

BAB II

KAJIAN PUSTAKA

2.1 Pendahuluan

Efisiensi dari suatu hotel adalah penentuan tarif kamar yang sesuai dengan fasilitas yang ditawarkan. Semakin banyak fasilitas yang ditawarkan namun memiliki tarif kamar yang kompetitif, membuat hotel tersebut menjadi pilihan bagi pelanggan. Penelitian ini menganalisa efisiensi tarif kamar sesuai dengan fasilitas yang ada. Keunggulan tarif dan fasilitas yang dimiliki memberikan nilai lebih bagi masyarakat dalam menentukan pilihannya dalam memilih sebuah hotel.

Penelitian ini merupakan pengembangan dari penelitian yang telah dilakukan oleh Dillah Hikmah Sari pada tahun 2006. Pada penelitian tersebut, peneliti menentukan hotel terbaik diantara hotel berbintang empat di Yogyakarta tanpa memberikan evaluasi efisiensi dan kinerja sensitivitas pada hotel yang tidak terpilih. Sedangkan pada penelitian ini penulis mencoba menindaklanjuti hasil pemilihan hotel terbaik terhadap efisiensi tarif berdasarkan fasilitas yang dimiliki hotel dengan menggunakan metode DEA (*Data Envelopment Analysis*) yang terintegrasi dengan *interface* berbasis *web* desain. Penelitian ini bertujuan untuk memberikan kemudahan bagi pelanggan hotel dalam mencari informasi untuk menentukan pilihan dan juga memberikan informasi mengenai evaluasi kinerja sensitivitas kepada manajemen hotel.

Penelitian sebelumnya yang menggunakan DEA adalah Muhammad Farouq Syahrir, (2008). penelitian ini dilakukan dengan tujuan menganalisa tingkat efisiensi Stasiun Telephon Otomat (STO) dari layanan Telkom Speedy

wilayah kota Yogyakarta yang mempunyai *input* dan *output* yang beragam secara kuantitatif.

Peneliti lain yang menggunakan DEA adalah Antoni Rahardjo (2009) meneliti tentang Efisiensi Antara Harga Dan Fitur *Handphone* Menggunakan Metode *The Analytical Hierarchy Process* (AHP) Dan *Data Envelopment Analysis* (DEA).

2.2 Hotel

2.2.1 Pengertian Hotel

Menurut keputusan Menteri Pariwisata, Pos, dan Telekomunikasi (No. KM37/PW-340/MPPT-86) diartikan sebagai suatu jenis akomodasi yang menggunakan seluruh atau sebagian bangunan untuk menyediakan jasa pelayanan penginapan, makanan, dan minuman serta jasa lainnya bagi umum yang dikelola secara komersil serta memenuhi ketentuan persyaratan yang ditetapkan di dalam keputusan pemerintah.

Dari pengertian tersebut dapat disimpulkan bahwa hotel harus menyediakan produk-produk tertentu baik yang bersifat nyata (*tangibles*) maupun yang bersifat tidak nyata (*untangibles*) serta fasilitas-fasilitas tertentu untuk memenuhi kebutuhan konsumennya, diantaranya adalah fasilitas kamar, fasilitas pelayanan makanan, dan minuman serta fasilitas penunjang lainnya.

2.2.2 Penggolongan Hotel Berbintang

Penggolongan hotel dibagi menjadi lima dan setiap tiga tahun Departemen Pariwisata, Pos, dan Telekomunikasi memberikan penggolongan bagi hotel berbintang sebagai berikut:

a. Hotel Bintang Satu (*)

1. Jumlah kamar *standard* minimal 15 kamar.
2. Dilengkapi kamar mandi dalam.
3. Luas kamar *standard* minimal 20 m².

b. Hotel Bintang Dua (**)

1. Jumlah kamar *standard* minimal 20 kamar.
2. Jumlah kamar *suite* minimal satu kamar.
3. Dilengkapi kamar mandi dalam.
4. Luas kamar *standard* minimal 20 m².
5. Luas kamar *suite* minimal 48 m².

c. Hotel Bintang Tiga (***)

1. Jumlah kamar *standard* minimal 30 kamar.
2. Jumlah kamar *suite* minimal dua kamar.
3. Dilengkapi kamar mandi dalam.
4. Luas kamar *standard* minimal 24 m².
5. Luas kamar *suite* minimal 48 m².

d. Hotel Bintang Empat (****)

1. Jumlah kamar *standard* minimal 50 kamar.
2. Jumlah kamar *suite* minimal tiga kamar.
3. Dilengkapi kamar mandi dalam.
4. Luas kamar *standard* minimal 24 m².

5. Luas kamar *suite* minimal 48 m².
- e. Hotel Bintang Lima (*****)
1. Jumlah kamar *standard* minimal 100 kamar.
 2. Jumlah kamar *suite* minimal empat kamar.
 3. Dilengkapi kamar mandi dalam.
 4. Luas kamar *standard* minimal 26 m².
 5. Luas kamar *suite* minimal 52 m².

2.2.3 Kategori Kamar Berdasarkan Tarif Kamar

Tidak semua kamar hotel sama, baik dari segi fasilitas, tarif maupun luas kamar. Pengelompokan kamar berdasarkan tarif adalah sebagai berikut:

a. *Suite*

Suite room berkategori SU, biasanya terdiri dari dua ruangan atau lebih, yaitu sebuah parlon (ruang tambahan) dan *sleeping room*.

b. *Deluxe*

Jenis kamar berkategori A yang umumnya mewakili tarif kamar maksimum yang dibayarkan kepada hotel untuk ukuran regular kamar tidur. Tipe kamar ini biasanya memberikan yang terbaik dari pemilihan lokasi, pemandangan, tingkat kenyamanan tinggi, perlengkapan, dan dekorasi yang indah serta menarik. Fasilitas yang disediakan berupa *minibar* atau kulkas, sedangkan tempat tidurnya biasanya sejenis *king bed*.

c. *Superrior*

Kamar berkategori B ini biasanya ditempatkan pada *ground floor* serta tempat tidurnya dari jenis *double* atau *king bed*.

d. *Standard*

Kamar bertipe *standard* atau berkategori C ini tipikal tempat tidur yang disediakan meliputi satu atau dua buah (*double bed*). Jenis kamar ini biasanya menawarkan paling sedikit hal menarik, termasuk lokasi.

e. *Economy*

Kamar untuk ekonomi berkategori P yang disediakan bagi suatu situasi yang mungkin terjadi bila menghadapi penghuni yang meluap. Tarif diberikan potongan harga atau pada saat promosi khusus. Kamar-kamar yang ditawarkan dengan ukuran tempat tidur yang lebih kecil dan lokasi yang kurang nyaman seperti di sebelah ruang *laundry*.

2.3 Efisiensi

2.3.1 Konsep Efisiensi

Efisiensi adalah tingkat maksimum penggunaan sumber produktifitas seperti tenaga kerja dan alat mesin dalam menyelesaikan pekerjaan cetak. Pengukuran efisiensi dilakukan dengan membandingkan kegiatan produksi aktual terhadap standar produksi.

Konsep lain dari efisiensi adalah '*Technical Efficiency*', yang mempunyai arti merubah beberapa *input* (seperti tenaga kerja, pendapatan) menjadi *output* dengan level performa yang tinggi. Penggunaan *input* dimanfaatkan semaksimal mungkin untuk menghasilkan jumlah *output* tertentu. (Shaooth *et.al*, 2006). Efisiensi diartikan juga sebagai gambaran sistem dengan performa yang baik dalam memaksimalkan *output* dari *input*.



'*Technical Efficiency*' dapat dicapai dengan sempurna (100%) jika dan hanya jika dalam satu unit usaha tidak ada *input* atau *output* yang ditingkatkan tanpa menjadikan *input* dan *output* yang lain menjadi lebih buruk (Cooper *et.al*, 2003). Artinya, sebuah unit usaha dikatakan '*technical efficiency*' saat tidak dapat menaikkan beberapa *output* atau mengurangi beberapa *input* tanpa menghilangkan *output* lain atau meningkatkan *input* yang lain. Efisien dalam menggunakan masukan (*input*) akan menghasilkan produktifitas yang tinggi, yang merupakan tujuan dari setiap organisasi apapun bidang kegiatannya. Hal yang paling rawan adalah apabila efisiensi selalu diartikan sebagai penghematan karena bisa mengganggu operasi, sehingga pada gilirannya akan mengganggu hasil akhir karena sasarannya tidak tercapai dan produktifitasnya juga akan tidak setinggi yang diharapkan (Suwandi, 2007).

Efisiensi juga bisa diartikan sebagai rasio antara *output* dengan *input*. Ada tiga faktor yang menyebabkan efisiensi (Suwandi, 2005), yaitu:

- a. Apabila dengan *input* yang sama dapat menghasilkan *output* yang lebih besar.
- b. *Input* yang lebih kecil dapat menghasilkan *output* yang sama.
- c. Dengan *input* yang lebih besar dapat menghasilkan *output* yang lebih besar lagi.

2.3.2 Efisiensi Relatif

Dari beberapa pengertian manajemen atau pengetahuan sosial secara teori sulit untuk mengetahui level dari efisiensi. Definisi dari efisiensi dapat diwakili dengan menggunakan informasi yang dapat digunakan untuk

mengukur atau sebagai parameter dari yang diteliti. Efisiensi relatif dapat didefinisikan bila dalam satu unit kerja atau pelayanan mencapai efisiensi yang penuh (100%) jika dan hanya jika kinerja dari unit kerja/ layanan yang lain tidak menunjukkan peningkatan *input* atau *output* tanpa menjadikan *input* dan 'i' yang lain menjadi lebih buruk. Dalam penelitian ini perihal yang dijadikan sebagai target untuk menganalisa nilai efisiensi relatif, yaitu efisiensi dari tiap DMU.

Pada efisiensi relatif DMU, pengukuran nilai efisien tergantung pada *input* dan *output* yang ditentukan. *Input* yang ada dalam penelitian ini adalah fasilitas hotel, ketersediaan fasilitas yang ada di dalam kamar. Sedangkan untuk *output* adalah tarif kamar dari tiap-tiap DMU.

2.4 Kuisisioner dan *Sampling*

2.4.1 Kuisisioner

Yaitu pengumpulan data dengan cara menyebarkan kumpulan pertanyaan yang berisi hal-hal yang ingin diteliti, dalam hal ini adalah tentang penilaian dan harapan konsumen terhadap pelayanan yang diberikan perusahaan.

Skala *Likert* merupakan skala yang berisi beberapa tingkat jawaban mengenai kepentingan responden terhadap suatu pernyataan yang dikemukakan mendahului opsi jawaban yang disediakan. Adapun sebagian dari jenis skala yang paling banyak digunakan dalam pengukuran kepuasan pelanggan (Gerson, 2004 : 68-69) adalah:

a. Skala 5 poin

1	2	3	4	5
Buruk (jauh di bawah harapan)	Agak baik (di bawah harapan)	Sedang (memenuhi harapan)	Baik (di atas harapan)	Sangat baik (jauh di atas harapan)

b. Skala 7 poin

1	2	3	4	5	6	7
Sangat buruk (tidak memuaskan)	Buruk	Agak buruk	Netral	Agak baik	Baik	Sangat baik (memuaskan)

c. Skala 10 poin (tambahkan poin 0 agar menjadi skala 11 poin)

1	2	3	4	5	6	7	8	9	10
Sangat tidak penting									Amat sangat penting

d. Skala 100 poin (tambahkan poin 0 agar menjadi skala 101 poin)



2.4.2 Sampling

Data dapat diambil secara keseluruhan pada suatu populasi dengan cara sensus. Jika ada keterbatasan kemampuan dengan cara sensus, maka dapat diusahakan dengan mengambil sebagian saja data dari populasi yang ada dengan cara *sampling*. Metode yang digunakan adalah *sampling* Aksidental *sampling* untuk konsumen, yaitu individu-individu yang

dijadikan sampel adalah konsumen yang kebetulan ada ditempat penelitian untuk dijadikan sebagai sumber data.

Jumlah sampel untuk responden ditentukan dengan menggunakan rumus (Supranto 1992) :

$$n = P(1-P) \left[\frac{Z_{\alpha/2}}{SE} \right]^2 \dots\dots\dots(2.1)$$

Dimana :

n = jumlah sampel SE = *Sampling Error*

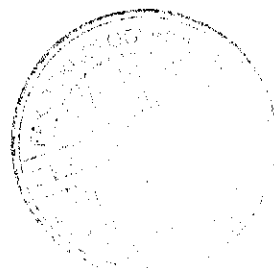
P = Proporsi sebenarnya dari populasi Z $\alpha/2$ = Tingkat keyakinan

2.5 Teknik Pengujian Instrumen

Ada dua syarat penting yang berlaku pada sebuah angket/ kuisioner, yaitu keharusan sebuah angket untuk *valid* dan *reliabel*.

2.5.1 Uji Validitas

Kesahihan (validitas) adalah tingkat kemampuan suatu instrumen untuk mengungkapkan sesuatu yang menjadi sasaran pokok pengukuran yang dilakukan dengan instrumen tersebut (Sutrisno Hadi, 1995). Suatu angket/ kuisioner dikatakan valid (sah) jika pertanyaan pada suatu angket mampu untuk mengungkapkan sesuatu yang akan diukur oleh angket tersebut. Analisis kesahihan butir dilakukan bertujuan untuk menguji apakah tiap-tiap butir pertanyaan telah mengungkapkan faktor yang ingin diselidiki sesuai dengan kondisi populasinya. Suatu butir dikatakan sah apabila korelasi butir dengan faktor positif dan peluang ralat p dari korelasi tersebut



maksimal 5%. Pengujian terhadap validitas item dalam penelitian ini dilakukan dengan menggunakan bantuan *software* SPSS 15.0.

2.5.2 Uji Reliabilitas

Setelah uji validitas dilakukan, maka langkah selanjutnya adalah melakukan uji reliabilitas. Instrumen tersebut harus *reliable*, artinya konstan didalam pengambilan data. Pengujian ini berkaitan dengan masalah adanya kepercayaan terhadap alat tes (instrumen). Suatu instrumen dapat memiliki tingkat kepercayaan yang tinggi jika hasil pengujian instrumen tersebut menunjukkan hasil yang tetap. Dengan demikian, masalah reliabilitas tes atau instrumen berhubungan dengan masalah ketetapan hasil. Jika terjadi perubahan pada hasil tes atau instrumen, maka perubahan tersebut dianggap tidak berarti.

Sutrisno Hadi (1995) juga mengatakan bahwa uji reliabilitas digunakan untuk menunjukkan sejauh mana alat pengukur (instrumen) dapat memperlihatkan kemantapan, keajegan, atau stabilitas hasil pengamatan bila diukur dengan instrumen tersebut dalam waktu berikutnya dengan kondisi tetap yang apabila diukur tidak terjadi perubahan. Keandalan berarti bahwa berapa kali pun atribut-atribut kuisioner ditanyakan kepada responden yang berlainan, hasilnya tidak akan menyimpang terlalu jauh dari rata-rata jawaban responden untuk atribut tersebut. Sama halnya dengan pengujian validitas di atas, pengujian reliabilitas ini juga dilakukan dengan menggunakan bantuan *software* SPSS 15.0.

2.6 Data Envelopment Analysis (DEA)

Data Envelopment Analysis (DEA) adalah sebuah pendekatan non parametrik yang pada dasarnya merupakan teknik berbasis *linear programming*. DEA bekerja dengan langkah identifikasi unit yang akan dievaluasi, *input* yang dibutuhkan serta *output* yang dihasilkan unit tersebut. Kemudian membentuk *efficiency frontier* atas set data yang tersedia dan menghitung nilai produktifitas dari unit-unit yang tidak termasuk dalam *efficiency frontier* serta mengidentifikasi unit mana yang tidak menggunakan *input* secara efisien relatif terhadap unit berkinerja terbaik dari set data yang dianalisa.

Metode ini diciptakan sebagai alat evaluasi kinerja suatu aktivitas sebuah unit entitas. Secara sederhana pengukuran dinyatakan dengan rasio :

$$Efficiency = \frac{output}{input} \dots\dots\dots(2.2)$$

yang merupakan satuan pengukuran produktifitas yang bisa dinyatakan secara parsial (misalnya: *output* per jam kerja ataupun *output* per pekerja, dengan *output* adalah penjualan, profit, dsb), ataupun secara total (melibatkan semua *output* dan *input* suatu entitas ke dalam pengukuran) yang dapat membantu menunjukkan faktor *input* (*output*) apa yang paling berpengaruh dalam menghasilkan suatu *output* (penggunaan suatu *input*). Hanya saja perluasan pengukuran produktifitas dari parsial ke total akan membawa kesulitan dalam memilih *input* dan *output* apa yang harus disertakan dan bagaimana pembobotannya. Cooper *et.al*, (2003).

Penggunaan bobot yang bersifat *fixed* yang diterapkan secara seragam pada semua *input* dan *output* dari entitas yang dievaluasi dikenal sebagai konsep *Total Factor Productivity* dalam ekonomi. Konsep ini berlawanan dengan

terkadang dalam praktek manajemen dapat menentukan bobot sebelumnya.

- f. *Homogeneity* : DEA menuntut seluruh DMU yang dievaluasi memiliki variabel *input* dan *output* yang sama jenisnya.

2.6.1 Prinsip Pokok *Data Envelopment Analysis*

Dalam menyelesaikan persoalan dengan DEA ada prinsip-prinsip yang harus dipahami, diantaranya adalah :

- a. *Input*
- b. *Output*
- c. *Efficiency*
- d. *Decision Making Unit (DMU)*

Kumpulan dari entitas yang akan dievaluasi merubah *multiple input* ke *multiple output*. Karena DEA memiliki banyak DMU, secara umum dapat dikatakan bahwa DMU satu harus lebih efisien dari DMU yang lain.

2.6.2 Langkah-langkah DEA

Sub bab ini membahas metode atau langkah-langkah penelitian yang digunakan dalam menganalisa dan memecahkan masalah (Orita, 2005) :

- a. Klasifikasi Pemilihan DMU (*Decision Making Unit*).
- b. Tahap Identifikasi Faktor yang Berpengaruh: diperoleh berdasarkan hasil *brainstormings*. Ada 42 faktor yang berpengaruh dalam proses pengolahan data.

- c. Tahap Pengelompokan *Input* dan *Output*: diperoleh berdasarkan faktor-faktor yang mempengaruhi proses pengolahan data. Untuk *output* terdiri dari 4 faktor dan *input* terdiri dari 38 faktor.
- d. Mengidentifikasi Model: dilakukan berdasarkan spesifikasi model dan sifat dari *input* dan *output* data.
- e. Pengumpulan Data.
- f. Pengolahan Data dan Analisa Data.

2.6.3 Perhitungan Matematis

Model dasar dari DEA adalah *Linear Programming*. *Linear programming* adalah model matematika yang digunakan untuk mengoptimalkan kegunaan suatu utilitas atau departemen dalam satu organisasi dengan sumber yang terbatas. Model *Linear Programming* (LP) mempunyai tiga elemen dasar (Taha, Hamdy A., 1997), yaitu :

- a. *Decision Variable*
- b. *Objective (goal)*
- c. *Constraint*

Selain variabel yang akan dimaksimal atau diminimalkan, dalam variabel keputusan juga terdapat variabel *slack* dan *surplus*. Variabel *slack* adalah variabel yang berfungsi untuk menampung sisa kapasitas atau kapasitas yang tidak digunakan pada kendala yang berupa pembatas. Variabel *slack* pada setiap kendala aktif pasti bersifat nol dan variabel *slack* pada setiap kendala tidak aktif pasti bersifat tidak aktif. Variabel *surplus* adalah variabel yang berfungsi untuk menampung kelebihan nilai ruas kiri pada kendala yang berupa syarat (Siswanto, 2007).

Formulasi (2.2) diatas dapat digunakan bila hanya terdapat satu *input* dan *output*. Pengukuran efisiensi relatif berdasarkan probabilitas yang tidak seimbang antara jumlah *input* dan *output* diperkenalkan oleh Farrel pada tahun 1957 (Emrouznejad , 1996). Persamaan umum adalah :

$$\text{Efisiensi} = \frac{\sum \text{output}}{\sum \text{input}} \dots\dots\dots(2.3)$$

Secara matematis hubungan diatas dapat dirumuskan menjadi :

$$\text{Efisiensi} = \frac{\sum_{r=1}^s u_r y_{ro}}{\sum_{i=1}^m v_i x_{io}} \leq 1 \dots\dots\dots(2.4)$$

$$\text{Dengan syarat : } \frac{\sum_{r=1}^s u_r y_{rj}}{\sum_{i=1}^m v_i x_{ij}} \leq 1; \quad j = 1, 2, \dots, n; \quad \varepsilon > 0 \dots\dots\dots(2.5)$$

Keterangan: $j = \text{DMU}, j = 1, 2, \dots, n$

$i = \text{input}, i = 1, 2, \dots, m$

$r = \text{Output}, r = 1, 2, \dots, s$

y_{rj} = nilai *output* ke-r dari DMU ke-j

x_{ij} = nilai dari *input* ke-r dari DMU ke-j

ε = angka positif yang kecil

u_r, v_i = bobot untuk *output* r , *input* i ($\geq \varepsilon$)

Permasalahan di atas tidak dapat diselesaikan secara terpisah karena berhubungan dengan *nonlinear (fractional) programming*. Charnes, Cooper, dan Rhodes pada tahun 1978 mengembangkan transformasi matematika dengan mengkonversikan persamaan matematika non linear diatas menjadi linear. Teori *duality* dan *algoritma simplex* yang terdapat dalam *linear programming* digunakan sebagai pemecah permasalahan (Shahooth *et.al*, 2006) menjadi:

Objection function :

$$\max h_k = \sum_{r=1}^s u_r y_{rj} \dots\dots\dots (2.6)$$

Subject to :

$$\sum_{i=1}^m v_i x_{ij} = 1 \dots\dots\dots (2.7)$$

$$\sum_{r=1}^s u_r y_{rj} - \sum_{i=1}^m v_i x_{ij} \leq 0 \dots\dots\dots (2.8)$$

$$u_r, v_i \geq \varepsilon \dots\dots\dots (2.9)$$

Keterangan : h_k = efisiensi relatif DMU_k

Transformasi ini dikembangkan untuk *fraktional program constrain*

$\sum v_i x_{ij} = 1$, berarti jumlah semua *input* adalah sama dengan 1 disebut dengan *CCR primal* model (Tsai *et.al*, 2006).

Tujuan dari formulasi diatas adalah untuk menentukan jumlah terbesar *output* yang dibobotkan dari DMU_k dengan menjaga jumlah dari *input* yang dibobotkan pada DMU agar rasio antar *output* yang dibobotkan dengan *input* yang dibobotkan bernilai kurang dari satu atau sama dengan

satu. Untuk *program linear* semakin banyak *constrain* maka semakin sulit untuk dipecahkan. Pada DEA terdapat cara untuk mengurangi jumlah *constrain* dalam model, pengurangan ini bertujuan sebagai target untuk memperbaiki produktifitas berdasarkan *input oriented* dan *output oriented* (Moses *et.al*, 2008). Model tersebut disebut dengan CCR *Dual Model*.

2.7 Perangkat Lunak

Agar dapat menjalankan aplikasi *web*, minimal harus tersedia dua *software* utama yaitu *web server* dan *web browser*. *Web server* merupakan tempat yang melayani *request*, sementara *web browser* digunakan untuk menampilkan hasil pada *user* (Andi Sunyoto, 2007). Komponen dasar yang akan digunakan dalam penelitian ini antara lain:

- a. *Web server*
- b. *Database server*
- c. Bahasa pemrograman

2.7.1 XHTML (*Extensible HyperText Markup Language*)

XHTML adalah kombinasi dari HTML dan XML. XHTML adalah bentuk HTML yang didefinisikan sebagai aplikasi XML yang berarti XHTML berisi semua elemen dalam HTML 4.0 dikombinasikan dengan *sintaks* XML.

XML adalah *Markup language* dimana semua harus dibuat secara benar sehingga dapat menghasilkan dokumen yang '*well-formed*'. XML didesain untuk mendeskripsikan data, sedangkan HTML didesain untuk menampilkan data. XHTML dibuat dengan aturan yang ketat dalam hal penulisan. Salah satu kelebihan XHTML adalah halaman XHTML dapat

dibaca pada peralatan yang mensupport XML seperti pada telepon selular dan PDA.

XHTML mengharuskan penulisan tag dengan menggunakan huruf kecil (*lowercase*) dan selalu menutup menggunakan penutup tag. Yang membedakan sistematika penulisan antara HTML dan XHTML (Andi Sunyoto, 2007) adalah:

- Semua elemen sebaiknya saling tersarang.
- Elemen XHTML harus selalu ditutup.
- Elemen XHTML harus ditulis dengan huruf kecil.
- Dokumen XHTML harus mempunyai satu *root* elemen.
- Nama atribut *lowercase*.
- Nilai atribut harus menggunakan tanda petik.
- Atribut tidak dapat disederhanakan.
- Atribut '*name*' diganti dengan atribut '*ID*'.

Elemen dasar yang wajib dimiliki XHTML adalah:

- *Doctype* yang terdiri atas *strict*, *transitional*, dan *frameset*.
- *Head*
- *Body*

2.7.2 Javascript

JavaScript adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti *IE*, *Mozilla Firefox*, *Netscape*, dan *Opera*. Kode *JavaScript* dapat disisipkan dalam halaman *web* menggunakan *tag script* (Andi Sunyoto, 2007). Beberapa hal tentang *JavaScript*:

- a. *JavaScript* didesain untuk menambah interaktif suatu *web*.
- b. *JavaScript* merupakan sebuah bahasa *scripting*.
- c. Bahasa *Scripting* merupakan bahasa pemrograman yang ringan.
- d. *JavaScript* berisi baris kode yang dijalankan di komputer (*web browser*).
- e. *JavaScript* biasanya disisipkan (*embedded*) dalam halaman HTML.
- f. *JavaScript* adalah bahasa interpreter (yang berarti skrip dieksekusi tanpa proses kompilasi).
- g. Setiap orang dapat menggunakan *JavaScript* tanpa membayar lisensi.

Variabel digunakan untuk menyimpan data atau informasi. Sebuah nilai variabel dapat diubah. Berikut ini adalah ketentuan nama variabel dalam *JavaScript*:

- Terdiri atas rangkaian huruf, angka, dan *underscore*.
- Harus diawali dengan huruf.
- *Case-sensitive* (membedakan huruf kecil dan huruf besar).
- Tidak harus deklarasi variabel dan tercipta sendiri ketika pertama kali digunakan.
- Berbentuk varian (tidak terikat pada satu tipe data, tetapi dapat diisi dengan tipe data yang berbeda dengan awalnya ketika pertama kali digunakan).

2.7.3 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang *powerfull* untuk membuat halaman *web* yang dinamis dan interaktif (Andi

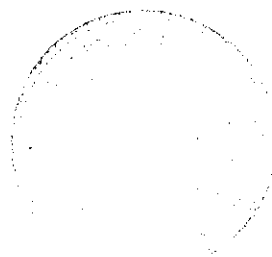
Sunyoto, 2007). PHP dikembangkan sebagai bahasa skrip *side-server programming* yang bersifat *open source* yang dapat digunakan dengan gratis serta dapat digabungkan dengan berbagai *server* yang berbeda-beda. PHP mempunyai kemampuan dapat mengakses *database* dan diintegrasikan dengan HTML. PHP semakin populer dikalangan pemakai karena memiliki kelebihan, antara lain:

- a. Mudah dibuat dan dijalankan.
- b. Mampu berjalan pada *web server* dengan sistem operasi yang berbeda-beda (UNIX, *windows*, dan *machintos*).
- c. PHP bisa didapatkan secara gratis.
- d. Dapat berjalan pada *web server* yang berbeda-beda (*Microsoft personal web server, Apache, IIS, Xitami, dll*).
- e. Dapat di *embedded*, dapat diletakkan dalam *tag* HTML.

Proses eksekusi PHP:

- a. User meminta *request* via *web browser* ke *web server*.
- b. *Web server* akan mengenali sebagai *file* PHP (.php, .php3, .phtml).
- c. *File* PHP dikirim ke *engine* PHP.
- d. *Output* dalam proses *engine* PHP dalam bentuk *plain* HTML.
- e. Dokumen hasil eksekusi dikirim balik ke *web server*.
- f. *Web server* meneruskan ke *browser* sebagai respons dari *request* untuk ditampilkan

Pada pembuatan *comment* dalam PHP, digunakan tanda // untuk *single line* (satu baris) dan /* untuk kumpulan kode program. Sedangkan variabel PHP tidak dideklarasikan dan bertipe *dynamic*. Tipe data di PHP dibagi menjadi:



- a. Tipe *scalar*: *boolean*, *integer*, *double*, *string*.
- b. Tipe data gabungan: *array*, objek.
- c. Tipe spesial: *resource*, *NULL*.
- d. Semua nama variabel diawali dengan tanda \$ (dolar)

Function adalah sebuah blok perintah yang dapat dijalankan kapan saja ketika diperlukan. Sebuah *function* dapat diatur dengan atau tanpa parameter. Parameter akan dimasukkan pada saat *function* dipanggil. *Function* juga dapat digunakan untuk mengembalikan nilai.

Variabel *string* digunakan untuk nilai berupa karakter *string*. Ada beberapa *built-in function string* yang sering digunakan untuk memanipulasi *string* di PHP. Setelah membuat *string* maka *string* tersebut dapat dimanipulasi, baik langsung menggunakan *function* maupun disimpan di dalam sebuah variabel. Berikut ini adalah fungsi-fungsi *string*:

- a. *Strlen*: mengembalikan nilai panjang *string*.
- b. *Strpos*: mencari *string* atau karakter di dalam *string* dan mengembalikan posisinya.
- c. *Substr*: mengambil sebagian isi *string*.
- d. *Strtolower*: mengembalikan nilai *string* menjadi huruf kecil semua.
- e. *Strtoupper*: mengembalikan nilai *string* menjadi huruf kecil semua.
- f. *Trim*: menghilangkan spasi di awal dan di akhir *string*.

2.7.4 My SQL

SQL (*Structured Query Language*) merupakan sebuah bahasa standar yang digunakan untuk mengakses dan melakukan manipulasi sistem *database*. *Statement* dalam SQL dapat digunakan untuk mengakses data atau meng *update* data pada suatu *database* (Yakub, 2008). Sedangkan My SQL merupakan sebuah *database* yang mendefinisikan struktur untuk menyimpan informasi. My SQL dapat melakukan proses *query* dan menentukan informasi yang diinginkan. Nilai yang dikembalikan dari proses *query* berupa *recordset*.

My SQL banyak digunakan untuk mengembangkan aplikasi dengan skala besar maupun kecil. Secara *de-facto* banyak aplikasi *web* yang menggunakannya. Beberapa *website* yang menggunakan My SQL antara lain adalah *friendster*, *yahoo*, dan *google*.

2.7.5 Koneksi PHP ke Database (My SQL)

My SQL sering digunakan bersama PHP. Sebelum dapat mengakses dan bekerja dalam *database*, terlebih dahulu adalah membuat koneksi. Dalam PHP hal ini dapat dilakukan dengan *function mysql_connect()* *function*. Untuk membuka koneksi dapat menggunakan perintah *mysql_connect*, sedangkan untuk menutup koneksi digunakan perintah *mysql_close()*.