

Rancang Bangun Aplikasi Sistem Pakar Berbasis Android Menggunakan Metode Forward Chaining Untuk Mendiagnosis Kerusakan Pada Hardware Laptop

Budi Wijaya
Program Studi Teknik Informatika
Universitas Ciputra
arrowbudi9914@gmail.com

Rinabi Tanamal
Program Studi Teknik Informatika
Universitas Ciputra
r.tanamal@ciputra.ac.id

Abstrak – Kebutuhan dan penggunaan *laptop* yang banyak sering diikuti oleh banyaknya permasalahan yang sering terjadi pada *laptop* yang digunakan. Beberapa permasalahan yang sering terjadi, seperti layar *laptop* tampil *blue screen*, baterai *laptop* yang cepat habis, *keyboard* tidak berfungsi dengan normal, ataupun *laptop* yang tiba-tiba mati dengan sendirinya. Bila permasalahan tersebut timbul, biasanya pengguna *laptop* yang masih awam tidak mengerti apa yang harus mereka lakukan. Biaya perbaikan *laptop* yang harus dibayar termasuk mahal. Tidak hanya itu, jika permasalahan tersebut dibawa ke tempat servis *laptop*, ada beberapa teknisi *laptop* yang salah mendiagnosis kerusakannya karena kurang pemahaman atau pengalaman mengenai masalahnya. Maka dari itu, untuk mengatasi beberapa permasalahan yang sering terjadi di atas, dibuatlah aplikasi sistem pakar berbasis Android menggunakan metode *forward chaining* yang dapat mendiagnosis kerusakan *hardware laptop* secara tepat. *Expert system shell* yang digunakan adalah McGoO yang selanjutnya menggunakan Thunkable sebagai pembuat aplikasi sistem pakar. Dengan sistem ini, diharapkan *user* dapat lebih cepat, mudah, dan efisien dalam memperbaiki *laptop* dan mengambil keputusan secara tepat. Kesimpulan yang didapat menunjukkan bahwa pembentukan pohon keputusan untuk diagnosis kerusakan *hardware laptop* dapat diimplementasikan ke dalam aplikasi. Berdasarkan hasil pengujian, aplikasi sistem pakar yang telah terbentuk dapat diakses melalui *smartphone* berbasis Android sehingga dapat membantu pengguna *laptop* untuk memahami dan mengatasi masalah dan gangguan pada bagian *hardware laptop*.

Kata Kunci: Android, Forward Chaining, Kerusakan Laptop, McGoO, Sistem Pakar.

I. PENDAHULUAN

A. Latar Belakang

Seiring dengan perubahan zaman, perkembangan teknologi informasi memberikan pengaruh yang cukup besar

pada kehidupan umat manusia. Teknologi tersebut dikembangkan untuk mempermudah serta mempercepat berbagai macam kegiatan yang biasanya dilakukan oleh manusia. *Laptop* merupakan salah satu perkembangan teknologi informasi yang berasal dari perubahan konsep komputer *desktop*. Berdasarkan data yang diolah oleh IDC Indonesia, *laptop notebook* mendominasi pasar komputer *desktop* di Indonesia pada tahun 2017. *Market share laptop notebook* berkisar antara 70-75% tiap kuartal [1]. Dengan data tersebut, dapat disimpulkan bahwa *laptop* banyak diminati karena memiliki *mobilitas* yang tinggi dan harganya yang cukup terjangkau di berbagai kalangan masyarakat. Kebutuhan dan penggunaan *laptop* yang banyak tersebut sering juga diikuti oleh banyaknya permasalahan yang sering terjadi pada *laptop* yang digunakan. Beberapa permasalahan yang sering terjadi, seperti layar *laptop* tampil *blue screen*, baterai *laptop* yang cepat habis, *keyboard* tidak berfungsi dengan normal, ataupun *laptop* yang tiba-tiba mati dengan sendirinya. Bila permasalahan tersebut timbul, biasanya pengguna *laptop* yang masih awam tidak mengerti apa yang harus mereka lakukan. Biaya perbaikan *laptop* yang harus dibayar termasuk mahal. Tidak hanya itu, jika permasalahan tersebut dibawa ke tempat servis *laptop*, ada beberapa teknisi *laptop* yang salah mendiagnosis kerusakannya karena kurang pemahaman atau pengalaman mengenai masalahnya. Maka dari itu, untuk mengatasi beberapa permasalahan yang sering terjadi di atas, dibuatlah aplikasi sistem pakar berbasis Android yang dapat mendiagnosis kerusakan *laptop* secara tepat. Dengan sistem ini, diharapkan *user* dapat lebih cepat, mudah, dan efisien dalam memperbaiki *laptop* dan mengambil keputusan secara tepat.

B. Rumusan Masalah

Berdasarkan latar belakang yang disampaikan di atas, maka rumusan masalah yang akan dibahas pada penelitian ini adalah “Bagaimana merancang bangun aplikasi sistem pakar berbasis Android menggunakan metode *forward chaining* untuk mendiagnosis kerusakan pada *hardware laptop* yang digunakan untuk membantu mengambil keputusan bagi pengguna maupun teknisi *laptop*?”

C. Tujuan Penulisan

Tujuan dari penelitian “Rancang Bangun Aplikasi Sistem Pakar Berbasis Android Menggunakan Metode *Forward Chaining* untuk Mendiagnosis Kerusakan pada *Hardware Laptop*” adalah untuk menghasilkan sebuah aplikasi sistem pakar berbasis Android yang dapat membantu pengguna awam maupun teknisi *laptop* mengambil keputusan pada kerusakan *hardware laptop* tersebut.

D. Ruang Lingkup

Untuk mengarahkan penyusunan dan penelitian ini, peneliti membatasi ruang lingkup penelitian sebagai berikut:

1. *Laptop* yang diteliti merupakan *laptop* tahun produksi 2010-2018, yang menggunakan sistem operasi versi Windows 7-Windows 10.
2. *Laptop* yang diteliti tidak termasuk *laptop* produk dari brand Apple.
3. Sistem pakar yang dirancang khusus untuk mendiagnosis ataupun mendeteksi kerusakan pada bagian *hardware laptop* saja.
4. Penyusunan fakta, aturan, dan ilmu pengetahuan berasal dari observasi dan hasil wawancara terhadap teknisi yang menangani *laptop* sebagai pakar. Observasi sendiri dilakukan dengan cara melihat secara detail proses atau kegiatan yang dipraktikkan oleh teknisi untuk melakukan diagnosis. Hasil wawancara didapatkan melalui tanya jawab mengenai proses diagnosis kerusakan *laptop* oleh teknisi.
5. Pembentukan fakta dan aturan sistem pakar dibuat dengan *software expert system shell* McGoo yang selanjutnya diimplementasikan ke dalam aplikasi Android.
6. Metode penalaran yang dipakai untuk penelitian ini menggunakan metode *forward chaining*.

II. LANDASAN TEORI

A. Sistem Pakar

Sistem Pakar atau yang dikenal dengan istilah *Expert System* merupakan sistem komputer yang fungsinya mengikuti semua aspek kemampuan pengambilan keputusan yang dimiliki oleh seorang pakar. Dengan kata lain, sistem ini berjalan layaknya seorang pakar khusus yang mampu memecahkan masalah rumit [2].

B. Metode *Forward Chaining*

Metode Runut Maju atau istilahnya *Forward Chaining* adalah metode dimana pencocokan pernyataan atau faktanya dimulai dari fakta atau *IF* terlebih dahulu, lalu dilanjutkan dengan *THEN* yang berisi dengan hipotesisnya [2].

C. McGoo

Program ini merupakan sebuah aplikasi untuk membuat sistem pakar berbasis *web* secara *online* [3].

D. Android

Android merupakan sistem operasi dengan Linux sebagai basisnya dan dapat digunakan untuk ponsel. Android menyediakan *platform* yang terbuka bagi para *developer* atau istilahnya *open source*, sehingga memungkinkan banyaknya aplikasi yang tercipta sendiri, kemudian dapat dijalankan pada *smartphone* [4].

E. Pohon Keputusan

Pohon Keputusan atau yang sering dikenal dengan istilah *Decision Tree* adalah suatu teknik dalam analisis pemecahan masalah dengan cara pemetaan mengenai alternatif-alternatif keputusan yang berbentuk seperti pohon. Pemetaan tersebut juga menunjukkan berbagai faktor probabilitas yang mempengaruhinya hingga pada akhirnya terdapat estimasi kesimpulan bila memilih salah satu alternatif yang tersedia. Metode *decision tree* ini mengubah fakta yang sangat besar menjadi pohon keputusan yang berisi aturan dan berguna dalam pengambilan keputusan [5].

III. ANALISIS DAN DESAIN

A. Pakar

Dalam penelitian ini, peneliti mengumpulkan informasi dan pengetahuan dari 5 teknisi *laptop* yang disebut sebagai pakar. Semua pakar yang dipilih oleh peneliti merupakan teknisi yang telah berpengalaman dalam bidang jasa *service laptop* lebih dari 3 tahun dan juga ada yang sejak tahun 2000 sampai sekarang sudah memahami bidang ini. Mereka telah menekuninya dan mengatasi berbagai jenis kerusakan *laptop* terutama bagian *hardware laptop*. Para teknisi *laptop* tersebut bekerja di berbagai toko yang berbeda dalam Hi-Tech Mall Surabaya, dimana tempat ini menjadi pusat perdagangan *laptop* dan komputer terbesar di Indonesia Timur. Hal tersebut dapat membuktikan bahwa pemilihan kelima pakar ini sangat tepat karena mereka telah mendalami berbagai macam masalah kerusakan *hardware laptop*.

B. Observasi

Dalam tahap ini, peneliti melakukan pengamatan terhadap pakar dalam proses kerja pakar tersebut untuk menangani *laptop*. Kegiatan observasi ini dilakukan langsung di tempat kerja teknisi *laptop*, tepatnya di Hi-Tech Mall Surabaya. Observasi dilakukan dengan mengamati dan memahami bagaimana cara pakar dalam menangani dan mengatasi kerusakan *hardware laptop* dari para *customer* dari gejala awal penyebab kerusakan sampai dengan mencari solusi yang tepat untuk memperbaiki *laptop*nya. Tujuan dari tahap observasi, yaitu peneliti dapat mengetahui bagaimana langkah-langkah yang tepat untuk mendiagnosis kerusakan *hardware laptop* dari awal hingga kembali berfungsi dengan normal. Hasil dari observasi ini selanjutnya akan membantu peneliti menyusun berbagai pertanyaan lebih lanjut guna mendapatkan pengetahuan yang lebih lengkap, dimana hal tersebut dibutuhkan untuk perancangan aturan-aturan (*rule*) dan pohon keputusan (*decision tree*).

C. Wawancara

Tahap ini merupakan tahap untuk mengumpulkan informasi dan pengetahuan tambahan dari beberapa teknisi, yang disebut dengan pakar. Cara pengumpulannya, yaitu dengan wawancara langsung untuk mendapatkan informasi lebih lengkap mengenai gejala-gejala apa yang menyebabkan kerusakan hingga kesimpulan bagian *hardware laptop* mana yang rusak dan cara mengatasinya. Wawancara juga digunakan untuk melengkapi pengetahuan yang ada bila ada kekurangan informasi dalam kegiatan observasi sebelumnya. Semua informasi yang didapatkan akan disusun untuk membuat pohon keputusan (*decision tree*), dan selanjutnya aplikasi berbasis Android mengadopsi hasil tersebut.

D. Analisis Masalah

Kebutuhan dan penggunaan *laptop* yang banyak sering diikuti oleh banyaknya permasalahan yang sering terjadi pada *laptop* yang digunakan. Bila permasalahan tersebut timbul, biasanya pengguna *laptop* yang masih awam tidak mengerti apa yang harus mereka lakukan. Biaya perbaikan *laptop* yang harus dibayar termasuk mahal. Tidak hanya itu, jika permasalahan tersebut dibawa ke tempat *service laptop*, ada beberapa teknisi *laptop* yang salah mendiagnosis kerusakannya karena kurang pemahaman atau pengalaman mengenai masalahnya.

E. Penyelesaian Masalah

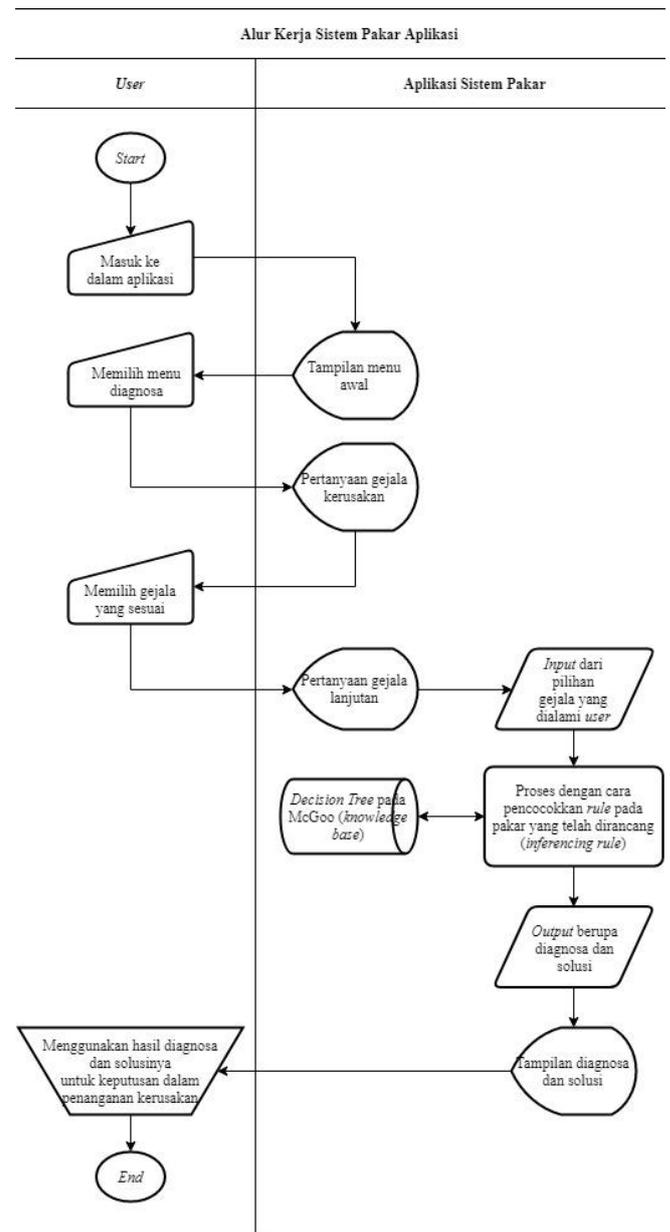
Sistem pakar berbasis Android dapat mendiagnosis kerusakan *hardware laptop* secara tepat dan dapat diakses dengan mudah melalui *smartphone* secara langsung. Aplikasi sistem pakar ini dapat membantu para pengguna *laptop* lebih cepat mengetahui permasalahan atau gejala yang dialami, kerusakan apa yang terjadi pada *laptop*, dan solusi penanganan yang harus dilakukan dengan efisien. Pengguna *laptop* juga akan terhindar dari oknum tempat *service* yang tidak bertanggung jawab. Tidak hanya itu, teknisi *laptop* yang kurang pengalaman akan terbantu dalam mendiagnosis kerusakan *laptop* yang diperbaikinya supaya tidak terjadi kesalahan diagnosis.

F. Desain Sistem

Setelah mendapatkan semua ilmu pengetahuan yang ada melalui observasi dan wawancara dengan pakar, tahap berikutnya adalah membuat Pohon Keputusan atau *Decision Tree* dengan menggunakan metode Runut Maju atau yang istilahnya *Forward Chaining*. Metode tersebut bekerja dengan cara dimana pencocokan pernyataan atau faktanya dimulai dari fakta atau *IF* terlebih dahulu, lalu dilanjutkan dengan *THEN* yang berisi hipotesisnya. Lalu, Pohon Keputusan menerjemahkannya dengan cara memetakan semua fakta pengetahuan yang didapatkan menjadi sebuah alternatif-alternatif sampai pada menemukan kesimpulan dari permasalahan yang diberikan dalam *software* McGo. Selanjutnya, dibentuklah sistem aplikasi berbasis Android dengan mengadopsi hasilnya tersebut.

G. Alur Kerja Sistem Pakar Aplikasi

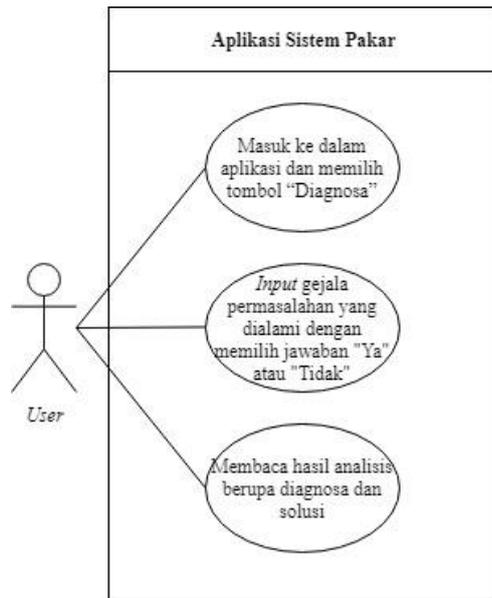
Cara kerja dari aplikasi sistem pakar yang dibuat dapat dilihat pada Gambar 1.



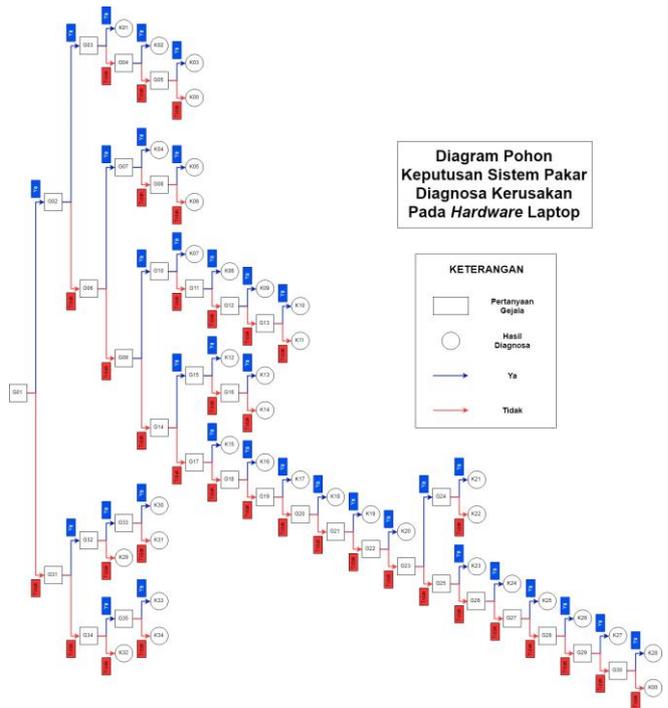
Gambar 1. Alur Kerja Sistem Pakar Aplikasi.

H. Use Case Diagram Aplikasi

Use Case merupakan sebuah pemodelan yang menjelaskan siapa saja yang berhak menggunakan fungsi-fungsi di dalam sistem dan apa saja yang dapat diperbuat. Dapat dilihat pada Gambar 2.



Gambar 2. Use Case Diagram.



Gambar 3. Pohon Keputusan Sistem Pakar.

I. Activity Diagram Aplikasi

Activity Diagram akan menjabarkan mengenai aktivitas yang dilakukan oleh user aplikasi dan sistem dari aplikasi itu sendiri. Aktivitas tersebut dimulai dari user membuka aplikasi sistem pakar pada *smartphone*. Di “Halaman Menu Utama”, user memilih menu “Diagnosis” untuk memulai diagnosis *laptop* dan nantinya sistem akan menampilkan halaman pertanyaan secara beruntun. User menjawab pertanyaan dengan memilih “Ya” atau “Tidak” sesuai keadaan yang sebenarnya dan nantinya sistem akan menampilkan hasil diagnosis berupa konklusi dan solusi.

J. Sequence Diagram Aplikasi

Sequence Diagram akan menjabarkan mengenai interaksi antar objek dengan user yang disebut dengan aktor. User masuk ke dalam aplikasi dan selanjutnya sistem menampilkan “Halaman Menu Utama”. Ketika user memilih menu “Diagnosis”, sistem akan menampilkan “Halaman Menu Pertanyaan Diagnosis” dan berisi pertanyaan secara beruntun. Tiap pertanyaan memiliki dua pilihan jawaban “Ya” dan “Tidak” yang harus dipilih salah satunya oleh user dan sistem akan menampilkan “Halaman Hasil Diagnosis” berisi konklusi dan solusi.

IV. IMPLEMENTASI

A. Pembentukan Decision Tree

Semua ilmu pengetahuan yang didapat hasil dari observasi dan wawancara dengan pakar tersebut dikumpulkan untuk membuat pohon keputusan (*decision tree*). Pohon Keputusan ini nantinya diterapkan dalam *software* McGoo. Dapat dilihat pada Gambar 3.

Dari diagram pohon keputusan di atas dapat dilihat alur pertanyaan dan hasil analisisnya. Pertanyaan yang tersedia hanya memiliki dua pilihan jawaban “Ya” dan “Tidak” untuk mendapatkan analisis. Untuk pilihan jawaban “Ya” berupa garis panah berwarna biru dan untuk pilihan jawaban “Tidak” berupa garis panah berwarna merah. Semua pertanyaan berbentuk persegi dan berkode “G”. Keterangan dari tiap kode pertanyaan dapat dilihat pada Tabel 1.

Tabel 1. Keterangan Kode Pertanyaan.

Kode Pertanyaan	Keterangan
G01	Apakah <i>laptop</i> dapat dihidupkan?
G02	Apakah <i>laptop</i> tiba-tiba mati saat mengoperasikan <i>software</i> atau program?
G03	Lakukan pemeriksaan voltase pada adaptor <i>charger</i> dengan multimeter, apakah <i>display</i> menunjukkan hasil yang berubah naik turun?
G04	Apakah <i>fan</i> atau kipas <i>processor</i> tidak terdengar suara putaran atau mati?
G05	Apakah baterai <i>drop</i> atau cepat habis dalam pemakaian <i>laptop</i> secara normal?
G06	Apakah layar LCD muncul <i>bluescreen</i> ?
G07	Apakah <i>laptop</i> sudah lama dipakai sekitar lebih dari 3 tahun?
G08	Apakah pernah menambah RAM baru untuk menambah kecepatan RAM?
G09	Apakah <i>laptop</i> mengalami <i>hang</i> ?

Kode Pertanyaan	Keterangan
G10	Apakah kondisi <i>body laptop</i> panas berlebih atau <i>overheating</i> ?
G11	Apakah sering mengoperasikan <i>laptop</i> untuk aplikasi <i>game</i> dan atau grafis yang berat?
G12	Apakah <i>laptop</i> sudah lama dipakai sekitar lebih dari 3 tahun?
G13	Apakah pernah menambah RAM yang baru untuk menambah kecepatan RAM?
G14	Apakah <i>laptop</i> tiba-tiba <i>restart</i> sendiri?
G15	Lakukan pemeriksaan voltase pada adaptor <i>charger</i> dengan multimeter, apakah <i>display</i> menunjukkan hasil yang berubah naik turun?
G16	Lakukan pemeriksaan RAM dan fisik RAM (IC, PIN, dan jalur PCB), apakah ada yang kotor atau rusak?
G17	Apakah ada bunyi “beep” yang panjang dan atau bisa berhenti ketika menekan salah satu tombol <i>keyboard</i> ?
G18	Apakah <i>touchpad-mouse</i> tidak dapat menggerakkan kursor secara normal?
G19	Apakah <i>sound</i> atau <i>speaker</i> tidak bunyi?
G20	Apakah baterai <i>drop</i> atau cepat habis dalam pemakaian <i>laptop</i> secara normal?
G21	Apakah ada tampilan garis-garis horizontal dan atau vertikal pada layar LCD?
G22	Apakah kadang tampil gambar dan kadang tidak ketika merubah posisi LCD ke depan dan ke belakang?
G23	Apakah tidak ada tampilan pada layar LCD meskipun mesin <i>laptop</i> hidup?
G24	Apakah ada tampilan gambar, jika <i>laptop</i> dihubungkan ke LCD Monitor atau Proyektor melalui <i>output</i> VGA Eksternal?
G25	Apakah LCD dalam kondisi pecah?
G26	Apakah kamera tidak dapat menampilkan gambar atau merekam video?
G27	Apakah <i>laptop</i> tidak dapat terkoneksi atau menjangkau semua jaringan internet, meskipun posisi <i>laptop</i> tidak jauh dari sumbernya?
G28	Apakah pembacaan data CD atau DVD tersendat-sendat atau tidak dapat terbaca?
G29	Apakah pembacaan data dari USB yang terhubung terputus-putus atau tidak dapat terbaca?
G30	Apakah tanggal dan waktu berubah saat <i>laptop</i> dimatikan?
G31	Apakah adaptor <i>charger</i> memiliki indikator lampu?
G32	Apakah indikator lampu pada adaptor <i>charger</i> menyala?

Kode Pertanyaan	Keterangan
G33	Apakah jarum konektor pada <i>port</i> konektor <i>power</i> putus atau patah?
G34	Lakukan pemeriksaan voltase pada adaptor <i>charger</i> dengan multimeter, apakah <i>display</i> menunjukkan angka sekitar 19 volt?
G35	Apakah jarum konektor pada <i>port</i> konektor <i>power</i> putus atau patah?

Semua hasil analisis berbentuk lingkaran dan berkode “K”. Keterangan dari tiap kode hasil analisis dapat dilihat pada Tabel 2.

Tabel 2. Keterangan Kode Hasil Analisis.

Kode Hasil Analisis	Keterangan
K01	Kerusakan pada adaptor <i>charger laptop</i> yang tegangan listrik <i>output</i> -nya tidak stabil.
K02	Kipas <i>processor</i> yang tidak berputar akibat banyaknya kotoran dan debu yang menghambat.
K03	Baterai rusak atau <i>drop</i> , beberapa sel tertentu dalam baterai sudah tidak berfungsi secara optimal.
K04	RAM yang sudah lemah karena pemakaian <i>laptop</i> yang telah berumur.
K05	Kesalahan pembacaan data pada RAM yang baru dengan <i>motherboard</i> karena ketidaksesuaian spesifikasi RAM yang dibutuhkan.
K06	Kerusakan pada <i>harddisk</i> yang mengalami <i>bad sector</i> lunak.
K07	Lubang ventilasi pembuangan dari kipas <i>processor</i> tertutup sehingga udara panas tidak bisa keluar yang mengakibatkan peningkatan panas berlebih di mesin <i>laptop</i> .
K08	Kondisi <i>chipset motherboard</i> , <i>chipset</i> VGA, ataupun <i>processor</i> yang terlalu panas.
K09	RAM yang sudah lemah karena pemakaian <i>laptop</i> yang telah berumur.
K10	Kesalahan pembacaan data pada RAM yang baru dengan <i>motherboard</i> karena ketidaksesuaian spesifikasi RAM yang dibutuhkan.
K11	Kerusakan pada <i>harddisk</i> yang mengalami <i>bad sector</i> lunak.
K12	Kerusakan pada adaptor <i>charger laptop</i> yang tegangan listrik <i>output</i> -nya tidak stabil.
K13	Bagian fisik RAM seperti bagian IC, PIN, atau jalur PCB yang kotor atau mengalami kerusakan.

Kode Hasil Analisis	Keterangan
K14	Kerusakan pada <i>harddisk</i> yang mengalami <i>bad sector</i> lunak dan atau fisik.
K15	<i>Keyboard</i> mengalami gangguan akibat kotoran dan berdebu.
K16	<i>Touchpad-mouse</i> mengalami kerusakan.
K17	<i>Speaker</i> internal atau <i>IC sound laptop</i> dalam kondisi rusak atau mati.
K18	Baterai rusak atau <i>nge-drop</i> , akibat dari seringnya tersambung adaptor <i>charger</i> sehingga pengisian baterai menjadi berlebihan meskipun baterai sudah penuh dan atau sambil mengoperasikan <i>laptop</i> .
K19	Kerusakan pada bagian komponen <i>inverter</i> .
K20	Gangguan pada kendornya konektor <i>VGA</i> , konektor kabel <i>inverter</i> , dan atau kabel layar <i>LCD</i> .
K21	Gangguan pada kendornya konektor <i>VGA</i> , konektor kabel <i>inverter</i> , dan atau kabel layar <i>LCD</i> .
K22	Kerusakan pada <i>chipset</i> <i>VGA</i> .
K23	Kerusakan pada <i>LCD</i> yang pecah diakibatkan dari benturan dengan benda keras, jatuh, ataupun terkena tekanan yang kuat.
K24	Kerusakan pada kamera <i>laptop</i> .
K25	Kerusakan pada <i>wireless</i> atau biasanya disebut <i>wifi</i> .
K26	Kerusakan pada <i>sensor optic</i> yang terdapat pada <i>optical drive</i> .
K27	Kerusakan pada <i>port</i> <i>USB</i> .
K28	Kerusakan pada <i>CMOS</i> baterai yang mati.
K29	Kerusakan pada adaptor <i>charger laptop</i> yang mati.
K30	Kerusakan pada jarum konektor yang terletak di dalam <i>port</i> konektor <i>power laptop</i> dimana kemungkinan jarum tersebut mengalami putus atau patah.
K31	Kerusakan pada <i>power supply</i> atau <i>IC Power</i> .
K32	Kerusakan pada adaptor <i>charger laptop</i> yang mati.
K33	Kerusakan pada jarum konektor yang terletak di dalam <i>port</i> konektor <i>power laptop</i> dimana kemungkinan jarum tersebut mengalami putus atau patah.
K34	Kerusakan pada <i>power supply</i> atau <i>IC Power</i> .
K00	Membawa ke tempat <i>service laptop</i> atau teknisi <i>laptop</i> untuk diagnosis lebih lanjut.

B. Implementasi *Decision Tree* ke Dalam McGoo

Tahap ini menjelaskan bagaimana pohon keputusan (*decision tree*) yang telah dibentuk sebelumnya diterapkan dalam *software* McGoo. Aplikasi ini digunakan untuk membuat sistem pakar berbasis *web* secara *online* yang sangat sederhana dan mudah dimengerti. Metode yang digunakan adalah *forward chaining* dimana pencocokan pernyataan atau faktanya dimulai dari gejala permasalahan atau *IF* terlebih dahulu, lalu dilanjutkan dengan *THEN* yang berisi analisisnya.

1. Code

Dalam McGoo ini berisikan *rule* atau aturan berupa pertanyaan yang tiap pertanyaannya memiliki dua *value*, yaitu “Ya” dan “Tidak”, serta terdapat konklusi berupa hasil analisis permasalahan yang terjadi. Contoh sebagian *code* dalam McGoo dapat dilihat pada Gambar 4.



Gambar 4. Rule dan Conclusion.

2. Tampilan McGoo

Tampilan ini berupa *user interface* berisikan pertanyaan yang harus dijawab oleh pengguna untuk mendapatkan hasil analisis yang diinginkan. Terdapat dua pilihan jawaban “Ya” dan “Tidak” yang dapat dipilih sesuai kondisi permasalahan yang dihadapi. Tampilan awal dari McGoo dapat dilihat pada gambar 5 yang berisikan pertanyaan awal keadaan *laptop* yang sebenarnya. Pengguna dapat memilih antara “Ya” dan “Tidak” untuk melanjutkan pertanyaan lainnya.



Gambar 5. Tampilan Awal McGoo.

Tampilan pertanyaan lanjutan untuk memastikan permasalahan apa yang sebenarnya terjadi pada *laptop* pengguna supaya mendapatkan analisis yang tepat untuk mengatasinya. Dapat dilihat pada Gambar 6.



Gambar 6. Tampilan Pertanyaan Lanjutan.

Tampilan hasil analisis berupa konklusi permasalahan kerusakan apa yang sebenarnya terjadi dan beberapa tips cara mengatasinya. Dapat dilihat pada Gambar 7.



Gambar 7. Tampilan Hasil Analisis.

C. Implementasi *Decision Tree* ke Dalam Aplikasi

Tahap ini menjelaskan bagaimana pohon keputusan (*decision tree*) yang telah dibentuk sebelumnya diterapkan dalam Thinkable. Thinkable digunakan untuk membuat aplikasi berbasis *web* secara *online* yang memiliki desain sangat sederhana dan mudah dimengerti. Di dalam Thinkable terdapat dua bagian utama, yaitu “*Design*” untuk mengatur desain tampilan dari aplikasi yang diinginkan dan “*Block*” untuk mengatur semua perintah dalam aplikasi yang dibentuk.

Aplikasi yang terbentuk dapat diakses melalui *smartphone* berbasis Android. Nama dari aplikasi ini adalah “PROVER” dimana merupakan singkatan dari “*Your Laptop Problem Resolver*”. Kegunaan dari aplikasi adalah untuk mendiagnosis kerusakan pada *hardware laptop* secara cepat dan tepat.

Dalam aplikasi yang dirancang ini memiliki beberapa bagian halaman utama beserta *code*-nya, sebagai berikut.

1. *Code* Halaman Depan



Gambar 8. *Code* Halaman Depan.

2. *Code* Halaman Menu Utama



Gambar 9. *Code* Halaman Menu Utama.

3. *Code* Halaman Cara Penggunaan



Gambar 10. *Code* Halaman Cara Penggunaan.

4. *Code* Halaman Tentang Kami



Gambar 11. *Code* Halaman Tentang Kami.

5. *Code* Halaman Menu Pertanyaan Diagnosis



Gambar 12. Sebagian *Code* Halaman Menu Pertanyaan Diagnosis.

6. Code Halaman Hasil Diagnosis



Gambar 13. Sebagian Code Halaman Hasil Diagnosis.

D. User Interface Aplikasi

1. Halaman Depan



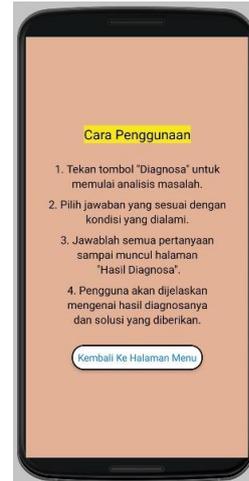
Gambar 14. User Interface Halaman Depan.

2. Halaman Menu Utama



Gambar 15. User Interface Halaman Menu Utama.

3. Halaman Cara Penggunaan



Gambar 16. User Interface Halaman Cara Penggunaan.

4. Halaman Tentang Kami



Gambar 17. User Interface Halaman Tentang Kami.

5. Halaman Menu Pertanyaan Diagnosis



Gambar 18. User Interface Halaman Menu Pertanyaan Diagnosis.

6. Halaman Hasil Diagnosis



Gambar 19. User Interface Halaman Hasil Diagnosis.

V. HASIL PENGUJIAN

A. Pengujian Akurasi

Dalam tahap ini, peneliti akan melakukan pengujian akurasi dari aplikasi yang telah dibuat. Pengujian dilakukan dengan cara membandingkan hasil analisis dari aplikasi dengan hasil analisis dari teknisi *laptop* yang disebut pakar. Tujuannya supaya mendapatkan tingkat akurasi ketepatan secara maksimal. Dapat dilihat pada Tabel 3.

B. Uji Coba Aplikasi Sistem Pakar Kepada *User Laptop*

Tahap ini menjelaskan bagaimana aplikasi sistem pakar yang telah dibuat akan diuji coba kepada pengguna *laptop*. Pengujian ini menggunakan responden sebanyak 20 pengguna *laptop*. Pengguna nantinya akan menggunakan aplikasi untuk menganalisis permasalahan pada *laptop* dengan menjawab pertanyaan yang disediakan dalam aplikasi secara runtut. Dengan menjawab semua pertanyaan yang ada, maka aplikasi memberikan hasil analisisnya dan solusi permasalahan. Kemudian peneliti akan memberikan kuesioner *offline* untuk diisi oleh pengguna *laptop* tersebut. Tujuan dari pengisian kuesioner itu adalah mendapatkan penilaian bagaimana tingkat kepuasan pengguna memakai aplikasi, serta kritik dan saran yang membangun untuk pengembangan aplikasi PROVER.

Tabel 3. Pengujian Aplikasi.

No.	Gejala yang Dialami	Hasil Analisis Aplikasi	Hasil Analisis Pakar	Sesuai atau Tidak Sesuai
1.	<i>Laptop</i> tiba-tiba mati sendiri; Kipas <i>processor</i> tidak terdengar suara putaran.	Kipas <i>processor</i> yang tidak berputar akibat banyaknya kotoran dan debu yang menghambat.	Kipas <i>processor</i> mati.	Sesuai
2.	<i>Laptop</i> tiba-tiba mati sendiri; Baterai cepat habis dalam pemakaian <i>laptop</i> secara normal.	Baterai rusak atau ngedrop.	Baterai rusak atau sudah tidak berfungsi.	Sesuai
3.	Layar tampil <i>bluescreen</i> ; Pemakaian <i>laptop</i> yang telah berumur.	RAM yang sudah lemah karena pemakaian <i>laptop</i> yang telah berumur.	Kondisi RAM yang lemah.	Sesuai
4.	Layar tampil <i>bluescreen</i> .	Kerusakan pada <i>harddisk</i> yang mengalami <i>bad sector</i> lunak.	<i>Hard disk</i> mengalami <i>bad sector</i> lunak.	Sesuai
5.	<i>Laptop</i> kondisi <i>hang</i> ; Kondisi <i>body laptop overheating</i> .	Lubang ventilasi pembuangan dari kipas <i>processor</i> tertutup sehingga udara panas tidak bisa keluar.	Lubang ventilasi pembuangan dari kipas <i>processor</i> tertutup.	Sesuai
6.	<i>Laptop</i> kondisi <i>hang</i> ; Sering mengoperasikan <i>laptop</i> untuk aplikasi <i>game</i> dan atau grafis yang besar dan berat.	Kondisi <i>chipset motherboard</i> , <i>chipset VGA</i> , ataupun <i>processor</i> yang terlalu panas.	Kondisi <i>chipset motherboard</i> , <i>chipset VGA</i> , ataupun <i>processor</i> yang terlalu panas.	Sesuai
7.	<i>Laptop</i> tiba-tiba <i>restart</i> sendiri; Hasil pemeriksaan RAM yang fisiknya ada kotoran atau rusak.	Bagian fisik RAM seperti bagian IC, PIN, atau jalur PCB yang kotor atau mengalami kerusakan.	Bagian fisik RAM kotor.	Sesuai
8.	Bunyi “beep” yang panjang; Beberapa tombol <i>keyboard</i> tidak berfungsi; Ada tombol yang menekan sendiri.	<i>Keyboard</i> mengalami gangguan akibat kotoran dan berdebu.	Kerusakan pada <i>keyboard</i> .	Sesuai
9.	Layar muncul garis – garis warna putih.	Kerusakan pada bagian komponen <i>inverter</i> .	Komponen <i>inverter</i> yang rusak.	Sesuai

No.	Gejala yang Dialami	Hasil Analisis Aplikasi	Hasil Analisis Pakar	Sesuai atau Tidak Sesuai
10.	Tidak ada tampilan apapun di layar LCD; Projektor ataupun monitor tidak dapat menampilkan gambar dari layar LCD utama.	Kerusakan pada <i>chipset</i> VGA.	<i>Chipset</i> VGA rusak.	Sesuai
11.	Indikator lampu adaptor tidak nyala.	Kerusakan pada adaptor <i>charger laptop</i> yang mati.	Adaptor <i>charger</i> tidak berfungsi atau mati.	Sesuai
12.	<i>Laptop</i> tidak dapat dihidupkan atau mati total.	Kerusakan pada <i>power supply</i> atau <i>IC Power</i> .	<i>IC Power</i> rusak.	Sesuai

C. Hasil Kuesioner

Kuesioner yang dibuat berisikan tanggapan kepuasan mengenai aplikasi yang digunakan oleh *user*. Nantinya *user* memberikan penilaian di tiap pertanyaan dalam kuesioner. Tiap pertanyaan memiliki 5 tingkat skor penilaian jawaban berdasarkan pengukuran Skala Likert [6]. Dapat dilihat pada Tabel 4.

Tabel 4. Skala Likert

Tingkat Skor	Keterangan	Nilai
1	Sangat Tidak Setuju	0% - 19,99%
2	Tidak Setuju	20% - 39,99%
3	Netral	40% - 59,99%
4	Setuju	60% - 79,99%
5	Sangat Setuju	80% - 100%

Setelah mengumpulkan semua penilaian dari 20 responden, peneliti menghitung total skor yang didapat dari tiap pertanyaan. Dapat dilihat pada Tabel 5.

Tabel 5. Hasil Persentase Skala Likert Kuesioner

No.	Pertanyaan	Nilai
1.	Aplikasi PROVER mudah digunakan.	85%
2.	Aplikasi PROVER mudah dimengerti dan dipahami.	87%
3.	Aplikasi PROVER memiliki tampilan yang menarik.	79%
4.	Aplikasi PROVER berguna dan membantu bagi saya.	91%
5.	Aplikasi PROVER memberikan informasi yang benar dan tepat.	86%

VI. HASIL PENGUJIAN

A. Kesimpulan

Beberapa kesimpulan yang didapat dari penelitian aplikasi sistem pakar yang telah dibuat dan diuji, sebagai berikut.

Penyusunan fakta, aturan, dan ilmu pengetahuan berasal dari studi literatur, observasi, dan hasil wawancara terhadap teknisi yang menangani *laptop* sebagai pakar. Tujuannya untuk membentuk pohon keputusan menggunakan metode

forward chaining yang kemudian diimplementasikan ke dalam *McGoo Software* sebagai *Expert System Shell* dan *Thinkable* sebagai pembuat aplikasi sistem pakar PROVER.

Aplikasi sistem pakar PROVER dapat diakses pada *smartphone* berbasis Android. Hasil pengujian tingkat akurasi ketepatan analisis dari aplikasi PROVER yang telah dibuat sudah sesuai 100% terhadap keadaan lapangan yang sebenarnya.

Hasil pengujian tingkat kepuasan pengguna memakai aplikasi PROVER berdasarkan pengukuran Skala Likert, sebagai berikut: Nilai tentang kemudahan untuk menggunakan aplikasi sebesar 85%. Nilai tentang kemudahan aplikasi untuk dimengerti dan dipahami sebesar 87%. Nilai tentang tampilan aplikasi sebesar 79%. Nilai tentang aplikasi berguna dan membantu pengguna *laptop* sebesar 91%. Nilai tentang aplikasi memberikan informasi yang benar dan tepat sebesar 86%.

B. Saran

Beberapa saran yang didapat dari penelitian aplikasi sistem pakar yang telah dibuat dan diuji untuk pengembangan yang lebih baik lagi, sebagai berikut.

1. Menambahkan fitur untuk mengisi gejala sendiri oleh *user* di luar pertanyaan gejala yang telah tersedia.
2. Menambahkan fitur khusus berupa kumpulan istilah penting bagian *hardware laptop*.
3. Tampilan dibuat lebih menarik.
4. Video, gambar, dan tulisan lebih diperbesar lagi.

UCAPAN TERIMA KASIH

Penulis ingin menyampaikan rasa terima kasih sebesar-besarnya kepada:

1. Tuhan Yang Maha Esa atas rahmat, berkat, dan kasih karunia-Nya sehingga penulis diberikan kelancaran dan pencerahan setiap langkah dalam penulisan.
2. Dosen pembimbing yang senantiasa meluangkan waktu dan pemikirannya di sela-sela kesibukan beliau untuk menuntun, membimbing, mengarahkan, memberi motivasi, serta saran.
3. Orang tua dan saudara kandung dari penulis yang selalu memberi semangat, dukungan, doa, dan motivasi.

REFERENSI

- [1] Perkasa, A. & Santhika, E. (2018). *Pasar Laptop Terus Mendominasi, Gerus Desktop dan Komersial*. Diakses dari CNN Indonesia: <https://www.cnnindonesia.com/teknologi/20180427093102-185294019/pasar-laptop-terus-mendominasi-gerus-desktop-dan-komersial> pada tanggal 28 April 2018.
- [2] Hayadi, B.H. & Rukun, K. (2016). *What is Expert System*. Yogyakarta: Deepublish Publisher.
- [3] McGoo Software. (n.d.). *About ES-Builder Web*. Diakses dari McGoo: https://www.mcgoo.com.au/html/es-builder_web.php
- [4] Jubilee Enterprise. (2015). *Mengenal Dasar-Dasar Pemrograman Android*. Jakarta: PT Elex Media Komputindo.
- [5] Nofriansyah, D. (2014). *Konsep Data Mining Vs Sistem Pendukung Keputusan*. Yogyakarta: Deepublish Publisher.
- [6] Tanamal, R. (2017). Analisis Faktor yang Paling Berpengaruh Pada Keinginan Menggunakan Aplikasi Grab di Kota Surabaya. *Jurnal Terapan Teknologi Informasi, Universitas Kristen Duta Wacana*, Vol. 1, No. 2, pp. 119-128.