BAB II

LANDASAN TEORI

Pada bab ini akan di jelaskan beberapa konsep yang dikaitkan dengan beberapa permasalahan di dalam penelitian yang dimana akan digunakan sebagai dasar pemahaman dalam mengimplementasikan konsep-konsep tersebut kedalam semua kegiatan pengembangan sistem.

2.1 Konsep Dasar Sistem

Konsep dasar sistem ada dua pendekatan yaitu penekanan pada prosedurnya dan penekanan pada komponennya. Menurut Jerry Fitz Gerald pengertian system yang menakankan pada prosedurnya adalah

sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu. (Jogiyanto,2002 : 15).

Adapun menurut Jogiyanto dalam bukunya “Analisis dan Desian Sistem Informasi” pengertian system yang menggunakan pendekatan pada komponennya yaitu.

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu”.(Jogiyanto,2005,2)

2.1.1 Karakteristik Sistem

Adapun system mempunyai beberapa katekteristik atau sifat-sifat tertentu antara lain yaitu:

a. Memiliki komponen

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak perduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

b. Batas sistem (boundary)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu system dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

c. Lingkungan luar sistem (environment)

Adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar system dapat menguntungkan dan merugikan.

1. Keuntungan : Merupakan energi dari sistem yang harus tetap dijaga dan dipelihara karena bersifat menguntungkan.
2. Kerugian : Merupakan energi dari sistem yang harus ditahan dan dikendalikan, energi ini bisa merusak kelangsungan hidup dari sistem.

d. Penghubung sistem (interface)

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Adapun kegunaan penghubung lainnya adalah.

1. Memungkinkan sumber-sumber dapat mengalir dari subsistem yang satu ke subsistem yang lainnya.
2. Keluaran dari subsistem menjadi masukan untuk subsistem yang lainnya melalui penghubung.
3. Satu subsistem dapat berintegrasi dengan subsistem yang lainnya untuk membentuk satu kesatuan

e. Masukan sistem (input)

Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses

untuk didapatkan keluaran. Sebagai contoh didalam system komputer, program adalah maintanance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.

f. Keluaran sistem (Output)

   Merupakan hasil dari energi yang diolah oleh sistem.

g. Pengolah sistem (Process)

   Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

h. Sasaran sistem

   Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

2.2 Konsep Dasar Informasi

Dalam manajemen, informasi merupakan data yang telah diproses sehingga mempunyai arti tertentu bagi penerimanya. Sumber dari informasi adalah Data, sedangkan data itu sendiri adalah kenyataan yang menggambarkan suatu kejadian, sedangkan kejadian itu merupakan suatu peristiwa yang terjadi pada waktu tertentu .dalam hal ini informasi dan data saling berkaitan.

Menurut Jogiyanto dalam buku *'Analisis dan desain sistem informasi'* adalah :

"Informasi diartikan sebagai data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya" . (Jogiyanto,2005; 8).

Adapun pengertian data adalah:

“ Data Merupakan kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.” (Jogyanto, 2005;8).

Pengertian Informasi selalu dikaitkan dengan data, namun arti dari masing-masing kata dalam pengertian tersebut berbeda. Keberadaan suatu data sangat menunjang terhadap informasi, karena data merupakan bahan mentah yang diperlukan untuk mengambil keputusan.

2.3 Konsep Dasar Sistem Informasi

Pengertian system informasi menurut Robert A.Leitch adalah:

‘Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi ,bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan’. (Jogiyanto,2005;11)

2.3.1 Komponen Sistem Informasi

Adapun komponen-komponen dari suatu system informasi meliputi:

1. Blok masukan (input)

Blok masukan ini mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukan, yang dapat berupa dokumen - dokumen dasar.

2 . Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematika yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (output)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkat manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi merupakan alat yang digunakan untuk menerima masukan, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi, perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Basis Data.

Basis data merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer, basis data diakses atau dimanipulasi dengan menggunakan paket perangkat lunak yang disebut data base manajemen sistem (DBMS).

6. Blok kendali

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem bisa dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.4 Pemodelan

2.4.1 Pengertian UML

Unified Modelling Language (UML) adalah sebuah “bahasa” yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah system. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (Object Modeling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering).

Sejarah UML sendiri cukup panjang. Sampai era tahun 1990 seperti kita ketahui puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi booch , metodologi coad , metodologi OOSE , metodologi OMT , metodologi shlaer-mellor , metodologi wirfs-brock ,

dsb. Masa itu terkenal dengan masa perang metodologi (*method war*) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan group/perusahaan lain yang menggunakan metodologi yang berlainan.

Dimulai pada bulan Oktober 1994 Booch, Rumbaugh dan Jacobson, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan mempelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 direlease draft pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh **Object Management.**

2.4.2 Diagram-Diagram Dalam UML

a. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Use case diagram dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan

merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

b. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi *class, package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain.

Class memiliki tiga area pokok :

1. Nama (dan stereotype)

2. Atribut

3. Metoda

Atribut dan metoda dapat memiliki salah satu sifat berikut :

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan
2. *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya
3. *Public*, dapat dipanggil oleh siapa saja

Class dapat merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class.* Dengan demikian *interface* mendukung resolusi metoda pada saat *run-time*. Sesuai dengan perkembangan *class* model, *class* dapat dikelompokkan menjadi *package*. Kita juga dapat membuat diagram yang terdiri atas *package*.

Hubungan Antar *Class* :

1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki

atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah

navigability menunjukkan arah *query* antar *class*.

2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas..”).

3. Pewarisan, yaitu hubungan hirarkis antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.

4. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.

c. Statechart Diagram

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari *stimuli* yang diterima. Pada umumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).

Dalam UML, *state* digambarkan berbentuk segiempat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu. Transisi antar *state* umumnya memiliki kondisi *guard* yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. *Action* yang dilakukan sebagai akibat dari *event* tertentu dituliskan dengan diawali garis miring. Titik awal dan akhir digambarkan berbentuk lingkaran berwarna penuh dan berwarna setengah.

d. Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal.

Activity diagram dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

e. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan *icon* khusus untuk objek *boundary*, *controller* dan *persistent entity*.

f. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. Messages dari level yang sama memiliki prefiks yang sama.

g. Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*.

Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

h. Devloyment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal

Sebuah *node* adalah server, *workstation*, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini.

2.5 Pengenalan Jaringan Komputer

Berdasarkan kriterianya, jaringan komputer dibedakan menjadi 4 yaitu:

1. Berdasarkan distribusi sumber informasi/data

a. Jaringan terpusat

Jaringan ini terdiri dari komputer klient dan server yang mana komputer klient yang berfungsi sebagai perantara untuk mengakses sumber informasi/data yang berasal dari satu komputer server

b.Jaringan terdistribusi

Merupakan perpaduan beberapa jaringan terpusat sehingga terdapat beberapa komputer server yang saling berhubungan dengan klient membentuk sistem jaringan tertentu.

2. Berdasarkan jangkauan geografis dibedakan menjadi:

a. Jaringan LAN

Merupakan jaringan yang menghubungkan 2 komputer atau lebih dalam cakupan seperti laboratorium, kantor, serta dalam 1 warnet.

b. Jaringan MAN

Merupakan jaringan yang mencakup satu kota besar beserta daerah setempat. Contohnya jaringan telepon lokal, sistem telepon seluler, serta jaringan relay beberapa ISP internet.

c. Jaringan WAN

Merupakan jaringan dengan cakupan seluruh dunia. Contohnya jaringan PT. Telkom, PT. Indosat, serta jaringan GSM Seluler seperti Satelindo, Telkomsel, dan masih banyak lagi.

3. Berdasarkan peranan dan hubungan tiap komputer dalam memproses data.

a. Jaringan Client-Server

Pada jaringan ini terdapat 1 atau beberapa komputer server dan komputer client. Komputer yang akan menjadi komputer server maupun menjadi komputer client dan diubah-ubah melalui *software* jaringan pada protokolnya. Komputer client sebagai perantara untuk dapat mengakses data pada komputer server sedangkan komputer server menyediakan informasi yang diperlukan oleh komputer client.

b. Jaringan Peer-to-peer

Pada jaringan ini tidak ada komputer client maupun komputer server karena semua komputer dapat melakukan pengiriman maupun penerimaan informasi sehingga semua komputer berfungsi sebagai client sekaligus sebagai server.

4. Berdasarkan media transmisi data

a. Jaringan Berkabel (*Wired Network*)

Pada jaringan ini, untuk menghubungkan satu komputer dengan komputer lain diperlukan penghubung berupa kabel jaringan. Kabel jaringan berfungsi dalam mengirim informasi dalam bentuk sinyal listrik antar komputer jaringan.

b. Jaringan Nirkabel (*Wireless Network*)

Merupakan jaringan dengan medium berupa gelombang elektromagnetik. Pada jaringan ini tidak diperlukan kabel untuk menghubungkan antar komputer karena menggunakan gelombang elektromagnetik yang akan mengirimkan sinyal informasi antar komputer jaringan.

2.6 Pengenalan *Website* dan Internet

2.6.1 Pengenalan *Website*

Website atau situs dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). Bersifat statis apabila isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Bersifat dinamis apabila isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website. Contoh website statis adalah berisi profil perusahaan, sedangkan website dinamis adalah seperti *Friendster*, *Multiply*, dll. Dalam sisi pengembangannya, website statis hanya bisa diupdate oleh pemiliknya saja, sedangkan website dinamis bisa diupdate oleh pengguna maupun pemilik.

2.6.2 Pengertian Internet

Internet merupakan hubungan antar berbagai jenis komputer dan jaringan di dunia yang berbeda sistem operasi maupun aplikasinya di mana hubungan tersebut memanfaatkan kemajuan media komunikasi (telepon dan satelit) yang menggunakan protokol standar dalam berkomunikasi yaitu protokol TCP/IP.

Dengan adanya TCP/IP ini, internet memiliki 3 keuntungan :

a. Memberi kesempatan internet menggunakan jalur komunikasi yang sama untuk pemakai yang berbeda pada saat yang sama. Karena paket-paket data tidak perlu dikirimkan bersama-sama, jalur komunikasi dapat membawa segala tipe paket data sementara mereka dikirimkan dari tempat yang satu ke tempat yang lain. Sebagai contoh, bayangkan sebuah jalan raya di mana mobil bergerak sepanjang jalan yang sama walaupun mereka menuju ke tempat-tempat yang berbeda-beda.
b. Memberi internet fleksibilitas. Sementara paket-paket data bergerak, mereka bergerak dari satu host ke host lain sampai mencapai tujuan akhir. Jika sebuah jalur komunikasi tidak berfungsi, sistem yang mengontrol aliran data dapat menggunakan jalur alternatif. Maka, paket-paket data dapat bergerak melalui jalur-jalur yang berbeda-beda.
c. Meningkatkan kecepatan transmisi data. Sebagai contoh, jika terjadi kesalahan, TCP meminta host asal mengirm kembali hanya paket-paket data yang mengandung kesalahan, bukan semua paket data. Ini berarti meningkatkan kecepatan transmisi data.

2.7 Perangkat Lunak Pendukung

Dalam pembuatan halaman web, kita membutuhkan perangkat lunak sebagai penunjang pembuatannya baik sebagai text editor maupun sebagai bahasa pemrogramannya.

2.7.1 Pengenalan PHP

PHP adalah salah satu bahasa sever-side yang didesain khusus untuk aplikasi web. PHP dahulunya merupakan proyek pribadi dari Rasmus Lerdorf (dengan dikeluarkannya php versi 1) yang digunakan untuk membuat home page pribadinya. Versi pertama ini berupa kumpulan script PERL. Untuk versi keduanya, Rasmus menulis ulang script-script PERL tersebut menngunakan bahasa C, kemudian menambahkan fasilitas untuk form html dan koneksi MYSQL. Adapun PHP didapat dari singkatan Personal Home Pages.

PHP merupakan bahasa script yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. Oleh karena itu, spesifikasi server lebih berpengaruh pada eksekusi dari script PHP daripada spesifikasi client. Namun tetap diperhatikan bahwa halaman web yang dihasilkan tentunya harus dapat dibuka oleh browser pada client.

2.7.2 Pengenalan MySQL

MySQL merupakan sebuah software yang berguna sebagai suatu database server yang cukup terkenal. Kepopulerannya seiring dengan penggunaan script PHP untuk web programming. Database itu sendiri merupakan suatu software yang bertugas untuk melayani permintaan (request) query dari client. MySQL sebagai suatu database server mempunyai beberapa kemampuan, salah satunya harus menyediakan suatu sistem manajemen database yang dapat mengatur bagaimana menyimpan, menambah, mengakses data dan transaksi-transaksi database lainnya. Transaksi-transaksi utama database yang bisa dilakukan oleh MySQL diantaranya :

1. Membuat database
2. Membuat tabel
3. Menghapus database
4. Menghapus tabel
5. Menambah data
6. Mengedit data
7. Menghapus data
8. Pemilihan data
9. dan lain-lain

2.7.3 Pengenalan Macromedia Dreamweaver 8

Macromedia Dreamweaver adalah sebuah HTML editor profesional untuk mendesain secara visual dan mengelola situs web maupun halaman web. Bilamana kita menyukai untuk berurusan dengan kode-kode HTML secara

manual atau lebih menyukai bekerja dengan lingkungan secara visual dalam melakukan editing, Dreamweaver mambuatnya menjadi lebih mudah dengan menyediakan tool-tool yang sangat berguna dalam peningkatan kemampuan dalam mendesain website.

Dreamweaver MX dalam hal ini digunakan untuk web desain.dreamweaver MX mengikutsertakan banyak tool untuk kode-kode dalam halaman web beserta fasilitas-fasilitasnya, antara lain : Referensi HTML, CSS dan Javascript, Javascript debugger, dan editor kode (tampilan kode dan Code inspector) yang mengizinkan kita mengedit kode Javascript, XML, dan dokumen teks lain secara langsung dalam Dreamweaver. Teknologi Dreamweaver Roundtrip HTML mampu mengimpor dokumen HTML tanpa perlu memformat ulang kode tersebut dan kita dapat menggunakan Dreamweaver pula untuk membersihkan dan memformat ulang HTML bila kita menginginkannya.

Selain itu Dreamweaver juga dilengkapi kemampuan manajemen situs, yang memudahkan kita mengelola keseluruhan elemen yang ada dalam situs. Kita juga dapat melakukan evaluasi situs dengan melakukan pengecekan broken link, kompatibilitas browser, maupun perkiraan waktu download halaman web.

2.8 Sistem Informasi Pelayanan Kamar Hotel

Berdasarkan penjelasan system, informasi, dan system informasi maka beberapa pengertian yang berhubungan dengan judul skripsi antara lain.

2.8.1 Pengertian Pelayanan

Pelayanan adalah suatu kegiatan atau urutan kegiatan yang terjadi dalam interaksi langsung antara seseorang dengan orang lain atau mesin secara fisik, dan menyediakan kepuasan pelanggan. Dalam kamus besar bahasa Indonesia dijelaskan pelayanan sebagai usaha melayani kebutuhan orang lain. [http://www.damandiri.or.id/file/*nurhasyimadunairbab2.pdf*/ 24 Maret 2010]

2.8.2 Pengertian Kamar

Menurut kamus besar Bahasa Indonesia kamar adalah ruang yang bersekat (tertutup) dinding yang menjadi bagian rumah/ bangunan (biasanya disekat atau dibatasi empat dinding).

2.8.3 Pengertian Pelayanan Kamar

Dilihat dari gabungan definisi diatas maka Sistem Informasi Pelayanan Hotel adalah suatu lembaga yang memberikan pelayanan kepada konsumen secara umum dengan tujuan untuk kepuasan pelanggan.

2.8.4 Pengertian Hotel Secara Umum

Hotel merupakan suatu badan usaha komersil yang bergerak dibidang jasa dan merupakan salah satu sector pariwisata di Indonesia, dari pengertian ini hotel dapat didefinisikan dari beberapa instansi, lembaga ataupun keterangan-keterangan antara lain:

1. SK Menteri Perhubungan No. PM. 10/PW 301/77 menyatakan bahwa hotel merupakan bentuk akomodasi yang dikelola secara komersil, dan disediakan bagian setiap orang untuk memperoleh pelayanan yaitu makan dan minum.
2. SK Menteri Pariwisata, Pos dan Telekomunikasi No: KM 37/ PW 304/ MPPT 1989 tanggal 7 juni 1986 menyatakan bahwa Hotel adalah suatu jenis akomodasi yang menggunakan sebagian atau seluruh bangunan untuk menyediakan jasa penginapan, makan dan minum.
3. Menurut American Encyclopedia mengemukakan bahwa hotel merupakan suatu badan usaha yang menyediakan penginapan serta sarana lainnya bagi keum pelancong dan musafir.