

BAB II

LANDASAN TEORI

2.1 Sistem Pendukung Keputusan

Menurut Bonczek, dkk (1980) mendefinisikan sistem pendukung keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, yaitu sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lain), sistem pengetahuan (repositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur), dan sistem pemrosesan masalah (hubungan antara dua komponen lainnya, terdiri dari satu atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk pengambilan keputusan). Keputusan yang baik dapat diambil melalui beberapa tahapan fase. Fase yang harus dilalui untuk mengambil suatu keputusan antara lain (Nofriansyah, 2014):

1. Intelligence

Tahap ini merupakan proses penelusuran dan pendeteksian dari ruang lingkup problematika secara proses pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasi masalah.

2. Design

Tahap ini merupakan proses menemukan, mengembangkan, dan menganalisis alternative tindakan yang bisa dilakukan. Tahap ini meliputi menguji kelayakan solusi.

3. Choice

Tahap ini merupakan tahap proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Hasil pemilihan tersebut kemudian diimplementasikan dalam proses pengambilan keputusan. Secara garis besar, sistem pendukung keputusan dibangun oleh tiga komponen utama (Nofriansyah, 2014), yaitu:

a. Subsistem Data (database)

Subsistem data merupakan komponen sistem pendukung keputusan yang berguna sebagai penyedia data bagi sistem. Data tersebut disimpan untuk diorganisasikan dalam sebuah basis data yang diorganisasikan oleh suatu sistem yang disebut dengan sistem manajemen basis data (database management system).

b. Subsistem model (modelbase)

Model adalah suatu tiruan dari alam nyata. Kendala yang sering dihadapi dalam merancang model adalah ketika model yang dirancang tidak mampu mencerminkan seluruh variabel alam nyata, sehingga keputusan yang diambil tidak sesuai dengan kebutuhan. Oleh karena itu, dalam menyimpan model harus memperhatikan dan menjaga fleksibilitasnya.

c. Subsistem dialog (user system interface)

Subsistem dialog adalah fasilitas yang mampu mengintegrasikan sistem yang terpasang dengan pengguna secara interaktif. Sistem diimplementasikan melalui subsistem dialog sehingga pengguna dapat berkomunikasi dengan sistem yang dibuat.

2.2 Lokasi

Menurut Heizer & Render (2015) lokasi adalah pendorong biaya dan pendapatan, maka lokasi seringkali memiliki kekuasaan untuk membuat strategi bisnis perusahaan. Lokasi yang strategis bertujuan untuk memaksimalkan keuntungan dari lokasi baru perusahaan. Menurut Kotler (2008) Salah satu kunci menuju sukses adalah lokasi, lokasi dimulai dengan memilih komunitas. Keputusan ini sangat bergantung pada potensi pertumbuhan ekonomis dan stabilitas, persaingan, iklim politik, dan sebagainya.

Sebelum suatu perusahaan mendirikan pabrik, biasanya direncanakan sebaik mungkin sebab letak berpengaruh pada biaya operasi/produksi, harga jual, serta kemampuan perusahaan untuk bersaing di pasar (Subagyo, 2000).

2.3 Penentuan Lokasi

Menurut Munawaroh (2013) salah satu strategi yang perlu diperhatikan oleh perusahaan adalah pemilihan lokasi, baik lokasi pabrik untuk perusahaan

manufaktur ataupun lokasi usaha untuk perusahaan jasa/retail maupun lokasi perkantornya. Pemilihan lokasi, diperlukan pada saat perusahaan mendirikan usaha baru, melakukan ekspansi usaha yang telah ada maupun memindahkan lokasi perusahaan ke lokasi lainnya.

Pemilihan lokasi sangat penting karena berkaitan dengan besar kecilnya biaya operasi, harga maupun kemampuan bersaing. Tujuan dari strategi lokasi adalah untuk memaksimalkan benefit perusahaan:

- a. Bagi industri, untuk meminimumkan biaya. Lokasi yang tepat mendekatkan lokasi gudang penyimpanan bahan dengan lokasi produksi bisa menghemat biaya transportasi.
- b. Bagi retail dan profesional service untuk maksimisasi revenue. Pemilihan lokasi retail dan profesional service yang mudah dijangkau konsumen memungkinkan terjadi penjualan dalam jumlah banyak, sehingga meningkatkan pendapatan perusahaan.
- c. Bagi lokasi gudang untuk memaksimalkan speed delivery dan biaya minimum. Jarak gudang dengan lokasi pabrik yang tepat akan mempercepat penyerahan barang sekaligus meminimalkan biaya.

Globalisasi mempengaruhi keputusan lokasi dalam hal:

- a. Teknologi transportasi dan komunikasi lebih baik.
- b. Sistem finansial terbuka.
- c. Kebutuhan barang impor meningkat.
- d. Kuota impor dan hambatan menurun.

Perusahaan dengan adanya globalisasi perlu mengelola operasi dalam keputusan lokasinya, yaitu dengan:

- a. Memperhatikan kesempatan pasar dan pesaing.
- b. Memenuhi standar kualitas dan waktu.
- c. Mampu mengkomunikasikan secara efektif, berkenaan dengan perbedaan bahasa di lokasi yang berbeda.

- d. Perbedaan norma dan kebiasaan pelanggan.
- e. Manajemen tenaga kerja yang lebih baik.
- f. Hukum dan peraturan yang berlaku perlu diperhatikan.
- g. Perbedaan biaya, meliputi biaya tenaga kerja, bahan baku, persediaan dan realstate Menurut Heizer & Render (2015), lokasi yang spesifik seringkali mempengaruhi pendapatan daripada terhadap biaya.

Terdapat 8 faktor yang menentukan volume dan pendapatan bagi perusahaan jasa:

- a. Daya beli konsumen pada area yang dituju.
- b. Jasa dan gambaran sesuai dengan demografis konsumen pada area yang dituju.
- c. Persaingan di dalam area.
- d. Kualitas persaingan.
- e. Keunikan dari lokasi perusahaan dan para pesaingnya.
- f. Kualitas fisik dari tempat fasilitas dan bisnis di sekitarnya.
- g. Kebijakan operasional perusahaan.
- h. Kualitas dari manajemen

Pemilihan lokasi usaha merupakan salah satu keputusan bisnis yang harus dibuat secara hati-hati. Penelitian-penelitian terdahulu menemukan bahwa lokasi usaha berhubungan dengan penjualan bisnis tersebut (Indarti, 2004). Namun, penelitian-penelitian tersebut masih didominasi oleh pemilihan lokasi di sektor manufaktur, industri teknologi tinggi, dan perbisnisan besar, dimana pemilihan lokasi usaha tersebut didorong oleh pertimbangan besarnya biaya transportasi bahan produksi.

Penentuan lokasi perlu dilakukan dengan matang yang terdiri dari lokasi untuk kantor pusat, cabang, dan pabrik. Dalam kaitannya dengan studi kelayakan bisnis, hal yang paling kompleks dan rumit yaitu penentuan lokasi pabrik. Pertimbangannya yaitu apakah dekat dengan bahan baku atau pasar atau konsumen, biaya dan luas produksi (Kasmir, 2014).

2.4 Metode Naive Bayes

Menurut pendapat kusrini dan luthfi Bayesian classification adalah pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas

keanggotaan suatu kelas. Bayesian classification didasarkan pada teorema Bayes yang memiliki kemampuan klasifikasi serupa dengan decision tree dan neural network. Bayesian classification terbukti memiliki akurasi dan kecepatan yang tinggi saat diaplikasikan ke dalam database dengan data yang besar (Kusrini dan Luthfi, 2009).

Naïve Bayes Classifier merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma menggunakan teorema Bayes dan mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variabel kelas (Patil dan Sherekar, 2013). Definisi lain mengatakan Naive Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman dimasa sebelumnya (Bustami 2014).

Algoritma Naive Bayes menggunakan dasar teorema Bayes dalam melakukan klasifikasi (Natalius, 2011).

$$P(H|X) = \frac{P(X|H)}{P(X)} \cdot P(H)$$

Keterangan: X = Data dengan class yang belum diketahui H = Hipotesis data merupakan suatu kelas spesifik P(H|X) = Probabilitas hipotesis H berdasarkan kondisi X (posteriori probabilitas) P(H) = Probabilitas hipotesis H (prior probabilitas) P(X|H) = Probabilitas X berdasarkan kondisi pada hipotesis H P(X) = Probabilitas X Persamaan teorema Bayes diatas dapat ditulis secara sederhana menjadi seperti Rumus dibawah ini (Natalius, 2011):

$$Posterior = \frac{prior \times likelihood}{evidence}$$

Algoritma Naive Bayes Classifier adalah metode yang didasari oleh probabilitas dan teorema Bayesian. Algoritma ini memiliki asumsi bahwa setiap variabel x bersifat bebas. Posterior adalah probabilitas masuknya sampel karakteristik tertentu atribut dalam kelas. Prior adalah probabilitas munculnya kelas. Likelihood adalah probabilitas kemunculan karakteristik-karakteristik atribut sampel terhadap kelas. Evidence adalah probabilitas kemunculan

karakteristik atribut sampel secara global (Bustami, 2014). Naive Bayes Classifier mengasumsikan bahwa keberadaan sebuah atribut tidak ada kaitannya dengan keberadaan atribut yang lain, maka dari itu perhitungan bisa dilakukan seperti pada Rumus dibawah ini (Natalius, 2011).

$$P(X|Ci) = \prod_{k=1}^n P(Xk|Ci)$$

Keterangan:

C_i = Kelas-kelas yang ada pada data = 1.

X = Data atribut yang kelasnya belum teridentifikasi.

$P(X|C_i)$ = Probabilitas atribut terhadap kondisi kelas.

$C_i P(X_k|C_i)$ = Probabilitas atribut ke sekian terhadap kondisi kelas C_i .

Menurut Raschka (2014) perhitungan prior dapat didapatkan seperti rumus dibawah ini.

$$P(C_i) = \frac{N_{ci}}{N_c}$$

Keterangan:

N_{ci} = Jumlah seluruh kelas ke-i.

N_c = Jumlah seluruh data (banyaknya data).

Kelas (label) dari data sampel X adalah kelas (label) yang memiliki nilai maksimum $P(X|C_i) \cdot P(C_i)$, karena $P(X|C_i)$ dapat diketahui melalui persamaan di atas (Abidin 2015). Secara matematis klasifikasi Naive Bayes Classifier bisa dirumuskan sebagai berikut (Natalius, 2011):

$$CNB = \operatorname{argmax} \prod_{k=1}^n P(X_k|C_i) P(C_i)$$

Keterangan:

$P(X_k|C_i)$ = probabilitas X_k terhadap kelas C_i (posterior).

$P(C_i)$ = probabilitas dari kelas C_i (prior).

Formula klasifikasi diatas tidak memuat nilai evidence, hal ini disebabkan karena evidence memiliki nilai yang positif dan tetap sama untuk semua kelas sehingga tidak mempengaruhi perbandingan nilai posterior (Natalius, 2011).

2.5 Metode Penelitian

Dalam menyusun laporan ini untuk memperoleh data dan informasi yang diperlukan penulis melakukan 2 metode, yaitu:

2.5.1 Metode Pengumpulan Data

Dalam penelitian ini metode penelitian bersifat kualitatif, Penelitian kualitatif disebut juga penelitian natural karena data pada penelitian ini bersifat alami atau natural.

4. Observasi : Melakukan pengamatan dan pendataan di PT Telkom Akses yang berhubungan dengan pembuatan sistem dan aplikasi rekomendasi kelayakan usaha depot air minum isi ulang.
5. Wawancara : Melakukan tanya jawab kepada pegawai PT Telkom Akses seputar penentuan lokasi pembangunan jaringan baru.
6. Studi Pustaka : Mempelajari buku-buku dan jurnal penelitian yang telah dilakukan berkaitan dengan permasalahan yang akan dituangkan dalam penyusunan laporan ini.

2.5.2 Metode Pengembangan sistem

Metode yang digunakan yaitu *System Development Life Cycle (SDLC)* adalah proses memahami bagaimana sistem informasi dapat mendukung kebutuhan, perancangan sistem, membangunnya dan memberikannya kepada pengguna (dennis, 2005). Fase-fase Sistem Development Life Cycle (SDLC) meliputi:

1. Perencanaan Sistem

Pada tahap perencanaan sistem, aktivitas-aktivitas yang dilakukan meliputi identifikasi apakah masalah-masalah yang ada bisa diselesaikan melalui pengembangan sistem dan penentuan teknologi dan pemilihan aplikasi.

2. Analisis Sistem

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau *Observasi* Akan memperoleh informasi sebanyak-banyaknya dari Petugas

LABKESDA sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh penggunanya tersebut.

3. Design Sistem

Proses design akan menterjemahkan syarat kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural.

4. Implementasi Sistem

Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan computer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap system tersebut dan kemudian bisa diperbaiki.

5. Pengujian Program

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan perencanaan, analisa, design dan pengkodean maka sistem yang sudah selesai dapat digunakan oleh penggunanya.

2.6 Pemodelan Sistem dengan UML (Unified Modeling Language)

UML singkatan dari *Unified Modelling Language* yang berarti bahasa pemodelan standar. UML memiliki sintaks dan semantik. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti standar yang ada. UML bukan hanya sekedar diagram, tetapi juga menceritakan konteksnya.

UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :



1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan system yang ada, proses-proses dan organisasinya (Prabowo Pudjo Widodo, 2011)

2.6.1 Use Case Diagram

1. Tabel Komponen *Use case*

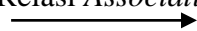


Diagram *use case* menyajikan interaksi antara *use case* dan aktor. Dimana aktor dapat berupa orang, peralatan atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan – persyaratan yang harus dipenuhi sistem dari pandangan pemakai. Beberapa komponen dalam *use case* antara lain :

Tabel 2.1 Komponen *Use Case Diagram* (Hamilton dan Russel, 2006)

Komponen Use case	Penjelasan
 Aktor	Merupakan sebuah komponen yang menggambarkan seseorang atau sesuatu (seperti perangkat atau sistem lainnya) yang berinteraksi dengan sistem.
 <i>Use case</i>	Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.

Ada beberapa relasi yang terdapat dalam use case, antaran lain :

Tabel 2.2 Relasi Tabel *Use Case* (Hamilton dan Russel, 2006)

Relasi <i>Use case</i>	Penjelasan
Relasi <i>Association</i> 	<i>Association</i> , menghubungkan link antar element
Relasi <i>Generalization</i> 	<i>Generalization</i> disebut juga <i>inheritance</i> (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
Relasi <i>Dependency</i> 	<i>Dependency</i> , sebuah elemen bergantung dalam beberapa

	cara ke elemen lainnya.
--	-------------------------

Tipe relasi / *stereotype* yang mungkin terjadi pada *use case diagram* :

Tabel 2.3 Tabel *Stereotype* yang mungkin terjadi (Hamilton dan Russel, 2006)


Relasi/ <i>Stereotype</i>	Penjelasan
<<include>>	Kelakuan yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, dimana pada kondisi ini sebuah <i>use case</i> adalah bagian dari <i>use case</i> lainnya.
<<extends>>	Kelakuan yang hanya berjalan dibawah kondisi tertentu seperti menggerakkan alarm.
<<Communicates>>	Ditambahkan untuk asosiasi yang mungkin menunjukan asosiasinya adalah <i>communicates association</i> .








2.6.2 Activity Diagram

1. Tabel Activity Diagram

Diagram aktivitas menggambarkan aliran fungsionalitas sistem. Dapat juga digunakan untuk menggambarkan aliran kejadian dalam *use case*. Berikut beberapa komponen yang terdapat dalam *Activity Diagram* :

Tabel 2.4 *Activity Diagram* (Hamilton dan Russel, 2006)




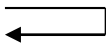

<i>Activity Diagram</i>	Penjelasan
 <i>Start State</i>	<i>Start State</i> , Sebagai tanda awal proses dari <i>activity diagram</i> .

 <i>State</i>	<i>State</i> , Berfungsi menampung event dalam <i>activity diagram</i> .
 <i>Activity</i>	<i>Activity</i> , Memiliki fungsi yang sama dengan <i>state</i> . Menampung event atau aktifitas pada proses sistem.
 <i>State Transition</i>	<i>State Transition</i> , Berfungsi untuk menunjukkan aliran atau urutan dari event atau aktifitas pada diagram.
 <i>Transition to self</i>	<i>Transition to Self</i> , Berfungsi untuk menunjukkan transisi sebuah event yang mengarah ke event itu sendiri.
 <i>Horizontal Synchronization</i>	<i>Horizontal Synchronization</i> , Berfungsi untuk mensinkronisasikan 2 cabang event yang posisinya horizontal.
 <i>Decision</i>	<i>Decision</i> , Digunakan ketika terjadi pemilihan 2 kondisi event pada diagram.
 End State	<i>End State</i> , Sebagai tanda akhir dari <i>activity diagram</i> .

2.6.3 Squence Diagram

Diagram sekuensial umumnya digunakan untuk menggambarkan suatu skenario atau urutan langkah – langkah yang dilakukan baik oleh *actor* maupun sistem yang merupakan respon dari sebuah kejadian untuk mendapatkan hasil atau output. Berikut adalah beberapa komponen yang terdapat dalam *sequence diagram* :

Tabel 2.5 *Sequence Diagram* (Hamilton dan Russel, 2006)

<i>Sequence Diagram</i>	Penjelasan
 <i>Actor</i>	<i>Actor</i> , Menggambarkan seseorang atau sesuatu (seperti perangkat atau sistem lain) yang berinteraksi dengan sistem.
 <i>Return Message</i>	<i>Return Message</i> , menggambarkan pesan atau hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.
 <i>Lifeline</i>	<i>Lifeline</i> , Eksekusi obyek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya)
 <i>Message to Self</i>	<i>Message to Self</i> , Menggambarkan pesan atau hubungan obyek itu sendiri yang menunjukkan urutan kejadian yang terjadi.
 <i>Object Message</i>	<i>Object Message</i> , Menggambarkan pesan atau hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.

2.7 Pengertian Data Mining

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar. (Turban, dkk. 2005)

Definisi umum dari data mining itu sendiri adalah proses pencarian pola-pola yang tersembunyi (hidden patern) berupa pengetahuan (knowledge) yang tidak diketahui sebelumnya dari suatu sekumpulan data yang mana data tersebut dapat berada di dalam database, data werehouse, atau media penyimpanan informasi yang lain. Hal penting yang terkait di dalam data mining adalah:

1. Data mining merupakan suatu proses otomatis terhadap data yang sudah ada.
2. Data yang akan diproses berupa data yang sangat besar.
3. Tujuan data mining adalah mendapatkan hubungan atau pola yang mungkin memberikan indikasi yang bermanfaat. (Kusrini dan Emha Taufiq, 2009)

Data mining dilakukan dengan tool khusus, yang mengeksekusi operasi data mining yang telah didefinisikan berdasarkan model analisis. Data mining merupakan proses analisis terhadap data dengan penekanan menemukan informasi yang tersembunyi pada sejumlah data besar yang disimpan ketika menjalankan bisnis perusahaan. Kemajuan luar biasa yang terus berlanjut dalam bidang data mining didorong oleh beberapa faktor antara lain:

1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam data warehouse, sehingga seluruh perusahaan memiliki akses ke dalam database yang andal.
3. Adanya peningkatan akses data melalui navigasi web dan internet.
4. Tekanan kompetisi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.
5. Perkembangan teknologi perangkat lunak untuk data mining (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan. (Larose, 2005)

Istilah data mining dan knowledge discovery in databases (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lainnya. Salah satu tahapan dalam keseluruhan proses KDD adalah data mining. Proses KDD itu ada 5 tahapan yang dilakukan secara terurut, yaitu:

1. Data selection Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. Pre-processing / cleaning Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus KDD. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data.

3. Transformation Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses coding dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data mining Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. Interpretation / evaluation Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap 9 ini merupakan bagian dari proses KDD yang disebut interpretation. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya. (Fayyad, 1996)

2.8 Definisi Aplikasi

Menurut Jogiyanto (2004, h. 20) dalam bukunya yang berjudul “Analisis dan Desain Sistem Informasi” aplikasi merupakan program yang berisikan perintah-perintah untuk melakukan pengolahan data. Jogiyanto menambahkan aplikasi secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal.

Dari pengertian diatas, dapat disimpulkan bahwa aplikasi merupakan *software* yang ditransformasikan ke komputer yang berisikan perintah-perintah yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data.

2.9 Internet dan Website

2.9.1 Internet

Internet merupakan sekumpulan jaringan komputer yang menghubungkan situs akademik, pemerintahan, komersial, organisasi, maupun perorangan. Internet menyediakan akses untuk layanan telekomunikasi dan sumber daya informasi untuk jutaan pemakainya yang tersebar di seluruh dunia. Layanan internet meliputi komunikasi langsung (email, chat), diskusi (*Usenet news, email, milis*), sumber daya informasi yang terdistribusi (*World Wide web, Gopher*), *remote login*, lalu lintas file (*Telnet, FTP*), dan aneka layanan lainnya.

2.9.2 Website

Website merupakan salah satu layanan yang didapat oleh komputer yang terhubung ke internet dengan fasilitas hypertext untuk menampilkan data berupa teks, gambar, suara, animasi dan multimedia lainnya. (Yeni Kustiyahningsih, 2011).

2.9.3 HTML

HTML adalah singkatan dari Hyper-Text Markup Language. HTML merupakan tulang punggung aplikasi web. Sekali pun sejumlah teknologi pendukung aplikasi web telah muncul, tetapi peran HTML tak tergantikan. Kehadiran PHP merupakan pelengkap yang memungkinkan aplikasi web tidak lagi bersifat statis, melainkan dinamis. Perubahan-perubahan pada sisi data tidak

perlu membuat aplikasi diubah. Cara seperti itu terutama dilakukan pada sisi server. Artinya, kode seperti PHP berjalan di server. Selain cara seperti itu, penambahan kode pada HTML juga bisa dilakukan pada sisi klien. (Abdul Kadir, 2011).

2.9.4 CSS

Cascading Style Sheet (CSS) merupakan salah satu bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam. Sama halnya styles dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML.

2.9.5 BOOTSRAP

Bootstrap adalah *front – end framework* yang di dalamnya terdapat CSS dan *JavaScript* untuk mempermudah pengembang dalam memulai pengembangan sebuah *web*. Pengembang yang beralih ke pengembangan *front – end* dari bahasa pemrograman untuk sisi server seperti *Java* atau PHP akan merasakan kesulitan saat harus berurusan dengan CSS dan *JavaScript*; namun dengan *Bootstrap* para pengembang dapat berkonsentrasi hanya pada penulisan HTML yang baik, dan menyerahkan CSS dan *JavaScript* kepada *Bootstrap*.

2.9.6 jQuery

JQuery adalah *library JavaScript* yang dibuat untuk memudahkan pembuatan website dengan HTML yang berjalan di sisi *client*. *JQuery* diluncurkan pada tanggal 26 Januari 2006 di Barcamp New York City oleh John Resig dan berlisensi ganda dibawah MIT dan GPL. Menurut Aloysius Sigit W (2011:1) *jQuery* adalah *library* atau kumpulan kode *JavaScript* yang sudah siap pakai menjadikan *jQuery* sebagai salah satu *JavaScript framework* terbaik saat ini. Sekarang *jQuery* dikembangkan oleh team *developer* yang dipimpin oleh Dave Metvin. Dipakai oleh lebih dari 55% dari 10.000 website yang paling sering dikunjungi. *Jquery* menjadi *library JavaScript* yang paling populer sekarang.

2.10 Tool Untuk Membangun Aplikasi Web

Web Server : Apache

Server HTTP Apache atau Server Web/WWW Apache adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP.

Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah

2.11 Script Pemrograman

2.11.1 Java Script

Java Script adalah kode untuk menyusun halaman web yang memungkinkan dijalankan di sisi klien (pada browser yang digunakan pemakai). Karena dijalankan di sisi klien, maka JavaScript digunakan untuk membuat tampilan lebih dinamis. (Abdul Kadir, 2011).

2.11.2 PHP

PHP adalah teknologi yang diperkenalkan tahun 1994 oleh Rasmus Lerdorf. Beberapa versi awal yang tidak dipublikasikan digunakan pada situs pribadinya untuk mencatat siapa saja yang mengakses daftar riwayat hidup onlinennya. Versi pertama digunakan oleh pihak lain pada awal tahun 1995 dan dikenal sebagai Personal Home Page Tools. Terkandung didalamnya sebuah parser engine (mesin pengurai) yang sangat disederhanakan, yang hanya mampu mengolah macro khusus dan beberapa utilitas yang sering digunakan pada pembuatan home page, seperti buku tamu, pencacah, dan hal semacamnya. Parser tersebut ditulis ulang pada pertengahan 1995 dan dinamakan PHP/FI Versiaon 2. FI(Form Interpreter) sendiri berasal dari kode lain yang ditulis juga oleh Rasmus, yang menterjemahkan HTML dari data. Ia menggabungkan script Personal Home Page Tools dengan Form Interpreter dan menambahkan dukungan terhadap server database yang menggunakan format mSQL sehingga lahirlah PHP/FI.

PHP/FI tumbuh dengan pesat, dan orang-orang mulai menyiapkan kode-kode programnya supaya bisa didukung oleh PHP. (Abdul Kadir, 2011).

2.12 Database Server : MySQL

MYSQL adalah database. Database sendiri merupakan suatu jalan untuk dapat menyimpan berbagai informasi dengan membaginya berdasarkan kategori-kategori tertentu. Dimana informasi-informasi tersebut saling berkaitan satu dengan yang lainnya.

