

BAB II LANDASAN TEORI

2.1. UJIAN ONLINE

2.1.1. Definisi Ujian Online

Ujian Online adalah sarana pendidikan yang mencakup motivasi diri sendiri, komunikasi, efisiensi, dan teknologi. ujian online berbasis web tidak mengenal batas ruang dan waktu, dapat menghasilkan informasi yang akurat (human error sangat kecil), tepat waktu (nilai langsung ditampilkan setelah ujian selesai atau waktu habis), dan relevan (nilai ditampilkan secara lebih tepat pada mahasiswa yang bersangkutan). Peningkatan kinerja sistem ujian online supaya menjadi lebih cepat, dapat menggunakan javascript yang berbasis *client side scripting*, (Irfan Hakim, 2007).

Ujian Online efisien karena memudahkan penguji untuk melihat dan memanager hasil-hasil ujian, Hasil ujian seluruh siswa disimpan di dalam database sehingga bisa disimpan dalam kurun waktu bertahun-tahun dan ditampilkan kembali sewaktu-waktu jika diperlukan.

Ujian Online juga mengeliminasi jarak dan arus pulang pergi. Jarak dieliminasi karena isi dari ujian online didesain dengan media yang dapat diakses dari terminal komputer yang memiliki peralatan yang sesuai dan sarana teknologi lainnya yang dapat mengakses jaringan atau Internet.

Istilah ujian online mengandung banyak pengertian, sehingga banyak pakar yang menguraikan tentang definisi ujian online dari berbagai sudut pandang salah satunya *LearnFrame.Com dalam Glossary of e-Learning Terms [Glossary, 2002]* menyatakan suatu definisi yang lebih luas bahwa: ujian

online adalah sistem pendidikan yang menggunakan aplikasi elektronik untuk mendukung ujian dengan media Internet, jaringan komputer, maupun komputer standalone, dari definisi-definisi yang muncul dapat kita simpulkan bahwa sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam proses ujian dapat disebut sebagai suatu ujian online (irfan hakim, 2005).

2.1.2. Fitur Ujian Online

Ujian Online memiliki fitur-fitur sebagai berikut (Cla & Mayer, 2009) :

1. Konten yang relevan dengan tujuan belajar.
2. Menggunakan Aplikasi windows based, tampilan aplikasi ujian online simple dan user friendly
3. Menggunakan elemen media seperti kalimat dan gambar untuk mendistribusikan konten dan metode ujian.
4. Waktu pelaksanaan ujian dapat diset oleh administrator yang berguna untuk memberikan batas waktu yang digunakan oleh peserta ujian.
5. One user one access, aplikasi menerapkan user/peserta ujian hanya sekali melaksanakan ujian.
6. Nilai ujian langsung dapat diketahui begitu siswa selesai mengerjakan ujian langsung.
7. Membangun wawasan dan teknik baru yang dihubungkan dengan tujuan belajar.

2.1.3. Keuntungan Menggunakan Ujian Online

Keuntungan menggunakan ujian online diantaranya sebagai berikut (Irfan Hakim, 2007) :

1. Fleksibel karena siswa dapat melaksanakan ujian online, di mana saja, dan dengan tipe soal yang berbeda sesuai kategori soal tersebut yang telah diset.
2. memiliki fitur autoscoring, yang secara otomatis menghitung hasil test saat itu juga setelah ujian selesai dan menampilkannya secara real-time. Dengan demikian penguji tidak perlu bersusah payah meluangkan banyak waktunya untuk membaca jawaban dan menghitung hasil test setiap peserta.
3. Menghemat waktu proses ujian atau efisiensi waktu.
4. Mengurangi biaya ujian, yang sebelumnya ujian konvensional menggunakan banyak kertas ujian.
5. Menghemat biaya peralatan ujian, yang sebelumnya menggunakan ballpoint, pensil, penghapus dll.
6. Menjangkau wilayah geografis yang lebih luas.
7. Melatih peserta lebih mandiri dalam mengerjakan soal, mengingat soal yang disajikan pilihan ganda dan soal random, yang memungkinkan peserta terhindar dari aksi contek-mencontek.
8. Peserta dapat lebih terkonsentrasi pada ujian mengingat waktu yang disediakan terbatas

2.1.4. Kerugian Menggunakan Ujian Online

Kerugian menggunakan ujian online diantaranya sebagai berikut (Irfan Hakim, 2007) :

1. Jarang adanya soal esai atau isian pada ujian yang dilaksanakan
2. Harus terkoneksi dengan internet dengan baik.
3. Perubahan paradigma, etika dan budaya ujian dikampus atau disekolah.
4. Waktu yang disediakan terbatas
5. Kesalahan dapat terjadi tidak hanya karena peserta, melainkan juga karena kesalahan teknis.
6. Memerlukan biaya apabila ujian dilaksanakan diluar jam belajar, untuk datang ke warnet apabila peserta tidak mempunyai koneksi dirumah
7. Peserta tidak bisa lagi mencontek atau membuka buku.

2.1. Sekilas tentang aplikasi web

2.2.1. Definisi

World Wide Web (WWW) atau biasa disebut dengan web, merupakan salah satu sumber daya internet yang berkembang pesat. Informasi web didistribusikan melalui pendekatan *hypertext*, yang memungkinkan suatu teks menjadi acuan untuk membuka dokumen yang lain, dengan pendekatan ini *user* dapat memperoleh informasi dengan meloncat dari suatu dokumen ke dokumen yang lain.

Sejarah web dimulai pada Maret 1989 ketika Tim Berner – Lee yang bekerja di laboratorium Fisika Partikel Eropa atau CERN (*Conseil Européen pour la Recherche Nucléaire*) di Genewa, Swiss, mengajukan protokol (suatu tata cara berkomunikasi) sistem distribusi informasi internet yang digunakan untuk berbagi informasi di antara para fisikawan. Protokol inilah yang dikenal dengan World Wide Web dan dikembangkan oleh World Wide Web Consortium (W3C). (Abdul Kadir, 2003).

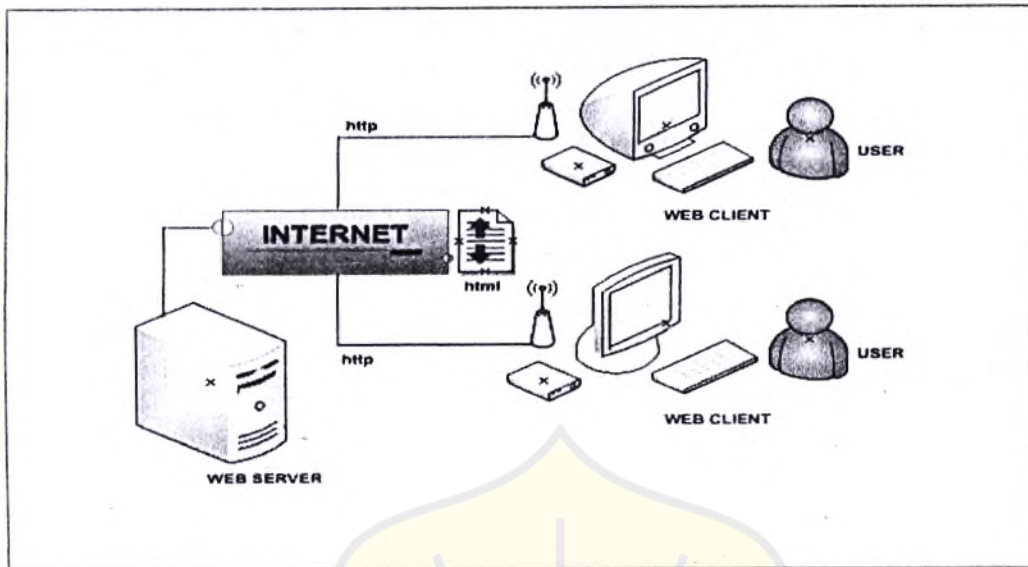
Pada awalnya web dibangun dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language*) dan protokol yang digunakan HTTP (*HyperText Transfer Protocol*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Pada saat ini, banyak skrip seperti itu, antara lain : PHP dan ASP, sedangkan contoh yang berupa objek antara lain adalah applet (Java). Aplikasi web sendiri dapat dibagi menjadi :

a. Web Statis

Web statis dibentuk hanya menggunakan HTML saja. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus menerus untuk mengikuti setiap perubahan yang terjadi (Abdul Kadir, 2003). Setiap ada pergantian isi dalam halaman website maka akan ada perubahan yang mendasar dalam program. Seringnya mengganti isi dalam website maka semakin sering pula memelihara program yang akan mengakibatkan terbuangnya waktu yang digunakan untuk memelihara program. Kelemahan ini diatasi dengan model aplikasi **Web Dinamis**.

b. Web Dinamis

Dengan memperluas kemampuan HTML, yaitu dengan menggunakan perangkat lunak tambahan seperti PHP atau ASP serta DBMS, perubahan informasi dalam halaman-halaman Web dapat ditangani melalui perubahan data, bukan melalui perubahan program (Abdul Kadir, 2003). Sebagai implementasinya, aplikasi web dapat dikoneksikan ke database. Dengan memanfaatkan database, informasi yang ingin ditampilkan dalam website disimpan terlebih dahulu dalam database. Setelah itu dengan memanfaatkan PHP atau ASP, data tersebut dipanggil dan ditampilkan dalam halaman website. Kelebihan yang diperoleh dengan menggunakan database dan PHP, seperti isi dalam halaman website yang selalu *up to date*, *filter* terhadap isi yang ingin disampaikan, interaksi dengan user, dll. Skema aplikasi berbasis web dapat dilihat pada gambar dibawah ini, dimana layanan informasi disediakan oleh web server dan diakses oleh web browser.



Gambar 2.1. Skema Aplikasi Berbasis Web

2.2.2. Kelebihan aplikasi berbasis web

Aplikasi berbasis web sangat baik untuk aplikasi dengan model *client server*. Berikut ini merupakan beberapa kelebihan dari aplikasi berbasis web:

1. *Running anywhere*, kita cukup menginstall aplikasi web pada satu *server* maka kita langsung dapat menjalankan aplikasi tersebut tanpa perlu menginstall dan melakukan konfigurasi apapun pada *client*. *Browser* pada sistem operasi telah tersedia secara *default* sehingga kebutuhan akan *browser* untuk menjalankan aplikasi bukanlah suatu kendala berarti.
2. *Easy to update*, cukup *update* pada *server* maka aplikasi pada *client* akan langsung menggunakan versi terupdate tanpa harus merubah apapun pada *client*.
3. *Hardware requirement* pada *client* tidak terlalu besar, karena aplikasi yang berjalan bersifat *stateless* dan pada sisi *client* hanya sebagai *interface* maka spesifikasi *hardware* pada *client* tidak harus canggih. Dengan komputer

Pentium 300MHz dan memori 128MB sudah cukup dapat menjalankan *browser* untuk menggunakan aplikasi web.

4. Tampilan dan *interface* dapat dibuat sesuka hati, dengan HTML + CSS maka tampilan apapun bisa dibuat pada aplikasi berbasis web sehingga memberikan *user experience* yang berbeda.

2.2.3. Pemrograman web

Dalam membangun sebuah *website* pada dasarnya digunakan bahasa untuk memprogram *website* tersebut. Berikut ini adalah beberapa bahasa yang umumnya sering digunakan dalam merancang dan memprogram sebuah *website*.

1. HTML (*HyperText Markup Language*)

HTML merupakan suatu bahasa pemrograman untuk membuat halaman *website*, kode HTML dapat ditulis langsung dengan editor teks seperti notepad atau secara tidak langsung melalui program seperti Macromedia Dreamweaver, Frontpage untuk lingkungan windows dan Quanta pada Linux. (Abdul Kadir, 2003).

2. PHP (*PHP: Hypertext Preprocessor*)

PHP dahulunya merupakan proyek pribadi dari Rasmus Lerdorf (dengan dikeluarkannya PHP versi 1) yang digunakan untuk membuat *homepage* pribadinya. Versi pertama ini berupa kumpulan *script* PERL. Untuk versi keduanya, Rasmus menulis ulang *script-script* PERL tersebut menggunakan bahasa C, kemudian menambahkan fasilitas untuk Form HTML dan koneksi MySQL. Adapun PHP didapat dari singkatan *Personal Home Pages*. (Jerrisky, 2003).

Setelah mengalami perkembangan oleh suatu kelompok *open source* (termasuk Rasmus) maka mulai versi 3 nya, PHP telah menampakkan keunggulannya sebagai salah satu bahasa *server scripting* yang handal. Melalui perkembangan yang pesat ini banyak fasilitas yang ditambahkan dan oleh kelompok ini PHP disebut sebagai "*PHP: Hypertext Preprocessor*". Sintak yang digunakan berasal dari bahasa C, Java maupun PERL. Sampai tulisan ini dibuat versi PHP yang terbaru adalah versi 5.2.0 Untuk *release* terbaru dari PHP dapat anda lihat pada *website* <http://www.php.net>.

PHP merupakan bahasa *script* yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang akan 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*. Dalam hal ini versi dari HTML yang digunakan harus didukung oleh *browser client*.

Untuk dapat menjalankan *script* PHP pada *website* kita, ada beberapa hal yang perlu kita tambahkan. Pertama tentunya kita harus mempunyai sebuah *web server* yang mengatur atau memberikan tempat untuk mengeksekusi *script* PHP. *Web server* ini diinstall pada komputer server kita. Saat ini PHP dapat dijalankan pada berbagai macam *web*

server seperti PWS, IIS, Xitami maupun Apache. Kemudian hal kedua yang perlu kita miliki adalah PHP Parser.

PHP Parser adalah program yang digunakan untuk menterjemahkan (*interpreter*) kode atau *script* dan kemudian mengeksekusinya. PHP Parser dapat berupa program yang dijalankan pada suatu *shell/ DOS prompt* yang biasanya berupa program yang telah terkompilasi yaitu *php.exe*. Selain itu PHP parser bisa juga berupa modul-modul yang diload oleh *web server*.

3. CSS (*Cascading style Sheets*)

CSS (*Cascading style Sheets*) digunakan dalam kode HTML untuk menciptakan suatu kumpulan *style* yang terkadang dapat digunakan untuk memperluas kemampuan HTML. Karena HTML murni tidak memungkinkan untuk menggunakan definisi *font* yang diterapkan pada setiap sel dan bahkan tag `<basefont>` pun tak berpengaruh pada sel tabel. (Abdul Kadir, 2003).

4. JavaScript

Javascript merupakan *script* yang ditempelkan pada kode HTML dan diproses pada sisi klien, dengan adanya *script* ini, kemampuan dokumen HTML menjadi lebih luas. Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk memvalidasi masukan-masukan pada formulir sebelum dikirim ke server. (Abdul Kadir, 2003).

JavaScript diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan "LiveScript" yang berfungsi

sebagai bahasa sederhana untuk *browser* Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali kita membuat kesalahan pada saat menyusun suatu program.

Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman "Java") pada masa itu, maka Netscape memberikan nama "JavaScript" kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai "Jscript" di *browser* Internet Explorer. JavaScript bergantung kepada *browser* (navigator) yang memanggil halaman web yang berisi *script* dari JavaScript dan tentu saja terselip di dalam dokumen HTML.

JavaScript juga tidak memerlukan kompilator atau penterjemah khusus untuk menjalankannya (pada kenyataannya kompilator JavaScript sendiri sudah termasuk di dalam *browser* tersebut). Lain halnya dengan bahasa "Java" yang memerlukan kompilator khusus untuk menterjemahkannya di sisi *user/ client*.

2.2.4. Tools pembangun aplikasi web

Dalam membangun sebuah aplikasi *website* ada banyak *software* yang khusus ditunjukan untuk membangun dan memperindah tampilan *website*. Beberapa *software* yang digunakan dalam tugas akhir ini adalah :

1. Macromedia Dreamweaver 8.0

Macromedia Dreamweaver 8.0 adalah sebuah *software web design* yang menawarkan cara mendesain *website* dengan dua langkah sekaligus dalam satu waktu, yaitu mendesain dan memprogram. Juga dapat dilihat pada mottonya "*The Solution for Professional Website Design and Production*", yang mana memiliki arti kurang lebih *Solusi bagi para Profesional Web Desainer untuk memproduksi Website*. Ia juga menghimpun banyak teknologi terkini untuk menemani dan membantu dalam merancang sebuah *website*.

Macromedia Dreamweaver 8.0 memiliki satu jendela yang disebut *code* tempat kode-kode HTML tertulis. Setiap kali mendesain *website* seperti menulis kata, meletakkan gambar, membuat tabel, dan proses lainnya, tag-tag HTML akan tertulis secara langsung (otomatis) mengiringi proses pengaturan *website*.

Alasan penulis menggunakan *software* ini antara lain karena teknologi yang digunakan *software* ini memungkinkan untuk mendesain sekaligus memprogram suatu *website* dalam waktu yang bersamaan, selain itu *software* ini sangat compatible dengan berbagai bahasa pemrograman untuk merancang *website*.

2. Adobe Photoshop CS 8.0

Adobe Photoshop merupakan *software image editor* yang digunakan untuk memanipulasi gambar. Alasan penulis menggunakan Adobe Photoshop CS versi 8.0 dikarenakan *software* ini dapat menghasilkan *image* dan file HTML yang digunakan untuk merancang

desain *website*. Kelebihan dari *software* ini terletak pada penggunaan *layer-layer* yang digunakan untuk memanipulasi suatu gambar.

3. XAMPP

XAMPP merupakan sebuah paket *software local web server* yang dapat menjalankan *script* PHP pada *website* kita, Versi XAMPP yang digunakan dalam pembuatan *website* ini adalah versi 1.7.1. versi ini merupakan paket yang berisi dari beberapa aplikasi diantaranya :

- Web server Apache versi 2.2.11
- PHP Engine versi 5.2.9
- Database MySQL versi 5.0.51a
- Aplikasi pengelola database phpMyAdmin versi 3.1.3.1

MySQL dan PHP merupakan kombinasi pasangan piranti (tools) yang banyak digunakan dalam pembangunan dan pengembangan web berbasis database. MySQL dan PHP dapat dikatakan sebagai pasangan software yang serasi untuk pengembangan web berbasis database, walaupun PHP juga dapat digunakan dengan database yang lain.

2.2.5. Database yang digunakan dalam website

Salah satu software DBMS yang biasa digunakan dalam perancangan website adalah MySQL, MySQL merupakan *software* sistem manajemen database (*Database Management System – DBMS*) yang sangat populer dikalangan pemrogram web, terutama di lingkungan linux dengan menggunakan *script* PHP

dan PERL. MySQL telah didistribusikan dengan lisensi open source GPL mulai versi 3.23, terhitung Juni 2000. Secara mendasar database adalah kumpulan data, disimpan dalam file, bisa dalam satu file ataupun beberapa file. (Betha Sidik, 2005).

2.3. Pemodelan sistem

2.3.1. Definisi UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan system yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. (Munawar, 2005)

UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, *Object Modeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode Booch dari Grady Booch sangat terkenal dengan nama metode *Design Object Oriented*. Metode ini menjadikan proses analisis dan *design* kedalam empat tahapan *iterative*, yaitu: identifikasi kelas-kelas dan obyek-obyek, identifikasi semantik dari hubungan obyek dan kelas tersebut, perincian *interface* dan implementasi. Pemodelan OMT yang dikembangkan oleh Rumbaugh didasarkan pada analisis terstruktur dan pemodelan *entity-relationship*. Tahapan utama dalam metodologi ini adalah analisis, *design system*, *design* obyek dan implementasi. Metode OOSE dari Jacobson lebih memberi penekanan pada

use case. OOSE memiliki tiga tahapan yaitu membuat *model requirement* dan analisis, *design* dan implementasi, dan model pengujian (*test model*).

2.3.2. Tujuan UML

Tujuan dari UML adalah:

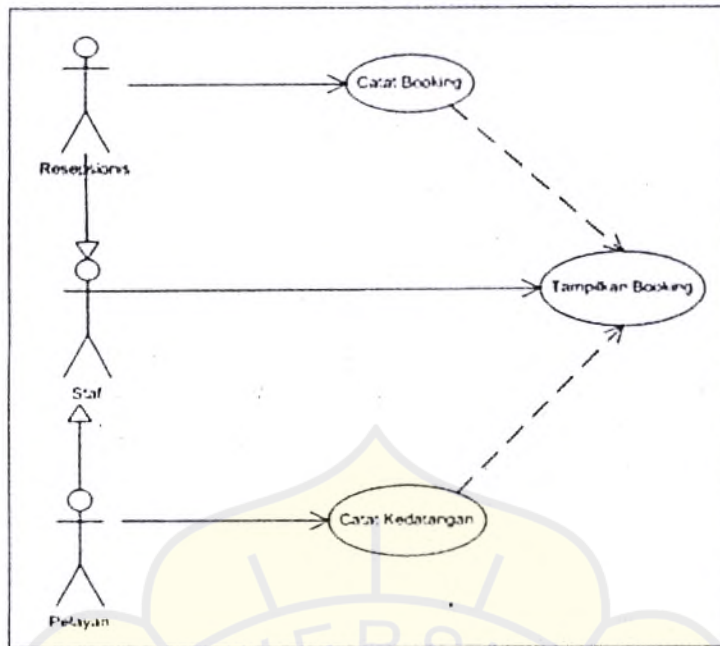
1. Memberikan model yang siap pakai, bahasa permodelan visual yang ekspresif untuk mengembangkan dan saling menukar model yang mudah dan dimengerti secara umum.
2. Memberikan bahasa permodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam permodelan.

2.3.3. Diagram-diagram yang terdapat pada UML

UML sendiri terdiri atas pengelompokkan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram menggambarkan permasalahan maupun solusi dari permasalahan suatu model. Dibawah ini merupakan beberapa diagram UML:

- Diagram Use Case

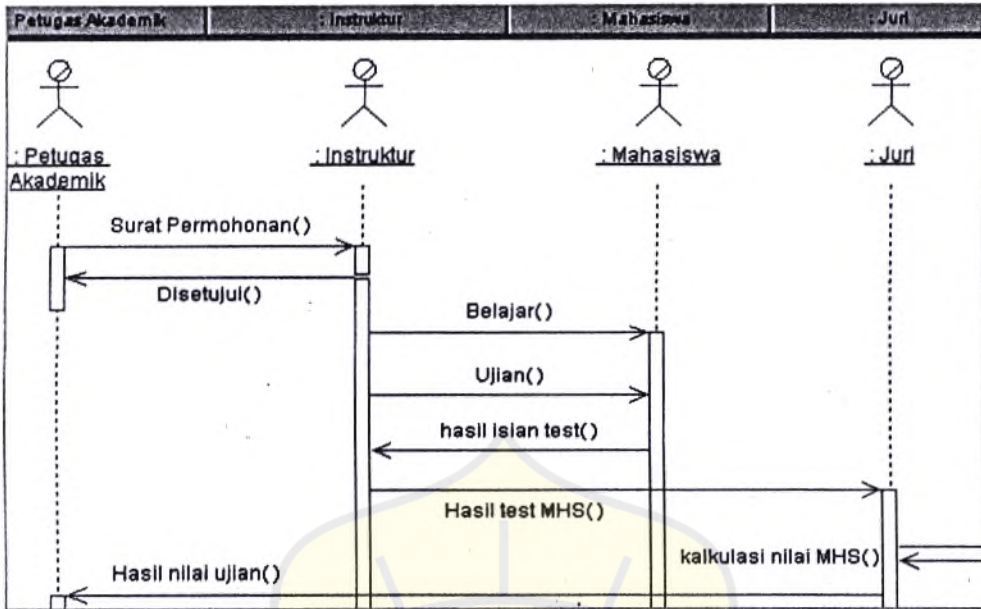
Diagram Use Case menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar, yang menjadi persoalannya yaitu apa yang dilakukan dan bukan bagaimana melakukannya. Diagram Use Case dekat kaitannya dengan kejadian-kejadian. Kejadian (*scenario*) merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan sistem.



Gambar 2.2 Contoh Diagram Use Case

- Diagram Sequence

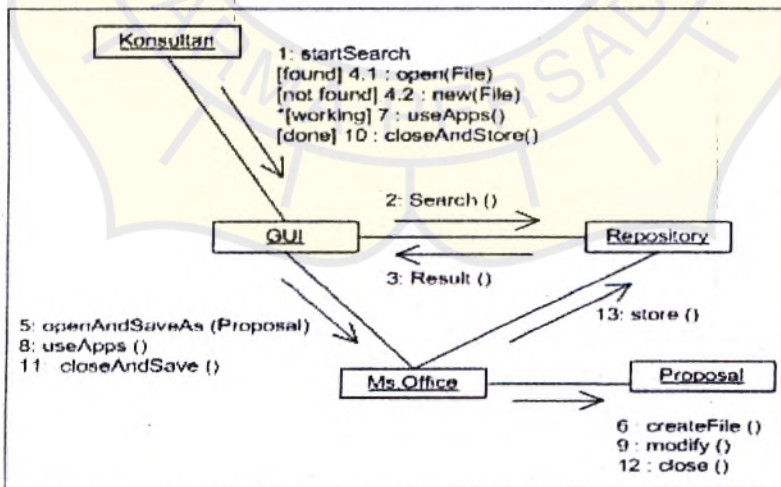
Diagram Class dan diagram Object merupakan suatu gambaran model statis. Namun ada juga yang bersifat dinamis, seperti Diagram Interaction. Diagram Sequence merupakan salah satu diagram Interaction yang menjelaskan bagaimana suatu operasi itu dilakukan, *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Obyek-obyek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.



Gambar 2.3 Contoh Diagram Sequence.

▪ Diagram Collaboration

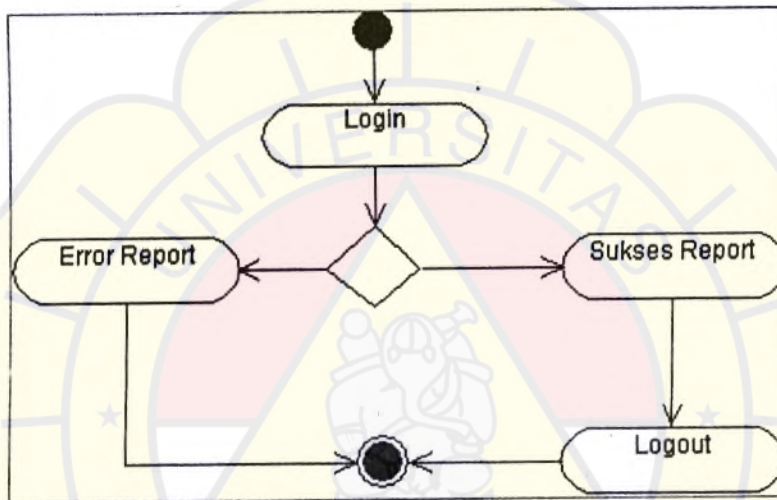
Diagram Collaboration juga merupakan diagram Interaction. Diagram membawa informasi yang sama dengan diagram Sequence, tetapi lebih memusatkan atau memfokuskan pada kegiatan obyek dari waktu pesan itu dikirimkan.



Gambar 2.4 Contoh Diagram Collaboration

- Diagram Activity

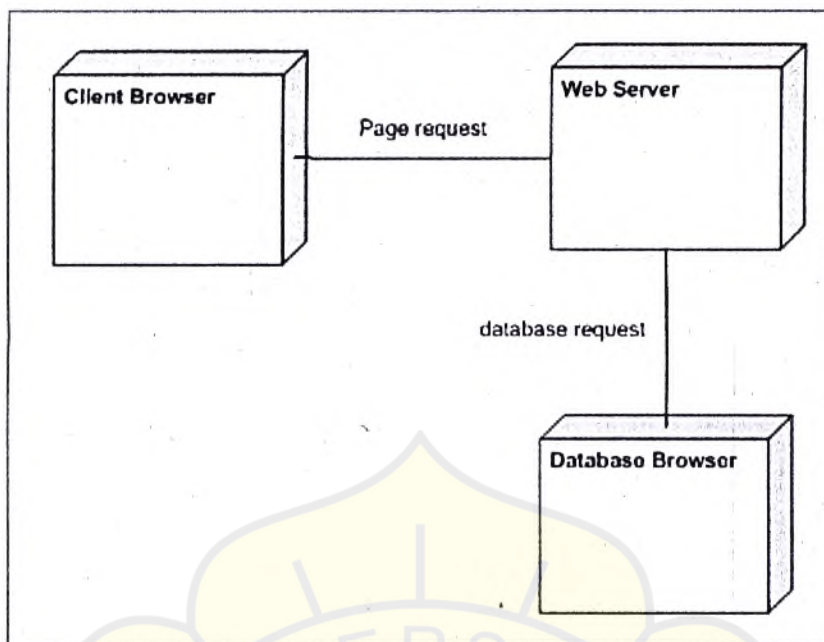
Pada dasarnya diagram Activity sering digunakan oleh *flowchart*. Diagram ini berhubungan dengan diagram Statechart. Diagram Statechart berfokus pada obyek yang dalam suatu proses (atau proses menjadi suatu obyek), diagram Activity berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain



Gambar 2.5 Contoh Diagram Activity

- Component Diagram

Komponen perangkat lunak adalah bagian fisik dari sebuah sistem yang menetap di komputer. komponen merupakan implementasi *software* dari sebuah *class*. Komponen bisa berupa tabel, *file data*, *file exe*, file DLL, dokumen dan lain-lain. *Component diagram* mengandung komponen, *interface* dan *relationship*. Komponen diagram ini digunakan pada saat anda ingin memecah sistem menjadi komponen-komponen dan ingin menampilkan hubungan-hubungan mereka dengan antarmuka atau pemecahan komponen menjadi struktur yang lebih rendah. Secara umum dapat kita katakan bahwa



Gambar 2.7 Contoh Deployment Diagram

2.4 Basis Data (Database)

Berdasarkan sumber (Dyah Wardhani Kusuma, 2007) pengertian basis data adalah :

Basis data merupakan kumpulan file/ tabel/ arsip yang saling berhubungan yang disimpan sedemikian rupa tanpa pengulangan (*redundancy*) dan dapat dimanfaatkan kembali untuk memenuhi kebutuhan secara cepat dan mudah.

Prinsip utama basis data adalah pengaturan data dengan tujuan fleksibilitas dan kecepatan dalam pengambilan data kembali. Adapun tujuan basis data diantaranya adalah sebagai berikut :

- a. Kecepatan dan kemudahan (*speed*) : agar user dapat menyimpan, memanipulasi, dan menampilkan kembali data lebih cepat dan mudah daripada cara biasa.
- b. Efisiensi ruang penyimpanan (*space*) : mengurangi *redundancy*, misalnya dengan pengkodean dan membuat relasi.

- c. Keakuratan (*accuracy*) : agar data sesuai dengan aturan dan batasan tertentu.
- d. Ketersediaan (*availability*) : agar dapat diakses oleh setiap user yang membutuhkan.
- e. Kelengkapan (*completeness*) : dengan menambahkan field pada tabel.
- f. Keamanan (*security*) : agar data yang rahasia tidak jatuh ke tangan user yang tidak berhak, misalnya : dengan pengkodean, *account* (*username* dan *password*), pembedaan hak akses untuk setiap user terhadap data yang dapat dibaca atau proses yang dapat dilakukan.
- g. Kebersamaan (*shareability*) : mendukung lingkungan *multiuser*, menghindari inkonsistensi data dan *deadlock*.

Ada beberapa definisi yang umum digunakan dalam basis data, yaitu :

1. Entitas : Entitas adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Pada bidang Administrasi Siswa misalnya, siswa, buku, pembayaran.
2. Atribut : Atribut biasa disebut juga data elemen, data field, atau data item yang digunakan untuk menerangkan suatu entitas dan mempunyai harga tertentu, misalnya atribut dari entitas pegawai diterangkan oleh, nama, umur, alamat, pekerjaan.
3. Data Value (Nilai Data) : Data Value adalah data aktual atau informasi yang disimpan pada tiap data, elemen, atau atribut. Atribut nama pegawai menunjukkan tempat dimana informasi nama karyawan disimpan, nilai datanya misalnya adalah Anjang, Arif, Suryo, dan lain-lain yang merupakan isi data nama pegawai tersebut.

5. Gunakan *layout* yang sederhana, anda hanya memerlukan 7 detik untuk menarik perhatian pengunjung akan website anda.
6. Bila anda memiliki sesuatu yang perlu dipublikasikan buatlah fungsi yang dapat membuat dokumen *print* atau *convert* ke file pdf, MS. Excel atau format informasi lainnya.
7. Jangan biarkan banyak bagian dari website yang kosong, manfaatkanlah sebaik mungkin seperti penambahan grafis atau gambar yang menarik dan berkaitan dengan website yang kita buat.
8. Buatlah mesin pencari pada setiap halaman, agar pengunjung mudah dalam mencari sesuatu tentang website kita.
9. Carilah 5 orang untuk menguji website anda, perhatikan navigasi dan tugas yang kita berikan.
10. Jangan gunakan sistem navigasi yang kompleks. Selain membuat pengunjung website merasa sulit, pengunjung akan lebih berkonsentrasi mempelajari sistem tersebut bukan memperhatikan isi dan tujuan website yang dibuat.
11. Jangan mempersulitkan akses pengunjung dengan grafis dan animasi yang tidak begitu penting.
12. Jangan membuat form terlalu panjang, sesuai keperluan saja.

4. File/ Tabel : Kumpulan record sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda nilai datanya.
5. Record/ Tuple : Kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap. Satu record mewakili satu data atau informasi.

2.5 Pengertian User Friendly

User friendly dapat diartikan bersahabat dengan pemakai, disukai pengguna, nyaman digunakan. Istilah ini banyak digunakan untuk berbagai program komputer, dimana interaksi antara pemakainya (manusia) dengan program itu sendiri, misalnya program tersebut ditampilkan dalam bentuk gambar-gambar yang mendukung penampilan dalam bentuk gambar-gambar.

Dalam membangun website, faktor *user friendly* sangat diperlukan agar pengunjung website dapat merasakan kemudahan dalam mengakses website yang kita bangun. *User friendly* juga sangat berpengaruh terhadap keberhasilan tujuan dari website yang dibangun, menurut (Uro Abd. Rohim, S.Kom., M.T, 2009) terdapat beberapa hal yang harus dilakukan dalam membangun website yang *user friendly* antara lain:

1. Periksa link dan pastikan berfungsi dengan baik.
2. Buatlah link yang menandakan bahwa link tersebut dapat di klik. Hal ini bisa dilakukan dengan menandai tulisan dengan garis bawah atau warna berbeda dan gunakan fungsi *mouse over*.
3. Jangan persulit link pengunjung dengan petunjuk yang berlebihan.
4. Kurangi penggunaan klik yang berlebihan dalam satu halaman.