

BAB III

LANDASAN TEORI

3.1 Website

Website adalah layanan internet yang paling banyak memiliki tampilan grafis dan kemampuan *link* yang bagus. *Website* dapat menghubungkan dari sembarang tempat dalam sebuah dokumen atau gambar ke sembarang tempat di dokumen lain. *Website* mengizinkan pemberian penyorotan/penggaris-bawahan pada kata-kata atau gambar dalam sebuah dokumen untuk menghubungkan atau menunjuk ke media lain seperti dokumen, atau file suara. Keistimewaan inilah yang menjadikan web sebagai service yang paling cepat pertumbuhannya (Tulangow, 2011).

Pada awalnya aplikasi *Website* dibangun dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language*). HTML adalah standar yang digunakan pada halaman *Website*. HTML bekerja dengan menggunakan HTTP (*HyperText Transfer Protocol*), yaitu *protocol* komunikasi yang memungkinkan *Web Server* berkomunikasi dengan *Web Browser*.

3.2 Personal Home Page (PHP)

PHP adalah bahasa pemrograman web yang didesain untuk membuat *web-based application*. PHP banyak dipakai untuk memrogram situs web dinamis (Minarni & Saputra, 2011). PHP adalah bahasa *scripting* yang menyediakan cara yang mudah dalam melekatkan program pada halaman *website*. Karena suatu halaman diproses terlebih dahulu oleh PHP sebelum dikirim ke *client*, maka *script* dapat menghasilkan isi halaman yang dinamis, seperti misalnya menampilkan hasil *query* dari MySQL

pada halaman tersebut. PHP pada mulanya berarti *Personal Home Page*, tetapi sekarang telah menggunakan nama "PHP *Hypertext Preprocessor*" (Fitriastuti, 2009).

Sebagai bahasa pemrograman untuk tujuan umum, kode PHP diproses oleh aplikasi penerjemah dalam modus baris - baris perintah modul dan melakukan operasi yang diinginkan sesuai sistem operasi untuk menghasilkan keluaran program di-*channel output* standar. Hal ini juga dapat berfungsi sebagai aplikasi grafis. PHP tersedia sebagai prosesor untuk *server web* yang paling modern dan sebagai penerjemah mandiri pada sebagian besar sistem operasi dan komputer *platform*. Untuk tujuan ini, kode PHP tertanam ke dalam dokumen sumber HTML dan diinterpretasikan oleh *server web* dengan modul PHP prosesor, yang menghasilkan dokumen halaman *website* (Zuliarso & Februariyanti, 2013).

3.3 Sistem Pakar

Sistem pakar atau *expert system* merupakan sistem yang mengadopsi pengetahuan manusia di masukkan ke dalam komputer, agar komputer dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja seorang pakar (Handayani & Sutikno, 2008). Seperti halnya seorang pakar, sistem pakar terfokus pada suatu dominan masalah yang spesifik.

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) (Minarni & Hidayat, 2013). Tujuan utama sistem pakar bukan untuk menggantikan kedudukan seorang ahli maupun pakar, tetapi untuk memasyarakatkan pengetahuan dan pengalaman pakar-pakar yang ahli di bidangnya.

Ciri-Ciri sistem pakar (Satwika, 2012):

1. Terbatas pada bidang yang spesifik.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat mengemukakan rangkaian alasan yang diberikan dengan cara yang dapat dipahami.
4. Berdasarkan pada rule atau kaidah tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap.
6. *Output* tergantung dari dialog dengan *user*.
7. *Knowledge base* dan *inference engine* terpisah.

Berikut ini beberapa keuntungan dari sistem pakar (Dahria, 2011):

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian para pakar.
4. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
5. Mampu beroperasi dalam lingkungan yang berbahaya.
6. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
7. Tidak memerlukan biaya saat tidak digunakan, sedangkan pada pakar manusia memerlukan biaya sehari-hari.
8. Dapat digandakan (diperbanyak) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya.

9. Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama.
10. Menghemat waktu dalam pengambilan keputusan.
11. Meningkatkan kualitas dan produktivitas.

3.4 Mesin Inferensi

Mesin inferensi (*inference engine*) merupakan bagian yang bertindak sebagai pencari solusi dari suatu permasalahan berdasar pada kaidah-kaidah yang ada dalam basis pengetahuan sistem pakar. Selama proses inferensi, mesin inferensi memeriksa status dari basis pengetahuan dan memori kerja (*working memory*) untuk menentukan fakta apa saja yang diketahui dan untuk menambahkan fakta baru yang dihasilkan ke dalam memori kerja tersebut. Fakta-fakta yang merupakan hasil dari proses inferensi disimpan dalam memori kerja.

Ada dua strategi pencarian dasar yang biasa digunakan oleh mesin inferensi yaitu (Dahria, 2011):

1. *Forward Chaining* (Runut Maju)

Runut maju merupakan strategi pencarian yang memulai proses pencarian dari sekumpulan data atau fakta. Dari data-data tersebut akan dicari suatu kesimpulan yang menjadi solusi dari permasalahan yang dihadapi (*data driven*).



Gambar 3.1. Proses inferensi runut maju

Kelebihan dari metode runut maju diantaranya adalah (Satwika, 2012):

- a. Dapat menghasilkan informasi baru dari jumlah data yang relative sedikit.
- b. Merupakan pendekatan yang baik untuk masalah tertentu seperti perencanaan, pengawasan, pengaturan dan interpretasi.
- c. Dapat bekerja baik dengan permasalahan yang membutuhkan informasi lebih dulu baru kemudian menarik kesimpulan.

2. *Backward Chaining* (Runut Balik)

Runut balik (*backward chaining*) merupakan penalaran yang dimulai dari kesimpulan dan kemudian dibuktikan kebenarannya (*goal driven*). Runut balik merupakan strategi pencarian data yang arahnya kebalikan dari runut maju.

3.5 Suku Karo

Penduduk asli yang mendiami wilayah Kabupaten Karo disebut Suku Bangsa Karo. Suku Bangsa Karo ini mempunyai adat istiadat yang sampai saat ini terpelihara dengan baik dan sangat mengikat bagi Suku Bangsa Karo sendiri. Suku ini terdiri 5 (lima) Merga, Tuter Siwaluh, dan Rakut Sitelu. Merga silima yakni Karo-Karo, Ginting, Sembiring, Tarigan, dan Perangin-angin. Dari kelima Merga ini masih terdapat sub-sub Merga.

Berdasarkan Merga ini maka tersusunlah pola kekerabatan atau yang dikenal dengan Rakut Sitelu, Tuter Siwaluh dan Perkade-kaden Sepuluh Dua Tambah Sada. Rakut Sitelu yakni Senina/Sembuyak, Kalimbubu dan Anak Beru. Tuter Siwaluh yaitu Sipemerren, Siparibanen,

Sipengalon, Anak Beru, Anak Beru Menter, Anak Beru Singikuri, Kalimbubu dan Puang Kalimbubu. Perkade-Kaden Sepuluh Dua yakni Nini, Bulang, Kempu, Bapa, Nande, Anak, Bengkila, Bibi, Permen, Mama, Mami, dan Bere-Bere (Karo, 2007).

3.6 Ertutur

Bagi masyarakat Karo, merga sangat penting gunanya dalam kehidupan. Merga berguna untuk mengekspresikan identitas diri serta hubungannya dengan ertutur (mencari hubungan kekerabatan). Merga dan beru dipakai di belakang nama diri. Di samping itu, masyarakat Karo tidak hanya mempunyai merga atau beru saja, tetapi sekaligus mewarisi beru dari ibu kandungnya yang disebut bere-bere. Jadi setiap pribadi mempunyai merga atau beru dan bere-bere.

Namun demikian, dalam pelaksanaan sehari-hari bere-bere tidak pernah dicantumkan sebagai identitas diri. Bere-bere akan ditanya dalam kegiatan ertutur, untuk mengetahui hubungan kekeluargaan seseorang. Walaupun masyarakat Karo mempunyai sistem parental akan tetapi yang paling penting adalah merga dan beru. Hal ini terbukti bahwa merga dan beru tetap dicantumkan setelah seseorang meninggal dunia (Tarigan, 2003).

3.7 CodeIgniter

CodeIgniter adalah *framework* PHP yang berjalan pada php 4 dan php 5. *Framework* diartikan sebagai kumpulan *script* seperti *class* dan *function* yang dapat membantu *programmer* dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke database, pemanggilan variabel, dan file. *Framework* dapat juga disebut sebagai kerangka kerja. CodeIgniter ini

bertujuan untuk memudahkan *programmer* dalam mengembangkan aplikasi secara cepat tanpa harus memulai pemrograman dari awal atau nol (Rosmala et al., 2011). Ada beberapa keuntungan menggunakan *framework* (Koespradono et al., 2013):

1. Mempercepat dan memudahkan pembangunan sebuah aplikasi web.
2. Relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu di dalam *framework*.
3. Lebih bebas dalam pengembangan jika dibandingkan CMS.