

BAB II

LANDASAN TEORI

2.1. Internet

Internet adalah singkatan dari *Interconnection Network* yang secara harfiah berarti hubungan antar jaringan komputer (*network*). Sedangkan *network* sendiri diartikan sebagai suatu sistem komunikasi data antar komputer. Contoh jaringan komputer yang paling sering kita temukan misalnya LAN (*Local Area Network*) yang menghubungkan komputer-komputer yang berada dalam suatu areal atau lokasi tertentu seperti kantor, sekolah, perusahaan, warnet, dan lain-lain.[5]

Pengertian internet secara umum (menurut bahasa) adalah kumpulan dari jaringan komputer yang terhubung dan bekerja sebagai suatu sistem. Sedangkan pengertian internet secara khusus adalah suatu jaringan komputer terbesar di dunia karena menghubungkan seluruh jaringan komputer yang ada di dunia ini. Untuk ringkasnya, internet adalah jaringan komputer global sedangkan jaringan komputer lokal (LAN) dinamakan Intranet. [5]

Untuk mendapatkan akses internet, sebuah komputer harus menggunakan jasa perusahaan penyedia layanan internet atau *Internet Service Provider* (ISP). Dengan jasa perusahaan ISP ini, kita bisa mendapatkan jalur internet (*online*) setelah menghubungkan komputer kita dengan komputer servernya. Hubungan (koneksi) antara komputer pengguna dengan komputer server perusahaan ISP tersebut biasanya dilakukan lewat jaringan telepon (*dial-up*) dengan menggunakan peralatan modem. Lewat komputer server yang dimiliki oleh ISP itulah kita bisa terhubung dengan jaringan komputer sedunia (internet). [5]

2.1.1. Sejarah Internet

Tahun 1972, Roy Tomlinson berhasil menyempurnakan program *e-mail* yang ia ciptakan setahun yang lalu untuk ARPANET. Program *e-mail* ini begitu mudah sehingga langsung menjadi populer. Pada tahun yang sama, *icon @* juga diperkenalkan sebagai lambang penting yang menunjukkan "at" atau "pada". Tahun 1973, jaringan komputer ARPANET mulai dikembangkan ke luar Amerika Serikat. Komputer University College di London merupakan komputer pertama yang ada di luar Amerika yang menjadi anggota jaringan Arpanet. Pada tahun yang sama, dua orang ahli komputer yakni Vinton Cerf dan Bob Kahn mempresentasikan sebuah gagasan yang lebih besar, yang menjadi cikal bakal pemikiran internet. Idc ini dipresentasikan untuk pertama kalinya di Universitas Sussex. [5]

Hari bersejarah berikutnya adalah tanggal 26 Maret 1976, ketika Ratu Inggris berhasil mengirimkan *e-mail* dari Royal Signals and Radar Establishment di Malvern. Setahun kemudian, sudah lebih dari 100 komputer yang bergabung di ARPANET membentuk sebuah jaringan atau *network*. Pada 1979, Tom Truscott, Jim Ellis dan Steve Bellovin, menciptakan *newsgroups* pertama yang diberi nama USENET. Tahun 1981 France Telecom menciptakan gebrakan dengan meluncurkan telepon televisi pertama, dimana orang bisa saling menelpon sambil berhubungan dengan video *link*. [5]

Karena komputer yang membentuk jaringan semakin hari semakin banyak, maka dibutuhkan sebuah protokol resmi yang diakui oleh semua jaringan. Pada tahun 1982 dibentuk Transmission Control Protocol atau TCP dan Internet Protokol atau IP yang kita kenal semua. Sementara itu di Eropa

muncul jaringan komputer tandingan yang dikenal dengan *Eunet*, yang menyediakan jasa jaringan komputer di negara-negara Belanda, Inggris, Denmark dan Swedia. Jaringan *Eunet* menyediakan jasa *e-mail* dan *newsgroup* USENET.[5]

Untuk menyeragamkan alamat di jaringan komputer yang ada, maka pada tahun 1984 diperkenalkan sistem nama domain, yang kini kita kenal dengan DNS atau *Domain Name System*. Komputer yang tersambung dengan jaringan yang ada sudah melebihi 1000 komputer lebih. Pada 1987 jumlah komputer yang tersambung ke jaringan melonjak 10 kali lipat menjadi 10.000 lebih.[5]

Tahun 1988, Jarko Oikarinen dari Finland menemukan dan sekaligus memperkenalkan IRC atau *Internet Relay Chat*. Setahun kemudian, jumlah komputer yang saling berhubungan kembali melonjak 10 kali lipat dalam setahun. Tak kurang dari 100.000 komputer kini membentuk sebuah jaringan. Tahun 1990 adalah tahun yang paling bersejarah, ketika Tim Berners Lee menemukan program editor dan *browser* yang bisa menjelajah antara satu komputer dengan komputer yang lainnya, yang membentuk jaringan itu. Program inilah yang disebut *www*, atau *World Wide Web*. [5]

Tahun 1992, komputer yang saling tersambung membentuk jaringan sudah melampaui sejuta komputer, dan di tahun yang sama muncul istilah *surfing the internet*. Tahun 1994, situs internet telah tumbuh menjadi 3000 alamat halaman, dan untuk pertama kalinya *virtual-shopping* atau *e-retail* muncul di internet. Dunia langsung berubah. Di tahun yang sama Yahoo! didirikan, yang juga sekaligus kelahiran *Netscape Navigator* 1.0.[5]

2.1.2. Fasilitas Internet

- **Website**

Website adalah sebutan bagi sekelompok halaman web, yang umumnya merupakan bagian dari suatu nama domain (*domain name*) atau subdomain di World Wide Web (WWW) pada internet. Situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi, gambar gerak, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan link-link. [5]

- **WWW**

World Wide Web adalah sebuah protokol yang diperkenalkan oleh Tim Berner-Lee, yaitu suatu tata cara berkomunikasi dalam sistem distribusi informasi internet yang digunakan untuk berbagai informasi diantara para fisikawan dan dikembangkan oleh *World Wide Web Consortium* (W3C) adalah konsorsium dari sejumlah organisasi yang berkepentingan dalam pengembangan berbagai standar yang berkaitan dengan web. [5]

1. WWW terdiri daripada satu sistem rangkaian yang menjalin komputer di seluruh dunia dan menyalurkan maklumat dalam bentuk multimedia seperti bunyi, gambar, animasi, video dan juga teks.[5]
2. Tapak atau tempat tersebut dapat dihubungkan melalui *hyperlink* dan membenarkan pengguna bergerak dari satu dokumen ke dokumen yang lain.[5]

3. Membuat pengguna bergerak dengan cepat dari satu dokumen ke dokumen yang lain di dalam internet.[5]
4. Merupakan satu sistem yang menyediakan kemudahan untuk mencari dan mencapai sebarang sumber maklumat yang terdapat dalam internet dengan mudah dan pantas.[5]

2.2. PHP

PHP = PHP : *Hypertext Preprocessor* bahasa pemrograman. PHP adalah bahasa *scripting open source* yang ditulis menggunakan sintaks bahasa C, Java, dan Perl yang sederhana dan mudah dipelajari.[2] PHP terbukti sangat *reliable* penggunaannya dan mempunyai dukungan yang kuat. Dukungan tersebut ialah kemampuan dari php untuk terintegrasi dengan berbagai macam jenis database, mulai dari dbase, ODBC, MySQL sampai ke ORACLE. Menggunakan PHP maka *maintenance* suatu situs web menjadi lebih mudah. Proses *update* data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan *script* PHP. PHP *script* digunakan untuk membuat fasilitas, seperti *login*, *guestbook* dan sebagainya. PHP merupakan bahasa *script server-side* yang disisipkan pada HTML. [3]

Tidak bisa dipungkiri bahwa keamanan akses di sebuah halaman web merupakan suatu hal yang sangat penting untuk diperhatikan. Data yang terdapat dalam suatu halaman web seringkali merupakan data penting yang tidak boleh diperlihatkan sembarangan kepada orang yang tidak berhak. Contohnya tidak mungkin seluruh data yang menyangkut privasi seseorang dalam suatu halaman web langsung ditampilkan kepada semua pengunjung yang membuka situs web

tertentu. Oleh karena itu dibuatlah suatu halaman *login* yang memungkinkan orang yang sudah mempunyai *key (password)* saja yang dapat mengaksesnya.[2]

2.2.1. Sejarah PHP

PHP dibuat pada musim gugur tahun 1994 oleh Ramus Lerdooff. PHP awalnya merupakan program CGI yang dikhususkan untuk menerima *input* melalui *form* yang ditampilkan dalam browser web. Software ini disebarakan sebagai perangkat lunak *Open Source*. Sampai sekarang versi PHP yang terbaru adalah PHP versi 4.2.0 *direleas* pada tanggal 22 April 2002 yang merupakan versi terbaik. Dengan tujuan utama sebagai perbaikan pada kesalahan-kesalahan terutama pada *Upload file*, melalui *Browser*, telah diperbaiki dan banyak penambahan fungsi yang lebih mudah bagi pengembang aplikasi untuk membuat program yang lebih baik lagi. [3]

2.2. MySQL

MySQL merupakan software database yang termasuk paling populer dilingkungan LINUX, kepopuleran ini karena ditunjang dari *performance query* dari data basenya yang saat ini bisa dikatakan paling cepat dan jarang bermasalah. Berangkat dari software yang *sharewere* MySQL populer, kini mulai versi 3.23 MySQL menjadi software *open source* yang berarti *free*. MySQL telah tersedia juga dilingkungan windows, software MySQL dilingkungan Windows dipasang pada *directory* C:\mysql.\Bin adalah direktori yang berisi daftar modul *executable* dari software MySQL. [2]

Dalam bahasa SQL pada umumnya informasi tersimpan dalam tabel-tabel yang secara logik merupakan struktur dua dimensi terdiri dari baris (*row* atau *record*) dan kolom (*column* atau *field*). Sedangkan dalam sebuah *database* dapat terdiri dari beberapa *table*. [2]

2.3. Dreamweaver

Dreamweaver merupakan software unggulan perusahaan pembuat software ternama, Macromedia, Inc. Sebelumnya kelompok software desain web lahir, flash, dreamweaver, dan fireworks. Ketiga software ini memasuki arena bisnis software desain web yang kian ramai, dan ketiga software itu dapat cepat merebut hati para webmaster di seluruh dunia.[1]

Ketika membuka Dreamweaver akan terbuka lembar kerja yang sudah siap pakai, setelah itu kita dapat menyisipkan gambar, text, atau warna dasar. Setelah itu kita dapat menyimpannya dengan cara klik menu file lalu pilih *save as* ketik nama file yang kita inginkan kemudian tekan tombol Ok, harus disimpan dalam bentuk HTML.[1]

Dalam Dreamweaver telah disediakan fasilitas untuk tulisan yang telah ditulis melalui : tekan menu text, kemudian dari menu tersebut paragraph format, font, besar tulisan, warna dan sebagainya dapat disesuaikan atau jika ingin cepat dapat mengaturnya melalui *shortcut-properties*. Cara mengatur tulisan di dalam Dreamweaver sama seperti saat mengatur tulisan dalam Microsoft word.[1]

2.4 . Tools Pendukung

Tools Pendukung merupakan perangkat yang mendukung dalam pembuatan penulisan skripsi ini. *Tools* pendukung dapat membantu untuk melihat jalannya proses pembuatan web dalam penulisan skripsi ini.

2.4.1. Use Case

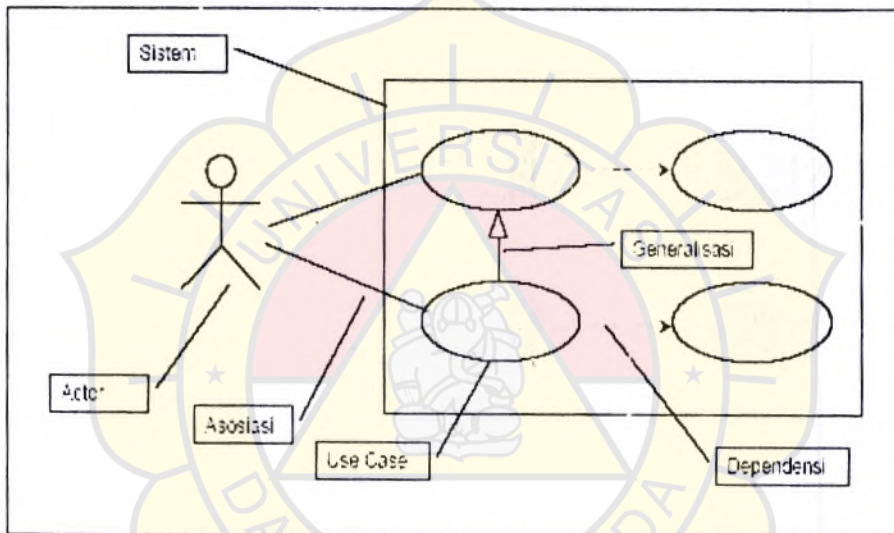
Pemodelan (*modeling*) adalah proses merancang piranti lunak sebelum melakukan pengkodean (*coding*). Model piranti lunak dapat dianalogikan seperti pembuatan *blueprint* pada pembangunan gedung. Membuat model dari sebuah sistem yang kompleks sangatlah penting karena kita tidak dapat memahami sistem semacam itu secara menyeluruh. Semakin kompleks sebuah sistem, semakin penting pula penggunaan teknik pemodelan yang baik. Dengan menggunakan model, diharapkan pengembangan piranti lunak dapat memenuhi semua kebutuhan pengguna dengan lengkap dan tepat, termasuk faktor-faktor seperti *scalability*, *robustness*, *security*, dan sebagainya. [5]

Use case diagram dapat memberikan penjeasan umum (*high level*) antara sistem dengan dunia luar serta fitur-fitur apa yang harus dimiliki oleh sistem dipandang dari dunia luar tersebut. [5]

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.[5]

Use case diagram dapat sangat membantu bila sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem.[5]

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. [5]



Gambar 2.1. Contoh Use Case Diagram

- **Sistem**

Menyatakan batasan sistem dalam relasi dengan *actor-actor* yang menggunakannya (di luar sistem) dan fitur-fitur yang harus disediakan dalam sistem. Sistem digambarkan dengan segiempat yang membatasi semua *Use Case* dalam sistem terhadap pihak mana sistem akan berinteraksi. Sistem disertai label yang menyebutkan nama dari sistem,

tapi umumnya. tidak digambarkan karena tidak terlalu memberi arti tambahan pada diagram.[5]

- **Actor**

Aktor bisa merupakan manusia, sistem, atau *device* yang memiliki peranan dalam keberhasilan operasi dari sistem. Actor digambarkan dengan *icon* yang mungkin bervariasi, namun konsepnya sama. Umumnya untuk orang digambarkan dengan sosok dengan kepala, badan, tangan, dan kaki.[5]

- **Use Case**

Mengidentifikasi fitur kunci dari sistem tanpa fitur sistem tidak akan memenuhi permintaan *user / actor*. Setiap *Use Case* mengekspresikan *goal* dari sistem yang harus dicapai diberi nama sesuai dengan *goalnya* dan digambarkan dengan elips dengan nama di dalamnya. Fokus *use case* tetap pada *goal* bukan bagaimana mengimplementasikannya walaupun *Use Case* berimplikasi pada prosesnya nanti.[5]

- **Asosiasi**

Mengidentifikasi interaksi antara setiap *actor* tertentu dengan setiap *Use Case* tertentu. Setiap asosiasi menjadi dialog yang harus dijelaskan dalam *Use Case Narrative*. Asosiasi digambarkan sebagai garis antara *actor* terhadap *use case* yang bersangkutan. Asosiasi bisa berarah (garis dengan anak panah) jika komunikasi satu arah. Umumnya terjadi kedua arah (tanpa anak panah) karena selalu diperlukan demikian.[5]

- **Stereotype**

Memungkinkan perluasan UML tanpa memodifikasinya. Berperan sebagai *kualifier* pada suatu elemen model. Menyediakan informasi lebih banyak mengenai peranan dari elemen tanpa menyebutkan implementasinya. Terutama untuk menggambarkan:

- *Use Case Dependency*
- *Class-class*
- *Package-package*
- *Classifier*

Notasi dalam diagram dengan *Guillemet* “<<...>>” [5]

- **Dependensi <<include>>**

Mengidentifikasi hubungan antar dua *Use Case* di mana yang satu memanggil yang lain. Jika pada beberapa *Use Case* terdapat bagian yang memiliki aktifitas yang sama maka bagian aktifitas tersebut biasanya dijadikan *Use Case* tersendiri dengan relasi dependensi setiap *Use Case* semula ke *Use Case* yang baru ini sehingga memudahkan pemeliharaan. Digambarkan dengan garis putus-putus bermata panah dengan notasi <<include>> garis. Arah mata panah sesuai dengan arah pemanggilan.[5]

- **Dependensi <<extend>>**

Jika pemanggilan memerlukan adanya kondisi tertentu maka berlaku dependensi <<extend>>. (konsep "extend" ini berbeda dengan 'extend' dalam Java!). Digambarkan serupa dengan dependensi <<include>> kecuali arah panah berlawanan.[5]

- **Generalisasi**

Mendefinisikan relasi antara dua *Actor* atau dua *Use Case* yang mana salah satunya menginherit dan menambahkan atau *override* sifat dari yang lainnya. Penggambaran menggunakan garis bermata panah kosong dari yang menginherit mengarah ke yang dis-desinherit.[5]

2.4.2. Rancangan Tampilan Program

Rancangan Tampilan Program merupakan serangkaian sketsa awal dibuat dalam bentuk persegi panjang yang menggambarkan suatu tampilan. Situs web merupakan kumpulan halaman web yang berhubungan dan saling terkait. Keterhubungan ini mempengaruhi keputusan para perancang web dalam menangani setiap elemen web dan proses-prosesnya. Kelebihan dari metodologi pembangunan situs web ini adalah kemampuan setiap elemen dalam menutupi kelemahan elemen yang lain. Contohnya, jika pada tahap mendesain, desainer menghasilkan karya yang buruk, tetapi mungkin saja pada tahap implementasi proses ini bisa jauh lebih baik. Seperti sebuah pernyataan objektif yang baik dapat memperbaiki sebuah pernyataan usulan yang buruk.[5]

2.4.3. Rancangan Keamanan Web

Bicara tentang sistem sekuriti yang baik sama artinya dengan bicara tentang sesuatu yang ideal. Dalam dunia teknologi informasi yang berkembang cepat, di mana setiap bulan bisa dibilang selalu ada penemuan teknologi baru, sistem sekuriti yang diterapkan tentunya harus mengikuti

perkembangan teknologi tersebut. Namun, mempertahankan sistem sekuriti yang ideal seperti itu tidaklah mudah.[6]

Contoh sederhana, meskipun komputer hadir dengan program anti-virus, belum tentu definisinya *up to date*. Jika program anti-virusnya sudah usang, ia tidak akan sanggup menangkal virus-virus baru, yang terus-menerus muncul. Lebih repot lagi, sebagian besar program anti-virus yang beredar di pasaran Indonesia ditujukan untuk menghadapi virus buatan luar, tapi justru tidak efektif menghadapi virus produksi dalam negeri.[6]

Dalam upaya mencari sistem sekuriti yang baik, kita dapat melihat dari perspektif disain sekuritinya. Secara umum, ada dua pendekatan yang berbeda terhadap sekuriti. Pendekatan yang pertama, memfokuskan terutama pada ancaman-ancaman dari luar (*eksternal*), dan secara umum memperlakukan sistem komputer itu sendiri sebagai sistem yang terpercaya (*trusted*). Sedangkan pendekatan kedua adalah menganggap sistem komputer itu sendiri sebagai sebuah sistem yang umumnya tak dapat dipercaya, dan merancang ulang sistem itu untuk membuatnya lebih aman dengan beberapa cara. [6]

Teknik ini menetapkan suatu pemisahan pengistimewaan (*privilege separation*), di mana sebuah entitas hanya mendapat pengistimewaan sejauh yang dibutuhkan bagi fungsinya. Dengan cara demikian, bahkan jika seorang *hacker* berhasil menyusup ke salah satu bagian dari sistem, sistem keamanan yang baik ini menjamin, akan sangat sulit bagi si penyusup untuk menyusup ke bagian-bagian sistem yang lain. [6]

Rancangan ini harus menggunakan "pertahanan mendalam." Artinya, pihak yang menyusup harus mampu menembus lebih dari satu subsistem,

sebelum bisa menerobos sekuriti sistem keseluruhan, dan mengakses informasi yang disimpannya. Subsistem-subsistem itu harus ditetapkan untuk mengamankan *setting*, dan bilamana mungkin harus dirancang bersifat “gagal-aman” (*fail secure*) ketimbang “gagal-tak aman” (*fail insecure*). [6]

Dalam sebuah web diperlukan suatu keamanan yang dapat memproteksi data dan kode yang digunakan dalam sebuah situs. Perancangan keamanan ini dapat diterapkan dalam bagian web. Contoh keamanan dalam sebuah web adaiah prosedur *login* yang memungkinkan untuk mengenali *user* yang akan masuk dalam situs. Proses *login* ini akan memilah *user* sesuai dengan kapasitasnya untuk menggunakan web. [6]

Web ini dirancang untuk digunakan oleh *user* yang telah ditentukan saja. Sehingga apabila ada *user* diluar yang ditentukan, maka ia tidak dapat menggunakannya. *User* yang dapat menggunakan web ini adalah Admin Perusahaan, Perusahaan, dan *Supplier*. *User* tersebut memiliki batasan penggunaan web. Batasan tersebut dapat dikenali dari *user ID* yang nantinya akan dimasukan oleh *user* untuk memenuhi prosedur *login* dan tentunya dapat menggunakan fasilitas dalam web.[6]