BAB II
TEORI PENUNJANG

2.1 Sistem rekomendasi tempat wisata

Sistem rekomendasi tempat wisata adalah sebuah sistem yang menyediakan saran dan memberikan informasi terkait tempat wisata untuk dapat dimanfaatkan oleh user. Perkembangan sistem rekomendasi bisa dilihat pada awalnya dari studi dibidang ilmu kognitif (*cognitive science*), teori perkiraan (*approximation theories*), pengambilan informasi (*information retrieval*), dan teori prediksi (*forecasting theories*), dan juga memiliki hubungan dengan ilmu manajemen (*management science*) dan pemodelan pilihan pelanggan dalam ilmu pemasaran. Meskipun demikian sistem rekomendasi muncul sebagai bidang penelitian tersendiri di pertengahan 1990-an ketika para peneliti mulai fokus pada permasalahan rekomendasi yang secara khusus berdasar pada sebuah struktur penilaian (*rating*). Sistem rekomendasi diberikan dalam berbagai bentuk, berbeda baik dalam faktor pengambilan keputusan, penyajian informasi, maupun media yang digunakan.

Tujuan sistem rekomendasi adalah memperkirakan penilaian-penilaian untuk hal-hal (*items*) yang belum pernah dilihat oleh seorang pengguna. Perkiraan
ini umumnya didasarkan pada penilaian si pengguna terhadap hal-hal setara lainnya beserta pertimbangan tambahan lainnya. Setelah perkiraan penilaian/*rating* didapat untuk hal-hal yang baru, maka beberapa rekomendasi bisa diberikan berdasar perkiraan *rating* tertinggi. Sistem rekomendasi sering diaplikasikan pada area dimana pilihan tersedia dalam jumlah relatif banyak. Informasi yang berhubungan dengan kriteria wisata dan semua fasilitasnya, termasuk harga tiket serta jarak adalah beberapa hal yang biasa dicari oleh para wisatawan dalam perjalanan wisatanya. Sementara itu, jumlah informasi yang tersedia baik *offline* maupun *online* dari jaringan internet bisa sangat berlebihan. Informasi yang sangat banyak ini bisa menjadi sangat rumit dan menghabiskan waktu untuk wisatawan dalam menentukan satu pilihan yang cukup tepat untuknya.

2.2 Logika fuzzy

Kata fuzzy merupakan kata sifat yang berarti kabur atau tidak jelas. Fuzziness atau kekaburan atau ketidakjelasan selalu meliputi keseharian manusia. (Kusumadewi,2004). Logika fuzzy adalah suatu cara yang tepat untuk memetakan ruang input ke dalam suatu ruang output. (Kusumadewi,2004). Logika fuzzy menggunakan ungkapan bahasa untuk menggambarkan nilai variabel. Logika fuzzy bekerja dengan menggunakan derajat keanggotaan dari sebuah nilai yang kemudian digunakan untuk menentukan hasil yang ingin dicapai berdasarkan atas spesifikasi yang telah ditentukan. Ada beberapa hal yang perlu diketahui dalam memahami sistem fuzzy (Kusumadewi,2004), yaitu: Variabel Fuzzy, Himpunan Fuzzy, Semesta Pembicaraan, dan Domain himpunan fuzzy.

2.2.1 Himpunan fuzzy

Himpunan tegas (crisp) A didefinisikan oleh item-item yang ada pada himpunan itu. Jika a € A, maka nilai yang berhubungan dengan A adalah 1. Namun jika a bukan anggota A ,maka nilai yang berhubungan dengan a adalah 0. notasi A = {x|P(x)} menunjukkan bahwa A berisi item x dengan P(x) benar. Jika XA merupakan fungsi karakteristik A dan properti P, maka dapat dikatakan bahwa P(x) benar, jika dan hanya jika XA (x)=1.

Himpunan fuzzy didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan real pada interval [0,1]. Nilai keanggotaannya menunjukkan bahwa suatu item dalam semesta pembicaraan tidak hanya bernilai 0 atau 1, namun juga nilai yang terletak diantaranya. Dengan kata lain, nilai kebenaran suatu item tidak hanya benar (1) atau salah (0) melainkan masih ada nilai-nilai yang terletak diantara benar dan salah (Kusumadewi,2004). Himpunan fuzzy memiliki 2 atribut, (Kusumadewi, 2004) yaitu :

1. Linguistik yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami. Contoh : murah, sedang, mahal.
2. Numeris yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel. Contoh : 100, 500, 1000, dan seterusnya.

2.2.2 Fuzzy keanggotaan

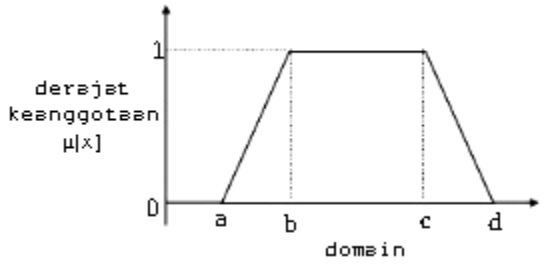
Fungsi keanggotaan (membership function) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya yang memiliki interval 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan, yaitu :

a. Representasi Linear

Pada representasi kurva linear, pemetaan input ke derajat keanggotannya digambarkan sebagai suatu garis lurus.

b. Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk kurva segitiga dengan penambahan beberapa titik yang memiliki nilai keanggotaan



Gambar II.1 *Representasi Kurva Trapesium*

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0 & ; \quad x \leq a \; atau \; x \geq d \\ (x-a)/(b-a) & ; \quad a \leq x \leq b \\ 1 & ; \quad b \leq x \leq c \\ (d-x)/(d-c) & ; \quad c \leq x \leq d \end{cases}$$

Gambar II.2 *Fungsi Keanggotaan*

c. Representasi Kurva Bahu

Himpunan fuzzy bahu digunakan untuk mengakhiri variabel suatu daerah fuzzy. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar.

2.3 Fuzzy Model Tahani

Fuzzy tahani merupakan salah satu metode fuzzy yang menggunakan basis data standar. Basis data *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya (Kusumadewi, 2010). Model Tahani tersusun atas tahapan yaitu:

a. Menggambarkan Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki internal antara 0 sampai 1, salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan melalui pendekatan fungsi. Beberapa fungsi yang dapat digunakan yaitu Representasi kurva Linier, Representasi Kurva Segitiga. Representasi

Kurva Trapesium. Representasi Kurva Bentuk Bahu, Representasi Kurva-S, Representasi Kurva Bentuk Lonceng (*Bell Curve*). Masing-masing fungsi tersebut, akan menghasilkan nilai antara "0" dan "1" dengan cara yang berbeda, sesuai dengan jenis representasi yang digunakan.

b. Fuzzifikasi

Fuzzifikasi adalah fase pertama dari perhitungan *fuzzy* yaitu pengubahan nilai tegas ke nilai *fuzzy*. Prosesnya adalah sebagai berikut: Suatu besaran analog dimasukkan sebagai masukan (*crisp input*), lalu *input* tersebut dimasukkan pada batas *scope*dari *membership function*. *Membership function* ini biasanya dinamakan *membership function input*. Keluaran dari proses fuzzifikasi ini adalah sebuah nilai*input fuzzy* atau yang biasanya dinamakan *fuzzy input*.

c. Fuzzifikasi *Query*

Fuzzifikasi *Query* diasumsikan sebuah *query* konvensional (*nonfuzzy*) DBMS yang akan mencoba membuat dan menerapkan sebuah sistem dasar logika *fuzzy query* .

d. Operator Dasar Zadeh untuk Operasi Himpunan *Fuzzy*.

Nilai keanggotaan sebagai dari 2 himpunan *fuzzy* dikenal dengan nama *Fire Strength* atau α-predikat. Sangat mungkin digunakan operator dasar dalam proses *query* berupa operator AND dan OR.

α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu A\ B = min(\mu_A[x], \mu_B[y])$. Sedangkan untuk hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu AUB = max(\mu_A[x], \mu_B[y])$.

Alternatif yang direkomendasikan adalah alternatif yang memiliki nilai *Fire Strength*atau tingkat kesesuaian dengan kriteria pilihan di atas angka 0 (nol) sampai dengan angka 1 (satu).

2.4 Website

Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet. Website merupakan komponent atau kumpulan komponen yang terdiri dari teks, gambar, suara animasi sehingga lebih merupakan media informasi yang menarik untuk dikunjungi. Penemu website adalah Sir Timothy John ¨Tim¨ Berners-Lee, sedangkan website yang tersambung dengan jaringan, pertamakali muncul pada tahun 1991. Maksud dari Tim ketika membuat website adalah untuk mempermudah tukar menukar dan memperbarui informasi kepada sesama peneliti di

tempat dia bekerja. Pada tanggal 30 April 1993, CERN (tempat dimana Tim bekerja) menginformasikan bahwa WWW dapat digunakan secara gratis oleh semua orang.

Sebuah website bisa berupa hasil kerja dari perorangan atau individu, atau menunjukkan kepemilikan dari sebuah organisasi, perusahaan, dan biasanya website itu menujukkan beberapa topik khusus, atau kepentingan tertentu. Sebuah website bisa berisi hyperlink (pranala) yang menghubungkan ke website lain, jadi, kadangkala perbedaan antara website yang dibuat oleh individu perseorangan dengan website yang dibuat oleh organisasi bisnis bisa saja tidak kentara.

Website ditulis, atau secara dinamik di konversi menjadi HTML dan diakses melalui sebuah program software yang biasa disebut dengan web browser, yang dikenal juga dengan HTTP Client. Halaman web dapat dilihat atau diakses melalui jaringan komputer dan internet, perangkatnya bisa saja berupa komputer pribadi, laptop, PDA ataupun telepon selular.

Sebuah website dibuat di dalam sebuah sistem komputer yang dikenal dengan server web, juga disebut HTTP Server, dan pengertian ini juga bisa menunjuk pada software yang dipakai untuk menjalankan sistem ini, yang kemudian menerima lalu mengirimkan halaman-halaman yang diperlukan untuk merespon permintaan dari pengguna. Apache adalah piranti lunak yang biasa digunakan dalam sebuah webserver, kemudian setelah itu adalah Microsoft Internet Information Services (IIS).

a. Macam-macam situs web

   Sebuah Website statik, adalah salah satu bentuk website yang isi di dalam website tersebut tidak dimaksudkan untuk di update secara berkala, dan biasanya di maintain secara manual oleh beberapa orang yang menggunakan software editor. Ada3 tipe kategori software editor yang biasa dipakai untuk tujuan maintaining ini, mereka adalah :

   1. Elemen 1 Penyunting teks. Contohnya adalah Notepad atau TextEdit, dimana HTML diubah di dalam program editor tersebut.
   2. Elemen 2 WYSIWYG editor. Contohnya Microsoft Frontpage dan Macromedia Dreamweaver, dimana situs di edit menggunakan GUI (Graphical User Interface) dan format HTML ini secara otomatis di generate oleh editor ini.
   3. Elemen 3 Editor yang sudah memiliki templat, contohnya Rapidweaver dan iWeb, dimana, editor ini membolehkan user untuk membuat dan mengupdate websitenya langsung ke server web secara cepat, tanpa harus mengetahui

apapun tentang HTML. Mereka dapat memilih templat yang sesuai dengan keinginan mereka, menambah gambar atau obyek, mengisinya dengan tulisan, dan dengan sekejap mereka sudah dapat membuat situs web tanpa harus melihat sama sekali kode-kode HTML.

Sebuah website dynamic adalah website yang secara berkala, informasi di dalamnya berubah, atau website ini bisa berhubungan dengan user dengan berbagai macam cara atau metode (HTTP cookies atau Variabel Database, sejarah kunjungan, variabel sesi dan lain-lain) bisa juga dengan cara interaksi langsung menggunakan form dan pergerakan mouse. Ketika web server menerima permintaan dari user untuk memberikan halaman tertentu, maka halaman tersebut akan secara otomatis di ambil dari media penyimpanan sebagai respon dari permintaan yang diminta oleh user. Sebuah situs dapat menampilkan dialog yang sedang berlangsung di antara dua user, memantau perubahan situasi, atau menyediakan informasi yang berkaitan dengan sang user.

2.5 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android pada awalnya dikembangkan oleh perusahaan bernama Android, Inc., dengan dukungan finansial yang berasal dari Google, yang kemudian Google pun membelinya pada tahun 2005. Sistem operasi android tersebut secara resmi dirilis pada tahun 2007, bersamaan dengan didirikannya sebuah perusahaan Open Handset Alliance, konsorsium dari beberapa perusahaan-perusahaan perangkat keras, perangkat lunak, serta telekomunikasi yang memiliki tujuan untuk memajukan standar terbuka dari perangkat seluler. Ponsel yang berbasis sistem operasi Android pertama dijual pada bulan Oktober 2008.

Sejarah android pada mulanya berasal dari perusahaan bernama Android, Inc. didirikan tempatnya di Palo Alto, California, pada Oktober tahun 2003 oleh Andy Rubin (pendiri Danger), Rich Miner seorang pendiri Wildfire Communications, Inc., Nick Sears seorang mantan VP T-Mobile, dan Chris White seorang kepala desain dan pengembangan antarmuka WebTV untuk mengembangkan sebuah "perangkat seluler pintar yang lebih sadar tentang lokasi dan preferensi penggunanya". Android saat ini digunakan sebagai media pembuat aplikasi handphone yang berbasis android. Android Studio adalah tool IDE (Integrated Development Environment) resmi dari Google untuk pembuatan aplikasi

Android. Di dalam Android Studio sudah tersedia editor kode, contoh kode, integrasi dengan GitHub dan banyak lainnya yang diperlukan dalam pembuatan aplikasi Android.

2.6 *Unified Modeling Language*

UML (Unified Modeling Language) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

2.6.1 Sejarah UML

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembang bahasa pemrograman berorientasi objek selanjutnya.

Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian diikuti dengan perkembangan bahasa pemrograman berorientasi objek yang lainnya seperti C objek, C++, Eiffel, dan CLOS. Secara aktual, penggunaan bahasa pemrograman berorientasi objek pada saat itu masih terbatas, namun telah banyak menarik perhatian di saat itu. Sekitar lima tahun setelah Smalltalk berkembang, maka berkembang pula metode pengembangan berorientasi objek. Metode yang pertama diperkenalkan oleh Sally Shaler dan Stephen Mellor (Shaler-Mellor, 1988) dan Peter Coad dan Edward Yourdon (Coad-Yourdon, 1991), diikuti oleh Grady Booch (Booch, 1991), James R. Rumbaugh, Michael R. Blaha, William Lorensen, Frederick Eddy, William Premerlani (Rumbaugh-Blaha-Premerlani-Eddy-Lorensen, 1991), dan masih banyak lagi.

Buku terkenal yang juga berkembang selanjutnya adalah karangan Ivar Jacobson (Jacobson, 1992) yang menerangkan perbedaan pendekatan yang fokus pada *use case* dan proses pengembangan. Sekitar lima tahun kemudian muncul buku yang membahas mengenai metodologi berorientasi objek yang diikuti dengan buku-buku yang lainnya. Didalamnya juga membahas mengenai konsep, definisi, notasi, terminologi, dan proses mengenai metodologi berorientasi objek.

Karena banyaknya metodologi-metodologi yang berkembang pesat itu, maka muncul ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Usaha penyatuan ini

banyak mengambil dari metodologi-metodologi yang berkembang saat itu. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep seperti konsep *Object Modeling Technique* (OMT) dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities, Collaborators* (CRC) dari Rebecca Wirfs-Brock (1990), konsep pemikiran Ivar Jacobson, dan beberapa konsep lainya dimana James R. Rumbaigh, Grandy Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan *Rational Software Corporation* menghasilkan bahasa yang disebut dengan *Unified Modeling Language* (UML).

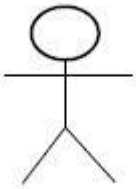
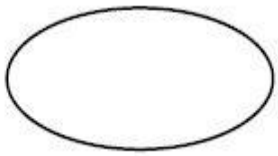
2.6.2 Diagram UML

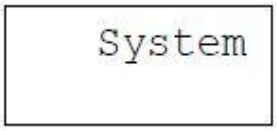
Berikut akan dijelaskan diagram yang akan digunakan dalam pembangunan sistem monitoring distribusi bantuan berbasis website ini, yaitu *use case diagram, class diagram, sequence diagram,* dan a*ctivity diagram.*

1. *Use Case Diagram*

   Use case diagram merupakan pemodelan untuk (*behavior)* aplikasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan aplikasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah aplikasi dan siapa saja yang berhak mengguankan fungsi-fungsi itu. Penjelasan tentang simbo-simbol *use case* dapat dilihat pada tabel 1.

Tabel 1 *Definisi Simbol Use Case Diagram*

Simbol	Nama Simbol	Fungsi
	Aktor	Merupakan orang, proses, atau sistem lain yang berinteraksi dengan aplikasi yang akan dibuat di luar aplikasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang
	*Use Case*	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case

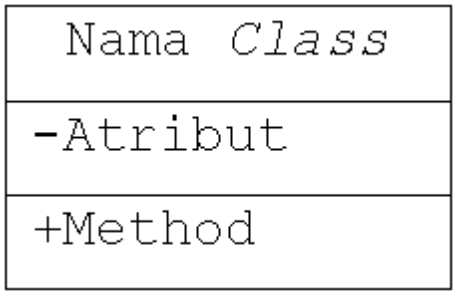
	Association	Merelasikan Aktor dengan *use case*
	*System Boundary*	Menggambarkan batasan sistem terhadap lingkungannya

2. *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut, merupakan variable-variabel yang dimiliki oleh satu kelas.
2. Operasi, atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Class diagram dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sesuai. Contoh *class diagram* dapat dilihat pada gambar II.3.



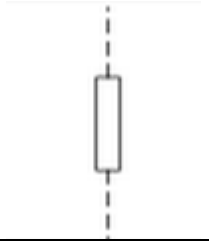
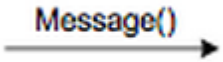
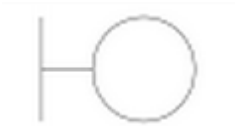
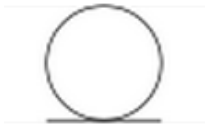
Gambar II.3 *Contoh Class Diagram*

3. *Sequence Diagram*

Sequence diagram menggambarkan *behavior* objek pada *use case* dengan mendeskripsikan waktu hidup dan pesan yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat *sequence diagram* juga dibutuhkan melihat scenario yang ada pada use case. Penjelasan tentang simbol-simbol *sequence diagram* dapat dilihat pada tabel 2.

Tabel 2 *Definisi Simbol Sequence Diagram*

Simbol	Nama Simbol	Fungsi

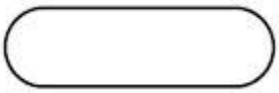
	*Lifelines*	Menggambarkan waktu mulai dan berakhirnya sebuah message
	*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktfitas yang terjadi
	*Actor*	Digunakan untuk menggambarkan user / pengguna
	*Boundary*	Digunakan untuk menggambarkan sebuah form atau antarmuka
	*Control Class*	Digunakan untuk menghubungkan boundary dengan entity
	*Entity Class*	Menyimpan data / informasi yang digunakan dalam perangkat lunak

4. *Activity Diagram*

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah *activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor. Tabel 3 adalah penjelasan tentang symbol-simbol *activity diagram*.

Tabel 3 *Definisi Simbol Activity Diagram*

Simbol	Nama Simbol	Fungsi
	*Initial State*	Pernyataan awal dari sebuah aktivitas

	*Final State*	Pernyataan akhir dari sebuah aktivitas
	*Transisi*	Menggambarkan alur dari aksi – aksi yang ada pada aktivitas
	Aksi	Menggambarkan aksi yang ada pada aktivitas

2.7 Bahasa Pemrograman Yang Digunakan

2.7.1 Hypertext Preprocessor (PHP)

PHP (*Hypertext Preprocessor)* yaitu bahasa pemrograman web server-side yang bersifat *open source*. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (*server side* HTML *embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman website 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/*up to date*. Semua script PHP dieksekusi pada server dimana *script* tersebut dijalankan.

Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.
Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, *interpreter* PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang *interpreter* PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis *interpreter* baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP (*Hypertext Preprocessing).*

Pada pertengahan tahun 1999, Zend merilis *interpreter* PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal

abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari *interpreter* PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

2.7.2 Database

Database atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak(program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, hubungan antar data yang tidak jelas dan juga update yang rumit.

2.7.3 My Structured Query Language (MySQL)

Menurut buku MySQL Reference Manual, MySQL (dibaca : *mai-ess-que-ell bukan mai-sekuel*) adalah suatu sistem manajemen database (*Database Mana gement System*) atau DBMS. Jadi, MySQL adalah sistem yang berguna untuk melakukan proses pengaturan koleksi-koleksi struktur data (*database*) baik yang meliputi proses pembuatan atau proses pengelolaan *database*. MySQL ini bersifat open source, artinya setiap orang dimungkinkan untuk menggunakan dan memodifikasinya. Program ini dapat diperoleh secara gratis pada alamat : http://www.mysg1.com. Sebenarnya MySQL ini suatu produk yang berjalan pada Linux, tetapi seiring berjalannya waktu dan pengetahuan maka para pengembang kernudian menu MySQL yang bisa diakses.

Beberapa kelebihan MySQL, antara lain:

1. MySOL dapat berjalan dengan stabil pada berbagai sistem operasi, seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, dan masih banyak lagi.

2. Bersifat Open Source, MySQL didistribusikan secara open source (gratis), di bawah lisensi GNU General Public Licence (GPL).

3. Bersifat Multiuser, MySOL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah.

4. MySQL memiliki kecepatan yang baik dalam menangani query (perintah SQL). Dengan kata lain, dapat memproses lebih banyak SQL per satuan waktu. Dan segi security atau keamanan data, MySQL memiliki beberapa lapisan sekuriti, seperti level *subnetmask,* nama *host* dan izin akses *user* dengan sistem perizinan yang mendetail serta *password* yang terenkripsi.[4]

2.7.4 Hyper Text Markup Languange (HTML)

HTML adalah bahasa pengkodeaan untuk menghasilkan dokumen-dokumen hypertext untuk digunakan di World Wide Web. HTML terlihat seperti bahasa pengkodean lama, yang di awali dan diakhiri dengan kode-kode html dimana menunjukkan bagaimana keluarannya pada saat di jalankan pada sebuah web browser.

2.7.5 Cascading Style Sheet (CSS)

CSS merupakan singkatan dari Cascading Style Sheet. CSS digunakan untuk mengubah tampilan (style) dari halaman web. Sebagaimana yang kita ketahui, halaman web modern terdiri dari 3 komponen dasar: HTML untuk membuat struktur, CSS untuk tampilan, dan JavaScript untuk interaksi.

Jika halaman web diibaratkan sebuah bangunan, CSS adalah tampilan luar dari bangunan tersebut, seperti warna dinding atau warna atap. Kerangka dasarnya dibuat dari HTML. Dengan demikian, kita bisa dengan mudah menukar warna dinding bangunan tanpa perlu mengubah struktur dasarnya.

2.8 Pengujian Beta dan Alfa

Apabila perangkat lunak dibuat untuk pelanggan maka dapat dilakukan aceeptance test sehingga memungkinkan pelanggan untuk memvalidasi seluruh keperluan. Test ini dilakukan karena memungkinkan pelanggan menemukan kesalahan yang lebih rinci dan membiasakan pelanggan memahami perangkat lunak yang telah dibuat.

2.8.1 Pengujian Alpha

Dilakukan pada sisi pengembang oleh seorang pelanggan. Perangkat Lunak digunakan pada setting yang natural dengan pengembang “yang memandang” melalui bahu pemakai dan merekam semua kesalahan dan masalah pemakaian.

2.8.2 Pengujian Beta

Dilakukan pada satu atau lebih pelanggan oleh pemakai akhir perangkat lunak dalam lingkungan yang sebenarnya, pengembang biasanya tidak ada pada pengujian ini. Pelanggan merekan semua masalah (real atau imajiner) yang ditemui selama pengujian dan melaporkan pada pengembang pada interval waktu tertentu.