BAB II
TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Informasi merupakan sumber daya yang strategis sepanjang hidup. Sebagai negara yang sedang membangun maka informasi merupakan bagian yang sangat penting dalam pembangunan. Selain itu informasi juga sangat diperlukan dalam bidang pendidikan dan penelitian untuk pengembangan ilmu pengetahuan, teknologi dan juga seni budaya.

Suatu perpustakaan, baik itu besar ataupun kecil, perlu diatur dengan sistem agar dapat memberi pelayanan yang baik dan memuaskan kepada para pengunjungnya. Setiap pengunjung yang ingin menggunakan perpustakaan tentunya mengharapkan pelayanan yang baik dari pihak perpustakaan itu sendiri. Pelayanan yang cepat dan tanggap merupakan salah satu yang diharapkan. Artinya untuk memperoleh layanan orang tidak harus menunggu lama. Juga diharapkan pelayanannya tepat waktu dan benar.

2.1.1 Perpustakaan

Perpustakaan adalah jendela dunia. Perpustakaan harus dapat memberikan bantuan dengan sebaik-baiknya kepada para pembaca untuk memperoleh bahan pustaka ataupun informasi yang sesuai dengan kebutuhannya. Oleh karena itu kegiatan pelayanan merupakan cara untuk mempertemukan para pembaca dan informasi yang dicari. Pelayanan yang baik merupakan salah satu hal terpnting dalam pengelolaan perpustakaan. Pelayanan yang ramah dan menarik dan mudah, tentulah dapat menarik minat para pengguna perpustakaan untuk terus berada di dalamnya.
Basuki Sulistyo dalam bukunya (Sulistyo, 2004) mengatakan bahwa :

> *"Perpustakaan adalah sebuah ruangan atau gedung yang digunakan untuk menyimpan buku dan terbitan lainnya yang biasanya disimpan menurut tata susunan tertentu yang digunakan pembaca bukan untuk dijual."*

2.1.2 Pengertian Sistem

Pengertian sistem sudah umum dan biasa dipakai dalam kehidupan sehari-hari, misalnya sistem pengajaran, sistem komputer, dan sistem tata surya. Dalam kehidupan sehari-hari ilmuwan mengartikan sistem sebagai suatu wadah yang menampung beberapa elemen yang saling berinteraksi di dalamnya.
Hanif Al Fatta dalam bukunya (Al Fatta, 2007) mengatakan bahwa :

> *"Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu."*

2.1.3 Konsep Analisis dan Perancangan Sistem

2.1.3.1 Pengertian Analisis Sistem

Analisis sistem adalah suatu proses analisa terhadap masalah yang perlu diselesaikan oleh sebuah organisasi dengan menggunakan sistem informasi, yang terdiri dari pendefinisian dan identifikasi spesifikasi kebutuhan yang harus dipenuhi oleh solusi dari sistem (Al Fatta, 2007). Analisa sistem adalah proses umum yang mencakup pendefinisian masalah, pengumpulan informasi yang berkaitan, pengembangan solusi, alternative, dan pemilihan di antara solusi yang ada tersebut (Al Fatta, 2007).

2.1.3.2 Pengertian Perancangan Sistem

Perancangan sistem adalah bagian dari perencanaan bisnis yang berhubungan dengan penyebaran sumber daya sistem informasi perusahaan yang mencakup manusia, perangkat keras dan perangkat lunak (Al Fatta, 2007).

2.1.4 Basis Data (Database)

Sistem basis data merupakan penyimpanan informasi yang terorganisasi secara komputerisasi sehingga memudahkan pemakai dalam pengolahannya dan penggunaannya (Subekti, 1997). Pada saat ini sistem basis data sudah dikembangkan pada mesin-mesin komputer kecil sampai komputer yang lebih besar seperti *mainframe*.

2.1.5 Normalisasi

Normalisasi merupakan suatu proses yang memudahkan desain struktur tabel secara benar sehingga *query* yang tak dapat ditanyakan tidak muncak. Di samping itu, normalisasi cenderung meminimumkan duplikasi data dalam suatu basis data. Ini memilliki keunggulan dalam mengurangi ruang simpan yang dibutuhkan maupun mempercepat *query*.

Terdapat tiga bentuk normal yang umum digunakan, yaitu 1NF (*First normal form*), 2NF (*second normal form*). Normalisasi ini dilakukan berdasarkan *functional dependency* antara atribut dalam sebuah relasi. *Functional dependency* ialah batasan antar dua set atribut dari basis data (Elmasri et al, 2000).

a. 1NF (First Normal Form)

Pada bentuk normal pertama (*first normal form* atau 1NF), sebuah relasi dimana setiap sel (perpotongan baris dan kolom) hanya mengandung satu nilai, setiap mengandung nilai antomic (*single value*). Untuk menjadikan bentuk tidak normal menjadi normal dengan cara mengisi data yang sesuai kedalam kolom kosong dari baris yang berulang, disebut dengan *flattening the table.*

b. 2NF (Second Normal Form)

Sebuah tabel berada pada bentuk normal kedua (2NF) jika dan hanya jika setiap atribut bukan *primary key* (PK) tergantung sepenuhnya pada *primary key* (PK). Untuk mengetahui apakah 1NF telah berada 2NF maka tentukan *primary key* dan *functional dependency*. Dalam bentuk kedua (2NF) tidak boleh ada partial *dependency*, atribut yang partial dependent harus dipisahkan.

2.1.6 Alat Pengembangan Sistem

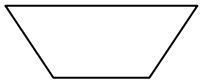
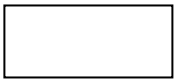
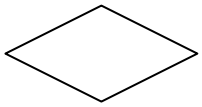
Alat Pengembangan Sistem adalah Menyusun sesuatu yang baru untuk menggantikan atau mengembangkan sistem yang lama secara keseluruhan (Purnomo, *2007*).

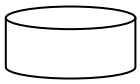
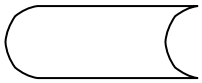
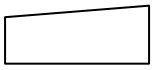
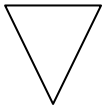
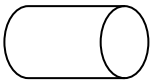
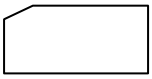
Komponen-komponen dalam pengembangan atau penyusunan sistem adalah sebagai berikut :

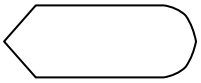
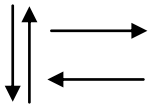
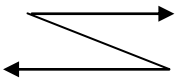
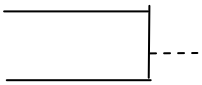
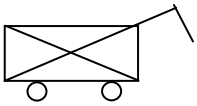
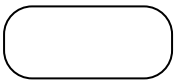
2.1.6.1 Diagram Alir Dokumen (*Mapping Chart*)

Diagram alir dokumen (*Mapping Chart*) merupakan bagan alir yang menunjukkan arus dari laporan formulir termasuk tembusan-tembusannya (Purnomo, 2006). Simbol-simbol dapat dilihat pada Tabel 2.1.

Tabel 2.1 Diagram Alir Dokumen

NO	Simbol	Keterangan
1		**Simbol Dokumen** Sebagai simbol dokumen *input* dan *output* baik untuk proses manual maupun komputerisasi.
2		**Simbol Kegiatan Manual** Sebagai simbol yang menunjukkan proses kegiatan yang dilakukan secara manual.
3		**Simbol Proses** Sebagai simbol yang menunjukkan kegiatan proses yang dilakukan menggunakan program aplikasi di komputer.
4		**Simbol Keputusan** Sebagai simbol yang digunakan untuk suatu penyelesaian kondisi dalam program.

5		**Simbol *Harddisk*** Sebagai simbol yang menunjukkan penyimpanan *input* atau *output* dalam *harddisk*.
6		**Simbol Disket** Sebagai simbol yang menunjukkan penyimpanan dalam disket.
7		**Simbol *Keyboard*** Sebagai simbol yang menunjukkan *input* menggunakan *online keyboard.*
8		**Simbol Arsip** Sebagai simbol yang menunjukkan penyimpanan berkas yang dipakai lagi.
9		**Simbol Penghubung** Sebagai simbol yang menunjukkan penghubung halaman yang sama (1) dan Menunjukkan penghubung ke halaman yang berbeda (2).
10		**Simbol Uang** Sebagai simbol yang menunjukkan alat tukar pembelian.
11		**Simbol Pita *Magnetic*** Menunjukkan *input* atau *output* menggunakan pita *magnetic.*
12		**Simbol *Drum Magnetic*** Sebagai simbol yang menunjukkan *input* atau *output* menggunakan drum magnetik.
13		**Simbol Kartu Plong** Sebagai simbol yang menunjukkan *input* atau *output* yang menggunakan kartu plong.

14		**Simbol *Display*** Sebagai simbol yang menunjukkan *output* yang ditampilkan di *monitor*.
15		**Simbol Garis Alir** Sebagai simbol yang menunjukkan arus dari proses.
16		**Simbol Hubungan Komunikasi** Sebagai simbol yang menunjukkan proses transmisi dan data melalui *channel*.
17		**Simbol Penjelasan** Sebagai simbol yang menunjukkan penjelasan dari suatu proses.
18		**Simbol Barang** Sebagai simbol yang menunjukkan pengiriman barang.
19		**Simbol Operasi Luar** Sebagai simbol yang menunjukkan operasi yang dilakukan di luar proses komputer.
20		**Titik Terminal** Sebagai simbol yang menunjukkan awal dan akhir dari suatu proses.

2.1.6.2 *Data Flow Diagram* (DFD)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan tempat asal data dan tempat tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang dihasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenalkan pada data tersebut (Kristanto, 2007).

Simbol yang digunakan dalam DFD adalah sebagai berikut :

a) *External entity* (Kesatuan luar)

Simbol *External Entity* tertera pada Gambar 2.1

Gambar 2.1 Simbol *External Entity*

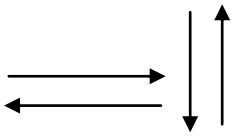
Kesatuan luar merupakan *subyek* di lingkungan luar sistem yang memberikan *input* atau *output*.

Kesatuan luar dapat berupa :

1) Kantor, departemen ataupun divisi dalam perusahaan tetapi di luar sistem yang sedang dikembangkan.
2) Orang atau sekelompok orang di organisasi tetapi di luar sistem yang sedang dikembangkan.

b) *Data Flow* (Arus Data)

Simbol *Data Flow* tertera pada Gambar 2.2



Gambar 2.2 Simbol *Data Flow*

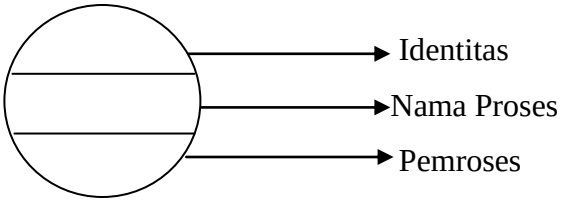
Arus data menunjukkan data yang mengalir diantara proses, simpanan data dan kesatuan luar, yang dapat berupa masukan untuk sistem maupun hasil dari proses sistem.

Arus data dapat berbentuk :

1) Formulir atau dokumen yang digunakan di perusahaan
2) Laporan yang terletak yang dihasilkan oleh sistem
3) Tampilan atau *output* dilayar komputer yang dihasilkan oleh sistem

c) *Process* (Proses)

Simbol proses tertera pada Gambar 2.3



Gambar 2.3 Simbol Proses

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil yang masuk ke dalam proses untuk dihasilkan arus data yang keluar dari proses.

Penjelasan dari simbol di atas adalah sebagai berikut :

1) Identitas proses

   Berupa suatu angka yang menunjukkan nomor urutan atau acuan dari proses dan ditulis pada bagian atas simbol proses.

2) Nama proses

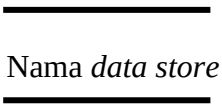
   Menunjukkan kegiatan yang dikerjakan oleh proses tersebut. Nama dari proses harus jelas dan lengkap menggambarkan kegiatan/proses yang sedang terjadi.

3) Pemroses

   Menunjukkan *user* yang melakukan proses dari suatu kegiatan.

d) *Data store* (Simpanan Data)

Simbol *data store* tertera pada Gambar 2.4



Gambar 2.4 Simbol Simpanan Data

Simpanan data merupakan simpanan dari data yang dapat berupa :

- Suatu *file* atau *data base* di sebuah sistem komputer
- Suatu arsip atau catatan manual

Setelah mengetahui uraian dari simbol-simbol yang digunakan dalam pembuatan DFD yang selanjutnya adalah tahapan pembuatan diagram DFD. Diagram yang dimaksud adalah sebagai berikut :

1) Diagram konteks

   Merupakan diagram yang menggambarkan suatu sistem beserta seluruh *external entity* yang terlibat dalam sistem, disertai dengan data yang digunakan atau diperlukan (hanya terlihat satu proses).

2) Diagram *level* 0

   Merupakan diagram yang menggambarkan seluruh proses yang ada pada diagram konteks, seluruh *external entity* dan seluruh arus data yang ada dalam sistem (terlihat lebih dari satu proses).

3) Diagram *level* 1

   Merupakan diagram yang menunjukkan penjelasan masing-masing proses yang terdapat pada diagram *level* 0.

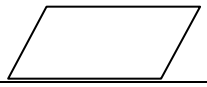
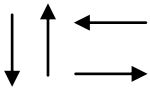
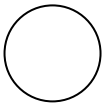
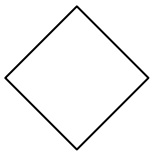
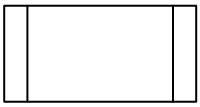
2.1.6.3 Kamus Data

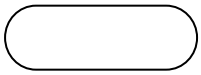
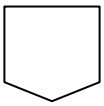
Kamus data adalah referensi yang digunakan oleh seorang analis sistem dalam merancang *database*, sehingga metode-metode yang digunakan nantinya tidak menyulitkan seorang *programmer* dalam membuat *coding* program serta tidak menyulitkan pelaku sistem yang mengikuti prosedur pemrograman yang dibuat oleh analis sistem (Febrian dan Andayani, 2008).

a. *Flowchart* Program (Bagan alir Program)

Bagan alir program adalah bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program digunakan untuk menggambarkan logika program yang dirancang. Simbol - simbol yang digunakan dalam ***Flowchart Program*** ditampilkan pada Tabel 2.2 :

Tabel 2.2 Diagram Alir Program

No	Simbol	Keterangan
1		***Input* atau *Output*** Untuk mewakili data *input* atau *output*.
2		**Proses** Menunjukkan proses yang sedang terjadi.
3		**Garis Alir** Untuk menunjukkan alur dari sebuah proses yang sedang terjadi.
4		**Penghubungan** Untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang masih sama.
5		**Keputusan** Untuk menyelesaikan kondisi dalam program.
6		**Prosedur** Untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ke tempat lain.

7		**Persiapan** Untuk memberikan nilai awal untuk suatu besaran.
8		**Titik Terminal** Untuk menunjukkan awal dan akhir dari suatu proses.
9		**Penghubungan** Untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang berbeda atau dihalaman lainnya.

2.1.7 Rekayasa Perangkat Lunak

Perangkat lunak adalah perintah yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan, struktur data yang memungkinkan program memanipulasi informasi secara proporsional, dan dokumen yang menggambarkan operasi dan kegunaan program. (Pressman, 2005).

Sedangkan rekayasa piranti lunak adalah pembuatan dan penggunaan prinsip-prinsip teknik yang bertujuan untuk mendapatkan *software* yang bernilai ekonomis yang reliabel dan bekerja secara efisien pada perangkat nyata (Pressman, 2005). Sedangkan menurut IEEE (IEE, 2003) telah mengembangkan definisi yang lebih komprehensif, yaitu:

1. Rekayasa perangkat lunak adalah aplikasi dari sebuah pendekatan kuantifiable, disiplin, dan sistematis kepada pengembang, operasi dan pemeliharaan perangkat lunak.
2. Studi tentang pendekatan-pendekatan seperti pada proses, metode dan alat bantu.

2.1.8 Alat Pengembangan Berbasis *Web*

2.1.8.1 *Web*

WWW atau yang sering disebut *world wide web* (jaringan dunia luas) adalah sebuah bagian dari internet yang sangat dikenal dalam dunia internet, dengan adanya WWW seorang pengguna dapat menampilkan sebuah halaman *virtual* yang disebut dengan *website* (Nugroho, 2004).

Halaman *web* tersusun atas satu atau beberapa komponen baik dokumen *web* itu sendiri ataupun komponen penyerta lain. Komponen-komponen tersebut biasanya dapat langsung ditampilkan oleh *browser (client),* tetapi terkadang harus menggunakan program pembantu atau aplikasi tambahan lain *(plug-in).*

Web browser adalah perangkat lunak *(software)* disisi klien yang digunakan untuk mengakses informasi *web*, memformat teks dan menempatkan grafik pada *layer*. Ada beberapa jenis *web browser* yang dipakai sebagai sumber tampilan antara lain *: Internet Explorer, Netscape Navigator, Mozila, Opera,* dsb.

Prinsip kerja pengaksesan sebuah halaman *web* yang berbasisi HTML adalah sebagai berikut :

a. *Browser* meminta sebuah halaman kesuatu situs *web* melalui protokol http.
b. *Web server* meminta permintaan.
c. *Web server* segera mengirimkan dokumen HTML yang diminta ke klien.
d. *Browser* pada klien membaca dan mengartikan dokumen yang diterima berdasarkan kode-kode pemformatan yang terdapat pada dokumen HTML, lalu menampilkan dalam versinya masing-masing.

Web server adalah *server* yang melayani permintaan klien terdapat halaman *web* seperti *apache,* IIS *(Internet Information Sistem)* dan berkomunikasi dengan

Middleware untuk menterjemahkan kode-kode tertentu, menjalankan kode-kode tersebut dan memungkinkan berinteraksi dengan basis data, PHP atau ASP. Adapun arsitektur aplikasi *server* adalah sebagai berikut :

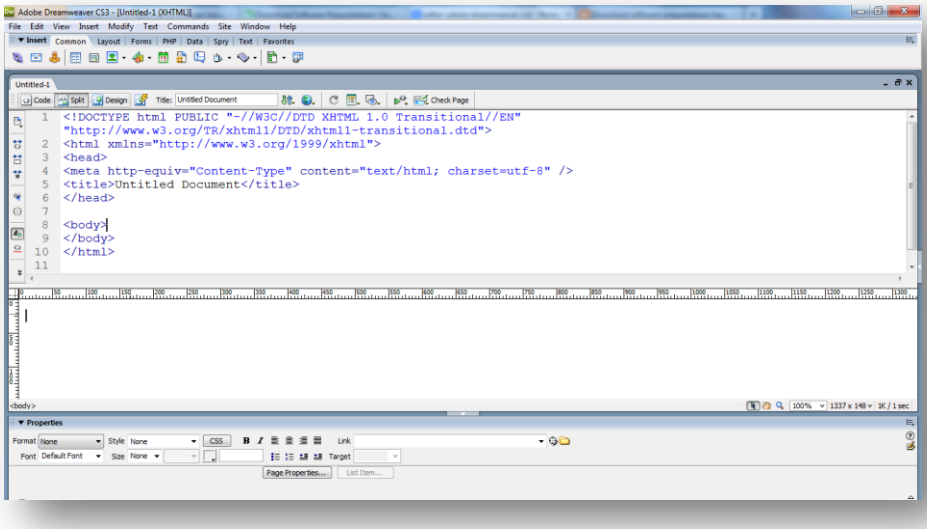
a. *Browser* atau klien berinteraksi dengan *web server*.
b. Secara internal *web server* berinteraksi dengan *middleware*.
c. *Middleware* yang berhubungan dengan *database*.

Teknologi yang berjalan di *server* antara lain : CGI *(Common Gateway Interface)*, ASP *(Active Server Page)*, JSP *(Java Server Page)*, PHP (Arif, Ayu Putri Wanda, dkk, 2013).

2.1.8.2 Adobe Dreamweaver

Adobe Dreamweaver merupakan sebuah *software web design* yang menawarkan cara mendesain *website* dengan dua langkah sekaligus dalam satu waktu, yaitu mendesain dan melakukan pemrograman. Adobe Dreamweaver memiliki jendela mini yang disebut HTML *source* tempat kode-kode HTML ditulis. Setiap kali *user* mendesain *website* seperti menulis kata-kata, meletakkan gambar, membuat tabel, dan proses lainnya. *Tag-tag* HTML ditulis secara langsung mengiringi proses pengaturan *website*. Artinya *user* memiliki kesempatan untuk mendesain *website* sekaligus mengenal *tag-tag* HTML yang membangun *website*, selain itu dapat mendesain *website* hanya lewat menuliskan *tag-tag* dan *text* lain di jendela HTML *source*.

Tampilan lembar kerja Adobe Dreamwaver ditunjukkan pada Gambar 2.5.



Gambar 2.5 Tampilan lembar kerja Adobe Dreamweaver

2.1.8.3 MySQL

MySQL merupakan *Relational Database Management Sistem* (RDMS) yang dikembangkan oleh sebuah perusahaan pengembangan perangkat lunak dan konsultan *database* bernama MySQL AB yang bertempat di Swedia. MySQL didistribusikan secara gratis di bawah lisensi GPL (*General Public License*), yang setiap orang bebas menggunakan MySQL. MySQL menggunakan suatu standar *SQL* (*Structures Query Language*), yaitu bahasa standar yang paling banyak digunakan untuk mengakses *database*.

2.1.8.4 Apache

Apache adalah sebuah HTTP *server*. Apache dibangun pertama kalinya berdasarkan pada kode-kode dan ide-ide yang terdapat pada HTTP *server* yakni NCSA httpd 1.3, pada awal 1995 (Purnamasari, 2010).

Pada saat ini Apache termasuk *web server* yang paling banyak digunakan, hal ini bukan dikarenakan Apache didapat secara gratis. Namun juga karena kemampuannya, secara fungsional tingkat efisien dan kecepatannya yang dapat dibandingkan dengan *server* HTTP UNIX lainnnya. Selain didapatkan secara gratis dan *open source*, pengembangan Apache juga dilakukan oleh banyak *programmer* secara sederhana (Purnamasari, 2010).

2.1.8.5 *Prototype*

"Prototype" adalah implementasi bagian dari produk *software* yang secara tipikal fungsinya dibatasi, reliabilitas rendah, tampilannya miskin, dan kurang ketegasan. *Prototype* sering dikembangkan secara cepat dalam bahasa tingkat tinggi atau dalam bahasa *prototype* tertentu, tanpa memperhatikan kebenaran dan ketegapan dan sebagainya.

Prototype memberikan ide bagi pembuat maupun pemakai potensial tentang cara sistem berfungsi dalam bentuk lengkapnya. Proses menghasilkan sebuah *prototype* disebut *prototyping*.

2.1.8.5.1 Jenis – Jenis *Prototype*

a. *Prototype* Jenis I

Prototype jenis I sesungguhnya menjadi sistem operasional.

Tahapan pengembangan *prototype* jenis I :

1. Mengidentifikasi kebutuhan pemakai.

   Analis sistem mewawancarai pemakai untuk mendapatkan gagasan dan apa yang diinginkan pemakai terhadap sistem.

2. Mengembangkan *prototype*.

   Analis sistem mungkin bekerjasama dengan spesialis informasi lain, menggunakan satu atau lebih peralatan *prototyping* untuk mengembangkan sebuah *prototype*. Contoh dari peralatan *prototyping* adalah *integrated application generator* dan *prototyping toolkits. Integrated application generator* adalah siste perangkat lunak jadi yang mampu menghasilkan semua tampilan yang didinginkan dalam sistem menu, laporan, layar dan , *databse*. *Prototyping toolkits* mencakup sistem-sistem perangkat lunak terpisah, yang masing-masing mampu menghasilkan sistem yang didinginkan.

3. Menentukan apakah *prototype* dapat diterima.

   Analis mendidik pemakai untuk menggunakan *protoype* dan memberikan kesempatan pada mereka untuk membiasakan diri dengan sistem. Pemakai member masukan kepada analisis apakah *prototype* memuaskan. Jika Ya, langkah 4 diambil; jika Tidak, *prototype* direvisi dengan mengulangi langkah 1, 2, dan 3 dengan pengertian yang lebih baik mengenai kebutuhan pemakai.

4. Menggunakan *prototype*.

   Prototype ini menjadi sistem operasional. Pendekatan ini hanya mungkin jika peralatan prototyping memungkinkan *prototype* memuat semua elemen penting dari sistem baru.

b. *Prototype* Jenis II

Prototype jenis II merupakan suatu model yang dapat dibuang yang berfungsi sebagai cetak biru bagi sistem operasional.

Tahapan pengembangan *prototype* jenis II :

1. Mengidentifikasikan kebutuhan pemakai.

   Analis sistem mewawancarai pemakai untuk mendapatkan gagasan dari apa yang diinginkan pemakai terhadap sistem.

2. Mengembangkan *prototype*

   Analis sistem mungkin bekerjasama dengan spesialis informasi lain, menggunakan satu atau lebih peralatan *prototyping* untuk mengembangkan sebuah *prototype*. Contoh dari peralatan *prototyping* adalah *integrated application generator* dan *prototyping toolkits. Integrated application generator* adalah siste perangkat lunak jadi yang mampu menghasilkan semua tampilan yang didinginkan dalam sistem menu, laporan, layar, *databse*, dan sebagainya. *Prototyping toolkits* mencakup sistem-sistem perangkat lunak terpisah, yang masing-masing mampu menghasilkan sistem yang didinginkan.
3. Menentukan apakah *prototype* dapat diterima.

   Analis mendidik pemakai untuk menggunakan *protoype* dan memberikan kesempatan pada mereka untuk membiasakan diri dengan sistem. Pemakai member masukan kepada analisis apakah *prototype* memuaskan. Jika Ya, langkah 4 yang diambil; jika Tidak, *prototype* direvisi dengan mengulangi langkah 1, 2, dan 3 dengan pengertian yang lebih baik mengenai kebutuhan pemakai.
4. Mengkodekan sistem operasinal.

   Programmer menggunakan *prototype* sebagai dasar untuk pengkodean (*coding*) sistem operasional.
5. Menguji sistem operasional.

   Programmer yang menguji sistem operasional tersebut. *Programmer* juga yang menentukan jika sistem operasional dapat diterima. Pemakai memberi masukkan kepada analis paakah sistem dapat diterima. Jika iya, langkah 7 (tujuh) dilakukan, jika tidak, langkah 4 dan 5 diulangi.
6. Menggunakan sistem operasional.

2.1.8.5.2 Keunggulan dan Kelemahan *Prototype*

a. Keunggulan Dalam Pembuatan *Prototype*

Keunggulan dalam pembuatan *prototype* bagi *engineer* sebagai berikut :

1. Kesalahpahaman antara sistem *developer* dan sistem *user* dapat diidentifikasi dan dapat dibetulkan. Ciri-ciri yang membuat bingung *user* dapat diidentifikasi dan dibetulkan. Kehilangan atau kesalahan persyaratan dapat diidentifikasi.
2. *Prototype* yang sedang bekerja mungkin sangat berguna dalam suatu pembuktian menejemen dimana suatu proyek adalah fleksibel sehingga menjamin kelangsungan dukungan.

b. Kelemahan Dalam Pembuatan *Prototype*

Kelemahan dalam pembuatan *prototype* bagi *engineer* sebagai berikut :

1. Ketergesaan untuk membuat *prototype* mungkin menghasilkan jalan pintas dalam definisi permasalahan, evaluasi alternatif, dan dokumentasi. Istilah “cepat dan kotor” digunakan untuk menjelaskan beberapa usaha *prototyping*.
2. Pemakai sangat tertarik dengan *prototype* tersebut sehingga mereka mengharapkan sesuatu yang tidak realistis dari sistem operasional itu.
3. *Prototype* jenis I mungkin tidak seefisien sistem yang dikodean dalam bahasa pemrograman.
4. Hubungan komputer-manusia yang disediakan oleh peralatan *prototyping* tertentu mungkin tidak mencerminkan teknik perancangan yang baik.

   (Al Bahra, 2006).

2.1.8.6 XAMPP

XAMPP adalah salah satu paket instalasi Apache, PHP, dan MySQL secara instant yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut sama seperti PHPTriad . Selain paket instalasi instant, XAMPP juga memberikan fasilitas pilihan penggunaan PHP 4 atau PHP 5. Untuk melakukan migrasi ke versi lebih tinggi juga sangat mudah dilakukan dengan bantuan *PHP-Switch* yang telah disertakan oleh XAMPP. Sama halnya dengan PHP, XAMPP bersifat *free* atau gratis untuk digunakan. Selain sebagai PHP *Server*, XAMPP juga memiliki mod perl yang berfungsi sebagai ASP *Server*, sehingga XAMPP dapat digunakan dengan bahasa pemrograman ASP (Komputer, 2009).

XAMPP adalah suatu langkah mudah untuk menginstal distribusi Apache yang memiliki fitur program MySQL, PHP, dan Perl. XAMPP juga sangat mudah digunakan. XAMPP cukup diunduh, diekstrak dan digunakan. XAMPP berdasar pada lisensi GNU (General Public Licence), sehingga instalasi XAMPP tidak memerlukan perubahan pada *register Windows* dan tidak perlu mengubah *file* konfigurasinya. Terdapat program *default* XAMPP yang ditulis dalam bahasa PHP menggunakan MySQL, dan aplikasi buku tamu yang ditulis dalam bahasa Perl, serta beberapa *utility* demo program lainnya (Komputer, 2009).