

SISTEM PAKAR DIAGNOSA KERUSAKAN MESIN SEPEDA MOTOR MATIC BERBASIS WEB MENGGUNAKAN CERTAINTY FACTOR

Dedi Saputra¹, Deasy Purwaningtias², Windi Irmayani³

^{1,2}Program Studi Sistem Informasi, Fakultas Teknologi Informasi, UBSI PSDKU Pontianak

³Program Studi Sistem Informasi Akuntansi, Fakultas Teknologi Informasi, UBSI PSDKU Pontianak
Jl. Abdurrahman Saleh No. 18 A Pontianak, Kalimantan Barat, 78124

*dedi.dst@bsi.ac.id

Abstrak

Membuat keputusan memainkan peranan penting dalam proses diagnosa kerusakan mesin sepeda motor. Umumnya diagnosa kerusakan secara efektif dan efisien ini hanya bisa dilakukan oleh pakar yang ahli pada bidangnya. Dengan ditemukannya kecerdasan buatan dalam dunia komputer membuat hal tersebut menjadi lebih mudah. Penggunaan kecerdasan buatan berbasis komputer dapat menjadi pemecahan masalah bagi pelaku pengambil keputusan untuk membuat keputusan berdasarkan kondisi yang tidak pernah dialami sebelumnya dan upaya untuk mendapatkan diagnosa tersebut. Dengan kombinasi mesin inferensi dan basis pengetahuan menjadikan lahirnya aplikasi diagnosa kerusakan ini. Aplikasi ini dilengkapi juga dengan fitur pengembangan basis pengetahuan secara bertahap dengan mengandalkan interpretasi pengetahuan pakar yang dituangkan dalam sistem tersebut membuat sistem tersebut dapat memberikan diagnosa dengan baik dan akurat. Dalam usaha untuk mencapai penelitian, aplikasi ini menggunakan metode *Certainty Factor* untuk memperoleh diagnosa dapat memproses kemunculan kondisi tidak terstruktur. Selain itu, penerapan metode ini pada basis aturan menghasilkan derajat keyakinan terhadap masing-masing diagnosa kerusakan.

Kata kunci : Sistem Pakar, Diagnosa, *Certainty Factor*

Abstract

Making decision has an important role in diagnosis machine damage process. Generally, damage diagnosis only can be used effectively and efficiently by the expert. Founding artificial intelligence in computer world makes it become easy. Using artificial intelligence can be a solution for decision maker to make decision based on unexpected situation and an effort to get the diagnose. The combination of the inference machine and knowledge-based creates this machine damage diagnosis application. The application is complete by the develop future based on knowledge gradually by using expert interpretation which shown in the system. That's why the system can give a good and accurate result. In the way to reach observation, this application uses *Certainty Factor* to get diagnosis which can process the appearabce of unstructural condition. Beside that, the implementation of this method creates the conviction of damage diagnosis.

Keywords : Expert System, Artificial Intellegent, *Certainty Factor*

PENDAHULUAN

Sepeda Motor merupakan kendaraan yang banyak digunakan oleh orang Indonesia. Dibandingkan dengan mobil, motor sendiri memiliki tingkat fleksibilitas lebih tinggi dilihat dari kemudahan penggunaan dan ukuran

kendaraan. Dengan berkembangnya teknologi otomotif lahirlah berbagai macam jenis motor. Selain motor bebek dan motor *sport* kini banyak orang lebih membeli dan menggunakan motor *matic*. Hal ini didukung dengan data pada Juni 2016 oleh Asosiasi Industri Sepedamotor

Indonesia (AIS) yang menyebutkan “*market share* skutik (*matic*) masih sangat jauh meninggalkan segmen lain, dengan penguasaan sebesar 69,31 persen”. Motor *matic* sendiri merupakan wujud perkembangan otomotif yang selalu berinovasi dengan teknologi-teknologi terbaru. Dibandingkan motor bebek dan motor *sport*, motor *matic* memiliki beberapa keunggulan seperti mudah mengoperasikannya serta fitur-fitur tambahan dengan teknologi-teknologi mutakhir yang tidak ada pada motor bebek dan motor *sport*. Teknologi-teknologi mutakhir inilah yang menjadi keunggulan motor *matic* merebut perhatian para pengguna sepeda motor.

Dikarenakan perbedaan teknologi baik dari mesin maupun fitur-fitur yang tertanam dalam motor *matic*, banyak pengguna yang terbiasa menggunakan sepeda motor bebek merasa kebingungan ketika mesin motor *matic* mereka mengalami kerusakan. Akhirnya pengendara motor *matic* menyerahkan sepenuhnya kepada mekanik untuk dilakukan pemeriksaan. Pengendara sepeda motor *matic* menjadi cenderung tidak peduli apakah kerusakan tersebut hanya diakibatkan oleh masalah yang sederhana atau rumit untuk diperbaiki.

Umumnya sebelum terjadi kerusakan pada sepeda motor *matic* yang digunakannya, pengendara akan merasakan gejala yang tidak biasa pada sepeda motor tersebut. Pengendara merasakan kejanggalan atau ketidaknyamanan pada sepeda motor yang dikendarainya. Jika pengendara dapat memprediksi kerusakan yang terjadi berdasarkan gejala yang sudah ada, maka tentu kerusakan dapat dihindari sesegera mungkin dan dilakukan perbaikan. Pemasalahannya tidak semua orang memiliki pengetahuan tentang mesin sepeda motor *matic* layaknya para mekanik.

Sistem pakar yang mengusul buat adalah solusi untuk permasalahan yang ada, dimana sistem ini mampu mendiagnosa kerusakan mesin pada sepeda motor khusus berjenis *matic*. Sistem ini merupakan representasi pengetahuan beberapa mekanik yang tersertifikasi dan dituangkan dalam bentuk perangkat lunak sehingga dapat digunakan oleh para pengendara untuk memprediksi kerusakan yang akan terjadi.

Diharapkan dengan sistem pakar diagnosa kerusakan mesin sepeda motor *matic* ini dapat membantu para pengendara mendiagnosa

kerusakan yang terjadi pada sepeda motor *matic* mereka sehingga dapat dilakukan perbaikan segera mungkin. Sistem ini juga menambah pengetahuan sehingga dapat membantu pengendara untuk lebih mengerti mengenai mesin sepeda motor *matic* mereka.

METODE

Lokasi dan Waktu Penelitian

Penelitian ini dilakukan pada unit kerja bengkel resmi Yamaha di Pontianak. Waktu penelitian selama 40 minggu. Pengumpulan data selama 8 minggu, analisis data selama 4 minggu, model perancangan selama 8 minggu, melakukan *coding* 16 minggu, implementasi selama 4 minggu.

Jenis dan Sumber Data

Jenis data yang digunakan pada penelitian ini berupa data primer dan data sekunder dengan tujuan untuk mendapatkan data yang akurat. Data primer didapat dari wawancara teknisi yang telah disertifikasi oleh Yamaha dan data sekunder didapat dari arsip-arsip resmi dari Yamaha.

Teknik Pengumpulan Data

Data sekunder yang digunakan dalam penelitian ini diperoleh dengan menggunakan teknik pengumpulan data sebagai berikut:

1. Observasi
Data-data diperoleh melalui arsip-arsip resmi tentang sistem pakar yang diperoleh dari internet. Selain dari internet data-data juga diperoleh dari data-data statistik berupa jurnal atau buku yang telah dipublikasikan.
2. Wawancara
Mewawancarai para teknisi motor *matic* yang telah disertifikasi oleh Yamaha.
3. Studi Pustaka
Data-data diperoleh melalui arsip-arsip resmi tentang sistem pakar yang diperoleh dari para ahli pada bengkel resmi Yamaha. Selain itu data-data juga diperoleh dari data-data berupa jurnal atau buku yang telah dipublikasikan.
Penulis melakukan penghimpunan informasi yang relevan dengan penelitian yang dilakukan oleh penulis. Setelah memperoleh data-data melalui studi pustaka penulis melakukan kajian terhadap fakta-fakta melalui buku-buku

sepeda motor maupun artikel-artikel seputar mesin sepeda motor.

Model Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan dalam penelitian menggunakan metode air terjun (*waterfall*). Tujuan penggunaan metode tersebut bertujuan menjaga waktu pengerjaan perangkat lunak dengan waktu sesingkat mungkin tanpa melupakan kualitas.

Beberapa tahapan dalam metode air terjun (*waterfall*) yaitu sebagai berikut :

1. *Communication (Project initiation&Requirements)*
Tahapan awal dari dimulainya metode air terjun (*waterfall*). Tahapan ini dipergunakan untuk memperoleh fakta-fakta yang ada. Data-data yang telah diperoleh kemudian dipelajari dan dianalisa untuk dijadikan sumber penelitian.
2. *Planning (Estimating, scheduling and tracking)*
Data-data penelitian yang telah diperoleh sebelumnya direpresentasikan dalam bentuk pengetahuan. Kebutuhan penggunaannya disesuaikan dengan fitur-fitur yang akan dibangun.
3. *Modeling (Analysis and design)*
Tahapan ini yaitu mengubah kebutuhan perangkat lunak menjadi sebuah model rancangan yang akan dipergunakan untuk generasi kode.
4. *Construction (Code and test)*
Dalam tahapan konstruksi ini, dilakukan generase kode melalui bahasa pemograman PHP, *Hyper Text Markup Language* (HTML), *Javascript* dan *Cascading Style Sheet* (CSS) . Kode-kode ini akan membentuk bagian-bagian program yang saling terhubung sesuai dengan fitur dan fungsi perangkat yang telah didefinisikan pada tahapan sebelumnya.
5. *Deployment (Delivery, support and feedback)*
Setelah tahapan konstruksi, data-data hasil penelitian dalam bentuk pengetahuan direpresentasikan dalam bentuk sistem secara bertahap, dimana setiap tahapannya dilakukan uji kasus untuk mengetahui kesesuaian hasil secara fungsional.

HASIL DAN PEMBAHASAN

Identifikasi Kebutuhan

Identifikasi dibedakan menjadi dua bagian sebagai berikut :

1. Kebutuhan Fungsional
Kebutuhan fungsional merupakan kebutuhan utama yang diharapkan sistem yang berkaitan langsung dengan sistem yang dibuat. Adapun kebutuhan fungsional yang dibutuhkan agar sistem dapat melakukan pengolahan data-data masukkan dan menghasilkan keluaran yang dibutuhkan adalah antara lain :
 - a. Prosedur *login*
Merupakan suatu proses yang harus dilakukan pada awal menggunakan aplikasi oleh *administrator*. Prosedur *login* dapat dilakukan dengan memasukkan *username* dan *password*. Jika *username* dan *password* cocok maka sistem akan mengarahkan *administrator* dari halaman *login* ke halaman *review* pengunjung.
 - b. Diagnosa Kerusakan
Sebuah fitur diagnosa penyakit dimana pengguna dapat melihat keluaran berdasarkan gejala yang telah dipilih secara simultan. Data gejala dan penyakit yang digunakan dalam diagnosa ini berdasarkan pada pakar yang telah dipilih oleh pengunjung. Dari gejala-gejala yang telah dimasukkan oleh pengunjung maka sistem akan melakukan proses penalaran dan kemudian akan menyajikan semua kemungkinan yang mungkin terjadi berdasarkan gejala-gejala yang ada. Data keluaran tersebut diurutkan berdasarkan ranking.
 - c. Statistik dan *review* pengunjung
Fitur ini merupakan sebuah umpan balik kepada para *administrator*. Dikatakan sebagai umpan balik karena setiap pengunjung melakukan diagnosa penyakit maka pada akhir diagnosa pengunjung dapat memberikan umpan balik beserta *rating* terhadap *administratror* tersebut. Sehingga *administrator* dapat melakukan perbaikan terutama pada pemutakhiran data-data basis pengetahuan untuk membuat diagnosa tersebut menjadi lebih tepat.
 - d. Fasilitas Penjelasan

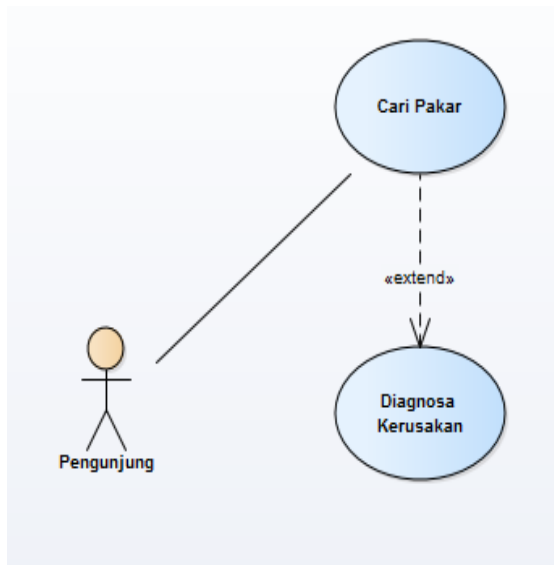
- Fasilitas penjelasan merupakan sebuah fitur yang dibutuhkan untuk mendukung pengambilan keputusan. Fasilitas penjelasan ini didasarkan pada gejala-gejala yang diterima oleh sistem menjelaskan bagaimana sebuah keputusan diperoleh dan juga menampilkan nilai kemungkinan dari suatu konklusi melalui metode *Certainty Factor* (CF). Nilai CF dikalkulasikan berdasarkan jawaban pengguna dengan nilai CF pada setiap aturan yang telah ada sebelumnya melalui rumus persamaan yang telah diprogramkan sebelumnya.
- e. Pengelolaan basis pengetahuan
- Basis pengetahuan merupakan hal yang menjadi tumpuan pada aplikasi ini. Data-data basis pengetahuan digunakan sebagai acuan pada saat diagnosa dijalankan. Pada prosedur ini terdapat fitur penambahan, penyuntingan, penghapusan serta konfigurasi aturan dari data-data yang akan digunakan. Data-data yang digunakan dalam prosedur ini terdiri dari data penyakit, data detail-penyakit, data gejala dan data aturan. Data-data tersebut dimasukkan oleh pengguna akan divalidasi dahulu oleh program. Jika telah benar maka program akan melakukan *entry* data tersebut pada *database* sesuai dengan tiru masing-masing. Apabila validasi gagal maka aplikasi akan memberitahu kepada pengguna untuk melakukan perbaikan pada masukan tersebut.
2. Kebutuhan Non Fungsional
- Kebutuhan non fungsional meliputi kebutuhan penunjang atau kebutuhan yang menjadi pendukung dari sistem yang dibuat, adapun kebutuhan tersebut antara lain :
- a. Kebutuhan Perangkat Keras
- Agar aplikasi ini dapat berjalan dengan baik maka dibutuhkan minimal spesifikasi perangkat keras untuk menunjang kinerja sistem. Adapun kebutuhan *hardware* yang diperlukan adalah sebagai berikut :
- 1.) CPU (*Central Processing Unit*)
 - a.) *Processor* : 1.4Ghz
 - b.) *RAM* : 2 GB DDR3
 - c.) *Harddisk* : 20 GB
 - 2.) *Monitor* (14" WXGA LCD / LED 1028x768)

- 3.) *Keyboard* (*Standard Keyboard* 86 keys)
 - 4.) *Mouse*
 - b. Kebutuhan Perangkat Lunak
- Adapun kebutuhan perangkat lunak yang diperlukan agar perangkat lunak dapat berjalan yaitu :
- 1.) *Sistem Operasi* : *Windows* 8.1
 - 2.) *Web server* : *XAMPP* 3.2.1
 - 3.) *Web Browser* : *Mozilla Firefox* atau *Google Chrome*
 - 4.) *DBMS* : *MySQL*
 - c. Kebutuhan *Brainware*
- Brainware* atau pengguna dalam aplikasi ini dapat dibedakan menjadi 2 pengguna yaitu : Pengguna Umum dan *Administrator*. Pengguna umum didefinisikan sebagai pengguna yang dapat menggunakan aplikasi tanpa melalui prosedur *login*. Berbeda dengan *administrator* yang harus melewati prosedur *login*. Pembagian kategori *Brainware* dibagi berdasarkan hak aksesnya sebagai berikut
- 1.) Pengguna umum
- Pengguna umum dapat memanfaatkan fitur-fitur yang disediakan pada aplikasi ini seperti melakukan diagnosa kerusakan dengan melakukan interaksi antarmuka program. Setelah melakukan diagnosa, pengguna umum dapat melihat hasil kerusakan yang telah disimpulkan terhadap gejala-gejala yang dipilih. Setelah selesai pengguna dapat mengulangi kembali dari awal untuk melakukan diagnosa kerusakan.
- 2.) *Administrator*
- Administrator* merupakan pengguna yang dibutuhkan untuk menjalankan aplikasi ini setelah melewati proses *login*. Fitur-fitur yang disediakan dalam *administrator* seperti melakukan pengolahan data penyakit, gejala, aturan (*rule CF*), serta melihat statistik pengunjung dan *review* yang diberikan pengunjung guna sebagai perbaikan untuk *administrator* kedepan.

Use Case Diagram

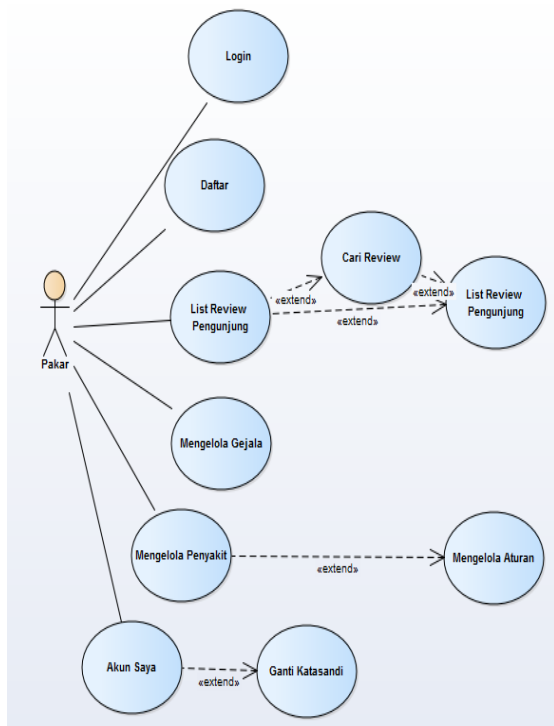
Use Case Diagram digunakan untuk menunjukkan hubungan antara pengguna atau *actor* yang akan berinteraksi dengan sistem yang akan dibuat. *Use Case Diagram* mendeskripsikan interaksi antara pengguna dengan sistem. Gambar dibawah ini menjelaskan mengenai interaksi antara pengguna dengan sistem yang akan dibuat

1. *Use Case* Pengunjung



Gambar 1. Use Case Pengunjung

2. Use Case Administrator atau Pakar

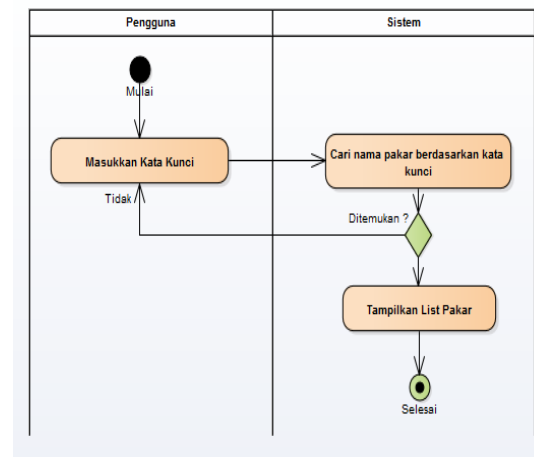


Gambar 2. Use Case Administrator atau Pakar

Activity Diagram

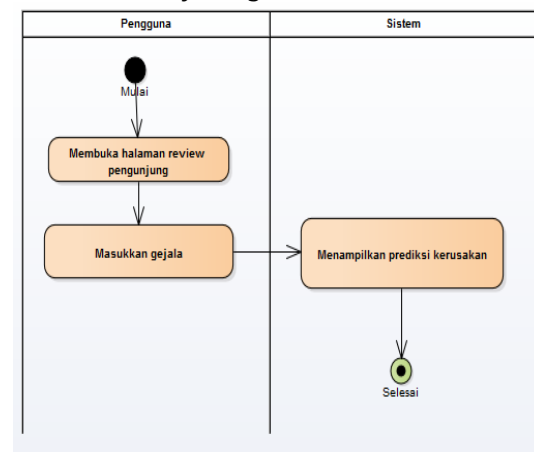
Activity Diagram menggambarkan aliran kerja dari sistem atau proses bisnis yang ada pada perangkat lunak. Berikut ada *activity diagram* dengan sistem yang akan dibuat:

1. Activity Diagram Cari Pakar



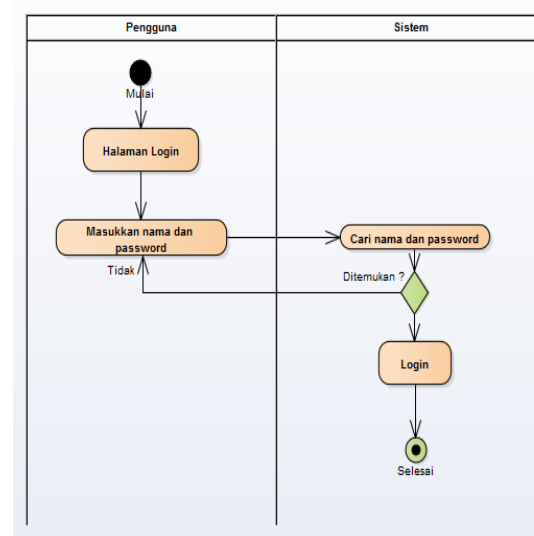
Gambar 3. Activity Diagram Cari Pakar

2. Activity Diagram Analisa



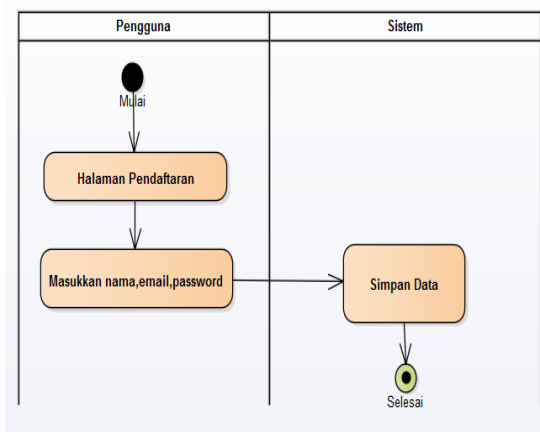
Gambar 4. Activity Diagram Analisa

3. Activity Diagram Login



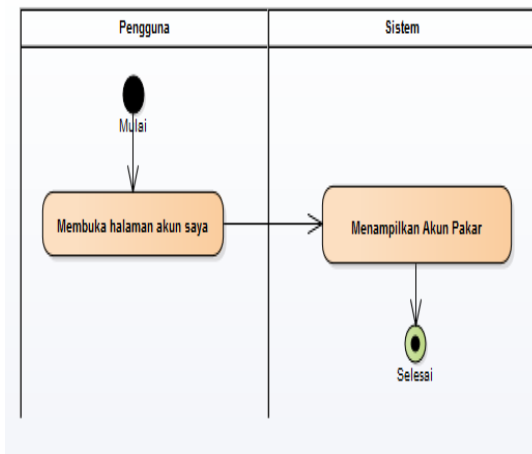
Gambar 5. Activity Diagram Login

4. Activity Diagram Daftar



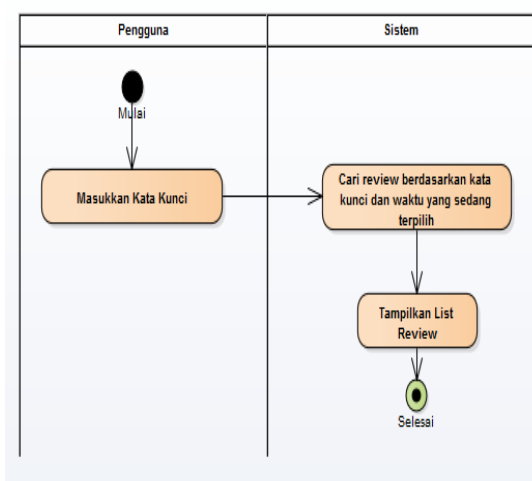
Gambar 6. Activity Diagram Daftar

5. Activity Diagram Akun Saya



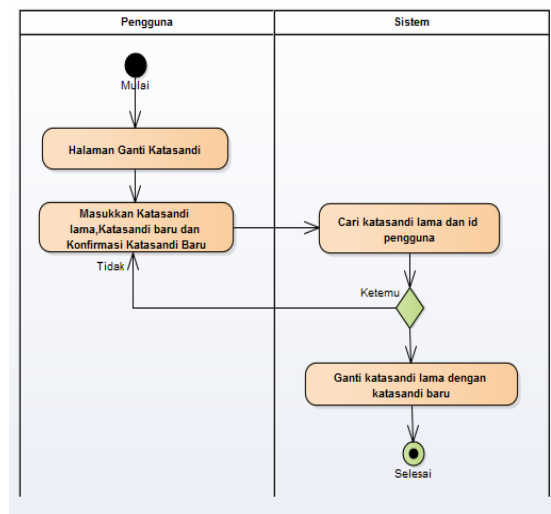
Gambar 7. Activity Diagram Akun Saya

6. Activity Diagram Cari Review Pengunjung



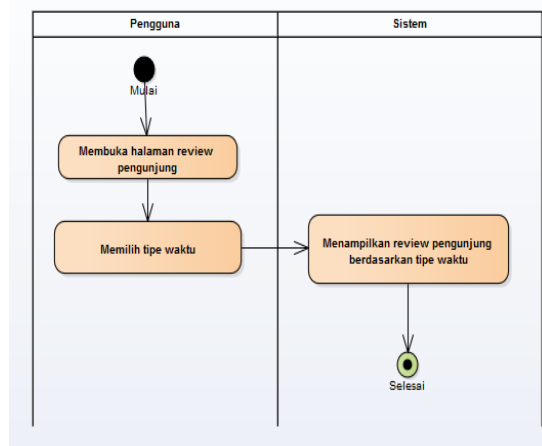
Gambar 8. Activity Diagram Cari Review Pengunjung

7. Activity Diagram Ganti Katasandi



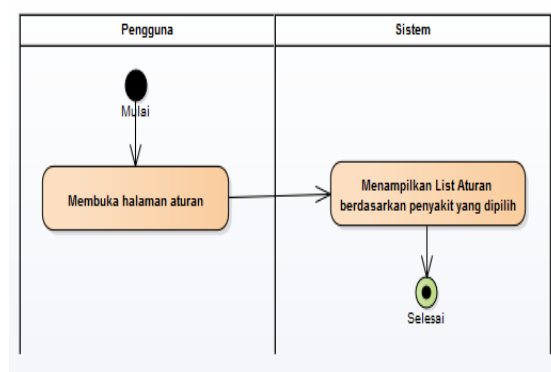
Gambar 9. Activity Diagram Ganti Katasandi

8. Activity Diagram List Review Pengunjung



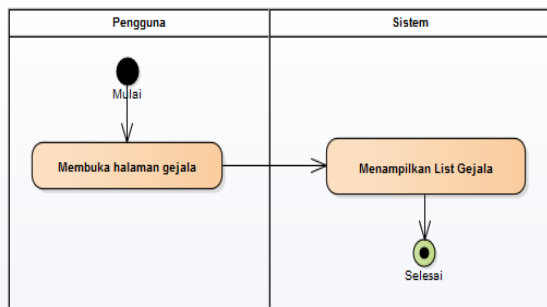
Gambar 10. Activity Diagram List Review Pengunjung

9. Activity Diagram Mengelola Aturan



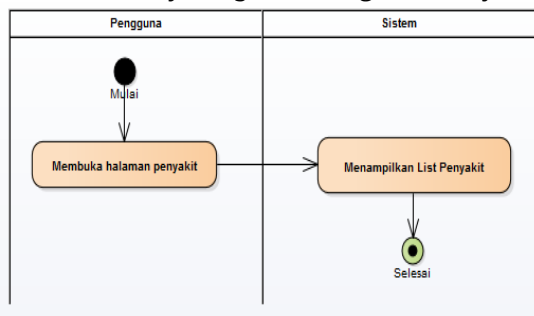
Gambar 11. Activity Diagram Mengelola Aturan

10. Activity Diagram Mengelola Gejala



Gambar 12. Activity Diagram Mengelola Gejala

11. Activity Diagram Mengelola Penyakit

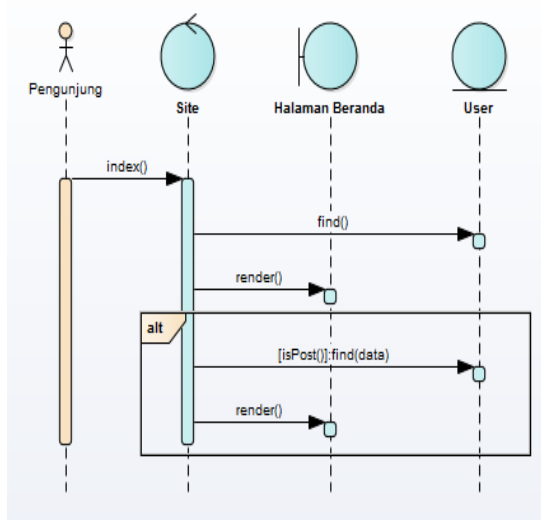


Gambar 13 Activity Diagram Mengelola Penyakit

A. Sequence Diagram

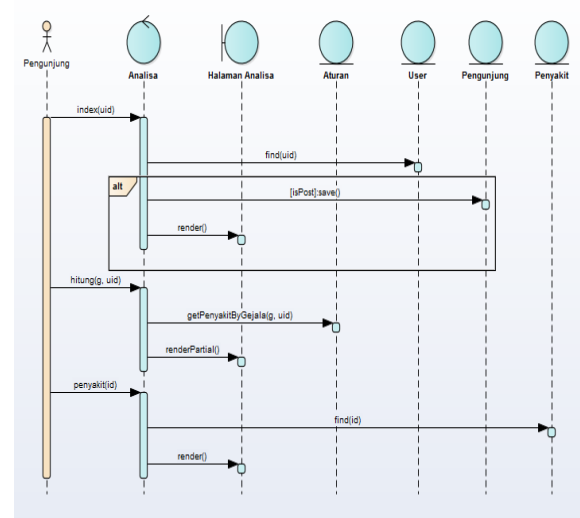
Sequence Diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima oleh objek. Berikut adalah sequence diagram dengan sistem yang akan :

1. Sequence Diagram Cari Pakar



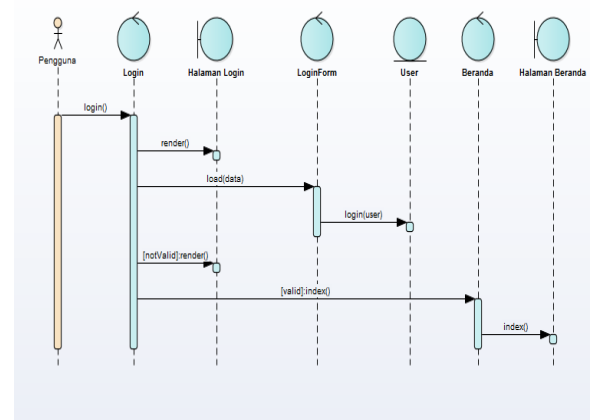
Gambar 14. Sequence Diagram Cari Pakar

2. Sequence Diagram Analisa



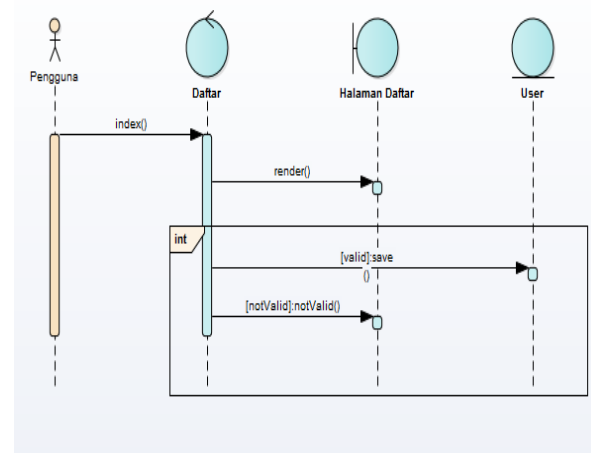
Gambar 15 Sequence Diagram Analisa

3. Sequence Diagram Login



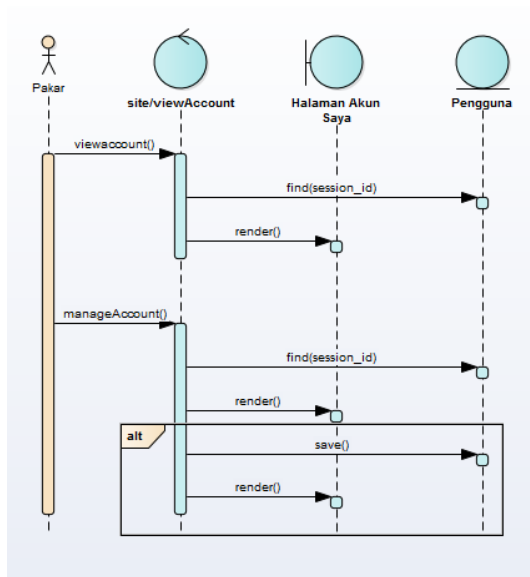
Gambar 16 Sequence Diagram Login

4. Sequence Diagram Daftar



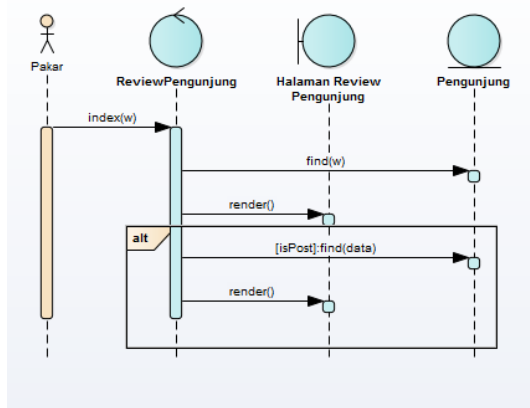
Gambar 17. Sequence Diagram Daftar

5. Sequence Diagram Akun Saya



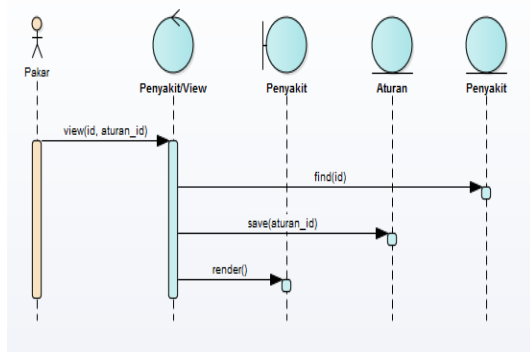
Gambar III.18. Sequence Diagram Akun Saya

6. Sequence Diagram List Review Pengunjung



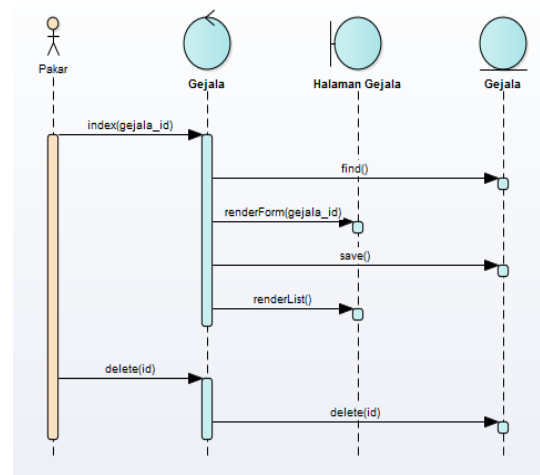
Gambar 19. Sequence Diagram List Review Pengunjung

7. Sequence Diagram Mengelola Aturan



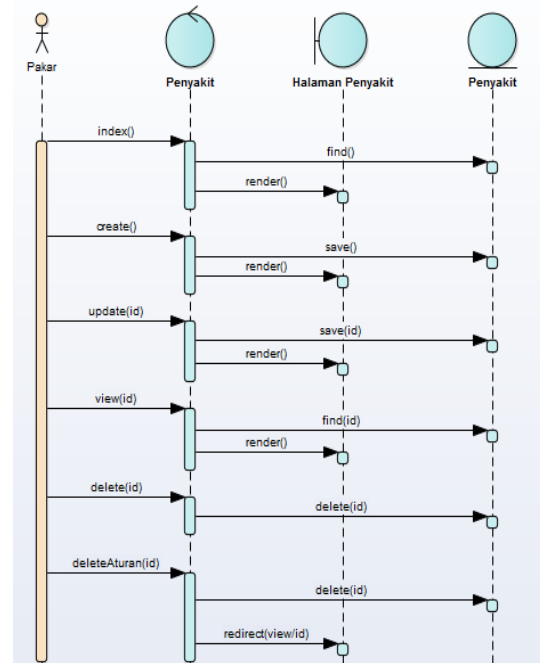
Gambar 20. Sequence Diagram Mengelola Aturan

8. Sequence Diagram Gejala



Gambar 21 Sequence Diagram Gejala

9. Sequence Diagram Penyakit



Gambar 22. Sequence Diagram Penyakit

B. Perancangan Perangkat Lunak

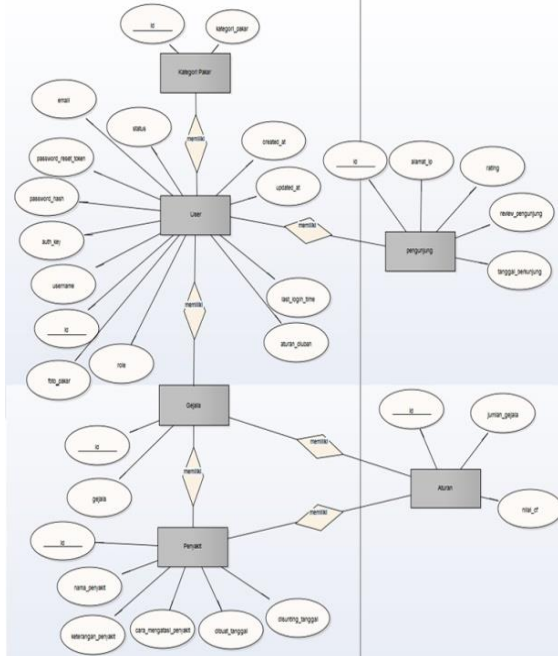
Perancangan perangkat lunak yang diperlukan untuk membangun aplikasi ini yaitu sebagai berikut :

Rancangan Basis Data

Dalam menulis laporan Tugas Akhir untuk menggambarkan struktur basis data pada aplikasi ini digunakan beberapa alat bantu perancangan program seperti ERD (Entity Relationship Diagram) dan LRS(Logical Record Structure).

1. ERD (Entity Relationship Diagram)

Dalam gambar dibawah ini dijelaskan entitas dengan entitas lain pada aplikasi ini saling berkaitan antara yang satu dengan yang lain. Dimana tiap-tiap entitas merepresentasikan tabel pada basis data yang digunakan.



Gambar 23. Entity Relationship Diagram(ERD)

2. LRS (Logical Record Structure)

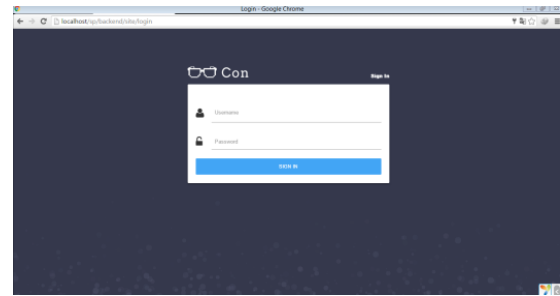


Gambar 24 Logical Record Structure (LRS)

C. Implementasi Rancangan Antar Muka Administrator.

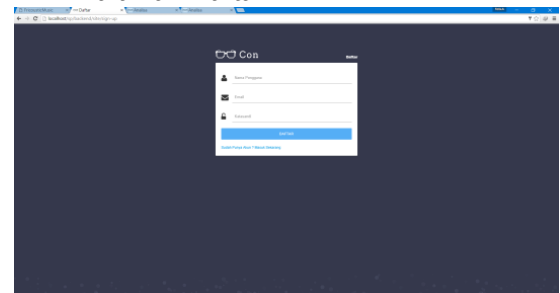
Berbeda dengan pengunjung yang tidak memerlukan prosedur *login*. Halaman *administrator* memerlukan prosedur *login* untuk mengakses halaman tersebut. Berikut adalah implementasi rancangan antar muka *administrator* :

1. Implementasi Rancangan Antar Muka Halaman Login



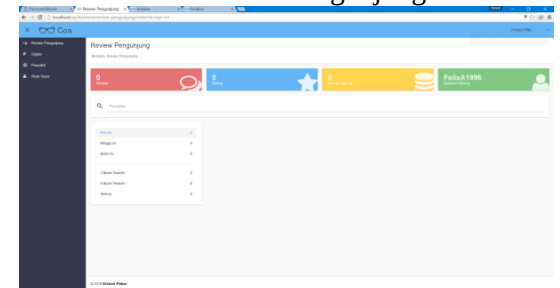
Gambar 25. Implementasi Rancangan Antar Muka Halaman Login

2. Implementasi Rancangan Antar Muka Halaman Daftar



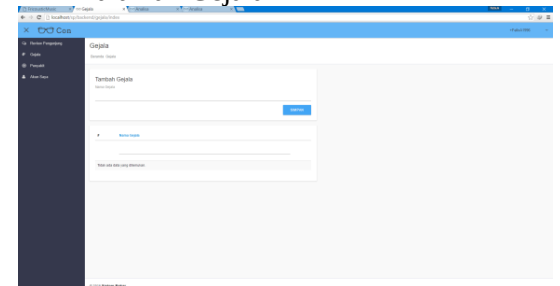
Gambar 26. Implementasi Rancangan Antar Muka Halaman Daftar

3. Implementasi Rancangan Antar Muka Halaman Review Pengunjung



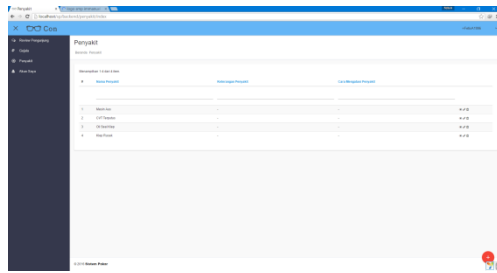
Gambar 27. Implementasi Rancangan Antar Muka Halaman Review Pengunjung

4. Implementasi Rancangan Antar Muka Halaman Gejala

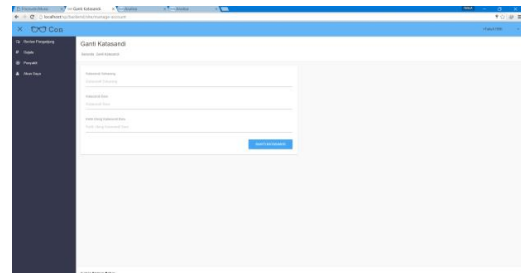


Gambar 28. Implementasi Rancangan Antar Muka Halaman Gejala

5. Implementasi Rancangan Antar Muka Halaman Penyakit

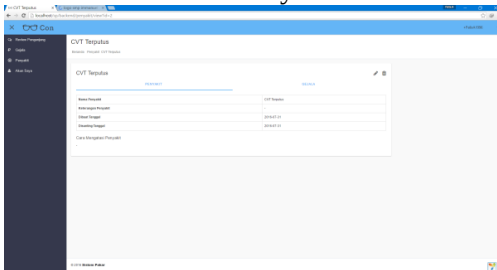


Gambar 29. Implementasi Rancangan Antar Muka Halaman Penyakit

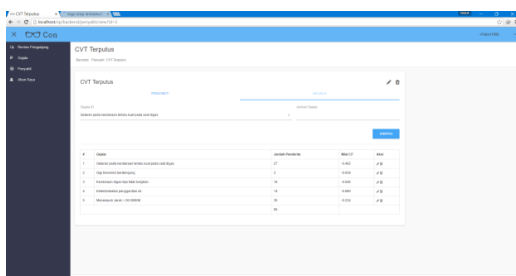


Gambar 33. Implementasi Rancangan Antar Muka Halaman Ganti Kata sandi

6. Implementasi Rancangan Antar Muka Halaman Detail Penyakit

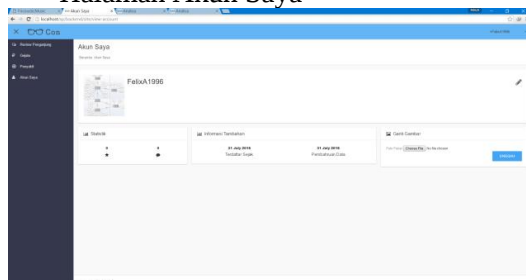


Gambar 30. Implementasi Rancangan Antar Muka Halaman Detail Penyakit Tab Penyakit



Gambar 31. Implementasi Rancangan Antar Muka Halaman Detail Penyakit Tab Gejala

7. Implementasi Rancangan Antar Muka Halaman Akun Saya



Gambar 32. Implementasi Rancangan Antar Muka Halaman Akun Saya

8. Implementasi Rancangan Antar Muka Halaman Ganti Kata sandi

SIMPULAN DAN SARAN

Kesimpulan

Penelitian ini menghasilkan perangkat lunak yang dipergunakan untuk melakukan diagnosa kerusakan pada mesin sepeda motor *matic* menggunakan XAMPP sebagai *web server*, MySQL sebagai basis data, PHP sebagai bahasa pemrograman. Kelengkapan basis pengetahuan yang ditanamkan dalam aplikasi sangat mempengaruhi hasil diagnosa yang ada. Serta penerapan metode *Certainty Factor* pada aplikasi ini membantu memberikan penjelasan terhadap hasil diagnosa berdasarkan derajat keyakinan dari setiap aturan yang ada jika diberikan gejala-gejala tertentu.

Aplikasi ini digunakan untuk melakukan diagnosa kerusakan berdasarkan basis pengetahuan yang telah ditanamkan serta tidak dapat memberikan penalaran diluar batas data yang telah ditanamkan sebelumnya.

Saran

Pengimplementasian derajat keyakinan (CF) akan lebih akurat jika para pakar menanamkan pengetahuan mereka dalam aplikasi ini. Sehingga hasil diagnosa kerusakan dapat menjadi lebih akurat berdasarkan gejala-gejala yang dipilih. Untuk dapat meningkatkan keakuratan diagnosa aplikasi dapat dipadukan dengan Sistem Diagnosa Kerusakan Elektronik, sehingga dengan adanya integrasi tersebut aplikasi dapat memperoleh basis pengetahuan dari mesin tersebut dan dapat menunjang proses pengambilan keputusan dalam diagnosa kerusakan tersebut.

UCAPAN TERIMA KASIH

Jurnal ini merupakan salah satu keluaran dari hibah Penelitian Dosen Pemula 2017 untuk tahun pendanaan 2018, maka dari itu penulis mengucapkan terima kasih banyak kepada

Kementerian Riset, Teknologi, dan Pendidikan Tinggi.

DAFTAR PUSTAKA

Kusrini. 2008. *Aplikasi Sistem Pakar Menentukan Faktor Kepastian Pengguna dengan Metode Kuantifikasi Pertanyaan*. Yogyakarta: Andi Offset.p
menggunakan-metode.html (2016)

Rosnelly, Rika. 2012. *Sistem Pakar Konsep dan Teori*. Diambil dari:

https://books.google.co.id/books/about/Sistem_Pakar.html?id=WrOACwAAQBAJ&redir_esc=y. (03 Mei 2016)

<http://www.metode-algoritma.com/2013/06/aplikasi-system-pakar->