

BAB II

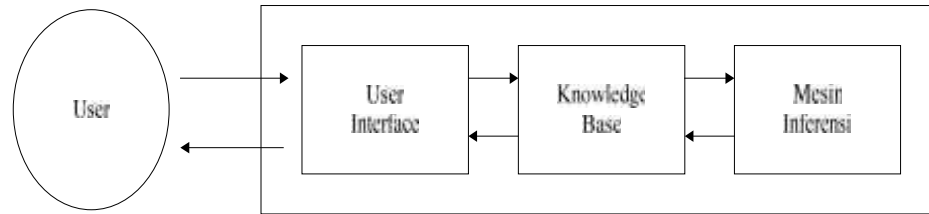
LANDASAN TEORI

2.1. Sistem Pakar

Sistem pakar adalah program komputer yang menggunakan pengetahuan pakar untuk mencapai tingkat kinerja yang tinggi pada area yang sempit . Secara umum, sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Pakar yang dimaksud disini adalah orang yang mempunyai keahlian khusus yang dapat menyelesaikan masalah yang tidak dapat diselesaikan orang awam. Sebagai contoh, dokter adalah seorang pakar yang mampu mendiagnosis penyakit yang diderita pasien serta dapat memberikan penatalaksanaan suatu penyakit. Contoh yang lain, montir adalah seorang yang mempunyai keahlian dan pengalaman dalam menyelesaikan kerusakan mesin motor/mobil; psikolog adalah orang yang ahli dalam memahami kepribadian seseorang, dan lain-lain. (Siswanto. 2010)

2.2. Struktur Sistem Pakar

Sistem pakar terdiri dari 3 komponen utama, yaitu : basis pengetahuan, motor *inferensi* dan *user inferensi*. Untuk lebih jelasnya, diagram blok umum *expert system* dapat dilihat pada gambar 2.1 dibawah ini :



Gambar 1.1 Diagram Blok Umum *Expert System*

Sistem pakar biasanya mengajukan pertanyaan-pertanyaan tertentu sampai dapat mengidentifikasi suatu objek yang sesuai dengan informasi yang diketahuinya. Ini merupakan bagian *software* spesialisasi tingkat tinggi yang berusaha menduplikasi fungsi suatu pakar dalam suatu bidang keahlian. Program ini bertindak sebagai penasehat atau konsultan dalam suatu lingkup keahlian tertentu, sebagai hasil pengetahuan yang telah dikumpulkan dari beberapa orang pakar.

Sistem pakar yang harus memiliki ciri-ciri sebagai berikut :

- 1) Memiliki fasilitas informasi yang handal
- 2) Mudah dimodifikasi
- 3) Dapat digunakan dalam berbagai jenis komputer
- 4) Memiliki komputer untuk belajar beradaptasi.

2.3. Faktor Kepastian (*Certainty factor*)

Dalam menghadapi suatu masalah, sering ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini bias berupa probabilitas atau kebolehjadian yang bergantung pada hasil suatu kejadian. Hasil yang tidak pasti disebabkan oleh dua faktor, yaitu aturan yang tidak pasti dan jawaban yang tidak pasti. Hal ini sangat mudah dilihat pada sistem diagnosis penyakit, dimana pakar tidak dapat

mendefinisikan hubungan antara gejala dengan penyebabnya secara pasti, dan pasien tidak dapat merasakan suatu gejala dengan pasti pula. Pada akhirnya ditemukan banyak kemungkinan penyakit.

Factor kepastian diperkenalkan oleh Shortliffe Buchanan dalam pembuatan MYCIN. *Certainty factor* merupakan nilai parameter klinis yang diberikan MYCIN untuk menunjukkan besarnya kepercayaan. *Certainty factor* didefinisikan sebagai berikut:

$$CF(H, E) = MB(H, E) - MD(H, E)$$

Dimana:

$CF(H, E)$: *Certainty factor* dari hipotesis H yang dipengaruhi oleh gejala E. besarnya CF berkisar antara -1 sampai dengan 1.

Nilai -1 menunjukkan ketidakpercayaan mutlak, sedangkan nilai 1 menunjukkan kepercayaan mutlak.

$MB(H, E)$: Ukuran kenaikan kepercayaan terhadap hipotesis H yang dipengaruhi oleh gejala E.

$MD(H, E)$: Ukuran kenaikan ketidakpercayaan terhadap hipotesis H yang dipengaruhi oleh gejala E.

Contoh perhitungan Certanty Factor

Seseorang berkonsultasi pada sistem pakar untuk mengetahui apakah terkena penyakit jantung koroner atau tidak. Sistem pakar mempunyai basis pengetahuan sebagai berikut :

Rule 1 :

IF Sesak nafas

AND Nyeri dada

AND Denyut jantung cepat

AND Keringat berlebihan

AND Kelelahan

AND Mual

AND Pusing

THEN Jantung Koroner

Langkah Pertama :

Pengguna konsultasi diberi pilihan jawaban yang masing-masing

bobotnya sebagai berikut :

No Keterangan Nilai User

1. Tidak 0
2. Tidak tahu 0,2
3. Sedikit yakin 0,4
4. Cukup yakin 0,6
5. Yakin 0,8
6. Sangat yakin 1

Pakar menentukan nilai CF (Certainty Factor) untuk masing-masing

gejala sebagai berikut :

CFpakar (Sesak nafas) = 1,0

CFpakar (Nyeri dada) = 1,0

CFpakar (Denyut jantung cepat) = 1,0

CFpakar (Keringat berlebih) = 0,8

CFpakar (Kelelahan) = 0,6

CFpakar (Mual) = 0,4

CFpakar (Pusing) = -0,4

Kemudian dilanjutkan dengan penentuan nilai bobot user, setelah dilakukan dialog antar sistem pakar dan user memilih jawabannya :

Misalkan :

1. Sistem pakar : Apakah Anda mengalami Sesak nafas ?

User : Cukup yakin (CFuser = 0,6)

2. Sistem pakar : Apakah Anda mengalami nyeri dada ?

User : Cukup yakin (CFuser = 0,6)

3. Sistem pakar : Apakah Anda mengalami denyut jantung yang cepat ?

User : Yakin (CFuser = 0,8)

4. Sistem pakar : Apakah Anda mengalami kringat berlebih ?

User : Sedikit yakin (CFuser = 0,4)

5. Sistem pakar : Apakah Anda mengalami kelelahan ?

User : Tidak Tahu (CFuser = 0,2)

6. Sistem pakar : Apakah Anda mengalami mual ?

User : Tidak (CFuser = 0)

7. Sistem pakar : Apakah Anda mengalami pusig ?

User : Cukup yakin (CFuser = 0,6)

Langkah Kedua :

Kaidah-kaidah atau rule tersebut kemudian dihitung nilai CF-nya dengan mengalikan CFpakar

dengan Cfuser menjadi :

$$CF[H,E]1 = CF[H]1 * CF[E]1$$

$$= 1,0 * 0,6$$

$$= 0,6$$

$$CF[H,E]2 = CF[H]2 * CF[E]2$$

$$= 1,0 * 0,6$$

$$= 0,6$$

$$CF[H,E]3 = CF[H]3 * CF[E]3$$

$$= 1,0 * 0,8$$

$$= 0,8$$

$$CF[H,E]4 = CF[H]4 * CF[E]4$$

$$= 0,8 * 0,4$$

$$= 0,32$$

$$CF[H,E]5 = CF[H]5 * CF[E]5$$

$$= 0,6 * 0,2$$

$$= 0,12$$

$$CF[H,E]6 = CF[H]6 * CF[E]6$$

$$= 0,4 * 0$$

$$= 0$$

$$CF[H,E]7 = CF[H]7 * CF[E]7$$

$$= (-0,4) * 0,6$$

$$= -0,24$$

Langkah ketiga :

Mengkombinasikan nilai CF dari masing-masing kaidah (rule)

$$CF_{combine} CF[H,E]1,2 = CF[H,E]1 + CF[H,E]2 * (1 - CF[H,E]1)$$

$$= 0,6 + 0,6 * (1 - 0,6)$$

$$old1 \diamond = 0,84$$

$$CF_{combine} CF[H,E]old1,3 = CF[H,E]old1 + CF[H,E]3 * (1 - CF[H,E]old1)$$

$$= 0,84 + 0,8 * (1 - 0,84)$$

$$old2 \diamond = 0,968$$

$$CF_{combine} CF[H,E]_{old2,4} = CF[H,E]_{old2} + CF[H,E]_4 * (1 - CF[H,E]_{old2})$$

$$0,968 + 0,32 * (1 - 0,968)$$

$$old3 \diamond = 0,978$$

$$CF_{combine} CF[H,E]_{old3,5} = CF[H,E]_{old3} + CF[H,E]_5 * (1 - CF[H,E]_{old3})$$

$$= 0,978 + 0,12 * (1 - 0,978)$$

$$old4 \diamond = 0,981$$

$$CF_{combine} CF[H,E]_{old4,6} = CF[H,E]_{old4} + CF[H,E]_6 * (1 - CF[H,E]_{old4})$$

$$= 0,981 + 0 * (1 - 0,981)$$

$$old5 \diamond = 0,981$$

$$CF_{combine} CF[H,E]_{old5,7} = CF[H,E]_{old5} + CF[H,E]_7$$

$$1 - \min [| CF[H,E]_{old5} |, | CF[H,E]_7 |]$$

$$= 0,981 + (-0,24)$$

$$1 - \min [|0,9521|, |-0,24|]$$

$$= 0,981 + (-0,24)$$

$$1 - (-0,24)$$

$$= 0,5976 \text{ old5}$$

$$CF[H,E]_{old5} * 100\% = 0,5976 * 100\%$$

$$= 59,76 \%$$

Dengan demikian dapat dikatakan bahwa perhitungan certainty factor pada penyakit jantung koroner memiliki persentase tingkat keyakinan 59,76%

(Informatikalen 2015)

2.4. Ketidakpastian

Ada tiga penyebab ketidakpastian aturan, yaitu aturan tunggal, penyelesaian konflik, dan ketidakcocokan antarkonsekuensi dalam

aturan. Aturan tunggal yang dapat menyebabkan ketidakpastian dipengaruhi oleh tiga hal, yaitu kesalahan, probabilitas, dan kombinasi gejala. Probabilitas disebabkan ketidakmampuan seorang pakar merumuskan suatu aturan secara pasti. Misalnya, jika seseorang mengalami sakit kepala, demam, dan bersin ada kemungkinan orang tersebut terserang penyakit flu, tetapi bukan berarti apabila seseorang mengalami gejala tersebut pasti terserang flu. Hanya karena aturan tunggalnya benar, belum tentu menjamin suatu jawaban bernilai benar. Hal ini masih dipengaruhi oleh kompatibilitas aturan. Inkompabilitas suatu aturan disebabkan oleh beberapa hal, yaitu:

a. Kontradiksi aturan, misalnya:

Aturan 3.1 : **jika** anak demam **maka** harus dikompres.

Aturan 3.2 : **jika** anak demam **maka** jangan dikompres.

b. Subsumsi aturan, misalnya:

Aturan 3.3 : jika e1 maka h

Aturan 3.4 : jika e1 dan e2 maka h

Jika hanya e1 yang muncul, maka masalah tidak akan timbul karena aturan yang akan muncul adalah aturan 3.3, tetapi apabila e1 dan e2 sama-sama muncul, maka kedua aturan (3.3 dan 3.4) sama-sama akan dijalankan.

c. Redundancy aturan, misalnya

Aturan 3.5 : jika e1 dan e2 maka h

Aturan 3.6 : **jika** e2 dan e1 **maka** h

Dalam kasus ini terdapat aturan yang berbeda, namun memiliki makna yang sama.

d. Kehilangan aturan, misalnya:

Aturan 3.7 : **jika** e4 **maka** h

Ketika e4 diabaikan maka h tidak pernah tersimpulkan.

e. Penggabungan data, misalnya pada diagnosis kesehatan. Seorang dokter dapat menyimpulkan suatu penyakit tidak hanya berdasarkan anamnesis tetapi juga hasil tes laboratorium, pemeriksaan kondisi tubuh, sejarah penyakit, dan lain-lain. Untuk itu diperlukan penggabungan semua data untuk dapat menyimpulkan suatu penyakit.

2.5. PHP (*Hypertext Preprocessor*)

PHP adalah suatu bahasa pemrograman *open source* yang digunakan secara luas terutama untuk pengembangan *web* dan dapat disimpan dalam bentuk HTML. Keuntungan utama menggunakan PHP adalah *script* PHP tidak hanya benar-benar sederhana bagi pemula, tetapi juga menyediakan banyak fitur tambahan untuk programmer profesional (Nugroho, 2012).

PHP merupakan *script* yang digunakan untuk membuat halaman website yang bersifat dinamis. Dinamis artinya bahwa halaman website yang ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru.

2.6. Adobe Dreamweaver

Adobe Dreamweaver merupakan program penyunting halaman web dari Adobe Sistem yang dulu dikenal sebagai Macromedia Dreamweaver dari Macromedia. Program ini banyak digunakan oleh pengembang web karena fitur fiturnya yang lengkap serta kemudahan

dalam penggunaannya. Versi terakhir Macromedia Dreamweaver sebelum Macromedia dibeli oleh Adobe Sistem yaitu versi 8. Kemudian setelah dibeli oleh Adobe Sistem berkembang ke Versi selanjutnya yakni versi 9 (CS3) versi 10 yang ada dalam Adobe Creative Suite 4 (CS4) dan versi terbarunya yaitu CS5. Dengan menggunakan program ini, seorang programmer web dapat dengan mudah membuat dan mendesain webnya karena bersifat WYSIWYG (*What You See Is What You Get*) yang artinya apa yang kita lihat pada halaman desain, maka semua itu akan kita peroleh pada browser. Dengan kelebihan ini seorang programmer atau web designer dapat langsung melihat hasil buaatannya tanpa harus membukanya pada browser. (Kristanto, 2010)

Sebagai editor web yang handal, Adobe Dreamweaver tentunya dilengkapi dengan kemampuan manajemen situs, yang memudahkan kita mengelola keseluruhan elemen yang ada dalam sebuah situs. Kita juga dapat melakukan evaluasi situs dengan melakukan pengecekan broken link, kompatibilitas browser, termasuk validasi tag-tag HTML dan CSS yang tidak sesuai dengan pedoman secara otomatis serta perkiraan waktu download pada sebuah halaman web.

2.7. Database

Database merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu. Ada beberapa hal yang harus di perhatikan dalam membangun suatu database, yaitu :

a. Entitas

Merupakan suatu object yang memiliki karakteristik tertentu sebagai suatu atribut, misalnya entitas mahasiswa memiliki atribut nama, alamat dan lain lain.

b. Field

Mempresesentasikan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain-lain.

c. Record

Kumpulan dari field membentuk suatu record. Record menggambarkan suatu unit data individu tertentu. Kumpulan dari record membentuk suatu file. Misalnya file personalia, tiap-tiap record dapat mewakili data tiap-tiap karyawan.

d. Attribute

Merupakan karakteristik dari sebuah entitas. Nama, alamat merupakan atribut dari entitas mahasiswa.

Database adalah sekumpulan tabel-tabel yang berisi data dan merupakan kumpulan dari field atau kolom. Struktur file yang menyusun sebuah database adalah Data Record dan Field. Data adalah suatu satuan informasi yang akan diolah. Sebelum diolah, data dikumpulkan didalam suatu file database. Record adalah data yang isinya merupakan satu kesatuan seperti Nama User dan password. Setiap keterangan yang mencakup Nama User dan Password dinamakan satu record. Setiap record diberi nomor urut yang disebut nomor record (Record number). Field adalah sub bagian dari record (Anhar, 2010).

2.8. Pemodelan Data

1) *Entity Relationship Diagram* (ERD)

Fungsi dari *Entity Relationship Diagram* adalah model data untuk menggambarkan hubungan antara satu entitas dengan entitas lain yang mempunyai relasi (hubungan) dengan batasan-batasan.

Batasan-batasan pada *relationship-type*, batasan struktur *cardinality ratio* memiliki jenis :

a. 1 : 1 (*One to one relationship*)

Satu entitas pada tipe entitas A berhubungan dengan satu entitas pada tipe entitas B dan sebaliknya.

b. 1 : N (*One to many Relationship*)

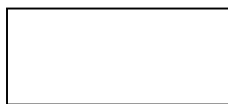
Suatu entitas di A dihubungkan dengan sejumlah entitas di B.

c. M : N (*Many to many Relationship*)

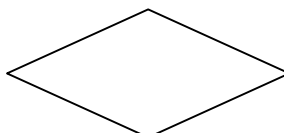
Sejumlah entitas di A dihubungkan dengan sejumlah entitas di B.

Beberapa simbol dalam penggambaran *Entity Relationship Diagram* (ERD) :

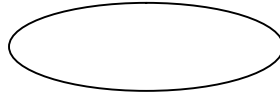
- a. Tipe Entitas, merupakan objek riil yang dapat dibedakan satu dengan yang lainnya dan tidak saling bergantung.



- b. *Type Relationship*, merupakan hubungan yang ada antara entitas satu dengan yang lainnya.



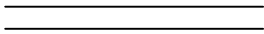
- c. Tipe Atribut, atribut menentukan suatu entitas dimana setiap atribut memiliki harga dan setiap entitas dapat memiliki lebih dari satu atribut.



- d. *Partial Participation*



- e. *Total Participation*



2) *Mapping Table*

Mapping Table merupakan gambaran hubungan antara tabel-tabel yang berperan membangun basis data. Tabel-tabel ini merupakan penjabaran dari entitas-entitas.