



SISTEM PAKAR IDENTIFIKASI KERUSAKAN SEPEDA MOTOR SUZUKI SATRIA F150 DENGAN TEOREMA BAYES

Foni Agus Setiawan^{1,2*}, Dewi Primasari², Wahyudin²

¹Pusat Penelitian Informatika, Lembaga Ilmu Pengetahuan Indonesia

²Program Studi Teknik Informatika, Fakultas Teknik dan Sains, Universitas Ibn Khaldun
Bogor, Indonesia

*e-mail: masagus@masagus.id

Abstrak

Sepeda motor merupakan jenis kendaraan beroda dua yang digerakkan oleh mesin. Sepeda motor terkadang mengalami kerusakan yang disebabkan oleh komponen yang telah aus, pemeliharaan yang tidak teratur, atau kesalahan pemakaian. Kerusakan tersebut ada yang bersifat ringan, sedang, atau berat. Sistem pakar identifikasi kerusakan sepeda motor Suzuki Satria F150 dibangun untuk dapat membantu mengidentifikasi kerusakan dan menawarkan solusi yang harus dilakukan untuk mengatasi kerusakan tersebut. Teorema Bayes digunakan sebagai model untuk pendugaan bagian atau komponen yang rusak. Metode inferensi backward chaining digunakan untuk menelusuri penyebab berdasarkan kerusakan yang terjadi. Seleksi aturan IF-THEN dalam mesin inferensi bermetode backward chaining diperkuat dengan prioritas pembobotan kandidat yang dilakukan berdasarkan teorema Bayes. Sistem pakar diimplementasikan dalam bentuk sebuah aplikasi berbasis web dengan basis pengetahuan terdiri dari 5 komponen, 48 gejala, 219 penyebab, dan 181 solusi perbaikan. Pengujian validitas sistem menunjukkan hasil 100% valid. Pengujian akurasi sistem dari 50 sampel menunjukkan tingkat akurasi sebesar 94%. Selain berguna bagi mekanik bengkel sepeda motor untuk mendukung pekerjaannya, sistem pakar ini juga bermanfaat bagi pengguna motor Suzuki Satria F150 sebagai alat bantu identifikasi awal kerusakan sehingga dapat diambil tindakan lebih lanjut.

Kata kunci: ; Inferensi Backward Chaining; Sistem Pakar; Suzuki Satria F150; Teorema Bayes;

Abstract

Motorbike is a type of two-wheeled vehicle driven by an engine. Motorbikes sometimes suffer damage caused by worn components, irregular maintenance, or misuse. The damage can be minor, moderate, or severe. The Suzuki Satria F150 motorbike damage identification expert system was built to help identify the damage and offer solutions that must be done to overcome the damage. Bayes' Theorem is used as a model for estimating the damaged part or component. Backward chaining inference method is used to trace the cause of the damage according to the observed symptoms. The IF-THEN rule selection in the backward chaining method inference engine is strengthened by the weighting priority of the candidates based on the Bayes' theorem. The expert system is implemented in the form of a web-based application consists of 5 components, 48 symptoms, 219 causes, and 181 repair solutions. System

validation testing shows the results are 100% valid. Accuracy system testing from 50 samples shows an accuracy rate of 94%. Apart from being useful for motorbike repair shop mechanics to support their work, this expert system is also useful for Suzuki Satria F150 motorbike users as a tool for early identification of damage so that further action can be taken.

Keywords: *Expert System; Bayes' Theorem; Backward Chaining Inference; Suzuki Satria F150*

PENDAHULUAN

Menurut Undang-Undang No. 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, sepeda motor adalah kendaraan bermotor beroda dua dengan atau tanpa rumah-rumah dan dengan atau tanpa kereta samping atau kendaraan bermotor beroda tiga tanpa rumah-rumah [1]. Suzuki adalah nama salah satu merek sepeda motor yang dijual di Indonesia. Merek Suzuki memiliki beberapa tipe motor, diantaranya adalah Suzuki Satria F150 yang merupakan tipe andalan. Suzuki Satria F150 atau dikenal juga sebagai Suzuki Satria FU adalah sepeda motor berkecepatan tinggi di kategori bebek 4-tak. Satria F150 dilengkapi mesin dengan teknologi tinggi, volume silinder 147,33 cc, transmisi 6-speed, *Double OverHead Camshaft* (DOHC), 4-valve, silinder tunggal, *liquid cooled*, dan beberapa kelebihan lainnya [2].

Mekanik yang menangani perbaikan kerusakan motor Suzuki Satria F150, terutama para mekanik pemula, terkadang menemui kendala karena harus hafal aturan diagnosis berdasarkan gejala-gejala dari motor yang diperiksa. Demikian pula dengan para pemilik/pengguna motor Suzuki Satria F150, ketika terjadi kerusakan, mereka memerlukan identifikasi awal agar dapat mengambil tindakan yang diperlukan. Apabila kerusakan yang terjadi tergolong ringan, pemilik/pengguna dapat mencoba melakukan perbaikan mandiri tanpa harus datang ke bengkel.

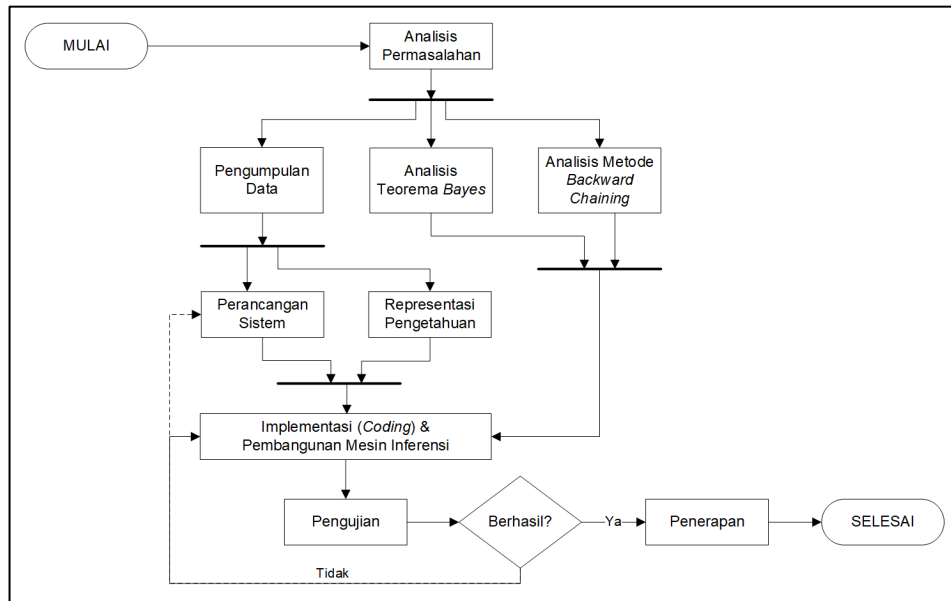
Penggunaan cara manual dengan mencari di buku Pedoman Perbaikan menjadi tidak praktis karena memerlukan waktu lama dalam proses pencariannya. Hal ini karena pengetahuan dikelompokkan per komponen dan gejala yang mungkin terjadi pada komponen tersebut. Padahal, satu gejala kerusakan mungkin dapat diakibatkan oleh beberapa penyebab dengan intensitas pengaruh yang berbeda-beda. Selain itu, tidak ada fasilitas untuk konsultasi terkait gejala kerusakan yang dialami pengguna.

Berdasarkan latar belakang permasalahan tersebut, penelitian ini dilakukan dengan tujuan untuk membangun sebuah sistem pakar diagnosis atau identifikasi kerusakan pada sepeda motor Suzuki Satria F150 dan menawarkan solusi perbaikannya. Sistem pakar dibangun dalam bentuk sebuah aplikasi berbasis web.

Model berdasar pada teorema Bayes digunakan untuk membantu penentuan penyebab kerusakan berdasarkan gejala yang diamati. Teorema Bayes digunakan karena intuitif namun sederhana dan telah berhasil diimplementasikan di berbagai bidang [3] guna menentukan probabilitas hubungan satu kejadian dengan kejadian yang lain. *Backward chaining* [4] digunakan sebagai metode inferensi karena penalaran bergerak mundur dari gejala yang diamati hingga ditemukan penyebab kerusakannya.

METODE PENELITIAN

Metode yang digunakan dalam penelitian ini terlihat pada Gambar 1.



Gambar 1. Metode penelitian.

a. Analisis Permasalahan

Analisis permasalahan yang dilakukan adalah memetakan hubungan antara gejala-gejala dengan penyebab kerusakan yang dialami. Satu gejala mungkin dapat diakibatkan oleh beberapa penyebab kerusakan dan satu penyebab kerusakan mungkin pula mengakibatkan beberapa gejala dengan tingkat pengaruh yang berbeda-beda.

b. Pengumpulan Data

Pengumpulan data dilakukan untuk mendapatkan data gejala, penyebab, dan solusi perbaikan. Pengumpulan data dilakukan melalui studi pustaka dari buku pedoman perbaikan Suzuki Satria F150 [5] yang divalidasi dan diperkaya dengan wawancara dengan kepala mekanik di sebuah bengkel resmi Suzuki di Kota Depok. Hasil cuplikan data gejala, penyebab kerusakan, dan solusi perbaikan untuk permasalahan tenaga mesin berkurang tersaji pada Tabel 1, Tabel 2, dan Tabel 3. Isi lengkap dari tabel-tabel tersebut yang terdiri dari 48 gejala, 219 penyebab, dan 181 solusi perbaikan dapat dilihat pada aplikasi sistem pakar yang dibangun¹.

Tabel 1. Daftar Gejala.

ID Gejala	Deskripsi Gejala
J1	Mesin tidak mau hidup atau susah dihidupkan
J2	Mesin mudah mati
J3	Mesin bunyi kasar
J4	Clutch selip
J5	RPM tinggi tapi kecepatan rendah
...	
J15	Kelebihan putaran mesin ketika langsam (<i>stationer</i>)

¹ <http://www.masagus.id/spsf150/>

Tabel 2. Daftar Penyebab Kerusakan.

ID Penyebab	Deskripsi Penyebab
K69	Celah <i>valve</i> terlalu besar
K18c	<i>Valve spring</i> lemah atau patah
K3b	Waktu pengaturan <i>valve</i> salah (<i>mistiming valve</i>)
K24c	<i>Ring piston</i> atau alur ring aus
K70	<i>Valve seating</i> tidak pas
K7c	Busi kotor
K71	Pemakaian tipe busi tidak tepat
K72	<i>Jet</i> di <i>carburetor</i> tidak tepat
K73	Penyetelan pelampung <i>carburetor</i> tidak tepat
K63b	<i>Air filter element</i> tersumbat
K74a	Udara tersedot dari <i>intake pipe</i>
K65b	Oli mesin terlalu banyak
K58b	<i>Camshaft</i> aus

Tabel 3. Daftar Solusi Perbaikan.

ID Solusi	Deskripsi Solusi
L69	Setel celah <i>valve</i> yang terlalu besar
L18	Ganti <i>valve spring</i> yang lemah atau patah
L3	Setel ulang waktu pengaturan <i>valve</i> yang salah (<i>mistiming valve</i>)
L24	Ganti <i>ring piston</i> atau alur ring yang aus
L70	Setel <i>valve seating</i> yang tidak pas
L7	Bersihkan busi yang kotor
L71	Ganti tipe busi yang tidak tepat
L72	Setel <i>jet</i> di <i>carburetor</i> yang tidak tepat
L73	Setel pelampung <i>carburetor</i> yang tidak tepat
L63	Bersihkan <i>air filter element</i> yang tersumbat
L74	Kencangkan atau ganti <i>intake pipe</i>
L65	Kurangi oli mesin yang terlalu banyak
L58	Ganti <i>camshaft</i> yang aus

c. Representasi Pengetahuan

Pada tahap ini dilakukan penyusunan basis pengetahuan sebagai dasar bagi mesin inferensi untuk mengambil kesimpulan dan menawarkan solusi. Basis pengetahuan tersusun atas aturan-aturan (*rules*) yang menghubungkan antara penyebab kerusakan, gejala, dan solusi yang ditawarkan. Cuplikan aturan untuk permasalahan tenaga mesin berkurang dapat dilihat pada Tabel 4.

Tabel 4. Representasi Pengetahuan.

ID Aturan	Aturan		Solusi
	<i>IF</i>	<i>THEN</i>	
Rule85	K69: Celah <i>valve</i> terlalu besar	J10: Tenaga mesin berkurang	L69
Rule86	K18c: <i>Valve spring</i> lemah atau patah	J10: Tenaga mesin berkurang	L18
Rule87	K3b: Waktu pengaturan <i>valve</i> salah (<i>mistiming valve</i>)	J10: Tenaga mesin berkurang	L3
Rule88	K24c: <i>Ring piston</i> atau alur ring aus	J10: Tenaga mesin berkurang	L24
Rule89	K70: Valve seating tidak pas	J10: Tenaga mesin berkurang	L70
Rule90	K7c: Busi kotor	J10: Tenaga mesin berkurang	L7
Rule91	K71: Pemakaian tipe busi tidak tepat	J10: Tenaga mesin berkurang	L71
Rule92	K72: <i>Jet</i> di <i>carburetor</i> tidak tepat	J10: Tenaga mesin berkurang	L72
Rule93	K73: Penyetelan pelampung <i>carburetor</i> tidak tepat	J10: Tenaga mesin berkurang	L73
Rule94	K63b: <i>Air filter element</i> tersumbat	J10: Tenaga mesin berkurang	L63
Rule95	K74a: Udara tersedot dari <i>intake pipe</i>	J10: Tenaga mesin berkurang	L74
Rule96	K65b: Oli mesin terlalu banyak	J10: Tenaga mesin berkurang	L65
Rule97	K58b: <i>Camshaft</i> aus	J10: Tenaga mesin berkurang	L58

d. Implementasi (*Coding*) dan Pembangunan Mesin Inferensi

Pada tahap ini dilakukan pembangunan aplikasi sistem pakar berbasis web. Bahasa pemrograman yang digunakan adalah PHP 7.3 dengan MySQL 5.6 sebagai server database. Server web yang digunakan adalah Apache versi 2.4. Aplikasi berjalan di atas sistem operasi Windows 10.

Mesin inferensi yang dibangun merupakan mesin inferensi berbasis aturan IF-THEN yang dikombinasikan dengan Teorema Bayes guna menghitung kemungkinan pengaruh penyebab kerusakan terhadap gejala yang diamati. Metode *backward chaining* digunakan karena penalaran bergerak mundur dari gejala/kerusakan yang diamati untuk menemukan penyebabnya.

Teorema Bayes

Sistem pakar yang dibangun adalah sistem pakar yang mampu melakukan penalaran berdasarkan tingkat pengaruh satu kejadian terhadap yang lain. Oleh karena itu, digunakan

teorema Bayes guna memodelkan tingkat keterkaitan antara satu kejadian dengan kejadian yang lain. Misalnya, pengaruh kondisi busi terhadap kelistrikan motor atau pengaruh kondisi *camshaft* terhadap tenaga mesin.

Teorema Bayes menjelaskan probabilitas suatu peristiwa berdasarkan pengetahuan sebelumnya (*prior knowledge*) tentang kondisi yang mungkin terkait dengan peristiwa tersebut [6]. Teorema Bayes didasarkan pada prinsip bahwa tambahan informasi dapat memperbaiki probabilitas. Penalaran menggunakan prinsip ini dikenal sebagai penalaran non-monotonik [7], yang berarti hasil penarikan kesimpulan dari bukti-bukti yang ada pada suatu saat dapat berubah ketika datang bukti-bukti yang baru karena berubahnya nilai-nilai probabilitas atau tingkat kepercayaan terhadap bukti-bukti tersebut.

Teorema Bayes digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang didapat dari hasil observasi. Teorema ini menerangkan hubungan antara probabilitas terjadinya peristiwa H dengan syarat peristiwa E telah terjadi, dengan persamaan [8]:

$$P(H_i|E) = \frac{P(E|H_i) * P(H_i)}{\sum_{k=m}^n P(E|H_k) * P(H_k)} \quad (1)$$

dengan

- $P(H_i|E)$ probabilitas hipotesis H_i benar jika diberikan *evidence* E
- $P(E|H_i)$ probabilitas munculnya *evidence* E , jika diketahui hipotesis H_i benar
- $P(H_i)$ probabilitas hipotesis H_i (menurut hasil sebelumnya) tanpa memandang *evidence* apapun
- m, n indeks bawah dan atas setiap hipotesis H_k yang mempengaruhi *evidence* E , dengan $m \leq i \leq n$.

Contoh kasusnya adalah sebagai berikut. Misalnya motor mengalami gejala kemudi goyang (tidak stabil). Penyebab yang mungkin mengakibatkan gejala tersebut adalah setelan suspensi kanan dan kiri tidak sama; suspensi depan bengkok; atau *axle shaft* roda depan bengkok atau ban tidak baik. Sehingga:

- E = kemudi goyang
- H_1 = setelan suspensi kanan dan kiri tidak sama
- H_2 = suspensi depan bengkok
- H_3 = *axle shaft* roda depan bengkok atau ban tidak baik

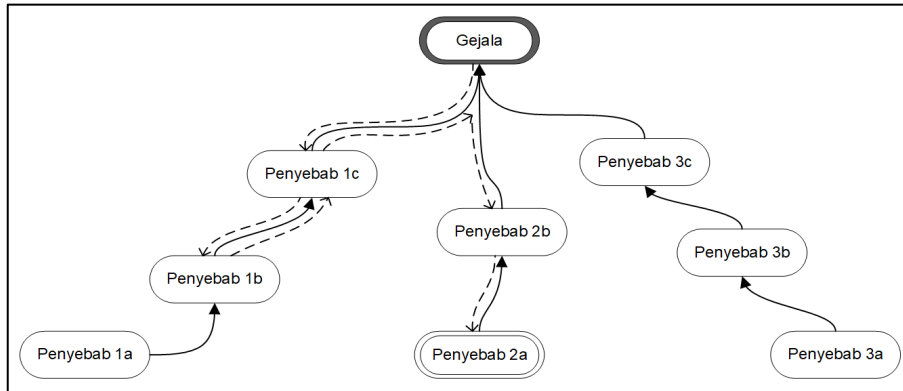
Jika hendak dicari probabilitas kemudi goyang karena disebabkan oleh setelan suspensi kanan dan kiri yang tidak sama = $P(E)$, maka:

$$P(H_1|E) = \frac{P(E|H_1) * P(H_1)}{P(E|H_1) * P(H_1) + P(E|H_2) * P(H_2) + P(E|H_3) * P(H_3)}$$

Backward Chaining

Backward chaining (atau *backward reasoning*) adalah metode inferensi yang bekerja secara mundur dari tujuan (*goal/hypotheses*) menuju aturan dan fakta-fakta (*consequent/antecedent*) untuk membuktikan kebenaran *hypotheses* tersebut [9]. Dengan kata lain, *backward chaining* adalah teknik pelacakan yang diinisiasi dengan tujuan [10]. Metode inferensi ini digunakan pada pembuktian teorema, mesin inferensi sistem pakar, *proof assistants*, dan bidang kecerdasan buatan lainnya.

Dalam teori permainan (*game theory*), metode ini digunakan untuk memecah sebuah permainan menjadi sub-sub permainan, mendapatkan solusi untuk setiap sub-permainan, yang pada akhirnya digunakan untuk mendapatkan solusi permainan tersebut. Metode ini biasa digunakan bersamaan dengan aturan-aturan inferensi dan implikasi logis. Sebuah sistem *backward chaining* biasanya menggunakan teknik *depth-first search* seperti yang digunakan dalam Prolog [11]. Gambar 2 mengilustrasikan metode *backward chaining* dalam melakukan inferensi untuk menemukan Penyebab 2a sebagai penyebab munculnya Gejala.



Gambar 2. Ilustrasi cara kerja inferensi *backward chaining*.

e. Pengujian

Sistem pakar yang dibangun selanjutnya akan diuji untuk melihat kinerjanya dari dua sisi, validitas dan akurasi sistem. Pengujian validitas dilakukan untuk melihat apakah fungsionalitas sistem telah bekerja sesuai yang diharapkan. Dalam kasus sistem pakar berbasis aturan, berarti menguji pula apakah aturan IF-THEN telah bekerja sesuai yang diharapkan. Pengujian akurasi dilakukan untuk melihat seberapa akurat sistem pakar dalam memberikan jawaban terhadap kasus yang dikonsultasikan kepadanya. Tingkat akurasi sistem dihitung dengan persamaan:

$$akurasi = \frac{\sum \text{sesuai}}{\sum \text{uji}} \tag{2}$$

dengan

$\sum \text{sesuai}$ = jumlah prediksi penyebab yang sesuai
 $\sum \text{uji}$ = jumlah pengujian

HASIL DAN PEMBAHASAN

Hasil pembangunan sistem yang dibahas terdiri dari hasil perancangan sistem dan hasil penerapan sistem.

a. Perancangan Sistem

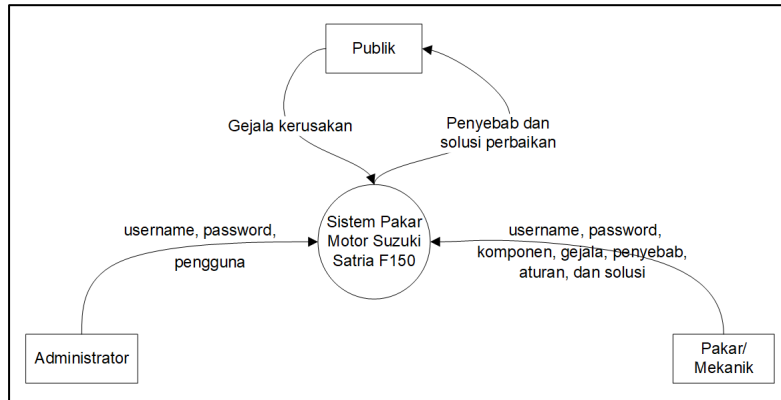
Perancangan sistem yang diulas mencakup diagram konteks, diagram alir data, dan diagram hubungan entitas.

Diagram Konteks

Diagram konteks merupakan gambaran keseluruhan dari sistem yang akan dibangun yang meliputi seluruh eksternal entitas yang terlibat dan arah arus data yang masuk ke sistem

dan yang keluar dari sistem. Diagram konteks sistem pakar ditunjukkan pada Gambar 3.

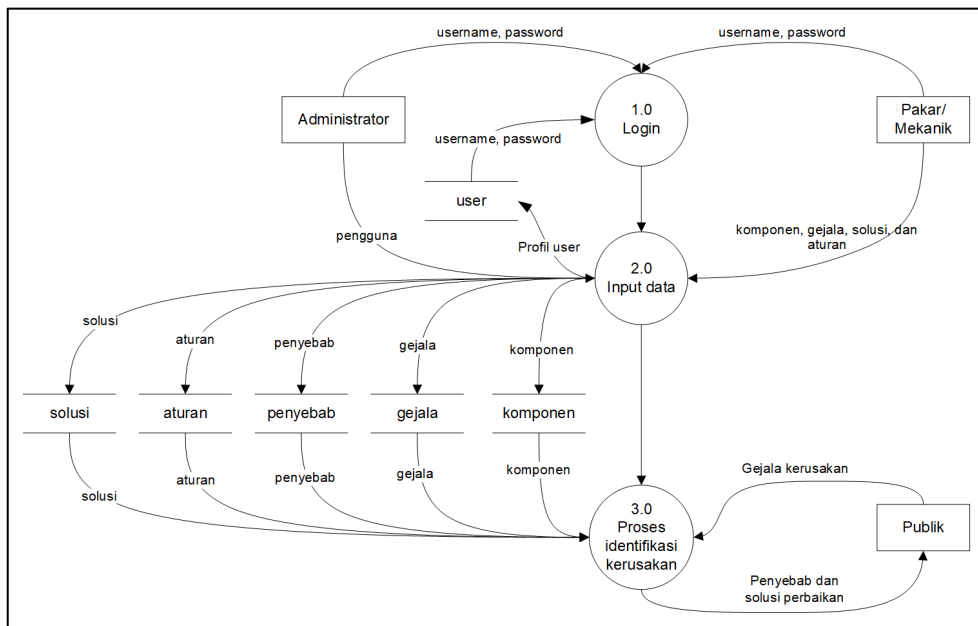
Terdapat tiga entitas yang merupakan tiga jenis pengguna sistem, yaitu Administrator, Pakar/Mekanik, dan Publik. Administrator bertugas mengelola data pengguna. Pakar/Mekanik bertugas mengisi data pengetahuan yang mencakup komponen, gejala, penyebab, aturan, dan solusi. Adapun Publik memanfaatkan sistem untuk melakukan konsultasi identifikasi kerusakan sepeda motor.



Gambar 3. Diagram konteks.

Diagram Alir Data

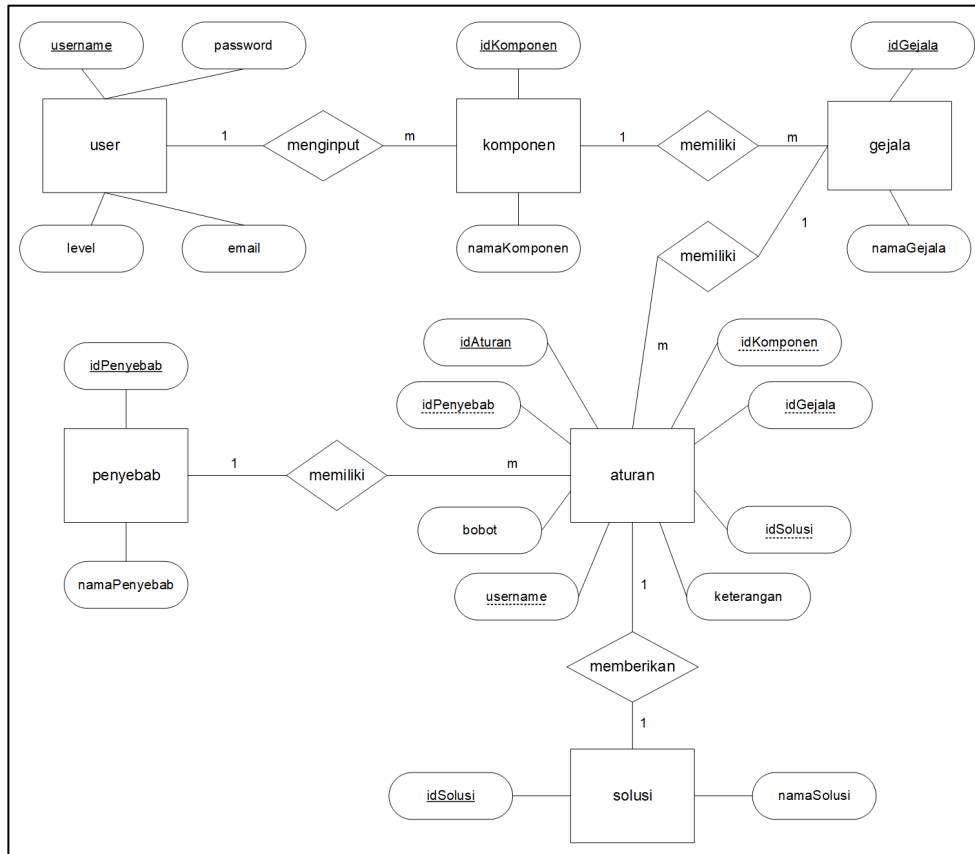
Diagram alir data sistem pakar identifikasi kerusakan motor Suzuki Satria F150 terlihat pada Gambar 4. Terdapat tiga proses utama sistem, yaitu Login bagi Administrator dan Pakar/Mekanik, Input data pengguna sistem dan data pengetahuan, serta Proses identifikasi kerusakan yang dilakukan ketika pengguna Publik berkonsultasi. Enam *data store* dibuat untuk mendukung proses sistem, yaitu *user*, *komponen*, *gejala*, *penyebab*, *aturan*, dan *solusi*.



Gambar 4. Diagram alir data.

Diagram Hubungan Entitas

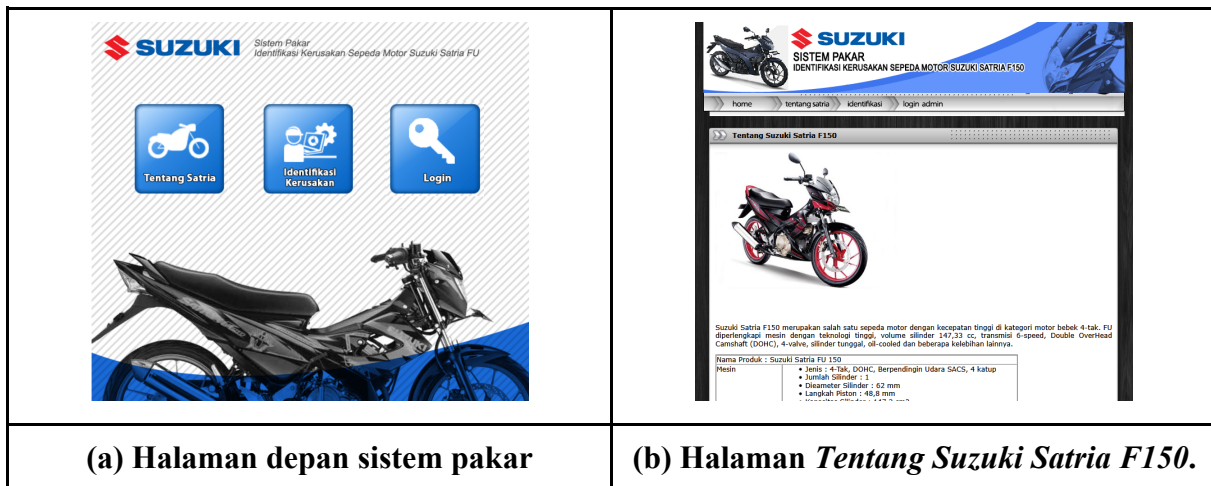
Enam *data store* berupa entitas yang diperoleh pada tahap pembuatan diagram alir data selanjutnya dipetakan untuk melihat hubungan antar-entitas tersebut. Gambar 5 memperlihatkan diagram hubungan antar-entitas. Dari diagram ini terlihat bahwa gejala dikelompokkan berdasarkan komponen. Satu gejala dapat memiliki banyak penyebab dan sebaliknya satu penyebab dapat mengakibatkan banyak gejala yang didefinisikan pada entitas *aturan*. Setiap aturan tertentu akan memberikan satu solusi perbaikan.



Gambar 5. Diagram hubungan entitas.

b. Penerapan Sistem

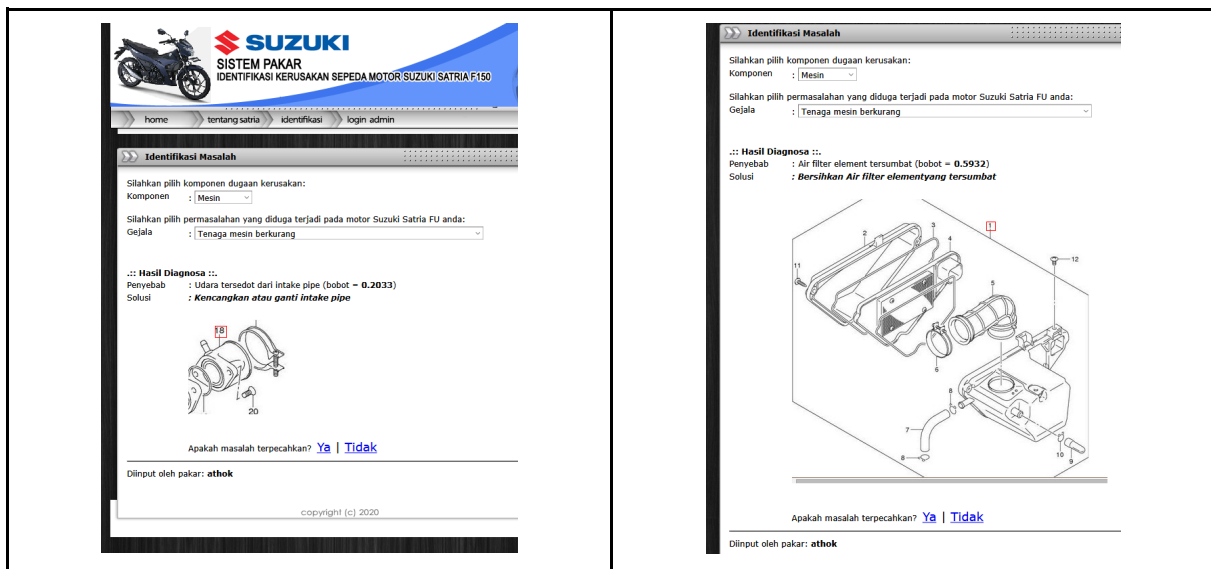
Perancangan yang telah dilakukan diterapkan ke dalam sebuah aplikasi berbasis web. Gambar 6(a) memperlihatkan halaman depan aplikasi yang terdiri dari tiga menu utama, yaitu *Tentang Satria*, *Identifikasi Kerusakan*, dan *Login*. Menu *Tentang Satria* menjelaskan deskripsi umum dan teknis tentang motor Suzuki Satria F150 yang tampilannya tampak seperti Gambar 6(b). Menu *Login* digunakan oleh Administrator dan Pakar/Mekanik untuk masuk ke dalam sistem guna mengelola pengguna dan basis pengetahuan sistem pakar. Menu *Identifikasi Kerusakan* digunakan oleh pengguna untuk berkonsultasi dengan sistem pakar. Gambar 7 memperlihatkan tampilan dari halaman identifikasi kerusakan.



(a) Halaman depan sistem pakar

(b) Halaman Tentang Suzuki Satria F150.

Gambar 6. Penerapan Sistem



Gambar 7. Halaman Identifikasi Kerusakan

Pertama-tama pengguna diberikan pilihan untuk menentukan komponen apa yang ingin diidentifikasi kerusakannya. Komponen terbagi ke dalam lima jenis, yaitu Mesin, Rangka/Chasis, Rem, Kelistrikan, dan Aki/Battery. Kemudian, pengguna menentukan gejala kerusakan yang dialami berkaitan dengan komponen tersebut. Sistem pakar lalu bekerja mendiagnosis kemungkinan penyebab kerusakan dan menampilkan hasil dengan kemungkinan (bobot) yang paling besar disertai dengan solusi perbaikan yang harus dilakukan. Selanjutnya, pengguna akan menguji apakah solusi yang diberikan oleh sistem mampu mengatasi kerusakan tersebut. Jika pengguna menjawab *Ya*, maka proses dialog selesai karena penyebab telah ditemukan. Apabila pengguna menjawab *Tidak*, sistem akan mencari kemungkinan penyebab berikutnya dengan bobot yang lebih kecil. Demikian proses ini terus berulang hingga pengguna menjawab *Ya*, atau sistem tidak lagi memiliki alternatif jawaban dan menampilkan pesan “Sistem tidak dapat menemukan penyebab lainnya”.

c. Pengujian Sistem

Pengujian sistem dilakukan dengan menguji fungsionalitasnya satu per satu. Hasil pengujian validitas menunjukkan bahwa sistem pakar yang dibangun telah berfungsi sesuai yang diharapkan terutama dalam memberikan jawaban ketika pengguna berkonsultasi dengan sistem.

Contoh kasus. Pengguna mengalami permasalahan pada komponen *mesin* dengan gejala *tenaga mesin berkurang*.

Probabilitas memandang gejala kerusakan :	Probabilitas tanpa memandang gejala :
1. Celah <i>valve</i> terlalu besar: 0,05	1. Celah <i>valve</i> terlalu besar: 0,03
2. <i>Valve spring</i> lemah: 0,05	2. <i>Valve spring</i> lemah: 0,03
3. Waktu pengaturan <i>valve</i> salah (<i>mistiming valve</i>): 0,05	3. Waktu pengaturan <i>valve</i> salah (<i>mistiming valve</i>): 0,03
4. <i>Ring piston</i> atau <i>cylinder</i> aus: 0,05	4. <i>Ring piston</i> atau <i>cylinder</i> aus: 0,03
5. <i>Valve seating</i> tidak pas: 0,05	5. <i>Valve seating</i> tidak pas: 0,03
6. Busi kotor : 0,10	6. Busi kotor : 0,15
7. Pemakaian tipe busi tidak tepat: 0,05	7. Pemakaian tipe busi tidak tepat: 0,03
8. <i>Jet</i> di <i>carburetor</i> tidak tepat: 0,05	8. <i>Jet</i> di <i>carburetor</i> tidak tepat: 0,03
9. Penyetelan pelampung <i>carburetor</i> tidak tepat: 0,05	9. Penyetelan pelampung <i>carburetor</i> tidak tepat: 0,03
10. <i>Air filter element</i> tersumbat: 0,25	10. <i>Air filter element</i> tersumbat: 0,35
11. Udara tersedot dari <i>intake pipe</i> : 0,15	11. Udara tersedot dari <i>intake pipe</i> : 0,20
12. Oli mesin terlalu banyak: 0,05	12. Oli mesin terlalu banyak: 0,03
13. <i>Camshaft</i> aus: 0,05	13. <i>Camshaft</i> aus: 0,03

dan jika probabilitas penyebab tanpa memandang gejala kerusakan: maka perhitungan nilai Bayes berdasarkan Persamaan (1) untuk penyebab *air filter element* tersumbat adalah:

$$\frac{(0,25) \times (0,35)}{(0,05 \times 0,03) + (0,05 \times 0,03) + (0,05 \times 0,03) + (0,05 \times 0,03) + (0,05 \times 0,03) + (0,10 \times 0,15) + (0,05 \times 0,03) + \dots + (0,05 \times 0,03) + (0,05 \times 0,03) + (0,25 \times 0,35) + (0,15 \times 0,20) + (0,05 \times 0,03) + (0,05 \times 0,03)}$$

$$= \frac{0,0875}{0,1475} = 0,5932$$

nilai ini sesuai dengan yang tertera pada Gambar 7.

Pengujian akurasi sistem dilakukan terhadap 50 sampel data. Dari 50 sampel data yang diuji, 47 penyebab yang diprediksi oleh sistem sesuai dengan kenyataan di lapangan. Sehingga, tingkat akurasi sistem sesuai dengan Persamaan (2) adalah:

$$akurasi = \frac{47}{50} = 0,94 = 94\%$$

KESIMPULAN

Sistem pakar identifikasi kerusakan motor Suzuki Satria F150 telah berhasil dibangun sebagai sebuah aplikasi berbasis web. Sistem pakar ini menggunakan teorema Bayes untuk menghitung nilai probabilitas penyebab berdasarkan gejala yang dialami dan *backward chaining* sebagai metode penalaran dalam mesin inferensinya. Dengan tingkat akurasi sebesar 94%, sistem pakar ini layak untuk digunakan pada implementasi di lapangan. Sistem pakar ini selain berguna bagi mekanik bengkel sepeda motor untuk mendukung pekerjaannya, juga bermanfaat bagi pengguna motor Suzuki Satria F150 sebagai alat bantu identifikasi awal kerusakan sehingga dapat diambil tindakan lebih lanjut.

DAFTAR PUSTAKA

- [1] Kemenkumham RI, *UU No. 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan*. Indonesia, 2009, p. 203.
- [2] Suzuki Indonesia, "All New Satria F150," 2020. [Online]. Available: <https://www.suzuki.co.id/motorcycle/all-new-satria-f150>. Diakses pada 27 Sep 2020.
- [3] N. Stylianides and E. Kontou, "Bayes Theorem and Its Recent Applications," *Leicester Undergrad. Math. J.*, vol. 2, March, 2020.
- [4] A. Rupnawar, A. Jagdale, and S. Navsupe, "Study on Forward Chaining and Reverse Chaining in Expert System," *Int. J. Adv. Eng. Res. Sci.*, vol. 3, no. 12, pp. 60–62, 2016.
- [5] Departemen Service Roda Dua, *Suzuki Satria F150 Pedoman Perbaikan*. Jakarta: PT Indomobil Suzuki Internasional, 2013.
- [6] J. Joyce, "Bayes' Theorem," in *The Stanford Encyclopedia of Philosophy*, Spring 2019, Editor: E. N. Zalta, Metaphysics Research Lab, Stanford University, 2019.
- [7] H. Kido, "Bayesian Entailment Hypothesis: How Brains Implement Monotonic and Non-monotonic Reasoning," *Comput. Sci.*, p. 7, 2020.
- [8] S. Kusumadewi, *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu, 2003.
- [9] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th Ed. New Jersey: Prentice Hall, 2020.
- [10] T. Tirkaamiasa and W. Usino, "Sistem Pakar Pembagian Waris Menggunakan Metode Forward dan Backward Chaining," *Jurnal Krea-TIF*, vol. 03 no. 02, p. 67, 2015.
- [11] M. Chein and M.-L. Mugnier, *Graph-based Knowledge Representation*, 1st ed. London: Springer-Verlag, 2009.