



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 Komunikasi

Kata komunikasi berasal dari bahasa Latin *communicatio* yang berarti ‘pemberitahuan’ atau ‘pertukaran pikiran’. Jadi, secara garis besar, dalam suatu proses komunikasi haruslah terdapat unsur-unsur kesamaan makna agar terjadi suatu pertukaran pikiran dan pengertian antara komunikator (penyebarnya) dan komunikan (penerima pesan) (Suprpto, 2009).

Proses komunikasi dapat diartikan sebagai ‘*transfer* informasi’ atau pesan (*message*) dari pengirim pesan sebagai komunikator dan kepada penerima sebagai komunikan. Dalam proses komunikasi tersebut bertujuan untuk mencapai saling pengertian (*mutual understanding*) antara kedua pihak yang terlibat dalam proses komunikasi. Dalam proses komunikasi, komunikator mengirimkan pesan/informasi kepada komunikan sebagai sasaran komunikasi (Suprpto, 2009).

2.2 Antarmuka Manusia dan Komputer

Prinsip dasar sebuah sistem komputer adalah masukan, proses, keluaran (*input, process, output*). Kepada komputer, pengguna

memberikan data masukan, yang biasanya berupa angka maupun deretan karakter. Data masukan ini kemudian diolah (diproses) oleh komputer menjadi keluaran yang diinginkan atau diharapkan pengguna (Santoso,2009).

Ketika seseorang bekerja dengan sebuah komputer, secara disadari atau tidak, dia melakukan interaksi dengan komputer dengan menggunakan cara-cara tertentu. Interaksi terjadi ketika pengguna memasukkan data, yang kemudian akan ditanggapi oleh komputer dengan menampilkan suatu keluaran ke layar tampilan atau ke pencetak. Lewat masukan dan keluaranlah pengguna dan komputer saling berinteraksi. Media interaksi diperlukan agar pengguna dan komputer dapat berinteraksi. Dengan adanya interaksi ini maka pengguna akan merasakan keramahan sistem komputer yang digunakannya (Santoso, 2009).

2.3 *Instant Messenger*

Instant Messaging merupakan bentuk komunikasi pesan singkat antara dua orang atau lebih menggunakan teks yang diketik. Teks dikirim melalui komputer yang terhubung melalui sebuah jaringan, LAN atau Internet (Aripin, 2011).

Teknologi *Instant Messaging* diciptakan untuk menutupi kelemahan *e-mail* yang dinilai kurang efisien dan tidak *real time*. Sebagian besar *software instant Messaging* terdiri dari dua komponen : *Software server* dan

client sebagai program *instant messaging* dapat dioperasikan melalui *web browser* (Aripin, 2011).

2.4 Website

Website adalah metode untuk menampilkan informasi-informasi di internet, baik itu merupakan teks, gambar, video, dan suara yang menghubungkan (*link*) dari dokumen satu dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui *browser* (Yuhefizar, 2008).

Website adalah kumpulan halaman *web* yang saling terhubung dan filenya saling terkait. *Web* terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas, dengan halaman-halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah *homepage* disebut *child page*, yang berisi *hyperlink* ke halaman lain dalam *web* (Agung, 2000).

Untuk dapat membuat sebuah *website* yang baik dan benar, kita perlu memahami beberapa bahasa pemrograman yaitu seperti:

2.4.1 Hyper Text Markup Language (HTML)

HTML yang biasa dikenal sebagai *web page*. HTML adalah sebuah instruksi-instruksi yang dibuat dan kemudian diterjemahkan oleh *browser* yang ada di komputer *client* sehingga isi formasinya dapat ditampilkan secara visual di komputer pengguna (Anamisa & Kustiyahningsih, 2010).

2.4.2 Personal HomePage (PHP)

PHP adalah bahasa skrip yang tertanam dalam HTML untuk dieksekusi, bersifat *server side* (Nugroho, 2006).

PHP adalah bahasa *server-side-scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side-scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di *server* kemudian hasilnya akan dikirimkan ke *browser* dengan *format* HTML (Arief, 2011).

2.4.3 Hyper Text Transfer Protocol (HTTP)

HTTP adalah suatu protokol yang menentukan aturan yang perlu diikuti oleh *web browser* dalam meminta atau mengambil suatu dokumen dan menyediakan dokumen yang diminta *browser* (Anamisa & Kustiyahningsih, 2010).

2.4.4 Javascript

Javascript adalah bahasa yang “*case sensitive*” artinya membedakan penamaan variabel atau fungsi yang menggunakan huruf besar dan huruf kecil (Anamisa & Kustiyahningsih, 2010).

2.4.5 Cascading Style Sheet (CSS)

CSS (*Cascading Style Sheet*) digunakan dalam kode HTML untuk menciptakan suatu kumpulan *style* yang dapat digunakan untuk memperluas kemampuan HTML, sebagai contoh kode HTML murni (tidak

menggunakan CSS) tidak memungkinkan untuk mengatur ukuran font yang diterapkan pada setiap sel (Sutarman, 2009).

2.4.6 *PhpMyAdmin*

PhpMyAdmin adalah suatu aplikasi GUI (*Graphical User Interface*) yang digunakan untuk mengelola *database* MySQL (Arief, 2011).



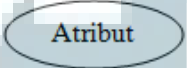

2.5 *Entity Relationship Diagram (ERD)*

ERD (*Entity Relationship Diagram*) adalah merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh *System Analyst* dalam tahap analisis persyaratan proyek pengembangan sistem (Brady & Loonam, 2010).

Tabel 2.1 menjelaskan notasi-notasi yang dapat digunakan pada *entity relationship diagram*.

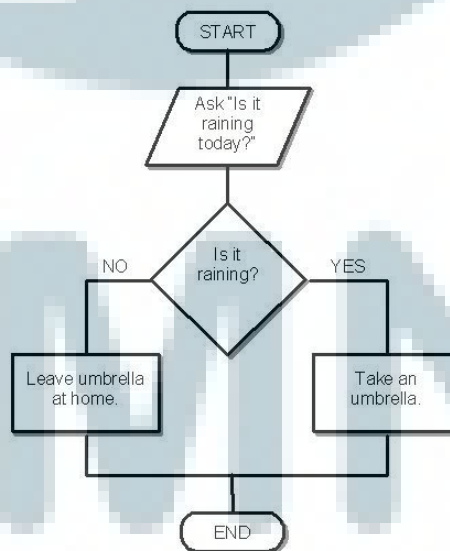
UMMN

Tabel 2.1 Entity Relationship Diagram

Notasi	Keterangan
 Entitas	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
 Relasi	Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
 Atribut	Atribut berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai <i>key</i> diberi garis bawah).
 Garis	Garis sebagai penghubung antara relasi dan entitas atau relasi dan entitas dengan atribut.

2.6 Diagram Flowchart

Diagram *flowchart* merupakan diagram yang menggambarkan prosedur suatu program secara logis dan sistematis. *Flowchart* digunakan terutama dalam hal dokumentasi dan representasi suatu algoritma (J.Hartono,2005).



Gambar 2.1 Contoh Digram Flowchart

Gambar 2.2 merupakan simbol-simbol yang umumnya dipakai di dalam diagram *flowchart*.

Simbol	Nama Simbol	Keterangan	Simbol	Nama Simbol	Keterangan
	Simbol input/output	Simbol input/output (<i>input/output symbol</i>) digunakan untuk mewakili data input/output		Simbol persiapan	Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran
	Simbol Proses	Simbol proses digunakan untuk mewakili suatu proses		Simbol titik terminal	Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk menunjukkan awal dan akhir dari suatu proses
	Simbol garis alir	Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus dari proses			
	Simbol penhubung	Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang sama atau di halaman lainnya			
	Simbol keputusan	Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyelesaian kondisi di dalam program			
	Simbol proses terdefinisi	Simbol proses terdefinisi (<i>predefined process symbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain			

Gambar 2.2 Simbol Diagram *Flowchart*

2.7 *Data Flow Diagram*

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikerjakan pada data tersebut (Kristanto, 2003).

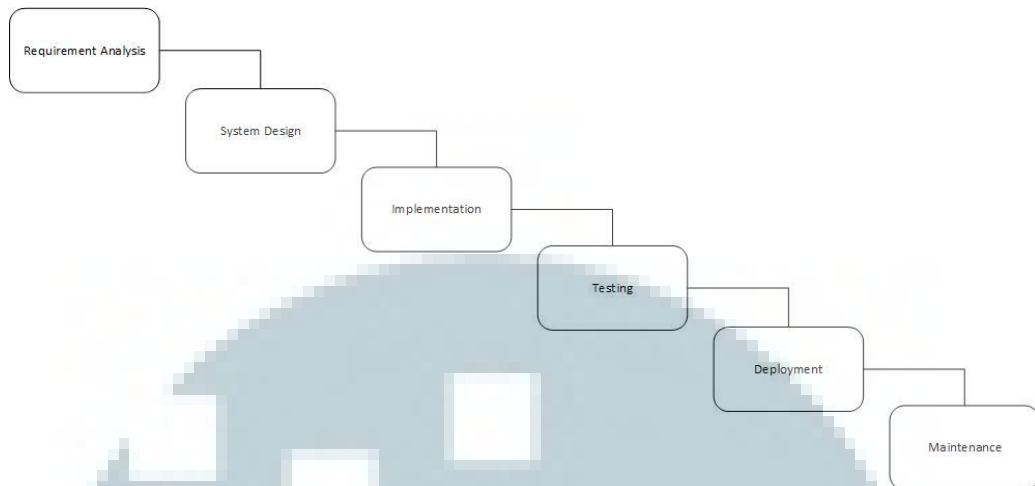
DFD menggambarkan penyimpanan data dan proses yang mentransformasikan data. DFD menunjukkan hubungan antara data pada sistem dan proses pada sistem (Kristanto, 2003).

2.8 SDLC (*Software Development Life Cycle*)

Menurut Roger S. Pressman, 1992 . SDLC *Waterfall* atau metode yang memiliki nama lain “*Linear Sequential Model*” ini merupakan model yang paling banyak dipakai pada *Software Engineering* dan merupakan model yang muncul pertama kali pada tahun 1970. Model ini memberikan pendekatan-pendekatan sistematis dan berurutan bagi pengembangan piranti lunak. Pembangunan dengan metode *waterfall* memiliki tujuan yang berbeda-beda untuk setiap tahapan pembangunannya. Setelah tahapan pembangunan selesai, hasil pengembangan ke tahap berikutnya dan tidak akan kembali ke tahapan yang sebelumnya.

Keuntungan dari menggunakan metode ini adalah memungkinkan adanya pengelompokkan kerja (*departementalization*) dan juga mengontrol manajerial. Setiap tahap pengembangan dapat diatur jadwalnya dan diberikan tenggat waktu, sehingga pembangunan akan lebih terstruktur dan selesai sesuai dengan rencana.

Sementara kerugian dari menggunakan metode ini adalah tidak memungkinkan adanya revisi. Setelah aplikasi sudah sampai pada tahap pengujian, sulit untuk kembali ke tahapan sebelumnya dan mengubah aplikasi, maka harus ada perencanaan yang matang pada saat pembangunan aplikasi dengan menggunakan metode ini.



Gambar 2.3 Contoh Model SDLC Waterfall

Berikut merupakan tahapan SDLC (*Software Development Life Cycle*) model *waterfall* (Roger S. Pressman, 1992) :

1. Requirement Analysis (Analisis Kebutuhan)

Merupakan tahap analisis dan perencanaan dimana penulis mencari informasi mengenai permasalahan yang terkait dengan topik. Kebutuhan, kegunaan, dan batasan *software* juga harus didapatkan pada tahap ini. Informasi-informasi tersebut nantinya akan dianalisis untuk digunakan pada tahap selanjutnya.

2. System Design (Desain Sistem)

Merupakan tahap perancangan dimana penulis mencoba membuat solusi dari permasalahan yang diperoleh dari tahap analisis. Tahap ini membantu dalam memberikan spesifikasi kebutuhan *hardware* dan sistem serta mendefinisikan arsitektur sistem secara keseluruhan.

3. *Implementation (Implementasi)*

Merupakan tahap implementasi dimana penulis mengimplementasikan rancangan dari tahap-tahap sebelumnya dan melakukan uji coba. Tahap ini penulis melakukan pemrograman. Pada tahap ini pengembang juga melakukan pemeriksaan terhadap modul yang telah dibuat, dan menentukan apakah aplikasi yang dibangun sudah memenuhi fungsi yang diinginkan.

4. *Testing (Pengujian)*

Merupakan tahap dimana aplikasi di ujicoba dan memastikan bahwa aplikasi telah mampu menyelesaikan permasalahan yang telah didapat dari tahap *requirement analysis* dan mengecek apakah masih aplikasi masih terdapat kesalahan atau tidak.

5. *Deployment (Penyebaran)*

Merupakan tahap dimana aplikasi telah berhasil di ujicoba dan telah dipergunakan oleh banyak orang.

6. *Maintenance (Pemeliharaan)*

Merupakan tahap dimana pengguna telah menggunakan aplikasi dan melakukan perbaikan apabila terdapat masalah.

2.9 Wawancara

Wawancara merupakan alat *re-checking* atau pembuktian terhadap informasi atau keterangan yang diperoleh sebelumnya. Teknik wawancara yang digunakan dalam penelitian kualitatif adalah wawancara mendalam. Wawancara mendalam (*in-depth interview*) adalah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab sambil bertatap muka antara pewawancara dengan informan atau orang yang diwawancarai, dengan atau tanpa menggunakan pedoman (*guide*) wawancara, di mana pewawancara dan informan terlibat dalam kehidupan sosial yang relatif lama.

Selanjutnya wawancara dapat dilakukan secara terstruktur dan tidak terstruktur, dan dapat dilakukan dengan tatap muka (*face to face*) maupun menggunakan telepon (Sugiyono, 2006; 138-140).

1. Wawancara Terstruktur

Pada wawancara ini digunakan sebagai teknik pengumpulan data, bila peneliti atau pengumpul data telah mengetahui dengan pasti tentang informasi apa yang akan diperoleh. Dalam prakteknya selain membawa instrument sebagai pedoman wawancara, maka pengumpul data juga dapat menggunakan alat bantu seperti *tape recorder*, gambar, brosur dan material lain yang dapat membantu dalam wawancara.

2. Wawancara tidak Terstruktur

Wawancara tidak terstruktur maksudnya adalah wawancara yang bebas di mana peneliti tidak menggunakan pedoman wawancara yang telah tersusun secara sistematis dan lengkap untuk pengumpulan datanya. Pedoman wawancara yang digunakan hanya berupa garis-garis besar permasalahan yang akan ditanyakan.

