

BAB II

LANDASAN TEORI

2.1 Pengertian Imunisasi dan Universal Child Immunization (UCI)

Imunisasi adalah suatu cara untuk meningkatkan kekebalan seseorang secara aktif terhadap suatu penyakit, sehingga bila kelak ia terpapar dengan penyakit tersebut tidak akan menderita penyakit tersebut.

Universal Child Immunization (UCI) adalah suatu keadaan tercapainya imunisasi dasar secara lengkap pada Semua Bayi. Bayi adalah anak dibawah umur 1 tahun (Kepmenkes No. 1611/MENKES/SK/XI/2005: 9).

2.1.1 Indikator UCI

Indikator dalam proses perhitungan UCI sebagai berikut: (1) DPT-1 : jangkauan/aksesibilitas pelayanan, (2) hepatitis B1 < 7 hari : jangkauan/aksesibilitas pelayanan, (3) campak : tingkat perlindungan (efektivitas program), (4) polio-4 : tingkat perlindungan (efektivitas program), dan (5) drop out DPT-1 – campak : efisiensi/manajemen program. Pemberian imunisasi pada bayi dapat dilihat pada tabel 2.1.

Tabel 2.1 Jadwal Pemberian Imunisasi Pada bayi

Vaksin	Pemberian Imunisasi	Selang waktu pemberian (minimal)	Umur (Bulan)	Keterangan
BCG	1x	-	0-11	
DPT	3x (1,2,& 3)	4 Minggu	2-11	
Polio	4x (1,2,3, & 4)	4 Minggu	0-11	
CAMPAK	1x	-	9-11	
HB	3x (1,2,& 3)	4 Minggu	0-11	Untuk bayi lahir di RS/Pusk/RB/Rumah

Vaksin	Pemberian Imunisasi	Selang waktu pemberian (minimal)	Umur (Bulan)	Keterangan
				oleh Nakes Pelaksanaan HB segera diberikan dalam 24jam pertama kelahiran, vaksin BCG, Polio, diberikan sebelum bayi pulang kerumah

Tabel 2.2 Jadwal Pemberian Imunisasi Vaksin DPT & HB

Umur	Vaksin	Tempat
Bayi Lahir di Rumah;		
0 Bulan	HB1	Rumah
1 Bulan	BCG, Polio 1	Posyandu*
2 Bulan	DPT1, HB2, Polio2	Posyandu*
3 Bulan	DPT2,HB3,Polio3	Posyandu*
4 Bulan	DPT3,Polio4	Posyandu*
9 Bulan	Campak	Posyandu*
Bayi Lahir di RS/RB/Bidan Praktek;		
0 Bulan	HB1,Polio1, BCG	RS/RB/Bidan
2 Bulan	DPT1, HB2, Polio2	RS/RB/Bidan#
3 Bulan	DPT2,HB3,Polio3	RS/RB/Bidan#
4 Bulan	DPT3,Polio4	RS/RB/Bidan#
9 Bulan	Campak	RS/RB/Bidan#

Tabel 2.3 Jadwal Pemberian Imunisasi Pada Bayi Vaksin DPT/HB Kombo

Umur	Vaksin	Tempat
Bayi Lahir di Rumah;		
0 Bulan	HB1	Rumah
1 Bulan	BCG, Polio 1	Posyandu*
2 Bulan	DPT1, HB kombo 1, Polio2	Posyandu*
3 Bulan	DPT2,HB kombo 2,Polio3	Posyandu*

Umur	Vaksin	Tempat
4 Bulan	DPT3,HB kombo 3,Polio4	Posyandu*
9 Bulan	Campak	Posyandu*
Bayi Lahir di RS/RB/Bidan Praktek;		
0 Bulan	HB1,Polio1, BCG	RS/RB/Bidan
2 Bulan	DPT1,HB kombo 1, Polio2	RS/RB/Bidan#
3 Bulan	DPT2,HB kombo 2,Polio3	RS/RB/Bidan#
4 Bulan	DPT3, HB kombo 3,Polio4	RS/RB/Bidan#
9 Bulan	Campak	RS/RB/Bidan#

Keterangan :

- * : Tempat Pelayanan lain
- # : Posyandu

Berdasarkan Kepmenkes (No.1611/MENKES/SK/XI/2005:20-21) untuk pemberian vaksin imunisasi setiap bayi.

2.1.2 Cakupan Desa/Kelurahan UCI

Pada sub bab ini akan dijelaskan tentang pengertian, definisi operasional, dan cara perhitungan rumus dari cakupan desa atau kelurahan UCI.

A. Pengertian

1. Bayi adalah anak berumur 29hari -11 bulan
2. Cakupan kunjungan bayi adalah Cakupan kunjungan bayi umur 29 hari – 11 bulan di sarana pelayanan kesehatan (Polindes, Pustu, Puskesmas, Rumah Bersalin dan Rumah Sakit) maupun di rumah, Posyandu, tempat penitipan anak, panti asuhan dan sebagainya melalui kunjungan petugas.
3. Setiap bayi memperoleh pelayanan kesehatan minimal 4 kali yaitu satu kali pada umur 29 hari-3 bulan, 1 kali pada umur 3-6 bulan, 1 kali pada umur 6-9 bulan, dan 1 kali pada umur 9-11 bulan.

4. Pelayanan Kesehatan tersebut meliputi pemberian imunisasi dasar (BCG, DPT/ HB1-3, Polio 1-4, Campak), stimulasi deteksi intervensi dini tumbuh kembang (SDIDTK) bayi dan penyuluhan perawatan kesehatan bayi
5. Penyuluhan perawatan kesehatan bayi meliputi: konseling ASI eksklusif, pemberian makanan pendamping ASI sejak usia 6 bulan, perawatan dan tanda bahaya bayi sakit (sesuai MTBS), pemantauan pertumbuhan dan pemberian vitamin A kapsul biru pada usia 6 – 11 bulan.
6. Indikator ini mengukur kemampuan manajemen program KIA dalam melindungi bayi sehingga kesehatannya terjamin melalui penyediaan pelayanan kesehatan.

B. Definisi Operasional

Cakupan kunjungan bayi adalah cakupan bayi yang memperoleh pelayanan kesehatan sesuai dengan standar oleh dokter, bidan, dan perawat yang memiliki kompetensi klinis kesehatan, paling sedikit 4 kali disatu wilayah kerja pada kurun waktu tertentu.

C. Cara Perhitungan Rumus

1. Rumus

Berikut adalah perhitngan dari rumus cakupan kunjungan bayi pada satu wilayah.

$$\text{Cakupan Kunjungan Bayi} = \frac{\text{Jumlah bayi memperoleh pelayanan kesehatan sesuai standar disatu wilayah kerja pada kurun waktu tertentu}}{\text{Jumlah seluruh bayi lahir hidup disatu wilayah kerja dalam kurun waktu yang sama}} \times 100\%$$

2. Pembilang

Jumlah bayi yang memperoleh pelayanan kesehatan sesuai dengan standar, paling sedikit 4 kali di satu wilayah kerja pada kurun waktu tertentu.

3. Penyebut

Seluruh bayi lahir hidup di satu wilayah kerja dalam kurun waktu sama.

Catatan:

Jika tidak ada data dapat digunakan angka estimasi jumlah bayi lahir hidup berdasarkan data BPS atau perhitungan CBR dikalikan jumlah penduduk.

4. Ukuran/Konstanta

Persentase (%)

2.2 Monitoring dan Evaluasi Program

Menurut Surat Keputusan Menteri Kesehatan Republik Indonesia Nomor (Kepmenkes No.1611/MENKES/SK/XI/2005: 42-43). *Monitoring* dan evaluasi merupakan salah satu fungsi manajemen untuk menilai keberhasilan pelaksanaan program. *Monitoring* dilaksanakan secara berkala dan terus menerus, untuk dapat segera mendeteksi bila ada masalah dalam pelaksanaan kegiatan yang telah direncanakan, supaya dapat dilakukan tindakan perbaikan segera. Evaluasi dilakukan setelah suatu jarak-waktu (interval) lebih lama, biasanya setiap 6 bulan s/d 1 tahun. Dengan evaluasi dapat dinilai sejauh mana tujuan dan target yang telah ditetapkan sebelumnya dicapai. Dalam mengukur keberhasilan tersebut

diperlukan indikator. Hasil evaluasi sangat berguna untuk kepentingan perencanaan dan pengembangan program.

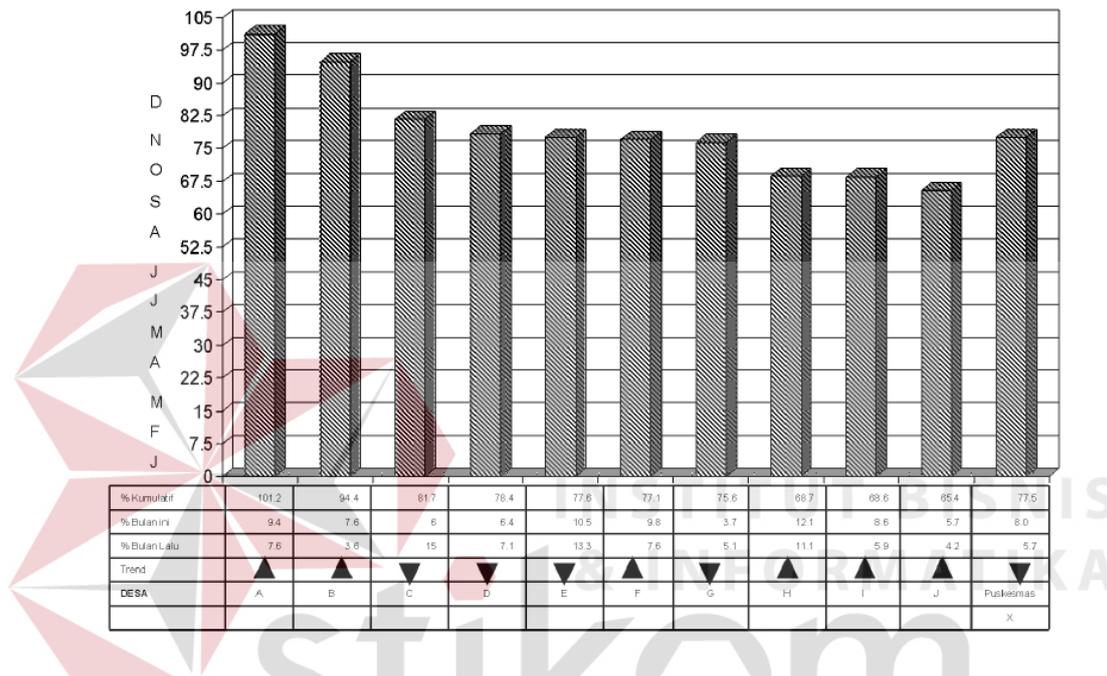
Salah satu fungsi penting dalam manajemen program adalah pemantauan. Dengan pemantauan kita dapat menjaga agar masing-masing kegiatan sejalan dengan ketentuan program. Pemantauan yang dimiliki oleh program imunisasi merupakan:

2.2.1 Pemantauan Wilayah Setempat (PWS)

Alat pemantauan ini berfungsi untuk meningkatkan cakupan, jadi sifatnya lebih memantau kuantitas program. Dipakai pertama kalinya di Indonesia pada tahun 1985 dan dikenal dengan nama *Local Area Monitoring (LAM)*. LAM terbukti efektif kemudian diakui oleh WHO untuk diperkenalkan dinegara lain. Grafik LAM kemudian disempurnakan menjadi yang di kenal sekarang dengan Pemantauan Wilayah Setempat (PWS). Prinsip PWS:

1. Memanfaatkan data yang ada: dari cakupan/laporan cakupan imunisasi
2. Menggunakan indikator sederhana: tidak terlalu banyak :
 - a. DPT-1 : Jangkauan/aksesibilitas pelayanan
 - b. Hepatitis B1 < 7 hari : Jangkauan/aksesibilitas pelayanan
 - c. Campak : Tingkat perlindungan (efektivitas program)
 - d. Polio-4 :Tingkat perlindungan (efektivitas program)
 - e. Drop Out DPT-1 – Campak : Efisiensi/manajemen program
3. Dimanfaatkan untuk pengambilan keputusan setempat
4. Teratur dan tepat waktu: Setiap bulan
 - a. Teratur untuk menghindari hilangnya informasi penting
 - b. Tepat waktu agar tidak terlambat dalam mengambil keputusan.

5. Lebih dimanfaatkan sendiri atau sebagai umpan balik untuk dapat mengambil tindakan daripada hanya dikirimkan sebagai laporan.
6. Membuat grafik yang jelas dan menarik untuk masing-masing indikator diatas, untuk memudahkan analisis.



Gambar 2.1 Contoh Grafik PWS DPT-1 Puskesmas X tahun 2003

2.2.2 Pengawasan

Pengawasan dapat didefinisikan sebagai proses untuk “menjamin” bahwa tujuan organisasi dan manajemen tercapai. Ini berkenaan dengan cara-cara membuat kegiatan yang sesuai perencanaan. Pengertian ini menunjukkan adanya hubungan sangat erat antara perencanaan dan pengawasan. Seperti terlihat dalam kenyataan, langkah awal proses pengawasan adalah sebenarnya langkah perencanaan, penetapan tujuan, standar atau sasaran pelaksanaan antara kegiatan. Karena sulit untuk membedakan antara rencana, standar atau apa itu pengawasan.

Definisi pengawasan yang dikemukakan oleh Robert J. Mockler berikut ini telah memperjelas unsur-unsur esensial proses pengawasan:

Pengawasan manajemen adalah suatu usaha sistematis untuk menetapkan standar pelaksanaan dengan tujuan-tujuan perencanaan, merancang sistem informasi umpan balik, membandingkan kegiatan nyata dengan standar yang telah ditetapkan sebelumnya, menentukan dan mengukur penyimpangan-penyimpangan, serta mengambil tindakan koreksi yang diperlukan untuk menjamin bahwa semua sumber daya perusahaan dipergunakan dengan cara paling efektif dan efisien dalam pencapaian tujuan-tujuan perusahaan. (Handoko, 2000: 359-361).

2.2.3 Monitoring dan Evaluasi

Monitoring dan Evaluasi adalah dua kata yang memiliki aspek kegiatan yang berbeda yaitu kata *monitoring* dan evaluasi. *Monitoring* merupakan kegiatan untuk mengetahui apakah program yang dibuat itu berjalan dengan baik sebagaimana mestinya sesuai dengan yang direncanakan, adakah hambatan yang terjadi dan bagaimana para pelaksana program itu mengatasi hambatan tersebut. *monitoring* terhadap sebuah hasil perencanaan yang sedang berlangsung menjadi alat pengendalian yang baik dalam seluruh proses implementasi.

“*Monitoring* lebih menekankan pada pemantauan proses pelaksanaan” (Departemen Pendidikan Nasional: 2001). *Monitoring* juga lebih ditekankan untuk tujuan supervisi.

Proses dasar dalam *monitoring* ini meliputi tiga tahap yaitu: (1) menetapkan standar pelaksanaan; (2) pengukuran pelaksanaan; (3) menentukan

kesenjangan (deviasi) antara pelaksanaan dengan standar dan rencana. Menurut Dunn (1981), monitoring mempunyai empat fungsi, yaitu:

- a. Ketaatan (*compliance*). *Monitoring* menentukan apakah tindakan administrator, staf, dan semua yang terlibat mengikuti standar dan prosedur yang telah ditetapkan.
- b. Pemeriksaan (*auditing*). *Monitoring* menetapkan apakah sumber dan layanan yang diperuntukkan bagi pihak tertentu bagi pihak tertentu (target) telah mencapai mereka.
- c. Laporan (*accounting*). *Monitoring* menghasilkan informasi yang membantu “menghitung” hasil perubahan sosial dan masyarakat sebagai akibat implementasi kebijaksanaan sesudah periode waktu tertentu.
- d. Penjelasan (*explanation*). *Monitoring* menghasilkan informasi yang membantu menjelaskan bagaimana akibat kebijaksanaan dan mengapa perencanaan dan pelaksanaannya tidak cocok.

Penilaian (Evaluasi) merupakan tahapan yang berkaitan erat dengan kegiatan *monitoring*, karena kegiatan evaluasi dapat menggunakan data yang disediakan melalui kegiatan *monitoring*. Dalam merencanakan suatu kegiatan hendaknya evaluasi merupakan bagian yang tidak terpisahkan, sehingga dapat dikatakan sebagai kegiatan yang lengkap. Evaluasi diarahkan untuk mengendalikan dan mengontrol ketercapaian tujuan. Evaluasi berhubungan dengan hasil informasi tentang nilai serta memberikan gambaran tentang manfaat suatu kebijakan. Istilah evaluasi ini berdekatan dengan penafsiran, pemberian angka dan penilaian. Evaluasi dapat menjawab pertanyaan “Apa perbedaan yang dibuat”. (William N Dunn, 2000).

Evaluasi bertujuan untuk mengetahui apakah program itu mencapai sasaran yang diharapkan atau tidak, evaluasi lebih menekankan pada aspek hasil yang dicapai (*output*). Evaluasi baru bisa dilakukan jika program itu telah berjalan dalam suatu periode, sesuai dengan tahapan rancangan dan jenis program yang dibuat dan dilaksanakan, misalnya disekolah, untuk satu caturwulan atau enam bulan atau satu tahun pelajaran.

2.3 Sistem Informasi

Sistem Informasi adalah kumpulan dari komponen yang saling terkait yang berjalan bersamaan secara kolektif untuk menjalankan input, proses, output penyimpanan dan pengendalian tidakan yang bertujuan untuk mengubah data kedalam sebuah informasi, sehingga informasi tersebut dapat digunakan untuk membantu dalam proses peramalan, perencanaan, pengendalian, koordinasi, pembuatan keputusan, dan kegiatan operasional di dalam organisasi. (Bocij, 2008:36).

2.3.1 Komponen Sistem

Pada point ini dapat dikemukakan bahwa istilah umum dari sistem terdiri dari lima komponen:

1. *Input* sistem dapat diartikan sebagai masukan dari proses yang akan menghasilkan sebuah output.
2. Proses adalah perubahan input menjadi sebuah output.
3. *Output* adalah hasil yang diciptakan oleh sistem.
4. *Feedback mechanism* adalah ketersediaan informasi pada kinerja sistem yang dapat digunakan untuk mengatur perilaku dari sistem tersebut.
5. Jika perubahan dibutuhkan oleh sistem, maka *control mechanism* akan melakukan pengaturan yang sesuai. (Bocij, 2008: 37).

2.3.2 Karakteristik Sistem

Berikut akan dijelaskan beberapa dari karakteristik sistem:

1. *The component of System work towards a collective goal* atau diketahui sebagai tujuan dari sistem. Tujuan sistem secara normal sangat spesifik dan sering ditunjukkan dalam satu kalimat.
2. *System do not operate in complete isolation* maksud dari hal tersebut adalah sistem hanya terdiri dari lingkungan sistem tersebut yang di dalamnya terdiri dari beberapa sistem lain dan pihak diluar sistem. Ruang lingkup sistem sendiri didefinisikan oleh batasan dari sistem itu sendiri. Apapun yang ada di luar batas merupakan bagian dari lingkungan sistem, apapun bentuk batasannya merupakan bagian dari sistem itu sendiri. Batasan juga menandakan tampilan antara sistem dan lingkungannya. Tampilan juga menggambarkan pertukaran antara sistem dan sistem yang lain.
3. *System can be complex and can be made up of other, smaller system.* Atau dikenal dengan subsistem. Sistem tersusun dari satu atau beberapa subsistem yang disebut dengan supra sistem. Tujuan dari subsistem adalah untuk mendukung tujuan suprasistem yang lebih luas. Sistem yang melakukan interaksi dengan lingkungannya disebut dengan sistem terbuka. Sistem terbuka ini mempengaruhi perubahan dari lingkungan sistem tersebut. Kebanyakan sistem informasi adalah sistem terbuka karena sistem informasi menerima masukan dan memberikan reaksi. Hal yang tidak biasa adalah secara keseluruhan sistem tertutup tidak melakukan interaksi dengan lingkungannya.

4. *Subsystem in a information system interact by exchanging information.* Atau yang dikenal dengan tampilan antar sistem. Untuk sistem informasi dan sistem bisnis mempunyai tampilan yang jelas sehingga penting untuk efisiensi organisasi.
5. *The linkage or coupling between subsystem varies.* Rangkaian ini menggambarkan betapa erat keterkaitan antar subsistem. Hal ini merupakan prinsip fundamental dari teori sistem dan desain sistem informasi bisnis yang subsistemnya seharusnya serangkaian.

Sistem atau subsistem yang tergantung pada sistem yang lain dikenal dengan rangkaian sistem terkait. Dalam beberapa kasus, keluaran dari satu sistem merupakan masukan dari sistem yang lain. Metode *just in time* yang digunakan oleh beberapa perusahaan manufaktur juga merupakan ilustrasi dari rangkaian sistem terkait.

Sistem yang tidak terkait merupakan sistem yang tingkat ketergantungannya dengan sistem yang lain lebih sedikit dan lebih mampu beradaptasi dengan situasi yang tidak diinginkan. Beberapa sistem cenderung mempunyai level *autonomi* yang lebih tinggi, dan lebih memberikan kebebasan untuk merencanakan dan mengendalikan aktivitasnya. Walaupun sistem yang tidak serangkaian lebih fleksibel dan *adaptive* daripada sistem yang serangkaian, akan tetapi kebebasan ini meningkatkan kemungkinan terjadinya ketidak efisienan dari sistem tersebut.

6. *System are hierarchical.* Hal ini seharusnya membuat kita menyadari bahwa bagian dari setiap sistem saling terkait. (Bocij, 2008: 37-38).

2.3.3 Membuat Informasi

Proses data diperlukan untuk menempatkan data-data tersebut ke dalam konteks yang berarti sehingga data-data tersebut dapat lebih mudah untuk dimengerti. Terdapat beberapa proses data yang berbeda yang dapat digunakan untuk melakukan transformasi data ke dalam informasi. Berikut adalah beberapa contoh dari data proses:

1. *Classification*. Hal ini terdiri dari penempatan data ke dalam beberapa kategori.
2. *Rearranging/sorting*. Hal ini terdiri dari pengumpulan data sehingga *item* dari data tersebut dapat dikelompokkan menjadi beberapa bagian.
3. *Aggregating*. Hal ini terdiri dari ringkasan data.
4. *Performing Calculation*. Contoh dari proses data ini adalah perhitungan gaji karyawan yang dihitung berdasarkan waktu mereka bekerja.
5. *Selection*. Hal ini terdiri dari pemilihan dari data item yang berdasarkan pada kriteria pemilihan. (Bocij, 2008: 8-9).

2.3.4 Nilai Informasi

Nilai dari sebuah informasi terdiri dari dua hal yaitu:

1. *Tangible value* yang merupakan nilai atau keuntungan yang dapat diukur secara langsung. Pada dasarnya hal ini berupa nilai finansial.
2. *Intangible Value* yang merupakan nilai atau keuntungan yang sulit untuk diukur karena berhubungan dengan nilai kuantitas. (Bocij, 2008: 9).

2.3.5 Sumber Informasi

Informasi dapat dikumpulkan melalui dua hal yakni dengan cara formal *communication* dan *informal communication*,

1. *Formal communication* terdiri dari informasi yang terdiri atas struktur yang tetap sehingga komunikasi seperti ini sering dianggap kaku. Contoh dari komunikasi ini adalah laporan-laporan perusahaan yang sudah mempunyai format yang tetap.
2. *Informal communication* lebih menggambarkan informasi yang didapatkan dari percakapan sehari-hari. (Bocij, 2008:10).

2.3.6 Kualitas Informasi

Pada dasarnya informasi memang memiliki karakteristik yang berbeda untuk menggambarkan kualitasnya. Perbedaan tersebut terletak pada informasi tersebut merupakan informasi yang baik atau informasi yang buruk. Baik atau buruknya informasi tersebut dapat diidentifikasi sesuai dengan atribut dari kualitas informasi tersebut.

1. Dimensi Waktu
 - a. *Timeliness*. Informasi seharusnya tersedia pada saat dibutuhkan.
 - b. *Currency*. Informasi seharusnya mampu memberikan data yang terbaru.
 - c. *Frequency*. Informasi seharusnya tersedia pada waktu yang regular.
 - d. *Time Period*. Informasi seharusnya mampu meng-*cover* sesuai dengan periode waktunya.
2. Dimensi konten
 - a. *Accuracy*. Informasi yang kesalahannya hanya sebatas pada nilai organisasi.

- b. *Relevance*. Informasi yang *disupply* seharusnya relevant dengan situasi yang ada dan mampu memenuhi kebutuhan informasi pengguna.
- c. *Completeness*. Seluruh informasi harus memenuhi kebutuhan informasi pengguna dan seharusnya dapat melengkapi seluruhnya.
- d. *Conciseness*. Hanya informasi yang relevan yang dapat memenuhi kebutuhan informasi pengguna dan seharusnya tersedia dalam bentuk paling sederhana.
- e. *Scope*. Ruang lingkup dari informasi yang *disupply* seharusnya sesuai dengan informasi yang dibutuhkan oleh pengguna.

3. Dimensi form

- a. *Clarity*. Informasi seharusnya dapat mewakili dalam bentuk yang sesuai dengan tujuan pengguna.
- b. *Detail*. Informasi seharusnya berisi tentang level detil yang sesuai untuk memenuhi kebutuhan informasi pengguna.
- c. *Order*. Informasi seharusnya tersedia sesuai dengan permintaan pengguna.
- d. *Presentation*. Informasi seharusnya dapat mewakili dalam bentuk yang sesuai dengan tujuan pengguna.
- e. *Media*. Informasi seharusnya dapat terwakili melalui media yang tepat.

4. Karakteristik tambahan

Dari beberapa atribut yang telah digambarkan di atas, terdapat beberapa karakteristik tambahan antara lain:

- a. Bagian penting dari informasi adalah *confidence* dalam hal sumber dari informasi tersebut didapat. Pengguna akan lebih memilih menerima dan

memperceyai informasi yang mereka dapat melalui sumber yang akurat dan dipercaya sebelumnya.

- b. Atribut dari kualitas informasi adalah informasi tersebut dapat dipercaya. Informasi seharusnya dapat dikemukakan oleh pengguna tanpa keraguan sehingga informasi tersebut dapat diandalkan dan tersedia saat dibutuhkan dengan konsistensi dan akurasi yang dapat dipercaya.
- c. Informasi yang tersedia seharusnya sesuai dengan aktivitas pengguna.
- d. Informasi seharusnya diterima oleh orang yang memang membutuhkannya sehingga informasi tersebut bernilai.
- e. Informasi seharusnya dapat ditransmisikan melalui *channels* yang benar.

(Bocij, 2008: 12-14).

Untuk implementasi pada Dinkes akan digunakan *Web 1.0*, dikarenakan Dinkes hanya membutuhkan web yang dapat menghubungkan informasi dan berbagi informasi sehingga tidak mengijinkan user untuk dapat berinteraksi.

2.4 Dashboard

Dashboard adalah tampilan visual dari informasi terpenting yang dibutuhkan untuk mencapai satu atau beberapa tujuan. Pemberian poin penting diatur dalam satu tampilan sehingga informasi dapat di-*monitoring* dengan mudah (Few, 2006).

2.4.1 Tujuan Penggunaan *Dashboard*

Dashboard adalah tampilan visual dari informasi terpenting yang dibutuhkan untuk mencapai satu atau beberapa tujuan. Pemberian point penting

diatur dalam satu tampilan sehingga informasi dapat di-*monitoring* dengan mudah (Few, 2006). Tujuan dalam penggunaan *Dashboard*, yaitu:

a. Mengkomunikasikan Strategi

Mengkomunikasikan strategi dan tujuan yang dibuat oleh eksekutif, kepada semua pihak yang berkepentingan, sesuai dengan peran dan levelnya dalam organisasi.

b. Me-*monitor* dan Menyesuaikan Pelaksanaan Strategi

Me-*monitor* pelaksanaan dari rencana strategi yang telah dibuat memungkinkan eksekutif untuk mengidentifikasi permasalahan kritis dan membuat strategi untuk mengatasinya.

c. Menyampaikan Wawasan dan informasi ke semua pihak

Menyajikan informasi menggunakan grafik, simbol, bagan dan warna yang memudahkan pengguna dalam memahami informasi secara benar.

2.4.2 Karakteristik *Dashboard*

Karakteristik *dashboard* operasional yaitu:

- a. Model pemrosesan yang berdsarkan kejadian yaitu menangkap kejadian setiap saat dari beberapa sistem yang mencakup dan mempengaruhi proses bisnis.
- b. Aturan bisnis yang kuat yaitu mengijinkan penggunaanya membuat peringatan, target, ambang untuk nilai kerja individu.
- c. *Dashboard* bisnis yang *user friendly* yaitu memperbarui nilai sebagai aliran kejadian melalui sistem dan menempatkan nilai tersebut dalam hubungan dengan menghubungkan ke pencapaian bisnis.

- d. Sebuah sistem aliran kerja yang bergabung dan bekerja sama yang memungkinkan pengguna untuk melalui proses secara formal dan informasi, yang dengan proses itu pengguna dapat berkolaborasi mendiskusikan hasilnya.

Karakteristik *dashboard* menurut Hariyanti, yaitu:

a. *Synergetic*

Ergonomis dan memiliki tampilan visual yang mudah dipahami oleh pengguna. *Dashboard* mensinergikan informasi dari berbagai aspek yang berbeda dalam satu layar.

b. *Monitor*

Menampilkan KPI yang diperlukan dalam pembuatan keputusan dalam domain tertentu, sesuai dengan tujuan pembangunan *dashboard* tersebut.

c. *Accurate*

Informasi yang disajikan harus akurat dengan tujuan untuk mendapatkan kepercayaan dari pengguna.

d. *Responsive*

Merespon threshold yang telah didefinisikan, dengan memberikan alert (seperti bunyi alarm, blinker, email) untuk mendapatkan perhatian pengguna terhadap hal-hal yang kritis.

e. *Timely*

Menampilkan informasi teknik yang diperlukan untuk pengambilan keputusan.

f. Interactive

Pengguna dapat melakukan *drill down* dan mendapatkan informasi yang lebih detail, analisis sebab akibat dan sebagainya.

g. More Data History

Melihat tren sejarah KPI contohnya perbandingan jumlah pencapaian penjualan periode saat ini dengan beberapa tahun yang lalu, untuk mengetahui apakah kondisi sekarang lebih baik atau tidak.

h. Personalized

Penyajian informasi spesifik untuk setiap jenis pengguna sesuai domain, tanggung jawab, hak akses, dan batasan akses data.

i. Analytical

Fasilitas untuk melakukan analisis, seperti analisis sebab akibat.

j. Collaborative

Fasilitas pertukaran catatan (laporan)

k. Trackability

Memungkinkan setiap pengguna untuk mengkustomisasi nilai yang akan dilacaknya. (Hariyanti, 2008:7).

2.4.3 Ciri-ciri Dashboard

Dashboard yang didesain baik, akan menampilkan informasi yang:

- a. Luar biasa terorganisir dengan baik.
- b. Meringkas, terutama dalam bentuk ringkasan dan bentuk pengecualian.
- c. Spesifik dan disesuaikan untuk user dan tujuan *dashboard*.

- d. Ditampilkan secara ringkas, kadang dalam media kecil yang mengkomunikasikan data dan pesan tersebut dengan jelas dan langsung pada intinya. (Few, 2006).

2.4.4 Klasifikasi *Dashboard*

Adapun klasifikasi *dashboard* akan dijelaskan sebagai berikut:

1. *Dashboard* untuk tujuan Strategi
 - a. Mendukung manajemen level strategis.
 - b. Informasi untuk membuat keputusan bisnis, memprediksi peluang, dan memberikan arahan pencapaian tujuan strategis.
 - c. Fokus pada pengukuran kinerja *high-level* dan pencapaian tujuan strategis organisasi.
 - d. Mengadopsi konsep *Balance Score Card*.
 - e. Informasi yang disajikan tidak terlalu banyak dan disajikan secara ringkas.
 - f. Informasional disajikan dengan mekanisme yang sederhana, melalui tampilan yang “*unidirectional*”.
 - g. Tidak didesain untuk berinteraksi, dalam melakukan analisis yang lebih detail.
 - h. Tidak memerlukan *realtime*.
2. *Dashboard* untuk Taktikal
 - a. Mendukung manajemen level taktikal.
 - b. Memberikan informasi yang diperlukan oleh analis untuk mengetahui penyebab suatu kejadian.
 - c. Fokus pada proses analisis untuk menemukan penyebab dari suatu kondisi atau keadaan tertentu.

- d. Dengan fungsi *drill-down* dan navigasi yang baik.
- e. Memiliki konten informasi yang lebih banyak (analisis perbandingan, pola/tren, evaluasi kinerja).
- f. Menggunakan media penyajian yang “cerdas”, yang memungkinkan pengguna melakukan analisis terhadap data yang kompleks.
- g. Didesain untuk berinteraksi dengan data.
- h. Tidak memerlukan data *realtime*.

3. *Dashboard* untuk operasional

- a. Mendukung manajemen level operasional.
- b. Memberikan informasi mengenai aktifitas yang sedang terjadi, beserta perubahannya secara *realtime* untuk memberikan kewaspadaan terhadap hal-hal yang perlu direspon secara cepat.
- c. Fokus pada *monitoring* aktifitas dan kejadian yang berubah secara konstan.
- d. Informasi disajikan spesifik, tingkat detailnya cukup dalam.
- e. Media penyajian sederhana.
- f. *Alert* disajikan dengan cara yang mudah dipahami, dan mampu menarik perhatian pengguna.
- g. Bersifat dinamis, sehingga memerlukan *realtime*.
- h. Didesain untuk berinteraksi dengan data, untuk mendapatkan informasi yang lebih detail, maupun informasi pada level yang lebih atas.

Untuk penyajian di Dinkes, *dashboard* yang akan digunakan adalah jenis operasional karena dalam hal ini Dinkes membutuhkan data yang *realtime* yang dapat digunakan untuk *memonitoring*, sehingga bila diketahui ada indikator yang

belum terpenuhi dapat segera dilakukan evaluasi guna untuk mengambil keputusan.

2.4.5 Kesalahan Umum Pembuatan *Dashboard*

Beberapa hal dibawah ini merupakan 13 kesalahan umum pada pembuatan *dashboard* (Few, 2006)

1. Melebihi batas pada satu layar monitor komputer. Hal ini mengacu pada tampilan *dashboard*.
2. Menyediakan data yang tidak memadai.
3. Menampilkan detail atau presisi yang berlebihan: *dashboard* hampir selalu memerlukan informasi tingkat tinggi untuk mampu mendukung penggunaanya untuk peninjauan cepat.
4. Memilih ukuran kurang tepat.
5. Memilih media tampilan yang tidak tepat atau salah memilih media. (*bar, pie, circle* atau radar).
6. Menyajikan variasi berbeda yang sia-sia
7. Menggunakan media tampilan yang desainya payah.
8. Menampilkan kuantitas data secara tidak akurat.
9. Mengatur tampilan data dengan payah. *Dashboard* pada dasarnya menampilkan informasi yang banyak dengan tampilan seminimalis mungkin.
10. Menyortir data penting secara tidak efektif atau tidak sama sekali. *Dashboard* yang baik adalah menonjolkan data yang lebih penting dibanding yang lain. sehingga pengguna langsung melihatnya.

11. Mengacaukan tampilan dengan dekorasi yang tak perlu. Sebaiknya tampilan tidak terlalu “wah”, hal ini akan menyebabkan mata penggunanya mudah lelah dikemudian hari.
12. Salah satu berlebihan menggunakan warna, gunakan warna yang tepat.
13. Mendesain tampilan yang tidak atraktif seperti tidak ada *Combobox*-nya

2.5 Siklus Hidup Pengembangan Sistem

Siklus hidup pengembang sistem atau *Software Development System Life Cycle* (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik) (Chandra, 2012 : 13).

2.5.1 Elisitasi Kebutuhan

Elisitasi atau pengumpulan kebutuhan merupakan aktivitas awal dalam proses rekayasa perangkat kebutuhan. Sebelum kebutuhan dapat dianalisis, dimodelkan, atau ditetapkan, kebutuhan harus dikumpulkan melalui proses elisitasi. Elisitasi kebutuhan adalah sekumpulan aktivitas yang ditujukan untuk menemukan kebutuhan suatu sistem melalui komunikasi dengan pelanggan, pengguna sistem dan pihak lain yang memiliki kepentingan dalam pengembangan sistem.

Sejalan dengan proses rekayasa kebutuhan secara keseluruhan, elisitasi kebutuhan bertujuan untuk:

- a. Mengetahui masalah apa saja yang perlu dipecahkan dan mengenali batasan-batasan sistem. Proses-proses dalam pengembangan perangkat lunak sangat ditentukan oleh seberapa dalam dan luas pengetahuan developer tentang permasalahan.
- b. Mengenali siapa saja para *stakeholder*, yaitu setiap pihak yang memiliki kepentingan terhadap sesuatu, dimana dalam konteks perangkat lunak adalah proyek pengembangan perangkat lunak itu sendiri, beberapa yang dapat dikatakan sebagai *stakeholder* antara lain adalah konsumen atau klien yang membayar sistem, pengembang yang merancang, membangun, dan merawat sistem, dan pengguna yang berinteraksi dengan sistem untuk mendapatkan hasil kerja mereka.
- c. Mengenali tujuan dari sistem yaitu sasaran-sasaran yang harus dicapai. Tujuan merupakan sasaran sistem yang harus dipenuhi, penggalan *high level goals* di awal proses pengembangan sangatlah penting karena bertujuan lebih terfokus pada ranah masalah dan kebutuhan *stakeholder* dari pada solusi yang dimungkinkan untuk masalah tersebut (Chandra, 2012 : 12-14).

2.5.2 Analisis

Analisis adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Tahap analisis sistem dilakukan setelah tahap perencanaan dan sebelum tahap desain sistem. Tahap analisis merupakan tahap yang kritis dan sangat

penting karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan ditahap selanjutnya (Jogiyanto, 2005: 129- 150).

1. Langkah-langkah Analisis Sistem

Langkah-langkah di dalam tahap analisis sistem hampir sama dengan langkah-langkah yang dilakukan dalam mendefinisikan proyek-proyek sistem yang akan dikembangkan di tahap perencanaan sistem.

Di dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem sebagai berikut ini:

- a. *Identify*, yaitu mengidentifikasi masalah
- b. *Understand*, yaitu memahami kerja dari sistem yang ada
- c. *Analyze*, yaitu menganalisis sistem
- d. *Report*, yaitu membuat laporan hasil analisis

2. Mengidentifikasi Masalah dan Analisis

Merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Tugas-tugas yang harus dilakukannya adalah sebagai berikut ini.

- a. Mengidentifikasi penyebab masalah.
- b. Mengidentifikasi titik keputusan.
- c. Mengidentifikasi personil-personil kunci

3. Memahami Kerja dari Sistem yang Ada

Langkah ke dua dari tahap analisis sistem adalah memahami kerja dari sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari secara terinci

bagaimana sistem yang ada beroperasi. Untuk mempelajari operasi dari sistem ini diperlukan data yang dapat diperoleh dengan cara melakukan penelitian.

4. Menganalisis Hasil Penelitian

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang telah dilakukan. Menganalisis hasil penelitian sering sulit dilakukan oleh analis sistem yang masih baru. Pengalaman menunjukkan bahwa banyak analis sistem yang masih baru mencoba untuk memecahkan masalah tanpa menganalisisnya.

5. Membuat Laporan Hasil Analisis

Setelah proses analisis sistem ini selesai dilakukan, tugas berikutnya dari analis sistem dan teamnya adalah membuat laporan hasil analisis. Laporan ini diserahkan kepada *steering committee* yang nantinya akan diteruskan ke manajemen. Tujuan utama dari penyerahan laporan ini kepada manajemen adalah:

- a) Pelaporan bahwa analisis telah selesai dilakukan;
- b) Meluruskan kesalah-pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analis sistem tetapi tidak sesuai menurut manajemen;
- c) Meminta pendapat-pendapat dan saran-saran dari pihak manajemen;
- d) Meminta persetujuan kepada pihak manajemen untuk melakukan tindakan selanjutnya (dapat berupa meneruskan ke tahap desain sistem atau mengehentikan proyek bila dipandang tidak layak lagi) (Jogiyanto, 2005: 130- 149).

2.5.3 Desain

Menurut John Burch & Gary Grudnitski, desain adalah penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

Analisis sistem dapat mendesain model dari sistem informasi yang diusulkan dalam bentuk *physical system* dan *logical model*. Bagan alir sistem (*system flowchart*) merupakan alat yang tepat digunakan untuk menggambarkan *physical system*.

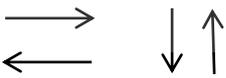
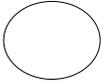
Logical model dari sistem informasi lebih menjelaskan kepada *user* bagaimana nantinya fungsi-fungsi di sistem informasi secara logika akan bekerja.

Logical model dapat digambarkan dengan menggunakan diagram arus data (*data flow diagram*). (John Burch & Gary Grudnitski, 1986 : 461).

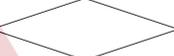
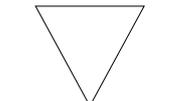
1. Flowchart

a. Flow Direction Symbol

Tabel 2.4 Flow Direction Symbol

	Simbol arus / <i>flow</i> , yaitu menyatakan jalannya arus suatu proses
	Simbol <i>connector</i> , berfungsi menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
	Simbol <i>off-page connector</i> , menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.

b. *Processing Symbols*Tabel 2.5 *Processing Symbols*

	<p>Simbol <i>process</i>, yaitu menyatakan suatu tindakan (proses) yang dilakukan oleh komputer</p>
	<p>Simbol <i>manual</i>, yaitu menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer</p>
	<p>Simbol <i>decision</i>, yaitu menunjukkan suatu kondisi tertentu yang akan menghasikan dua kemungkinan jawaban : ya / tidak.</p>
	<p>Simbol <i>preparation</i>, yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal</p>
	<p>Simbol <i>terminal</i>, yaitu menyatakan permulaan atau akhir suatu program.</p>
	<p>Simbol <i>offline-storage</i>, menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu</p>
	<p>Simbol <i>manual-input</i>, memasukkan data secara manual dengan menggunakan online keyboard</p>

c. *Input/Output Symbol*

Tabel 2.6 *Input / Output Symbol*

	Simbol <i>input-output</i> menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
	Simbol <i>storage</i> menyatakan input berasal dari disk atau <i>output</i> disimpan ke disk
	Simbol <i>document</i> mencetak keluaran dalam bentuk dokumn (melalui printer)
	Simbol <i>display</i> mencetak keluaran dalam layar monitor.

2. *Data Flow Diagram*

DFD adalah diagram yang menggunakan notasi-notasi ini untuk menggambarkan arus dari data sistem, sekarang di kenal dengan nama diagram arus data (*data flow diagram*). DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan di kembangkan secara logika tanpa mempertibangkan lingkungan fisik dimana data tersebut mengalir.

a. *External Entity*

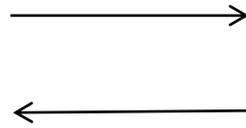
External entity merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.



Gambar 2.2 Simbol *Eksternal Entity*

b. *Data Flow*

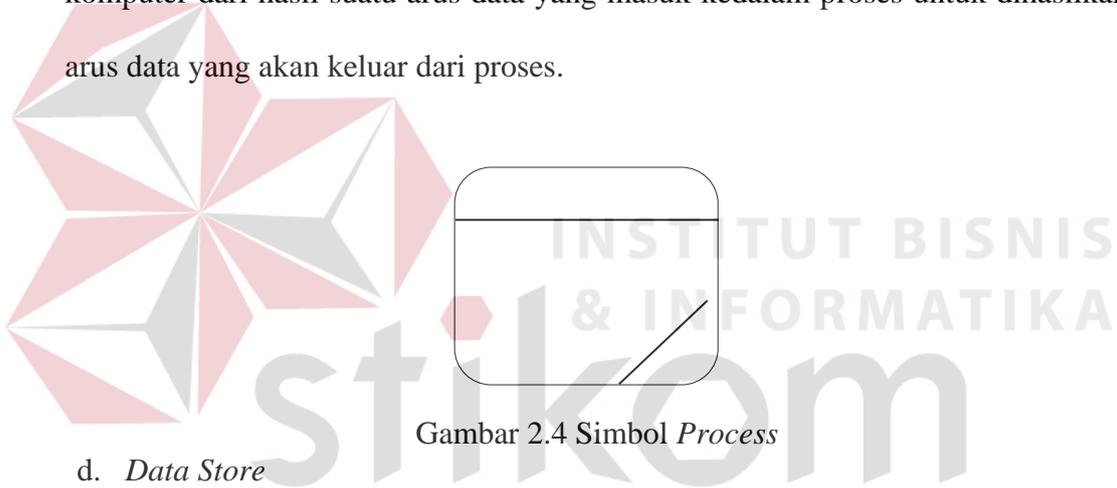
Data flow menunjukkan arus dari data yang berupa masukan untuk sistem atau hasil dari proses sistem dan dapat berbentuk sebagai berikut ini.



Gambar 2.3 Simbol *Data flow*

c. *Process*

Process adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses.



Gambar 2.4 Simbol *Process*

d. *Data Store*

Data store adalah simpanan dari data yang berupa, suatu file database di sistem komputer, arsip atau catatan manual, dan suatu tabel acuan manual.



Gambar 2.5 Simbol *Data source*

2.5.4 *Construction*

Software construction lebih diartikan sebagai pembuatan detail dari suatu pekerjaan, menciptakan satu *software* yang penting yang dikombinasikan dengan *code*, proses verifikasi, *testing unit*, dan testing yang terintegrasi, serta proses

debuging. *Software construction* lebih sering dihubungkan dengan proses desain dan proses testing. Hal ini dikarenakan proses tersebut saling ketergantungan satu sama lain, dimana *software construction* merupakan keluaran dari desain *software* dan juga sebagai masukan dari *software testing*. *Software construction* bertipikal memproduksi volume konfigurasi item yang lebih tinggi dan juga dibutuhkan dalam mengelola sebuah software proyek (file sumber, isi, test cases, dll) (England, John Wiley & Sons, 2004: 65-67)

1. *Software Construction Fundamentals*

Pada tahap pertama, dilakukan pendefinisian dasar tentang prinsip-prinsip yang digunakan dalam proses implementasi seperti minimalisasi kompleksitas, mengantisipasi perubahan, dan standar yang digunakan.

2. *Managing Construction*

Bagian ini mendefinisikan tentang model implementasi yang digunakan, rencana implementasi, dan ukuran pencapaian dari implementasi tersebut.

3. *Practical Considerations*

Bagian ini membahas tentang desain implementasi yang digunakan, bahasa pemrograman yang digunakan, kualitas dari implementasi yang dilakukan, proses pengetesan dan integritas.

Dalam proses pengimplementasian ini, digunakan beberapa aplikasi pendukung yaitu:

a. Bahasa Pemrograman PHP

Bahasa Pemrograman PHP adalah bahasa pemrograman yang bekerja dalam sebuah webserver. Script-script PHP harus tersimpan dalam sebuah server dan dieksekusi atau diproses dalam server tersebut. Dengan menggunakan

program PHP, sebuah *website* akan lebih interaktif dan dinamis. (Madcoms, 2011: 186).

b. Database MySQL

Database MySQL adalah jenis *database* yang sangat populer dan digunakan pada banyak *website* di internet sebagai bank data, selain itu *Database* MySQL juga dapat dijalankan di beberapa *platform*, antara lain linux, windows, dan sebagainya (Madcoms, 2011 : 215).

2.5.5 Testing dan Implementasi

Tahap ini mendemonstrasikan sistem perangkat lunak yang telah selesai dibuat untuk dijalankan, apakah telah sesuai dengan kebutuhan yang telah dispesifikasikan dan dapat diadaptasi pada lingkungan sistem yang baru. Tahapan ini tertuang dalam suatu dokumen *Test Plan*, yang dimulai dari membuat *Software Testing fundamentals* yang berisi tentang penjelasan penting mengenai terminology testing, kemudian selanjutnya merancang *Test Levels* yang terbagi antara target pengetesan dan objektif dari pengetesan. Pada tahap berikutnya adalah mendefinisikan *Test Techniques*, yaitu tentang bagaimana teknik yang digunakan termasuk dasar-dasar pengetesan berdasarkan intuisi dan pengalaman serta teknik pengetesan secara teknik *coding*, teknik kesalahan, teknik penggunaan, dan teknik terkait lainnya. Tahap selanjutnya adalah mendefinisikan *Test – Related Measures*, yaitu ukuran-ukuran pencapaian testing yang telah dilakukan untuk kemudian dievaluasi kembali. Tahap terakhir adalah mendefinisikan *test Process* yang berisi tentang aktivitas testing. (England, John Wiley & sons, 2004 : 73-74).

2.5.6 Maintenance

Pada tahap ini akan dilakukan pendeskripsian pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan pengguna termasuk implementasi akhir dan proses peninjauan kembali. Pemeliharaan sistem ini terdiri dari beberapa jenis yaitu:

- a. *Corrective*, yaitu memperbaiki desain dan *error* pada program.
- b. *Adaptive*, yaitu memodifikasi sistem untuk beradaptasi dengan perubahan lingkungan.
- c. *Perfective*, yaitu melibatkan sistem untuk menyelesaikan masalah baru atau mengambil kesempatan untuk penambahan fitur.
- d. *Preventive*, yaitu menjaga sistem dari kemungkinan masalah di masa yang akan datang.

Prosedur pemeliharaan tersebut disusun dalam beberapa tahapan. Tahap awal adalah menyusun *software maintenance fundamentals* yang berisi tentang dasar-dasar pemeliharaan, segala yang dibutuhkan untuk melakukan pemeliharaan, dan kategori pemeliharaan. Selanjutnya adalah mendefinisikan *Key Issues in Software Maintenance*, yang berisi tentang teknik pemeliharaan, manajemen pemeliharaan dan biaya, serta ukuran pemeliharaan perangkat lunak. Tahap selanjutnya adalah mendefinisikan proses dan aktivitas pemeliharaan tersebut ke dalam *Maintenance Process*. (England, John Wiley & Sons, 2004: 90-91).