BAB II

LANDASAN TEORI DAN

REPRESENTASI PENGETAHUAN

2.1. Sistem Pakar

Sistem pakar adalah sebuah perangkat lunak komputer yang memiliki basis pengetahuan untuk domain tertentu dan menggunakan penalaran inferensi menyerupai seorang pakar dalam memecahkan masalah [FARI94]. Sistem pakar adalah sebuah teknik inovatif baru dalam menangkap dan memadukan pengetahuan [FARI94]. Kekuatannya terletak pada kemampuannya memecahkan masalah-masalah praktis pada saat berhalangan. Kemampuan sistem pakar ini karena didalamnya terdapat basis pengetahuan yang berupa pengetahuan non formal yang sebagian besar berasal dari pengalaman. Seperti manusia, pengetahuannya bersifat teoritis dan praktis dan telah disempurnakan lewat pengalaman dalam domain tersebut. Sistem pakar merupakan program-program praktis yang menggunakan strategi heuristic yang dikembangkan oleh manusia untuk memecahkan masalah-masalah yang spesifik.

Sifat utama sistem pakar adalah ketergantungan sistem ini pada pengetahuan manusia yang pakar dalam suatu bidang dalam menyusun strategi pemecahan persoalan yang dihadapi oleh sistem [SAD91]. Pengetahuan pakar (*expert knowladge*) merupakan kombinasi pemahaman teoritis tentang suatu

persoalan dan sekumpulan aturan pemecahan persoalan heuristik dimana pengalaman telah menunjukan kefektifannya dalam persolan itu [SAND91].

Karena keheuristikannya dan sifatnya yang berdasarkan pengetahuan, maka umumnya sistem pakar bersifat :

a. Terbuka untuk diperiksa, baik dalam menampilkan langkah-langkah antara maupun dalam menjawab pertanyaan-pertanyaan tentang proses solusi.

b. Mudah dimodifikasi, baik dengan menambah maupun menghapus suatu kemampuan dari basis pengetahuannya.

c. Heuristik dalam menggunakan pengetahuan (yang sering kali tidak sempurna) untuk memperoleh solusi.

Beberapa alasan sebuah sistem pakar bersifat terbuka terhadap pemeriksaan, yaitu :

a. Jika seorang pakar seperti dokter atau insinyur menerima rekomendasi dari komputer, mereka harus puas terhadap jawaban yang diberikan.

b. Dapat mengevaluasi setiap aspek dan keputusan yang diambil selama proses mendapatkan solusi dan memungkinkan adanya penambahan sejumlah informasi atau aturan baru untuk mengembangkan kinerjanya.

Beberapa keuntungan penerapan sistem pakar adalah sebagai berikut [FARI94] :

a. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.

b. Meningkatkan produktivitas akibat meningkatnya kualitas hasil pekerjaan, meningkatnya kualitas pekerjaan ini disebabkan meningkatnya efisiensi kerja.

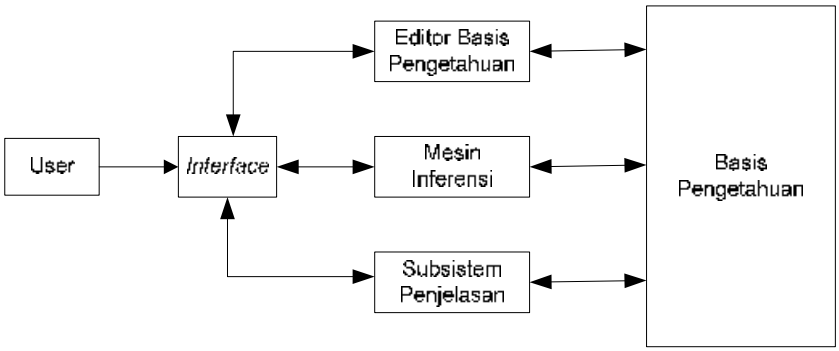
c. Menghemat waktu kerja dan menyerhanakan pekerjaan.

d. Merupakan arsip yang terpercaya dari sebuah keahlian, sehingga bagi pemakai sistem pakar akan seolah-olah berkonsultasi langsung dengan sang pakar, meskipun mungkin sang pakar telah meninggal.

e. Memperluas jangkauan, dari keahlain seorang pakar. Dimana sebuah sistem pakar yang telah disahkan akan sama saja artinya dengan seorang pakar yang tersedia dalam jumlah besar (dapat diperbanyak dengan kemampuan yang percis sama), dapat diperoleh dan dipakai dimana saja.

2.2 Arsitektur Sistem Pakar

Pada program sistem pakar, pemakai berinteraksi dengan sistem pakar lewat *interface* pemakai yang melakukan akses yang lebih nyaman bagi manusia pemakainya [SAND91].



Gambar 2.1 Arsitektur Sistem Pakar

Sistem pakar menggunakan berbagai jenis *interface*, misalnya pertanyaan dan jawaban, grafik dan sebagainya. Sedangkan subsitem penjelasan (*explanation*) dalam sistem pakar memungkinkan program untuk menjelaskan penalarannya kepada pemakai yang antara lain menyangkut kebutuhan sistem akan sepotong data tertentu. Selain itu kebanyakan sistem pakar juga melibatkan editor basis pengetahuan yang dapat mengakses subsistem penjelasan dan membantu programer untuk menentukan tempat hambatan dalam kinerja program. Sementara itu mesin inferensi (*inference engine*) dalam sistem pakar menerapkan pengetahuan untuk solusi dari masalah yang sebenarnya. Ini merupakan interpreter bagi basis pengetahuan.

2.3 Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan merupakan inti program sistem pakar dimana basis pengetahuan ini merupakan representasi pengetahuan (*Knowledge*

Respresentation) dari seorang pakar. Basis pengetahuan ini tersusun atas faktat yang berupa informasi tentang obyek, dan kaidah (*rule*) yang merupakan informasi tentang cara bagaimana membangkitkan fakta baru dari fakta yang sudah diketahui.

Basis pengetahuan memuat domain pengetahuan yang berhubungan dengan aspek *"apa yang harus diketahui"* dari suatu objek. Basis pengetahuan diproses oleh program yang mengandung algoritma yang dibutuhkan.

Hal penting dari pemisahan basis pengetahuan dari proses basis pengetahuan adalah ide utama dari sistem pakar barbasis pengetahuan. Keuntungan utamanya adalah aturan-aturan dapat ditambah, diubah dan dihapus dari basis pengetahuan sistem pakar ini dengan mudah.

Basis pengetahuan terdiri dari fakta-fakta dan aturan-aturan. Fakta adalah informasi tentang kondisi suatu objek dalam sistem. Aturan adalah informasi tentang bagai mana cara membangkitkan suatu fakta baru dari fakta yang diketahui.

2.3.1. Macam-macam Pengetahuan

Pengetahuan berdasarkan sumbernya dibedakan menjadi pengetahuan formal (*deep knowledge*) dan pengetahuan non formal (*shallow atau surface knowledge*). Sedangkan berdasarkan cara merepresentasikan, pengetahuan dibedakan menjadi tiga, yaitu

pengetahuan heuristik, pengetahuan prosedural dan pengetahuan deklaratif [FARI94].

Pengetahuan formal adalah pengetahuan yang terdapat dalam buku-buku, jurnal, bulletin ilmiah dan sebagainya. Pengetahuan formal ini dianggap sebagai pengetahuan bersifat umum, sedangkan pengetahuan non formal adalah pengetahuan-pengetahuan praktis dalam bidang tertentu yang diperoleh seorang pakar dari pengalaman dari bidang tersebut dalam jangka waktu yang cukup lama.

Pengetahuan heuristik adalah pengetahuan yang berbentuk hirarki. Biasanya pengetahuan heuristik ini digambarkan dalam bentuk diagram pohon pengetahuan.

Pengetahuan prosedural adalah pengetahuan yang dapat direpresentasikan sebagai suatu proses. Dalam program komputer, pengetahuan prosedural ini disimpan dalam bentuk kode.

Pengetahuan deklaratif adalah pengetahuan yang dapat disimpan bentuk berkas data, sehingga dapat disimpan secara terpisah dari program.

2.3.2. Representasi Pengetahuan

Perepresentasian dimaksudkan untuk menangkap sifat-sifat penting problema dan membuat informasi itu dapat diakses oleh prosedur pemecahan masalah [SAND93]. Bahasa representasi harus dapat

membuat seoarang pemrogram mampu mengekpresikan pengetahuan yang diperlukan untuk mendapatkan solusi masalah. Ekspresivitas dan efisiensi merupakan dimensi utama dalam melakukan evaluasi bahasa-bahasa representasi pengetahuan. Kadang-kadang, ekspresivitas harus dikorbankan untuk meningkatkan efisiensi. Namun, dilain kesempatan, representasi yang terlalu ekspesif sangat tidak efisien untuk digunakan dalam menyelesaikan masalah-masalah tertentu. Maka yang kemudian harus dilakukan adalah pengoptimalan keduanya.

Dari kesimpulan diatas maka dapat dikatakan bahwa perepresentasian seharusnya :

a. Memadai untuk mengekpresikan informasi yang diperlukan.

b. Mendukung eksekusi yang efisien dari kode yang dihasilkan.

c. Menghasilkan sesuatu yang alamiah untuk mengekspresikan pengetahuan yang dibutuhkan.

Ada empat kriteria dalam memilih teknik representasi pengetahuan, yaitu [FARI94] :

a. Kemampuan representasi, artinya teknik yang dipilih harus merepresentasikan semua jenis pengetahuan yang akan dimasukan kedalam sisitem pakar.

b. Kemudian dalam penalaran, artinya teknik yang dipilih harus mudah diproses untuk memperoleh kesimpulan.

c. Efisiensi proses akuisisi, artinya teknik yang dipilih harus membantu pemindah pengetahuan dari pakar kedalam komputer.

d. Efisiensi proses penalaran, artinya teknik yang dipilih harus dapat diproses dengan efisien untuk mencapai kesimpulan.

2.3.2.1. Metode Representasi Pengetahuan

Ada berbagai metode representasi pengetahuan yang biasa digunakan yaitu metode kalkulus predikat, bingkai (*frame*), jaringan semantic (*semantic Network*), metode kaidah produksi dan representasi logika [FARI94].

2.3.2.1.1. Bingkai (*Frame*)

Sebuah *frame* adalah sebuah struktur data yang menyimpan semua pengetahuan yang menjelaskan dan menggambarkan detail dari suatu objek [WTAE86]. Dengan kata lain semua detil tentang objek disimpan dalam satu unit yang disebut ***frame*.**

Bingkai (Frame) adalah blok-blok atau potongan-potongan yang berisi potongan mengenai objek-objek khusus, kejadian, lokasi, situasi ataupun elemen-elemen lainnya dengan ukuran relatif besar [FARI94]. Blok-blok ini menggambarkan objek-objek secara rinci. Detil

diberikan dalam bentuk rak (*slot*) yang menggambarkan berbagai atribut dan karakteristik daripada objek tersebut.

Basis pengetahuan yang direpresentasikan dengan *frame* terdiri dari kumpulan *frame*. Antara *frame* yang satu dengan yang lain dihubungkan sehingga membentuk stuktur hirarki, seperti struktur pohon. *Frame* biasanya bisa memperluas kemampuan jaringan semantik karena dapat digunakan untuk merepresentasikan objek yang kompleks dengan representasi yang sederhana, yaitu dengan jelas membungkusnya dalam suatu *frame.*

Sebuah *frame* terdiri dari dua elememn dasar, yaitu *slot* dan *facet.* Sebuah *slot* adalah sebuah atribut yang menggambarkan objek. Misalnya *frame* mobil, memiliki *slot* berat, panjang, dan warna. Setiap *slot* memiliki satu atau lebih *facet*. *Facet* sendiri terdiri dari dua komponen, yaitu *jenis facet* dan *nilai facet.* Jenis *facet* ada beberapa macam, yaitu :

1. *Value* (val), adalah jenis *facet* yang merepresentasikan nilai tertentu dari atribut yang dituliskan dalam *slot.* Sebagai contoh *slot* warna bisa memiliki *facet* value(hijau), value(merah), value(kuning).

2. *Default* (def), adalah jenis *facet* yang digunakan untuk menentukan nilai dasra dari *slot.* Jika *slot*-nya kosong, tidaj diisi dengan nilai baru, maka yang berlaku adalah nilai dasar tersebut. Sebagai contohsebuah *frame* memilki *slot* jumlah roda memiliki *facet* default (4). Jika *facet* tersebut diisi dengan nilai lain maka nilai yang dipakai adalah 4.

3. *Range* (range), adalah jenis facet yang digunakan hanya untuk slot yang bertipe nemeric. Misalnya slot yang memiliki kemungkinan nilai dari 1 sampai dengan 100.

4. *If Added* (add), adalah jenis *facet* yang mengandung pengetahuan prosedural. Jika nilai *slot* yang bersangkutan ditambah atau diubah maka *facet* ini akan mengaktifkan sebuah prosedur. Prosedur seperti ini diseut dengan *demons*.

5. *If Needed* (need), adalah jenis *facet* yang mengaktifkan prosedur ketika tidak ada nilai dalam *slot* dan nilai tersebut diperlukan. Prosedir tersebut berisi prosedur untuk meminta masukan data dari *user* atau prosedur untuk menghitunh sebuah nilai berdasarkan nilai lain yang ada.

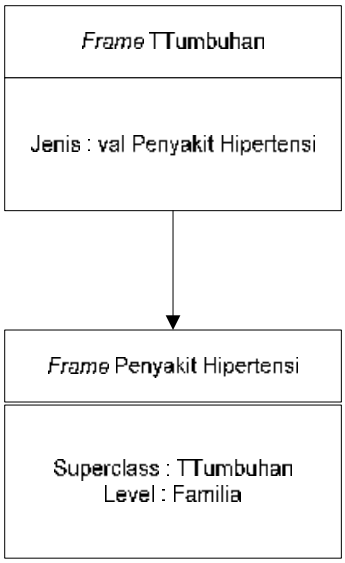
6. *Other*, adalah jenis *facet* yang beisi *pointer* yang menunjuk ke *frame* lain, atau yang menunjuk pada sebuah aturan, atau yang sesuai dengan kebutuhan.

2.3.2.1.2. Hirarki Frame

Frame-frame dalam basis penetahuan ada yang berdiri sendiri, tidak memiliki relasi satu sam lain dan ada yang diorganisasikan menjadi satu sturktur hirarki atau struktur pohon dengan relasi seperti AKO (*a kind of*),*superclass specification of.* Jika *frame* dalam basis pengetahuan

diorganisasikan secara hirarki, maka hal ini menyediakan fasilitas untuk pewarisan. *Frame* yang ada di hirarki bawah akan mewarisi atribut dai frame yang diatasnya. Makin tinggi posisi *frame* dalam struktur pohon, makin umum deskripsi yang diberikan tentang objek.

Ketika kita menggambarkan sebuah objek fisik, seperti tanaman obat, maka dilakukan instansiasi terhadap *frame* yang hirarki relatif bawah. Hasil instansiasi ini disebut *frame child.* Contoh *frame* dan hirarki *frame* dapat dilihat pada gambar dibawah ini :



Gambar 2.2. Representasi Frame dalam Bentuk Diagram

Pada gambar diatas dapat dilihat bahwa *frame* Mtanaman Obat memiliki *slot* ‘jenis memiliki *facet* dan val (Penyakit Hipertensi)’. Relasi antara *frame* memakai *superclass.* Arti dari hubungan ini tergantung pada *user* yang mendefinikannya. Misal *frame* Penyakit Hipertensi mewarisi *frame* TTumbuhan Obat.

2.3.2.1.3.Inferensi dengan Frame

Penalaran dengan *frame* lebih kompleks daripada penalaran dengan aturan. Slot yang kosong dapat diisi berdasarkan kondisi tertentu yang harus dipenuhi, dengan mengambil data dari *slot* lain yang berisi nilai. Jadi penalaran berbasis pada *frame* akan mirip dengan konfirmasi terhadap dugaan (slot yang sudah terisi) dan sering kali juga melibatkan pengisian *slot* (*uncofimed expectation*) dengan nilai tertentu berdasarkan *current* situasi.

2.3.3. Akusisi Basis Pengetahuan

Pada proses akusisi pengetahuan, dilakukan pengumpulan seluruh fakta yang diperlukan baik fakta yang teramati maupun fakta yang dibangkitkan. Kaitan antara fakta-fakta tersebut diwujudkan dalam bentuk aturan-aturan.

Pada tahap akusisi pengetahuan ini biasanya dikerjakan oleh seorang *knowledge engineer*. *Knowledge engineer* adalah orang yang memiliki latar belakang pengetahuan tentang komputer dan mengerti cara pengembangan sistem pakar [FARI94].

Dalam tahap akusisi pengetahuan ini, seorang *knowledge engineer* berusaha menyerap pengetahuan. Pengetahuan diperoleh dari seorang pakar, dilengkapi dengan buku, jurnal ilmiah dan sebagainya. Pengetahuan

yang diperoleh haruslah selengkap mungkin, sebab akan mempengaruhi kemampuan sistem pakar yang akan dibuat.

Pengetahuan dari pakar dapat diperoleh melalui wawancara. Dalam hal ini *knowledge engineer* menyodorkan permasalahan dan pakar menjelaskan proses penyelesaian masalah tersebut. Selama kegiatan ini *knowledge engineer* harus sabar, komunikatif dan kreatif sebab seringkali pakar sulit mengekspresikan proses tersebut.

Teknik-teknik dan metode untuk memperoleh pengetahuan dari pakar [FARI94], antara lain :

Tabel 2.1. Teknik untuk memperoleh pengetahuan dari pakar

Teknik	**Penjelasan**
Observasi	Melihat langsung pakar menyelesaikan masalah dilapangan.
Diskusi Masalah	Menggali data, pengetahuan dan prosedur yang dibutuhkan untuk menyelesaikan masalah dari pakar.
Deskripsi Masalah	Pakar mendefinisikan masalah pada setiap kategori solusi dalam domain permasalahan.
Analisis Permasalahan	Memberikan beberapa persoalan kepada pakar untuk menyelesaikan raingkaian penalaran.

Tabel 2.2. Metode untuk memperoleh pengetahuan dari pakar

Metode	**Penjelasan**
Tatacara Perbaiakan	Pakar memberikan beberapa masalah untuk diselesaikan oleh *knowledge engineer* dan pakar memperbaiki cara penyelasaian tersebut berdasarkan aturan dari hasil wawancara.
Tatacara Pengujian	Pakar mengevaluasi dan mengkritik prototype kaidah dan struktur pengendalian dari sistem yang dibangun.
Tatacara Validasi	*Knowledge Engineer* membentuk prototype sistem pakar berdasarkan hasil penyelesaian masalah yang diajukan lalu diuji oleh pakar lain.

2.3.4. Rekayasa Basis Pengetahuan

Proses memetakan sumber pakar hasil dari proses akuisisi pengetahuan ke dalam basis pengetahuan sistem pakar disebut *rekayasa pengetahuan.* Orang yang bertugas memetakan pengetahuan sumber pakar ke basis pengetahuan sistem pakar disebut *knowledge angineer.*

Hal penting dalam rekayasa pengetahuan adalah membuat basis pengetahuan yang akurat sesuai dengan keadaan yang sebenarnya. Agar

basis pengetahuan yang dibuat akurat perlu dilakukan pengujian oleh pakar sebelum basis pengetahuan diimlementasikan.

Pada tahap rekayasa basis pengetahuan ini *knowledge engineer* mulai menentukan teknik-teknik yang digunakan dalam representasi basis pengetahuan, stategi kontrol, bentuk antarmuka dan perangkat pengembangan untuk implementasi. Hasil dari rekayasa pengetahuan adalah basis pengetahuan yang terstruktur, sistematik dan lengkap, yang telah dikodekan sesuai dengan perangkat lunak yang dipilih.

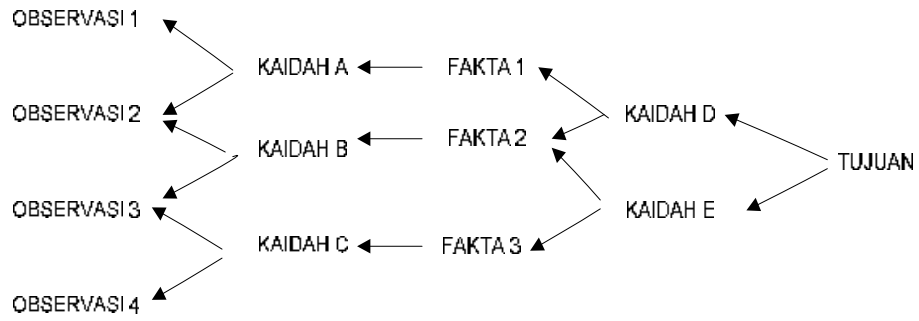
Untuk memperoleh sistem pakar yang benar-benar mewakili pakar perlu dilakukan pengujian terhadap hasil implementasi secara keseluruhan.

2.4. Mesin Inferensi

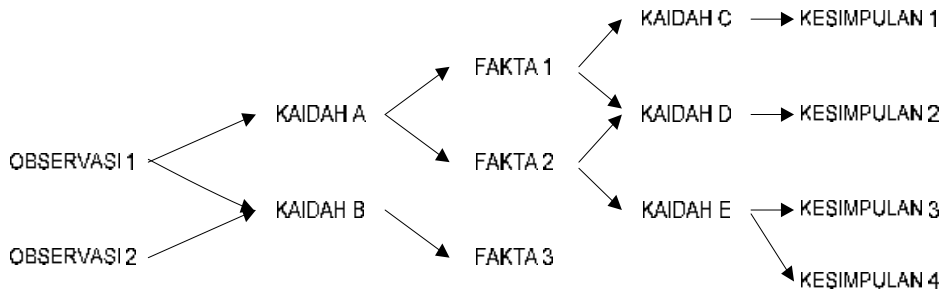
Mesin Inferensi adalah bagian yang mengandung mekanisme fungsi berfikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini akan menganalisa suatu masalah tertentu dan selanjutnya akan mencari jawaban atau kesimpulan yang terbaik.

Ada dua tipe teknk infernsi yaitu **pelacakan kebelakang *(backward chaining)*** yang memulai penalarannya dari sekumpulan hipotesa menuju fakta-fakta yang mendukung hipotesa trsebut dan **pelacakan kedepan *(forward chaining)*** yang merupakan kebalikan dari

pelacakan kebelakang yaitu memulai dari sekumpulan data menuju kesimpulan [FARI94].



Gambar 2.3. Pelacakan ke belakang



Gambar 2.4. Pelacakan ke depan

2.5. Antarmuka Pemakai

Antarmuka pemakai adalah bagian penghubung antara program sistem pakar dengan pemakai. Pada bagian ini akan terjadi dialog antara program dan pemakai. Program akan memberikan fasilitas untuk konsultasi masalah mengajukan pertanyaan-pertanyaan yang sedag dihadapi atau berbentuk menu pilihan. Program sistem pakar akan mengambil kesimpulan berdasarkan pertanyaan dari pemakai tadi.

Antarmuka merupakan fasilitas untuk berkomunikasi antara pemakai dengan sistem pakar. Misalnya untuk memasukan data, menjelaskan suatu aksi, memberikan saran, membagun basis pengetahuan (sebagai modul untuk modifikasi pengetahuan) dan sebagainya.

2.6. Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak adalah hasil rekayasa sistem menggunakan perangkat keras yang terdiri atas tiga kunci utama yaitu metode, perangkat, dan prosedur yang memungkinkan manajer mengontrol proses pembangunan perangkat lunak dan menyediakan dasar pembuatan perangkat lunak dengan kualitas tinggi melalui cara produktif [PRES92].

Dalam membangun suatu perangkat lunak selalu timbul barbagai macam kesulitan. Kesulitan yang terdapat dalam pembangunan perangkat lunak dapat diatasi dengan cara mengenali permasalahan yang ada dan penyebab tumbulnya permasalahan tersebut. Tapi solusi ini harus dapat memberikan bantuan yang berguna bagi pembangunan perangkat lunak, meningkatkan kualitas perangkat lunak dan akhirnya menerapkan perangkat lunak pada perangkat keras.

2.6.1. Paradigma Pengembangan Perangkat Lunak Yang Digunakan

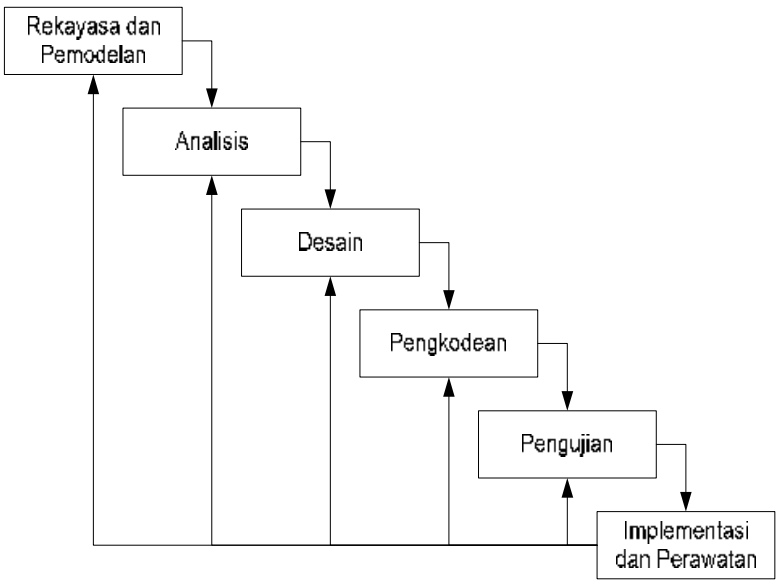
Beberapa metodologi pembangunan perangkat lunak dikembangkan orang. Masing-masing metodologi mempunyai kelebihan

dan kekurangan. Walaupun demikian setiap pendekatan yang ada berusaha agar dapat menghasilkan suatu perangkat lunak yang baik dan sesuai dengan kebutuhan pemakai.

2.6.2. Tahapan-tahapan dalam Pengembangan Rekayasa Perangkat Lunak

Dari sekian banyak model yang ada, untuk membangun aplikasi perangkat lunak sistem pakar untuk mengidentifikasi tanaman obat tradisional ini menggunakan metode *waterfall* karena pada tahapan model ini berurutan dan memudahkan penerapannya.

Langkah-langkah dari rancangan sistem dengan menggunakan model *waterfall* ini dapat dilihat dari gambar berikut dibawah ini :



Gambar 2.5. Urutan Kejadian Pada Metode Waterfall

Paradigma *waterfall* dengan langkah sebagai berikut :

a. **Rekayasa dan Pemodelan,** mendefinisikan sistem, batasan-batasan, tujuan, dan kebutuhan-kebutuhan pembangunan sistem yang menyangkut sumber daya perangkat keras, manusi, biaya serta jadwal (harus ada dokumentasi).

b. **Analisis,** memerluka kebutuhan yang difokuskan pada perangkat lunak, pemahaman tentang domain informasi, fungsi, kelakuan (*behavior*).

c. **Deain,** Performasi, interaksi sesuai dengan kebutuhan dari keinginan pemakai kebutuhan-kebutuhan sistem dan perangkat lunak harus didokumentasikan dan didiskusikan dengan pelanggan (caon pemakai).

d. **Pengkodean,** tahapan penerjemahan hasil perancangan (detil) kedalam program-program dengan menggunakan bahasa pemograman yang sesuai dengan pemakai. Hasil implementasi harus didokumentasikan.

e. **Pengujian,** program yang telah dibuat harus diuji. Proses pengujian difokuskan pada kebenaran logika internal perangkat lunak dan fungsional sistem serta interaksi antara sistem dengan pemakai. Hasil

pengujian harus didokumentasikan lengkap dengan atribut-atribut penguji (siapa, kenapa, jenis keslahan).

f. **Implementasi dan Perawatan,** sistem yang telah diuji, kemudian diserahkan kepada pamaki untuk diinstall dan diopersikan sesuai dengan kebutuhannya. Tahapan perawatan dibutuhkan dalam masa itu untuk dilakukan pengecekan keslahan operasionalnya atau perubahan yang diinginkan pemakai (merupakan tahapan terpanjang dan harus ditentukan kapan waktunya).

2.7. Permasalahan Pada Tanaman Obat

Sejak ribuan tahun yang lalu, abad dan pengobatan tradisional sudah ada di Indonesia, jauh sebelum pelayanan kesehatan formal dengan obat-obatan modernnya dikenal masyarakat. Pengobatan tradisional dengan memanfaatkan tumbuhan berkhasiat obat merupakan pengobatan yang dimanfaatkan dan diakui masyarakat dunia yang menandai kesadaran untuk kembali ke alam (*back to nature*) adalah untuk mencapai kesehatan yang optimal dan untuk mengatasi berbagai penyakit secara alami.

Indonesia memiliki lahan hutan tropis cukup luas dengan keanekaragaman hayati, baik flora maupun fauna. Kita boleh berbangga dengan kekayaan tumbuhan brkhasiat obat yang tidak dimiliki negara lain. Lebih kurang 30.000 sampai 40.000 jenis tumbuhan tersebar diseluruh kepulauan Indonesia dan kekayaan laut dapat dimanfaatkan sebagai bahan

obat. Dalam rangka memanfaatkan tumbuhan berkhasiat obat tersebut, haruslah dipikirkan juga bagaimana agar tumbuhan tersebut tidak punah.

2.8. Representasi Pengetahuan Tanaman Obat Tradisional

2.8.1. Permasahan Tanaman Obat Tradisional

Keberadaan pengobatan tradisioanal Indonesia saat ini dikhawatirkan hilang karena pengobatan tradisional dianggap kuno, tidak ilmiah, tidak rasioal, kampungan, primitif karena tidak dilakukan uji klinis. Selain itu minat masyarakat juga kurang karena informasi yang diperoleh sangatlah kurang meskipun sekarang banyak klinik-kinik pengobatan tradisional dan buku-buku pengobatan yang informasinya kurang, tetapi hal ini sering menyebabkan masyarakat kurang percaya dengan pengobatan tradisioanal. Untuk mengatasi masalah ini peran seorang pakar sangat dibutuhkan untuk mengetahui dan mempelajari tanaman berkhasiat obat dengan melakukan tanya jawab langsung sehingga dapat menghasilkan keterangan yang maksimal atas penjelasan dari pakar tersebut.

Permasalahan lain dapat timbul ketika masyarakat sedang membutuhkan keterangan dan penjelasan dari pakar yang kadang terbentur dengan masalah waktu dan keadan yang tidak memungkinkan. Dimana terdapat sat pakar itu sedang tidak ada ditempat sementara keterangan dan penjelasannya sangat dibutuhkan, atau ternyata pakar itu sedang sakit

sehingga tidak dapat ditemui, bahkan ketika pakar itu sendiri telah tiada suatu saat nanti.

Hal ini dapat menjadi faktor penghambat bagi orang yang ingin mengetahui bagaimana cara mengidentifikasi tanaman obat akan mendapat kesulitan.

2.8.2 Sekilas Permasalahan Penyakit Darah Tinggi (*Hipertensi*)

Tekanan darah tinggi (*hipertensi*) memang merupakan penyakit yang menakutkan. Apabila gejalanya muncul tiba-tiba, kenyamanan hidup penderita menjadi sangat terganggu. Seperti banyak kronis lainnya, pada hipertensi pun berbagai penyakit dapat menyertai dan timbul bersamaan sehingga dapat memperparah keadaan. Sewaktu-waktu penderita dapat mengalami kelumpuhan bahkan kematian akibat stroke.

Seseorang dikatakan menderita penyakit tekanan darah tinggi bila nilai tekanan darah sistoliknya diatas 140 mm Hg dan diastoliknya diatas 90 mm Hg. Hipertensi yang diketahui penyebabnya disebut hipertensi sekunder, jumlahnya kurang dari10% dari seluruh penderita. Hipertensi sekunder disebabkan antara lain oleh penyakit parenkhim ginjal, penyakit pembuluh darah ginjal, akibat pemakaian obat dan lain-lain. Riwayat keluarga juga merupakan salah satu faktor yang berperan pada hipertensi esensial. Biasanya bila salah satu atau bahkan kedua orang tuanya menderita hipertensi maka anak-anaknya dapat menderita hipertensi juga.

Secara umum gejala yanng dikeluhkan oleh penderita hipertensi adalah sakit kepala, rasa pegal dan tidak nyaman pada tengkuk, perasaan berputar seperti tujuh keliling serasa ingin jatuh, berdebar atau detak jantung terasa cepat, dan telinga berdenging.

2.9. Analisa Sistematika dan Morpologi Tanaman Obat untuk Penyakit Tekanan Darah Tinggi (Hipertensi)

2.9.1. Sistematika Tanaman Obat

a. Familia : *Amaranthaceae*

Nama Latin : *Amaranthus spinosus L.*

Nama Daerah : Bayam Duri

b. Familia *Amaranathaceae*

Nama Latun : *Celosia argentea L.*

Nama Daerah : Boroco / Bayam Ekor Belanda

c. Familia : *Rubiaceae*

Nama Latin *Morinda cirtifolia L.*

Nama Daerah : Mengkudu

d. Familia : *Euphobiaceae*

Nama Latin : *Phylianthus urinaria Linn.*

Nama Daerah : Meniran

e. Familia : *Umbelliferae* (*Apiaceae*)

Nama Latin : *Centella asiantika L.*

Nama Daerah : Pegagan / Daun Kaki Kuda

f. Familia : *Apocynceae*

Nama Latin : *Rauvolfia Serpentina* (L.) *Bethan ex Kurz.*

Nama Daerah : Pule Pandak / Akar Tikus

g. Familia : *Umbelliferae* (*Apiaceae*)

Nama Latin : *Hydrocotyle sibthorpioides Lam.*)

Nama Daerah : Semanggi Gunung / Patikan Cina

h. Familia : *Apocynaceae*

Nama Latin : *Catharanthus roseus* (L) *G.Don.*

Nama Daerah : Tapak Dara

i. Familia : *Compositae* (*Asteraceae*)

Nama Latin : *Sonchus arvensis L.*

Nama Daerah : Tempuyung

j. Familia : *Compositae.*

Nama Latin : *Helianthus annuus L.*

Nama Daerah :Bunga Matahari

k. Familia : *Apocynaceae*

Nama Latin : *Ervatamia divaricata* (L.) *Burk.*

Nama Daerah : Bunga Mondokaki

l. Familia : *Apocynaceae*

Nama Latin : *Alstonia scholaris* (L.) *R. Br*

Nama Daerah : Pulai

m. Familia : *Acanthaceae*

Nama Latin : *Andrographis paniculata Ness.*

Nama Daerah : Sambiloto

n. Familia : *Liliaceae*

Nama Latin : *Allium sativum, Linn.*

Nama Daerah : Bawang Putih

o. Familia : *Rubiaceae*

Nama Latin : *Ixora stricter Roxb.*

Nama Daerah : Siantan

p. Familia : *Theaceae*

Nama Latin : *Camellia sinensis (L.) Kunze)*

Nama Daerah : Teh

q. Familia : *Cariccaceae*

Nama Latin : *(Carica papaya, Linn.)*

Nama Daerah : Pepaya

r. Familia : *Lauraceae*

Nama Latin : *(Persea gratissima gaertn.)*

Nama Daerah : Alpukat

s. Familia : *Oxalidaceae*

Nama Latin : *(Averhoa carambola.)*

Nama Daerah : Belimbing Manis

t. Familia : *Cucurbitaceae*

Nama Latin : *(Cucumis sativus L.)*

Nama Daerah : Ketimun

u. Familia : *Meliaceae*

Nama Latin : *(Swietenia mahagoni Jacq.)*

Nama Daerah : Mahoni

v. Familia : *Cannaceae*

Nama Latin : *(Canna indica Linn.)*

Nama Daerah : Bunga Tasbih

w. Familia : *Euphorbiaceae*

Nama Latin : *(Antidesma bunius (L.) Spreng.)*

Nama Daerah : Buni

x. Familia : *Oxalideceae*

Nama Latin : *(Oxalis corniculata Linn.)*

Nama Daerah : Calingcing

y. Familia : *Manispermaceae*

Nama Latin : *(Cylea barbata, Miers.)*

Nama Daerah : Cingcau

z. Familia : *Planfaginaccae*

Nama Latin : *(Plantago mayor L.)*

Nama Daerah : Daun Sendok

aa. Familia : *Labiyatae*

Nama Latin : *(Leonurus sibiricus L.)*

Nama Daerah : Ginjean

2.9.2. Morfologi Tanaman Obat

Morfologi tanaman obat adalah sebagai berikut :

Tabel 3.1 Tabel Morfologi Tanaman Obat

Nama Tanaman Obat	Ciri-ciri Tanaman Obat
Bayam Duri (*Amaranthus spinosus L.*)	a. Batang (Bentuk bulat berkayu dan berambut halus, Warna kemerahan, Tinggi 30-100 cm, Cara tumbuh tegak) b. Daun (Bentuk daun bulat telur memanjang, Bentuk tulang daun menonjol dan tepi rata (beringgit)) c. Bunga (Bentuk bunga tunggal, Warna hijau keputihan)
Boroco / Bayam Ekor Belanda (*Celosia argentea L.)*	a. Batang (Bentuk batang bulat dengan alur kasar memanjang, Warna Ungu, Tinggi 30-100 cm, Cara tumbuh tegak) b. Daun (Bentuk daun bulat memenjang, Bentuk tulang daun menyirip dan tepi bergerigi) c. Bunga(Bentuk bunga bulat silindris, Warna ungu) d. Buah (Bentuk bulat telur sungsang, Warna hitam cerah)
Mengkudu (*Morinda citrifolia L.)*	a. Batang (Bentuk batang bersegiempat, warna coklat gelap, Tinggi mencapai 3-8 meter, Cara tumbuh tegak)

	b. Daun (Bentuk daun bulat telur samapai elips, Bentuk tulang daun menyirip dan tepi rata) val Bunga (Bentuk bunga seperti terompet, warna putih) c. Buah (Bentuk bulat lonjong, Warna kuning pucat)
Meniran (*Phylianthus urinaria Lam.)*	a. Batang (Warna hijau pucat atau hijau kemerahan, Tinggi sekitar 50 cm, Cara tumbuh tegak) b. Daun (Bentuk daun bulat telur memanjang bagian ujung tumpul atau runcing) c. Buah (Bentuk bulat)
Pegagan / Daun Kaki Kuda (*Centella asiatica L.*)	a. Daun (Bentuk daun bulat atau ganjil, Bentuk tulang daun menyirip dan tepi daun bergigi) b. Bunga (Bentuk bunga seperti payung, Warna putih atau merah muda) c. Buah (Bentuk lonjong atau pipih)
Pule Pandak / Akar Tikus (*Ramvolia Serpentina (L.) Betham ex Kurz.*)	a. Batang (Bentuk batang silindris, Warna cokelat abu-abu, Tinggi batang mencapai 1 meter, Cara tumbuh tegak) b. Daun (Bentuk daun bulat telur memanjang, Bentuk tulang menyirip dan tepi rata) c. Bunga (Bentuk bunga seperti payung, Warna bunga putih) d. Buah (Bentuk bulat lonjong, Warna hitam)
Semanggi Gunung / Patikan Cina	a. Batang (Bentuk batang bulat lunak dan berongga, Warna hijau, tinggi 10-30 cm, Cara tumbuh

(*Hydrocotyle sibthorpioides Lam.*)	merayap) b. Daun (Bentuk daun bulat atau ginjal) c. Bunga (Bentuk seperti bonggol, warna kuning) d. Buah (Warna merah muda kuning)
Tapak Dara (*Catharanthus roses (L.) G.Don)*	a. Batang (Bentuk bulat berkayu, Warna kemerahan, Tinggi mencapai 120 cm, Cara tumbuh tegak) b. Daun (Bentuk daun bulat telur memanjang, Bentuk tulang daun menyirip dan tepi rata) c. Bunga (Bentuk seperti terompet, Warna putih, ungu muda atau putih) d. Buah (Bentuk buah bumbung berbulu, Warna hijau pucat)
Tempuyang (*Sonchus arvensis L.*)	a. Batang (Bentuk batang bulat berongga dan bersusuk, Warna hijau, Tinggi mencapai 0,6-2 m, Cara tumbuh tegak) b. Daun (Bentuk daun lonjong atau lanset, Bentuk tulang daun menyirip dan tepi rata) c. Bunga (Bunga berbentuk bonggol, Warna kuning cerah) d. Buah (Bentuk buah pipih memanjang, Warna cokelat kekuningan)
Bunga Matahari (*Helianthus annuus L.)*	a. Batang (bentuk batang bulat berbulu dan batang basah, Warna hijau, tinggi mencapai 2 meter, cara tumbuh tegak)

	b. Daun (Bentuk daun seperti jantung, bentuk tulang daun menyirip dan tepi rata) c. Bunga (Bentuk seperti cawan, warna kuning)
Bunga Mondokaki (*Ervatamia divaricata (L.) Burk.)*	a. Batang (Bentuk batang bulat berkayu, warna hijau gelap, tinggi 0,5-3 m, Cara tumbuh tegak) b. Daun (Bentuk daun bulat telur memanjang, Bentuk tulang daun menyirip dan tepi rata) val Bunga (Warna putih) c. Buah (Bentuk buah bulat panjang, Warna merah)

2.0. Klasifikasi Tanaman Obat untuk Penyakit Hipertensi

Klasifikasi tanaman obat ini berisikan tentang kandungan kimia, bagian ynag dipakai, dan cara pengolahan yanng merupakan basis pengetahuan dari tanaman obat untuk mengatasi hipertensi, yaitu :

a. **Bayam Duri** (*Amaranthus spinosus L.*)

- Cara Pengolahan :
  - Bahan : Sepotong akar bayam duri berikut bonggolnya.
  - Cara membuat : Akar berikut bonggol bayam duri dicuci bersih, potng-potong seperlunys rebus dengan 2 gelas air bersih sampai tersisa 1 gelas.
  - Aturan pemakaian : Diminum sekaligus.

b. **Boroco / Bayam Ekor Belanda** (*Celosia argentea L.*)

- Cara Pengolahan :
  - Bahan : Biji boroco sebanyak 30 gram.
  - Cara membuat : Biji direbus dengan 1 gelas air bersih sampai tersisa ½ gelas, setelah dingin disaring.
  - Aturan pemakaian : Dibagi untuk 2 kali minum.

c. **Mengkudu** (*Morinda citrifolia L.*)

- Cara Pengolahan :
  - Bahan : Buah mengkudu yang telah masak sebanyak 2 buah.
  - Cara membuat : Buah mengkudu dicuci bersih lalu diparut, peras lalu saring airnya untuk diminum.
  - Aturan Pemakaian : lakukan 2-3 kali sehari.

d. **Meniran** (*Phylianthus urinaria Linn*)

- Cara Pengolahan :
  - Bahan : Daun meniran ¼ genggam.
  - Cara membuat : Daun meniran dicuci bersih lalu digiling halus, tambahkan ½ cangkir air matang, peras dan saring.

- Aturan Pemakaian : Air yang telah disarig diminum 2-3 kali sehari.

e. **Pegagan / Daun Kaki Kuda** (*Centella asiatica L.*)

- Cara Pengolahan :

  - Bahan : Daun pegagan sebanyak 20 lembar.

  - Cara membuat : Daun dicuci bersih, masukan dalam panic, rebus dengan 3 gelas air bersih sampai tersisa 2¼ gelas, setelah dingin lalu saring.

  - Aturan Pemakaian : Dibagi untuk 3 kali minum. Minum dengan air gula secukupnya, lakukan setiap hari.

f. **Pule Pandak / Akar Tikus** (*Rauvolfia Serpentina* (L.) *Bethan ex Kurz.*)

- Cara Pengolahan :

  - Bahan : Akar pule pandak sebanyak 50 gram.

  - Cara membuat : Akar dicuci bersih, lalu dipotong tipis-tipis masukan dalam panic, tambahkan 3 gelas air bersih lalu direbus sampai tersisa 1 gelas. Setelah dingin disaring.

  - Aturan Pemakaian : Dibagi 2 kali minum, pagi dan sore hari.

g. **Semanggi Gunung / Patikan Cina** (*Hyrocotyle sibthorpioides Lam.*)

- Cara Pengolahan :
  - Bahan : Daun semanggi gunung yang masih segar sebanyak 30 gram.
  - Cara membuat : Daun semanggi gunung dicuci bersuh, rebus dengan 3 gelas air bersih sampai tersisa 1 gelas, setelah dingin disaring,
  - Aturan Pemakaian : Dimunum sekaligus.

h. **Tapak Dara** (*Catharanthus roseus* (L.) *G.Don.*)

- Cara Pengolahan :
  - Bahan : Daun tapak dara sebanyak 10 gram.
  - Cara membuat : Daun dicuci bersih, rebus dengan 3 gelas air bersih sampai tersisa 1 gelas, setelah dingin disaring.
  - Aturan Pemakaian : Diminum sekaligus.

i. **Tempuyung** *(Sonchus arvensis L.)*

- Cara Pengolahan :
  - Bahan : Daun tempuyung sebanyak 5 lembar.

- Cara membuat : Daun tempuyung dicuci bersih lalu diasapkan sebentar.
- Aturan Pemakaian : Dimakan sebagai lalab saat makan nasi, lakukan 2-3 kali sehari.

j. **Bunga Matahari** *(Helianthus annuus L.)*

- Cara Pengolahan :
  - Bahan : 60 gram dasar bunga matahari, 30 gram rambut jagung, gula secukupnya.
  - Cara membuat : Semua bahan direbus dengan 600 cc air hingga tersisa 300cc, lalu airnya saring.
  - Aturan Pemakaian : dan diminum hangat-hangat.

k. **Bunga Mondokaki** *(Ervatamia divaricata (L.) Burk,)*

- Cara Pengolahan :
  - Bahan : 15 gram daun mondokaki.
  - Cara membuat : Daun mondokaki direbus dengan 400cc air hingga tersisa 200cc lau airnya digunakan untuk menjus 100 gram seledri jenis kecil.
  - Aturan Pemakaian : Jusnya diminum.

This document was created with Win2PDF available at http://www.daneprairie.com.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.