesanan dan biaya pengadaan persediaan barang

Pada dasarnya laporan inventori dimaksudkan untuk mengajukan informasi mengenai keadaan atau kondisi *stock* yang ada pada saat itu, yang akan digunakan oleh pihak yang berkepentingan sebagai pertimbangan di dalam pengambilan keputusan.

1. Mempertahankan stabilitas kelangsungan operasi organisasi dengan penyediaan barang.
2. Menghilangkan resiko keterlambatan datangnya barang yang dibutuhkan.
3. Memberikan pelayanan kepada pelanggan dengan sebaik-baiknya.

Selain itu laporan dengan sistem inventori bertujuan untuk mendapatkan kualitas suatu informasi antara lain :

1. *Relevan*

*Relevansi* suatu informasi harus dihubungkan dengan maksud penggunaannya. Bila suatu informasi tidak *relevan* untuk keperluan para pengambil keputusan, informasi demikian tidak ada gunanya betapa kualitas terpenuhi. Dalam pertimbangan *relevansi* suatu informasi, perhatian akan difokuskan pada kebutuhan pemakai dan bukan kebutuhan pihak tertentu.

1. Dapat dimengerti

Informasi harus dapat dimengerti oleh pemakainya.

1. Netral

Informasi diarahkan pada kebutuhan umum pemakai dan tidak tergantung pada kebutuhan dari pihak tertentu.

1. Tepat waktu

Informasi harus disampaikan sedini mungkin untuk dapat digunakan sebagai bahan pertimbangan dalam pengambilan keputusan ekonomi dan untuk menghindari tertundanya keputusan tersebut.

1. Akurat

Informasi harus akurat, harus bebas dari kesalahan-kesalahan dan jelas maksudnya. Informasi harus akurat dari banyaknya kemungkinan terjadi gangguan dan penyimpangan dan dapat merubah bahkan merusak informasi tersebut.

Ada beberapa sistem pencatatan persediaan yaitu:

1. *Periodic System*, yaitu pada setiap akhir periode dilakukan perhitungan secara fisik dalam menentukan jumlah persediaan akhir.
2. *Perpetual System* atau juga disebut *Book Inventories*, yaitu setiap mutasi dari persediaan sebagai akibat dari pembelian ataupun penjualan dicatat atau dilihat dalam kartu administrasi persediaannya.

Ada beberapa cara yang dapat digunakan untuk menilai suatu persediaan, diantaranya dengan :

1. *first-in, first out* (FIFO)
2. *last in, first-out* (LIFO)
3. rata-rata tertimbang (*weighted average*)
   * 1. **Metodologi Pengembangan *Waterfall***

Menurut Shalahuddin (2015:28) “Model SDLC air terjun (*waterfall*) sering juga disebut Model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*)”. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung(*support*).

Berikut gambar ilustrasi model *waterfall* (M.Shalahuddin, 2015:29) :



Dalam pengembangannya metode *waterfall* memiliki beberapa tahapan yaitu: *requirement* (analisis kebutuhan), design sistem (*system design*), *Coding & Testing*, Penerapan Program, pemeliharaan.

1. *Requirement* (analisis kebutuhan).

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau *study literatur*. Seseorang sistem analisis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan sistem analisis untuk menterjemahkan kedalam bahasa pemrograman.

1. *Design System* (disain sistem)

Proses *design* akan menterjemahkan syarat kebutuhan kesebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, *representasi interface*, dan *detail* (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistemnya

1. *Coding & Testing* (penulisan sinkode program / *implemention*)

*Coding* merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh *programmer* yang akan meterjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

1. Penerapan / Pengujian Program (*Integration & Testing*)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, *design* dan pengkodean maka sistem yang sudah bisa digunakan oleh *user*.

1. Perangkat lunak yang susah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (*periperal* atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.
2. **Tools Program**

# Pengenalan *PHP (PHP Hipertext Prepocessor)*

Menurut Sutaji (2012:2) menyimpulkan bahwa “PHP adalah kode/skrip yang akan dieksekusi pada server side”. *Skrip PHP* akan membuat suatu aplikasi dapat di-integrasikan ke dalam *HTML*, sehingga suatu halaman *web* tidak lagi bersifat statis, namun menjadi bersifat dinamis.

*PHP* bersifat *open source*, sehingga dapat dipakai secara cuma-cuma dan mampu lintas *platform*, yaitu dapat berjalan pada sistem operasi *Windows* maupun *Linux*. Selain itu *PHP* sudah dapat berinteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda, yaitu seperti: *DBM, FilePro (Personic, Inc*), *Informix, Ingers, InterBase, Microsoft Access, MSSQL, MySql, Oracle*, *PostgrSQL*, dan *Sybase*”. Dengan demikian, menampilkan data yang bersifat dinarnis, yang diambil dari *database*, merupakan hal yang mudah untuk diimplementasikan. Itulah sebabnya sering dikatakan bahwa *PHP* sangat cocok untuk membangun halaman-halaman *web* dinamis.

1. ***Cascading Style Sheet* (*CSS*)**

*Cascading Style Sheet (CSS)* atau yang biasa disingkat dengan CSS, merupakan suatu dokumen yang digunakan untuk melakuakan pengaturan halaman *Web* yang ditulis dengan *HTML* atau *XHTML*. Menurut Kadir dkk (2013:323) “CSS adalah kode yang dimaksudkan untuk mengatur tampilan halaman web”

*CSS* digunakan oleh *Web Designer* 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*). Pemisah ini ditujukan agar dapat mmemisahkan *aksesibilitas* isi, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada strukur isi.

# *PHPMyAdmin*

*PHPMyAdmin* merupakan sebuah program yang berbasis *web* yang dibuat menggunakan aplikasi *PHP*. Program ini berfungsi untuk mengakses *database MySQL*, Menurut Sukarno ( 2006:143 ) “untuk melakukan pembuatan dan pengelolaan *database* dapat dilakukan pada browser, yaitu yang dikenal dengan *PHPMyadmin*”. Bisa juga disebut sebagai fungsi *PHP-MySQL*.

Menurut Sukarno ( 2006:211 ) ” fungsi *php-mysql* adalah suatu fungsi yang menjembatani antara *php* sebagai *programming web server*, dan *mysql* sebagai *database*, sehingga data-data yang terdapat pada *database mysql* dapat ditampilkan pada *browser*”. Dengan *PHPMyadmin* pengguna awam tidak harus mampu mengetahui sintak-sintak *SQL* dalam pembuatan *database* dan *table*.

# *MySQL*

Menurut Sukarno ( 2006:3 ) “*MySQL* adalah merupakan perangkat lunak untuk sistem manajemen database ( *Database Managemen System* )”. *MySQL* salah satu jenis database *server* yang sangat terkenal. Kepopulerannya disebabkan *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses databasenya. Pada *MySQL* sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. Untuk mengelola *database MySQL* ada beberapa cara yaitu melalui *prompt* DOS *(tool command line)* dan dapat juga menggunakan program *utility* seperti:

* + 1. *PHP MyAdmin*
    2. *MySQLGUI*
    3. *MySQL Manager Java Based*
    4. *MySQL Administrator for windows.*

1. **Pengenalan *Xampp***

*Xampp* merupakan *softwere* yang mudah digunakan, gratis dan mendukung instalasi di *linux* dan *Windows*. Keuntungan lainnya adalah hanya menginstal satu kali sudah tersedia *Apache, Web Server, Mysql Database Server*.

Fungsinya sebagai *server* yang berdiri sendiri atau biasa disebut dengan (*localhost*), yang terdiri atas program *Apache HTTP Server, MYSQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP dan Perl*. Nama *Xampp* merupakan singkatan dari X (empat system operasi apapun), *Apache, MySQL, PHP* dan *Perl*.

# *Macromedia Dreamweaver 8*

Salah satu *software* web *editor* adalah *Macromedia Dreamweaver 8* yang merupakan penyempurnaan dari versi sebelumnya dan tentu saja semakin mudah dalam penggunaannya. Oleh karena itu, *software* ini paling *inovatif* dan lebih lengkap dibandingkan *software web editor* lain. Adapun pengertian dari *Macromedia Dreamweaver 8* ini adalah program aplikasi *professional* untuk mengedit *HTML* secara *visual*. Program Aplikasi *Macromedia Dreamweaver 8* menyertakan banyak perangkat yang berkaitan dengan pengkodean dan *fitur* seperti *HTML, CSS*, serta *JavaScript*.

Fasilitas terbaru dari *Macromedia Dreamweaver 8* adalah *Zoom Tool and Guides*, *Panel CSS* yang baru, *Code Collapse*, *Coding Toolbar*, dan *Insert Flash Video*. *Macromedia Dreamweaver 8* mendukung pemrograman *script server-side*, seperti *PHP, ASP, ASP.NET*, *ColdFusion* dan *JSP.* Pemrograman *script server-side* maksudnya adalah *script* yang digunakan dalam pemrograman *web* dinamis dimana semua perintahnya dieksekusi pada *server.* Fungsi *server* disini adalah sebagai pemroses *script* dan hasilnya dikembalikan dalam bentuk *tag-tag* *HTML* yang kemudian ditampilkan dalam *browser.*

# *Unified Modeling Language (UML)*

Menurut Shalahuddin (2015:137) “ *UML* merupakan bahasa visual untuk permodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

Menurut Shalahuddin (2015:140) “ Secara fisik, *UML* adalah sekumpulan spesifikasi yang dikeluarkan oleh *OMG* ( *Object Management Group* )”. *UML* terbaru adalah *UML 2.3* yang terdiri dari 4 macam spesifikasi yaitu *diagram* *interchange Specification, UML Infrastructure, UML Superstructure*, dan *Object Constraint Languange (OCL)*. Seluruh spesifikasi dapat diakses di *website* *http://www.omg.org*. Berikut beberapa jenis diagram :

1. ***Diagram Use Case (Use-Case-Diagram*)**

Menurut Shalahuddin (2015:155) “*Use case* merupakan permodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat”. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem informasi yang akan dibuat. Ada dua hal utama pendefinisian pada *use case* yaitu aktor dan *use case*.

1. Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan di buat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. Menggambarkan pihak-pihak yang berperan dalam sistem.
2. *Use Case* merupakan fungsionalitas yang di sediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
3. **Diagram aktivitas (*Activity Diagram*)**

Menurut Shalahuddin (2015:161) “*Activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor jadi aktivitas yang dapat dilakukan oleh sistem.

1. **Diagram Interaksi (*Sequence Diagram*)**

Menurut Shalahuddin (2015:165) “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”. Oleh karena itu untuk menggambar diagram sekuen maka harus di ketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang yang di instansiasi menjadi objek itu.

1. **Diagram *Deploymen* (*Deployment Diagram* )**

Menurut Shalahuddin (2015:154) “*Deployment diagram* menujukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Diagram deploymen* juga dapat digunakan untuk memodelkan hal-hal berikut :

1. Sistem tambahan (*embended system*) yang menggambarkan rancangan *device, node*, dan *hardware.*
2. Sistem *client atau server* dan *hardware*.
3. Sistem terdistribusi murni.
4. Rekayasa ulang aplikasi.
5. ***Entity Relationship Diagram* (*ERD*)**

Menurut Shalahuddin (2015:55) “*Entity Relationship Diagram* adalah bentuk paling awal dalam melakukan perancangan basis data relasional”.

Sebelum kita membahas lebih jauh tentang bagaimana diagram *ER* tersebut dapat kita gambarkan, maka terlebih dahulu kita mengetahui komponen-komponen pembentuk *ER* sesuai namanya ada (dua) yaitu komponen utamannya yaitu Entitas (*entity*), Relasi (*relation*) komponen yang kedua ini dideskripsikan lebih jauh melalui sejumlah *Atribut* atau *Property*.

1. Entitas (*entity*)

Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat di bedakan sesuatu yang lain. Contoh entitas :

1. Semua pelanggan atau pelanggan saja :

Dengan entitas Budiman, Suherman, Aminah dan seterusnya.

1. Semua Mobil atau mobil saja :

Dengan entitas mobil Suzuki, mobil Honda, mobil Toyota dan lainnya

1. Semua kampus atau kampus saja :

Dengan entitas Mahasiswa atau mahasiswa saja.

1. Atribut (*Attributed Properties*)

Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (*property*) dari entitas tersebut. Sebagaimana telah disebutkan sebelumnya, penentuan atribut-atribut yang relavan bagi sebuah entitas merupakan hal penting lainnya dalam pembentukan model data. Yang relavan untuk lebih di perhatikan dalam pembentukan model *ER* adalah kedudukan atribut dan entitas. Harus dapat kita bedakan dan ketahui mana atribut yang berfungsi *key primer* dan mana yang bukan (*atribut deskriptif*). Contoh :

1. Atribut nim merupakan *key* untuk atribut himpunan mahasiswa, karena nim merupakan pengidentifikasi entitas yang unik untuk semua entitas dalam himpunan entitas tersebut. Atribut lainya (nm\_mhs, almt\_mhs, tgl\_lhr) merupakan atribut *deskriptif*.
2. Kumpulan atribut kode\_jns\_brg dan no\_brg merupakan *key* himpunan entitas barang.
3. Relasi (*Relationship*) dan Himpunan relasi

Relasi menunjukan adanya hubungan di antara jumlah entitas yangberasal dari himpunan entitas yang berbeda. Kumpulan semua relasi di antara entitas-entitas yang terdapat di himpunan entitas, himpunan entitas itu membentuk himpunan relasi (*relationship sets*). Sebagaimana istilah himpunan entitas yang banyak sekali disingkat menjadi entitas (walaupun sebenarnya memiliki perbedaan makna) istilah relasi jarang sekali digunakan dan lebih sering di singkat relasi saja. Relasi antar kelas dikelompokan dalam tiga jenis :

1. *One To One*

Kerelasian jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi sebuah kejadian atau transaksi pada kedua entitas.

Contoh : setiap mahasiswa hanya dimungkinkan mempunyai seorang wali mahasiswa.

1. *One To Many*

Kerelasian jenis ini terjadi jika kejadian atau transaksi di anatra dua entitas yang berhubungan hanya memungkinkan terjadi satu kali dalam entitas pertama dan dapat terjadi lebih dari satu kali kejadian atau transaksi pada entitas kedua. Contoh : setiap mahasiswa hanya memungkinkan memilih sebuah program studi, sebaiknya sebuah program studi dapat dipilih oleh lebih dari satu orang mahasiswa.

1. *Many To Many*

Kerelasian jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi lebih dari satu dalam entitas pertama dan entitas kedua. Contoh : Lebih dari satu mahasiswa dapat memilih lebih dari satu mata kuliah.

# *Website*

*Website* merupakan *webpage* beserta *homepage*, yang merupakan *system* yang luas dari *server* yang menawarkan informasi untuk semua orang melalui jaringan. Informasi tersebut tersebut berupa teks, gambar, suara, dan tipe data *link*, yang merupakan navigasi kehalaman lainnya (*hyperlink*). *Homepage* merupakan halaman yang pertama kali muncul ketika pertama kali dibuka,sedang *webpage* adalah halaman yang ada pada *website*.

# *Web Browser*

Dalam dunia *web*, perangkat lunak *client*, yaitu *browser web* mempunyai tugas yang sama yaitu menterjemahkan informasi yang diterima oleh *server web* dan menampilkannya pada layer komputer pengguna, oleh karena *HTTP* memungkinkan *server web* mengirimkan beragam data, seperti teks atau gambar, *browser* harus bisa mengenali berbagai macam data yang akan diterimanya, dan selanjutnya harus tahu cara untuk menampilkanya dengan benar. Teks ditampilkan sebagai teks dan gambar ditampilkan sebagai gambar. Umumnya *browser web* menerima data dalam bentuk *HTML*. *File* *HTML* sebenarnya adalah *file* teks biasa yang selain berisi informasi yang hendak ditampilkan kepada pengguna, juga mempunyai perintah - perintah untuk mengatur tampilan data tersebut. *Browser* lah yang memiliki kuasa penuh dalam menterjemahkan perintah - perintah tadi. Meskipun sudah dibuat *consensus* untuk menstandarkan format dan elemen - elemen *HTML*, setiap jenis *browser* bisa menterjemahkan *file HTML* secara berbeda.

Beberapa *server web* memiliki *feature* seperti *server side programming, sevurity control* dan lain sebagainya. Meskipun beragam macamnya, secara fungsional semua jenis *server web* adalah sama saja, yaitu berfungsi melayani permintaan - permintaan dari *browser web*.

Banyak *web browser* yang bisa digunakan untuk mengakses *web*, diantaranya *google crome*, dan masih banyak lagi *web browser* lain yang bisa digunakan untuk mengakses *web*.

1. **Sekilas Tenang *Google Chrome***

September 2008, *Google* melancarkan sebuah serangan langsung ke lingkup *web browser* dengan meluncurkan *Chrome*, sebuah *web browser* anyar yang diklaim memiliki terobosan baru dalam kecepatan, kemudahan penggunaan, serta integrasi yang baik dengan aplikasi berbasis *web*.

Sehari sebelum peluncuran tanggal 2 September 2008, *Google* mengeluarkan sebuah komik *online* yang menjelaskan latar belakang dibuatnya *Chrome* ini. Lewat komik tersebut, *user* dapat mendapatkan penjelasan lengkap mengenai *web browser*.

Aplikasi ini menelurkan terobosan baru dalam segi pengelolaan memori pada suatu *browser*. Setiap tab di *browser* akan memiliki *mapping* sendiri dalam memori. Oleh karena itu, setiap *tab* akan memiliki proses tersendiri dalam sistem operasi. Jika ada satu *tab* yang mengalami *error*, maka hal tersebut tidak akan mempengaruhi *tab* lain dan *browser* secara keseluruhan.

*Google* juga telah menemukan *algoritma* efisien untuk melakukan *rendering* terhadap *Javascript*. Dari hasil *benchmark* awal yang dilakukan *CNet, Chrome* unggul mutlak dalam hal kecepatan melakukan *render Javascript.* Pada era *web* 2.0 yang bertumpu pada teknologi *Javascript* seperti *AJAX*, *Chrome* dapat menjadi aplikasi pilihan para pengguna *browser* untuk mendapatkan kenyamanan dalam merambah internet.

*Chrome* juga mengintegrasikan *Google Gears* di dalam inti *browser* tersebut. Seperti yang diketahui *Gears* merupakan tools dari *Google* yang memungkin aplikasi *web-base Google* seperti *Docs, Readers*, ataupun *Gmail* dijalankan secara *offline*.

Lewat *feature Gears* dan kecepatan dalam *rendering Javascript*, *Google Chrome* dikonsepkan sebagai dasar untuk memperkuat lini aplikasi *web* *Google* seperti *Docs*, *Spreadsheet*, *Calendar*, dan sebagainya. *Google* memang mempromosikan *trend* penggunaan aplikasi berbasis *web* menggantikan aplikasi *desktop* konvensional sebagai solusi aktivitas berkomputasi masa depan.

*Google* menohok langsung ke jantung pasaran *web browser* yang masih didominasi oleh *Internet Explorer* (sekitar 75%). Langkah peluncuran ini dianggap sebagai langkah antisipatif *Google* dalam menghadapi datangnya *Internet Explorer* versi 8. Bukan rahasia lagi bila penguasaan pasar *browser* oleh *Internet Explorer* tersebut berdampak serius pada bisnis layanan pencarian *Google*.

# *World Wide Web (WWW)*

Sebuah situs *web* adalah sebutan bagi sekelompok halaman *web* (*web page*), yang umumnya merupakan bagian dari suatu nama domain (*domain name*) atau *subdomain* di *World Wide Web* (WWW) di Internet. WWW terdiri dari seluruh situs *web* yang tersedia kepada publik. Halaman-halaman sebuah situs *web* diakses dari sebuah *URL* yang menjadi "akar" (*root*), yang disebut *homepage* (halaman induk; sering diterjemahkan menjadi "beranda", "halaman muka"), dan biasanya disimpan dalam *server* yang sama. Tidak semua situs *web* dapat diakses dengan gratis. Beberapa situs *web* memerlukan pembayaran agar dapat menjadi pelanggan, misalnya situs-situs yang menampilkan *film cinema*, situs-situs berita, layanan surat elektronik (*e-mail*), dan lain-lain.

Terminologi website adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah *domain* atau *subdomain*, yang tempatnya berada didalam *World Wide Web (*WWW) di Internet. Sebuah *web* *page* adalah dokumen yang ditulis dalam format *HTML (Hyper Text Markup* *Language*), yang hampir selalu bisa diakses melalui *HTTP*, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar.

* 1. **Penelitian Terkait**

Berikut ini beberapa penelitian-penelitian terkait yang menjadi referensi penulisan skripsi ini :

Menurut Noviandi,dkk (2012:1) Bank Sampah Garut merupakan sebuah perusahaan yang bergerak dibidang pemanfaatan pengelolaan sampah yang berasal dari limbah rumah tangga, kantor, dan lembaga pendidikan (sekolah). adapun sampah yang diolah adalah berupa sampah anorganik seperti, sampah plastik, kertas, logam, dll. Dewasa ini jumlah dari nasabah Bank Sampah Garut meningkat cukup pesat, hal tersebut secara otomatis tentu akan membuat barang yang akan masuk bertambah banyak. Bank Sampah Garut awalnya menjadwalkan penarikan sampah setiap seminggu sekali ketiap Bank Sampah Unit Sekolah, tentunya akan mendapatkan permasalahan baru dalam hal penyimpanan barang karena daya tamping gudang yang tetap, dan barang yang akan di daur ulang oleh bagian produksi maupun barang yang akan dijual kepihak lain masih relative tetap, hal tersebut tentunya akan berpotensi membuat barang yang disimpan melebihi dari daya tamping gudang. Selain itu administrasi transaksi barang yang masuk kegudang saat ini masih menyatu dengan pencatatan transaksi penarikan tabungan sampah ke Bank Sampah Unit Sekolah. Begitu juga dalam pencatatan data barang keluar, baik itu sampah yang akan di daur ulang sendiri oleh Bagian Produksi di Bank Sampah Garut, maupun barang yang dijual ke pihak lain, keduanya belum memiliki sistem inventori yang terpisah. Menurut Noviandi,dkk (2012:2) Hal–hal tersebut diatas, dapat dihindarkan apabila terdapat suatu sistem persediaan barang yang lebih baik dan dapat menghasilkan sebuah penjadwalan pengambilan setoran tabungan yang tepat. Sistem informasi inventori barang ini sangat penting untuk mendukung kegiatan operasional suatu perusahaan. Sistem Informasi yang dibutuhkan perusahaan khususnya tentang inventory barang dengan aplikasi komputer, diharapkan dapat mempercepat dalam menyelesaikan pekerjaan perusahaan dan dihasilkan data yang akurat dengan waktu yang lebih cepat.

Menurut Rusdah (2011:1) Puskesmas Kecamatan Kebon Jeruk mengalami kesulitan dalam administrasi pendataan obat, karena selain untuk kebutuhan sendiri, Puskesmas Kecamatan Kebon Jeruk menyediakan obat untuk digunakan oleh 7 (tujuh) Puskesmas Kelurahan yang ada diwilayah Kecamatan Kebon Jeruk. Sistem manual pada Puskesmas Kecamatan Kebon Jeruk yang berjalan saat ini tentu tidak dapat berjalan dengan maksimal dikarenakan adanya kesalahan –kesalahan yang mungkin dilakukan seperti pencatatan data obat yang salah atau kurang teliti sehingga mengakibatkan persediaan obat menjadi kosong atau berlebih, dan menumpuknya obat kadaluarsa.Untuk menghindari kemungkinan timbulnya kesalahan – kesalahan tersebut maka dibutuhkan suatu sistem terkomputerisasi guna menyediakan informasi yang lebih cepat, tepat dan akurat kepada pihak gudang dan unit / Puskesmas Kelurahan. Menurut Rusdah (2011:9) Sistem Informasi Persediaan Obat pada Puskesmas Kecamatan Kebon Jeruk, dikembangkan untuk memberikan solusi dari permasalahan pendaataan obat yang selama ini dihadapi oleh bagian gudang farmasi.