

SISTEM PAKAR DIAGNOSA KERUSAKAN PADA KOMPUTER DENGAN METODE *FORWARD CHAINING* BERBASIS WEB

Naning^{1*}, Agung Riyantomo²

Jurusan Teknik Informatika , Fakultas Teknik, Universitas Wahid Hasyim
Jl. Menoreh Tengah X/22, Sampangan, Semarang 50236.

*Email: naningayuannisa@gmail.com

Abstrak

Setiap alat dan komponen-komponen pada komputer memiliki tingkat kepentingan sendiri-sendiri, serta dapat mengalami kerusakan pada komponen komputer. Kerusakan ini biasanya menjadi suatu permasalahan bagi pengguna. Terkadang masalah tersebut, dapat ditangani oleh seorang yang mempunyai pengetahuan sangat dasar tentang kerusakan komputer. Tetapi terkadang kerusakan komponen komputer tersebut membutuhkan pengetahuan yang tinggi akan *troubleshooting* komponen-komponen komputer, sehingga membutuhkan seorang teknisi komputer untuk memperbaikinya. Pentingnya ilmu penanganan kerusakan komponen komputer tersebut, dapat menambah pengetahuan, jika suatu saat mengalami permasalahan kerusakan pada komponen komputer dapat memeriksa sendiri, sebelum membawa ke tempat *service* komputer. Namun adanya kondisi saat ini, ketersediaan dan keberadaan teknisi komputer tidak selalu ada sehingga menjadi penghambat saat ada pengguna komputer yang ingin berkonsultasi tentang kerusakan pada komponen komputer. Metode pengembangan sistem yang digunakan dalam pembuatan sistem pakar adalah metode *waterfall* dengan menggunakan metode inferensi *forward chaining* dan pemrograman PHP, dan database MySQL. Kesimpulan yang didapat dari hasil penelitian ini antara lain aplikasi sistem pakar dirancang mampu mendiagnosis kerusakan komponen komputer berdasarkan gejala yang diinput oleh admin.

Kata kunci: *forward chaining*, kerusakan komponen komputer, sistem pakar.

1. PENDAHULUAN

Setiap alat dan komponen yang terdapat di dalam komputer memiliki tingkat kepentingan sendiri-sendiri, serta dapat mengalami kerusakan. kerusakan ini biasanya menjadi suatu masalah bagi pengguna komputer. Untuk mengatasi kerusakan tersebut, dapat ditangani oleh orang yang mempunyai pengetahuan sangat dasar tentang kerusakan komputer. Tetapi terkadang permasalahan tersebut juga memerlukan tingkat pengetahuan yang tinggi akan *troubleshooting* komponen-komponen komputer, sehingga membutuhkan teknisi khusus untuk memperbaikinya.

Pentingnya ilmu pendeteksian masalah kerusakan komponen komputer tersebut tentunya akan menambah pengetahuan, jadi bila suatu saat mengalami permasalahan kerusakan komponen komputer dapat melakukan pemeriksaan sendiri terlebih dulu sebelum membawa ke tempat *service* komputer. Namun adanya kondisi saat ini, ketersediaan dan keberadaan ahli komputer tidak selalu ada sehingga menjadi penghambat saat ada pengguna komputer yang ingin bertanya tentang kerusakan pada *hardware* komputer. Berdasarkan hal tersebut menunjukkan bahwa perlu adanya suatu sistem yang dapat dijadikan sebagai alat untuk mempermudah pengguna komputer mendapatkan informasi dan melakukan konsultasi mengenai kerusakan *hardware* komputer. Dengan adanya sistem pakar, pihak yang bukan pakar dapat mengatasi permasalahan yang biasa diselesaikan oleh pakar, melalui gejala-gejala kerusakan yang timbul pada perangkat komputer, maka sistem pakar dapat mendeteksi komponen komputer yang mengalami kerusakan seperti layaknya teknisi komputer.

Merujuk dari permasalahan tersebut maka penulis ingin membuat sistem diagnosa kerusakan komputer menggunakan metode *forward chaining* berbasis *web* yang dapat membantu mengidentifikasi kerusakan pada komponen komputer, menemukan penyebab kerusakan pada komputer berdasarkan gejala yang terjadi dan memberikan solusi. Rumusan masalah dalam penelitian ini yaitu bagaimana membuat sebuah Sistem Pakar Diagnosa Kerusakan Pada Komputer Dengan Metode *Forward Chaining* Berbasis Web yang mendeteksi kerusakan VGA, RAM, Power Supply, HDD, Monitor dan *Motherboard*. Selain itu bagaimana memberikan solusi informasi kepada *user* komputer dalam menemukan letak permasalahan yang terjadi pada VGA, RAM, Power Supply,

HDD, Monitor dan *Motherboard*. Sistem hanya mendiagnosis kerusakan pada komputer meliputi kerusakan pada VGA, RAM, Power Supply, HDD, dan *Motherboard*. Sistem pakar yang dibuat menggunakan metode *forward chaining* berbasis *web* dengan menggunakan bahasa pemrograman PHP dan basis data *MYSQL*. Tujuan penelitian ini yaitu menghasilkan sistem pakar yang dibuat untuk mempermudah pengguna komputer dalam mendiagnosis kerusakan VGA, RAM, Power Supply, HDD, Monitor dan *Motherboard*.

Penelitian terkait digunakan sebagai referensi dalam perancangan sistem yang diambil dari penelitian terdahulu sebagai acuan dalam perancangan dan pembangunan sistem. Sistem pakar diagnosa kerusakan *mainboard* komputer *expert system diagnosis of computer mainboard failure*. (Ambarita, 2017). Latar belakang penelitian ini adalah, Banyaknya permasalahan yang sering dikeluhkan oleh pengguna komputer, Sebagian dari pengguna komputer beranggapan bahwa pekerjaan menelusuri kerusakan komponen komputer terlalu rumit untuk dilakukan sehingga memerlukan teknisi dalam menangani kerusakan pada *mainboard* tersebut. Berdasarkan masalah tersebut, perlu adanya suatu aplikasi. Sistem Pakar Diagnosa Kerusakan *Mainboard*. Sistem dapat memberikan kemudahan kepada pengguna dalam mendiagnosis kerusakan pada *mainboard* komputer melalui *form* diagnosa kemudian menampilkan hasil diagnosa meliputi kerusakan, jumlah gejala, detail solusi, sistem dibuat menggunakan metode *forward chaining*, bahasa pemrograman yang digunakan PHP dan *database* *MySql* dan sistem ini berbasis *web*. Sistem pakar diagnosa kerusakan *mainboard* komputer *expert system diagnosis of computer mainboard failure* dinilai kurang, karena perangkat komputer tidak hanya *Mainboard* saja. Penelitian terkait berikutnya yaitu Penerapan Metode *Forward Chaining* Pada Sistem Pakar Kerusakan Komputer. (Oktapiani, 2017). Latar belakang penelitian ini adalah, sulitnya mencari informasi mengenai gejala kerusakan pada perangkat komputer dan cara penanganan awal untuk mengatasi adanya gejala kerusakan, berdasarkan masalah tersebut, perlu adanya perangkat lunak *system* pakar kerusakan komputer, *system* dapat memberikan kemudahan pengguna dalam mendiagnosis kerusakan perangkat komputer melalui menu-menu yang sudah tersedia pada *system*, mulai dari menu *system* pakar, dimana menu *system* pakar ini menampilkan berbagai pertanyaan gejala kerusakan pada komputer, *system* pakar dibuat menggunakan bahasa pemrograman visual basic. Namun *system* dalam tampilan masih kurang optimal karena pertama kali *system* dibuka langsung menampilkan berbagai menu, sedangkan pengguna hanya membutuhkan menu *system* pakar saja. Terdapat penelitian terkait lainnya yaitu Rancang bangun sistem pakar diagnosa kerusakan pada komputer berbasis desktop dengan metode *forward chaining*. (Yusuf, 2018). Latar Belakang penelitian ini adalah, banyaknya pengguna komputer dan kurangnya ketersediaan teknisi komputer, perlunya tenaga mesin guna memenuhi kebutuhan manusia, salah satunya memberi pelayanan dalam menentukan diagnosa kerusakan pada perangkat komputer, dalam permasalahan tersebut perlu adanya sistem pakar kerusakan pada komputer, sistem dapat memberi kemudahan kepada pengguna dalam menentukan diagnosa kerusakan pada perangkat komputer meliputi diagnosa kerusakan VGA, Booting, HDD, dan *Motherboard*, sistem dibuat menggunakan *database* *SQLite* dan bahasa pemrograman Java. Sistem pakar yang diterapkan pada penelitian tersebut berupa aplikasi desktop. Aplikasi desktop dinilai kurang populer pada saat ini. Perbedaan dengan penelitian yang akan dilakukan oleh penulis yaitu deteksi kerusakan pada perangkat komputer. Penulis membuat sistem diagnosa kerusakan pada komputer meliputi kerusakan pada VGA, RAM, Power Supply, HDD, Monitor dan *Motherboard*.

“Sistem pakar (*Expert System*) adalah salah satu cabang dari AI (*Artificial Intelligence*) khusus untuk penyelesaian masalah tingkat manusia yang pakar. Sistem pakar adalah sistem yang menggunakan pengetahuan manusia yang direkam ke dalam komputer untuk memecahkan persoalan yang memerlukan keahlian manusia. Sedangkan AI itu sendiri atau yang disebut juga kecerdasan buatan adalah tingkah laku mesin yang jika dilakukan oleh manusia, akan disebut cerdas”. (Turban, 1995). Metode *forward chaining* adalah metode pencarian atau teknik pelacakan ke depan yang dimulai dengan informasi yang ada dan penggabungan *rule* untuk menghasilkan satu kesimpulan. (Ramadhan & Pane, 2018). Pada penelitian ini metode pengembangan sistem yang digunakan adalah *waterfall*. Metode *waterfall* menyarankan pengembangan perangkat lunak secara sistematis dan runtut yang dimulai dari tingkatan sistem tertinggi dan berlanjut ke tahap analisis, desain, pengkodean, pengujian dan pemeliharaan. Kelebihan dari metode ini adalah terstruktur, dinamis, dan sequintal (Pressman, 2002).

2. METODOLOGI

Pada penelitian ini metode perancangan aplikasi yang digunakan adalah *forward chaining*. Metode *forward chaining* adalah teknik pencarian yang dimulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian *IF_THEN*. Bila ada fakta yang cocok dengan bagian *IF*. Maka *rule* tersebut dieksekusi.

Tabel 1. Rule

NO	Aturan
R1	IF G001 Komputer mati total AND G002 Setelah dihidupkan, tidak ada tampilan di monitor AND G003 Lampu indikator (led) di panel depan menyala (CPU) AND G004 Lampu indikator (led) monitor berkedip-kedip AND G005 Tidak ada suara beep di speaker AND G006 Komputer mati sendiri saat digunakan AND G007 Power supply tidak berputar AND G008 komputer hang AND G010 muncul pesan “press F1 to continou” setelah kita menekan tombol F1 tidak masuk Operating system dan muncul pesan “Operating system not found” AND G011 komputer restart sendiri AND G014 layar timbul garis, bercak-bercak warna, bahkan gambar semu THEN P01 Motherboard.
R2	IF G001 Komputer mati total AND G002 Setelah dihidupkan, tidak ada tampilan di monitor AND G003 Lampu indikator (led) di panel depan menyala (CPU) AND G004 Lampu indikator (led) monitor berkedip-kedip AND G006 Komputer mati sendiri saat digunakan AND G007 Power supply tidak berputar THEN P02 Power Supply.
R3	IF G008 Komputer Hang AND G009 harddisk failure AND G010 muncul pesan “press F1 to continou” setelah kita menekan tombol F1 tidak masuk Operating system dan muncul pesan “Operating system not found” AND G011 komputer restart sendiri AND G012 harddisk tidak terdeteksi THEN P03 Harddisk.
R4	IF G002 Setelah dihidupkan, tidak ada tampilan di monitor AND G008 komputer hang AND G011 komputer restart sendiri AND G013 driver sering stopped working AND G014 layar timbul garis, bercak-bercak warna, bahkan gambar semu AND G015 komputer bluescreen AND G016 Terdengar bunyi beep 1x panjang 3x pendek THEN P04 VGA CARD.
R5	IF G017 monitor tidak mau menyala AND G018 monitor gelap saat loading Windows AND G019 bunyi beep panjang 1 kali dan pendek 2 kali AND G020 indikator power monitor tidak menyala AND G021 pada sudut monitor tampak bercak kebiru-biruan AND G022 ukuran tampilan monitor tidak sesuai keinginan AND G023 lampu indikator pada monitor hidup berwarna oranye, atau lampu berkedip-kedip AND layar monitor tampak buram THEN P05 Monitor.
R6	IF G002 Setelah dihidupkan, tidak ada tampilan di monitor AND G006 Komputer mati sendiri saat digunakan AND G008 komputer hang AND G011 komputer restart sendiri AND G014 layar timbul garis, bercak-bercak warna, bahkan gambar semu AND G015 komputer bluescreen AND G025 komputer tidak menyala lampu indikator berkedip AND G026 bunyi beep 2 kali AND G027 bunyi beep 3 kali, beep terus 3 kali berturut-turut pada saat komputer dihidupkan AND G028 bunyi beep 11 kali AND G029 bunyi beep panjang pada saat komputer dihidupkan AND G030 komputer sering crash AND G031 komputer sangat lambat AND G032 komputer tidak mau booting AND G033 Pada saat blue screen terdapat pesan “PEN LIST CORRUP” THEN P06 Ram.

Bila sebuah *rule* dieksekusi, maka sebuah fakta baru (bagian *THEN*) dieksekusi ke dalam database. Setiap *rule* hanya boleh dieksekusi sekali saja. (Hayadi, 2018)

Tahap-tahap pembangunan yang digunakan adalah:

a. *Analysis*

Pada tahap ini penulis melakukan analisis dan mengumpulkan data yang dibutuhkan untuk pembuatan Sistem Pakar Diagnosa Kerusakan Pada Komputer Dengan Metode *Forward Chaining* Berbasis *Web*, dari hasil wawancara dengan pakar serta beberapa referensi terkait.

b. *Design/Perancangan*

Pada tahap ini penulis memodelkan kebutuhan perangkat lunak dari tahap analisis kebutuhan ke desain. Pemodelan yang dilakukan yaitu pemodelan proses dengan UML, dan desain antar muka pengguna (*user interface*) menggunakan mockup, dalam melakukan perancangan, penulis juga menentukan rancangan *input* dan *output* yang digunakan. Perancangan input terdiri dari gejala kerusakan, untuk perancangan output terdiri dari diagnosa perangkat.

c. *Coding*

Pada tahap pengkodean penulis merealisasikan desain ke dalam bahasa pemrograman PHP dalam hal ini penulis menggunakan PHP Native. Hasil dari pengkodean adalah hasil dari desain yang telah di rancang, untuk mengetahui arus pekerjaan

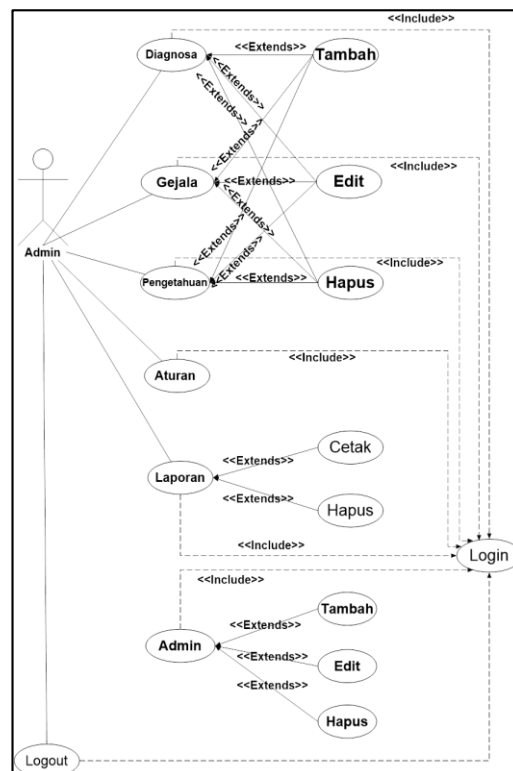
d. *Testing*

Ditahap testing perangkat lunak penulis menguji *function* pada kode yang dibuat dan fungsional dengan menguji link, button dan menu-menu yang ada dan memastikan bahwa semua bagian telah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) untuk memastikan bahwa keluaran dari perangkat lunak sesuai dengan yang diinginkan.

e. *Maintenance*

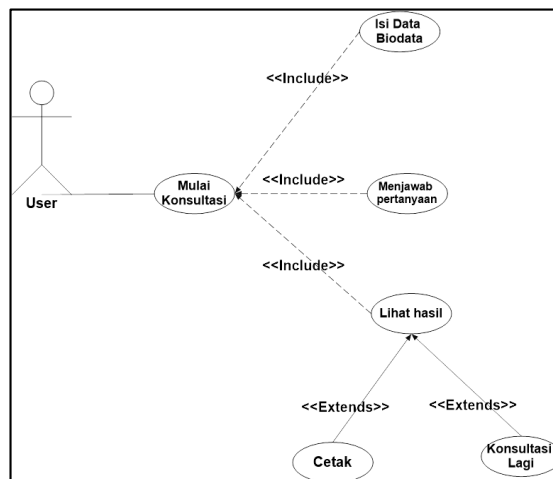
Pada tahap ini penulis melakukan Pemeliharaan terhadap perangkat lunak dimulai dengan pengoperasian sistem dan jika diperlukan untuk melakukan perbaikan kecil. Kemudian, jika waktu penggunaan sistem habis maka akan masuk lagi pada tahap perencanaan.

Tahap desain merupakan tahap penjelasan kebutuhan yang telah direncanakan menjadi bentuk representasi dari sistem yang akan dibuat, dengan merancang *Use Case Diagram*, *Activity Diagram*, *ERD*(*Entity Reality Diagram*), perancangan *User Interface*. Diagram use case admin menggambarkan hubungan antara aktor admin dengan use case fungsionalitas dari sistem, seperti yang ditunjukkan pada Gambar 1.



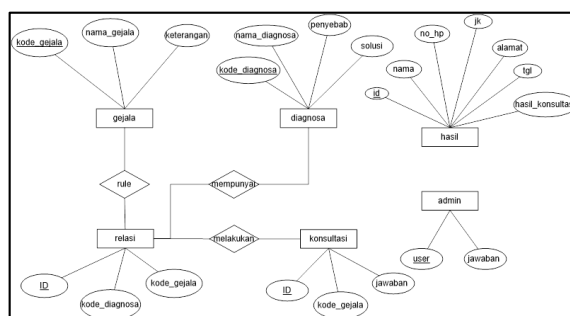
Gambar 1. Diagram Use case admin

Diagram use case user menggambarkan hubungan antara aktor user dengan use case fungsionalitas dari sistem, seperti yang ditunjukkan pada Gambar 2.



Gambar 2. Diagram Use Case User

Diagram *Entity Relationships* menggambarkan hubungan (*relationship*) antara satu entitas dengan entitas lain, lengkap dengan atribut dari masing-masing entitas, seperti yang ditunjukkan pada Gambar 3.



Gambar 3. Diagram Entity Relationships dari Sistem

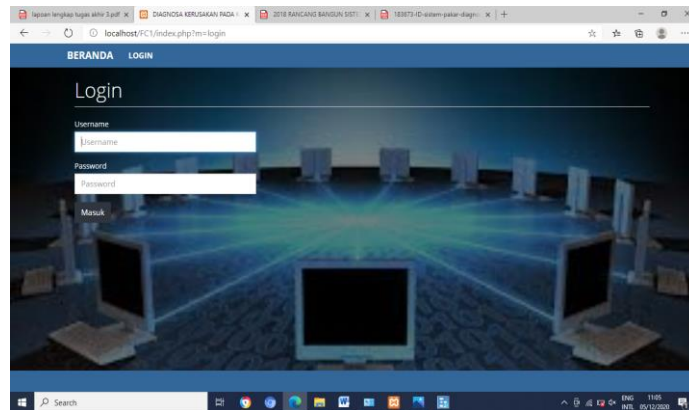
3. HASIL DAN PEMBAHASAN

Halaman beranda dari sistem pakar yang dihasilkan, memuat tombol mulai berkonsultasi yang digunakan jika pengguna ingin melakukan konsultasi mengenai kerusakan pada komputer, seperti yang ditunjukkan pada Gambar 4.



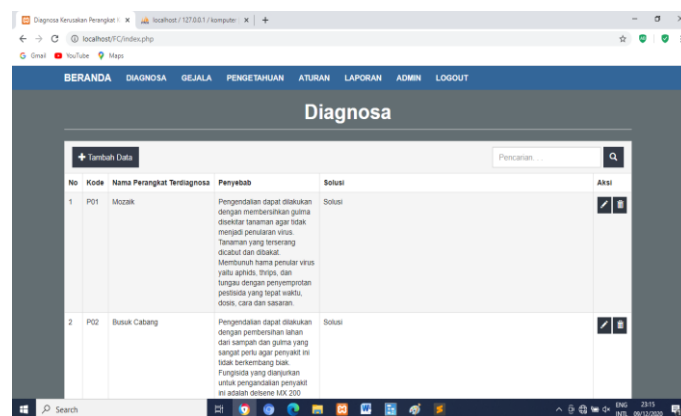
Gambar 4. Halaman Beranda

Halaman login admin digunakan untuk mengakses sistem pakar dengan memasukkan username dan password, seperti yang ditunjukkan pada Gambar 5.



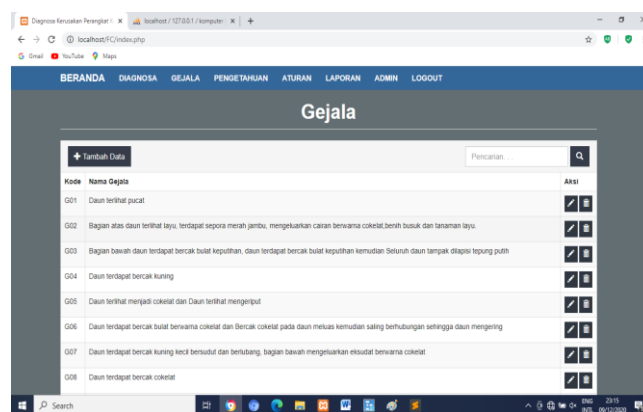
Gambar 5. Halaman Login

Halaman diagnosa digunakan untuk mengelola nama perangkat, penyebab kerusakan, dan solusinya, seperti yang ditunjukkan pada Gambar 6.



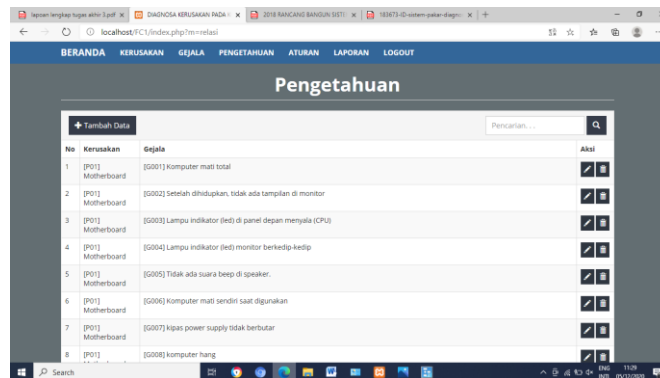
Gambar 6. Halaman Diagnosa

Halaman gejala digunakan untuk mengelola gejala-gejala yang dapat dijadikan sebagai bahan diagnosa kerusakan, seperti yang ditunjukkan pada Gambar 7.



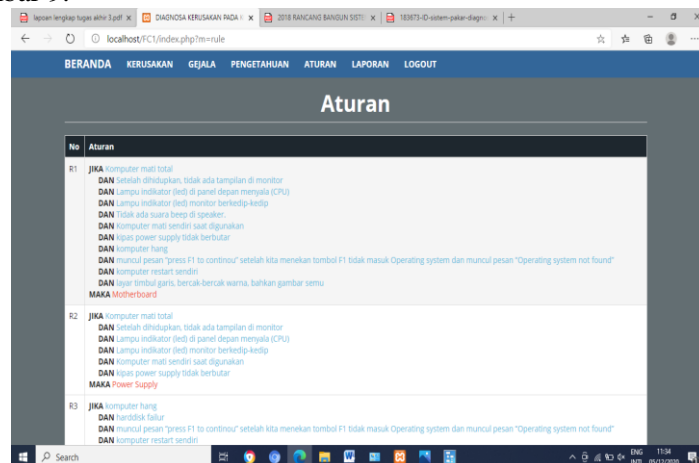
Gambar 7. Halaman Gejala

Halaman pengetahuan digunakan untuk mengelola perangkat yang dideteksi rusak beserta gejalanya, seperti yang ditunjukkan pada Gambar 8.



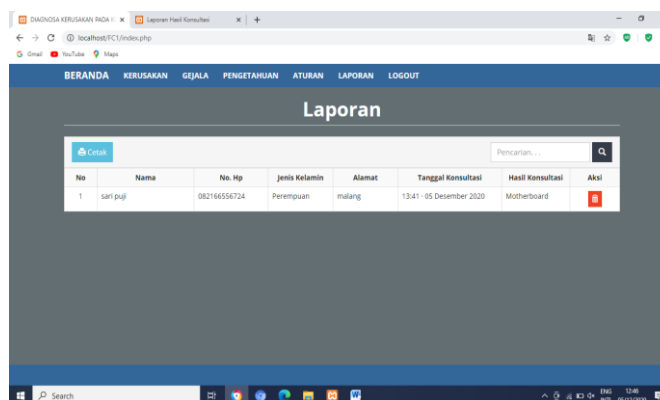
Gambar 8. Halaman Pengetahuan

Halaman aturan digunakan untuk mengelola aturan-aturan IF THEN rule, seperti yang ditunjukkan pada Gambar 9.



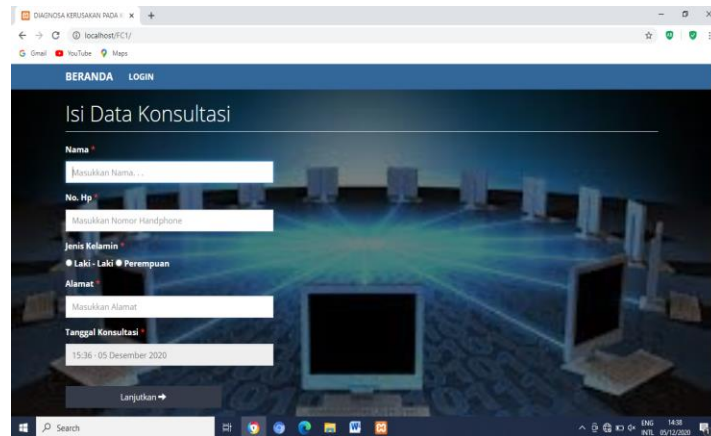
Gambar 9. Halaman Aturan

Halaman laporan digunakan untuk mengelola laporan konsultasi yang sudah dilakukan oleh pengguna, seperti yang ditunjukkan pada Gambar 10.



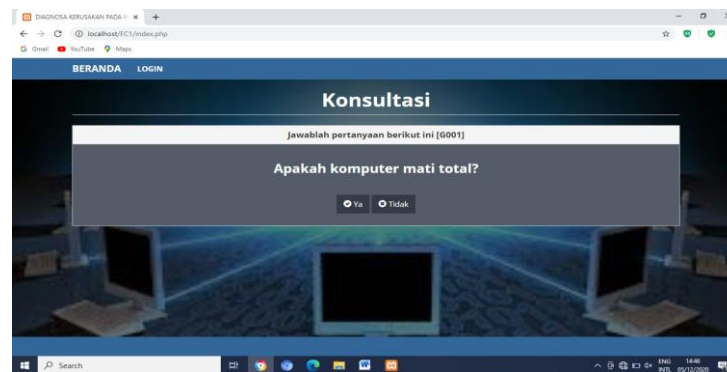
Gambar 10. Halaman Laporan

Halaman isi data konsultasi digunakan untuk mengisi data awal dari pengguna yang ingin melakukan konsultasi melalui sistem pakar, seperti yang ditunjukkan pada Gambar 11.



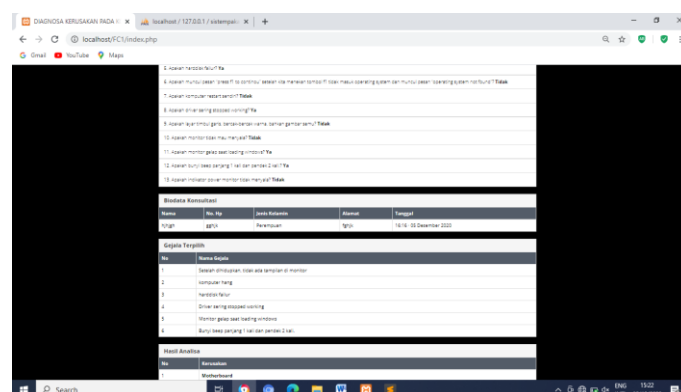
Gambar 11. Halaman Isi Data Konsultasi

Halaman konsultasi digunakan pengguna melakukan konsultasi melalui sistem pakar, seperti yang ditunjukkan pada Gambar 12.



Gambar 12. Halaman Konsultasi

Halaman hasil konsultasi digunakan pengguna melihat hasil konsultasi melalui sistem pakar, seperti yang ditunjukkan pada Gambar 13.



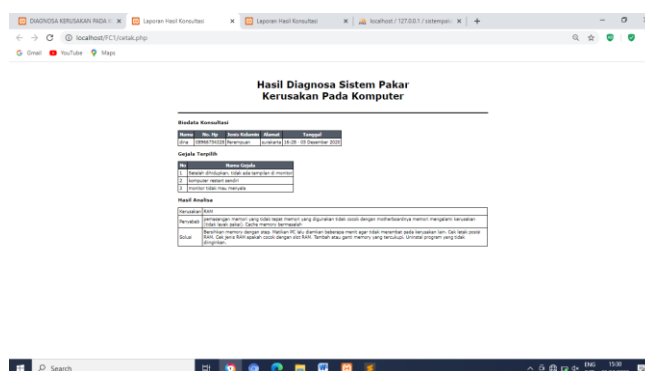
Gambar 13. Halaman Hasil Konsultasi

Pengujian sistem pakar yang sudah dihasilkan ini menggunakan metode black box, seperti yang ditunjukkan pada Tabel 1.

Tabel 1. Hasil Pengujian Black Box

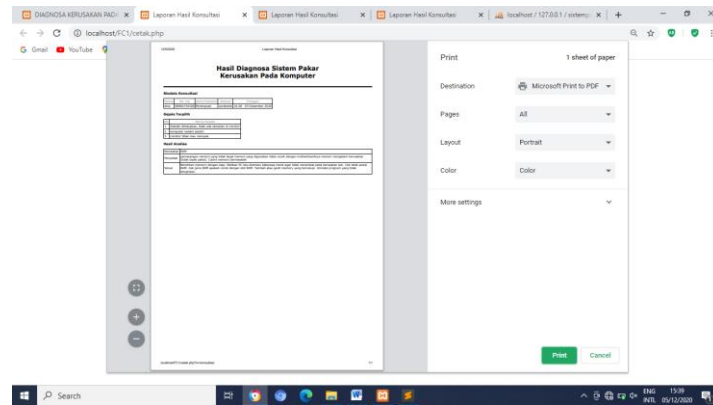
No	Input	Fungsi	Output	Hasil
1	Login	Menampilkan Halaman Login Admin Untuk Masuk Ke Dalam Menu Form Login	Tampilan Halaman Login Admin	Berhasil
2	Beranda	Menampilkan Menu Halaman Admin	Tampilan Menu Halaman Admin	Berhasil
3	Menu Kerusakan	Menampilkan Halaman Tambah, Edit Hapus Data Kerusakan	Tampilan Menu Halaman Kerusakan	Berhasil
4	Menu Gejala	Menampilkan Menu Gejala, Tambah, Edit Hapus Data Gejala	Tampilan Menu Gejala	Berhasil
5	Menu Pengetahuan	Menampilkan Menu Pengetahuan, Tambah, Edit Dan Hapus Data Pengetahuan	Tampilan Menu Pengetahuan	Berhasil
6	Menu Aturan	Menampilkan Menu Aturan	Tampilan Menu Aturan	Berhasil
7	Menu Laporan	Menampilkan Menu Laporan Konsultasi Dari User	Tampilan Laporan Konsultasi User	Berhasil
8	Menu Data Konsultasi	Menampilkan Menu Data Konsultasi	Tampilan Menu Data Konsultasi	Berhasil
9	Menu Konsultasi	Menampilkan Menu Konsultasi	Tampilan Menu Konsultasi	Berhasil
10	Menu Hasil Konsultasi	Menampilkan Menu Hasil Konsultasi	Tampilan Menu Hasil Konsultasi	Berhasil
11	Menu Hasil Diagnosa	Menampilkan Menu Hasil Diagnosa	Tampilan Hasil Diagnosa	Berhasil
12	Menu Halaman Cetak Diagnosa	Menampilkan Data Hasil Diagnosa Berupa Format Pdf	Tampilan Cetak Hasil Diagnosa Format Pdf	Berhasil

Halaman hasil diagnosa digunakan pengguna melihat hasil diagnosa kerusakan melalui sistem pakar, seperti yang ditunjukkan pada Gambar 14.



Gambar 14. Halaman Hasil Diagnosa User

Halaman cetak hasil diagnosa digunakan pengguna mencetak hasil diagnosa kerusakan melalui sistem pakar, seperti yang ditunjukkan pada Gambar 15.



Gambar 15. Halaman Cetak

4. KESIMPULAN

Berdasarkan penelitian sistem pakar diagnosa kerusakan pada komputer dengan metode forward chaining berbasis web ini, peneliti telah berhasil membuat sistem pakar diagnosa kerusakan pada komputer. Sistem dapat memberikan kemudahan kepada masyarakat atau pengguna komputer dalam mendiagnosa kerusakan pada komputer. Sistem pakar yang dibangun dapat memberikan hasil diagnosa berdasarkan fakta-fakta yang telah diberikan. Adapun saran yang dapat dikemukakan yaitu sistem ini dapat dikembangkan menggunakan metode algoritma lain dalam pembangunannya yang lebih lanjut serta penambahan data-data jenis kerusakan, dan gejala serta solusi dan pertanyaan gejala sesuai dengan perkembangan teknologi.

DAFTAR PUSTAKA

- Ambarita, R. (2017). Sistem Pakar Diagnosa Kerusakan Mainboard Komputer Expert System Diagnosis Of Computer Mainboard Failure. *Indonesian Journal on Information System*, ISSN 2548-6438.
- Hayadi, H. B. (2018). *Sistem pakar penyelesai kasus menentukan minat baca, kecenderungan, dan karakter siswa dengan metode forward chaining*. Yogyakarta: Deepublish.
- Nugroho, A. (2009). *Rekayasa Perangkat Lunak Menggunakan UML & Java*. Yogyakarta: ANDI.
- Oktapiani, R. (2017). Penerapan Metode Forward Chaining Pada Sistem Pakar Kerusakan Komputer. *IJCIT (Indonesian Journal on Computer and Information Technology)*, 14-23.
- Pressman, R. S. (2002). *Rekayasa Perangkat Lunak Pendekatan (BUKU SATU)*. Yogyakarta: ANDI.
- Ramadhan, P. S., & Pane, F. S. (2018). *Mengenal Metode Sistem Pakar*. Ponorogo: Uwais Inspirasi indonesia.
- Turban, E. (1995). *Decision Support and Expert Systems, Management Support System*. New York: Prentice Hall International .
- Yusuf, I. M. (2018). Rancang Bangun Sistem Pakar Diagnosa Kerusakan Pada Komputer Berbasis Dekstop Dengan Metode Forward Chaining. *Seminar Nasional Royal*, hlm. 149-153. <https://zenodo.org/record/2528539#.YBurAXtKjIU>