**BAB II**

**LANDASAN TEORI**

* 1. **Penelitian Terdahulu**

Penelitian terdahulu menerapkan sistem informasi evaluasi terhadap evaluasi penilaian kinerja guru di Jepara dengan menggunakan *teacher performance assessment instrument* yang meliputi *teaching plans and materials, classroom procedure* dan *interpersonal skill*, sistem evaluasi terhadap pelaporan monografi pada kantor Kecamatan Semarang Utara dibuat karena pelaporan masih berupa pencatatan secara manual, sulitnya pada saat mencari data dan sering mengalami keterlambatan, sistem evaluasi Himpunan Petani Pengguna Air (HIPPA) Jawa Timur untuk monitoring pelaporan sehingga penilaian terhadap kelembagaan HIPPA dapat dilakukan dengan cepat, serta penelitian terdahulu sistem informasi evaluasi terhadap kegiatan dosen di Fakultas Informatika Institut Teknologi Telkom, data yang digunakan yaitu data beban kinerja dosen.

* 1. **Sistem**

Asal kata Sistem berasal dari bahasa Latin *systema*  dan bahasa Yunani *sustema.*Sistem adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan.  Atau dapat juga dikatakan bahwa sistem adalah

sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.Suatu sistem sangatlah dibutuhkan dalam suatu perusahaan atau instansi pemerintahan, karena sistem sangatlah menunjang terhadap kinerja perusahaan atau instansi pemerintah, baik yang berskala kecil maupun besar.Supaya dapat berjalan dengan baik diperlukan kerjasama diantara unsur-unsur yang terkait dalam sistem tersebut.

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu”. Istilah sistem secara umum dapat didefinisikan sebagai kumpulan hal atau elemen yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan. Sistem mempunyai karakteristik atau sifat – sifat tertentu, yaitu : Komponen Sistem, Batasan Sistem, Lingkungan Luar Sistem Penghubung Sistem, Masukan Sistem, Keluaran Sistem, Pengolahan Sistem dan Sasaran Sistem. Jadi, secara umum Pengertian Sistem adalah perangkat unsur yang teratur saling berkaitan sehingga membentuk suatu totalitas. (Jogianto, 2005)

* 1. **Sistem Informasi**

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sistem informasi merupakan suatu perkumpulan data yang terorganisasi beserta tatacara penggunaanya yang mencangkup lebih jauh dari pada sekedar penyajian.Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tatacara penggunaanya. Keberhasilan suatu sistem informasi yang diukur berdasarkan maksud pembuatanya tergantung pada tiga faktor utama, yaitu : keserasian dan mutu data, pengorganisasian data, dan tatacara penggunaanya.untuk memenuhi permintaan penggunaan tertentu, maka struktur dan cara kerja sistem informasi berbeda-beda bergantung pada macam keperluan atau macam permintaan yang harus dipenuhi.

Suatu persamaan yang menonjol ialah suatu sistem informasi menggabungkan berbagai ragam data yang dikumpulkan dari berbagai sumber.Untuk dapat menggabungkan data yang berasal dari berbagai sumber suatu sistem alih berupa transformation data sehingga jadi tergabungkan atau compatible. Berapapun ukurannya dan apapun ruang lingkupnya suatu sistem informasi perlu memiliki ketergabungan compatibility data yang disimpannya. (Arbi, 2000)

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.Sistem informasi adalah sekumpulan prosedur organisasi yang dilaksanakan untuk mencapai suatu tujuan yaitu memberikan informasi bagi pengambil keputusan dan untuk mengendalikan organisasi.

Informasi dalam lingkup sistem informasi memiliki beberapa ciri  yaitu:

1. Baru, informasi yang didapat sama sekali baru dan segar bagi penerima.
2. Tambahan, informasi dapat memperbaharui atau memberikan tambahan pada informasi yang telah ada.
3. Korektif, informasi dapat menjadi suatu koreksi atas informasi yang salah sebelumnya.
4. Penegas, informasi dapat mempertegas informasi yang telah ada.
   1. **Dasar Teori**
      1. **Pengertian Sistem Informasi**

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. (Jogiyanto.HM, 1999:11).

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah *blok* bangunan (*building block*) yaitu *blok* masukan (*input block*), *blok* model *(model block), blok* keluaran *(output block), blok* teknologi *(technology block), blok* basis data (*database block*) dan *blok* kendali (*control block*). Sebagai suatu sistem, keenam *blok* tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk suatu kesatuan untuk mencapai sasarannya (Jogiyanto.HM, 1999:12).

* + 1. **Internet dan Web**

Pada dasarnya *internet* adalah jaringan komputer yang besar, dimana masing- masing komputer saling berhubungan dan saling berbagi TCP/IP (Transmission *Control Protocol/Internet Protocol). Internet berfungsi sebagai pengantar informasi* yang berada dalam antar komputer, ini dapat disebut sebagai layanan komputer dalam komunikasi global. Internet merupakan jaringan kumpulan dari beberapa jaringan yang tersebar diseluruh dunia.

*Internet* adalah kumpulan komputer antara satu wilayah dan wilayah lainnya yang terkait dan saling berkomunikasi, dimana letak keterkaitan dan komunikasi ini diatur oleh protokol sebagai pengaturnya.

*World Wide Web* adalah sekelompok dokumen multimedia yang terkoneksi menggunakan hypertext Link.

Setiap pribadi atau organisasi perusahaan yang memiliki informasi di *World Wide Web (WWW)* memerlukan sebuah alamat atau *Uniform Resource Locator (URL)* khusus yang disebut *Website.* Setiap informasi yang disimpan dalam suatufile yang berbeda disebut web page. Dalam page tersimpan berbagai informasi lain baik dalam web page yang sama atau web page lain pada website yang berbeda.

Sebelum mengakses berbagai informasi pada suatu website, yang petama dijumpai adalah halaman web pembuka yang disebut homepage. Homepage ini merupakan halaman pertama dari suatu website yang berisi apa dan siapa dari perusahaan atau organisasi pemilik website tersebut. Homepage tersebut juga berisi link-link ke dokumen informasi lebih lanjut tentang isi website tersebut. Pada dasarnya homepage adalah sarana untuk memperkenalkan secara singkat tentang apa yang menjadi isi dari keseluruhan website dari suatu organisasi atau pribadi.

Web menggunakan protocol yang disebut HTTP (*Hypertext Transfer Protocol*) yang berjalan pada TCP/IP. Dokumen Web ditulis dalam format HTML (*Hypertext Markup Language*). Dokumen ini diletakkan dalam Web server (server yang melayani permintaan halaman Web) dan diakses oleh klien (pengakses informasi) melalui perangkat lunak atau yang sering disebut *browser*.

* 1. **Pengertian Basis Data**

Basis data dapat didefinisikan dalam sejumlah sudut pandang, seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang tersimpan aecara bersama sedemikian rupa dan tanpa pengulangan (*redudansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file* atau tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis (Fathansyah, CV Informatika 1999).

Ada Beberapa Definisi Dalam Basis Data Diantaranya yaitu :

1. *File* adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda-beda datanya.
2. *Entity* adalah orang, tempat, kejadian atau konsep yang informasinya direkam.
3. *Record* adalah kumpulan elemen-elemen yang saling berkaitan dan menginformasikan tentang suatu *entity* secara lengkap. Satu *record* mewakili satu data atau informasi.

Dalam implementasi pemrograman object atau Entity dalam database tersebut biasanya dituangkan dalam bentuk tabel, sedangkan atribut yang ada didalamnya dituliskan dalam bentuk field, didalam tabel juga berisi satu atau beberapa record. Jika atribut merupakan karakteristik atau sifat yang dimiliki oleh sebuah entity, record dapat didefinisikan sebagai data yang dimiliki oleh entity tersebut.

* 1. **UML (*Unified Modeling Language*)**

*Unified Modelling Language* (UML) adalah salah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi obyek (Munawar, 2005:18). 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 mengkomuni- kasikan rancangan mereka dengan yang lain.

UML merupakan kesatuan dari bahasa pemodelan yang dikembang-kan 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.* Untuk perancangan sebuah aplikasi dengaan menggunakan UML, UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Tipe diagram UML dapat dilihat pada table 2.1 sebagai berikut.

**Tabel 2.1** Tipe Diagram UML

|  |  |
| --- | --- |
| Diagram | Tujuan |
| *Activity* | Perilaku prosedural dan parallel |
| *Class* | *Class*, Fitur dan relasinya |
| *Communication* | Interaksi diantara obyek. Lebih menekankan ke  Link |
| *Component* | Struktur dan koneksi dari komponen |
| *Composite structure* | Dekomposisi sebuah class saat *runtime* |
| *Deployment* | Penyebaran/ instalasi ke klien |
| *Interaction Overview* | Gabungan antara *activity* dan *sequence* diagram |
| *Object* | Contoh konfigurasi *instance* |
| *Package* | Struktur hierarki saat kompilasi |
| *Sequence* | Interaksi antara obyek. Lebih menekankan pada  Urutan |
| *State Machine* | Bagaimana *event* mengubah sebuah obyek |
| *Timing* | Interkasi antara obyek. Lebih menekankan pada  Watu |
| *Use Case* | Bagaimana *User* berintaraksi dengan sebuah  System |

* + 1. **Pengertian Use Case**

**Use case adalah** kegiatanatau urutan interaksiyang saling berkaitan antara sistem. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Use case juga digunakan untuk membentuk perilaku (behavior) sistem yang akan dibuat. Sebuah use case menggambarkan sebuah interkasi antara pengguna (aktor) dengan sistem yang ada (Kusumadewi, 2003).

* + 1. **Tujuan Use Case**

1. Memetakkan kebutuhan sistem
2. Mempresentasikan interaksi pengguna terhadap sistem
3. Untuk mengetahui kebutuhan diluar sistem
   * 1. **Deskripsi Use Case**

a. Diagram use case merupakan pemodelan untuk menggambarkan kelakuan (behavior) sistem yang akan dibuat.

b. Diagram use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat.

c. Diagram use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Yang ditekankan pada diagram ini adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”.

d. Sebuah use case merepresentasikan sebuah interaksi antara aktor (user atau sistem lainya) dengan sistem.

e. Use case menjelaskan secara sederhana fungsi sistem dari sudut pandang user.

Sukamto dan Shalahuddin (2013:155), “Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat”. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

* + 1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan

sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat

itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor

belum tentu merupakan orang.

* + 1. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unityang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram use case dapat dilihat pada tabel 2.2 menurut Rosa dan Shalahuddin (2013:156):

**Tabel 2.2** Simbol-Simbol Diagram Use Case

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan** |
| 1 | *Use case* | Fungsionalisasi yang di sediakan system sebagai untit-unit yang saling bertukar pesan antar unit atau aktor, biasanya di nyatakan dengan menggunakan kata kerja di awal frase name use case. |
| 2 | Aktor /actor | Orang, proses atau sistem yang lain yang berinteraksi dengan system informasi yang akan di buat diluar sistemyang akan di buat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tetapi aktor belum tentu menggunakan orang; biasanya di nyatakan menggunakan kata benda di awal frase nama actor |
| 3 | Asosiasi / association | Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor |
| 4 | Ekstensi / extend | Relasi*use case* tambahan ke sebuah *use case* dimana use case yang di tambahkan dapat berdiri sendiri walaupun tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemograman beroriantasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang di tambahakan misal arah panah mengarah pada *use case* yang di tambahakan |
| 5 |  | Hubungan generalisasi dan spesialisasi antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya |
| 6 |  | Relasi *use case* tambahakan ke subuah *use* *case* dimana *use case* di tambhakan memerlukan *use case* ini menjalakan fungsinya atau syarat di jalankan *use case* ini |

* + 1. **Diagram Aktivitas (*Diagram Activity*)**

Sukamto dan Shalahuddin (2013:161), Diagram activity atau diagram aktivitas menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas dapat dilihat pada tabel 2.3 menurut Rosa dan Shalahuddin (2013:162:­­

**Tabel 2.3** Simbol-Simbol Diagram Aktivity

|  |  |  |
| --- | --- | --- |
| No | **Simbol** | **Deskripsi** |
| 1 | Status awal | Status awal aktivitas sistem, sebuah  diagram aktivitas memilih sebuah status awal. |
| 2 | Aktivitas | Aktivitas yang di lakukan sistem aktivitas  biasanya di awali dengan kata kerja |
| 3 | Percabangan / *decision* | Asosiasi percabangan dimana ada pilihan  aktivitas lebih dari |
| 4 | Penggabungan / *join* | Asosiasi penggabungan dimana lebih satu  aktivitas di gabung menjadi satu |
| 5 | Status akhir | Status akhir yang di lakukan sistem sebuah  diagram aktivitas memiliki sebuah status akhir |

* + 1. **Diagram Kelas (*Class Diagram*)**

Sukamto dan Shalahuddin (2013:141), “class diagram atau Diagram kelas 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 variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

1. Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

1. Kelas yang diambil dari pendefinisian use case (*controller*)

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

1. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas dapat dilihat pada tabel 2.4 menurut Rosa dan Shalahuddin (2013:146):

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Deskripsi** |
| 1 | |  | | --- | | **nama\_kelas** | | +atribut | | +operasi() | | Kelas pada struktur sistem. |
| 2 | Antarmuka / *Interface*    **nama\_interface** | Sama dengan konsep *interface* dalampemrograman berorientasi objek. |
| 3 | Asosiasi / *Association* | Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.* |
| 4 | Asosiasi berarah /  *Directed association* | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.* |
| 5 | Generalisasi | Relasi antarkelas dengan makna generalisasi - spesialisasi (umum khusus). |
| 6 | Kebergantungan /  *Dependency* | Relasi antarkelas dengan makna kebergantungan antarkelas. |
| 7 |  | Relasi antarkelas dengan makna semua-bagian (*whole-part*). |

**Tabel 2.4** Simbol-Simbol Diagram Kelas

* + 1. ***Sequence Diagram* (Diagram Sekuen)**

Sukamto dan Shalahuddin (2013:165), *Sequence diagram* atau diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirim dan diterima antar objek. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah simbol-simbol yang ada pada diagram sekuen dapat lihat pada tabel 2.5 menurut Sukamto dan Shalahuddin (2013:165):

**Tabel 2.5** Simbol-Simbol Diagram Sekuen

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan** |
| 1 | Aktor  atau  tanpa waktu aktif | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 2 | Garis hidup / *Lifeline* | Menyatakan kehidupan suatu objek. |
| 3 | Objek | Menyatakan objek yang berinteraksi pesan. |
| 4 | Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |
| 5 | Pesan tipe *create* | Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat. |
| 6 | Pesan tipe *call* | Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri. |
| 7 | Pesan tipe *send* | Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi. |
| 8 | Pesan tipe *return*  *1: keluaran* | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |
| 9 | Pesan tipe *destroy* | Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri,sebaiknya jika ada *create* maka ada *destroy* |

* 1. **Konsep Pemrograman HTML**

HTML (Hypertext Markup Language) merupakan suatu bahasa pemrograman untuk membangun sebuah dokumen dalam bentuk halaman di web (walaupun banyak orang yang keberatan menyebut HTML sebagai bahasa pemrograman karena dirasa terlalu sederhana), hanya terdiri atas tag <html> sampat </html>. Selain itu juga, HTML tidak memiliki ’looping’ seperti bahasa pemrograman lain.

Namun dalam hal ini, tidak bisa dipungkiri bahwa HTML selalu ada pada sebuah halaman web page, karena HTML dapat dipergunakan sebagai hypertext link yang mampu membentuk hubungan antara teks dan dokumen lain.

Elemen pada HTML didefinisikan sebagai suatu kode tertentu yang akan menyediakan tempat untuk meletakkan beberapa kode didalamnya.

Skema elemen HTML adalah sebagai berikut:

<html>

<head>

<title> </title>

</head>

<body>

</body>

</html>

Keterangan :

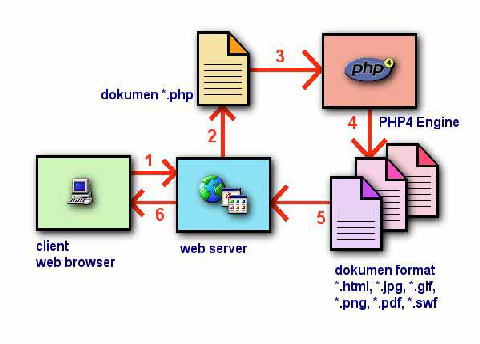
1. Dokumen HTML selalu diawali dengan tag pembuka <html> dan diakhiri dengan tag penutuh </html>.
2. Teks yang terdapat diantara tag <head> dan </head> merupakan informasi header. Informasi ini tidak ditampilkan pada windows browser.
3. Teks yang terdapat diantara tag <title> dan </title> merupakan judul yang akan ditampilkan pada dokumen.
4. Bagian body merupakan isi dokumen HTML. Pada tag ini, semua informasi, baik itu teks, gambar, sound, maupun lainnya dapat ditampilkan pada halaman ini. Teks yang terdapat di antara tag <body> dan </body> merupakan teks atau informasi yang akan ditampilkan pada browser.
   1. **Konsep Pemrograman PHP**

PHP merupakan singkatan dari *Hypertext Preprocessor*, merupakan bahasa script server-side yang disisipkan pada HTML (*Hypertext Markup language*). Kemampuan PHP yang paling diandalkan dan signifikan adalah dukungan kepada banyak database. Membuat halaman web yang menggunakan data dari data base dengan sangat mudah untuk dilakukan.

PHP dibuat pertama kali oleh seorang yang bernama Rasmus Lerdorf yang pada awalnya dibuat untuk menghitung jumlah pengunjung pada homepagenya. Diawal Januari 2001 PHP telah dipakai oleh 5 (lima) juta domain diseluruh dunia dan ini akan terus bertambah karena mudahnya aplikasi PHP dibandingkan dengan bahasa Pemrograman web yang lainnya.

Saat ini PHP menjadi sangat populer dan telah menggantikan Perl yang sebelumnya juga populer sebagai salah satu bahasa scripting web. PHP telah menjadi modul Apache terpopuler dan melebihi Frontpage dan Perl sehingga PHP hingga saat ini terus meningkat penggunanya bahkan telah digunakan oleh jutaan domain dan jutaan alamat IP. PHP telah digunakan oleh berbagai situs populer baik dalam maupun luar negeri.

Interpreter PHP dalam mengeksekusi kode PHP pada sisi server (disebut server-side) dan berbeda dengan mesin maya Java yang mengeksekusi program pada sisi client (client-side). Proses eksekusi kode PHP yang disisipkan pada halaman HTML secara diagram dapat digambar sebagai berikut.



Gambar 2.10 Proses Eksekusi Kode PHP Yang Disisipkan Pada Halaman HTML

Secara singkat , kelebihan-kelebihan PHP meliputi :

1. Script PHP sederhana , mudah dibuat dan mempunyai kecepatan akses tinggi.
2. Dapat berjalan dalam web server yang berbeda dan dalam sistem operasi yang berbeda. PHP dapat berjalan pada sistem operasi Linux/Unix , Windows , dan Macintosh.
3. Bersifat *Open Source* sehingga diterbitkan secara gratis.
4. Dapat berjalan pada server web Microsoft Personal Web server, Apache, IIS, Xitami dan sebagainya.
5. Termasuk bahasa yang *embedded* (bisa ditempel atau diletakkan dalam tag HTML)
   1. **Apache**

Webserver merupakan suatu server internet yang menggunakan protocol HTTP (*Hypertext Transfer Protocol* ) untuk melayani semua proses pentransferan data.

Web server Apache merupakan salah satu web server yang paling banyak digunakan oleh para pembuat situs dalam internet yang berfungsi sebagai pengendali program yang berbasis internet.Dalam web server apache mempunyai beberapa variabel yang mempunyai fungsi tersendiri untuk operasi program yang dijalankan .

* 1. **Database MySQL**

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user, serta menggunakan perintah standard SQL (*Structured Query Language*). MySQL pertama kali dirintis oleh seorang programmer database bernama Michael Widenius.

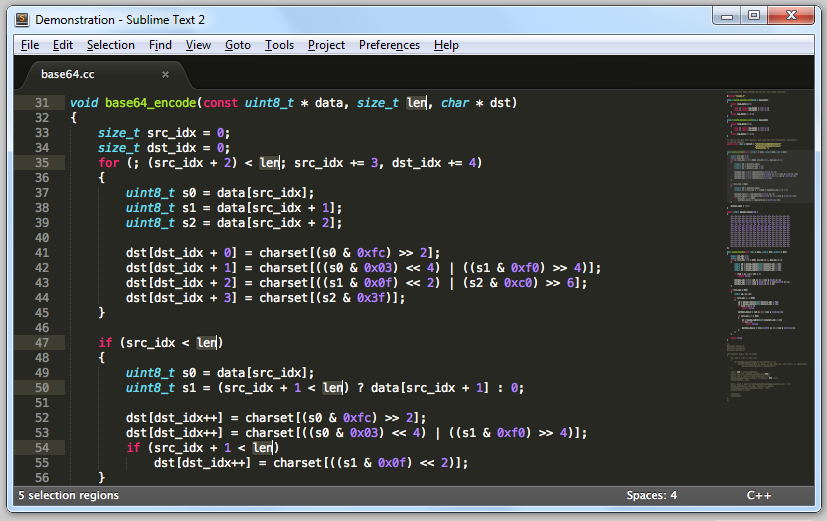
Database MySQL, biasa dibaca mai-es-ki-el atau bisa juga mai-se-kuel, merupakan suatu perangkat lunak database yang berbentuk database relational atau dalam bahasa basisdata sering disebut dengan *Relation Database Management System* (RDBMS) yang menggunakan suatu bahasa permintaan bernama SQL.

Perbedaan antara MySQL dan SQL adalah : MySQL merupakan program database *server* yang memerlukan bahasa permintaan dalam melayani permintaan *user*, SQL merupakan bahasa permintaan yang digunakan dalam MySQL.

* 1. Sublime Text

Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim, Aplikasi ini sangatlah fleksibel dan powerfull. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan sublime-packages. Sublime Text bukanlah aplikasi open source dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, akan tetapi beberapa fitur pengembangan fungsionalitas (packages) dari aplikasi ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki linsensi aplikasi gratis.

Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur syntax highlight hampir di semua bahasa pemrogramman yang didukung ataupun dikembangkan oleh komunitas seperti; C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML. Biasanya bagi bahasa pemrograman yang didukung ataupun belum terdukung secara default dapat lebih dimaksimalkan atau didukung dengan menggunakan add-ons yang bisa didownload sesuai kebutuhan user.



**Gambar 2.8** Sublime Text

Berikut beberapa fitur yang diunggulkan dari aplikasi Sublime Text:

1. *Goto* *Anything*

Fitur yang sangat membantu dalam membuka file ataupun menjelajahi isi dari file hanya dengan beberapa *keystrokes*.

1. *Multiple* *Selections*

Fitur ini memungkinkan *user* untuk mengubah secara interaktif banyak baris sekaligus, engubah nama variabel dengan mudah, dan memanipulasi file lebih cepat dari sebelumnya.

1. *Command* *Pallete*

Dengan hanya beberapa keystorkes, user dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalu menu.

1. *Distraction* *Free* *Mode*

Bila user memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu *user* dengan memberikan tampilan layar penuh.

1. *Split* *Editing*

Dapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan *editing* perpecahan. Mengedit sisi file dengan sisi, atau mengedit dua lokasi di satu file. Anda dapat mengedit dengan banyak baris dan kolom yang *user* inginkan.

1. *Instant* *Project* *Switch*

Menangkap semua file yang dimasukkan kedalam *project* pada aplikasi ini. Terintegrasi dengan fitur *Goto Anything* untuk menjelajahi semua file yang ada ataupun untuk beralih ke file dalam *project* lainnya dengan cepat.

1. *Plugin* *API*

Dilengkapi dengan plugin API berbasis Phyton sehingga membuat aplikasi ini sangat tangguh.

1. *Customize* *Anything*

Aplikasi ini memberikan *user* fleksibilitas dalam hal pengaturan fungsional dalam aplkasi ini.

1. *Cross* *Platform*

Aplikasi ini dapat berjalan hampir disemua *operating system* modern seperti Windows, OS X, dan Linux *based operating system*.