

SKRIPSI

**SISTEM PAKAR IDENTIFIKASI KERUSAKAN
MOBIL NISSAN GRAND LIVINA MENGGUNAKAN
METODE *FORWARD CHAINING***



Disusun oleh :

HENDRI KURNIAWAN

16.0504.0144

**PROGRAM STUDI TEKNIK INFORMATIKA S1
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAGELANG**

2021

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi pada Revolusi Industri 4.0 dewasa ini, mobil menjadi produk manufaktur yang paling banyak digunakan sebagai kendaraan angkut. Manusia berlomba-lomba menciptakan mobil yang lebih ramah lingkungan, hemat bahan bakar, lebih ergonomis, lebih *user friendly* dan sebagainya. Sesuai dengan kebutuhan manusia, mobil mempunyai konfigurasi *body*, mesin bahkan kelas yang berbeda. Beberapa kategori mobil di Indonesia seperti Sedan, Station Wagon, Hatchback, Coupe dan Convertible, Sport Utility Vehicle (SUV), Multi Purpose Vehicle (MPV), dan Pick-Up (Mobil88, 2019).

Selain itu, 10 merek mobil terlaris di Indonesia yang dirilis Gabungan Industri Kendaraan Bermotor (GAIKINDO) mencatatkan penjualan merk Toyota sebanyak 331.797 unit selama 2019, diposisi kedua Daihatsu berhasil menjual 202.738 unit, diposisi ketiga ditempati merk Honda mencapai angka penjualan 137.339 unit, diposisi keempat merk Mitsubishi Motors berhasil menjual 119.011 unit, sedangkan diposisi kelima merk Suzuki mampu menjual 100.383 unit, diposisi keenam merk Wuling produk asal Tiongkok menjadi merk terlaris dengan jumlah penjualan 22.343 unit, diposisi ketujuh ditempati merk Nissan dengan angka penjualan mencapai 12.302 unit, diposisi berikutnya merk Datsun menjadi urutan kedelapan dengan hasil penjualan 6.487 unit, diposisi kesembilan merk Mazda berhasil menjual 4.884 unit, sementara diposisi kesepuluh merk DFSK mampu menjual

sebanyak 3.857 unit (Toyota & Motor, 2019). Mobil nissan menjadi salah satu merek mobil terlaris di indonesia berdasarkan penjualan di tahun 2019. Para produsen mobil ini berlomba untuk merebut pangsa pasar dengan menawarkan berbagai teknologi canggih untuk dinikmati konsumen. Kendaraan ini banyak digunakan masyarakat sebagai kendaraan pribadi. Hal ini dapat dilihat dari data penjualan Nissan Grand Livina dari tahun ke tahun. Sepanjang Januari 2020 secara *year on year* (dibanding Januari 2019), penjualan (*wholesales*) Nissan naik 51%. Jika di Januari 2019 penjualan (*wholesales*) masih sebanyak 387 unit, pada Januari 2020 lalu, penjualan (*wholesales*) yang dibukukan sebanyak 585 unit (Livina et al., 2020). Melihat penjualan yang meningkat dari tahun ke tahun, mobil Nissan adalah salah satu yang sangat diminati masyarakat Indonesia umumnya adalah Nissan Grand Livina.

Kendalanya kerusakan pada mobil terjadi akibat kelalaian dalam melakukan perawatan atau minimnya pengetahuan dalam merawat mobil. Pemilik mobil baru menyadari kerusakan setelah mobil tidak dapat beroperasi sebagaimana mestinya. Misalnya, jika mobil tidak dapat di start atau mesin mobil berisik dan tidak mempunyai gambaran mengapa hal tersebut terjadi (Satwika, 2012).

Salah satu contoh konkrit yang dapat dilihat buktinya adalah jika terjadi permasalahan pada mobil akan membuat pusing bagi pemilik kendaraan tersebut (Abdullah, 2016). Kebanyakan pengendara cenderung menyerahkan mobil kepada mekanik tanpa mengetahui sebenarnya kerusakan tersebut merupakan kerusakan sederhana atau terlalu rumit untuk diperbaiki. Menyerahkan kerusakan pada mekanik merupakan sebuah langkah praktis dan

menjadi sebuah solusi yang paling mudah. Karena ketidaktahuan pemilik mobil dengan kerusakan yang terjadi. Namun tidak semua pemilik mobil ini selalu dibuat senang ataupun nyaman.

Sistem pakar merupakan program pemberi *advice*/nasihat terkomputerisasi yang ditunjukan untuk meniru proses *reasoning* (pertimbangan) dan pengetahuan dari pakar dalam menyelesaikan masalah yang spesifik. Bidang ini digunakan lebih banyak dari pada bidang-bidang kecerdasan buatan lainnya. Sistem Pakar menarik minat yang besar dalam suatu organisasi disebabkan kemampuannya dalam meningkatkan produktifitas dan dalam meningkatkan gugus kerja di berbagai bidang tertentu dimana pakar manusia akan mengalami kesulitan dalam mendapatkan dan mempertahankan kemampuan itu (Ramadiani & Mulawarman, 2018). Dalam sistem pakar, komponen yang berfungsi sebagai otak untuk memecahkan masalah/ berfikir seperti seorang pakar disebut mesin inferensi. *Forward chaining* adalah salah satu metode dalam mesin inferensi. Terdapat 2 model mesin inferensi yaitu *forward chaining* dan *backward chaining* (Al-ajlan, 2015). Metode *forward chaining* merupakan metode pelacakan yang dimulai dari fakta yang ada untuk menghasilkan tujuan atau menarik kesimpulan (Al-ajlan, 2015). Sedangkan metode *backward chaining* merupakan metode pelacakan yang dimulai dari harapan atau tujuan dan kemudian mencoba untuk menemukan bukti atau fakta yang mendukung hipotesa (Al-ajlan, 2015).

Metode *forward chaining* dan *backward chaining* mempunyai karakteristik bidang masing-masing, dimana *forward chaining* cocok digunakan dalam perencanaan, perancangan dan pemantauan proses (Akil,

2017) sedangkan *backward chaining* digunakan untuk klasifikasi dan diagnosa (Akil, 2017). Penggunaan metode *forward chaining* / *backward chaining* dapat dilihat dari kebutuhan kebutuhan dalam membangun sistem. Jika tujuan sistem dibangun untuk menemukan kesimpulan dari fakta yang diberikan, maka sistem tersebut harus menggunakan metode *forward chaining*. Dalam proses perhitungannya digunakan metode *certainty factor*, *certainty factor* merupakan salah satu teknik yang digunakan untuk mengatasi ketidakpastian dalam pengambilan keputusan (Yusman et al., 2017). Dengan menerapkan metode *certainty factor* untuk perhitungan kemungkinan kerusakan berdasarkan gejala yang dipilih *user* maka *user* akan menerima hasil berupa kemungkinan terbesar kerusakan yang terjadi sehingga *user* bisa mengetahui apa yang rusak pada kendaraanya. Hasil perhitungan ditampilkan berupa persentase kerusakan yang dihitung berdasarkan nilai MB dan MD yang telah ditetapkan oleh pakar (Nugroho, 2020).

Implementasi sistem pakar banyak digunakan dalam bidang teknologi karena sistem pakar dipandang sebagai cara penyimpanan pengetahuan pakar pada bidang tertentu dalam program komputer sehingga keputusan dapat diberikan dalam melakukan penalaran secara cerdas. Maka dari itu penelitian ini akan menggunakan metode *forward chaining* yang digunakan dalam sistem identifikasi kerusakan mobil. Untuk membantu mengoptimisasi kerja daripada manusia dalam mengidentifikasi kerusakan mobil dengan ini peneliti mengambil judul **“SISTEM PAKAR IDENTIFIKASI KERUSAKAN MOBIL NISSAN GRAND LIVINA MENGGUNAKAN METODE FORWARD CHAINING”**.

B. Rumusan Masalah

Berdasarkan uraian latar belakang, dapat dirumuskan permasalahan yang ada yaitu Bagaimana membangun sistem pakar dengan menerapkan metode *Forward Chaining* dan *Certainty Factor* untuk mengetahui kerusakan pada Mobil Nissan Grand Livina melalui pertanyaan gejala.

C. Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Membangun sebuah sistem identifikasi kerusakan mobil Nissan Grand Livina berdasarkan basis pengetahuan yang tersimpan di dalamnya dengan menggunakan metode *Forward Chaining* dan *Certainty Factor*.
2. Menghasilkan sistem yang dapat digunakan untuk membuat keputusan mengenai kerusakan pada mobil Nissan Grand Livina sesuai pakar.

D. Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Menerapkan metode *Forward chaining* dan *Certainty Factor* untuk membantu dalam identifikasi dan melakukan tindakan awal jika terjadi kerusakan pada mobil Nissan Grand Livina yang dimiliki.
2. Memberikan sarana penyampaian informasi yang sesuai pakar jika terjadi kerusakan pada mobil Nissan Grand Livina.

BAB II

TINJAUAN PUSTAKA

A. Penelitian Relevan

1. Penelitian yang dilakukan (Septarini & Nugroho, 2018) Dengan judul “Rancang Bangun Aplikasi Sistem Diagnosa Kerusakan Pada Mobil Daihatsu Xenia 1.3 M/T Dengan Metode *Forward Chaining*”. Dalam penelitian ini, peneliti melakukan perencanaan sistem dalam membuat basis pengetahuan memakai pohon keputusan dan *if-then* sebagai representasi pengetahuan. Pembuatan metode inferensi memakai metode penelusuran ke depan (*forward chaining*) yang telah dimodifikasi sehingga sesuai dengan permasalahan. Dalam penelitian ini terdapat metode pengembangan sistem, metode yang digunakan adalah *waterfall* dimana terdapat lima tahap, diantaranya pengujian pada tahap ini sistem yang telah dibangun dicoba apakah *User stories* sudah dipenuhi dan apabila terpenuhi *software* siap dirilis. Hasil dari aplikasi ini dapat disimpulkan bahwa program sudah cukup baik walaupun jenis kerusakan yang dihasilkan belum lengkap.
2. Penelitian yang dilakukan (Rusdiansyah, 2017) Dengan judul “Sistem Pakar Mendeteksi Kerusakan Mesin VVT-I Berbasis WEB Pada Kendaraan Toyota Vios”. Dalam penelitian ini, peneliti menggunakan metode *forward chaining* dengan menggunakan topologi penelusuran *Best First Search* melakukan pencarian dengan menelusuri setiap simpul pada level yang sama terlebih dahulu, kemudian turun ke level

yang berikutnya sampai ditemukan suatu goal. Aplikasi yang digunakan adalah *Adobe Dreamweaver CS5* Dengan *database* menggunakan *MySQL*. Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat disimpulkan bahwa Sistem pakar gangguan mendiagnosa kerusakan pada mesin Toyota Vios membantu para pengguna mobil Vios dalam mendiagnosa kerusakan dan gambaran kerusakan pada mesin Toyota Vios.

3. Penelitian yang dilakukan (Sulaiman, 2020) Dengan judul “Perancangan Prototipe Sistem Pakar Diagnosa Kerusakan Mobil Toyota Tipe Mpv Menggunakan Metode *Forward* dan *Backward Chaining* Berbasis Android”. Dengan metode inferensi *forward chaining* dan *backward chaining*. Metode *forward chaining* digunakan untuk menarik kesimpulan kerusakan dari beberapa fakta berupa gejala yang telah dipilih *user*. Metode *backward chaining* dalam penelitian ini berfungsi untuk menampilkan nama kerusakan, gejala-gejala yang menandai kerusakan tersebut. Hasil dari penelitian ini yaitu sistem pakar diagnosa kerusakan mobil toyota menggunakan metode *forward chaining* dan *backward chaining* dapat digunakan untuk membantu pengguna dalam mendapatkan pengetahuan tentang kerusakan mobil melalui hasil diagnosa.

Berdasarkan uraian diatas, dapat diketahui bahwa sistem pakar sudah banyak diaplikasikan dalam bidang otomotif yang dapat digunakan untuk mendiagnosa kerusakan pada mobil. Sistem pakar dapat bermanfaat bagi para pengguna / orang yang membutuhkan informasi tentang kerusakan namun para pemilik kendaraan tidak menemukan bengkel / seorang ahli sebagai tempat untuk bertanya. Penggunaan metode *forward chaining*

dalam sistem pakar berfungsi untuk mencari / menentukan kesimpulan / hasil diagnosa dengan cara menelusuri gejala / fakta dari *user*. Metode *forward chaining* bekerja dengan data yang tersedia dan menggunakan aturan-aturan inferensi untuk mendapatkan data yang lain sampai sasaran atau kesimpulan didapatkan.

Untuk itu peneliti menerapkan metode *forward chaining* dalam sistem diagnosa kerusakan mobil grand livina untuk mendapatkan hasil diagnosa berupa kerusakan yang dialami berdasarkan gejala yang dipilih *user* dan cara pengendaliannya / solusi.

B. Penjelasan Secara Teoritis Masing-Masing Variable

1. Sistem Pakar

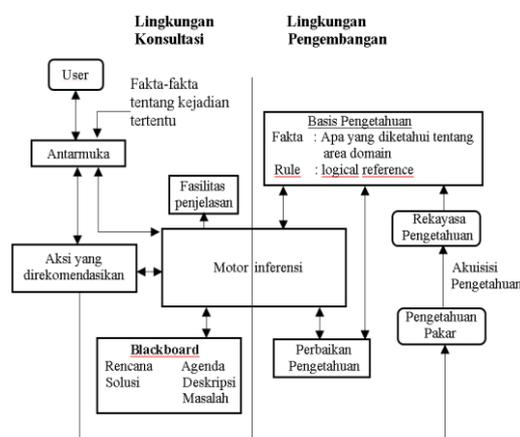
Menurut (Sulaiman, 2020) Sistem Pakar adalah sistem perangkat komputer yang menggunakan ilmu, data, dan teknik berfikir dalam mengambil *decision* atau kesimpulan untuk menyelesaikan masalah yang biasanya hanya dapat diselesaikan oleh pakar dalam bidang yang berkaitan. Sistem pakar dapat diartikan sebagai sistem yang mencoba mengenal pengetahuan manusia ke dalam komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh pakar sehingga setiap orang dapat menggunakannya untuk memecahkan berbagai masalah yang bersifat spesifik dalam hal ini adalah permasalahan pada mobil grand livina.

Menurut (Sapri, 2018) Sistem pakar merupakan kepakaran yang ditransfer dari seorang pakar (atau sumber kepakaran yang lain) ke

komputer, pengetahuan yang ada disimpan dalam komputer, dan pengguna dapat berkonsultasi pada komputer itu untuk suatu nasehat, lalu komputer dapat mengambil inferensi (menyimpulkan, mendeduksi, dll.) seperti layaknya seorang pakar, kemudian menjelaskannya ke pengguna tersebut, bila perlu dengan alasan-alasannya.

Berdasarkan definisi sistem pakar yang telah dipaparkan maka dapat disimpulkan bahwa sistem pakar merupakan program komputer yang dapat berfikir seperti manusia yang ahli dalam sebuah bidang tertentu untuk memecahkan masalah dan memberikan saran yang sesuai.

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) yang digunakan untuk membangun komponen komponennya dan memperkenalkan pengetahuan ke dalam *knowledge base* (basis pengetahuan). Lingkungan konsultasi (*consultation environment*) yang digunakan untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar (Nasir & Gultom, 2018). Komponen-komponen sistem pakar disajikan dalam Gambar 2.1



Gambar 2.1 Komponen Komponen Sistem Pakar

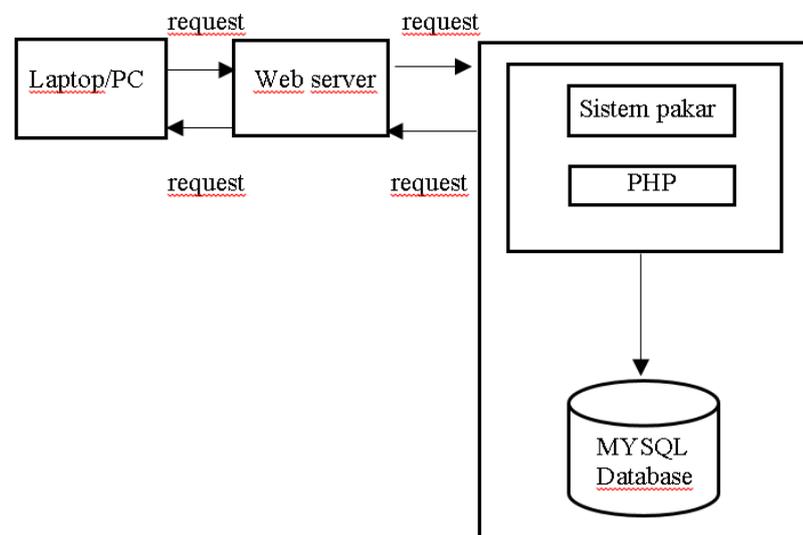
Sumber:(Fikri & Widians, 2017), (Nasir & Gultom, 2018)

Adapun penjelasan dari gambar komponen dalam sistem pakar menurut (Nasir & Gultom, 2018) yaitu :

a. Antarmuka Pengguna (*User Interface*)

User interface merupakan bagian dari sistem pakar yang digunakan sebagai media atau alat komunikasi antar user dan sistem. Di dalam user interface ini dibedakan dua user :

- 1) User umum adalah user yang menggunakan sistem pakar ini untuk mencari informasi dari gangguan-gangguan yang dialami atau sekedar mencari informasi jenis-jenis kerusakan mesin pada Nissan Grand Livina beserta gejalanya.
- 2) User administrator adalah user yang bertugas untuk melakukan proses editing penambahan dan perawatan data di dalam sistem pakar jika diperlukan perubahan.



Gambar 2.2 Blok Arsitektur Sistem

Dari gambar di atas, dapat dijelaskan bahwa urutan event yang terjadi dalam sistem adalah

- 1) *User / admin* melakukan *request* alamat *URL* yang dituju ke *Web Server* pada Laptop/PC.
- 2) *Request* tersebut akan ditransmisikan via internet dari Laptop.
- 3) Kemudian *request* akan diteruskan ke *web server*. Server membaca header dan memproses permintaan.
- 4) Kode program *PHP* yang terdapat dalam dokumen ini dikompilasi dengan sistem pakar dan diformat sesuai dengan kebutuhan. Jika memang dibutuhkan untuk penggunaan *database*, maka akan terjadi pula koneksi ke *database* yang digunakan, yaitu *MySQL*.
- 5) Dokumen yang telah diproses ini akan dikirimkan kembali melalui *Web Server* sebagai *respons* atas *request* sebelumnya.
- 6) Pada *Web Server*, isi dari dokumen dikompres menjadi data *biner* dan dikirimkan ke Laptop.

b. Basis Pengetahuan

Basis Pengetahuan Basis pengetahuan memperoleh pengetahuan dari pakar dan atau sumber dokumen lainnya. Pengetahuan yang masih menggunakan bahasa alami ini harus dibawa ke bahasa yang dimengerti komputer. Tahap pengembangan basis pengetahuan meliputi :

- 1) Mendefinisikan kemungkinan penyelesaian. Dalam tahap ini yang dilakukan adalah menentukan domain pengetahuan ke dalam daftar kemungkinan penyelesaian jawaban, pilihan atau rekomendasi lain.

- 2) Mendefinisikan data masukan. Dalam tahap ini yang dilakukan adalah identifikasi dan mendaftar semua data yang diperlukan sistem.
 - 3) Pengembangan garis besar. Dalam tahap ini yang dilakukan adalah menambah domain penyelesaian dan data masukan yang diperlukan untuk mengatasi kesulitan dalam menulis aturan.
 - 4) Menggambar pohon pengetahuan, dalam tahap ini yang dilakukan adalah membuat konstruksi sebuah pohon keputusan dan pencarian.
 - 5) Membuat matrik akuisisi pengetahuan. Dalam hal ini yang dilakukan adalah membuat akuisisi basis pengetahuan pengetahuan berbentuk sebuah matrik.
 - 6) Pengembangan software, dalam hal ini yang dilakukan adalah menulis basis pengetahuan yang sudah ada dan siap digunakan kedalam bahasa yang dimengerti oleh komputer.
- c. Akuisisi Pengetahuan (*knowledge acquisition*)

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer. Dalam tahap ini knowledge engineer berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dari pakar, dilengkapi dengan buku, basis data, laporan penelitian dan pengalaman pemakai.

Metode akuisisi pengetahuan :

1) Wawancara

Metode yang paling banyak digunakan, yang melibatkan pembicaraan dengan pakar secara langsung dalam suatu wawancara.

2) Analisis protokol Dalam metode ini pakar diminta untuk melakukan suatu pekerjaan dan mengungkapkan proses pemikirannya dengan menggunakan kata-kata. Pekerjaan tersebut direkam, dituliskan, dan dianalisis.

3) Observasi pada pekerjaan pakar Pekerjaan dalam bidang tertentu yang dilakukan pakar direkam dan diobservasi

4) Induksi aturan dari contoh

Induksi adalah suatu proses penalaran dari khusus ke umum. Suatu sistem induksi aturan diberi contoh-contoh dari suatu masalah yang hasilnya telah diketahui. Setelah diberikan beberapa contoh, sistem induksi aturan tersebut dapat membuat aturan yang benar untuk kasus-kasus contoh. Selanjutnya aturan dapat digunakan untuk menilai kasus lain yang hasilnya tidak diketahui.

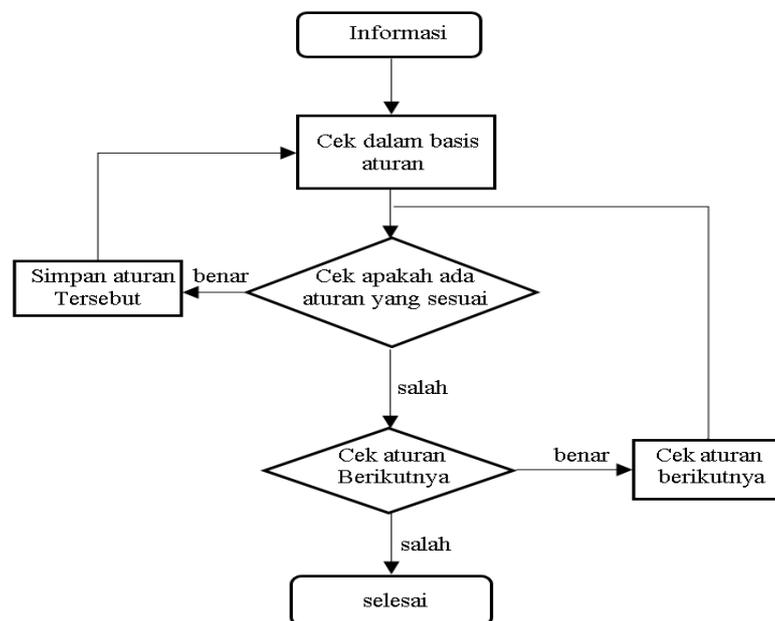
d. Mesin/Motor Inferensi (*Inference Engine*)

Mekanisme inferensi adalah bagian dari sistem pakar yang melakukan penelusuran dengan menggunakan isi daftar aturan berdasarkan urutan pola tertentu. Selama proses konsultasi antar

sistem dan pemakai, mekanisme inferensi menguji aturan satu demi satu sampai kondisi aturan itu benar.

Secara umum ada dua teknik utama yang digunakan dalam mekanisme inferensi untuk pengujian aturan, yaitu penelusuran maju (*forward chaining*) dan penelusuran mundur (*backward chaining*).

Dalam mencari kerusakan mobil akan dimulai dengan memberikan pertanyaan mengenai gangguan yang dialami sehingga diperoleh suatu diagnosa kerusakan dan hasil akhir kesimpulan kerusakan mesin tersebut. Proses pelacakan kedepan (*forward chaining*) pada sistem analisa kerusakan mesin secara umum dapat digambarkan sebagai berikut :



Gambar 2.3 Proses pelacakan ke depan/ Forward Chaining

e. *Workplace / Blackboard*

Workplace merupakan area dari sekumpulan memori kerja (*working memory*), digunakan untuk merekam kejadian yang

sedang berlangsung termasuk keputusan sementara. Ada 3 keputusan yang

dapat direkam :

- 1) Rencana : bagaimana menghadapi masalah
- 2) Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi
- 3) Solusi : calon aksi yang akan dibangkitkan

f. Fasilitas Penjelasan

Adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan :

- 1) mengapa suatu pertanyaan ditanyakan oleh sistem pakar ?
- 2) bagaimana konklusi dicapai ?
- 3) mengapa ada alternatif yang dibatalkan ?
- 4) rencana apa yang digunakan untuk mendapatkan solusi ?

g. Perbaikan Pengetahuan

Pakar memiliki kemampuan untuk menganalisis dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya. Kemampuan tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya dan juga mengevaluasi apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

2. Teori *Certainty Factor*

Menurut (Azhar, 2019) *Certainty Factor* adalah satu metode untuk membuktikan apakah satu fakta itu pasti ataukah tidak pasti yang berbentuk metric biasanya digunakan dalam sistem pakar. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti. Dalam mengekspresikan derajat keyakinan digunakan suatu nilai yang disebut *Certainty Factor* (CF) untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data. Menurut (Adhar, 2018) untuk menghitung CF (keyakinan) dari kesimpulan diperlukan bukti pengkombinasian sebagai berikut:

Rumus untuk menghitung keyakinan.

$$MB(h, e1 \wedge e2) = MB(h, e1) + MB(h, e2) * (1 - MB[h, e1]) \dots \dots \dots (2.1)$$

Rumus untuk menghitung ke tidak yakinan.

$$MD(h, e1 \wedge e2) = MD(h, e1) + MD(h, e2) * (1 - MD[h, e1]) \dots \dots \dots (2.2)$$

Rumus untuk menghitung tingkat keyakinan

$$CF[H, E] = MB[H, E] - MD[H, E] \dots \dots \dots (2.3)$$

Keterangan:

CF : *Certainty Factor* (Faktor Kepastian) dalam Hipotesis H yang berpengaruh oleh Fakta E.

MB : *Measure of Belief* (tingkat keyakinan), adalah ukuran kenaikan dari kepercayaan hipotesis H dipengaruhi oleh Fakta E.

MD : *Measure of Disbelief* (tingkat ketidakpercayaan), adalah kenaikan dari ketidakpercayaan hipotesis dipengaruhi fakta E.

E : *Evidence* (Peristiwa atau fakta).

H : Hipotesis (Dugaan)

Dimana nilai-nilai CF, MB, dan MD adalah seperti dibawah ini

- a. $CF[H,E]$ = *Certainty Factor* dari hipotesis H yang dipengaruhi oleh gejala (*evidence*) E. Besarnya CF berkisar antara -1 sampai dengan 1
Nilai -1 menunjukkan ketidakpercayaan mutlak, sedangkan 1 menunjukkan mutlak.
- b. $MB[H,E]$ = ukuran kepercayaan / tingkat keyakinan (*measure of increased belief*) terhadap hipotesis H, jika diberikan / dipengaruhi oleh *evidence* (gejala) E (besarnya berkisar antara 0 dan 1).
- c. $MD[H,E]$ = ukuran ketidakpercayaan / tingkat ketidakyakinan (*measure of increased disbelief*) terhadap hipotesis H, jika diberikan / diperbaharui *evidence* E (besarnya berkisar antara 0 dan 1).

Menurut (Yusman et al., 2017), *Certainty theory* ini diusulkan oleh Shortliffe dan Buchanan pada tahun 1975 untuk mengakomodasi ketidakpastian pemikiran (*inexactreasoning*) seorang pakar, guna menggambarkan tingkat keyakinan pakar terhadap masalah yang sedang dihadapi. Faktor kepastian ini juga berguna untuk mengatasi ketidakpastian dalam menentukan kerusakan yang mempunyai gejala yang sama. Dengan ukuran kepercayaan dan ketidakpercayaan suatu gejala terhadap kerusakan yang sama, maka dapat didapat suatu nilai kepastian/faktor kepastian (CF). Nilai yang digunakan sebagai level kepercayaan pada suatu hipotesis yang memberikan suatu informasi yang tersedia adalah *certainty factor*. Disajikan pada tabel 2.1

Tabel 2.1 *Value Interpretation Certainty Factor*
 Sumber:(Yusman et al., 2017)

Kepercayaan	CF
Tidak pasti	-1.0
Hampir tidak pasti	-0,8
Kemungkinan tidak	-0.6
Mungkin Tidak	-0.4
Tidak tahu	-0.2 to 0.2
Mungkin	0.4
Kemungkinan benar	0.6
Hampir pasti	0.8
Pasti	1

3. Metode *Forward Chaining*

Strategi pencarian yang memulai proses pencarian dari sekumpulan data atau fakta, dari data-data tersebut dicari suatu kesimpulan yang menjadi solusi dari permasalahan yang dihadapi. Cara kerja dari metode ini adalah mesin inferensi atau *inference engine* menyalakan atau memilih rule-rule dimana bagian premisnya cocok dengan informasi yang ada pada bagian *working memory* (Satwika, 2012). Aktivitas sistem dilakukan berdasarkan siklus mengenal-beraksi. Pertamata, sistem mencari semua aturan yang kondisinya terdapat di memori kerja kemudian memilih salah satunya dan menjalankan aksi yang bersesuaian dengan aturan tersebut. Pemilihan aturan yang akan dijalankan berdasarkan strategi tetap yang disebut strategi penyelesaian konflik. Aksi tersebut menghasilkan memori kerja baru dan siklus diulangi lagi sampai tidak ada aturan yang dapat dipicu, atau tujuan yang dikehendaki sudah terpenuhi. Disajikan pada tabel 2.2

Tabel 2.2 Contoh aturan menggunakan penalaran *forward chaining*
 Sumber: (Purbo, 2019)

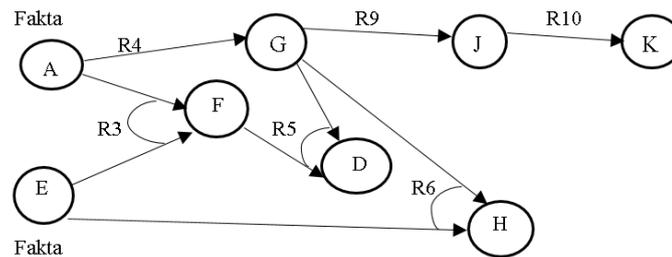
No	Aturan
R1	IF A & B THEN C
R2	IF C THEN D
R3	IF A & E THEN F
R4	IF A THEN G
R5	IF F & G THEN D
R6	IF G & E THEN H
R7	IF C & H THEN I
R8	IF I & A THEN J
R9	IF G THEN J
R10	IF J THEN K

Pada Tabel 2.2 terlihat ada 10 aturan yang tersimpan dalam basis pengetahuan. Jika fakta awal yang diberikan hanya: A dan F (artinya: A dan F bernilai benar). Ingin dibuktikan apakah K bernilai benar (hipotesis: K). Langkah-langkah inferensi adalah sebagai berikut:

- a. Dimulai dari R-1, A merupakan fakta sehingga bernilai benar, sedangkan B belum bisa diketahui kebenarannya, sehingga C pun juga belum bisa diketahui kebenarannya. Oleh karena itu tidak didapatkan informasi apapun pada R1 ini. Sehingga kita menuju ke R2.
- b. Pada R2 tidak diketahui informasi apapun tentang C, sehingga tidak bias dipastikan kebenaran D. Oleh karena itu tidak didapatkan informasi apapun pada R1 ini. Sehingga harus menuju ke R3.
- c. Pada R3, baik A maupun E adalah fakta sehingga jelas benar. Dengan demikian F sebagai konsekuen juga ikut benar. Sehingga sekarang terdapat fakta baru yaitu F. Karena F bukan hipotesis yang hendak dibuktikan (= K) maka penelusuran dilanjutkan ke R4.
- d. Pada R4, A adalah fakta sehingga jelas benar. Dengan demikian G sebagai konsekuen juga ikut benar. Sehingga sekarang didapatkan

fakta baru yaitu G. Karena G bukan hipotesis yang hendak dibuktikan ($= K$), maka penelusuran dilanjutkan ke R5.

- e. Pada R5, baik F maupun G bernilai benar berdasarkan aturan R3 dan R4. Dengan demikian G sebagai konsekuen juga ikut benar. Sehingga sekarang terdapat fakta baru yaitu D. Karena D bukan hipotesis yang hendak dibuktikan, maka penelusuran dilanjutkan ke R6.
- f. Pada R6, baik A maupun G adalah benar berdasarkan fakta dari R4. Dengan demikian H sebagai konsekuen juga ikut benar. Sehingga sekarang terdapat fakta baru yaitu H. Karena H bukan hipotesis yang hendak dibuktikan, maka penelusuran dilanjutkan ke R7.
- g. Pada R7, meskipun H benar berdasarkan R6, namun tidak diketahui kebenaran C sehingga, I pun juga belum bisa diketahui kebenarannya. Oleh karena itu tidak didapatkan informasi apapun pada R7 ini. Sehingga dilanjutkan menuju ke R8.
- h. Pada R8, meskipun A benar karena fakta, namun tidak diketahui kebenaran I, sehingga J pun juga belum bisa diketahui kebenarannya, oleh karena itu tidak didapatkan informasi apapun pada R8 ini. Sehingga dilanjutkan menuju ke R9.
- i. Pada R9, J bernilai benar karena G benar berdasarkan R4. Karena J bukan hipotesis yang hendak dibuktikan, maka penelusuran dilanjutkan ke R10.
- j. Pada R10, K bernilai benar karena J benar berdasarkan R9. Karena K sudah merupakan hipotesis yang hendak dibuktikan, maka terbukti bahwa K adalah benar.



Gambar 2.2 Alur *Forward Chaining* Sumber: (Purbo, 2019)

Kelebihan dari metode runut maju ini diantaranya adalah :

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

4. Hypertext Preprocessor (PHP)

PHP memiliki kepanjangan Hypertext Preprocessor. PHP telah menjadi salah satu bahasa pemrograman yang paling kuat untuk mengembangkan aplikasi web (Arrhioui et al., 2017). PHP merupakan bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. (Fitri Ayu, 2018).

5. My Structured Query Language (MYSQL)

MySQL adalah salah satu jenis database server yang sangat terkenal. kepopulerannya disebabkan Mysql menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Mysql termasuk jenis RDBMS (Relational Database Management System). Pada Mysql, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

Untuk mengelola database Mysql ada beberapa cara yaitu melalui prompt DOS (tool command line). MySQL didistribusikan gratis dibawah lisensi GPL (*General Public License*). Dimana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan *closed source* atau komersial (Maulana, 2016).

6. Unified Modeling Language (UML)

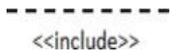
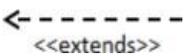
Menurut (Hendini, 2016) *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek. UML meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem. Diagram Unified Modelling Language (UML) antara lain sebagai berikut (Suendri, 2018):

A. Usecase Diagram

Usecase Diagram menyatakan visualisasi interaksi yang terjadi antara pengguna (aktor) dengan sistem. Diagram ini bisa menjadi gambaran yang bagus untuk menjelaskan konteks dari sebuah sistem sehingga terlihat jelas batasan dari sistem (Kurniawan, 2018):

Tabel 2.3 Notasi *Usecase Diagram*

Simbol	Keterangan
	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan <i>actor</i>

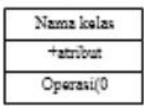
	Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem
	Asosiasi antara aktor dan <i>usecase</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung
	Asosiasi antara aktor dan <i>usecase</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
	<i>Include</i> , merupakan di dalam <i>usecase</i> lain (<i>required</i>) atau pemanggilan <i>usecase</i> oleh <i>usecase</i> lain
	<i>Extend</i> , merupakan perluasan dari <i>usecase</i> lain jika kondisi atau syarat terpenuhi

Sumber : (simatupang julianto, 2019)

b. *Class Diagram*

Class diagram merupakan suatu set objek yang memiliki atribut yang sama, kelas kadang disebut kelas objek (Suendri, 2018):

Tabel 2.4 Notasi *Class Diagram*

Simbol	Keterangan
	Kelas pada struktur sistem
	Konsep <i>interface</i> dalam pemrograman berorientasi objek
	Kelas antar kelas dengan makna umum asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	Kelas antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

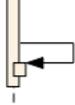
	Kelas antar kelas dengan makna <i>generalisasi-spesialisasi</i> (umum khusus)
	Kelas antar kelas dengan makna kebergantungan antar kelas
	Kelas antar kelas dengan makna semua bagian (<i>whole-part</i>)

Sumber : (simatupang julianto, 2019)

c. *Squency Diagram*

Squency Diagram merupakan gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram* (Suendri, 2018):

Tabel 2.4 Notasi *Squency Diagram*

Simbol	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data
	<i>Boundary Class</i> , berisi kumpulan class yang menjadi interface atau interaksi antar satu atau lebih aktor dengan sistem
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas
	<i>Message</i> , simbol mengirim pesan antar class
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi

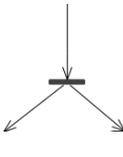
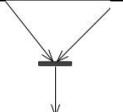
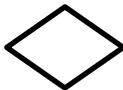
-	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat activation
---	---

Sumber : (simatupang julianto, 2019)

d. *Activity Diagram*

Activity Diagram merupakan aktivitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing-masing aksi tersebut dimulai, keputusan yang mungkin terjadi hingga berakhirnya aksi (Suendri, 2018).

Tabel 2.5 Notasi *Activity Diagram*

Simbol	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	<i>End Point</i> , akhir aktivitas
	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis
	<i>Fork/</i> percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan parallel menjadi satu
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi
	<i>Decision Points</i> , menggambar kan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>
	<i>Swimlane</i> , pembagian activity diagram untuk menunjukkan siapa melakukan apa

Sumber : (simatupang julianto, 2019)

C. Landasan Teori

Berdasarkan hasil analisa dari penelitian relevan diatas dan penjelasan dari variabel – variabel yang berkaitan dengan penelitian ini, maka

Implementasi sistem pakar banyak digunakan dalam bidang teknologi karena sistem pakar dipandang sebagai cara penyimpanan pengetahuan pakar pada bidang tertentu dalam program komputer sehingga keputusan dapat diberikan dalam melakukan penalaran secara cerdas. Untuk membantu optimalisasi kerja daripada manusia dalam memberikan solusi dan cara mengetahui kerusakan mobil nissan grand livina dengan ini peneliti mengambil judul “Sistem Pakar Identifikasi Kerusakan Mobil Nissan Grand Livina Menggunakan Metode *Forward Chaining*”.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

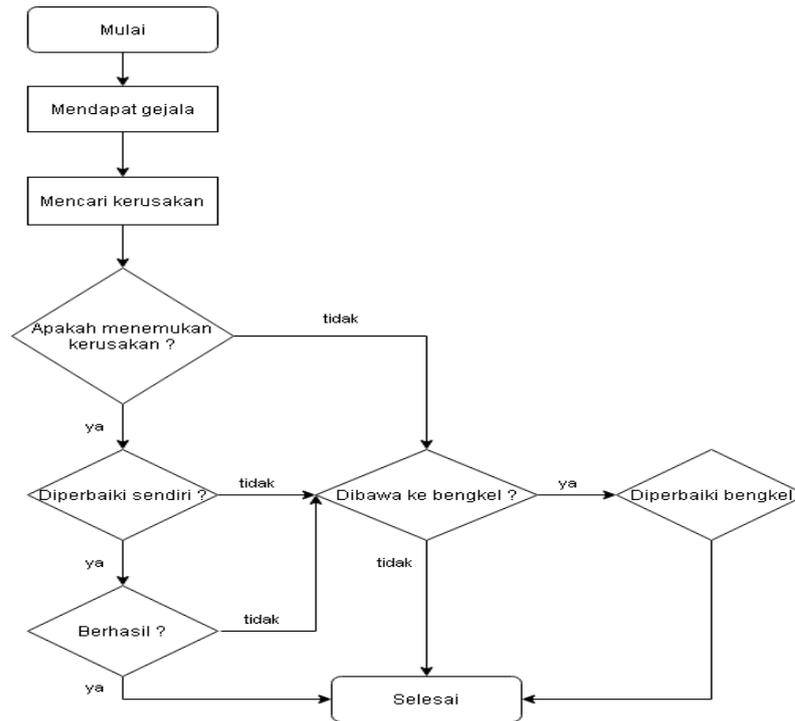
A. Analisis sistem

1. Analisis Sistem Berjalan

Sistem yang berjalan saat ini untuk penyelesaian masalah kerusakan mesin mobil nissan grand livina yang digunakan *user* sebagai berikut :

- a. *User* mendapat informasi gejala yang terjadi dengan menanyakan ke bengkel atau ahli mesin.
- b. *User* mencari kerusakan yang sesuai dengan kondisi mobil.
- c. Jika *user* menemukan kerusakan sesuai dengan solusi yang diberikan tetapi tidak dapat memperbaiki kerusakan sendiri maka disarankan untuk membawa ke bengkel.

Prosedur diatas dapat digambarkan dalam bentuk *flowchart* disajikan pada gambar 3.1.



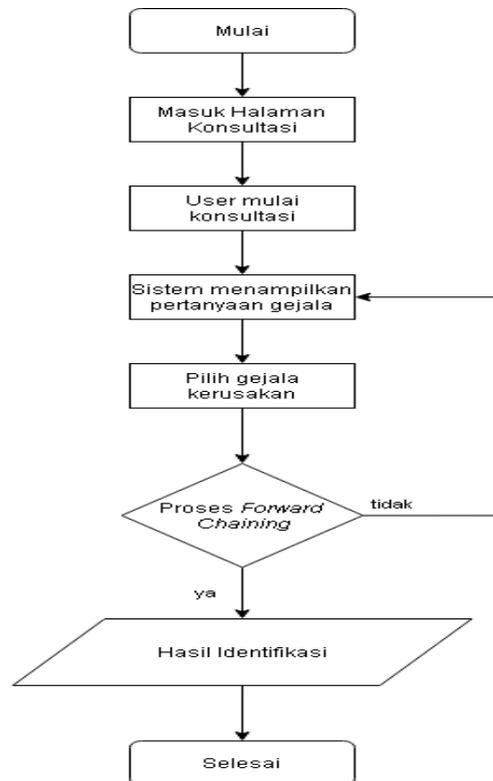
Gambar 3.1 *flowchart* sistem berjalan

2. Analisis Sistem Yang Diajukan

Untuk mengatasi masalah yang ada, penelitian ini mengajukan sebuah sistem pakar kerusakan mobil nissan grand livina dengan prosedur sistem sebagai berikut :

- a. *User* masuk halaman konsultasi melalui *web browser*.
- b. *User* mulai memilih gejala yang sesuai dengan yang dialami.
- c. Sistem akan menampilkan setiap pertanyaan gejala sesuai basis pengetahuan yang telah tersimpan dan menggunakan mesin inferensi *forward chaining*.
- d. Jika ditemukan hasil kerusakan, sistem akan berhenti dan menampilkan data kerusakan yang dialami beserta solusi dari kerusakan tersebut.

Prosedur diatas dapat digambarkan dalam bentuk *flowchart* pada gambar 3.2.



Gambar 3.2. *flowchart* sistem yang diajukan

3. Analisis Pengumpulan Data

Adapun metode pengumpulan data penelitian ini diantaranya, yaitu:

- a. Observasi: Langkah pengamatan atau observasi ini dilakukan untuk mengamati dan mempelajari kondisi kegiatan dalam identifikasi kerusakan mobil nissan grand livina secara langsung.
- b. Wawancara: Pada penelitian ini melakukan tanya jawab kepada pihak yang bersangkutan di saat melakukan penelitian yakni Bapak Dr. Budi Waluyo, ST., M.T. selaku akademisi dan kaprodi Teknik Otomotif

Universitas Muhammadiyah Magelang dan Bapak Gaga Putra Setiawan A.Md, selaku pemilik bengkel dan montir.

- c. Studi Literatur: Studi ini dilakukan dengan mengumpulkan informasi atau data sebanyak-banyaknya dari kepustakaan dalam bentuk buku panduan nissan grand livina, jurnal, skripsi, artikel serta sumber-sumber lain yang berhubungan dengan penelitian.

4. Analisis Kebutuhan Fungsional

Fungsi yang diharapkan pada sistem pakar ini adalah:

- a. Menyimpan dan menampilkan gejala kerusakan pada mobil sebagai dasar analisis kerusakan mobil nissan grand livina.
- b. Menyimpan dan menampilkan solusi perbaikan untuk mengatasi kerusakan mobil nissan grand livina berdasarkan jenis kerusakan yang terjadi.

5. Analisis Sistem Pakar Dengan Metode *Forward Chaining*

Menurut (Nasir & Gultom, 2018) , variabel penelitian pada dasarnya adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi dan kesimpulannya. Variabel harus didefinisikan secara operasional agar lebih mudah dicari hubungannya antara satu variabel dengan lainnya dan pengukurannya. Adapun manfaat operasional variabel antara lain untuk mengidentifikasi kriteria yang sedang didefinisikan. Dalam penelitian ini operasional variabel yang digunakan adalah kerusakan mobil nissan grand livina.

Berikut adalah data gejala hasil wawancara dengan Bapak Gaga Putra Setiawan A.md selaku pemilik Bengkel sekaligus Mekanik. Disajikan pada tabel 3. 1

Tabel 3.1 Data Gejala

Kode	Gejala
G1	Apakah lampu indikator-indikator pada dashboard tidak mau menyala ketika kontak sudah ON ?
G2	Apakah lampu indikator check engine mati ketika kontak ON ?
G3	Apakah tidak bisa distarter dalam kondisi mesin sudah panas ?
G4	Apakah saat mesin dingin susah dihidupkan ?
G5	Apakah mesin mendadak mau mati saat pedal gas diinjak ?
G6	Apakah mesin mati setelah melepas pedal gas ?
G7	Apakah terdengar bunyi gluduk saat lepas gas?
G8	Apakah tidak bisa digas ?
G9	Apakah mesin pincang saat di tanjakan ?
G10	Apakah mesin menjadi pincang saat sudah panas ?
G11	Apakah mesin cepat panas ketika AC hidup ?
G12	Apakah mesin mati setelah AC dihidupkan ?
G13	Apakah indikator temperature naik atau mesin mudah panas ?
G14	Apakah tidak ada hembusan angin pada AC?
G15	Apakah kipas ac tidak berputar?
G16	Apakah ac tiba tiba panas?
G17	Apakah oli bercampur air ?
G18	Apakah oli mesin berkurang ?
G19	Apakah knalpot mengeluarkan asap putih ?
G20	Apakah tercium bau menyengat pada knalpot ?
G21	Apakah bau terbakar di area transmisi?
G22	Apakah bau mesin bau menyengat?
G23	Apakah idle tinggi ?
G24	Apakah konsumsi BBM boros ?
G25	Apakah pedal rem terasa keras ?
G26	Apakah pedal rem terasa dalam ?
G27	Apakah saat di hidupkan mesin, pedal rem ketika di injak tidak merendah G1 ?
G28	Apakah kuantitas minyak rem pada resevoir menurun ?
G29	Apakah terdengar bunyi mendesis ketika pedal rem di injak ?
G30	Apakah rem mobil terasa bergetar ?

Berikut adalah data kerusakan beserta solusi didapatkan dari hasil wawancara dengan Bapak Gaga Putra Setiawan A.md. disajikan pada tabel 3. 2

Tabel 3.2 Data Kerusakan dan Solusi

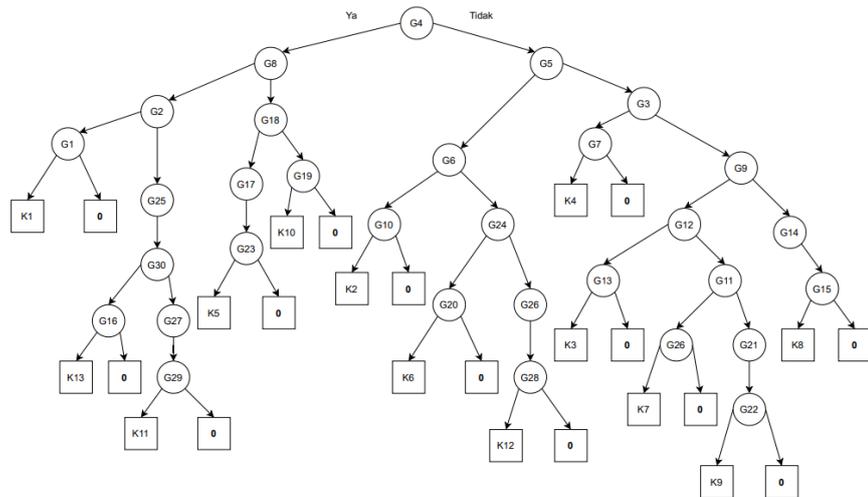
Kode	Kerusakan	Solusi
K1		a. Periksa lampu indikator aki/baterai (pastikan bahwa masih bagus) b. Bersihkan pull pull aki dari jamur/kerak dan kencangkan apabila kendur c. Periksa fuse (putus/tidak)
K2	Fuel pump	Membersihkan atau mengganti pompa bahan bakar yang baru
K3	Thermostat	Kita tidak dapat memperbaiki, disarankan untuk pergi ke bengkel
K4	Ignition coil	Kita tidak bisa memeriksa ignition coil, jika tidak ingin repot maka disarankan untuk ke bengkel
K5	APP sensor	Kita hanya bisa melepas dan membersihkan soket pedal gas jika masih tidak efek segera bawa ke bengkel
K6	O2 sensor	Kita tidak bisa mengerjakan sendiri karena membutuhkan alat scan
K7	Brake master cylinder	a. Bukalah tutup tampungan oli b. Coba cek kuantitas oli rem c. Jika oli rem lower atau keruh maka disarankan untuk mengganti.
K8	Kerusakan AC/ freon habis	a. jika freon habis isi ulang freon b. mengganti seal dengan yang baru atau membersihkan c. periksa sekering, kabel, konektor, kelistrikan dan socket kabel.
K9	Mesin Overheat	a. buka kap mesin agar panas cepat menyingkir b. periksa tabung air radiator dan isi jika perlu tetapi jangan buka tutup radiator bila masih mesin panas c. cari kebocoran pada sistem pendingin d. jika masih overheat segera hubungi bengkel agar tidak terlalu parah
K10	Ring piston	Kita tidak bisa memperbaiki sendiri disarankan untuk pergi ke bengkel
K11	Malfungsi booster rem	a. periksa perubahan pada stasioner mesin, jika ada kebocoran stasioner mesin naik b. periksa kevakuman manifold menuju boster, jika komponen tidak bisa menghisap maka karet membran mengalami kebocoran c. periksa selang vakum dan tabung boster.pastikan ada kebocoran atau tidak
K12	Kampas rem tipis	Tidak dapat bertahan lama jika sudah tipis, maka harus segera ganti dengan yang baru.
K13	Disk brake	Kita tidak dapat memperbaiki, bawalah ke bengkel

6. Rule Basis Pengetahuan Pakar

Rule basis sistem merupakan salah satu komponen yang ada dalam sistem pakar. Aturan tersebut berbentuk *IF-THEN*. Rule basis sistem yaitu cara untuk menyimpan dan memanipulasi pengetahuan untuk

No	Kode	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13
24	G24						*						*	
25	G25											*		*
26	G26							*					*	
27	G27											*		
28	G28												*	
29	G29											*	*	
30	G30											*		*

Berdasarkan tabel keputusan tersebut maka pohon keputusannya dapat dilihat pada Gambar 3.3 sebagai berikut:



Gambar 3.3 Pohon Keputusan

Metode penalaran yang digunakan pada sistem pakar ini menggunakan penalaran maju atau *forward chaining* dimana penalaran dilakukan dimulai dari fakta awal menuju tujuan akhir.

Salah satu cara untuk merepresentasikan penalaran tersebut agar lebih mudah digunakan adalah dengan pohon keputusan (*decision tree*). Pembuatan pohon keputusan digunakan untuk menyederhanakan proses akuisisi pengetahuan supaya lebih mudah diubah dalam bentuk kaidah atau aturan. Konsep dari pohon keputusan adalah mengubah data menjadi pohon keputusan dan aturan-aturan keputusan.

Jika pengguna menginputkan jawaban YA sampai di tahap akhir maka sistem akan menampilkan data kerusakan dan jika pengguna menginputkan jawaban TIDAK sampai di tahap akhir maka sistem akan menampilkan tidak ditemukan jawaban.

7. *Certainty Factor*

Pada tabel 3.6 berikut ini berisi nilai bobot dari gejala yang diperoleh dari hasil wawancara dengan Bapak Gaga Putra Setyawan A.Md selaku pemilik bengkel dan mekanik.

Tabel 3.6 Tabel nilai bobot

Kode	Gejala	MB	MD
G1	Apakah lampu indikator-indikator pada dashboard tidak mau menyala ketika kontak sudah ON ?	0,9	0,04
G2	Apakah lampu indikator check engine mati ketika kontak ON ?	0,8	0,02
G3	Apakah tidak bisa distarter dalam kondisi mesin sudah panas ?	0,7	0,03
G4	Apakah saat mesin dingin susah dihidupkan ?	0,9	0,01
G5	Apakah mesin mendadak mau mati saat pedal gas diinjak ?	0,8	0,05
G6	Apakah mesin mati setelah melepas pedal gas ?	0,7	0,03
G7	Apakah terdengar bunyi gluduk saat lepas gas?	0,8	0,02
G8	Apakah tidak bisa digas ?	0,7	0,01
G9	Apakah mesin pincang saat di tanjakan ?	0,8	0,03
G10	Apakah mesin menjadi pincang saat sudah panas ?	0,8	0,04
G11	Apakah mesin cepat panas ketika AC hidup ?	0,9	0,05
G12	Apakah mesin mati setelah AC dihidupkan ?	0,8	0,03
G13	Apakah indikator temperature naik atau mesin mudah panas ?	0,9	0,01
G14	Apakah tidak ada hembusan angin pada AC?	0,8	0,03
G15	Apakah kipas ac tidak berputar?	0,8	0,02
G16	Apakah ac tiba tiba panas?	0,7	0,02
G17	Apakah oli bercampur air ?	0,7	0,03
G18	Apakah oli mesin berkurang ?	0,9	0,01
G19	Apakah knalpot mengeluarkan asap putih ?	0,8	0,02
G20	Apakah tercium bau menyengat pada knalpot ?	0,9	0,04
G21	Apakah bau terbakar di area transmisi?	0,9	0,03
G22	Apakah bau mesin bau menyengat?	0,8	0,02
G23	Apakah idle tinggi ?	0,7	0,01
G24	Apakah konsumsi BBM boros ?	0,8	0,02
G25	Apakah pedal rem terasa keras	0,8	0,04
G26	Apakah pedal rem terasa dalam	0,8	0,02
G27	Apakah saat di hidupkan mesin, pedal rem ketika di injak tidak merendah G1	0,7	0,04
G28	Apakah kuantitas minyak rem pada resevoir menurun	0,9	0,01
G29	Apakah terdengar bunyi mendesis ketika pedal rem di injak	0,8	0,04
G30	Apakah rem mobil terasa bergetar?	0,9	0,01

Untuk menghitung keyakinan diambil dari tabel 2.1 Berikut perhitungannya ;

$$\begin{aligned}
 1). \text{MB1} &= \text{MB1} + \text{MB2} * (1 - \text{MB1}) \\
 &= 0,9 + 0,8 * (1 - 0,9) \\
 &= 0,9 + 0,08 \\
 &= 0,98
 \end{aligned}$$

$$\begin{aligned}
 \text{MB2} &= \text{MB1} + \text{MB4} * (1 - \text{MB1}) \\
 &= 0,98 + 0,9 * (1 - 0,98) \\
 &= 0,98 + 0,018 \\
 &= 0,998
 \end{aligned}$$

$$\begin{aligned}
 \text{MB3} &= \text{MB1} + \text{MB8} * (1 - \text{MB1}) \\
 &= 0,998 + 0,7 * (1 - 0,998) \\
 &= 0,998 + 0,0014 \\
 &= 0,9994
 \end{aligned}$$

Untuk menghitung ketidakyakinan diambil dari tabel 2.2 Berikut perhitungannya ;

$$\begin{aligned}
 1). \text{MD1} &= \text{MD1} + \text{MD2} * (1 - \text{MD1}) \\
 &= 0,04 + 0,02 * (1 - 0,04) \\
 &= 0,04 + 0,0192 \\
 &= 0,0592
 \end{aligned}$$

$$\begin{aligned}
 \text{MD1} &= \text{MD1} + \text{MD4} * (1 - \text{MD1}) \\
 &= 0,0592 + 0,01 * (1 - 0,0592) \\
 &= 0,0592 + 0,009408 \\
 &= 0,068608
 \end{aligned}$$

$$\begin{aligned}
 MD1 &= MD1 + MD8 * (1 - MD1) \\
 &= 0,068608 + 0,01 * (1 - 0,068608) \\
 &= 0,068608 + 0,00931392 \\
 &= 0,07792192
 \end{aligned}$$

Untuk mengetahui CF (keyakinan) kerusakan, diambil dari tabel 2.3

Berikut perhitungannya ;

$$\begin{aligned}
 1). CF1 &= MB - MD \\
 &= 0,9994 - 0,07792192 \\
 &= 0,92147808 * 100 \\
 CF &= 92,147\%
 \end{aligned}$$

B. User Interface

Berdasarkan analisis sistem yang diajukan maka akan dibuat perancangan sistem secara keseluruhan dengan menggunakan UML (*Unified Modelling Language*) sebagai gambaran alur sistem dan EER (*Enhanced Entity Relationship*) sebagai perancangan database. Perancangan sistem ini merupakan gambaran keseluruhan dari hasil akhir dari sistem yang diajukan.

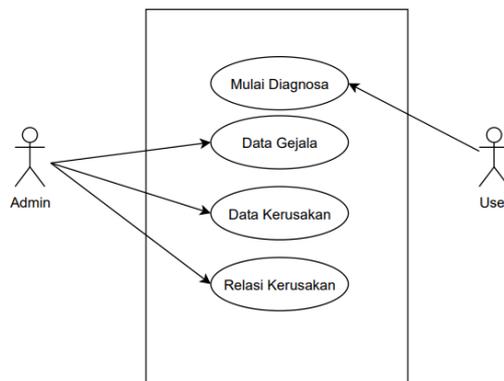
1. Perancangan Sistem

Dalam membuat desain sistem, perancang memanfaatkan UML karena merupakan sebuah standar penulisan dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik. Berikut diagram UML yang akan digunakan dalam pengembangan sistem, yaitu :

a. *Use Case*

Merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat *actor* yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem.

1. Pada gambar 3.4 *use case diagram* dibawah ini yang menggambarkan interaksi yang terjadi. Terdapat 2 aktor pada sistem yaitu: User dan Admin. User dapat melakukan konsultasi serta dapat melihat hasil identifikasi yang telah diisi. Admin dapat menambahkan kerusakan, gejala, solusi, nilai *CF* dan hasil diagnosa.



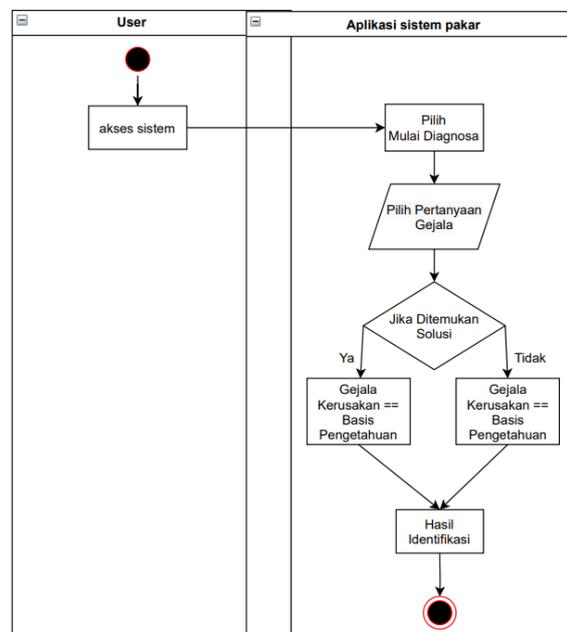
Gambar 3.4 Usecase Diagram

b. *Activity Diagram*

Activity diagram digunakan untuk memodelkan langkah-langkah proses atau kegiatan sistem atau menggambarkan aktifitas yang dilakukan oleh user pada sistem pakar identifikasi kerusakan mobil nissan grand livina. Pada *Activity Diagram* ini meliputi rancangan *Activity Diagram* User, Data Gejala, Data Kerusakan dan Relasi Kerusakan

1. Activity Diagram User

Pada gambar 3.5 *Activity Diagram User* dibawah ini menggambarkan interaksi antara user dengan sistem. User memilih menu mulai diagnose dan sistem akan menampilkan pertanyaan gejala, dimana user diminta untuk memilih jawaban sesuai dengan kondisi mobilnya. Apabila telah dipilih user dapat melihat hasil identifikasi kerusakan yang terjadi sesuai gejala yang dipilih.

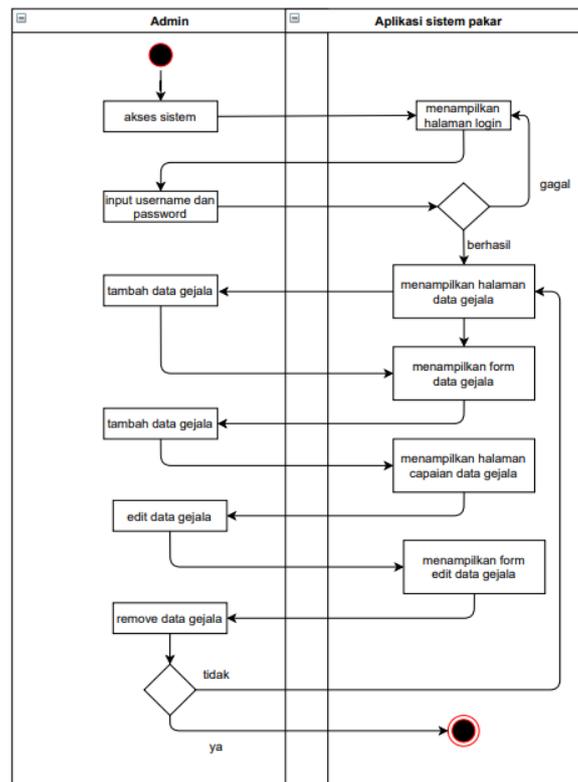


Gambar 3. 5 *Activity Diagram User*

2. Activity Diagram Data Gejala

Pada gambar 3.6 *Activity Diagram Data Gejala* dibawah ini merupakan aktifitas untuk pemilihan menu data gejala. Setelah memilih menu tambah data akan muncul *form* tambah gejala baru yang berisi kode dan nama gejala. Dimana setelah menambahkan gejala baru maka sistem akan menyimpan di database dan sistem akan

menampilkannya pada data gejala. Jika ada penulisan salah pada saat menginputkan data gejala dan sudah tersimpan maka pilih menu edit untuk membenarkan. Jika data gejala yang sudah diinput tidak sesuai maka pilihlah menu remove untuk menghapus.

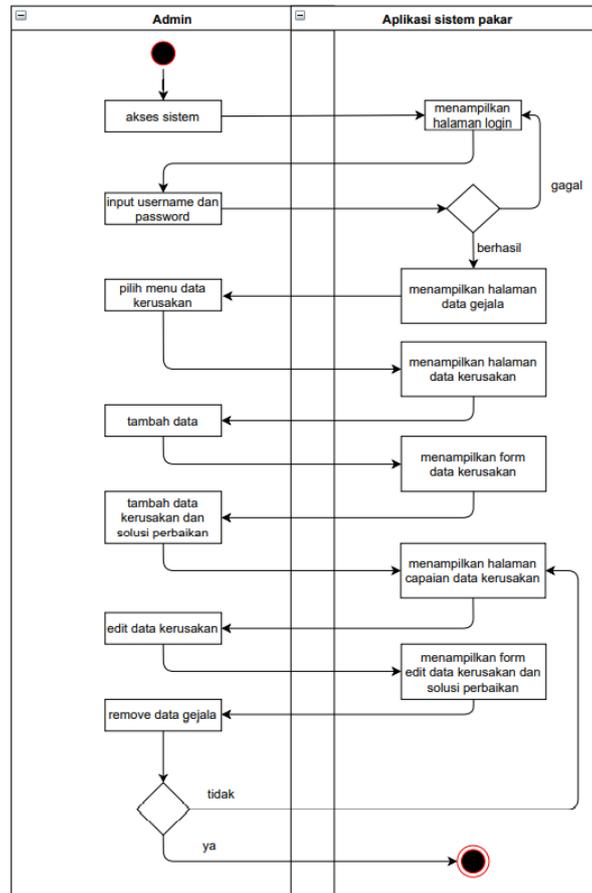


Gambar 3. 6 *Activity Diagram* Data Gejala

3. *Activity Diagram* Data Kerusakan

Pada gambar 3.7 *Activity Diagram* Data Kerusakan dibawah ini merupakan aktifitas untuk pemilihan menu data kerusakan. Setelah memilih menu tambah data akan muncul *form* tambah kerusakan baru yang berisi kode, nama kerusakan dan solusi perbaikan. Dimana setelah menambahkan kerusakan baru maka sistem akan menyimpan di database dan sistem akan menampilkannya pada data kerusakan. Jika ada penulisan salah pada saat menginputkan data kerusakan dan

sudah tersimpan maka pilih menu edit untuk membenarkan. Jika data kerusakan yang sudah diinput tidak sesuai maka pilihlah menu remove untuk menghapus.

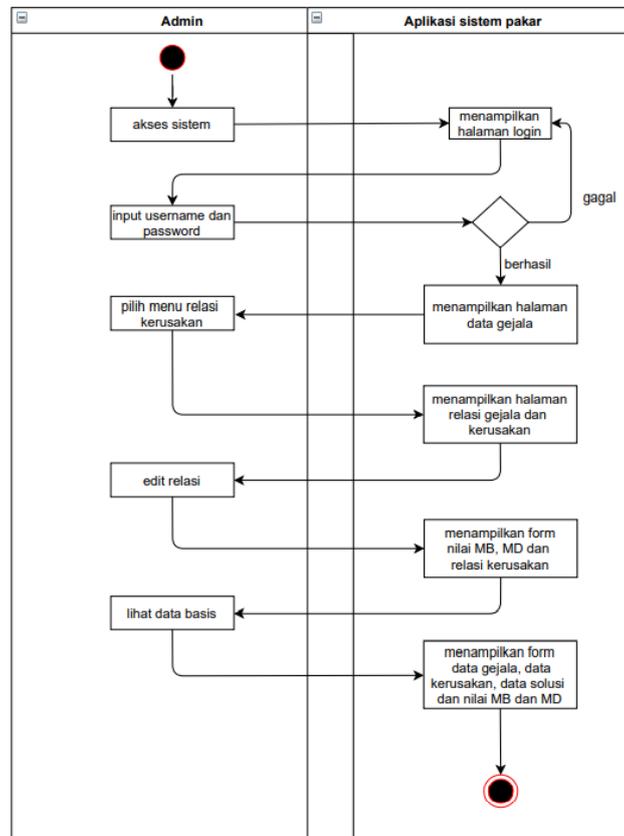


Gambar 3.7 Activity Diagram Data Kerusakan

4. Activity Diagram Relasi Kerusakan

Pada gambar 3.8 Activity Diagram Relasi Kerusakan dibawah ini merupakan aktifitas untuk pemilihan menu relasi kerusakan dan inti dari sistem ini untuk data akan diolah secara *forward chaining* dan *certainty factor*. Pada menu ini pilih edit, akan muncul *form* edit relasi yang berisi nama gejala, penilaian MB dan MD serta kerusakan yang berhubungan dengan gejala. Dimana setelah edit relasi maka sistem

akan menyimpan di database dan sistem akan menampilkannya pada relasi kerusakan.



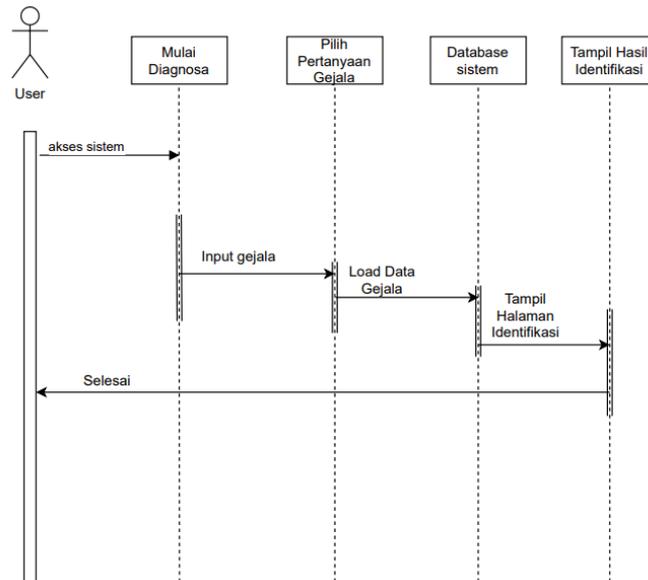
Gambar 3.8 Activity Diagram Relasi Kerusakan

c. Sequence Diagram

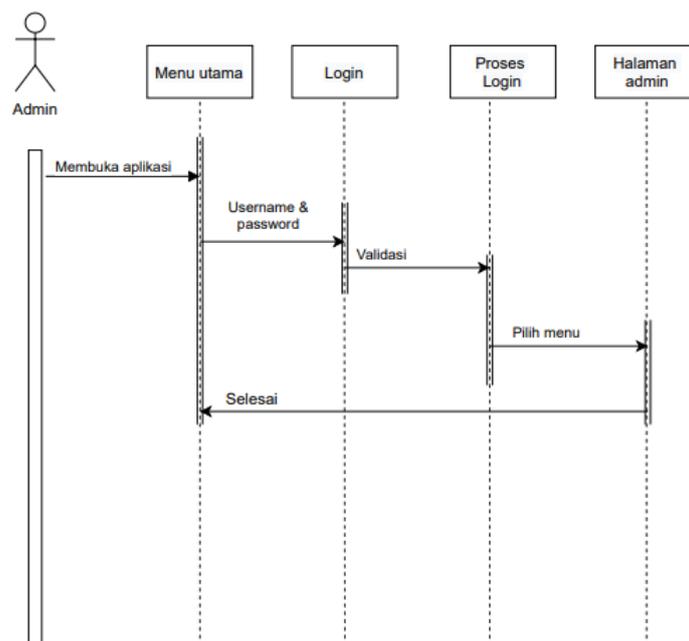
Pada Diagram ini akan menggambarkan pesan yang dilakukan maupun diterima oleh aktor pada sebuah *lifeline*. *Lifeline* setiap aktor bisa berbeda sesuai kebutuhan proses kegiatan yang harus dilakukan aktor terhadap sistem. Diagram ini menggambarkan kegiatan atau proses berjalannya komponen dalam sistem (interaksi pengguna dengan sistem) untuk menghasilkan suatu output tertentu.

1) Diagram Squence User

Pada gambar 3.7 menjelaskan, user masuk ke menu utama, kemudian sistem akan menampilkan pertanyaan gejala. jika sudah mengisi pertanyaan maka akan masuk ke halaman hasil identifikasi.



Gambar 3. 7 Squence Diagram User

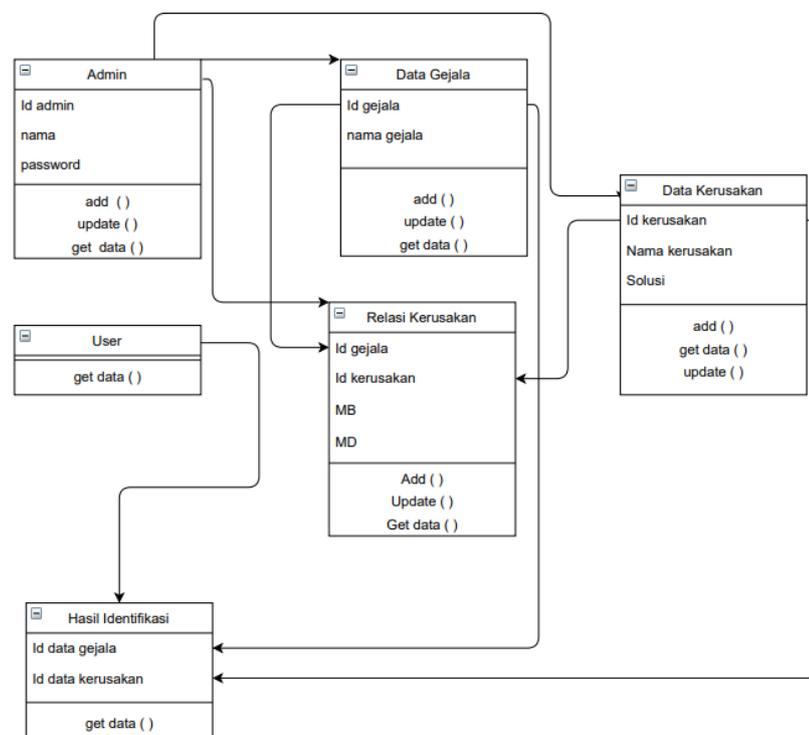


Gambar 3. 8 Squence Diagram Admin

Pada gambar 3.8 menjelaskan, admin membuka aplikasi kemudian login dan memasukkan username serta password, lalu sistem menampilkan menu admin. Setelah itu memilih menu untuk menambahkan gejala, kerusakan, solusi dan basis pengetahuan.

a. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem :



Gambar 3. 9 *Class Diagram*

Pada gambar 3.9 adalah *class diagram* pada perancangan sistem. Terdapat 8 kelas yaitu Admin, User, Gejala, Kerusakan, Solusi dan Basis Pengetahuan.

2. Desain *Interface*

Desain interface dirancang untuk membuat tampilan sebuah aplikasi yang akan dibuat sesuai dengan analisis kebutuhan. *Interface* yang akan dirancang sebagai berikut:

a. Halaman Utama

Halaman utama terdapat menu mulai diagnosa dan menu login admin. menu mulai diagnosa digunakan untuk masuk ke halaman pertanyaan gejala dan halaman login admin digunakan admin untuk melihat, menambah dan menghapus data. Halaman ini disajikan pada Gambar 3.11 dan halaman login admin 3.12 dibawah ini



Gambar 3. 11 Halaman Utama

b. Halaman Login Admin

Halaman login admin merupakan proses masuk ke dalam sistem, dalam halaman login, admin yang terlibat dalam sistem diminta untuk memasukkan *username* dan *password* untuk dapat masuk kedalam sistem dan menjalankan fungsi yang ada pada sistem. Halaman ini disajikan pada Gambar 3.13 dibawah ini

Selamat Datang

Silahkan Masukkan Alamat Email dan Password Anda

Email

Password

Masuk

Gambar 3. 12 Halaman Login Admin

c. Halaman Data Gejala

Halaman Data Gejala merupakan halaman untuk melihat data gejala yang sudah di inputkan, di halaman ini berisi kode gejala dan gejala serta terdapat menu tambah data, edit dan menghapus. Halaman ini disajikan pada gambar 3.13 dibawah ini

Data Gejala / Kriteria

Show 10 Entries

TAMBAH DATA

Search :

NO	KODE	Gejala	OPSI
1	G1	Apakah lampu indikator-indikator pada dashboard tidak mau menyala ketika kontak sudah ON ?	EDIT REMOVE
2	G2	Apakah lampu indikator check engine mati ketika kontak ON ?	EDIT REMOVE
3	G3	Apakah tidak bisa distarter dalam kondisi mesin sudah panas ?	EDIT REMOVE
4	G4	Apakah saat mesin dingin susah dihidupkan ?	EDIT REMOVE

Data Gejala

Data Kerusakan

Relasi Kerusakan

Gambar 3.13 Halaman Data Gejala

d. Halaman Tambah Gejala

Halaman Tambah Gejala merupakan halaman untuk menambahkan data gejala, di halaman ini admin melakukan penambahan kode

gejala dan pertanyaan untuk gejala. Halaman ini disajikan pada gambar 3.14 dibawah ini

Gambar 3.14 Halaman Tambah Gejala

e. Halaman Data Kerusakan

Halaman Data Kerusakan merupakan halaman untuk melihat data kerusakan yang berisi kode, nama kerusakan dan solusi perbaikan yang sudah di inputkan, di halaman ini terdapat menu tambah data, edit dan menghapus. Halaman ini disajikan pada gambar 3.15 dibawah ini

NO	KODE	NAMA	SOLUSI PERBAIKAN	OPSI
1	G1	Alsi	a. Periksa lampu indikator aki/baterai (pastikan bahwa masih bagus)...	EDIT REMOVE
2	G2	Fuel Pump	Membersihkan atau mengganti pompa bahan bakar yang baru	EDIT REMOVE
3	G3	Thermostat	Kita tidak dapat memperbaiki, disarankan untuk pergi ke bengkel	EDIT REMOVE
4	G4	Ignition Coil	Kita tidak bisa memeriksa ignition coil, jika tidak ingin repot maka disarankan untuk ke bengkel	EDIT REMOVE

Gambar 3.15 Data Kerusakan

f. Halaman Tambah Kerusakan

Halaman Tambah Kerusakan merupakan halaman untuk menambahkan data kerusakan, di halaman ini admin melakukan penambahan kode kerusakan, nama kerusakan dan solusi perbaikan. Halaman ini disajikan pada gambar 3.16 dibawah ini

Gambar 3.16 Tambah Kerusakan

g. Halaman Relasi Kerusakan

Halaman Relasi Kerusakan merupakan halaman untuk melihat data kerusakan yang berisi alternative/nama kerusakan, kode kerusakan dan basis data yang sudah di inputkan, di halaman ini terdapat menu lihat basis data dan edit. Halaman ini disajikan pada gambar 3.17 dibawah ini

NO	ALTERNATIF	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	OPSI
1	Aki	YES	YES	-	YES	-	-	-	YES	-	-	-	-	-	-	-	edit
2	Fuel Pump	-	-	-	YES	YES	YES	-	-	-	YES	-	-	-	-	-	edit
3	Thermostat	-	-	-	YES	-	-	-	-	YES	-	-	YES	YES	-	-	edit
4	Ignition Coll	-	-	YES	-	YES	-	YES	-	-	-	-	-	-	-	-	edit

Gambar 3.17 Relasi Kerusakan

h. Halaman Edit Relasi

Halaman Edit Relasi merupakan halaman untuk memberikan nilai MB dan MD serta relasi yang berubungan dengan kerusakan dengan memberikan pilihan YES. Halaman ini disajikan pada gambar 3.18 dibawah ini

KRITERIA	MB	MD	KI (AKI)
G1 1 Apakah lampu indikator-indikator pada dashboard tidak mau menyala ketika kontak sudah ON ?	0,9	0,04	<input type="radio"/> YES
G2 2 Apakah lampu indikator check engine mati ketika kontak ON ?	0,8	0,02	<input type="radio"/> NO
G3 3 Apakah tidak bisa distarter dalam kondisi mesin sudah panas ?	-	-	<input type="radio"/> YES
G4 4 Apakah saat mesin dingin susah dihidupkan ?	0,9	0,01	<input type="radio"/> YES

Gambar 3.18 Edit Relasi

i. Halaman Relasi Kerusakan dan Solusi

Halaman Relasi Kerusakan dan Solusi merupakan halaman untuk melihat data yang sudah di inputkan. Halaman ini berisi gejala, nama kerusakan, solusi dan nilai MD/MB. Halaman ini disajikan pada gambar 3.19 dibawah ini

Relasi Kerusakan & Kerusakan

Show KEMBALI

10 Search :

Entries

NO	GEJALA	KERUSAKAN	SOLUSI	MB	MD
1	Apakah lampu indikator-indikator pada dashboard tidak mau menyala ketika kontak sudah ON ?	Aki	a. Periksa lampu indikator aki/baterai (pastikan bahwa masih bagus)...	0,9	0,04
2	Apakah lampu indikator check engine mati ketika kontak ON ?	Fuel Pump	Membersihkan atau mengganti pompa bahan bakar yang baru	0,8	0,02
3	Apakah saat mesin dingin susah dihidupkan ?	Thermostat	Kita tidak dapat memperbaiki, disarankan untuk pergi ke bengkel	0,9	0,01
4	Apakah tidak bisa digas ?	Ignition Coil	Kita tidak bisa memeriksa ignition coil, jika tidak ingin repot maka disarankan untuk ke bengkel	0,7	0,01

Gambar 3.19 Relasi Kerusakan Dan Solusi

j. Halaman Mulai Diagnosa

Halaman Mulai Diagnosa merupakan halaman pertanyaan gejala yang dialami oleh mobil dengan memilih opsi YA atau

TIDAK kemudian pilih menu Simpan Jawaban dan sistem akan melakukan proses *Forward Chaining* dan proses *Certainty Factor*.

Halaman ini disajikan pada gambar 3.19 dibawah ini

Pilihlah Button Ya atau Tidak Pada Pertanyaan yang Menurut Anda Benar !

Mulai ulang Keluar

G-4 Apakah Saat Mesin Dingin Susah Dihidupkan ?

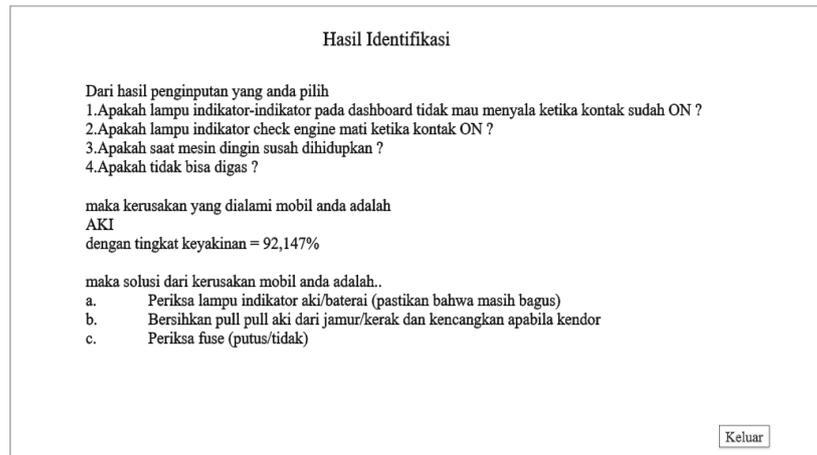
Ya
 Tidak

Simpan Jawaban

Gambar 3.19 Halaman Mulai Diagnosa

k. Halaman Hasil Identifikasi

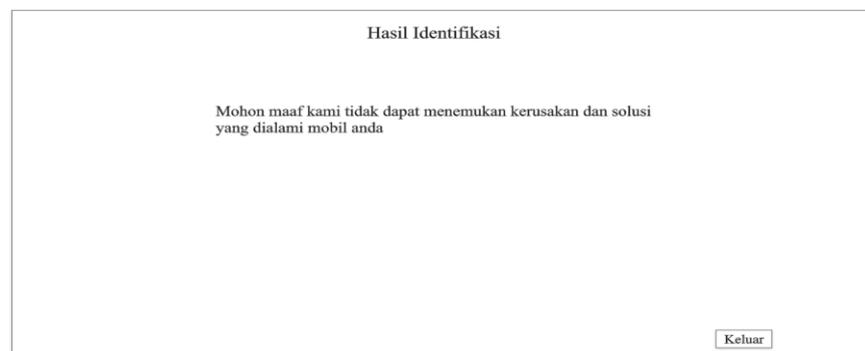
Halaman Hasil Identifikasi merupakan hasil dari pengguna setelah menginputkan gejala yang dialami mobil grand livina. Berikut adalah tampilan hasil identifikasi pengguna sistem ini yang disajikan pada Gambar 3.20



Gambar 3.20 Halaman Hasil Identifikasi

l. Halaman Jika Tidak Ditemukan Jawaban

Halaman ini merupakan tampilan hasil identifikasi dengan menginputkan gejala yang dialami mobil grand livina tersebut namun tidak ditemukan di basis pengetahuan yang tersimpan. Halaman ini disajikan pada Gambar 3.21



Gambar 3.21 Halaman Jika Tidak Ditemukan Jawaban

BAB VI

KESIMPULAN DAN SARAN

A. Kesimpulan

Dari uraian pembahasan yang telah disampaikan, dapat diambil kesimpulan sebagai berikut:

1. Sistem pakar yang telah dibuat mampu melakukan proses penalaran data dengan *forward chaining* dan memberikan tingkat keyakinan pada kerusakan.
2. Sistem pakar yang telah dibangun ini mendapatkan presentase sebesar 79,7% ini dapat disimpulkan sistem sudah dianggap baik dan layak digunakan.

B. Saran

Adapun berbagai saran untuk melengkapi kesimpulan yang diambil adalah :

1. Untuk pengembangan sistem pakar kedepan diharapkan dapat diimplementasikan dalam mobile phone agar mudah dilakukan setiap saat.
2. Diharapkan kedepan dapat ditambahkan fitur untuk gambar dan video pada saat perbaikan kerusakan sehingga pengguna lebih memahami.

DAFTAR PUSTAKA

- Abdullah, D. (2016). Sistem Pakar Mendiagnosa Gejala Kerusakan Mesin Mobil Toyota Menggunakan Metode Case Based Reasoning. *Jurnal Pilar Nusa Mandiri*, 14(January 2015), 35–40.
- Adhar, D. (2018). SISTEM PAKAR DETEKSI GEJALA KERUSAKAN PADA MOTOR Matic HONDA VARIO 125 BERBASIS WEB MENGGUNAKAN METODE CERTAINTY FACTOR. *Jurnal Sistem Informasi Kaputama (JSIK)*, 2(1), 66–72.
- Akil, I. (2017). ANALISA EFEKTIFITAS METODE FORWARD CHAINING DAN BACKWARD CHAINING PADA SISTEM PAKAR. *Jurnal Pilar Nusa Mandiri*, 13(1), 35–42.
- Al-ajlan, A. (2015). The Comparison between Forward and Backward Chaining. *International Journal of Machine Learning and Computing*, 5(2), 106–113. <https://doi.org/10.7763/IJMLC.2015.V5.492>
- Arrhioui, K., Mbarki, S., Betari, O., Roubi, S., & Erramdani, M. (2017). A Model Driven Approach for Modeling and Generating PHP CodeIgniter based Applications. *Transactions on Machine Learning and Artificial Intelligence*, 5(4). <https://doi.org/10.14738/tmlai.54.3189>
- Azhar, Z. (2019). PENDETEKSIAN KERUSAKAN SEPEDA MOTOR DENGAN SISTEM PAKAR. *JURTEKSI (Jurnal Teknologi Dan Sistem Informasi)*, v(2), 167 – 174. <https://doi.org/10.33330/jurteks.v5i2.340>
- Fikri, A. F., & Widians, J. A. (2017). Sistem Pakar Diagnosa Kerusakan Pada Mobil Strada Triton Menggunakan Certainty Factor. *Prosiding Seminar Ilmu Komputer Dan Teknologi Informasi*, 2(1), 21–26.
- Fitri Ayu, N. P. (2018). Perancangan Sistem Informasi Pengolahan Data Praktek Kerja Lapangan (PKL) Pada Devisi Humas PT. Pegadaian. *Intra-Tech*, 2(2), 12–26.
- Hendini, A. (2016). *Pemodelan UML sistem informasi monitoring Penjualan dan stok barang (STUDI KASUS: DISTRO ZHEZHA PONTIANAK)*. IV(2), 107–116.
- Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 77. <https://doi.org/10.25126/jtiik.201851610>
- Livina, N. N., Jakarta, M., Nissan, P. T., Indonesia, M., Industri, G., & Bermotor, K. (2020). *Setelah Ambyar , Penjualan Nissan Melonjak Signifikan*. February, 2018–2020.

- Maulana, H. (2016). ANALISIS DAN PERANCANGAN SISTEM REPLIKASI DATABASE MYSQL DENGAN MENGGUNAKAN VMWARE PADA SISTEM. *InfoTekJar(Jurnal Nasional Informatika Dan Teknologi Jaringan)*, 1(1), 32–37.
- Mobil88. (2019). *Beberapa Kategori Mobil yang Patut Diketahui!* 5–8. <https://www.mobil88.astra.co.id/mobil88/in/blog/beberapa-kategori-mobil-yang-patut-diketahui>
- Nasir, J., & Gultom, Z. H. (2018). Sistem Pakar Untuk Mendeteksi Kerusakan Pada Sepeda Motor Dengan Menggunakan Metode Forward Chaining Berbasis Web. *Jurnal Teknologi Informasi & Komunikasi Digital Zone*, 9(1), 42–58.
- Nugroho, K. (2020). KENDARAAN PADA MOBIL WULING CONFERO S MENGGUNAKAN METODE CERTAINTY FACTOR. *Jurnal Sistem Informasi*, 7(1), 63–69.
- Purbo, T. (2019). *APLIKASI DIAGNOSIS KERUSAKAN SEPEDA MOTOR BEBEK METODE FORWARD CHAINING BERBASIS ANDROID*.
- Ramadiani, R., & Mulawarman, U. (2018). Sistem Pakar Identifikasi Kerusakan Pada Mobil. *Jurnal Informatika Mulawarman*, 6(1), 30–38.
- Rusdiansyah. (2017). SISTEM PAKAR MENDETEKSI KERUSAKAN MESIN VVT-I BERBASIS WEB PADA KENDARAAN TOYOTA VIOS. *Jurnal PILAR Nusa Mandiri*, 13(2), 255–260.
- Sapri. (2018). Perangkat Lunak Sistem Pakar Untuk Mengidentifikasi Kerusakan Pada Mobil Toyota Kijang Menggunakan Metode Forward Chaining. *Jurnal Informatika Kaputama (JIK)*, 2(1), 17–27.
- Satwika, I. B. D. (2012). RANCANG BANGUN SISTEM DIAGNOSIS KERUSAKAN PADA MOBIL MENGGUNAKAN METODE FORWARD CHAINING. *Jurnal Elektronik Ilmu Komputer*, 1(2), 66–72.
- Septarini, R. S., & Nugroho, T. (2018). Rancang Bangun Aplikasi Sistem Diagnosis Kerusakan Pada Mobil Daihatsu Xenia 1 . 3 M / T Dengan Metode Forward Chaining. *Jurnal Teknik Informatika (JIKA) Universitas Muhammadiyah Tangerang*, 84–93.
- simatupang julianto, sianturi setiawan. (2019). PERANCANGAN SISTEM INFORMASI PEMESANAN TIKET BUS PADA PO. HANDOYO BERBASIS ONLINE Julianto. *Simatupang, Julianto Sianturi, Setiawan*, 3(2), 11–25.

- Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, 3(1), 1–9. <http://jurnal.uinsu.ac.id/index.php/algorithm/article/download/3148/1871>
- Sulaiman, M. M. (2020). Perancangan Prototipe Sistem Pakar Diagnosa Kerusakan Mobil Toyota Tipe Mpv Menggunakan Metode Forward Dan Backward Chaining Berbasis Android. *Journal Of Artificial Intelligence And Innovative Applications Vol.*, 1(1), 6–11.
- Tobin, M. J. (2004). Asthma, Airway Biology, and Nasal Disorders in AJRCCM 2003. *American Journal of Respiratory and Critical Care Medicine*, 169(2), 265–276. <https://doi.org/10.1164/rccm.2312011>
- Toyota, L. P. T., & Motor, A. (2019). *Merek Mobil Terlaris di Indonesia Selama 2019 Adalah ...* 4–6. <https://otodriver.com/berita/2020/merek-mobil-terlaris-di-indonesia-selama-2019-adalah-mercacajlah>
- Yusman, H., Efendi, R., & Coastera, F. F. (2017). PADA MESIN MOBIL TOYOTA DENGAN METODE CERTAINTY FACTOR(CF) BERBASIS ANDROID. *Rekursif*, 5(3), 317–330. <http://ejournal.unib.ac.id/index.php/rekursif/%0A>