BAB II

LANDASAN TEORI

Untuk menyelesaikan permasalahan diatas tentunya harus ada landasan teorinya. Adapun landasan teori dari pembuatan tugas akhir ini adalah :

2.1 Web Mining

Banyak yang percaya, Oren Etzioni adalah orang yang pertama mengusulkan tentang istilah *Web mining* pada paper yang dibuat pada tahun 1996 (Yan Wang, 2000). Pada paper tersebut, Oren Etzioni mengusulkan Web mining menggunakan teknik data mining untuk menemukan informasi yang bermanfaat di dalam dokumen dan service WWW secara otomatis. Tetapi walaupun Web Mining diturunkan dari Data Mining tetapi keduanya tidak bisa disamakan, Web mining jauh lebih komplek karena melibatkan data yang tidak tersruktur. Tahap-tahap yang dilakukan dalam Web Mining adalah sebagai berikut :

1. *Resource Discovery*, pada tahap ini dilakukan pencarian sumber dari informasi yang ingin diperoleh dari Web.
2. *Information Extraction*, secara otomatis memilih dan melakukan pre-processing untuk mendapatkan informasi yang spesifik dari sumber yang mudah di dapat sebelumnya.
3. *Generalization*, secara otomatis menemukan pola umum yang digunakan pada Web sites yang digunakan.
4. *Analysis*, menganalisa pola tersebut.

STIKOMP SURABAYA

Pendek kata, Web Mining adalah suatu teknik yang digunakan untuk menemukan informasi yang bermanfaat pada Web. Di dalam Web terdapat tiga tipe data yaitu *data on the Web* (content), *Web log data* (usage) dan *Web structure data*. Tiga tipe data tersebut dapat diklasifikasikan menjadi *content data*, *usage data*, *structure data* dan *user profile data*. Dari tiga tipe data yang terdapat di Web maka Web Mining dapat diklasifikasikan menjadi tiga jenis yaitu

1. Web Content Mining

Web content mining digunakan untuk mencari informasi dari sumber online atau dokumen Web yang tersedia, secara otomatis. Dokumen Web memiliki beberapa tipe data seperti text, image, audio, video, metadata dan hyperlinks. Beberapa data tersebut bersifat semi-terstruktur seperti dokumen HTML, atau yang lebih tersruktur seperti data pada table atau database yang di *generate* dari halaman HTML, tetapi kebanyakan dari data tersebut berbentuk data text yang tidak terstruktur. Karakteristik dari data Web yang tidak tersturktur memaksa Web content mining ke arah pendekatan yang lebih rumit.

2. Web Structure Mining

Kebanyakan dari *tools* yang digunakan untuk mencari informasi di web hanya menggunakan informasi yang berbentuk text dan mengabaikan link dari informasi yang bisa menjadi sangat bernilai. Tujuan dari Web structure mining adalah menghasilkan ringkasan yang terstruktur tentang Web site dan Web page. Teknik, Web content mining hanya memfokuskan pada struktur inti dari dokumen, sedangkan Web structure mining mencoba untuk menemukan struktur link dari hyperlinks yang terdapat pada dokumen. Berdasar pada topologi dari hyperlinks, Web structure

STIKOMPS SURABAYA

mining akan mengklasifikasikan Web pages dan men-*generate* informasi, seperti persamaan dan hubungan antara Web sites yang berlainan.

3. Web Usage Mining

Web usage mining mencoba untuk menemukan informasi yang bermanfaat dari data sekunder yang di dapat dari interaksi pengguna yang melakukan *browsing* di Web. Teknik ini menfokuskan pada bagaimana memprediksi perilaku dari pengguna internet saat mereka berinteraksi dengan Web. Tujuan dari teknik ini antara lain untuk memprediksi perilaku pengguna terhadap suatu Web site, membandingkan antara harapan dan kenyataaan dari penggunaan suatu Web site dan menyesuaikan isi Web site berdasarkan keinginan dari pengguna.

2.2 Web Content Management

Jika ingin membangun sebuah situs yang besar dan terintegrasi antara halaman yang satu dengan yang lainnya, membuatnya agar mempunyai tampilan yang menarik, dapat berubah-ubah setiap waktu, dapat menambahkan isi maupun menunya tanpa harus mengganti semua halaman satu per satu. Oleh karenanya, suatu situs akan membutuhkan suatu sistem yang dapat bekerja dan mengintegrasikannya dalam satu kesatuan yang terpisahkan. Sistem tersebut dapat kita sebut sebagai suatu *Web Content Management*.

Web Content Management secara sederhana dapat diartikan sebagai sebuah sistem yang memberikan kemudahan kepada para penggunanya dalam mengelola dan mengadakan perubahan isi sebuah website dinamis tanpa sebelumnya dibekali pengetahuan tentang hal-hal yang bersifat teknis. Dengan demikian, setiap orang,

STIKOM SURABAYA

penulis maupun editor, setiap saat dapat menggunakannya secara leluasa untuk membuat, menghapus atau bahkan memperbaharui isi website tanpa campurtangan langsung dari pihak webmaster.

Karena memisahkan antara isi dan desain, konsistensi tampilan dapat senantiasa dijaga dengan baik. Setiap bagian dari website dapat memiliki isi dan tampilan yang berbeda-beda, tanpa harus khawatir kehilangan identitas dari website secara keseluruhan. Oleh karena semua data disimpan dalam satu tempat, pemanfaatan kembali dari informasi yang ada untuk berbagai keperluan dapat dengan mudah dilakukan. Web Content Management juga memberikan kefleksibelen dalam mengatur alur kerja atau workflow dan hak akses, sehingga memperbesar kesempatan berpartisipasi dari pengguna dalam pengembangan website. Hal ini akan sangat menguntungkan bila website yang dikelola memiliki kompleksitas yang tinggi dan mengalami kemajuan yang cukup pesat.

2.3 Internet Agent

Pertama-tama diawali dengan mendefinisikan agent dari arti kamus. Di dalam kamus *Webster's New World Dictionary*, agent didefinisikan sebagai: *A person or thing that acts or is capable of acting or is empowered to act, for another* (Wahono, 2003). Disini ada dua point yang bisa diambil:

1. Agent mempunyai kemampuan untuk melakukan suatu tugas atau pekerjaan.
2. Agent melakukan suatu tugas atau pekerjaan dalam kapasitas untuk sesuatu, atau untuk orang lain.

STIKOMPS SURABAYA

Ditarik dari point-point diatas Caglayan (Caglayan dkk, 1997) mendefinisikan software agent sebagai suatu entitas software komputer yang memungkinkan user (pengguna) untuk mendelegasikan tugas kepadanya secara mandiri (*autonomous*).

Kemudian beberapa peneliti lain menambahkan satu point lagi, yaitu bahwa *agent* harus bias berjalan dalam kerangka lingkungan jaringan (*network environment*). Definisi *agent* dari para peneliti lain pada hakekatnya adalah senada, meskipun ada yang menambahkan atribut dan karakteristik *agent* ke dalam definisinya.

Untuk memperdalam pemahaman tentang *software agent*, fungsi, peran, dan perbedaan mendasar dikaitkan software program yang ada, berikut ini akan dijelaskan tentang beberapa atribute dan karakteristik yang dimiliki oleh *software agent*. Tentu tidak semua karakteristik dan atribut terangkum dalam satu *agent*. Pada hakekatnya daftar karakteristik dan atribut dibawah adalah merupakan hasil survei dari karakteristik yang dimiliki oleh *agent-agent* yang ada pada saat ini.

1. Autonomy

Agent dapat melakukan tugas secara mandiri dan tidak dipengaruhi secara langsung oleh user, *agent* lain ataupun oleh lingkungan (*environment*). Untuk mencapai tujuan dalam melakukan tugasnya secara mandiri, *agent* harus memiliki kemampuan kontrol terhadap setiap aksi yang mereka perbuat, baik aksi keluar maupun kedalam. Dan satu hal penting lagi yang mendukung autonomy adalah masalah intelegensi (*intelligence*) dari *agent*.

STIKOMPS SURABAYA

2. Intelligence, Reasoning, dan Learning

Setiap *agent* harus mempunyai standar minimum untuk bisa disebut *agent*, yaitu intelegensi (*intelligence*). Dalam konsep *intelligence*, ada tiga komponen yang harus dimiliki: internal *knowledge base*, kemampuan *reasoning* berdasar pada *knowledge base* yang dimiliki, dan kemampuan *learning* untuk beradaptasi dalam perubahan lingkungan.

3. Mobility dan Stationary

Khusus untuk *mobile agent*, harus memiliki kemampuan yang merupakan karakteristik tertinggi yang dimiliki yaitu mobilitas. Sebaliknya dari hal tersebut adalah *stationary agent*. Bagaimanapun juga keduanya tetap harus memiliki kemampuan untuk mengirim pesan dan berkomunikasi dengan *agent* lain.

4. Delegation

Sesuai dengan namanya dan seperti yang sudah kita bahas pada bagian definisi, *agent* bergerak dalam kerangka menjalankan tugas yang diperintahkan oleh user. Fenomena pendelegasian (*delegation*) ini adalah karakteristik utama suatu program disebut *agent*.

5. Reactivity

Karakteristik *agent* yang lain adalah kemampuan untuk bisa cepat

Menurut Nwana (Nwana, 1996), *Agent* bisa diklasifikasikan menjadi delapan sesuai dengan karakteristiknya termasuk didalamnya adalah *Internet Agent* antara lain:

STIKOMPS SURABAYA

1. Collaborative Agent

Agent yang memiliki kemampuan melakukan kolaborasi dan koordinasi antar agent dalam kerangka *Multi Agent System* (MAS).

2. Interface Agent

Agent yang memiliki kemampuan untuk berkolaborasi dengan user, melakukan fungsi *monitoring* dan *learning* untuk memenuhi kebutuhan user.

3. Mobile Agent

Agent yang memiliki kemampuan untuk bergerak dari suatu tempat ke tempat lain, dan secara mandiri melakukan tugas di tempat barunya tersebut, dalam lingkungan jaringan komputer.

4. Information dan Internet Agent

Agent yang memiliki kemampuan untuk menjelajah internet untuk melakukan pencarian, pemfilteran, dan penyajian informasi untuk user, secara mandiri. Atau dengan kata lain memanage informasi yang ada di dalam jaringan Internet.

5. Reactive Agent

Agent yang memiliki kemampuan untuk bisa cepat beradaptasi dengan lingkungan baru dimana dia berada.

6. Hybrid Agent

Agent yang memiliki katakteristik yang merupakan gabungan dari karakteristik yang sudah kita sebutkan sebelumnya adalah masuk ke dalam *hybrid agent*.

STIKOMPS SURABAYA

7. Heterogeneous Agent System

Dalam lingkungan *Multi Agent System* (MAS), apabila terdapat 2 atau lebih *hybrid agent* yang memiliki perbedaan kemampuan dan karakteristik, maka sistem MAS tersebut kita sebut dengan *heterogeneous agent system*.

2.4 Text Mining

Teknik yang umum dipakai untuk mengambil atau mengkais informasi yang bermanfaat dari sekumpulan data yang besar kita kenal dengan *Data Mining*. Software data mining yang terbaru yang telah dibuat pada tahun 2002 memungkinkan kita untuk mengambil informasi yang bermanfaat dari dokumen yang berbasis *text*. Proses atau teknik ini kita kenal dengan *Text Mining*.

Text Mining kita kenal sebagai Text data mining atau menemukan informasi dari dokumen atau data yang berbasis text, umumnya proses yang dilakukan adalah dengan menemukan text atau informasi yang diinginkan dengan pola yang *non-trivial* dari dokumen yang tidak tersturktur. Itu bisa dilihat sebagai pengembangan dari data mining yang menemukan informasi dari data yang terstruktur.

Saat ini kebanyakan informasi yang disimpan melaui form berbentuk text. Text Mining dipercaya memiliki nilai jual yang tinggi dibandingkan dengan Data Mining. Faktanya dari penelitian yang pernah dilakukan, 80% informasi dari perusahaan berbentuk dokumen text. Text mining lebih komplek dari Data mining karena melibatkan data text yang tidak tersturktur dan tidak jelas. Text mining biasanya melibatkan disiplin ilmu lain seperti text analysis, information extraction, clustering, categorization, visualization, database technology, machine learning dan data mining.

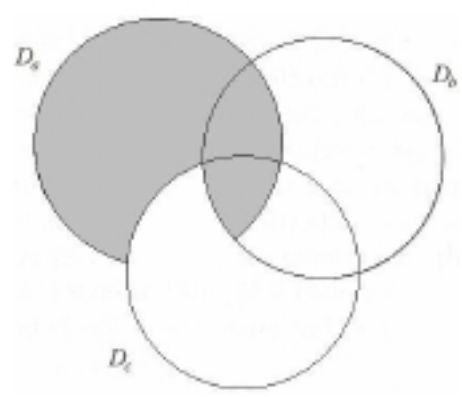
STIKOMPS SURABAYA

2.5 Fuzzy Search

Fuzzy search adalah suatu teknik pencarian yang memungkinkan untuk digunakan mencari informasi yang tidak pasti (Antikainen dkk, 1999). Kekuatan Fuzzy Search adalah dapat menemukan hasil yang tidak pasti tepat pada query. Logika Fuzzy Search melakukan pencarian berdasarkan pada persamaan karakter dari suatu kata bukan berdasarkan arti dari kata tersebut. Fuzzy Search digunakan terutama untuk pencarian text karena memiliki kemampuan untuk mengenali penulisan kata yang salah pada suatu dokumen. Hal ini sangat penting karena kesalahan sangat mungkin terjadi dalam suatu penulisan dokumen.

Fuzzy Search menemukan persamaan berdasarkan pada kata dasar. Ukuran ketepatan terhadap kata dasar ditentukan dengan parameter "*matchingLevel*". Parameter matchingLevel menentukan berapa persen huruf dari suatu kata harus terpenuhi agar bisa dikatakan tepat dengan kata yang dicari. Nilai *default* untuk parameter ini adalah 75%. Sebagai contoh, kata "Rossberg", 75% dari kata yang harus tepat adalah "Rossbe". Selain itu dengan Fuzzy Search juga bisa dicari kata yang sudah dilengkapi dengan awalan atau akhiran.

2.5.1 Fuzzy Set



Gambar 2.1. Dokumen Fuzzy Set untuk query $[q = k_a \wedge (k_b \vee k_c)]$

STIKOMPS SURABAYA

Dalam setiap model sistem *retrieval*, elemen umum yang biasa digunakan adalah dokumen, *keywords* dan *query*. Dokumen adalah sumber dimana sistem akan mencari keyword yang diinpukan. Dan *query* digunakan untuk memberitahukan keywords yang akan dicari dalam dokumen. Biasanya dokumen-dokumen tersebut disimpan dalam sutu database.

Fuzzy set pada gambar 1 adalah model dasar pada set teori dan operasi umum yang digunakan. Pada gambar 1 "D" dimisalkan dokumen dan "k" adalah keyword. Sebagai contoh pada gambar 1 terdapat tiga dokumen dan satu keyword dan query nya adalah [$q = k_a \wedge (k_b \vee k_c)$]. Artinya keyword muncul dalam dokumen a dan keyword tersebut juga muncul pada document atau dokumen c. Area yang berwarna abu-abu adalah hasil query.

Dengan menggunakan fuzzy set, batas yang digunakan untuk membedakan antara satu dokumen dengan dokumen yang lain tidak terlalu ketat. Fuzzy set memungkinkan dilakukan pencarian keyword dalam suatu dokumen walaupun keyword tersebut tidak benar-benar tepat ada dalam dokumen tersebut.

2.5.2 Algoritma Edit Distance

Algoritma Edit Distance adalah algoritma yang paling umum digunakan untuk menghitung jarak atau perbedaan antara kata yang diinputkan user dengan setiap kata yang ada dalam dokumen. Semakin kecil nilai tersebut maka perbedaan kata semakin kecil. Jika nilainya nol artinya kata tersebut sama dengan yang diinputkan oleh user. Algoritma tersebut digambarkan pada Gambar 2.2.

STIKOMPS SURABAYA

```
EditDistance( S, T )
{
  // Initialization
  For i=0 to n do D[i,0] = i;
  For i=0 to m do D[0,i] = i;

  // Recurrence
  For i=0 to n do
  {
    For j=1 to m do
    {
      sub = D[i-1,j-1] + matchcost(si,tj);
      ins = D[i-1,j] + insert cost;
      rem = D[i,j-1] + delete cost;

      D[i,j] = min( sub, ins, rem );
    }
  }
}
```

Gambar 2.2 Algoritma Edit Distance

Algoritma diatas memberikan gambaran bagaimana jarak antara kata dihitung. Parameter dari algoritma tersebut adalah sumber kata (S) dan target kata (T). Pertama, algoritma tersebut akan menghitung jumlah huruf dari target kata dan sumber kata yang digunakan untuk membentuk matrix yang dapat dilihat pada gambar 3. Kemudian dihitung berapa huruf yang harus diubah, dihapus dan disisipi untuk mendapatkan kata yang benar.

STIKOMP SURABAYA

		r	e	s	r	i	v	e	r
	0	1	2	3	4	5	6	7	8
r	1	0	1	2	3	4	5	6	7
e	2	1	0	1	2	3	4	5	6
t	3	2	1	1	2	3	4	5	6
r	4	3	2	2	1	2	3	4	5
i	5	4	3	3	2	1	2	3	4
e	6	5	4	4	3	2	2	2	3
v	7	6	5	5	4	3	2	3	[illegible]
e	8	7	6	6	5	4	3	[illegible]	3

Gambar 2.3. Matrix yang dibentuk dengan Algoritma Edit Distance

Sebagai contoh pada gambar diatas dibandingkan dua buah kata yaitu "*resriver*" dan "*retrieval*". Kata pada baris horisontal adalah kata yang diinputkan dan kata pada baris vertikal adalah kata yang benar. Nilai pada baris diagonal menjelaskan berapa banyak huruf harus diubah, dihapus atau dimasukkan atau disisipi dari kata yang diinputkan untuk mendapatkan kata yang benar. Hasil dari perhitungan edit distance pada contoh diatas dapat dilihat pada bagian hitam, seperti terlihat pada Gambar 2.3. Nilai edit distancenya yaitu Edit Distance (8,8) = 3.

Untuk menghitung tingkat kesalahannya, dengan membagi hasil perhitungan edit distance dengan jumlah huruf kata yang dibandingkan. Adapun rumus perhitungan tingkat kesalahnnya adalah sebagai berikut :

$$\alpha = \frac{\text{Nilai Edit Distance}}{\text{Panjang String}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

STIKOMPS SURABAYA

Dimana α = Tingkat Kesalahan (Error).

2.6 Association Rule

Association Rule adalah metode untuk menganalisa data yang memiliki kecenderungan penggunaan item tertentu untuk mendapatkan kemungkinan keluaran yang memiliki tingkat frequensi penggunaan item tertinggi. Association Rule mencari semua transaksi dalam database yang mengekspresikan item X terhadap item Y atau X → Y. Dimana X dan Y adalah set dari item. Dapat diartikan menjadi X ∩ Y = [1,2]. Diberikan set pada transaksi A, untuk kemudian men-*generate* semua Association Rules yang pasti atau mendekati nilai minimal spesefik dari support.

Pemecahan masalahnya dapat diturunkan menjadi 2 hal, yaitu :

1. Menemukan semua itemset pada *transaction* support.
2. Menemukan *Frequency Itemset* dengan untuk menghasilkan ragam kemungkinan atau kesimpulan.

2.6.1 Menemukan Semua Itemset Pada Transaction Support

Support pada pola Association Rule mengacu pada prosentase proses transaksi data yang relevan dimana pola tersebut benar.

$$\text{Support (A... Z)} = \frac{\sum \text{transaction containts item}}{\sum \text{transaction}} \quad \text{.....(2)}$$

Kumpulan support dari masing-masing item dalam itemset, kemudian dihitung prosentase kehadirannya.

$$\text{Support (A... Z)} = \left(\frac{\sum \text{transaction containts item}}{\sum \text{transaction}} \right) \text{ X } 100\% \quad \text{.....(3)}$$

STIKOMPS SURABAYA

Setelah menemukan presentasi support dari itemsets, maka kemudian dilanjutkan ke proses selanjutnya yaitu menemukan Frekuensi Itemset.

2.6.2 Menemukan Frekuensi Itemset

|A| = m, menjadi jumlah dari item. Ruang pencarian dari penjumlahan semua frekuensi itemset adalah 2^m , dengan eksponensial m. Satu cara membuktikan bahwa masalah dalam pencarian kumpulan frekuensi dari ukuran tertentu adalah NP-*complete*, dengan menguranginya untuk kesimpangan masalah diantara dua kelompok. Dari formula frequency itemset dibawah, maka akan ditemukan hasil simpulan.

$$\text{Frequency Itemset} = \left(\frac{\sum \text{candidate item}}{\sum \text{itemset transaction}} \right) \text{X } 100\% \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

2.7 Regex

Menurut Steven Haryanto (2004:2), *Regular expression* (ekspresi regular, regexp, regex, RE) adalah sebuah bahasa mini untuk mendeskripsikan string atau teks. Regex dapat dipakai untuk mencocokkan sebuah string dengan sebuah pola. Pencocokkan string dengan regex jauh lebih ampuh. Selain menguji kecocokan atau mencari substring dalam string, regex juga dapat dipakai mengekstrak string dari teks. Terakhir regex juga bisa dipakai untuk membelah dan mensubtitusi substring dengan string lain.

STIKOMPS SURABAYA

Keberadaaan regex di berbagai tool dan bahasa pemrograman membuktikan bahwa regex sangat berguna dalam mengolah teks. Alasan yang dapat menjawab kenapa menggunakan regex adalah sebagai berikut :

1. Regex sangat ampuh, pola regex dapat digunakan untuk mencocokkan dan mengekstrak pola string mulai dari angka, nomor telepon, kata, URL, hingga pola yang didefinisikan sendiri. Regex juga dapat mengambil potongan string yang diinginkan dari file log, dokumen HTML, bahkan file biner sekalipun.
2. Regex amat ringkas, artinya regex dapat menggantikan belasan hingga puluhan baris kode program hanya dengan sebuah pola.
3. Regex cepat, artinya dengan menggunakan regex puluhan atau ratusan perbandingan string secara manual dapat dihindari. Library regex yang ada sekarang ini, rata-rata sudah teroptimasi sehingga pola yang komleks sekalipun bias dijalankan dengan cepat.

PHP memiliki dua set fungsi regex. Yang pertama adalah fungsi-fungsi dengan prefiks `ereg_` seperti `ereg_match()`, `eregi_replace()` dan lain sebagainya. Huruf e pada reg artinya extended. Artinya, fungsi-fungsi regex ini adalah fungsi regex flavor POSIX. Fungsi-fungsi `ereg_*()` tidak akan dibahas lebih lanjut karena keterbatasan sintaks regex POSIX. Yang kedua adalah fungsi-fungsi dengan prefiks `preg_`, seperti `preg_match()` dan `preg_replace()`. Fungsi-fungsi `preg_*()` menggunakan *library* PCRE yang mendekati sintaks Perl. Berikut ini fungsi-fungsi yang disediakan PHP untuk menggunakan PCRE beserta sintaksnya:

STIKOMP SURABAYA

```
int preg_match(str pola, str subyek[,array metches[,flag]]);
int preg_match_all(str pola, str subyek, array matches[,int order]);
mixed preg_replace(mixed pola, mixed pengganti, mixed subyek[,int
limit]);
mixed preg_replace_callback(mixed pola, callback, mixed subyek[,int
limit]);
array preg_split(str pola,o str subyek[,int limit[,int flags]]);
```

Gambar 2.4. Fungsi-fungsi Regex dengan library PCRE pada PHP

2.8 PHP

Menurut dokumen resmi PHP (Abdul Kadir, 2002), PHP singkatan dari *PHP Hypertext Preprocessor*. PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasilnya akan dikirim ke klien, tempat pemakai menggunakan *browser*. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yaitu pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi *tool* yang disebut "Personal Home Page". Paket inilah yang menjadi cikal-bakal PHP.

PHP adalah *server side scripting environment* yang dapat digunakan untuk membuat dan menjalankan aplikasi – aplikasi di web server agar menjadi lebih interaktif dan *programmable*. Dengan PHP aplikasi – aplikasi yang ada di web server benar – benar akan dijalankan di web server tanpa mengharuskan adanya tambahan atau syarat tertentu untuk sisi client (web browser). PHP biasanya dijadikan sebagai module dalam suatu web server agar bisa mengeksekusi file-file PHP yang tersedia di

STIKOMP SURABAYA

web server. PHP dapat berjalan di hampir seluruh platform, open source dan berlisensi Gnu Public License (GPL).

Sebagai tambahan untuk memanipulasi isi dari halaman web, PHP juga dapat mengirimkan HTTP header yang dapat digunakan untuk setting cookies, mengatur proses authentikasi dan meredirect user. PHP juga mempunyai koneksi dengan banyak database termasuk dengan ODBC serta berintegrasi dengan keragam library eksternal yang membantu web developer untuk melakukan sesuatunya. PHP menyatu dengan halaman web sehingga tidak dibutuhkan aplikasi khusus untuk membuatnya. Secara sintaks PHP serupa dengan bahasa C dan Perl. Web developer tidak harus mendeklarasikan variable sebelum menggunakannya dan dengan PHP mudah untuk membuat array dan hash (array berasosiasi).

2.9 MySQL

MySQL adalah sebuah aplikasi SQL database server yang multi-user, multithread. Tujuan utama dari MySQL adalah kecepatan dan mudah penggunaannya. Ini yang menyebabkan banyak aplikasi – aplikasi web yang berbasiskan database selalu menggunakan MySQL sebagai database enginenya.

Beberapa karakteristik yang dimiliki oleh MySQL server :

1. Sepenuhnya *multi-threaded* dengan menggunakan kernel threads yang berarti dapat dengan mudah menggunakan banyak CPU yang tersedia.
2. Aplication Program Interface (API) untuk bahasa C, C++, Eiffel, Java, Perl, PHP, Python dan TCL.
3. Bekerja pada banyak platform sistem operasi.

STIKOMP SURABAYA

4. Mendukung penuh operator dan fungsi dalam sintak SELECT dan WHERE sebagi bagian query.
5. Pembatasan hak dan password yang sangat fleksible dan aman serta mendukung verifikasi berdasarkan host.
6. Password aman karena dilakukan enkripsi ketika password dikirim ke server.
7. Mampu mengolah database yang besar.

STIKOMP SURABAYA