



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Menurut Romney dan Steinbart (2006), sistem merupakan gabungan komponen yang saling terhubung dan saling berinteraksi untuk mencapai sebuah tujuan. O'Brien dan Marakas (2008), berpendapat sistem adalah gabungan komponen yang saling terhubung dengan memiliki fungsi yang jelas dan saling terintegrasi satu dengan yang lain hingga mencapai tujuan dengan cara memproses input menjadi sebuah output yang terorganisasi. Menurut Mulyadi (2008:2) tentang definisi sistem adalah sekelompok unsur-unsur yang sangat penting dan sangat berhubungan satu dengan lainnya. Sedangkan menurut Whitten dan Bentley (2007), komponen yang saling terhubung dalam sebuah sistem dan dapat bekerja bersama-sama untuk mencapai hasil yang diinginkan.

Dari beberapa definisi tersebut, dapat disimpulkan sistem adalah sebuah kumpulan unsur elemen yang sangat penting dan saling berinteraksi untuk mencapai suatu tujuan tertentu melalui tiga tahapan yaitu *input*, *process* dan *output*.

2.2 Pengertian Informasi

Menurut O'Brien dan marakas (2008) sebuah informasi merupakan data yang telah dikonversi dan diolah menjadi sesuatu yang lebih bermakna dan berguna bagi yang membutuhkannya. Informasi merupakan data yang sudah diolah dan diorganisir sehingga menghasilkan sesuatu yang lebih bernilai dan bermanfaat. Kenneth C. Laudon (2008), berpendapat bahwa Informasi adalah data yang telah diolah kedalam bentuk yang dapat digunakan oleh manusia.

Dapat ditarik sebuah kesimpulan bahwa informasi merupakan data yang telah diolah dan menjadi sebuah bentuk yang bernilai bagi penerimanya dalam mengambil sebuah keputusan saat ini atau di masa yang akan datang.

2.3 Sistem Informasi

Menurut Jhon Forbes Nash (1995:8) tentang definisi sistem informasi yang telah diterjemahkan oleh La Midjan dan Azhar Susanto, menyatakan bahwa sistem informasi merupakan sebuah kombinasi antar manusia, alat teknologi, media, prosedur dan pengendalian yang bertujuan untuk mengatur jaringan komunikasi yang sangat penting, proses untuk transaksi-transaksi tertentu, dan membantu para pemakai serta menyediakan ladsan atas pengambilan sebuah keputusan yang tepat. Sedangkan menurut R. Kelly, Rainer Jr. dan Efraim Turban (2009), menyatakan bahwa sistem informasi adalah sebuah bentuk proses dalam mengumpulkan, menyimpan, menganalisis, dan membagi informasi untuk tujuan tertentu. Komponen dari sistem informasi terdiri atas:

- a. *Hardware*: sekumpulan perangkat keras yang menerima sebuah data yang kemudian memproses dan menapilkannya. Contoh *hardware* adalah CPU (*Central Processing Unit*), *harddisk*, *monitor*, *printer*, *scanner*.
- b. *Software*: sebuah perangkat lunak yang memiliki tujuan untuk memproses sebuah data dengan menggunakan bantuan dari prangkat keras.
- c. *Database*: sebuah kumpulan data atau *file* yang berisi tabel dan biasanya saling berhubungan (relasional).
- d. *Network*: penghubung yang memungkinkan perangkat elektronik terhubung ke jaringan dengan tujuan dapat saling berkomunikasi antara satu komputer dengan komputer lain dalam berbagi sumber daya.

- e. *Procedures*: sebuah intruksi mengenai cara menggabungkan sebuah komponen-komponen di atas untuk memproses informasi dan menghasilkan *output* yang diharapkan oleh pengguna.
- f. *People*: istilah lain dikenal juga dengan nama *Brainware* atau individu yang berinteraksi secara langsung dengan perangkat keras (*hardware*) dan perangkat lunak (*software*) serta menggunakan hasil *output*-nya.

2.4 Data

Pengertian data adalah sebuah catatan berisi kumpulan fakta dan benar nilainya yang telah diketahui dan dianggap. Diketahui artinya sesuatu yang telah terjadi dan memiliki bukti (fakta). Data juga didefinisikan sebagai sekumpulan informasi yang diperoleh dari hasil sebuah pengamatan suatu objek. Menurut Wahyu Supriyanto dan Ahmad Muhsin (2008) data adalah bahan baku informasi, dapat didefinisikan sebagai simbol-simbol yang mewakili kuantitas, fakta, tindakan, benda dan sebagainya. Data adalah sesuatu yang mewakili tentang objek dan peristiwa yang memiliki nilai dan arti yang penting bagi *user* (Hoffer, Prescott dan Mcfadden, 2005, p5). Sedangkan menurut Connolly dan Begg (2010, p36) menjelaskan bahwa data adalah jembatan yang menghubungkan antara manusia dengan mesin.

Dengan demikian data dapat disimpulkan suatu penghubung antara komponen manusia dan komponen mesin yang sifatnya sangat penting.

2.5 Database

Perkembangan jaman yang pesat kini membuat semakin meningkatnya minat pengguna dalam menggunakan sistem untuk aktivitas sehari-harinya, tentu itu sangat dibutuhkan sebuah *database* dalam menyimpan data-data dalam jumlah yang sangat besar. Pada umumnya

database digunakan pada perusahaan untuk menyimpan data-data seperti penjualan, pembelian, produk pada sebuah barang, hingga data tentang konsumen mereka.

Menurut Gary B. Shelly dan Misty E. Vermaat (2008), *database* adalah sebuah koleksi data yang diatur dan dikelola untuk mempermudah pengguna dalam menggunakannya. Sedangkan menurut Connolly dan Begg (2010, p65), mendeskripsikan *database* merupakan kumpulan dari data yang terkait secara logis dan merupakan deskripsi dari data yang dibangun agar dapat memenuhi kebutuhan perusahaan.

Definisi tentang *database* yang dijelaskan oleh para ahli diatas, disimpulkan bahwa *database* merupakan sebuah data yang saling terhubung dan memiliki manfaat memudahkan dalam mencari sebuah informasi kebutuhan dari organisasi atau perusahaan dan penggunanya.

2.6 Database Management System (DBMS)

Sistem Manajemen Basis Data atau *System Database Management* (DBMS) merupakan perangkat lunak yang digunakan untuk menciptakan sistem basis data yang telah komputerisasi. Tujuan dari DBMS yaitu agar dapat membantu dalam pemeliharaan dan pengelolaan dari kumpulan data yang dimiliki. DBMS merupakan jembatan antara pengguna dengan basis data.

Menurut pendapat yang dikemukakan oleh Connolly (2010), DBMS diartikan sebagai aplikasi yang digunakan untuk membuat, mengelola, dan mendeskripsikan *database* itu sendiri dan menurut Hoffer (2005, p49), DBMS merupakan sebuah perangkat lunak yang menggunakan metode yang sistematis untuk membuat baru, menghapus, menyimpan, mengambil, dan memperbarui suatu data didalam *database*.

Ada terdapat 2 (dua) bagian utama dalam DBMS yaitu :

a. DDL (*Data Definition Language*)

Menurut Connolly dan Begg (2010, p92) DLL merupakan bahasa yang digunakan untuk *database administrator* dan mengizinkan untuk mendeskripsikan suatu entitas, dan atribut yang saling terhubung dalam aplikasi yang saling bersamaan.

b. DML (*Data Manipulation Language*)

Data Manipulation Language adalah sebuah bahasa yang digunakan untuk memanipulasi data serta menyediakan fasilitas pengoperasian data untuk akses data. Connolly dan Begg (2010, p92).

Connolly dan Begg (2010, p66), menjelaskan bahwa DBMS secara umum dapat memberikan beberapa fasilitas yaitu:

1. Dapat mengizinkan pengguna untuk mengontrol dan mendefinisikan *database* dalam *Data Definition Language* (DDL).
2. Dapat memudahkan pengguna dalam memasukan, menghapus, menambahkan, mengubah suatu data dari *database* ke *Data Manipulation Language* (DML).

Dari pengertian yang telah dijelaskan diatas kita dapat menarik sebuah kesimpulan bahwa DBMS adalah sebuah sistem perangkat lunak yang memungkinkan pengguna agar dapat mengontrol, memelihara, menyimpan, menghapus, merubah, dan mengambil data dari *database*.

2.7 Website

Pengertian *website* menurut Saputoro (2007:1) adalah kumpulan dari halaman-halaman yang berguna untuk menampilkan informasi berupa teks, animasi, suara, gambar diam ataupun gerak, hingga data gabungan dari semuanya yang membentuk satu rangkaian yang saling terkait dengan jaringan-jaringan halaman (*Hyperlink*).

Dalam menyediakan sebuah *website*, menurut Saputoro (2007:1-2) terdapat unsur-unsur pendukungnya antara lain:

1. Nama *Domain*

Nama *Domain* atau biasa kita sebut dengan URL (*Uniform Resource Locator*) adalah sebuah nama alamat yang identik dan unik di dunia *internet* dan digunakan untuk mengidentifikasi situs *web*. Nama *domain* dapat digunakan dengan menyewa sesuai dengan harga dan jenisnya. Jenis-jenis *domain* juga memiliki fungsi dan kepentingan yang berbeda-beda seperti contoh nama *domain* yang berakhiran *com*, *net*, *info*, *biz*, *name*, dan *ws*. Khusus untuk Negara Indonesia terdapat domain seperti berikut:

- *co.id* : Badan usaha yang memiliki badan hukum yang sah.
- *ac.id* : Lembaga pendidikan RI
- *mil.id* : Militer dan pertahanan RI
- *go.id* : Lembaga pemerintahan RI
- *or.id* : Organisasi
- *web.id* : Untuk yang sifatnya perorangan atau pribadi
- *sch.id* : Lembaga Pendidikan
- *war.net.id* : Industri warung *internet* yang ada di Indonesia

2. *Web Hosting*

Web Hosting menjadi tempat yang berfungsi menyimpan *file-file*, data, gambar, video, dan lain-lain yang akan ditampilkan pada situs *web*. *Web hosting* memiliki kapasitas penyimpanan dalam ukuran MB (*Mega Byte*) hingga GB (*Giga Byte*) tergantung pengguna pengguna dalam menyewa sesuai dengan kebutuhannya.

3. *Scripts Program*

Scripts Program biasa kita sebut juga dengan Bahasa pemrograman, yaitu Bahasa yang digunakan untuk menerjemakan perintah dalam setiap membuat *website*.

Jenis-jenis Bahasa pemrograman yang dipakai para *programmer website* antara lain; HTML (*Hyper Text Markup Language*), ASP (*Active Server Pages*), JSP (*Java Server Pages*), *Java Scripts*, PHP (*Programming Hypertext Processor*), dan sebagainya.

HTML merupakan bahasa dasar *website*. Dengan adanya tambahan bahasa program seperti PHP, JSP, XML, dan lainnya akan menjadikan pengaturan *website* itu sendiri menjadi interaktif dan dinamis.

4. *Desain Website*

Desain Website sangat menentukan kualitas dari suatu situs *website* karena pengaruh tampilan akan mengundang pengguna untuk melihat isi *website* tersebut. Beberapa program yang tersedia untuk membangun desain situs *web* seperti *Adobe Photoshop* untuk membuat gambar atau animasi, *Adobe Dreamweaver* untuk desain tampilan *web*, *Microsoft FrontPage*, dan lain-lain.

5. *Program transfer data ke pusat data*

Program transfer data ke pusat data adalah berupa *file-file* yang terpisah seperti Bahasa program, data informasi teks, gambar dan lain sebagainya, dan dapat terpublikasi secara lengkap pada situs *web* setelah melakukan pada *browser*.

6. *Publikasi Website*

Sebuah *website* tidak memiliki arti apa-apa jika tidak ada kunjungan oleh pengguna *internet*. Efektif atau tidaknya sebuah situs *website* tergantung banyaknya komentar dan kunjungan terhadap *website* itu sendiri. Sehingga sangat diperlukannya publikasi dan promosi.

Kesimpulan yang penulis dapatkan mengenai *website* adalah sebuah penyebaran informasi melalui jaringan *internet*, dan *website* merupakan kumpulan dari banyaknya dokumen yang tersebar di beberapa *server* yang berada di seluruh penjuru dunia dan terhubung menjadi satu jaringan yang dinamakan *internet*.

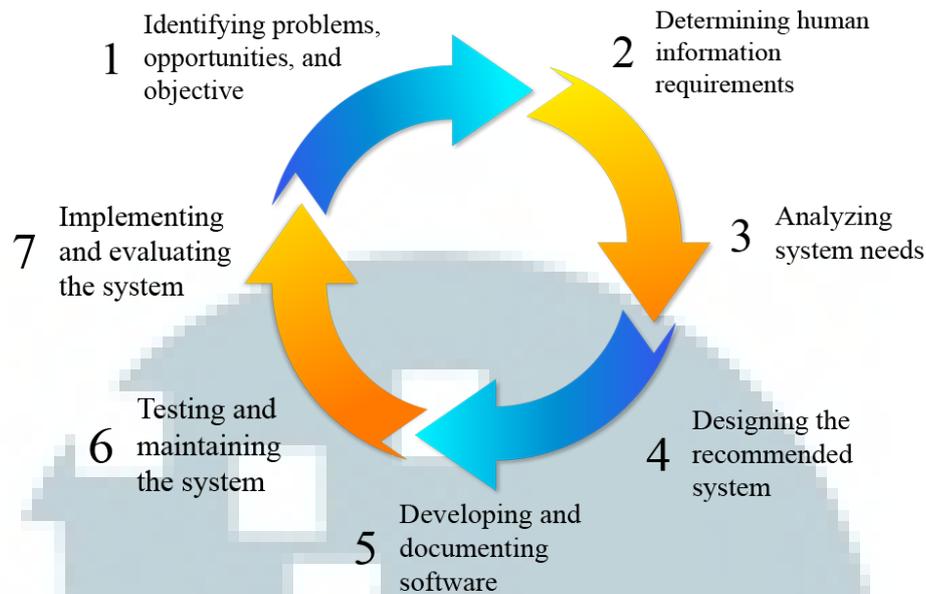
2.8 Analisis Sistem dan *System Development Life Cycle* (SDLC)

Menurut Whitten dan Bentley (2007) mendefinisikan tentang analisis sistem sebagai bentuk proses untuk menyelesaikan permasalahan bisnis sebagai bagian dari perbaikan dan menentukan kebutuhan bisnis dan prioritas sebagai solusinya.

Sehingga dapat ditarik kesimpulan bahwa analisis sistem sangat dibutuhkan dalam menyelesaikan suatu permasalahan dalam perusahaan atau organisasi dengan mengambil informasi yang diinginkan untuk menciptakan solusi melalui sistem tersebut.

SDLC (*System Development Life Cycle*) merupakan tahapan-tahapan pekerjaan yang dilakukan oleh sistem analis dan *programmer* dalam membangun sebuah sistem informasi. Menurut Kendall (2006), *System Development Life Cycle* adalah pendekatan yang bertahap untuk melakukan analisis dan membangun perancangan sistem dengan menggunakan siklus yang spesifik terhadap kegiatan pengguna.

U
M
N



Gambar 2.1 Tujuh Fase *System Development Life Cycle*

(Kendall, 2006:8)

Berikut ini adalah ketujuh fase *System Development Life Cycle* menurut Kendall:

1. *Identifying problems, opportunities, and objective*

Identifikasi Masalah, Kesempatan, dan Sasaran. Pada fase yang pertama ini, seorang sistem analis harus terlibat dalam mengidentifikasi masalah, kesempatan, dan sasaran. Langkah ini merupakan sesuatu yang sangat kritis untuk menentukan kesuksesan dari suatu proyek, karena tidak ada seorangpun yang berkeinginan adanya pemborosan waktu dalam menangani sebuah masalah. Selain masalah hal yang perlu diidentifikasi adalah kesempatan. Kesempatan merupakan situasi agar seorang analis dapat berbuat yang lebih baik dengan menggunakan sistem informasi komputer. Hal yang tidak kalah pentingnya yaitu mengidentifikasi sasaran, karena analis harus menemukan langkah pertama dimana proyek ini dapat dijalankan.

Aktivitas pada fase pertama ini terdiri dari adanya wawancara mengenai manajemen pengguna, merangkum dan mengumpulkan informasi yang didapatkan,

menghitung perkiraan jangkauan proyek, hingga mendokumentasikan hasilnya. Sehingga hasil dari fase ini adalah suatu laporan yang terdiri atas definisi masalah dan rangkuman sasaran (Kendall, 2006:9).

2. *Determining human information requirements*

Menentukan Kebutuhan Informasi Pengguna. Masuk pada fase yang kedua selanjutnya seorang analis menentukan kebutuhan informasi pengguna dengan menggunakan bantuan peralatan yang mudah dipahami. Dalam fase ini merancang bagaimana seorang pemakai dapat berinteraksi dengan baik pada pekerjaannya dengan sistem informasi yang ada. Tentu seorang sistem analis akan menggunakan metode yang interaktif seperti wawancara, kuesioner, atau pengambilan sampel untuk mengetahui lebih dalam tentang kebutuhan informasi pengguna.

3. *Analyzing system needs*

Analisis Kebutuhan Sistem sangat dapat membantu seorang sistem analis dalam menentukan kebutuhan suatu sistem. Seorang sistem analis dapat menggunakan *tools* seperti *data flow diagram* (DFD) yang akan menjelaskan tentang *input*, *process*, dan *output* dari diagram aktivitas sistem yang menampilkan rangkaian sebuah kejadian. Dalam fase ini seorang sistem analis juga menganalisis dalam pembuatan struktur keputusan. Struktur keputusan berupa sebuah kondisi seperti kondisi alternatif, dan aturan yang dapat digunakan untuk mengambil keputusan secara tepat. Terdapat tiga metode umum dalam pembuatan struktur keputusan. Antar lain : *structured English*, tabel keputusan (*decision table*), dan pohon keputusan (*decision tree*) (Kendall, 2006:10).

4. *Designing the recommended system*

Mendesain Sistem yang Direkomendasikan. Untuk mendesain sebuah sistem, seorang sistem analis tentu harus menggunakan informasi yang telah diperoleh dari

fase-fase yang sebelumnya untuk dapat menyelesaikan rancangan logika yang digunakan pada sistem informasi. Seorang sistem analis kemudian merancang prosedur untuk pengguna agar bias membantu secara akurat dalam mengolah data menjadi informasi yang benar. Inti yang didapat pada fase ini yaitu harus merancang sebuah *output*, *input*, struktur *file*, program *file*, program, prosedur, perangkat keras dan perangkat lunak yang diperlukan untuk mendukung sistem informasi (Kendall, 2006:11).

5. *Developing and documenting software*

Mengembangkan dan Mendokumentasikan Perangkat Lunak. Pada fase yang kelima ini, seorang sistem analis sudah bekerja sama dengan seorang *programmer* dalam mengembangkan perangkat lunak sesuai yang diinginkan dari analisis yang telah dilakukan dalam kebutuhan sistem. Selama fase ini, seorang sistem analis juga bekerja dengan pemakai (*user*) dalam mendokumentasikan perangkat lunak, melalui prosedur *manual*, *online help*, hingga *file* seperti *Read Me* yang disisipkan pada perangkat lunak yang baru. *Programmer* memiliki peranan yang sangat penting dalam merancang sebuah program. Untuk menjamin kualitasnya, seorang *programmer* menyediakan salah satu desain yang menjelaskan kelengkapan sebuah program kepada tim pembangun program (Kendall, 2006:11).

6. *Testing and Maintaining the system*

Uji Coba dan Perawatan Sistem. Sebelum sistem informasi digunakan, tentunya hal yang terlebih dulu dilakukan adalah *testing*. *Testing* memiliki tujuan yaitu untuk mengurangi banyaknya masalah yang fatal sebelum sistem digunakan oleh pengguna pada umumnya. Rangkaian uji coba pertama dijalankan dengan menggunakan sampel data yang kemudian membandingkan dengan data sebenarnya dari sistem yang telah ada sebelumnya. Karena sifatnya *Maintaining the system* tentu pada fase

ini akan selalu dibawa kepada aktivitas keseharian dari sistem informasi yang berjalan. Beberapa perawatan sistem yang akan akan terjadi seperti (*program updates*), yang dapat dilakukan secara otomatis. (Kendall, 2006:11).

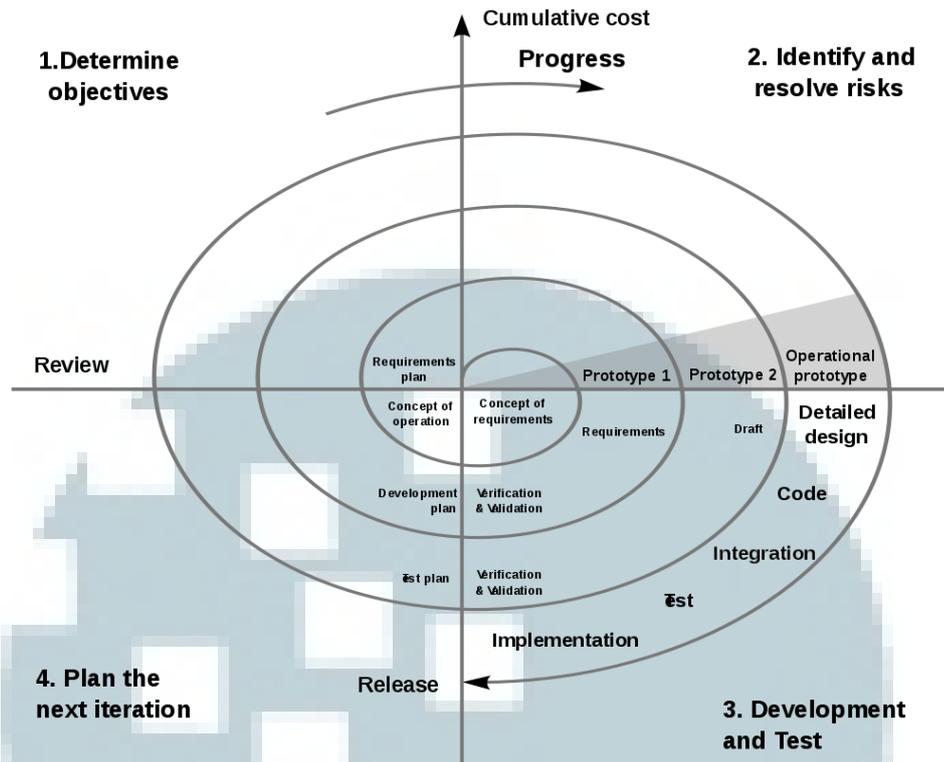
7. *Implementing and evaluating the system*

Pada fase yang terakhir yaitu mengimplementasikan dan mengevaluasi sistem, seorang sistem analis akan mengajarkan cara pakai dalam menggunakan sistem informasi yang telah dibuat. Proses implementasi sendiri terdiri atas pengkonversian *file* dari format yang lama ke format yang baru, membangun sebuah *database*, *install* kebutuhan, hingga membawa sistem yang baru kepada lembaga produksi. Evaluasi terdiri atas penilaian sejauh mana suatu sistem yang telah dibangun dan seberapa bagus sistem yang telah dijalankan (Kendall, 2006:11).

2.9 SDLC *Spiral Model*

Menurut Boehm (1986) spiral model digunakan untuk proyek yang besar karena proses pengembangan aplikasinya menggabungkan antara komponen *prototyping* dan desain, dalam upaya memperoleh manfaat dari konsep *top-down* dan *bottom-up*. *Spiral model* juga biasa disebut dengan *SDLC Spiral Lifecycle* karena secara umum pada model ini dapat mengulang tahapan yang telah dilalui seiring berjalannya proses.

U
M
N



Gambar 2.2 Spiral Model

(Sumber: Boehm, 1986)

Terdapat tahapan-tahapan dalam metode spiral ini, yaitu:

a. *Determine Object*

Adanya penentuan tujuan pada objek, sistem akan dijelaskan sangat detail pada tahap ini.

b. *Identify and Resolve Risk*

Mulai mengidentifikasi pemecahan masalah, dan proses desain pada sistem. Pada tahap ini merupakan tahapan yang paling terpenting dalam model ini, karena merupakan langkah awal dalam penanggulangan masalah secara menyeluruh. Sehingga, manfaatnya akan dapat menghemat biaya dalam pengembangan sistem.

c. *Development and Test*

Pada tahap ini akan dilakukan pengembangan mulai dari *coding*, mengintegrasikan sistem, hingga *testing*.

d. *Plan the next iteration*

Pada tahap ini yang akan dilakukan adalah melakukan tahap pengulangan yang bertujuan untuk memperbaiki sistem yang telah dibangun. Sehingga, sistem semakin matang dan siap pakai sesuai dengan kebutuhannya.

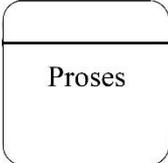
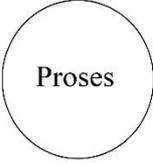
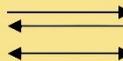
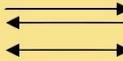
Dengan demikian, SDLC *spiral model* ini merupakan sebuah proses yang berulang-ulang seiring dengan jalannya suatu proses hingga mendapatkan hasil yang maksimal sesuai yang diinginkan.

2.10 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) berfokus kepada aliran data yang masuk dan keluar dari sebuah sistem pemrosesan data (Kendall, 2008). Beberapa keuntungan dalam menggunakan DFD adalah sebagai berikut:

- a. Bebas dalam menjalankan sebuah implementasi secara teknis dari sebuah sistem yang terlalu dini.
- b. Pemahamannya lebih kepada interaksi atau keterkaitan antar sebuah sistem dan subsistem yang saling berhubungan.
- c. Melakukan ulasan atau analisis pada sistem mengenai data yang diolah.

Menurut Bentley dan Whitten (2007) *data flow diagram* merupakan suatu alat yang menggambarkan aliran data melalui sistem dan pengolahan yang dilakukan oleh sistem. Sedangkan menurut Yourdon (1989; p.139-173) *Data Flow Diagram* (DFD) adalah suatu hal yang sangat penting bagi seorang sistem analis. Dengan menggunakan DFD, sistem analis akan dapat memahami aliran-aliran data dalam sebuah system.

Gane/Sarson	Yourdon/De Marco	Penjelasan
		Dapat menggambarkan tentang asal atau tujuan data di luar sebuah sistem
		Menggambarkan tentang entitas atau suatu proses dimana aliran datanya masuk kemudian ditransformasikan ke aliran data keluar
		Menggambarkan tentang aliran data berjalan
		Menggambarkan tentang suatu data disimpan

Gambar 2.3 Simbol DFD

(Sumber: Alan Dennis and Barbara Haley Wixom, 2008)

Pada gambar diatas terdapat dua gambar standard DFD yang pertama (sebelah kiri) diciptakan oleh Gane dan Sarson, untuk gambar standard yang kedua diciptakan oleh DeMarco dan Yourdon. (Hoffer, George, & Valacich, 2008)

Pada *Data Flow Diagram* terbagi menjadi :

- a. Diagram Konteks : Menggambarkan suatu lingkaran yang umum dan dapat mewakili seluruh proses yang terdapat didalam suatu sistem. Diagram konteks merupakan tingkatan tertinggi dalam DFD dan biasanya ditandai dengan nomor 0 (nol). Pada diagram ini sama sekali tidak membuat penyimpanan data dan sangat tampak sederhana untuk diciptakan.

b. Diagram Nol (diagram level -1)

Menggambarkan suatu lingkaran besar namun sudah mewakili lingkaran-lingkaran kecil seperti sub sistem yang ada di dalamnya. Di dalam diagram ini memuat untuk penyimpanan sebuah data.

c. Diagram Rinci

Merupakan uraian secara rinci dan lebih detail dari diagram 0 (nol) yang berisi proses-proses yang menggambarkan bab dari sub sistem diagram 0 (nol).

Jadi dapat disimpulkan *data flow diagram* merupakan sebuah alat bantu dalam menggambarkan atau menjelaskan sistem yang sedang secara logika yang bertujuan untuk memudahkan pengguna dalam memahami apa yang akan dikerjakan atau dikembangkan oleh sistem secara terstruktur dan jelas.

Berikut merupakan simbol-simbol yang digunakan dalam *Data Flow Diagram* (DFD):

1. Entitas (*Entity*)

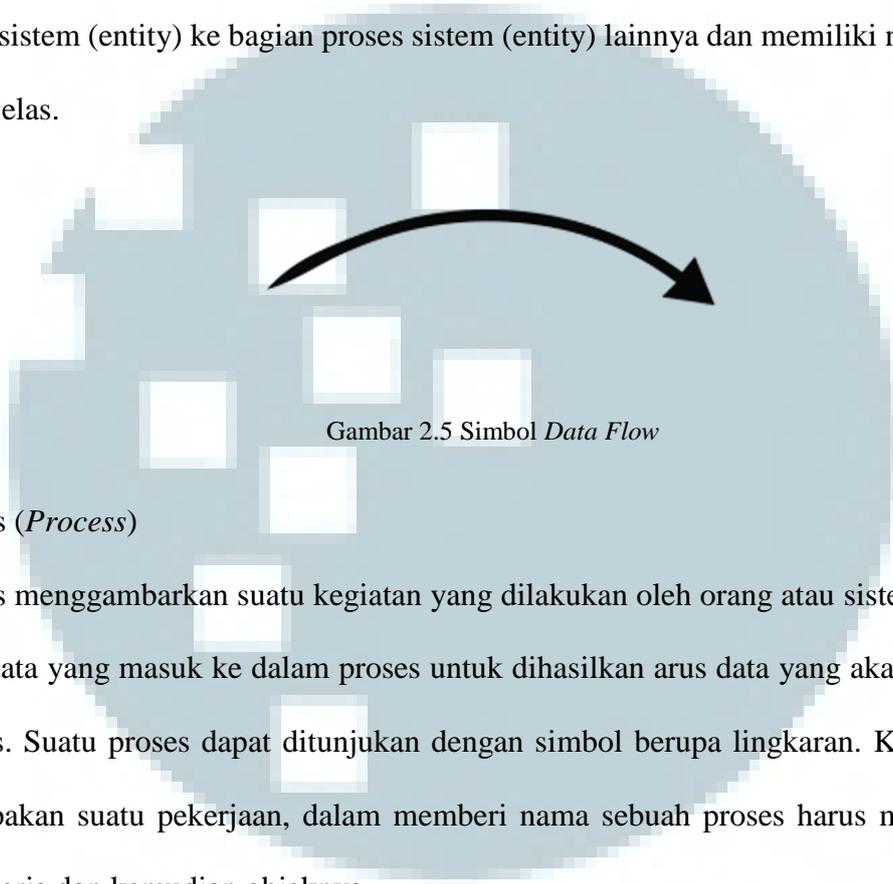
Entitas merupakan sebuah kelas dari orang, objek, dan kejadian atau sebuah konsep yang diperlukan untuk menyimpan data. Simbol entitas sebagai penanda suatu data berasal atau dapat juga suatu data berakhir. Entitas juga merupakan tujuan dari aliran sebuah data.



Gambar 2.4 Simbol Entitas

2. Arus Data (*Data Flow*)

Simbolnya berupa tanda panah yang memiliki arti untuk mempresentasikan adanya sebuah aliran data yang digunakan untuk menunjukkan arus dari data atau informasi dari suatu sistem (*entity*) ke bagian proses sistem (*entity*) lainnya dan memiliki nama dan arti yang jelas.



Gambar 2.5 Simbol *Data Flow*

3. Proses (*Process*)

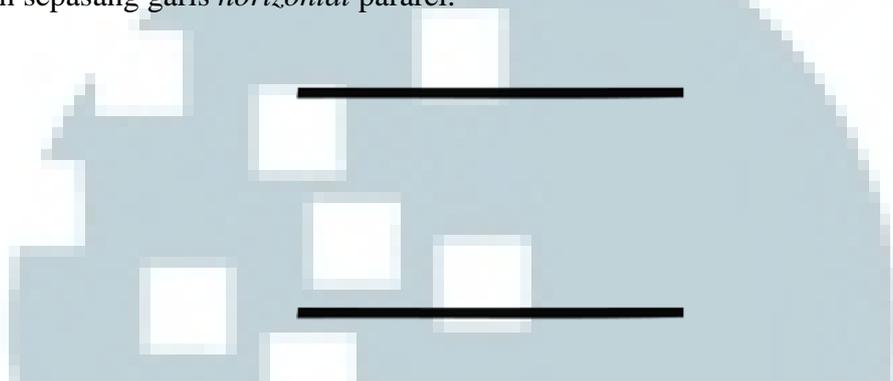
Proses menggambarkan suatu kegiatan yang dilakukan oleh orang atau sistem dari suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Suatu proses dapat ditunjukkan dengan simbol berupa lingkaran. Karena proses merupakan suatu pekerjaan, dalam memberi nama sebuah proses harus menggunakan kata kerja dan kemudian objeknya.



Gambar 2.6 Simbol Proses

4. Simpanan Data (*Data Store*)

Simpanan data merupakan simpanan dari data berupa suatu *file* atau *database* di sistem komputer, agenda atau buku. Dengan kata lain, simpanan data merupakan sebuah tempat dikumpulkannya data-data yang kemudian dikelompokkan sesuai jenisnya, sehingga mengetahui informasi untuk suatu proses tertentu. Simpanan data dapat disimbolkan dengan sepasang garis *horizontal* paralel.



Gambar 2.7 Simbol *Data Store*

2.11 *Entity Relationship Diagram (ERD)*

Menurut salah satu dari parah ahli tentang ERD, Brandy dan Loonam (2010), *entity relationship diagram* merupakan teknik yang digunakan untuk memodelkan kebutuhan-kebutuhan data dari suatu organisasi, dan biasanya digunakan oleh sitem analis dalam tahapan analisis persyaratan pengembangan sistem proyek. Sedangkan menurut Kendall (2008), *entity relationship model* adalah suatu representasi dari grafik suatu *entity relationship model*. Berikut adalah simbol-simbol yang digunakan dalam membangun sebuah ERD:

a. Entitas (*Entity*)

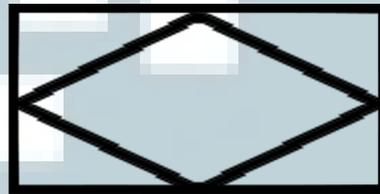
Entitas didefinisikan sebagai sebuah kelas, seorang, atau benda. Suatu konsep yang diperlukan dalam menyimpan sebuah data.



Gambar 2.8 Simbol Entitas

b. Entitas Asosiatif (*Associative Entity*)

Entitas Asosiatif terbentuk dari hubungan banyak ke banyak antara entitas-entitas yang lain. Simbol dari entitas asosiatif berupa sebuah persegi panjang dengan jajar genjang di dalamnya.



Gambar 2.9 Simbol Entitas Asosiatif

c. Hubungan (*Relationship*)

Biasa disebut juga dengan relasi yaitu sebuah peristiwa yang menghubungkan sebuah entitas pada entitas yang lain.



Gambar 2.10 Simbol sebuah *relationship*

d. Atribut (*Attribute*)

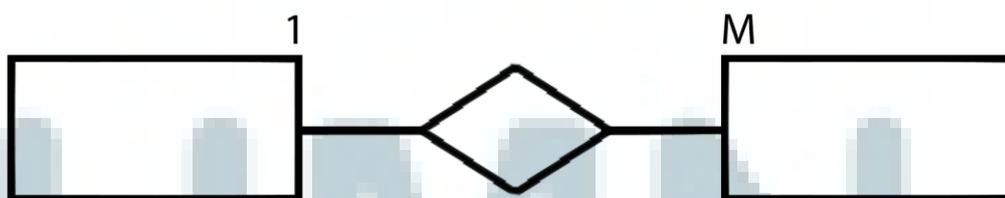
Atribut merupakan suatu penjelasan dari sebuah karakteristik entitas. Setiap entitas harus memiliki atribut yang unik untuk mengenali data dan jenisnya. Atribut yang dimaksud disebut dengan *key*, terdapat beberapa jenis *key* seperti berikut ini:

- *Primary key*, merupakan suatu atribut yang tidak hanya dapat mengidentifikasi secara unik suatu kejadian yang spesifik, akan tetapi juga dapat mewakili setiap kejadian dari suatu *entity*.
- *Secondary key*, disebut juga *alternate key* karena *key* yang tidak terpilih menjadi *primary key*.
- *Composite key*, suatu kelompok dari atribut yang dapat mengidentifikasi suatu entitas.
- *Foreign key*, merupakan suatu atribut yang melengkapi satu *relationship* yang menunjukkan keinduknya.

Menurut Romney terdapat 3 (tiga) macam hubungan pada *entity relationship diagram*, yaitu:

a. *One to many (1:N) relationship*:

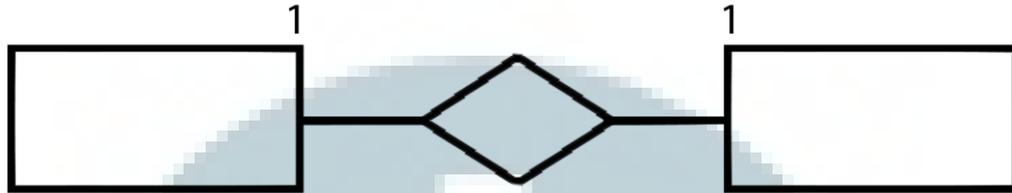
Sebuah hubungan antara 2 (dua) entitas yang dimana satu entitas dalam hubungan ini dapat berasosiasi dengan banyak entitas yang lain dalam sebuah hubungan.



Gambar 2.11 *One to many*

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

Sebuah hubungan antara 2 (dua) entitas dimana satu entitas hanya dapat berasosiasi dengan satu entitas yang lain dalam sebuah hubungan.



Gambar 2.12 *One to one*

c. *Many to many (M:N) relationship:*

Hubungan antara 2 (dua) entitas dimana setiap entitas dapat berasosiasi dengan banyak entitas lainnya.



Gambar 2.13 *Many to many*

Dalam membuat sebuah ERD, tentu memiliki beberapa tahapan yang wajib diperhatikan.

Tahapan-tahapan tersebut, yaitu:

- a. Identifikasi dan memilih seluruh himpunan dari entitas yang terlibat.
- b. Menentukan *atribut key* dari masing-masing himpunan entitas.
- c. Identifikasi dan menetapkan seluruh himpunan relasi antara himpunan entitas yang ada dengan *foreign key*.

- d. Menentukan kardinalitas relasi untuk himpunan, dan melengkapi sebuah himpunan entitas dan relasi dengan atribut deskriptif.

2.12 Normalisasi

Normalisasi merupakan suatu teknik analisis data dalam mengelompokkan atribut-atribut sehingga terbentuk relasi yang berorientasi pada objek dengan sangat baik. Normalisasi juga sebuah proses untuk menyederhanakan *relationship* antar elemen data. Menurut pendapat Kendal dan Kendal (2008; p.492) normalisasi adalah transformasi pandangan dari pengguna yang kompleks dan menyimpan data untuk satu set yang lebih kecil struktur data yang stabil sehingga struktur data normalisasi lebih mudah dipelihara daripada struktur yang kompleks.

Dalam pembuatan normalisasi terdapat beberapa tahapan-tahapan penting yaitu:

1. Bentuk Tidak Normal (*Unnormalized Form*)

Suatu bentuk tidak normal yang merupakan kumpulan data yang akan direkam, tidak ada keharusan untuk mengikuti format tertentu, data bisa tidak lengkap atau bahkan terduplikasi. Data dikumpulkan apa adanya sesuai dengan saat input.

2. Bentuk Normal Kesatu (1NF/ *First Normal Form*)

Sebuah entitas yang atributnya tidak lebih dari satu nilai untuk satu entitas. *Repeating group* harus dihilangkan agar menjadi tunggal yang saling berinteraksi diantara setiap baris kolom pada suatu tabel.

3. Bentuk Normal Kedua (2NF/ *Second Normal Form*)

Sebuah bentuk normal kedua yang bentuk datanya telah memenuhi kriteria bentuk normal kesatu. Atribut yang bukan kunci harus bergantung pada normal kedua yang harus sudah ditentukan oleh kunci *field* atau dengan kata lain entitas yang atributnya *non-primary-key* tergantung pada *primary key*.

4. Bentuk Normal Ketiga (3NF/ *Thrid Normal Form*)

Suatu relasi yang harus pada bentuk normal kedua dan semua atributnya bukan primer atau entitas yang atribut *non-primary-key* tidak tergantung pada *non-primary-key* lainnya.

2.13 Pengertian PHP (*Hypertext Preprocessor*)

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada saat itu, php merupakan sebuah kumpulan skrip yang digunakan untuk mengolah data dari *web*. Pada saat ini php dikenal sebagai *Hypertext preprocessor* yang dipakai untuk membangun sebuah aplikasi *web* yang mendukung *database*.

Sedangkan menurut Kevin Yank (2012), PHP merupakan *server-side scripting language* yang dapat digunakan pengguna untuk memasukan macam-macam intruksi ke dalam halaman *web* yang nantinya akan dieksekusi oleh *software* pada umumnya seperti *apache* sebelum halaman tersebut dikirim ke *browser* yang memintanya.

2.14 Apache

Apache merupakan salah satu jenis dari *web server*. *Web server* merupakan *server* yang menangani komunikasi data pada internet melalui jalur *world wide web* (www) atau *Hypertext Markup Language* (http). Selain itu, hampir semua *bundle software* untuk pengembangan *web* yang menggunakan Apache sebagai *web server* seperti contohnya *XAMPP*, *EasyPHP*, *WAMPP* dan sebagainya. Karena konfigurasi yang mudah menjadikan apache digunakan untuk konten yang berbasis PHP. (sumber: <http://www.apache.org/>)

2.15 Crowdsourcing

Isilah *crowdsourcing* mulai dipopulerkan oleh Jeff Howe pada tahun 2006, melalui artikelnya yang dimuat pada *Wired Magazine*. Beliau menjelaskan bahwa *crowdsourcing* dapat

diartikan sebagai bentuk model dalam pemecahan masalah dan pembuatan sebuah produk secara *online* dan mengalihkan tugas yang sebelumnya dikerjakan oleh para karyawan ke khalayak ramai. Dalam buku berjudul *Crowdsourcing as a model for problem solving an introduction and cases* karya Daren C. Brabham (2008), bahwa *crowdsourcing* memiliki manfaat untuk pemecahan masalah di berbagai bidang yang sudah terbukti dengan keberhasilannya dengan pemanfaatan kolaborasi masal melalui *internet*.

Jadi dapat disimpulkan bahwa *crowdsourcing* adalah sebuah bentuk kegiatan guna mendapatkan pelayanan yang dibutuhkan, seperti ide, atau konsep yang terkait dengan meminta kontribusi dari orang banyak dengan pemanfaatan media *internet* dan sistem informasi. Kolaborasi masal oleh para pengguna juga dapat hadir dalam bentuk seperti evaluasi (*review, vote, tag*), berbagi (benda, ilmu pengetahuan), kemudian membangun semacam *software*, dan sistem. Kolaborasi tersebut dapat hadir dalam bentuk implisit dimana *crowdsourcing* menjadi produk tambahan pada sistem lain dari aktivitas pada sistem yang diterapkan pada usaha digitalisasi konten melalui koreksi ejaan dalam *search engine*.

2.16 Trotoar

Menurut Danisworo (1991), trotoar merupakan jalur khusus untuk pejalan kaki yang dibuat terpisah dari jalur kendaraan umum, dan biasanya terletak bersebelahan atau berdekatan. Pengertian tentang trotoar ini juga sesuai dengan Ogden (1996) yang menyatakan, *footpath* atau *side walk* memiliki arti sebuah jalur khusus untuk pejalan kaki yang mengambil bagian dari jalan kendaraan atau jalur yang terpisah dan untuk pejalan kaki saja, tetapi sebagian ada jalur pejalan kaki yang digunakan bersama-sama dengan pengguna sepeda. Sedangkan menurut Shirvani (1985) mendefinisikan bahwa trotoar merupakan elemen yang penting dalam perancangan kota, yaitu membentuk hubungan antara aktivitas-aktivitas pada suatu lokasi.

Idealnya fasilitas pejalan kaki berupa trotoar ditempatkan di:

1. Daerah-daerah perkotaan yang tingkat kepadatan penduduknya tinggi.
2. Jalan yang memiliki rute angkutan umum yang tetap.
3. Daerah yang memiliki aktivitas berlanjut yang tinggi, seperti contoh pusat perbelanjaan, perkotaan, atau tempat terdapatnya lokasi pariwisata.
4. Lokasi yang memiliki kebutuhan yang tinggi dengan jangka waktu yang pendek, seperti misalnya stasiun - stasiun, terminal, sekolah, rumah sakit, dan sebagainya.
5. Lokasi yang mempunyai permintaan yang tinggi untuk kegiatan sehari - hari seperti tempat ibadah, tempat olah raga.



UMN