

## BAB II

### LANDASAN TEORI

#### 2.1 Sistem

Sistem dapat didefinisikan dengan dua kelompok pendekatan. Pertama, lebih menekankan pada prosedur yang digunakan dalam sistem dan mendefinisikan sistem sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu. Kedua, lebih menekankan pada elemen atau komponen penyusun sistem dan mendefinisikannya sebagai kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu (Jogiyanto, 2003:1).

#### 2.2 Data dan Informasi

Data adalah sebuah kebenaran, atau kenyataan, contoh nama pegawai, order penjualan, nomor penjualan. (Stair, 2006) Sistem yang terbuka merupakan sistem yang dihubungkan dengan arus sumber daya luar dan tidak.

Informasi adalah sekumpulan kebenaran atau kenyataan yang terorganisir sedemikian rupa yang menyebabkan mereka memiliki nilai tambah daripada kumpulan kebenaran itu sendiri. (Stair, 2006)

Untuk menjadi bernilai bagi manajemen dan pembuat keputusan, informasi seharusnya memiliki karakteristik seperti dibawah ini:

1. Akurat

Informasi yang akurat adalah informasi yang bebas dari *error*. Dalam beberapa kasus, informasi yang tidak akurat dihasilkan karena data yang digunakan pada pemrosesan tidak akurat.

2. Lengkap

Informasi yang lengkap berisi semua kebenaran (data) yang lengkap. Contoh informasi mahasiswa keluar tidak akan lengkap tanpa informasi alasan mahasiswa keluar.

3. Ekonomis

Informasi seharusnya ekonomis dalam pembuatannya. Para pembuat keputusan selalu akan membandingkan nilai guna informasi dan biaya yang dikeluarkan untuk membuatnya

4. Fleksibel

Informasi yang fleksibel dapat digunakan untuk berbagai tujuan

5. Handal

Informasi yang handal dapat diandalkan. Dalam banyak kasus, kehandalan sebuah informasi bergantung pada metode pengumpulan data tersebut. Dalam contoh lain, kehandalan ini bergantung pada sumber dari informasi tersebut.

6. Relevan (berhubungan)

Informasi yang relevan penting bagi pembuat keputusan. Istilahnya, informasi bahwa harga kayu turun, tidak relevan bagi pabrik *chip* komputer.

7. Sempel

Informasi seharusnya juga simpel, tidak terlalu rumit. Informasi yang mutakhir dan detil mungkin tidak dibutuhkan. Kenyataannya, informasi yang berlebih dapat menyebabkan *overload* informasi, dimana para pembuat keputusan mempunyai informasi berlebih dan tidak bias menentukan mana yang penting.

8. Tepat waktu

Informasi yang tepat waktu adalah informasi yang ada pada saat yang dibutuhkan.

9. Dapat dibuktikan

Informasi seharusnya dapat dibuktikan. Ini berarti anda dapat memeriksa untuk memastikan bahwa informasi tersebut benar, mungkin dengan memeriksa sumber lain untuk informasi yang sama.

10. Dapat diakses

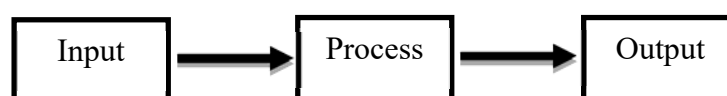
Informasi seharusnya mudah diakses oleh pengguna yang mendapatkan bentuk informasi yang tepat dan disaat yang tepat untuk mendapatkan yang mereka butuhkan.

11. Aman

Informasi seharusnya aman dari jamahan pengguna yang tidak berhak mengakses.

### 2.3 Sistem Informasi

Menurut Kadir (2003), sistem informasi adalah sebuah konsep yang memiliki definisi tertentu yang digunakan untuk dapat dimengerti oleh pihak tertentu sehingga informasi tersebut memiliki makna dan tujuan yang tepat untuk dilakukan penindaklanjutan, informasi tersebut dimasukkan kedalam sistem yang menjadikan fungsi menjadi input informasi dan dilakukan proses yang menghasilkan output informasi.



Gambar 2.1 Model Proses Transformasi

Menurut Herlambang dan Tanuwijaya (2005), kegiatan sistem informasi mencakup :

1. *Input*, merupakan suatu kegiatan untuk menyediakan data untuk diproses.
2. Proses, merupakan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
3. *Output*, merupakan suatu kegiatan yang menghasilkan laporan dari proses.
4. Penyimpanan, merupakan suatu kegiatan untuk memelihara dan menyimpan data.

### 2.3.1 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).

### 2.3.2 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. Dan 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 mengcover 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 detail 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 sudah 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 mempercayai 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).

#### 2.3.3 Sistem Informasi Akademik

Menurut Etin (2011) secara spesifik sistem informasi akademik memiliki beberapa karakter yang cukup luas, yaitu sistem informasi akademik bermakna sebagai pendekatan-pendekatan dalam melakukan proses manajemen; komputer hanya merupakan komponen, atau alat bukan fokus sentral dari sistem informasi akademik; Pimpinan berperan aktif dalam rangka sistem sebagai pengguna informasi bukan sebagai tenaga teknis ataupun operator komputer, dan esensi sistem informasi administrasi terletak pada sistem terpadu dan sistem terencana,

bukan hanya urusan mekanisme pengolahan data. Maka, sistem informasi akademik adalah sistem yang dihimpun dari berbagai macam data yang dikelola se-otomatis mungkin dengan alat dan metode sehingga menghasilkan informasi yang diperlukan bagi terlaksananya kegiatan akademis.

Sedangkan menurut Satoto (2009) sistem informasi akademik adalah perangkat lunak yang digunakan untuk menyajikan informasi dan menata administrasi yang berhubungan dengan kegiatan akademis. Dengan menggunakan perangkat lunak seperti ini diharapkan kegiatan administrasi akademis dapat dikelola dengan baik dan informasi yang diperlukan dapat diperoleh dengan mudah dan cepat.

## 2.4 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).

### 2.4.1 Karakteristik Dashboard

Tujuan dalam penggunaan dashboard menurut Eckerson (Hariyanti, 2008:7), 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. Memonitor dan menyesuaikan pelaksanaan strategi.

Memonitor pelaksanaan dari rencana dan 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 penggunaan dalam memahami dan mempersepsi informasi secara benar.

- d. Memonitor dan menyesuaikan pelaksanaan strategi

Memonitor pelaksanaan dari rencana dan strategi yang telah dibuat. Memungkinkan eksekutif untuk mengidentifikasi permasalahan kritis dan membuat strategi untuk mengatasinya.

- e. Menyampaikan wawasan dan informasi ke semua pihak

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

#### **2.4.2 Karakteristik Dashboard**

Karakteristik Dashboard operasional yaitu :

- a. Model pemrosesan yang berdasarkan 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 bekerjasama yang mengizinkan pengguna untuk memulai proses secara formal dan informal, yang dengan proses itu pengguna dapat berkolaborasi mendiskusikan hasilnya.

### 2.4.3 Ciri-ciri Dashboard

Dashboard yang didesain baik, akan menampilkan informasi yang :

- a. Luar biasa terorganisasi 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 langsung pada intinya. (Few, 2006:97).

## 2.5 Analisis dan Desain Perangkat Lunak

Analisis sistem atau perangkat lunak dilakukan dengan tujuan untuk dapat mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikannya.

Perancangan desain perangkat lunak merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau

tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.

Menurut Kendall dan Kendall (2003), analisis dan perancangan sistem berupaya menganalisis *input* data atau aliran data secara sistematis, memproses atau mentransformasikan data, menyimpan data, dan menghasilkan *output* informasi dalam konteks bisnis khusus. Kemudian, analisis dan perancangan sistem tersebut dipergunakan untuk menganalisis, merancang dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang bisa dicapai melalui penggunaan sistem informasi terkomputerisasi.

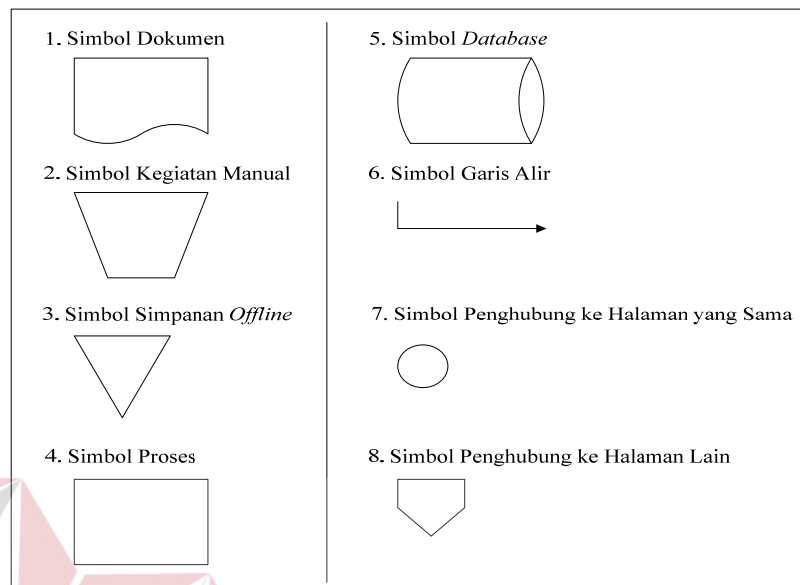
Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini juga akan menyebabkan kesalahan di tahap selanjutnya. Dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem sebagai berikut:

- 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.

### **2.5.1 Bagan Alir Sistem**

Menurut Hartono (2005:796), “Bagan alir sistem (*System Flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dalam sistem”. Bagan alir sistem digunakan untuk menjelaskan urutan kegiatan yang dilakukan oleh sistem. Pengertian bagan alir dokumen (*Document Flowchart*) menurut Hartono (2005) adalah bagan alir dari laporan dan formulir. Berikut ini

merupakan penjelasan dan simbol yang digunakan untuk menggambarkan kedua bagan alir tersebut :



Gambar 2.2 Simbol Bagan Alir Sistem Dan Bagan Alir Dokumen

#### 1. Simbol Dokumen

Simbol ini menunjukkan dokumen *input* dan *output* baik untuk proses manual atau komputer.

#### 2. Simbol Kegiatan Manual

Simbol ini menunjukkan pekerjaan manual.

#### 3. Simbol Simpanan *Offline*

Simbol ini menunjukkan *file* non-komputer yang diarsip.

#### 4. Simbol Proses

Simbol ini menunjukkan kegiatan proses dari operasi program komputer.

#### 5. Simbol *Database*

Simbol ini menunjukkan tempat untuk menyimpan data hasil operasi komputer.

#### 6. Simbol Garis Alir

Simbol ini menunjukkan arus dari proses.

## 7. Simbol Penghubung

Simbol ini menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.

### 2.5.2 Data Flow Diagram

*Data Flow Diagram* atau DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik di tempat data tersebut mengalir. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur dan dapat mengembangkan arus data di dalam sistem dengan terstruktur dan jelas.

Menurut Yourdon (1989), Simbol yang digunakan dalam menggambarkan DFD ada empat (4), yakni :

#### a. *External Entity*

Entitas luar merupakan aktor yang berkomunikasi dengan sistem. Aktor biasanya menunjukkan individu, kelompok (departemen, divisi, dan lainnya), sistem terkomputerisasi, dan organisasi di luar sistem yang dibuat. Dalam beberapa kasus, aktor dapat berupa sistem lain, seperti sistem terkomputerisasi lain yang digunakan untuk berkomunikasi dengan sistem yang dibuat. Aktor dipresentasikan dalam bentuk kotak yang dapat dilihat pada Gambar 2.3.

Tiga hal yang perlu diperhatikan dalam membuat aktor, yaitu:

1. Aktor berada di luar sistem yang dibuat, alur yang dibuat mengkoneksikan aktor ke berbagai proses dalam sistem .
2. Dalam menggambarkan aktor, baik analis sistem maupun desainer sistem tidak berhak mengganti nama aktor sesuai keinginan. Penamaan aktor

digambarkan harus sesuai dengan kebutuhan sistem yang sebenarnya, bukan dengan niat mempermudah atau memberikan penamaan secara bebas.

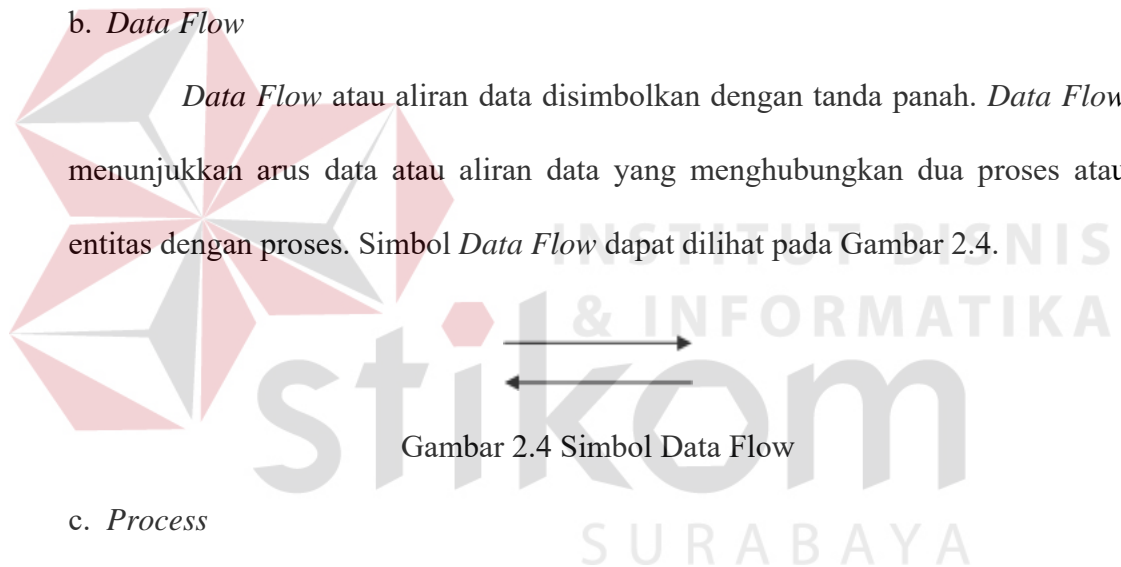
3. Alur relasi antar aktor tidak perlu digambarkan, apabila relasi tersebut perlu digambarkan sebagai bagian dari sistem, ada baiknya relasi tersebut dibuat sebagai sebuah proses.



Gambar 2.3 Simbol External Entity

b. *Data Flow*

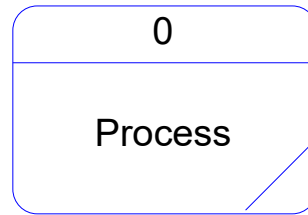
*Data Flow* atau aliran data disimbolkan dengan tanda panah. *Data Flow* menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses. Simbol *Data Flow* dapat dilihat pada Gambar 2.4.



Gambar 2.4 Simbol Data Flow

c. *Process*

Suatu *Process* yang meliputi beberapa tindakan atau sekelompok tindakan dari arus data yang masuk untuk dijalankan atau diproses agar menghasilkan suatu keluaran.. Mayoritas penamaan proses akan mendeskripsikan kegiatan apa yang akan dilakukan pada proses tersebut. Terkadang proses juga dapat berupa nama orang atau sekelompok orang, komputer, prangkat mekasnis. Simbol *Process* dapat dilihat pada Gambar 2.5.

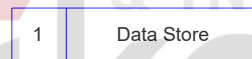


Gambar 2.5 Simbol Process

#### d. *Data Store*

*Data Store* adalah simbol yang digunakan untuk melambangkan proses penyimpanan data. Suatu nama perlu diberikan pada *Data Store* untuk menunjukkan nama dari *file*-nya. Gambar 2.5 merupakan simbol *file* penyimpanan/*Data Store* yang dapat sebagai gambaran:

1. Suatu file atau *database* di sistem komputer.
2. Suatu arsip atau catatan manual.
3. Suatu tabel acuan manual.



Gambar 2.6 Simbol Data Store

## 2.6 PHP

Menurut dokumen resmi PHP, PHP merupakan singkatan dari *Programming Hypertext Preprocessor*. PHP merupakan bahasa berbentuk skrip yang ditempatkan di dalam *server* dan diproses di *server*. Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, PHP dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, pengguna dapat menampilkan isi suatu *database* pada halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *ColdFusion*, ataupun *Perl*. Namun perlu diketahui bahwa PHP sebenarnya dapat

dipakai secara *command line*, artinya skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *web browser*.

Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip *Perl* yang dapat diamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi *tool* yang disebut "*Portable Home Page*". Paket inilah yang menjadi cikal bakal dari PHP. Pada tahun 1995, Rasmus menciptakan PHP/F1 versi 2. Pada versi inilah pemrogram dapat menempelkan kode terstruktur di dalam *tag* HTML. Yang menarik, kode PHP juga dapat berkomunikasi dengan *database* dan melakukan perhitungan-perhitungan yang kompleks. (Kadir, 2008:2).

## 2.7 MySQL

Sebuah perangkat sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database management system*) yang dapat *multithread* dan *multi-user*.

Sebagai *database server*, MySQL dapat dikatakan lebih unggul daripada *database server* lainnya, terutama dalam kecepatan. Berikut ini keunggulan MySQL antara lain:

### 1. Portability

MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server dan solaris.

### 2. Multiuser

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

### 3. *Security*

MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta *password* terenkripsi.

### 4. *Scalability dan limit*

MySQL mampu menangani *dabase* dalam skala besar, dengan jumlah *record* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

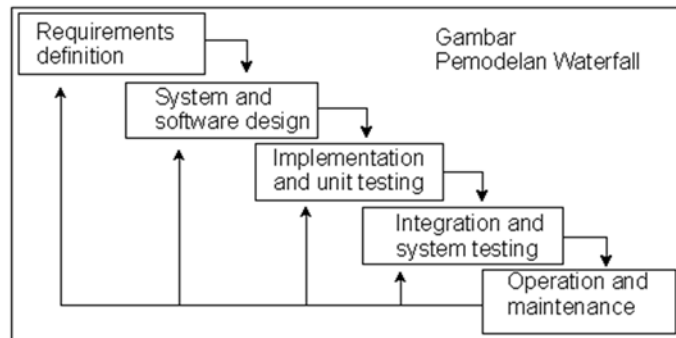
## 2.8 Siklus Hidup Pengembangan Sistem

Siklus hidup pengembangan sistem adalah nama lain dari *Software Development Life Cycle* (SDLC) ini merupakan siklus hidup dari pengembangan sistem yang digunakan untuk menggambarkan tahapan utama dan langkah-langkah di dalam tahapan tersebut dalam proses pengembangannya (Hartono, 2001). Tahapan-tahapannya adalah *requirements* (analisis sistem), *analysis* (analisis kebutuhan sistem), *design* (perancangan), *coding* (implementasi), *testing* (pengujian), dan *maintenance* (perawatan). Ada beberapa model SDLC. Model yang cukup populer dan banyak digunakan adalah waterfall. Beberapa model lain SDLC misalnya *fountain*, *spiral*, *rapid*, *prototyping*, *incremental*, *build & fix*, dan *synchronize & stabilize*.

Menurut Pressman (2015), menjelaskan bahwa nama lain dari model *waterfall* adalah *Linear Sequential Model*. Model ini merupakan model yang paling banyak dipakai dalam *Software Engineering*. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap



analisis, desain, *coding*, *testing* dan *maintenance*. Berikut ini adalah gambar dari model *waterfall* secara umum.



Gambar 2.7 Model Waterfall menurut Pressman dalam Harnaningrum (2007)

Gambar 2 menunjukkan tahapan umum dari model proses *waterfall*. Akan tetapi, Pressman (2015) memecah model ini meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya. Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam model *waterfall*:

### 1. *Requirements definition*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada perangkat lunak. Untuk mengetahui sifat dari program yang akan dibuat, maka *software engineer* harus mengerti tentang domain informasi dari perangkat lunak, misalnya fungsi yang dibutuhkan, *user interface*, dan sebagainya. Dari kedua aktivitas tersebut (pencarian kebutuhan sistem dan perangkat lunak) harus didokumentasikan dan ditunjukkan kepada pelanggan.

### 2. *System and software design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan di atas menjadi representasi ke dalam bentuk “*blueprint*” perangkat lunak sebelum pengkodean dimulai. Desain harus dapat mengimplementasikan kebutuhan yang

telah disebutkan pada tahap sebelumnya. Seperti kedua aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari perangkat lunak.

### 3. *Implementation and unit testing*

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses pengkodean. Tahap ini merupakan implementasi dari tahap desain yang secara teknis nantinya dikerjakan oleh *programmer*.

### 4. *Integration and system testing*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan perangkat lunak. Semua fungsi-fungsi perangkat lunak harus diujicobakan, agar perangkat lunak bebas dari *error* dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

### 5. *Operation and maintenance*

Pemeliharaan suatu perangkat lunak diperlukan, termasuk di dalamnya adalah pengembangan, karena perangkat lunak yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada *error* kecil yang tidak ditemukan sebelumnya atau ada penambahan fitur-fitur yang belum ada pada perangkat lunak tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi atau perangkat lainnya.



Tabel 2. 1 Klasifikasi penilaian di UNSIDA

Nilai Angka	Huruf Mutu	Klasifikasi
> 81	A	Sangat Baik
75 - 80	AB	Antara Sangat baik dan baik
66 - 70	B	Baik
61 - 65	BC	Antara Baik dan Cukup
56 - 60	C	Cukup
41 - 55	D	Kurang
< 40	E	Gagal

Penilaian keberhasilan studi semester dilakukan tiap akhir semester. Penilaian ini meliputi semua matakuliah yang direncanakan oleh mahasiswa dalam FRS/ KRS yang sah pada semester tersebut dengan menggunakan rumus IP sebagai berikut :

$$IP = \frac{\sum_{i=1}^n (Ni \times ki)}{\sum_{i=1}^n ki}$$

Dengan keterangan :

IP = Indeks Prestasi      Ni = Nilai mutu mata kuliah i  
n = jumlah mata kuliah      Ki = Bobot sks mata kuliah i

Waktu perkuliahan dalam satu semester untuk setiap matakuliah adalah 19 minggu dengan rincian 14 minggu masa kuliah, 2 minggu masa UTS, 1 minggu masa tenang dan 2 minggu masa UAS.

Baik mahasiswa dan dosen prosentase kehadiran dari 16 kali pertemuan minimal 75%. Jika mahasiswa tidak dapat memenuhi prosentase kehadiran maka, mahasiswa tsb tidak dapat mengikuti ujian akhir atau UAS. Jika dosen, maka wajib melengkapi perkuliahan pada masa minggu tenang.

Tabel 2. 2 Tabel Indeks prestasi dan beban studi di UNSIDA

<u>Indeks Prestasi (IP)</u>	<u>Beban Studi</u>
$\geq 3,00$	Maksimal 24 SKS
2,50 s/d < 2,99	Maksimal 21 SKS
2,00 s/d 2,49	Maksimal 18 SKS
1,50 s/d 1,49	Maksimal 15 SKS
< 1,50	Maksimal 12 SKS

