

# Deteksi Kesalahan Eksekusi Perangkat Lunak Berbasis Sistem Operasi Windows Dengan Metode *Knowledge Acquisition*

Ade Andri Hendriadi<sup>1</sup>, Septian Maulana<sup>2</sup>

<sup>1, 2</sup> Jurusan Teknik Informatika

Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang

Jl. HS. Ronggowaluyo, Telukjambe Timur, Karawang 41361

Email: ade.andri@staff.unsika.ac.id & septian.maulana@student.unsika.ac.id

**Abstrak.** Penelitian ini berfokus pada *software* yang berjalan di *Operating System Windows*. Sistem pakar yang di rancang bersifat umum dan bebas digunakan oleh masyarakat luas, tujuan dari sistem pakar ini yaitu membantu pengguna *computer (users)* dalam mengatasi permasalahan *software*. Penelitian ini di rancang dengan menggunakan metodologi penelitian ESDLC, metode pendukung *Knowledge Acquisition*, metode inferensi *Forward Chaining* dan implementasi berbasis *web* pakar *software*, penggunaan metode *Forward Chaining* dapat mewakili representasi dalam penyajian data dari segi alur untuk merelasikan antara *troubleshooting* dan *cases*. *Forward Chaining* memberikan hasil yang cukup baik dari segi penyajian data yang bersifat *data driven*, telah dilakukan evaluasi *user* dari segi *user interface* dan *user experience* yang menghasilkan 28.48% responden memberikan penilaian kategori sangat baik, 44.24% responden memberikan penilaian kategori baik, 23.03% responden memberikan penilaian kategori cukup dan 4.24% responden memberikan nilai kurang terhadap aplikasi yang dikembangkan dan telah dilakukan evaluasi *user* dari segi kelengkapan informasi dan penyelesaian masalah yang menghasilkan 27.5% responden memberikan penilaian kategori sangat baik, 40% responden memberikan penilaian kategori baik, 25.83% responden memberikan penilaian kategori cukup, 5.83% responden memberikan nilai kurang dan 0.83% responden memberikan nilai sangat kurang terhadap aplikasi yang dikembangkan.

**Kata kunci:** *sistem pakar, forward chaining, knowledge acquisition, esdlc.*

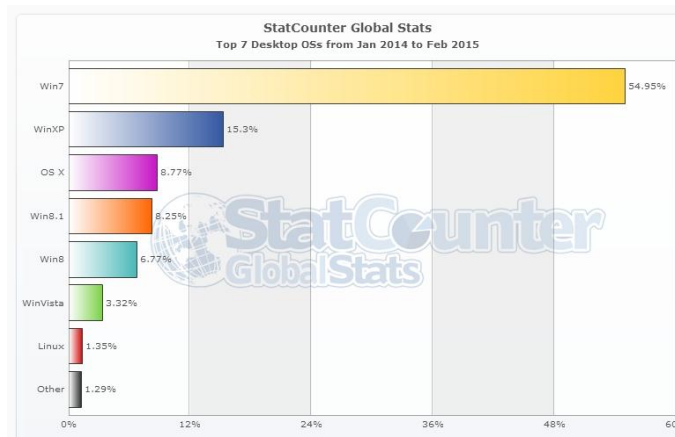
## 1 Pendahuluan

### 1.1 Latar Belakang

Pada zaman globalisasi dunia IT semakin meningkat, era komputerisasi pun tiba pada saatnya, semua sistem perekonomian, surat menyurat, administrasi, sistem pendidikan akademik, dll. Semuanya di kemas dalam bentuk sistem informasi baik dari penyimpanan *data, input data, output data* maupun melakukan proses

*data*. Seiring berjalannya waktu sistem yang sudah ada di dalam komputer pasti memiliki suatu *trouble* (permasalahan) baik itu permasalahan kecil, maupun permasalahan fatal yang berakibat kepada *problem* dari segi *software* (perangkat lunak), pada pembahasan kali ini berfokus pada *software* yang berjalan pada *Operating System Windows*, target penelitian ditargetkan terhadap kalangan yang masih mengenyam pendidikan baik itu SD, SMP, SMA/SMK maupun diperkuliahan, lalu tidak jarang juga pekerja kantoran menggunakan perangkat komputer untuk mendukung kinerja dengan apa yang ingin dikerjakan sesuai dengan kebutuhan, akan lebih ditekankan objek penelitian pada lingkungan pengguna komputer *Operating System Windows* yang bersifat umum.

Sistem di rancang dengan sistem pakar berbasis *web* dengan menggunakan metodologi rekayasa perangkat lunak ESDLC (*Expert System Development Life Cycle*), metode pengembangan sistem pakar *Knowledge Acquisition* atau bisa disebut akuisisi pengetahuan dan menggunakan metode pendukung *Forward Chaining* atau biasa disebut pencarian ke depan yang kita tahu bahwa metode *Forward Chaining* ini melakukan pendekatan pada pencarian yang di mulai dari informasi masukan. Selanjutnya mencoba untuk menarik suatu kesimpulan atau mengambil suatu gambaran kesimpulan hasil diagnosis dari masalah tertentu dan dapat menghasilkan solusi untuk pengambilan suatu keputusan.



Gambar 1 Grafik Pengguna

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah di bahas sebelumnya dan hasil dari identifikasi masalah yang dilakukan, maka di ambil perumusan masalah yaitu sebagai berikut:

1. Bagaimana merancang suatu *troubleshooting software* komputer dengan *Operating System Windows* berbasis *web* dengan menggunakan metode *Forward Chaining*?
2. Bagaimana cara menerapkan metodologi ESDLC dan metode penelitian Akuisisi Pengetahuan (*Knowledge Acquisition*) dalam mengumpulkan suatu informasi penting yang dibutuhkan?
3. Bagaimana mengimplementasikan dan menganalisis suatu *troubleshooting software* berbasis *web*?

### 1.3 Batasan Masalah

Berdasarkan latar belakang yang telah di bahas sebelumnya dan hasil dari identifikasi masalah yang dilakukan, maka di ambil perumusan masalah yaitu sebagai berikut :

1. Sistem pakar ini di rancang berbasis web menggunakan bahasa pemrograman PHP dan manajemen database MySQL.
2. Penerapan sistem bisa berupa langsung ke dalam web hosting agar bisa digunakan oleh orang banyak, namun bisa juga dijalankan di dalam localhost tapi tidak terpublikasikan.
3. Akan diberikan control admin yang dapat menambahkan Knowledge Base (KB) untuk memperkaya pengetahuan sistem pakar yang terus berkembang.
4. Interface sistem pakar akan di buat sederhana agar memudahkan pengguna awam untuk melakukan konsultasi.
5. Sistem pakar ini fokus pada permasalahan software yang terdapat dalam ruang lingkup Operating System Windows.
6. Bahan acuan referensi pengetahuan system pakar ini berasal dari official website Microsoft dan Microsoft Fix it.
7. Solusi yang diberikan harus sedetail mungkin informasinya agar mudah dipahami oleh pengguna sistem pakar *troubleshooting software*, akan di dukung oleh rincian artikel dari materi-it.com support dari pakar-software.com.

### 1.4 Tujuan Masalah

Berdasarkan perumusan masalah yang telah di bahas sebelumnya, maka dilakukan tujuan penelitian yaitu sebagai berikut :

1. Dapat merancang sistem pakar *troubleshooting software* berbasis *web* dengan menggunakan metode *Forward Chaining*.
2. Dapat menggunakan metodologi ESDLC dan metode penelitian Akuisisi Pengetahuan (*Knowledge Acquisition*).

3. Dapat mengimplementasikan aplikasi sistem pakar *troubleshooting software* berbasis *web* yang hasilnya akan di analisis dari diagnosis *problem software* berupa *cases*.

## 2 Landasan Teori

### 2.1 Sistem Pakar

Sistem pakar merupakan sistem yang berusaha mengakuisisi pengetahuan manusia ke dalam komputer dengan tujuan agar komputer dapat menyelesaikan masalah tertentu seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik di rancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awam dapat menyelesaikan berbagai permasalahan yang cukup rumit dan sebenarnya hanya dapat diselesaikan oleh para ahli pakar bidang tertentu. Sistem pakar ini akan membantu aktivitas sebagai asisten yang sangat berpengalaman bagi para ahli, mengingat bahwa para ahli tidak dapat selalu menyimpan suatu pengalaman secara utuh, lengkap dan detail dibandingkan dengan sistem pakar [1].

### 2.2 Pengertian Computer

Istilah komputer (*Computer*) berasal bahasa latin *Computare* yang bisa diartikan dengan menghitung (*to compute atau reckon*).

### 2.3 Pemrograman PHP

Pemrograman PHP yang merupakan pemrograman berbasis *web*, pengertian menurut [2] yaitu sebagai berikut :

“*PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.*”.

PHP singkatan dari PHP *Hypertext Preprocessor* yang digunakan sebagai bahasa *script server-side* dalam mengembangkan *Web* yang disisipkan pada dokumen HTML [3].

### 2.4 Basis Data

Pengertian antara *basis* dan *data* [4] menjelaskan :

Basis Data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi). Relasi biasanya ditunjukkan dengan kunci (*key*) dari tiap *file* yang ada. Dalam satu *file* terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu *record* terdiri dari field-field yang saling berhubungan dan menunjukkan dalam satu pengetahuan yang lengkap dalam satu *record*.

## 2.5 *World Wide Web*

*World wide web* menurut W3.org yaitu sebagai berikut :

“*The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.*” [5].

## 2.6 *Troubleshooting Software*

*Troubleshooting software* pada *Operating System Windows* sama halnya dengan *tools* yang sudah disediakan *Microsoft Fix it* yaitu sebagai berikut :

“*Automatically diagnose and repair common software problems in Windows, Internet Explorer or other Microsoft products with Microsoft Fix it solutions.*” [6].

## 2.7 *Metode Knowledge Acquisition*

Akuisisi Pengetahuan adalah proses pembangunan sistem, transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program *computer*. Menurut Firebaugh (1989) terdapat enam metode utama dalam Akuisisi Pengetahuan (*Knowledge Acquisition*) yaitu sebagai berikut [7]:

### 1. Tahap Identifikasi

Tahap Identifikasi meliputi penentuan komponen-komponen kunci dalam sistem yang sedang di bangun. Komponen kunci ini adalah *knowledge engineer*, pakar, karakteristik masalah, sumber daya dan tujuan. *Knowledge engineer* dan pakar bekerja bersama untuk menentukan berbagai aspek masalah, seperti dari proyek, *data input* yang dimasukkan, bagian-bagian penting dan interaksinya, bentuk dan isi dari penyelesaian dan kesulitan-kesulitan yang mungkin terjadi dalam pembangunan sistem.

Perlu diketahui sumber pengetahuan seperti *basis data*, sistem informasi manajemen, buku *text*. Selain menentukan sumber pengetahuan, pakar juga mengklarifikasi dan menentukan tujuan-tujuan sistem dalam proses penentuan masalah, identifikasi masalah berusaha untuk mengetahui apa saja yang dibutuhkan dalam sistem.

### 2. Tahap Konseptualisasi

Konsep-konsep kunci dan hubungannya yang telah ditentukan pada tahap pertama dibuat lebih jelas dalam tahap konseptualisasi. Hasil identifikasi masalah dikonseptualisasikan dalam bentuk relasi antar *data*, hubungan antar pengetahuan dan konsep-konsep penting yang akan diterapkan dalam sistem.

### 3. Tahap Formalisasi

Tahap ini meliputi pemetaan konsep-konsep kunci, sub-masalah dan bentuk aliran informasi yang telah ditentukan dalam tahap-tahap sebelumnya ke dalam representasi formal yang paling sesuai dengan masalah yang ada.

4. Tahap Implementasi

Tahap ini meliputi pemetaan pengetahuan dari tahap sebelumnya yang telah diformalisasi ke dalam skema representasi pengetahuan yang dipilih.

5. Tahap Pengujian

Setelah sistem yang di bangun dalam tahap sebelumnya berhasil menangani dua atau tiga contoh, sistem tersebut harus menjalani serangkaian pengujian dengan teliti menggunakan beragam sampel masalah. Masalah-masalah yang ditemukan dalam pengujian ini biasanya dapat dibagi dalam tiga kategori, yaitu kegagalan *input/output*, kesalahan logika dan strategi kontrol, suatu sistem harus menjalani serangkaian pengujian dengan teliti menggunakan beragam sampel masalah lainnya.

6. Pemeliharaan

Suatu unsur penting pada semua tahap dalam proses akuisisi pengetahuan adalah kemampuan untuk kembali ke tahap-tahap sebelumnya untuk memperbaiki sistem. Tahap pemeliharaan ini sangat penting agar sistem yang di bangun tidak *out of date* dan tidak terawat karena jarang dilakukan *maintenance*.

## 2.8 *Forward Chaining*

*Forward chaining* adalah suatu rantai yang dicari atau dilintasi dari suatu permasalahan untuk memperoleh solusi dengan penalaran dari fakta menuju konklusi yang terdapat dari fakta. [7].

## 2.9 Representasi Pengetahuan

Representasi pengetahuan merupakan suatu metode yang digunakan untuk mengkodekan pengetahuan dalam sebuah sistem pakar yang berbasis pengetahuan. Representasi disini maksudnya untuk menangkap suatu sifat-sifat terpenting dari suatu permasalahan agar dibuatkan suatu informasi yang dapat menangani suatu permasalahan [8].

### 3 Objek dan Metodologi Penelitian

#### 3.1 Objek Penelitian

Penelitian dengan mengambil objek dari sistem pakar yang di teliti saat ini yaitu berupa *software* suatu perangkat komputer. Adapun penelitian ini dilakukan untuk membantu penyelesaian masalah bagi pengguna awam komputer (*user*) dalam menyelesaikan masalahnya sendiri yang berkaitan dengan *software* komputer.

Sistem pakar *troubleshooting software* komputer di rancang untuk mengatasi berbagai permasalahan tentang *software* komputer yang disebabkan oleh pengguna komputer (*user*) itu sendiri yang menyebabkan *software* komputer mengalami *problem* ataupun berbagai kendala dan permasalahan karena ketidaktahuan pengguna komputer (*user*) dalam ruang lingkup *software* komputer yang berfokus pada *Operating System Windows*.

Sebagian permasalahan yang sering terjadi dalam ruang lingkup *Operating System Windows* yaitu sebagai berikut :

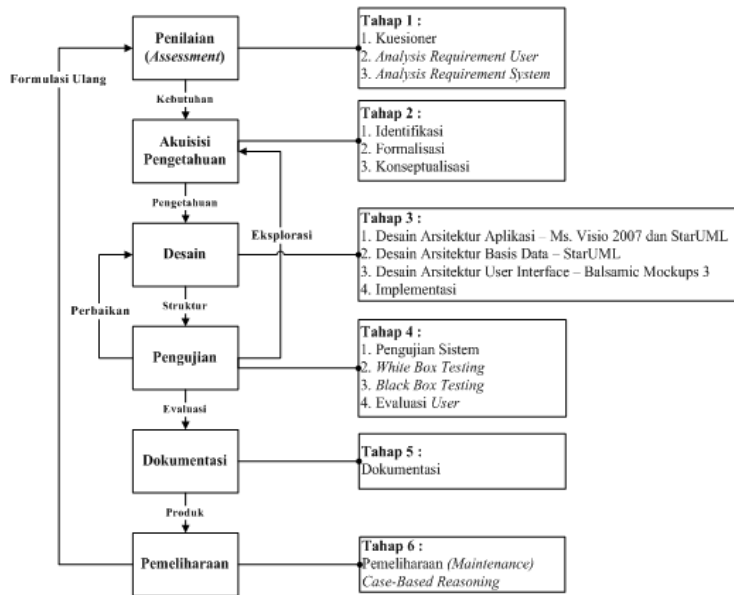
1. *Bluescreen*
2. *System Crash*
3. *Slow System*
4. *System Corrupt*
5. *Driver Error*
6. *Virus*
7. *Hang Problem*
8. *Not Responding*
9. *Dump Registry*
10. *Missing File*

Masih banyak permasalahan yang sering dihadapi pada *Operating System Windows* dan biasanya pengguna komputer (*user*) membiarkannya dan akhirnya menyebabkan *problem* yang lebih fatal. Aplikasi sistem pakar *troubleshooting software* di rancang dan diterapkan berbasis *web* yang bertujuan untuk memberikan informasi bermanfaat dari berbagai permasalahan *problem software* yang dihadapi pengguna komputer (*user*).

#### 3.2 Metodologi Penelitian

Metodologi yang digunakan dalam sistem pakar *troubleshooting software* komputer berbasis *web* yang sedang dilakukan proses perancangan penelitian yaitu menggunakan metode ESDLC (*Expert System Development Life Cycle*). Sistem pakar ini akan membahas beberapa rencana yang akan diterapkan

kepada sistem yang sedang dilakukan penelitian, di dalam tahapan ESDLC terdapat metode pendukung yang berguna untuk membuat suatu penelitian lebih spesifik dengan objek yang sedang diteliti. Metode pendukung dalam tahapan model penelitian ini yaitu berupa metode struktur sistem pakar yaitu Akuisisi Pengetahuan (*Knowledge Acquisition*), mesin inferensi *Forward Chaining*, penalaran berbasis kasus (*Case-Based Reasoning*), Kuesioner dan Evaluasi User yaitu *Rating Scale*.



Gambar 2 Rancangan Penelitian

## 4 Hasil dan Pembahasan

### 4.1 Penilaian

Penilaian adalah tahap awal dalam perancangan sebuah perangkat lunak (*software*) dalam skripsi ini yang menggunakan Metode ESDLC (*Expert System Development Life Cycle*). Pada tahap penilaian menghasilkan berupa *data* penting guna kebutuhan pembangunan sistem yang ditujukan kepada *user*.

### 4.2 Identifikasi

Proses identifikasi dilakukan dengan menyiapkan beberapa fitur, fasilitas dan sumber daya untuk mengembangkan pengetahuan tentang *troubleshooting software* komputer pada *Operating System Windows*.



### 4.3 Konseptualisasi

Konseptualisasi dilakukan dengan memberikan informasi secara rinci tentang *cases* dan solusi penyelesaian masalah (*troubleshooting*) yang akan dikelompokkan menjadi lima kategori permasalahan yaitu *system windows*, *bluescreen*, *driver*, *security* dan *network software* komputer.

Masalah *problem software* yang di bahas sangat luas maka dari itu sangat dibutuhkan pengembangan penambahan pengetahuan dengan berbagai *cases* yang ada dan dirasakan oleh *user* akan disampaikan melalui formulir *feedback* yang khusus disediakan dalam sistem pakar *troubleshooting software*.

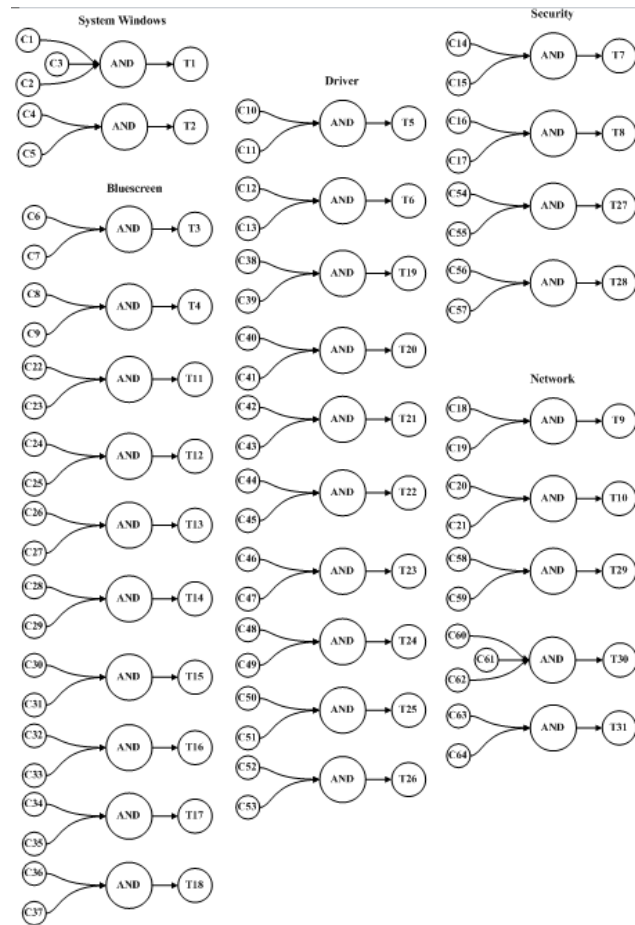
### 4.4 Formalisasi

Formalisasi dilakukan pemetaan konsep kunci, sub masalah dan aliran informasi yang saling berhubungan antara *Cases* dengan *Troubleshooting*.

### 4.5 Implementasi *Forward Chaining*

Setelah dilakukan penggambaran tentang implementasi Pohon dan Jaringan semantik, selanjutnya akan menggambarkan implementasi secara keseluruhan tentang alur *path* dari implementasi *forward chaining* antara *troubleshooting* dan *cases*. Implementasi dari penggambaran lengkap tentang metode *forward chaining* yang digunakan dalam alur *path* sistem pakar *troubleshooting software* berserta kondisi yang berada dalam *knowledge base* pakar *software*, bagaimana kasus dan solusi penyelesaian masalah tercipta dan saling terhubung.

Alasan yang kuat menggunakan mesin inferensi *forward chaining* yaitu karena tujuan utama-nya *data driven* berupa kasus (*cases*) yang telah banyak terkumpul menghasilkan sebuah *output decision* berupa solusi penyelesaian masalah (*troubleshooting*) dalam hal *software*. Semua pengetahuan tentang pakar *software* tersimpan di dalam *database* sebagai pengetahuan yang akan terus di *update* secara berkala demi tercapainya *knowledge base* yang bermanfaat dalam menyelesaikan permasalahan *software operating system windows*. Berikut ini penggambaran lengkap tentang *map knowledge base* pakar *software* dari lima *category* yang telah tersimpan ke dalam *database* berserta relasi yang telah terhubung antara *troubleshooting* dan *cases*.



Gambar 3 Implementasi Forward Chaining

#### 4.6 Implementasi

Pada tahap implementasi akan dilakukan tahap dalam pembuatan aplikasi seperti desain *database* dan desain *interface*. DBMS yang digunakan dalam implementasi pembuatan sistem pakar ini yaitu MySQL, untuk desain *user interface* menggunakan *Balsamiq Mockups*.

#### 4.7 Evaluasi User

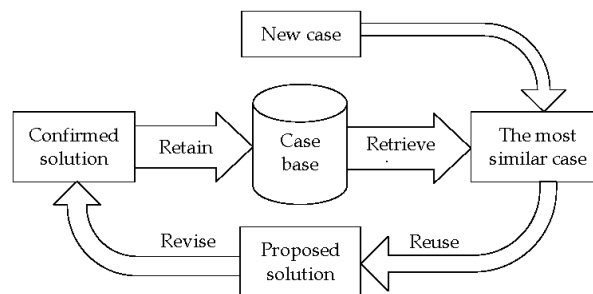
Hasil penelitian dari desain dan implementasi pakar *software* akan dilakukan pengujian apakah sistem aplikasi yang di rancang dapat diterima dengan baik atau tidak melalui evaluasi kepada *users* yaitu pengguna aplikasi pakar *software*.

Berdasarkan hasil penilaian *user interface* dan *user experience* pada tabel di atas yang terdiri dari 4 indikator pertanyaan sebagai penilaian, di peroleh 28.48% responden memberikan penilaian kategori sangat baik, 44.24% responden memberikan penilaian ketegori baik, 23.03% responden memberikan penilaian ketegori cukup dan 4.24% responden memberikan nilai kurang terhadap aplikasi yang dikembangkan.

Berdasarkan hasil penilaian kelengkapan inforamasi dan penyelesaian masalah pada tabel di atas yang terdiri dari 3 indikator pertanyaan sebagai penilaian, di peroleh 27.5% responden memberikan penilaian kategori sangat baik, 40% responden memberikan penilaian ketegori baik, 25.83% responden memberikan penilaian ketegori cukup, 5.83% responden memberikan nilai kurang dan 0.83% responden memberikan nilai sangat kurang terhadap aplikasi yang dikembangkan.

#### 4.8 Case-Based Reasoning

Penalaran yang digunakan dalam pakar *software* ini menggunakan penalaran berbasis kasus yang melakukan proses pemanfaatan kasus yang sudah ada sebelumnya untuk meningkatkan perngetahuan *knowledge base* yang tersimpan di dalam *database*, berikut penggambaran dari *case-based reasoning*.



Gambar 4 Case-Based Pakar Software

#### 4.9 Maintenance

Pada tahap *maintenance* akan dilakukan kegiatan penting dalam siklus hidup ESDLC yang membuat sistem pakar yang di rancang menjadi lebih *up-to-date*, tahap *maintentance* dilakukan oleh *administratror* maupun *knowledge engineer*, penambahan pengetahuan sangat diperlukan dalam pengembangan sistem pakar ini.

## 5 Kesimpulan dan Saran

### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dari itu didapatkan sebuah hasil yang dapat disimpulkan yaitu sebagai berikut :

- Penggunaan metode *forward chaining* dapat mewakili representasi dalam penyajian alur informasi untuk merelasikan antara *troubleshooting* dan *cases*, *forward chaining* memeberikan hasil yang cukup baik karena hal yang dibutuhkan dalam sistem pakar ini yaitu *data driven*.
- Metodologi ESDLC membantu menyusun kerangka kerja yang terencana dalam perancangan sistem pakar *troubleshooting software* dan menggunakan metode pendukung akuisisi pengetahuan.
- Penerapan sistem pakar berbasis *web* yang telah diimplementasikan dengan memaksimalkan penyajian informasi *problem software* berupa konsultasi sesuai area permasalahan, terdapat *management control panel* dengan dua *level* yang dapat menambahkan permasalahan baru yang menghasilkan suatu penyelesaian masalah dengan solusi terbaik.

### 5.2 Saran

Berdasarkan penelitian yang telah dilakukan, terdapat berbagai saran untuk penelitian selanjutnya agar sistem yang di bangun bisa lebih baik yaitu sebagai berikut :

1. Penggunaan metode *forward chaining* dapat mewakili representasi dalam penyajian alur informasi untuk merelasikan antara *troubleshooting* dan *cases*, *forward chaining* memeberikan hasil yang cukup baik karena hal yang dibutuhkan dalam sistem pakar ini yaitu *data driven*.
2. Metodologi ESDLC membantu menyusun kerangka kerja yang terencana dalam perancangan sistem pakar *troubleshooting software* dan menggunakan metode pendukung akuisisi pengetahuan.
3. Penerapan sistem pakar berbasis *web* yang telah diimplementasikan dengan

## 6 Referensi

- [1] Kusumadewi, S. (2003). Artificial Intelligence (Teknik dan Aplikasinya). Yogyakarta : Graha Ilmu.

- [2] <http://php.net/>. 13 Januari 2015 19:41
- [3] Peranginagin, Kasiman. (2006). *Aplikasi Web dengan PHP dan MySQL*. Yogyakarta : Andi Offset.
- [4] Yakub. (2012). *Pengantar Sistem Informasi*. Yogyakarta : Graha Ilmu.
- [5] [www.w3.org](http://www.w3.org) .2015
- [6] [Support2.microsoft.com/fixit/id-id](http://support2.microsoft.com/fixit/id-id). 2015
- [7] Arhami, M. (2005). *Konsep Dasar Sistem Pakar*. Yogyakarta : Andi.
- [8] Kusrini. (2006). *Sistem Pakar, Teori dan Aplikasi*. Yogyakarta : Andi.
- [9] Apache Friends. <https://www.apachefriends.org/>. 13 Januari 2015 19:49.
- [10] Ayuliana. (2009). *Teknik Pengujian Perangkat Lunak*. Maret 2009.
- [11] Basuki, Awan P. (2014). *Proyek Membangun Website Berbasis PHP dengan Codeigniter*. Yogyakarta : Lokomedia.
- [12] Dharwiyanti, S. (2003). *Pengantar Unified Modeling Language (UML)*. Ilmu Komputer. 2003.
- [13] Ginting, Budi S. (2014). *Perancangan Sistem Pakar Diagnosa Kerusakan Blackberry Smartphone Berbasis WEB*. Informasi dan Teknologi Ilmiah (INTI). Vol : III, No : 2. Juni 2014.
- [14] Handayani, L. (2014). *Sistem Pakar Diagnosa Kerusakan Notebook Berbasis Wap Mobile Menggunakan Metode Breadth First Search*. Pelita Informatika Budi Darma. Vol : VI, No : 1. Maret 2014.
- [15] Jamal, A. (2014). *Rancang Bangun Sistem Pakar Diagnosa Kerusakan Notebook pada Widodo Computer Ngadirojo Kabupaten Pacitan*. Sentra Penelitian Engineering dan Edukasi. Vol : XI, No : 1. Februari 2014.
- [16] Kadir, A. (2014). *Pengenalan Sistem Informasi (Edisi Revisi)*. Yogyakarta : Andi.
- [17] Mardison. (2012). *Penerapan Sistem Pakar untuk Mendeteksi Kerusakan Handphone (Studi Kasus Handphone Nokia 2115)*. Jurnal Teknologi Informasi & Pendidikan. Vol : V, No : 2. September 2012.
- [18] Peranginagin, Kasiman. (2006). *Aplikasi Web dengan PHP dan MySQL*. Yogyakarta : Andi Offset.
- [19] PhpMyAdmin. <http://www.phpmyadmin.net/>. 13 Januari 2015 19:46.
- [20] Ridwan. (2010). *Metode dan Teknik Menyusun Tesis*. Bandung : ALFABETA.
- [21] Rizky, S. (2011). *Konsep Dasar Rekayasa Perangkat Lunak (Software Reengineering)*. Jakarta : Prestasi Pustaka.
- [22] Rosa. (2014). *Rekyasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung : Informatika.
- [23] Simarmata, J. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta : Andi.
- [24] Solikhun. (2013). *Sistem Pakar Diagnosa Kerusakan Transmisi RBS dengan Menggunakan Metode Forward Chaining*. Seminar Nasional Teknologi Informasi dan Komunikasi (SNASTIKOM). 2013.
- [25] StatCounter. <http://gs.statcounter.com/>. 20 Februari 2015 20:15.

- [26] Sugiyono. (2012). Metode Penelitian Kuantitatif Kualitatif dan R&D. Alfabeta : Bandung.
- [27] Sutarman. (2009). Pengantar Teknologi Informasi. Jakarta : Bumi Aksara.
- [28] Suyatno. (2011). Artificial Intelligence. Bandung : Informatika.
- [29] Thomas J, McCabe. (1996). *Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric*. NIST Special Publication 500-235. 1996.
- [30] Unified Modeling Language (UML). <http://www.uml.org/>.23 Januari 2015 14:43.
- [31] Yakub. (2012). Pengantar Sistem Informasi. Yogyakarta : Graha Ilmu.