

BAB III

LANDASAN TEORI

Dalam membangun aplikasi ini, terdapat teori-teori ilmu yang terkait yang digunakan untuk membantu penelitian serta menyelesaikan permasalahan yang ada dan berkaitan dengan sistem yang akan dibuat. Tujuannya adalah agar aplikasi ini memiliki pijakan pustaka yang dapat dipertanggungjawabkan.

3.1 Komputer

Komputer adalah sebuah mesin elektronik yang beroperasi di bawah kontrol intruksi yang tersimpan di memori yang dapat menerima data, memanipulasi data berdasarkan aturan tertentu, menghasilkan keluaran dan menyimpan hasil untuk penggunaan di masa depan, komputer diklasifikasikan dalam tujuh kelompok, yaitu komputer pribadi, komputer *mobile*, dan perangkat *mobile*, *game consoles*, server, *mainframes*, super komputer dan komputer yang tertanam. (Shelly dan Rosenblatt, 2011)

Komputer pribadi adalah sebuah komputer yang dapat menggerakkan sendiri seluruh kegiatan *input*, pemrosesan, *output*, dan penyimpanan. Komputer pribadi terdiri dari dua macam, yaitu komputer *desktop* dan komputer *notebook*. Komputer *notebook* dirancang untuk keperluan *mobile*, sedangkan komputer *desktop* dirancang dengan unit sistem, perangkat masukan, perangkat keluaran, dan perangkat lainnya yang diletakkan di atas atau di bawah meja. (Shelly dan Rosenblatt, 2011)

Komputer terdiri dari dua aspek dasar, yaitu aspek teknis dan aspek non-teknis. Aspek teknis terdiri dari *hardware* (perangkat keras), *software* (perangkat

lunak), dan *brainware* (tenaga pelaksana atau pengguna). Beberapa komponen elektronik dan mekanik yang terdapat pada suatu komputer dikenal sebagai *hardware*. (Shelly dan Rosenblatt, 2011)

3.2 Kerusakan Komputer

Masalah pada *hardware* (perangkat keras) umumnya timbul karena usia perangkat, aus, ketidakstabilan tegangan listrik, kecerobohan pemakai, pemakaian yang tidak menurut prosedur, dan lain sebagainya. Perangkat keras yang berpotensi mengalami kerusakan diantaranya : monitor, *hardisk*, *diskdrive*, *card* (seperti *video card* dan *sound card*), *power supply*, RAM, *processor*, kipas, dan *motherboard*. (Supriyanto, 2005)

Metode yang digunakan untuk mencari dan memecahkan masalah pada umumnya terdiri dari dua langkah. Langkah pertama adalah menelusuri ke sumber masalah. Langkah kedua adalah memisahkan masalah dengan menggunakan standar perangkat lunak dan perangkat keras terkait

3.3 PHP (*Hypertext Preprocessor*)

PHP yang merupakan bahasa pemrograman berbasis *web* yang memiliki kemampuan untuk memproses data dinamis. (Abdul, 2008). PHP dikatakan sebuah *server-side embedded script language*, artinya sintaks-sintaks dan perintah yang diberikan akan sepenuhnya dijalankan oleh *server* tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan di *server*. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode PHP untuk mengirimkan

permintaan ke *server*. Ketika menggunakan PHP sebagai *server-side embedded script language* maka *server* akan melakukan hal-hal sebagai berikut :

1. Membaca permintaan dari *client*.
2. Mencari halaman di server.
3. Melakukan instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman.
4. Mengirim kembali halaman tersebut kepada *client* melalui internet atau intranet.

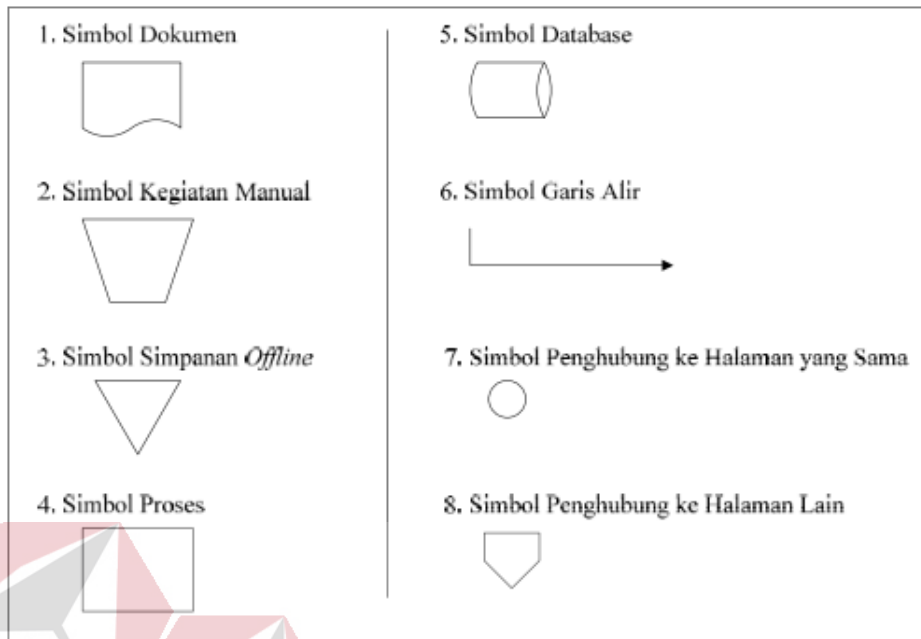
3.4 Website

Website adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. (Hidayat, 2010). Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Sehingga dapat dikatakan bahwa, pengertian *website* adalah kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar, animasi, suara, dan atau gabungan semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman *website* dengan halaman *website* lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

3.5 System Flow

System flow atau bagan alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. *System flow* menunjukkan urutan-urutan dari prosedur yang ada didalam sistem dan menunjukkan apa yang dikerjakan oleh sistem.

Berikut adalah simbol-simbol yang terdapat pada *system flow*



Gambar 3. 1 Simbol-simbol pada *system flow*

1. Simbol dokumen

Menunjukkan dokumen *input* dan *output* baik proses *manual* atau komputer.

2. Simbol kegiatan manual

Menunjukkan kegiatan non-komputer yang dilakukan.

3. Simbol simpanan *offline*

Menunjukkan file non-komputer yang diarsip.

4. Simbol proses

Menunjukkan kegiatan proses dari operasi program komputer.

5. Simbol database

Menunjukkan tempat untuk menyimpan data hasil operasi komputer.

6. Simbol garis air

7. Simbol penghubung di dalam halaman
Menunjukkan penghubung ke halaman yang sama.
8. Simbol penghubung di lain halaman
Menunjukkan penghubung ke beda halaman.
9. Simbol *display*
Menunjukkan respon kepada *user* setelah dilakukan kegiatan.

3.6 Data Flow Diagram (DFD)

Data flow Diagram (DFD) adalah suatu gambaran grafis dari suatu sistem yang menggunakan sejumlah bentuk-bentuk simbol menggambarkan bagaimana data mengalir melalui suatu proses yang saling berkaitan. (McLeod, 2001) *Data Flow Diagram* lebih menekankan pada proses yang terjadi. *Data Flow Diagram* fokus pada aliran data dari dan kedalam sistem serta memproses data tersebut (Kendall dan Kendall, 2003). Penggambaran DFD disusun berdasarkan tingkatan dibawah ini :

1. *Context Diagram*

Context diagram adalah diagram awal yang terdiri dari sebuah proses dan menggambarkan area lingkup proses.

2. Diagram Level 0

Diagram Level 0 adalah diagram yang menggambarkan proses penting dari sistem serta interaksi *entity*, proses, alur data, dan *data source*.

3. Diagram Detail

Diagram detail adalah penguraian dalam proses yang ada terhadap diagram level 0. Diagram ini merupakan diagram yang paling rendah dan tidak dapat diuraikan lagi.

Data Flow Diagram (DFD) memiliki empat komponen, yaitu :

1. *External Entity*

External entity adalah kesatuan di lingkungan sistem yang dapat berupa orang atau sistem yang berada di lingkungan luar sistem yang memberikan masukan (*input*) atau keluaran (*output*). Proses diberi nama untuk menerangkan proses yang dilaksanakan.

2. Proses

Proses adalah komponen yang berfungsi untuk mentransformasikan sistem dari masukan (*input*) menuju ke keluaran (*output*). Proses ini diberi nama untuk menerangkan proses yang dilaksanakan.

3. Alur Data

Aluran data digambarkan dengan anak panah yang menuju ke dalam proses maupun ke luar proses. Alur data dipakai untuk menerangkan perpindahan data atau informasi dari suatu bagian ke bagian lainnya.

4. *Data Source*

Data source adalah tempat pengumpulan data (data tersimpan) yang disimbolkan dengan dua garis *horozontal parallel*. *Data source* perlu diberikan nama untuk menjelaskan nama dari *file*-nya. *Data store* menyangkut dengan penyimpanan data dengan cara terkomputerisasi.

3.7 Desain Sistem

Desain sistem adalah tahap setelah analisis dari siklus pengembangan sistem, pendefinisian dari kebutuhan fungsional dan persiapan untuk rancang

bangun, implementasi, menggambarkan bagaimana suatu sistem dibentuk. (Jogiyanto, 2005)

Desain sistem adalah menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan, tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah intruksi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem. (Jogiyanto, 2005)

3.8 System Development Life Cycle

System Development Life Cycle (SDLC) ialah proses untuk memahami bagaimana sebuah sistem dapat mendukung kebutuhan bisnis dengan merancang suatu sistem, membangun sistem tersebut, dan menyampaikannya kepada pengguna. (Denis, A.; Wixom, B. H.; Tegarden, D.; 2009)

SDLC memiliki empat fase dasar, yaitu *planning*, *analysis*, *design*, dan *implementation* yang dimana tiap fase tersebut terdiri dari serangkaian langkah yang menggunakan cara tertentu dalam menghasilkan tujuan yang ingin dicapai. Berikut ini penjelasan dari tiap fase dari SDLC :

1. Identifikasi (*Planning*)

Fase *planning*, merupakan proses dasar dalam memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana proyek akan dibangun.

2. Analisa (*Analysis*)

Fase *analysis*, merupakan jawaban dari pertanyaan siapa yang akan menggunakan sistem, apa yang akan dilakukan oleh sistem, dan dimana serta kapan sistem tersebut akan digunakan. Pada fase ini perlu menginvestigasi sistem yang telah ada sebelumnya, mengidentifikasi peluang untuk perbaikan dan mengembangkan konsep yang baru untuk sistem yang akan dibuat.

3. Desain (*design*)

Fase *design*, ialah fase untuk menentukan bagaimana sistem akan beroperasi, dalam hal ini antara lain perangkat keras (*hardware*), perangkat lunak (*software*), infrastruktur jaringan (*user interface*), *forms*, dan laporan (basis data dan file yang dibutuhkan aplikasi).

4. Implementasi (*Implementation*)

Fase *implementation*, ialah fase akhir dari SDLC yaitu pada saat sistem selesai dibuat. Implementasi pada fase ini paling banyak mengambil perhatian karena dalam keseluruhan sistem, tahap implementasi, adalah tahap yang paling banyak memakan waktu serta biaya karena mencoba keseluruhan sistem.