

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN
MESIN *FOTOCOPY* SHARP MX-M264N
BERBASIS *WEB* DENGAN METODE
*FORWARD CHAINING***

SKRIPSI



**Oleh:
Alex Darwin
130210004**

**PROGRAM STUDI TEKNIK INFORMATIKA
UNIVERSITAS PUTERA BATAM
2017**

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN
MESIN *FOTOCOPY* SHARP MX-M264N
BERBASIS *WEB* DENGAN METODE
*FORWARD CHAINING***

SKRIPSI
Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana



Oleh
Alex Darwin
130210004

**PPROGRAM STUDI TEKNIK INFORMATIKA
UNIVERSITAS PUTERA BATAM
2017**

PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (sarjana, dan/atau magister), baik di Universitas Putera Batam maupun di perguruan tinggi lain.
2. Skripsi ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi.

Batam, 11 Februari 2017

Yang membuat pernyataan,

Alex Darwin
130210004

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN
MESIN *FOTOCOPY* SHARP MX-M264N
BERBASIS *WEB* DENGAN METODE
*FORWARD CHAINING***

Oleh
Alex Darwin
130210004

SKRIPSI
Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana

Telah disetujui oleh Pembimbing pada tanggal
seperti tertera di bawah ini

Batam, 11 Februari 2017

Pastima Simanjuntak, S.Kom., M.SI.
Pembimbing

ABSTRAK

Pesatnya perkembangan teknologi di era maju ini, mesin *fotocopy* berperan penting dan signifikan untuk setiap perusahaan, kantor, maupun usaha *printing*. Pengetahuan tentang mesin *fotocopy* sangat dibutuhkan dalam menjalani usaha mesin *fotocopy* atau kantor agar dapat berjalan dan berkembang lebih pesat. Namun bagi orang yang baru memulai usaha *fotocopy* mengalami kesulitan bila terjadi kerusakan pada mesin *fotocopy*-nya. Keterbatasan buku panduan tentang mesin *fotocopy* SHARP MX-M264N menjadi salah satu faktor pemilik *fotocopy* yang tidak dapat menguasai dan mengetahui komponen mesin *fotocopy* tersebut. Belum adanya suatu sistem yang mampu membantu pengguna untuk memperbaiki kerusakan pada mesin *fotocopy* tersebut. Sulitnya berkonsultasi dengan teknisi mengenai kerusakan yang terjadi pada mesin *fotocopy*. Keterbatasan akan teknisi mesin *fotocopy* yang profesional dalam merk SHARP. Pembuatan aplikasi sistem pakar ini untuk memudahkan pengguna dalam mengetahui informasi lebih cepat mengenai berbagai macam atau jenis kerusakan pada mesin *fotocopy* SHARP MX-M264N. Aplikasi sistem pakar berbasis *web* ini dirancang dengan menggunakan bahasa pemrograman *PHP* dan *MySQL* untuk membuat *database*. Dalam proses penarikan kesimpulan, sistem pada aplikasi menggunakan metode *Forward Chaining* dimana gejala kerusakan dan aturan algoritma ditentukan oleh pakar dan pengguna hanya memilih fakta yang menurutnya merupakan gejala kerusakan pada mesin *fotocopy* yang dialami. Hasil yang diperoleh dari pembuatan aplikasi ini bahwa informasi tentang kerusakan terkait dapat lebih mudah diperoleh dengan dibuatnya suatu sistem pakar untuk mendiagnosa kerusakan pada mesin *fotocopy* SHARP MX-M264N dan dapat digunakan serta dipelajari dengan mudah oleh pengguna.

Kata kunci: Sistem Pakar, Mesin *fotocopy* SHARP MX-M264N, Metode *Forward Chaining*, *WEB*, *PHP*, *MySQL*

ABSTRACT

The rapid development of technology in this advanced era, photocopiers important and significant role for any company, office, or business printing. Knowledge of photocopiers is needed in running a business or an office copy machine to run and grow more rapidly. But for people who are just starting a business copier experiencing difficulties when there is damage to his copy machine. Limitations of manual books on copying machines SHARP MX-M264N be one factor photocopy owners who can not master and identify the components of the copy machine. The absence of a system that can help users to repair damage to the copy machine. Difficult to consult with the technician about the damage that occurs in a copy machine. Limitations will photocopiers professional technicians in the brand SHARP. Making this expert system application to allow users to quickly find out more information about the various kinds or types of damage to the copying machines SHARP MX-M264N. Web-based expert system application is designed by using the programming language PHP and MySQL to create a database. In the process of drawing conclusions, on the application system using the method in which the symptoms of damage Forward Chaining algorithms and rules determined by experts and users simply select a fact which he said is a symptom of damage to the copy machine experienced. Results obtained from the making of this application that the relevant information about the crash can be more easily obtained with the making of an expert system for diagnosing damage to the copying machines SHARP MX-M264N and can be used and studied easily by the user.

Keywords: Expert System, Photocopiers SHARP MX-M264N, Forward Chaining method, WEB, PHP, MySQL

KATA PENGANTAR

Puji syukur kehadirat Sang Triratna, Sang Buddha dan Bodhisattva, yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati.

Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Teknik Informatika Universitas Putera Batam.
3. Ibu Pastima Simanjuntak, S.Kom., M.SI. selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam.
5. Steven dan staff PT Syindo Mulia Perkasa.
6. Kepada kedua orang tua penulis, yang terus mendoakan keberhasilan penulis menyelesaikan skripsi ini.

7. Teman-teman seperjuangan yang juga selalu memberikan motivasi baik berupa *sharing* pendapat, motivasi dan hal-hal lainnya dalam rangka pembuatan skripsi ini.
8. Serta semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu dalam penyusunan skripsi ini.

Semoga Sang Triratna, Sang Buddha dan Bodhisattva membalas kebaikan dan selalu mencurahkan hidayah serta taufik-Nya, Amin.

Batam, Februari 2017

Penulis

1 BAB I PENDAHULUAN

1.1 Latar Belakang Penelitian

Pesatnya perkembangan teknologi di era maju ini, mesin *fotocopy* berperan penting dan signifikan untuk setiap perusahaan, kantor, maupun usaha *printing*. Spesifikasi dan fitur dari mesin-mesin *fotocopy* berubah dengan cepat dirangsang oleh faktor lingkungan dan ekonomi. Masalah ini menyajikan sebuah tantangan bagi teknisi mesin *fotocopy* dan penggunanya dalam penanganan kerusakan mesin *fotocopy* maupun malfungsi. Dengan banyaknya perubahan fitur yang kompleks pada mesin *fotocopy* digital yang perlu dipelajari, maka keberhasilan untuk menemukan kesalahan sangat tergantung pada keahlian individu. Namun, ketergantungan pada para ahli dapat diminimalkan dengan menyimpan dan mendokumentasikan keahlian ke dalam aplikasi komputer.

Pengetahuan tentang mesin *fotocopy* sangat dibutuhkan dalam menjalani usaha mesin *fotocopy* atau kantor agar dapat berjalan dan berkembang lebih pesat. Namun bagi orang yang baru memulai usaha *fotocopy* mengalami kesulitan bila terjadi kerusakan, pemilik *fotocopy* tidak dapat memperbaikinya dengan alasan kekurangan pengetahuan tentang mesin *fotocopy* itu sendiri. Keterbatasan buku panduan tentang mesin *fotocopy* SHARP MX-M264N menjadi salah satu faktor pemilik *fotocopy* yang tidak dapat menguasai dan mengetahui komponen mesin *fotocopy* tersebut. Selain keterbatasan buku tentang mesin *fotocopy* SHARP MX-M264N, juga belum terdapat sebuah sistem yang mampu membantu pengguna untuk mengetahui kerusakan mesin *fotocopy* tersebut. Kesulitan berkonsultasi dengan teknisi mesin *fotocopy* SHARP juga merupakan suatu permasalahan

yang sering dialami pengguna. Dan keterbatasan akan teknisi mesin *fotocopy* yang profesional dalam merk SHARP masih menjadi faktor bagi pengguna untuk berkonsultasi.

Mesin *fotocopy* merupakan perangkat elektromekanika yang berfungsi untuk menggandakan *master document* atau *file*. Mesin *fotocopy* adalah alat komputik canggih yang berfungsi untuk menyalin *image master* menjadi *image* baru di media kertas. Meskipun demikian mesin ini mengkhususkan diri untuk industri *digital printing* dan jasa dokumentasi. Fitur yang paling mencolok adalah kemampuan untuk memainkan peran penting dalam setiap proses dokumentasi.

Sistem pakar adalah bagian dari kecerdasan buatan yang berisi kombinasi pemahaman teoritis tentang suatu persoalan dan sekumpulan aturan pemecahan persoalan heuristik yang dikembangkan oleh manusia untuk dapat memecahkan problema pada suatu domain yang spesifik. Sistem pakar merupakan perangkat lunak yang digunakan untuk memecahkan masalah yang biasanya diselesaikan oleh seorang pakar. Aturan-aturan di dalamnya memberitahukan ke program, bagaimana ia memberlakukan informasi-informasi yang tersimpan. Berdasarkan itu program memberikan solusi-solusi atau bantuan pengambilan keputusan mengenai permasalahan tertentu, mirip dengan saran seorang pakar.

Sistem pakar ini menggunakan metode inferensi *forward chaining* (pelacakan ke depan) yaitu graf penelusuran kerusakan mesin *fotocopy*. Sehingga pada penelusuran kerusakan yaitu fakta-fakta yang dimasukkan oleh pengguna, dengan aturan yang telah disimpan dalam sistem satu demi satu sampai dapat diambil suatu kesimpulan, yaitu kerusakan mesin *fotocopy*.

Sebelum kegiatan pelaksanaan kegiatan penelitian dilaksanakan, peneliti sudah terlebih dahulu membaca dan mempelajari beberapa hasil penelitian terdahulu yang

berkaitan dengan penelitian yang sedang dilakukan untuk memudahkan proses penyelesaian penulisan ini. Penelitian terdahulu yang dipelajari oleh peneliti di dalam menyempurnakan penulisan ini dapat di lihat sebagai berikut: Penelitian Prasetyo Adi Wibowo dan Sugeng Purbawanto mengenai Sistem Pakar Untuk Mendiagnosa Kerusakan Televisi Dengan Metode *Forward Chaining* menggunakan *PHP* dan *MySQL* membahas mengenai suatu program sistem pakar untuk mendiagnosa kerusakan televisi dengan menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai *database* sistem. Penelitian Jusuf Wahyudi dan Juju Jumadi mengenai Sistem Pakar Kerusakan *Handphone Nokia 5130 XpressMusic* dengan metode *Forward Chaining* membahas mengenai sistem pakar untuk mendiagnosa kerusakan *Handphone Nokia 5130 XpressMusic* yang dirancang menggunakan metode *Forward Chaining* dan dibangun dengan bahasa pemrograman *Visual Basic 6.0*.

Berdasarkan uraian diatas, peneliti tertarik untuk melakukan penelitian dengan judul **“SISTEM PAKAR MENDIAGNOSA KERUSAKAN MESIN *FOTOCOPY SHARP MX-M264N* BERBASIS WEB DENGAN METODE *FORWARD CHAINING*”**.

1.2 Identifikasi Masalah

Berdasarkan uraian-uraian diatas, maka peneliti menguraikan permasalahan yang ada sebagai berikut:

1. Banyaknya pengguna mesin *fotocopy* ini yang tidak dapat memperbaiki sendiri dengan alasan kekurangan ilmu pengetahuan dan keterbatasan buku panduan tentang mesin *fotocopy* itu sendiri.
2. Belum adanya suatu sistem pakar untuk membantu pengguna dalam memperbaiki kerusakan pada mesin *fotocopy* tersebut.
3. Keterbatasan akan teknisi mesin *fotocopy* yang profesional dalam merk SHARP.

1.3 Pembatasan Masalah

Agar pembahasan terarah dan sesuai dengan yang diharapkan, mengingat banyaknya ruang lingkup permasalahan yang ada dan keterbatasan informasi yang dimiliki oleh peneliti, maka peneliti membuat batasan masalah. Adapun pembatasan masalah dan ruang lingkup dalam penelitian ini adalah sebagai berikut:

1. Mesin *fotocopy* yang digunakan dalam penelitian ini adalah merk SHARP MX-M264N.
2. Menggunakan teknik inferensi runut maju (*Forward Chaining*).
3. Sistem pakar ini menggunakan bahasa pemrograman *PHP* berbasis web dengan menggunakan *Adobe Dreamweaver*.
4. Sistem pakar ini menggunakan *database MySQL*.

1.4 Perumusan Masalah

Berdasarkan latar belakang di atas, maka penulis merumuskan permasalahan sebagai berikut:

1. Bagaimana merancang sebuah aplikasi sistem pakar untuk mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N menggunakan metode *Forward Chaining* dengan bahasa pemrograman PHP?
2. Bagaimana implementasi sistem pakar dalam mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N untuk dapat memberikan solusi yang berguna untuk penggunaanya?

1.5 Tujuan Penelitian

Berdasarkan perumusan masalah di atas, maka tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Untuk mengetahui rancangan aplikasi sistem pakar untuk mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N menggunakan metode *Forward Chaining* dengan bahasa pemrograman PHP.
2. Untuk mengetahui hasil implementasi sistem pakar dalam mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N.

1.6 Manfaat Penelitian

Setiap mahasiswa khususnya penulis yang melakukan penelitian pada suatu objek sangat mengharapkan agar hasil dari penelitian yang dilakukan ini dapat bermanfaat bagi semua kalangan yang membutuhkan. Manfaat-manfaat tersebut antara lain:

1.6.1 Aspek Teoritis

Bagi penulis, penelitian ini berguna untuk menambah wawasan dan memperdalam pengetahuan tentang merancang sebuah sistem pakar untuk mendeteksi kerusakan mesin *fotocopy*, yaitu:

1. Berdasarkan hasil penelitian ini, penulis secara tidak langsung dapat memperkaya pengetahuan dan menambah wawasan berpikir khalayak merancang sebuah sistem pakar untuk mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N.
2. Hasil penelitian ini dapat dijadikan sebagai sumber bacaan atau literatur bagi peneliti berikutnya yang mempunyai kemiripan dengan variabel penelitian yang akan diteliti.
3. Hasil penelitian yang telah dituangkan dalam bentuk tulisan ini akan dapat menambah perbendaharaan jumlah populasi karya ilmiah yang berkaitan tentang merancang sebuah sistem pakar untuk mendeteksi kerusakan mesin *fotocopy* SHARP MX-M264N, khususnya di Program Studi Teknik Informatika.

1.6.2 Aspek Praktis

Dari segi praktis, penelitian ini memberikan gambaran nilai aspek kepada penulis, objek penelitian dan pihak-pihak lain. Di antara aspek praktis tersebut adalah sebagai berikut:

1. Hasil dari kegiatan penelitian ini bertindak sebagai masukan bagi pengguna mesin *fotocopy* SHARP MX-M264N tentang merancang sebuah sistem pakar untuk mendeteksi kerusakan mesin *fotocopy*.
2. Mengetahui sampai sejauh mana kualitas ataupun kemampuan mahasiswa dalam mengimplementasikan teori yang ditelitinya ke dalam dunia nyata yang sebenarnya.

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

2.1.1 Kecerdasan Buatan (*Artificial Intelligence*)

Menurut Sutojo, dkk. (2011: 1) kecerdasan buatan berasal dari bahasa Inggris “*Artificial Intelligence*” atau disingkat AI, yaitu *artificial* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. Berikut adalah beberapa definisi kecerdasan buatan yang telah didefinisikan oleh beberapa ahli.

Alan Turing, ahli matematika berkebangsaan Inggris menetapkan definisi *Artificial Intelligent*: "Jika komputer tidak dapat dibedakan dengan manusia saat berbincang melalui terminal komputer, maka bisa dikatakan komputer itu cerdas, mempunyai kecerdasan ". John McCarthy dari Stanford mendefinisikan kecerdasan sebagai "kemampuan untuk mencapai sukses dalam menyelesaikan suatu permasalahan". Herbert Alexander Simon: "Kecerdasan buatan (*artificial intellegence*) merupakan kawasan penelitian, aplikasi, dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang dalam pandangan manusia adalah cerdas".

Rice and Knight: "Kecerdasan buatan (AI) merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia". *Encyclopedia Britannica*: "Kecerdasan buatan (AI) merupakan cabang dari ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan dan memproses informasi berdasarkan metode heuristik atau dengan berdasarkan sejumlah aturan".

Menurut Winston dan Prendergast, tujuan dari kecerdasan buatan adalah:

- a. Membuat mesin jadi lebih pintar (tujuan utama).
- b. Memahami apa itu kecerdasan (tujuan ilmiah).
- c. Membuat mesin lebih bermanfaat (tujuan *entrepreneurial*).

Program konvensional hanya dapat menyelesaikan persoalan yang di program secara spesifik. Jika ada informasi baru, sebuah program konvensional harus diubah untuk menyesuaikan diri dengan informasi baru tersebut. Hal ini tidak hanya menyebabkan boros waktu, namun juga dapat menyebabkan terjadinya *error*. Sebaliknya, kecerdasan buatan memungkinkan komputer untuk berpikir atau menalar dan menirukan proses belajar manusia sehingga informasi baru dapat diserap sebagai pengetahuan, pengalaman, dan proses pembelajaran serta dapat digunakan sebagai acuan di masa-masa yang akan datang.

Dari sini dapat dikatakan bahwa: "Cerdas adalah memiliki pengetahuan, pengalaman, dan penalaran untuk membuat keputusan dan mengambil tindakan. Jadi, agar mesin bisa cerdas (bertindak seperti manusia) maka harus diberi bekal pengetahuan dan diberi kemampuan untuk menalar".

Jika dibandingkan dengan kecerdasan alami (kecerdasan yang dimiliki oleh manusia), kecerdasan buatan memiliki keuntungan komersil, antara lain (Turban):

- a. Kecerdasan buatan lebih bersifat permanen. Kecerdasan alami akan cepat mengalami perubahan. Kemampuan kecerdasan buatan tidak akan pernah berubah selama programnya tidak diubah oleh *programmer*. Berbeda dengan kecerdasan alami. Karena sifat manusia yang subjektif, pelupa, dan makin lama makin tua hingga kemampuan berpikirnya berkurang seiring bertambahnya waktu, kemampuan kecerdasan alami cenderung tidak permanen.
- b. Kecerdasan buatan lebih mudah diduplikasi dan disebar.
- c. Kecerdasan buatan lebih murah dibandingkan dengan kecerdasan alami.
- d. Kecerdasan buatan bersifat konsisten.
- e. Kecerdasan buatan dapat didokumentasi. Solusi dan keputusan yang dibuat oleh kecerdasan buatan dapat didokumentasi dengan mudah karena disimpan di dalam *harddisk* dan pencarian datanya relatif lebih mudah dilacak. Sedangkan untuk kecerdasan alami, hal ini sangat sulit dilakukan.
- f. Kecerdasan buatan dapat mengerjakan pekerjaan lebih cepat dibanding dengan kecerdasan alami.
- g. Kecerdasan buatan dapat mengerjakan pekerjaan lebih baik dibanding dengan kecerdasan alami.

Sementara itu, kecerdasan alami memberikan keuntungan sebagai berikut:

- a. Kreatif. Pengetahuan seorang manusia selalu bertambah seiring dengan perkembangan waktu. Sifat bosan manusia pun mengakibatkan ia harus berpikir kreatif untuk mencari solusi-solusi terbaru. Berbeda dengan kecerdasan buatan, penambahan pengetahuan harus dilakukan pada sistem yang telah dibangun.
- b. Kecerdasan alami memungkinkan orang menggunakan pengalaman secara langsung. Sedangkan pada kecerdasan buatan harus bekerja dengan input-input simbolik.
- c. Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

Pada umumnya pemrograman konvensional hanya digunakan sebagai alat hitung, sedangkan kecerdasan buatan digunakan untuk menyelesaikan suatu permasalahan. Oleh karena itu, ada beberapa perbedaan yang mendasar antara kecerdasan buatan dan pemrograman konvensional. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain: sistem pakar, logika fuzzy, dan jaringan syaraf tiruan.

1. Sistem Pakar

Menurut Sutojo, dkk. (2011: 13) sistem pakar adalah sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah.

2. Logika Fuzzy

Menurut Sutojo, dkk. (2011: 211) logika fuzzy adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, *multi-channel* atau *workstation* berbasis akuisisi data, dan sistem kontrol.

3. Jaringan Syaraf Tiruan

Menurut Sutojo, dkk. (2011: 283) jaringan saraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak manusia. Elemen kunci dari paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang saling berhubungan (neuron), bekerja serentak untuk menyelesaikan masalah tertentu.

2.1.2 Sistem Pakar

Menurut Budiharto dan Suhartono (2014: 132) sistem pakar adalah program komputer yang menyelimuti penilaian dan perilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bidang tertentu. Biasanya, sistem seperti ini berisi basis pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan penambahan basis pengetahuan atau set aturan.

Pemrosesan yang dilakukan oleh sistem pakar merupakan pemrosesan pengetahuan, bukan pemrosesan data pada sistem komputer konvensional. Pengetahuan (*knowledge*) adalah pemahaman secara praktis maupun teoritis terhadap suatu obyek atau domain tertentu. Pengetahuan yang digunakan pada sistem pakar merupakan serangkaian informasi mengenai gejala-diagnosa, sebab-akibat, aksi-reaksi tentang suatu domain tertentu (misalnya, domain *diagnose* medis). Secara umum, definisi tradisional sebuah program komputer biasanya terdiri dari algoritma dan struktur data sedangkan dalam sistem pakar definisi berubah menjadi mesin inferensi dan pengetahuan.

Dengan sistem pakar, permasalahan yang seharusnya hanya dapat diselesaikan oleh para pakar/ahli, dapat diselesaikan oleh orang biasa/awam. Sedangkan untuk para ahli, sistem pakar membantu aktivitas mereka sebagai asisten yang seolah-olah sudah mempunyai banyak pengalaman.

Menurut Sutojo, dkk. (2011: 160) istilah sistem pakar berasal dari istilah *knowledge-based expert system*. Istilah ini muncul karena untuk memecahkan masalah, sistem pakar menggunakan pengetahuan seorang pakar yang dimasukkan ke dalam komputer. Seseorang yang bukan pakar menggunakan sistem pakar untuk meningkatkan kemampuan pemecahan masalah, sedangkan seorang pakar menggunakan sistem pakar untuk *knowledge assistant*. Berikut adalah beberapa pengertian sistem pakar.

1. Menurut Turban

“Sistem pakar adalah sebuah sistem yang menggunakan pengetahuan manusia di mana pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah yang biasanya membutuhkan kepakaran atau keahlian manusia”.

2. Menurut Jackson

“Sistem pakar adalah program komputer yang merepresentasikan dan melakukan penalaran dengan pengetahuan beberapa pakar untuk memecahkan masalah atau memberikan saran”.

3. Menurut Luger dan Stubblefield

“Sistem pakar adalah program yang berbasis pengetahuan yang menyediakan solusi ‘kualitas pakar’ kepada masalah-masalah dalam bidang (domain) yang spesifik”.

2.1.2.1 Manfaat Sistem Pakar

Menurut Sutojo, dkk. (2011: 160) sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang diberikan, di antaranya:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat daripada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan proses pengetahuan seorang pakar.
7. Andal. Sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit.

8. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti. Berbeda dengan sistem komputer konvensional, sistem pakar dapat bekerja dengan informasi yang tidak lengkap. Pengguna dapat merespons dengan: “tidak tahu” atau “tidak yakin” pada satu atau lebih pertanyaan selama konsultasi dan sistem pakar tetap akan memberikan jawabannya.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.

2.1.2.2 Kelemahan Sistem Pakar

Menurut Sutojo, dkk. (2011: 161) selain manfaat, ada juga beberapa kekurangan yang ada pada sistem pakar, diantaranya:

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar.

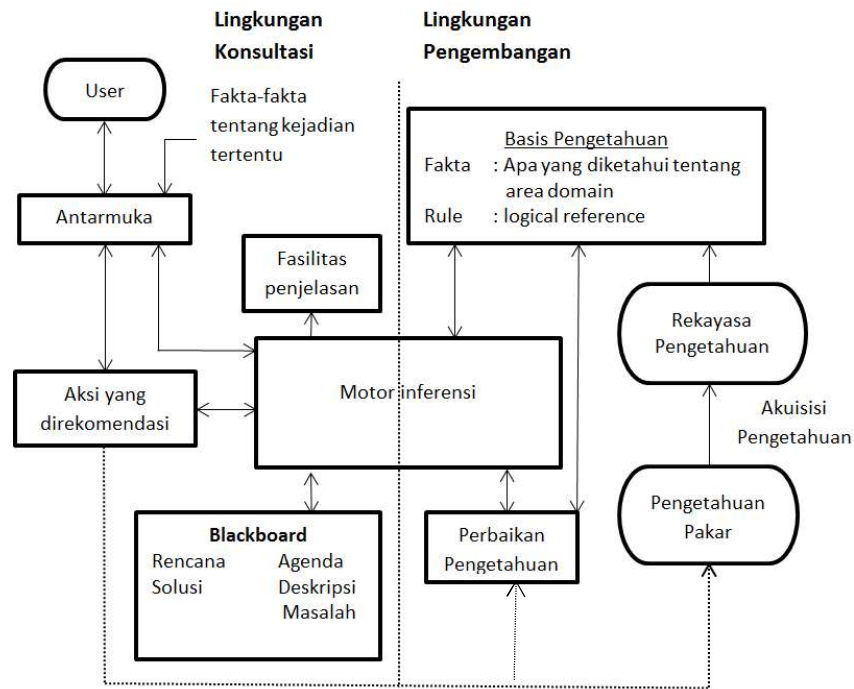
2.1.2.3 Ciri-ciri Sistem Pakar

Menurut Sutojo, dkk. (2011: 162) ciri-ciri dari sistem pakar adalah sebagai berikut:

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami.
4. Bekerja berdasarkan kaidah/*rule* tertentu.
5. Mudah dimodifikasi.
6. Basis pengetahuan dan mekanisme inferensi terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

2.1.2.4 Struktur Sistem Pakar

Menurut Sutojo, dkk. (2011: 166) ada dua bagian penting dari sistem pakar, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan ke dalam *knowledge base* (basis pengetahuan). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar. Gambar 2.1 menunjukkan komponen-komponen yang penting dalam sebuah sistem pakar.



Gambar 2.1 Komponen-komponen Sistem Pakar

Sumber: Sutojo, dkk. (2011:167)

Keterangan:

1. Akuisisi Pengetahuan

Subsistem ini digunakan untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan dengan format tertentu (dalam bentuk representasi pengetahuan). Sumber-sumber pengetahuan bisa diperoleh dari pakar, buku, dokumen multimedia, basis data, laporan riset khusus, dan informasi yang terdapat di *web*.

2. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan mengandung pengetahuan yang diperlukan untuk memahami, memformulasikan, dan menyelesaikan masalah. Basis pengetahuan terdiri dari dua elemen dasar, yaitu:

- a. Fakta, misalnya situasi, kondisi atau permasalahan yang ada.
- b. *Rule* (Aturan), untuk mengarahkan penggunaan pengetahuan dalam memecahkan masalah.

3. Mesin Inferensi (*Inference Engine*)

Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian, yaitu strategi yang berfungsi sebagai panduan arah dalam melakukan proses penalaran. Ada tiga teknik pengendalian yang digunakan, yaitu *forward chaining*, *backward chaining*, dan gabungan dari kedua teknik tersebut.

4. Daerah Kerja (*Blackboard*)

Untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang terjadi, sistem pakar membutuhkan *Blackboard*, yaitu area pada memori yang berfungsi sebagai basis data. Tiga tipe keputusan yang dapat direkam pada *blackboard*, yaitu:

- a. Rencana: bagaimana menghadapi masalah.
- b. Agenda: aksi-aksi potensial yang sedang menunggu untuk dieksekusi.
- c. Solusi: calon aksi yang akan dibangkitkan.

5. Antarmuka Pengguna (*User Interface*)

Digunakan sebagai media komunikasi antara pengguna dan sistem pakar. Komunikasi ini paling bagus bila disajikan dalam bahasa alami (*natural language*) dan dilengkapi dengan grafik, menu, dan formulir elektronik. Pada bagian ini akan terjadi dialog antara sistem pakar dan pengguna.

6. Subsistem Penjelasan (*Explanation Subsystem / Justifier*)

Berfungsi memberi penjelasan kepada pengguna, bagaimana suatu kesimpulan dapat diambil. Kemampuan seperti ini sangat penting bagi pengguna untuk mengetahui proses pemindahan keahlian pakar maupun dalam pemecahan masalah.

7. Sistem Perbaiki Pengetahuan (*Knowledge Refining System*)

Berfungsi memperbaiki pengetahuan (*knowledge refining system*) dari seorang pakar diperlukan untuk menganalisis pengetahuan, belajar dari kesalahan masa lalu, kemudian memperbaiki pengetahuannya sehingga dapat dipakai pada masa mendatang. Kemampuan evaluasi diri seperti itu diperlukan oleh program agar dapat menganalisis alasan-alasan kesuksesan dan kegagalannya dalam mengambil kesimpulan. Dengan cara ini basis pengetahuan yang lebih baik dan penalaran yang lebih efektif akan dihasilkan.

8. Pengguna (*User*)

Pada umumnya pengguna sistem pakar bukanlah seorang pakar (*non-expert*) yang membutuhkan solusi, saran, atau pelatihan (*training*) dari berbagai permasalahan yang ada.

2.1.2.5 Mesin Inferensi (*Inference Engine*)

Menurut Hartati dan Iswanti (2008: 5) mesin inferensi merupakan otak dari sistem pakar, berupa perangkat lunak yang melakukan tugas inferensi penalaran sistem pakar, biasa dikatakan sebagai mesin pemikir (*Thinking Machine*). Pada prinsipnya mesin inferensi inilah yang akan mencari solusi dari suatu permasalahan. Menurut Sutojo, dkk. (2011: 168) mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian, yaitu strategi yang berfungsi sebagai panduan arah dalam melakukan proses penalaran. Ada tiga teknik pengendalian yang digunakan, yaitu *forward chaining*, *backward chaining*, dan gabungan dari kedua teknik tersebut.

2.1.2.6 Metode *Forward Chaining*

Menurut Sutojo, dkk. (2011: 171) *forward chaining* adalah teknik pencarian yang dimulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian *IF* dari *rules IF-THEN*. Bila ada fakta yang cocok dengan bagian *IF*, maka *rule* tersebut dieksekusi. Bila sebuah *rule* dieksekusi, maka sebuah fakta baru (bagian *THEN*) ditambahkan ke dalam *database*. Setiap kali pencocokan, dimulai dari *rule* teratas. Setiap *rule* hanya boleh dieksekusi sekali saja. Proses pencocokan berhenti bila tidak ada lagi *rule* yang bisa dieksekusi.

2.2 Variabel Penelitian

Menurut Sugiyono (2014: 38) variabel penelitian adalah suatu atribut atau sifat atau nilai dari orang, obyek atau kegiatan yang mempunyai variasi tertentu yang ditetapkan oleh peneliti untuk dipelajari dan ditarik kesimpulannya. Variabel dalam penelitian ini adalah Mesin *Fotocopy*.

2.2.1 Mesin *Fotocopy*

Menurut Heriadi (2012: 39) mesin *fotocopy* merupakan perangkat elektromekanika yang berfungsi untuk menggandakan *master document* atau *file*. Dan mesin *fotocopy* adalah alat komputik canggih yang berfungsi untuk menyalin *image master* menjadi *image* baru di media kertas.

Jenis mesin *fotocopy* yang telah ditemukan, antara lain:

1. Mesin *fotocopy* kecil,
2. Mesin *fotocopy* standar,
3. Mesin *fotocopy* besar.

Kerusakan yang terdapat pada mesin *fotocopy* SHARP MX-M264N:

1. CCD (*Charger Couple Device*) *White Level Error*, level pembacaan pada referensi piring CCD putih tidak tepat, sehingga tidak dapat digunakan untuk penerangan lampu penggandaan (*copy*).
2. *Scanner Feed Trouble*, *scanner feed* tidak dapat diselesaikan dalam waktu tertentu.

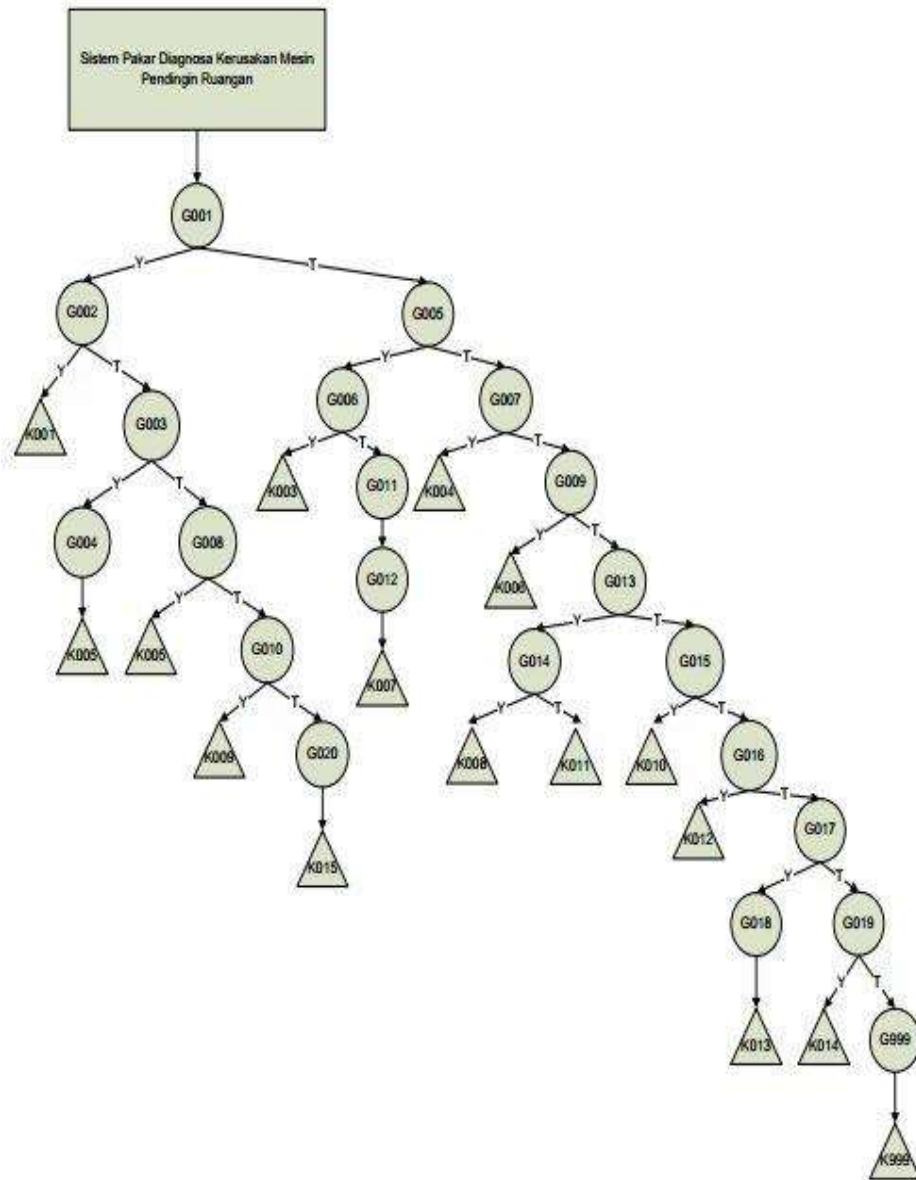
3. *Copy Lamp (Xenon Lamp) error*, lampu penggandaan (*copy*) tidak menyala.
4. *IMC PWB (Printed Wiring Board) Communication Trouble*, masalah pada komunikasi antara MCU dan IMC PWB.
5. *LSU (Laser Unit) Trouble*, sinyal BD dari LSU tidak terdeteksi dalam siklus konstan (dalam posisi OFF atau ON).
6. *Paper Exit Fan Lock Trouble*, *paper exit fan* tidak berputar. Dan sinyal dari *paper exit fan* yang terkunci akan terdeteksi tiap 1 detik atau lebih.
7. *Power Fan Lock Trouble*, *exhaust* atau *intake* pada motor *fan* tidak berputar. Dan sinyal dari *fan* yang terkunci akan terdeteksi tiap 1 detik atau lebih.

2.3 Software Pendukung

2.3.1 Pohon Keputusan

Menurut Sulianta dan Dominikus (2010: 56) pohon keputusan adalah pohon yang ada dalam analisa pemecahan masalah, pemetaan mengenai alternatif-alternatif pemecahan masalah yang dapat diambil dari masalah tersebut.

Berikut contoh pohon keputusan *forward chaining*:



Gambar 2.2 Pohon Keputusan *Forward Chaining*
 Sumber: Guntur dan Nita (2016:103)

2.3.2 Tabel Keputusan

Menurut Sulianta dan Dominikus (2010: 54) tabel keputusan (*decision table*) digunakan ketika *rule* atau aturan yang ada saling bebas. Baris-baris dalam *decision table* dapat mengandung pertanyaan dengan beberapa alternatif jawaban, hasil sementara, dan solusi.

Berikut ini contoh tabel keputusan *forward chaining*:

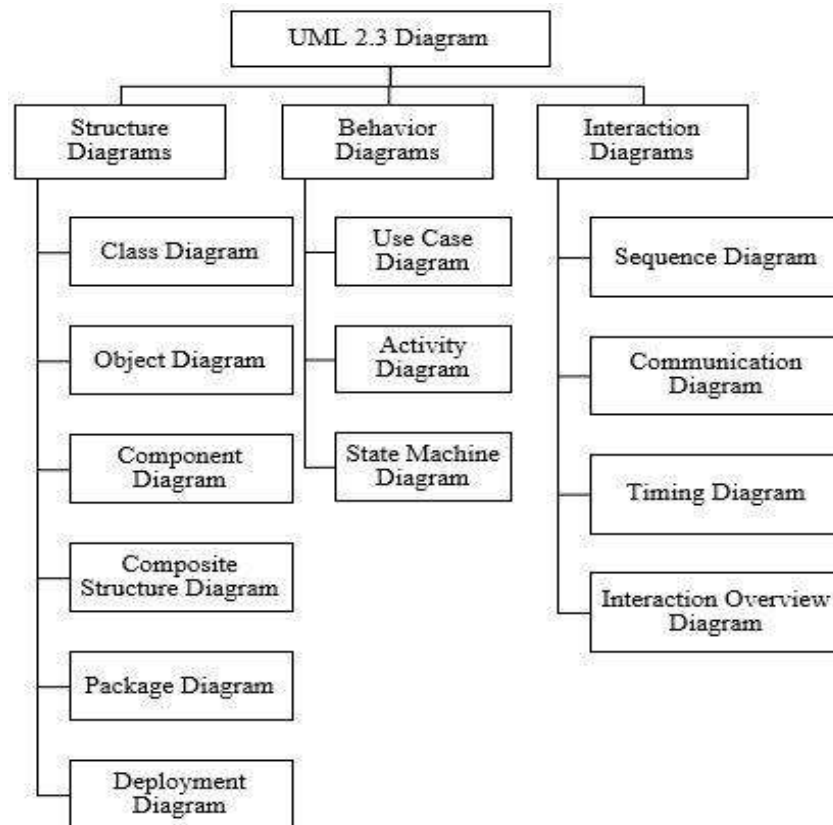
Tanda Gangguan	Tabel Keputusan											
	Kode Gangguan											
	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12
TG001	*	*	*									
TG002	*					*						
TG003	*											
TG004	*		*			*						
TG005	*			*								
TG006	*	*	*	*	*							
TG007		*	*	*	*	*						
TG008		*		*	*			*	*	*	*	
TG009		*										
TG010			*									
TG011				*	*		*	*	*	*	*	
TG012				*	*			*	*	*	*	
TG013				*								

Gambar 2.3 Tabel Keputusan *Forward Chaining*

Sumber: Sulianta dan Dominikus (2010:55)

2.3.3 *Unified Modeling Language (UML)*

Menurut Rosa dan Shalahuddin (2011: 117) sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language (UML)*. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak.



Gambar 2.4 Diagram UML

Sumber: Rosa dan Shalahuddin (2011: 121)

Berikut ini penjelasan singkat dari pembagian kategori tersebut.

- a. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- b. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- c. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.3.3.1 Class Diagram

Menurut Rosa dan Shalahuddin (2011: 122) diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2. Kelas yang menangani tampilan sistem

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3. Kelas yang diambil dari pendefinisian *use case*

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.

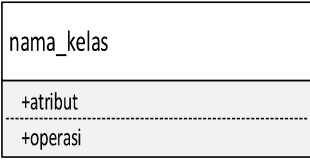
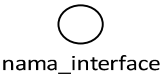


4. Kelas yang diambil dari pendefinisian data

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.


Jenis-jenis kelas di atas juga dapat digabungkan satu sama lain sesuai dengan pertimbangan yang dianggap baik asalkan fungsi-fungsi yang sebaiknya ada pada struktur kelas tetap ada. Susunan kelas dapat juga menambahkan kelas utilitas seperti koneksi ke basis data, membaca *file* teks, dan lain sebagainya sesuai kebutuhan.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

Tabel 2.1 Simbol-simbol pada diagram kelas

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka / <i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi / <i>association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Asosiasi berarah / <i>direct association</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

Tabel 2.2 Lanjutan

<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
---	---

Kebergantungan / <i>dependency</i> ----->	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> -----◇	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

Sumber: Rosa dan Shalahuddin (2011: 123)

2.3.3.2 Use Case Diagram

Menurut Rosa dan Shalahuddin (2011: 130) *use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

2.3.3.3 Activity Diagram

Menurut Rosa dan Shalahuddin (2011: 134) diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan di sini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.



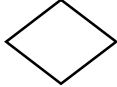


Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis di mana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* di mana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian di mana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

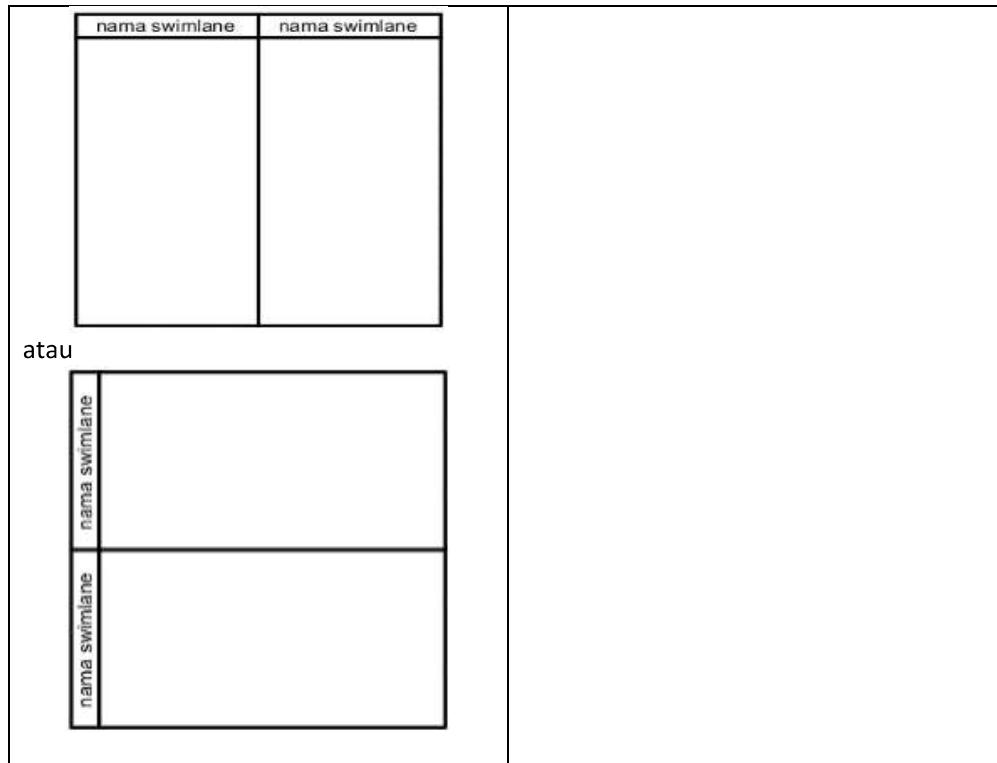
Tabel 2.2 Simbol-simbol pada diagram *activity*

Simbol	Deskripsi
--------	-----------

<p>Status awal</p> 	<p>Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal</p>
<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja</p>
<p>Percabangan / <i>decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu</p>
<p>Penggabungan / <i>join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu</p>
<p>Status akhir</p> 	<p>Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir</p>

Tabel 2.2 Lanjutan

<p><i>Swimlane</i></p>	<p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.</p>
------------------------	---



Sumber: Rosa dan Shalahuddin (2011: 134)


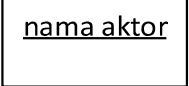

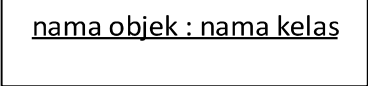

2.3.3.4 *Sequence Diagram*

Menurut Rosa dan Shalahuddin (2011: 137) diagram sekuen menggambarkan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Banyaknya diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin


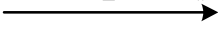
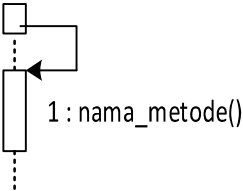
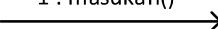
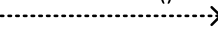
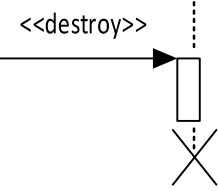
banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah symbol-simbol yang ada pada diagram sekuen:

Tabel 2.3 Simbol-simbol pada diagram *sequence*

Simbol	Deskripsi
<p>Aktor / actor</p>  <p>Nama aktor</p> <p>Atau</p>  <p>Tanpa waktu aktif</p>	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang dan biasanya dinyatakan dengan menggunakan kata benda di awal frase nama aktor.</p>
<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>

Tabel 2.3 Lanjutan

<p>Pesan tipe <i>create</i></p> <p style="text-align: center;"><<create>></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe <i>call</i></p> <p style="text-align: center;">1 : nama_metode()</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <p>Arah panah mengarah pada objek yang memiliki operasi, karena ini memanggil operasi / metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe <i>send</i></p> <p style="text-align: center;">1 : masukan()</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> <p style="text-align: center;">1 : keluaran()</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i>.</p>

Sumber: Rosa dan Shalahuddin (2011: 138)

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang berjalan terlebih dahulu.

2.3.4 WEB

Menurut Wahana Komputer (2009: 2) *web* merupakan media informasi berbasis jaringan komputer yang dapat diakses di mana saja dengan biaya relatif murah. *Web* merupakan bentuk implementasi dari bahasa pemrograman *web* (*web programming*). Sejarah perkembangan bahasa pemrograman web diawali dengan munculnya HTML (*Hypertext Markup Language*), yang kemudian dikembangkan dengan munculnya CSS (*Cascading Style Sheet*) yang bertujuan untuk memperindah tampilan *website*. Pada kenyataannya, bahasa tersebut masih memiliki kelemahan, yaitu skrip program dapat dilihat secara utuh, sehingga anda dapat mengetahui kerangka situs tersebut dan memberi peluang hacker untuk mengubahnya dengan mudah.

Bahasa pemrograman saat ini sudah sangat berkembang dengan berbagai kemudahan dalam penyajian dan *interface* yang lebih *friendly use*. Penyajian yang baik dari suatu bahasa pemrograman akan menghasilkan sebuah web yang dinamis sehingga pengunjung akan lebih mudah mendapatkan informasi yang dibutuhkan.

Beberapa bahasa pemrograman web berbasis *server* (*server side*) mulai muncul dan dikembangkan oleh beberapa perusahaan perangkat lunak, seperti: ASP (*Active Server Pages*) oleh *Microsoft*, JSP (*Java Server Pages*) oleh *Sun Microsystem*, dan CGI (*Common Gateway Interface*).

Bahasa pemrograman tersebut memiliki beberapa kendala, misalnya: *Active Server Pages* keluaran *Microsoft* hanya dapat dijalankan melalui sistem operasi buatan *Microsoft*, dan *Common Gateway Interface* yang hanya berjalan pada sistem operasi berbasis UNIX. Untuk mengatasi keterbatasan tersebut, muncullah bahasa pemrograman *web* yang dapat dijalankan pada sistem operasi *Microsoft*, sekaligus sistem operasi berbasis UNIX, yaitu: bahasa pemrograman PHP (*PHP Hypertext Preprocessor*).

2.3.5 PHP (*PHP Hypertext Preprocessor*)

PHP (Hypertext Preprocessor) merupakan bahasa pemrograman berbasis *web* yang memiliki kemampuan untuk memroses dan mengolah data secara dinamis. *PHP* dapat dikatakan sebagai sebuah *server-side embedded script language*, artinya semua sintaks dan perintah program yang anda tulis akan sepenuhnya dijalankan oleh *server*, tetapi dapat disertakan pada halaman HTML biasa. Pada umumnya, semua aplikasi yang dibangun menggunakan *PHP* akan memberikan hasil *web browser*, tetapi prosesnya secara seluruhan dijalankan pada server. (Wahana Komputer, 2009: 3)

Menurut Supono dan Putratama (2016:5) Bahasa pemrograman *PHP* merupakan bahasa pemrograman yang paling banyak digunakan, tentu karena berbagai alasan, salah satunya adalah mempunyai beberapa kelebihan dibandingkan bahasa pemrograman lainnya yang sejenis. Berikut ini kelebihan bahasa pemrograman *PHP*:

1. *PHP* adalah bahasa multiplatform yang artinya dapat berjalan di berbagai mesin dan sistem operasi (*Linux, Unix, Macintosh, Windows*) dan dapat

dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem lainnya.

2. *PHP* bersifat *Open Source* yang berarti dapat digunakan oleh siapa saja secara gratis.
3. *Web Server* yang mendukung *PHP* dapat ditemukan di mana-mana dari mulai *apache*, *IIS*, *Lighttpd*, *nginx*, hingga *Xitami* dengan konfigurasi yang relatif mudah dan tidak berbelit-belit, bahkan banyak yang membuat dalam bentuk paket atau *package* (*PHP*, *MySQL*, dan *Web Server*).
4. Dalam sisi pemahaman, *PHP* adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. Banyak bertebaran aplikasi dan program *PHP* yang gratis dan siap pakai seperti *WordPress*, *PrestaShop*, dan lain-lain.
6. Dapat mendukung banyak *database*, seperti *MySQL*, *Oracle*, *MS-SQL* dan seterusnya.

Dari sekian banyaknya kelebihan yang dimiliki oleh bahasa pemrograman *PHP*, tentu tidak berarti tidak ada kekurangannya. Berikut ini kekurangan bahasa pemrograman web *PHP* yang mungkin menjadi pertimbangan dalam memilih bahasa pemrograman ini.

1. *PHP* tidak mengenal *package*.
2. Jika tidak di-*encoding*, maka kode *PHP* dapat dibaca semua orang dan untuk meng-*encoding*-nya dibutuhkan *tool* dari *Zend* yang mahal sekali biayanya.
3. *PHP* memiliki kelemahan keamanan. Jadi *programmer* harus jeli dan berhati-hati dalam melakukan pemrograman dan konfigurasi *PHP*.

2.3.6 MySQL

Menurut Nugroho (2008: 91) *MySQL (My Structured Query Language)* atau yang biasa dibaca mai-sekuel adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah *Open Source*. *MySQL* sebenarnya produk yang berjalan pada platform Linux, dengan adanya perkembangan dan banyaknya pengguna, serta lisensi dari *database* ini adalah *Open Source*, maka para pengembang kemudian merilis versi *windows*.

Selain itu, *MySQL* juga merupakan program pengakses *database* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna). Kelebihan lain dari *MySQL* adalah menggunakan bahasa query (permintaan) standar SQL (*Structures Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur, SQL telah distandarkan untuk semua program pengakses *database* seperti *Oracle*, *PosgreSQL*, *SQL server* dan lain-lain.

Sebuah program penghasil *database*, *MySQL* tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna (*interface*) yang berguna sebagai program aplikasi pengakses *database* yang dihasilkan. *MySQL* dapat didukung oleh hampir semua program aplikasi baik yang *Open Source* seperti PHP maupun yang tidak *Open Source* yang ada pada platform *windows* seperti *visual basic*, *Delphi* dan lainnya.

Program-program aplikasi yang mendukung *MySQL*:

1. *PHP (Page Hipertext Preprosesor)*
2. *Borland Delphi, Borland C++ Builder*
3. *Visual Basic 5.0/6.0 dan .Net*

4. Visual FoxPro

2.3.7 Dreamweaver

Menurut Supono dan Putratama (2016: 14) *dreamweaver* merupakan perangkat lunak yang digunakan untuk membuat atau meng-*edit* halaman *web*. *Dreamweaver* sangat cocok untuk pengguna pemula atau pengguna yang sedang belajar membuat halaman *web*. *Dreamweaver* menyediakan fitur-fitur untuk mempermudah pembuatan halaman *web*, sehingga tampilan *web* yang dibuat nanti akan sama dengan tampilan pada saat proses perancangan halaman *web*. *Dreamweaver* mendukung beberapa bahasa pemrograman, antara lain: *HTML, PHP, CSS, javascript, php*, dan sebagainya.

Sebagai editor, *Dreamweaver* mempunyai sifat yang WYSIWYG yang artinya apa yang kita lihat pada halaman desain, maka semuanya itu akan kita peroleh pada browser. Dengan kelebihan ini sehingga seorang programmer (pembuat program) atau desainer (pembuat desain web) dapat langsung melihat hasil buatanya tanpa harus membukanya pada browser (aplikasi pengakses web seperti *Internet Explorer, Mozilla*, dll). (Nugroho, 2008: 2)

2.3.8 Notepad ++

Menurut Supono dan Putratama (2016: 13) *notepad ++* merupakan aplikasi editor gratis serta *powerful* yang dapat digunakan oleh seorang pengembang aplikasi (*programmer*) untuk menuliskan sebuah kode-kode program. *Notepad ++* mendukung banyak bahasa pemrograman, diantaranya: *Assembly, C, C++, C#, CSS, HTML, Java*,

Javascript, Pascal, Perl, PHP, Python, Ruby, Shell, SQL, VB, XML, dan lain-lain. *Notepad ++* ini memiliki banyak kelebihan bila dibandingkan dengan *notepad* bawaan *windows* yang pertama, seperti memiliki GUI yang baik dan menarik. Selain itu, *Notepad ++* juga dapat ditambahkan berbagai *plugin* yang bisa semakin mempermudah pekerjaan *programmer*. Dan juga *notepad ++* terdapat versi portabelnya. Dan kelebihan lainnya adalah *notepad ++* ini dapat diunduh dan digunakan secara gratis.

2.4 Penelitian Terdahulu

Sebelum kegiatan pelaksanaan kegiatan penelitian dilaksanakan, penulis sudah terlebih dahulu membaca dan mempelajari beberapa hasil penelitian terdahulu yang berkaitan dengan penelitian yang sedang dilakukan untuk memudahkan proses penyelesaian penulisan ini. Penelitian terdahulu yang dipelajari oleh penulis di dalam menyempurnakan penulisan ini dapat di lihat sebagai berikut:

Guntur dan Nita Merlina (2016) Sistem Pakar Diagnosa pada Mesin Pendingin Ruangan dengan Metode *Forward Chaining* dengan ISSN 1978-1946, diperoleh fakta bahwa Mesin pendingin ruangan sudah menjadi kebutuhan dasar bagi banyak masyarakat. Dalam penggunaan mesin pendingin ruangan kemungkinan besar membutuhkan perawatan secara berkala, hal ini yang mendorong pembangunan sistem pakar untuk mendiagnosa kerusakan mesin pendingin ruangan. Sistem pakar merupakan salah satu cabang kecerdasan buatan yang mempelajari bagaimana mengadopsi cara seorang pakar berpikir dan bernalar dalam menyelesaikan suatu permasalahan, dan membuat suatu keputusan maupun mengambil kesimpulan dari sejumlah fakta yang ada. Metode inferensi yang digunakan adalah *forward chaining*, yaitu proses inferensi yang

memulai pencarian dari premis atau data masukan berupa gejala menuju pada konklusi atau kesimpulan kerusakan serta memberikan solusi mengenai kerusakan tersebut. Hasil pengujian yang dibuat berdasarkan kuesioner menunjukkan bahwa, program mudah digunakan, bermanfaat dalam memberikan informasi tentang kerusakan mesin pendingin ruangan dan membuat konsultasi lebih efisien dengan hasil yang sesuai dengan diagnosa teknisi AC.

Prasetyo Adi Wibowo dan Sugeng Purbawanto (2015) Sistem Pakar Untuk Mendiagnosa Kerusakan Televisi Dengan Metode *Forward Chaining* menggunakan *PHP* dan *MySQL* dengan ISSN 2252-6811, diperoleh fakta bahwa memiliki televisi sangatlah mudah, namun perawatannya yang sulit. Akan sangat membantu jika memiliki pengetahuan tentang elektronika. Oleh karena itu dikembangkan sistem pakar untuk mendiagnosa kerusakan televisi. Sistem pakar mencoba mencari solusi sebagaimana yang dilakukan seorang pakar. Sistem pakar juga dapat memberikan penjelasan terhadap langkah yang diambil dan memberikan saran atau kesimpulan yang ditemukannya. Mendeteksi kerusakan sejak dini dapat mencegah kerusakan lebih parah dan menangani kerusakan tersebut dengan benar. Sistem pakar dapat membantu untuk mendiagnosa kerusakan dan memberikan solusi atas kerusakan tersebut. Pembangunan sistem ini bertujuan untuk membantu pengguna mengatasi kerusakan televisi. Sistem ini dibangun menggunakan metode *forward chaining*. *Forward chaining* digunakan untuk menguji faktor-faktor yang dimasukkan dengan aturan yang disimpan dalam sistem hingga dapat diambil kesimpulan. Sistem pakar ini memiliki fungsi khusus bagi pakar untuk menambah, mengubah dan menghapus pengetahuan atau aturan tanpa harus membuka *database*. Sistem pakar ini dibuat dengan menggunakan *PHP* dan *MySQL* sebagai basis datanya.

Rosmawati Tamin (2015) Sistem Pakar untuk Diagnosa Kerusakan Pada Printer menggunakan Metode *Forward Chaining* dengan ISSN 2442-4512, diperoleh fakta bahwa mensubstitusikan pengetahuan manusia ke dalam bentuk sistem dengan menampung kemampuan atau keahlian seorang pakar untuk melakukan proses analisa suatu masalah sehingga sistem dapat sistem bekerja menyelesaikan masalah sebagaimana manusia mengerjakannya dan menyelesaikan masalah tersebut. Kerusakan printer terkadang menjadi masalah besar ketika seorang yang awam tidak mengetahui letak kesalahan printer maka dibutuhkan sistem yang mampu bekerja otomatis untuk memberikan solusi kerusakan printer. Penelitian ini bertujuan untuk merancang sebuah sistem yang dapat digunakan untuk menangani kerusakan printer. Pengguna aplikasi ini seolah-olah berhadapan langsung dengan pakar di bidang *hardware* khususnya printer. Perencanaan sistem dilakukan dengan membuat *knowledgebase* menggunakan *decision tree* dan aturan *if-then* sebagai representasi pengetahuan. Sistem dibuat dengan menggunakan metode *forward chaining* dan bahasa pemrograman *Visual Basic*. Hasil penelitian ini mengungkapkan jenis-jenis kerusakan yang terjadi pada sebuah printer serta penanganan dari kerusakan tersebut. Pengujian aplikasi juga dilakukan untuk mengetahui akurasi dan variasi serta *user friendly* dan fleksibilitas sistem. Hasil dari keseluruhan pengujian ini dapat disimpulkan bahwa program sudah cukup baik walaupun jenis kerusakan yang dihasilkan belum lengkap karena pada sistem ini hanya mendeteksi 15 jenis kerusakan mesin secara umum.

Cholil Jamhari, Agus Kiryanto dan Sri Huning Anwariningsih (2014) Sistem Pakar Diagnosis Kerusakan Sepeda Motor Non *Matic* dengan ISSN 2337-4349, diperoleh fakta bahwa penelitian ini bertujuan untuk merancang sistem pakar di bidang otomotif khususnya sepeda motor non *matic*. Perencanaan sistem dalam membuat

knowledgebase memakai pohon keputusan dan kaidah produksi sebagai representasi pengetahuan. Pembuatan metode inferensi memakai metode *forward chaining* yang telah dimodifikasi sehingga sesuai dengan permasalahan. Implementasi program sistem pakar ini menggunakan *visual studio 2013*. Aplikasi ini akan menghasilkan jenis-jenis kerusakan yang terjadi pada sepeda motor non *matic* serta penanganan dari kerusakan tersebut. Pengujian aplikasi terdiri atas dua jenis pengujian, yaitu: pengujian akurasi dan variasi serta pengujian *user friendly* dan fleksibilitas. Akurasi dan variasi diuji dengan melakukan analisis terhadap hasil dari aplikasi. Pengujian *user friendly* dan fleksibilitas menggunakan metode wawancara terhadap teknisi. Hasil dari keseluruhan pengujian ini dapat disimpulkan bahwa program sudah cukup baik dalam mendeteksi kerusakan mesin secara umum yang terjadi pada sepeda motor non *matic* serta disertakan cara perbaikan. Dari penjelasan tersebut dapat diperoleh kesimpulan bahwa sistem ini mempermudah pengguna dalam menggunakan aplikasi ini.

Supyani, Bebas Widada, dan Wawan Laksito (2013) Aplikasi Diagnosa Kerusakan Mesin Sepeda Motor Bebek 4 Tak dengan Metode *Forward Chaining* dengan ISSN 2338-4018, diperoleh fakta bahwa perencanaan sistem dalam membuat *knowledgebase* memakai aturan *if-then* sebagai representasi pengetahuan. Pembuatan metode inferensi memakai metode *forward chaining* yang telah dimodifikasi sehingga sesuai dengan permasalahan. Implementasi program sistem pakar ini menggunakan bahasa pemrograman PHP. Aplikasi ini akan menghasilkan jenis-jenis kerusakan yang terjadi pada sepeda motor serta solusi penanganan dari kerusakan tersebut. Adapun metode pengumpulan data yang dilakukan oleh penulis adalah metode wawancara, metode observasi dan metode studi pustaka. Tujuan dari penelitian ini adalah membantu menganalisa kerusakan mesin sepeda motor bagi mekanik pemula dan para siswa yang

sedang prakerin, hasilnya tercipta sebuah aplikasi yang dapat mendiagnosa kerusakan mesin sepeda motor untuk membantu siswa yang sedang melaksanakan prakerin dalam mendiagnosa kerusakan mesin.

2.5 Kerangka Pemikiran

Menurut Sugiyono (2014: 60) Uma Sekaran dalam bukunya *Business Research* (1992) mengemukakan bahwa, kerangka berfikir merupakan model konseptual tentang bagaimana teori berhubungan dengan berbagai faktor yang telah diidentifikasi sebagai masalah yang penting.

Identifikasi masalah dalam penelitian ini adalah banyaknya pengguna mesin *fotocopy* ini yang tidak dapat memperbaiki sendiri dengan alasan kekurangan ilmu pengetahuan dan keterbatasan buku panduan tentang mesin *fotocopy* itu sendiri, belum adanya suatu sistem yang mampu membantu pengguna untuk memperbaiki kerusakan pada mesin *fotocopy* tersebut, sulitnya berkonsultasi dengan teknisi mengenai kerusakan yang terdapat pada mesin *fotocopy*, keterbatasan akan teknisi mesin *fotocopy* yang profesional dalam merk SHARP. Variabel yang digunakan dalam penelitian ini adalah mesin *fotocopy* SHARP MX-M264N.

Berdasarkan judul penelitian di atas, penulis membuat kerangka pemikiran dalam penelitian ini sebagai berikut:



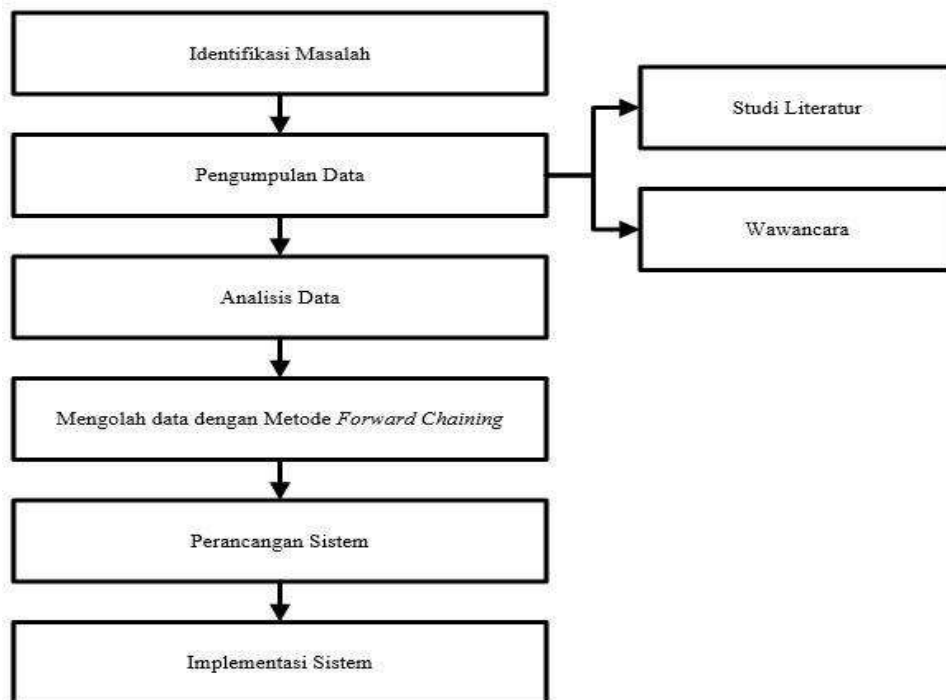
Gambar 2.5 Kerangka Pemikiran

Sumber: Data Penelitian (2017)

2 BAB III METODE PENELITIAN

3 3.1 Desain Penelitian

Menurut Martono (2010: 131) bahwa desain penelitian adalah penjelasan mengenai berbagai komponen yang akan digunakan peneliti dan kegiatan yang akan dilakukan selama proses penelitian. Desain penelitian ini bertujuan untuk melaksanakan penelitian sehingga diperoleh suatu logika dalam pengujian untuk memperoleh hasil penelitian dan dalam membuat kesimpulan sesuai dengan fokus penelitian.



Gambar 3.1 Desain Penelitian

Sumber: Data Penelitian (2017)

Berikut ini adalah penjelasan tentang Desain Penelitian diatas:

1. Identifikasi Masalah

Tahap identifikasi masalah ini untuk menentukan masalah-masalah yang terdapat pada penelitian tersebut yang berjudul "SISTEM PAKAR MENDIAGNOSA KERUSAKAN MESIN FOTOCOPY SHARP MX-M264N BERBASIS *WEB* DENGAN MENGGUNAKAN METODE *FORWARD CHAINING*". Untuk mengetahui masalah-masalah yang ada pada penelitian tersebut.

2. Pengumpulan Data

Tahap selanjutnya adalah pengumpulan data. Pengumpulan data dapat diambil dari buku dan jurnal yang berhubungan dengan kerusakan pada mesin fotocopy SHARP MX-M264N. Penelitian ini mendapatkan data-data kerusakan pada mesin fotocopy SHARP MX-M264N dengan melakukan wawancara langsung pada pakar. Pengumpulan data mengenai gejala-gejala kerusakan pada mesin fotocopy SHARP MX-M264N dan cara memperbaikinya.

3. Studi Literatur

Studi literatur dilakukan dengan cara mempelajari aspek-aspek yang berkaitan dengan penelitian ini, diantaranya adalah mencari jenis-jenis kerusakan yang dibahas penulis tentang kerusakan mesin fotocopy SHARP MX-M264N, gejala-gejala yang menyebabkan kerusakan pada mesin fotocopy SHARP MX-M264N, teori metode *forward chaining*. Data-data yang digunakan dalam studi literatur didapat dengan cara mengumpulkan jurnal, penelusuran internet, dan buku yang berkaitan dengan topik.

4. Wawancara

Wawancara dilakukan dengan mengadakan tanya jawab langsung dengan pakar yang berhubungan dengan kerusakan mesin fotocopy SHARP MX-M264N, sehingga nanti

pada penelitian ini data yang didapat akan lebih akurat yang tidak terpaku pada studi literatur.

5. Analisis Data

Setelah pengumpulan data, tahap selanjutnya adalah Analisis Data. Data dan informasi yang dikumpulkan akan digunakan untuk mendukung penelitian tersebut. Data didapatkan melalui wawancara langsung dengan pakar tentang kerusakan - kerusakan yang terdapat pada mesin fotocopy SHARP MX-M264N ataupun buku yang berhubungan dengan itu.

6. Mengolah Data dengan Metode *Forward Chaining*

Pada tahap ini, penulis mendapatkan data-data hasil dari wawancara dengan pakar tentang gejala kerusakan mesin fotocopy SHARP MX-M264N dan cara memperbaikinya dengan menggunakan metode *forward chaining* untuk mempermudah penelitian ini.

7. Perancangan Sistem

Tahap selanjutnya adalah perancangan sistem. Perancangan sistem ini untuk mendiagnosa kerusakan pada mesin fotocopy SHARP MX-M264N berbasis *web*. Perancangan dari model sistem, perancangan sistem input dan merancang *rule-rule* yang akan digunakan dalam mendiagnosa kerusakan mesin fotocopy SHARP MX-M264N berbasis *web* berdasarkan data yang ada.

8. Implementasi Sistem

Setelah perancangan sistem, diharapkan sistem tersebut mampu memecahkan masalah tentang kerusakan pada mesin fotocopy SHARP MX-M264N dan memudahkan *user* mencari informasi dan data tentang kerusakan tersebut.

4 3.2 Teknik Pengumpulan Data

Menurut Sugiyono (2014: 224) teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Tanpa mengetahui teknik pengumpulan data, maka peneliti tidak akan mendapatkan data yang memenuhi standar data yang ditetapkan.

Data yang dikumpulkan harus cukup valid untuk digunakan. Validitas data dapat ditingkatkan jika alat pengukur serta kualitas dari pengambilan datanya sendiri cukup valid.

Dalam teknik pengumpulan data ini, penulis menggunakan pendekatan teknis, yaitu:

1. Wawancara

Wawancara merupakan metode pencarian dan pengumpulan informasi data dengan cara melakukan tanya jawab kepada seorang pakar secara langsung.

2. Studi Literatur

Penelusuran literatur adalah cara pengumpulan data dengan menggunakan sebagian atau seluruh data yang telah ada atau laporan data dari penelitian sebelumnya. Penelusuran literatur disebut juga pengamatan tidak langsung.

5 3.3 Operasional Variabel

Operasional variabel adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi tentang hal tersebut, kemudian ditarik kesimpulannya. (Sugiyono, 2010: 58).

Operasional variabel merupakan penarikan batasan yang menjelaskan ciri-ciri spesifik yang lebih substantif dari suatu konsep. Tujuannya agar peneliti dapat mencapai suatu alat ukur yang sesuai dengan variabel yang sudah di definisikan sebelumnya, maka peneliti harus memasukkan proses atau alat ukur yang akan digunakan untuk kuantifikasi gejala atau variabel yang ditelitinya. Penelitian ini menggunakan satu variabel saja, yaitu Mesin Fotocopy SHARP MX-M264N.

Indikator: Kerusakan CCD (*Charger Couple Device*) *White Level Error*

Penyebab:

- Kegagalan instalasi *flat cable* pada unit CCD
- Adanya debu pada cermin, lensa dan permukaan *white plate*
- Gangguan pada pencahayaan lampu penyalin
- Unit CCD tidak berfungsi normal
- Unit MCU PWB tidak berfungsi normal

Solusi:

- Bersihkan bagian cermin, lensa dan permukaan *white plate*
- Periksa kuantitas cahaya pada lampu penyalin dan operasinya
- Periksa bagian unit CCD
- Periksa unit MCU PWB

Indikator: Kerusakan *Scanner Feed Trouble*

Penyebab:

- Unit cermin mengalami kecacatan
- Kabel *scanner* tidak tersambung
- Terjadi kesalahan pada *origin detection sensor*
- Pemanfaatan cermin pada *motor* tidak berfungsi normal

Solusi:

- Periksa operasi *scanning*
- Ketika *mirror not feeds*: periksa kabel *scanner* yang tidak tersambung, periksa pemanfaatan dan konektor antara cermin *motor* dan unit MCU PWB, ganti unit *mirror*, ganti unit MCU PWB
- Ketika *mirror feeds*: periksa *mirror home position sensor*

Indikator: Kerusakan *Copy Lamp (Xenon Lamp) Error*

Penyebab:

- Unit lampu penyalin tidak berfungsi normal
- Pemanfaatan unit lampu penyalin tidak optimal
- Unit CCD PWB tidak berfungsi normal

Solusi:

- Periksa unit lampu penyalin
- Ketika unit lampu penyalin nyala: periksa penyuplaian dan konektor antara unit CCD dan unit MCU PWB
- Saat unit lampu penyalin tidak menyala: periksa penyuplaian dan konektor antara lampu penyalin dan unit MCU PWB
- Ganti unit lampu penyalin
- Ganti unit MCU PWB

Indikator: Kerusakan IMC PWB *Communication Trouble*

Penyebab:

- Konektor pada unit IMC PWB tidak tersambung
- Kerusakan pada pin konektor *motherboard*
- Cacatnya ROM pada unit IMC PWB, terjadi kegagalan pada data

Solusi:

- Periksa konektor pada unit IMC PWB dan unit MCU PWB
- Periksa bagian *grounding* pada mesin fotocopy
- Periksa ROM pada unit IMC PWB

Indikator: Kerusakan LSU *Trouble*

Penyebab:

- Konektor unit LSU atau bagian dalam unit LSU tidak tersambung
- Putaran pada *polygon motor* tidak berfungsi normal
- *Laser* tidak menerang
- Kegagalan pada unit MCU PWB

Solusi:

- Periksa pada bagian konektor unit LSU yang tidak tersambung
- Periksa bagian operasi unit LSU
- Periksa bahwa putaran pada *polygon motor* berputar normal
- Periksa *laser* pada pencahayaan LED
- Ganti unit LSU dan unit MCU PWB

Indikator: Kerusakan *Paper Exit Fan Lock Trouble*

Penyebab:

- Koneksi pada bagian unit *paper exit fan* bermasalah
- Kegagalan pada unit MCU PWB

Solusi:

- Periksa koneksi pada bagian unit *paper exit fan*
- Ganti unit *paper exit fan*
- Ganti unit MCU PWB

Indikator: Kerusakan *Power Fan Lock Trouble*

Penyebab:

- Koneksi pada kipas bermasalah, penyuplaian tidak tersambung
- Kegagalan pada unit MCU PWB

Solusi:

- Periksa koneksi pada unit *exhaust/intake fan*
- Ganti unit *exhaust/intake fan*
- Ganti unit MCU PWB

Sumber: Data Penelitian (2017)

3.4 Perancangan Sistem

6 3.4.1 Knowledge Base (Basis Pengetahuan)

Dalam penelitian ini, penulis menggunakan sebuah *software* untuk merancang sistem untuk mendiagnosa kerusakan pada mesin fotocopy SHARP MX-M264N berbasis *web*. *Software* tersebut berupa aplikasi *editor dreamweaver*, PHP (*hypertext processor*), dan *MySQL*. Aplikasi *editor dreamweaver* digunakan untuk mendesain tampilan layar pada sebuah *website*. Aplikasi PHP merupakan aplikasi untuk menyambungkan *website* ke *database*, dan *MySQL* adalah aplikasi untuk membuat *database*. Dalam penelitian ini, penulis merancang sistem tersebut dari:

Tabel 3.1 Kode Nama Kerusakan

Kode	Nama Kerusakan
R01	CCD <i>White Level Error</i>
R02	<i>Scanner Feed Trouble</i>
R03	<i>Copy Lamp (Xenon Lamp) Error</i>
R04	IMC PWB <i>Communication Trouble</i>
R05	LSU <i>Trouble</i>
R06	<i>Paper Exit Fan Lock Trouble</i>
R07	<i>Power Fan Lock Trouble</i>

Sumber: Data Penelitian (2017)

Tabel 3.2 Kode Penyebab

Nama Kerusakan	Kode	Penyebab
<i>CCD White Level Error</i>	P01	Kegagalan instalasi <i>flat cable</i> pada unit CCD
	P02	Adanya debu pada cermin, lensa dan permukaan <i>white plate</i>
	P03	Gangguan pada pencahayaan lampu penyalin
	P04	Unit CCD tidak berfungsi normal
	P05	Unit MCU PWB tidak berfungsi normal
<i>Scanner Feed Trouble</i>	P06	Unit cermin mengalami kecacatan
	P07	Kabel <i>scanner</i> tidak tersambung
	P08	Terjadi kesalahan pada <i>origin detection sensor</i>
	P09	Pemanfaatan cermin pada <i>motor</i> tidak berfungsi normal
<i>Copy Lamp (Xenon Lamp) Error</i>	P10	Unit lampu penyalin tidak berfungsi normal
	P11	Pemanfaatan unit lampu penyalin tidak optimal
	P12	Unit CCD PWB tidak berfungsi normal
<i>IMC PWB Communication Trouble</i>	P13	Konektor pada unit IMC PWB tidak tersambung
	P14	Kerusakan pada pin konektor motherboard
	P15	Cacatnya ROM pada unit IMC PWB, terjadi kegagalan pada data

Tabel 3.2 Lanjutan

<i>LSU Trouble</i>	P16	Konektor unit LSU atau bagian dalam unit LSU tidak tersambung
	P17	Putaran pada <i>polygon motor</i> tidak berfungsi normal
	P18	<i>Laser</i> tidak menerang
	P19	Kegagalan pada unit MCU PWB
<i>Paper Exit Fan Lock Trouble</i>	P19	Kegagalan pada unit MCU PWB
	P20	Koneksi pada bagian unit <i>paper exit fan</i> bermasalah
<i>Power Fan Lock Trouble</i>	P19	Kegagalan pada unit MCU PWB
	P21	Koneksi pada kipas bermasalah, penyuplaian tidak tersambung

Sumber: Data Penelitian (2017)

Tabel 3.3 Kode Solusi

Nama Kerusakan	Solusi
<p>CCD <i>White Level Error</i></p>	<ul style="list-style-type: none"> • Bersihkan bagian cermin, lensa dan permukaan <i>white plate</i> • Periksa kuantitas cahaya pada lampu penyalin dan operasinya • Periksa bagian unit CCD • Periksa unit MCU PWB
<p><i>Scanner Feed Trouble</i></p>	<ul style="list-style-type: none"> • Periksa operasi <i>scanning</i> • Ketika <i>mirror not feeds</i>: periksa kabel <i>scanner</i> yang tidak tersambung, periksa pemanfaatan dan konektor antara cermin <i>motor</i> dan unit MCU PWB, ganti unit <i>mirror</i>, ganti unit MCU PWB • Ketika <i>mirror feeds</i>: periksa <i>mirror home position sensor</i>
<p><i>Copy Lamp (Xenon Lamp) Error</i></p>	<ul style="list-style-type: none"> • Periksa unit lampu penyalin • Ketika unit lampu penyalin nyala: periksa penyuplaian dan konektor antara unit CCD dan unit MCU PWB • Saat unit lampu penyalin tidak menyala: periksa penyuplaian dan konektor antara lampu penyalin dan unit MCU PWB • Ganti unit lampu penyalin • Ganti unit MCU PWB

Tabel 3.3 Lanjutan

<p>IMC PWB <i>Communication Trouble</i></p>	<ul style="list-style-type: none"> • Periksa konektor pada unit IMC PWB dan unit MCU PWB • Periksa bagian <i>grounding</i> pada mesin fotocopy • Periksa ROM pada unit IMC PWB
<p>LSU <i>Trouble</i></p>	<ul style="list-style-type: none"> • Periksa pada bagian konektor unit LSU yang tidak tersambung • Periksa bagian operasi unit LSU • Periksa bahwa putaran pada <i>polygon motor</i> berputar normal • Periksa <i>laser</i> pada pencahayaan LED • Ganti unit LSU dan unit MCU PWB

<i>Paper Exit Fan Lock Trouble</i>	<ul style="list-style-type: none"> • Ganti unit MCU PWB • Periksa koneksi pada bagian unit <i>paper exit fan</i> • Ganti unit <i>paper exit fan</i>
<i>Power Fan Lock Trouble</i>	<ul style="list-style-type: none"> • Ganti unit MCU PWB • Periksa koneksi pada unit <i>exhaust/intake fan</i> • Ganti unit <i>exhaust/intake fan</i>

Sumber: Data Penelitian (2017)

Tabel 3.4 Tabel Keputusan Kerusakan Mesin *Fotocopy*

Kode Nama Kerusakan	Kode Penyebab																				
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21
R01	√	√	√	√	√																
R02						√	√	√	√												
R03										√	√	√									
R04													√	√	√						
R05																√	√	√	√		
R06																			√	√	
R07																			√		√

Sumber: Data Penelitian (2017)

Dalam penelitian ini terdapat beberapa aturan (*rule*) yang berkaitan dengan variabel sebagai berikut:

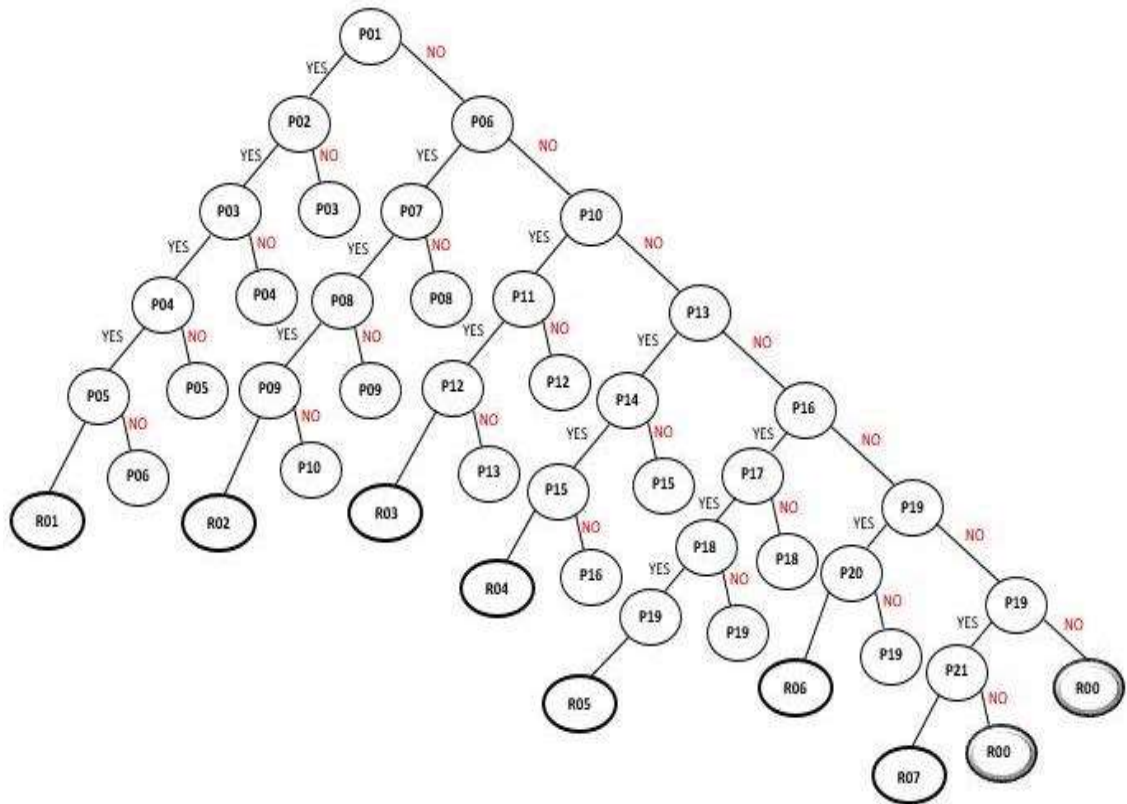
1. *IF P01 AND P02 AND P03 AND P04 AND P05 THEN R01*
2. *IF P06 AND P07 AND P08 AND P09 THEN R02*
3. *IF P10 AND P11 AND P12 THEN R03*
4. *IF P13 AND P14 AND P15 THEN R04*
5. *IF P16 AND P17 AND P18 AND P19 THEN R05*
6. *IF P19 AND P20 THEN R06*

7. IF P19 AND P21 THEN R07

Penjelasan dari aturan di atas:

1. Jika Kegagalan instalasi *flat cable* pada unit CCD dan Adanya debu pada cermin, lensa dan permukaan *white plate* dan Gangguan pada pencahayaan lampu penyalin dan Unit CCD tidak berfungsi normal dan Unit MCU PWB tidak berfungsi normal maka kerusakan CCD *White Level Error*.
2. Jika Unit cermin mengalami kecacatan dan Kabel *scanner* tidak tersambung dan Terjadi kesalahan pada *origin detection sensor* dan Pemanfaatan cermin pada *motor* tidak berfungsi normal maka kerusakan *Scanner Feed Trouble*.
3. Jika Unit lampu penyalin tidak berfungsi normal dan Pemanfaatan unit lampu penyalin tidak optimal dan Unit CCD PWB tidak berfungsi normal maka kerusakan *Copy Lamp (Xenon Lamp) Error*.
4. Jika Konektor pada unit IMC PWB tidak tersambung dan Kerusakan pada pin konektor motherboard dan Cacatnya ROM pada unit IMC PWB, terjadi kegagalan pada data maka kerusakan IMC PWB *Communication Trouble*.
5. Jika Konektor unit LSU atau bagian dalam unit LSU tidak tersambung dan Putaran pada *polygon motor* tidak berfungsi normal dan *Laser* tidak menerang dan Kegagalan pada unit MCU PWB maka kerusakan *LSU Trouble*.
6. Jika Kegagalan pada unit MCU PWB dan Koneksi pada bagian unit *paper exit fan* bermasalah maka kerusakan *Paper Exit Fan Lock Trouble*.
7. Jika Kegagalan pada unit MCU PWB dan Koneksi pada kipas bermasalah, penyuplaian tidak tersambung maka kerusakan *Power Fan Lock Trouble*.

Berdasarkan tabel aturan di atas, maka dapat dibuat sebuah pohon keputusan sebagai berikut:



Gambar 3.2 Pohon Keputusan

Sumber: Data Penelitian (2017)

7 3.4.2 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. (Rosa dan Shalahuddin, 2011)

Selain itu, UML juga merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek. (Rosa dan Shalahuddin, 2011)

UML dideskripsikan oleh beberapa diagram, diantaranya:

1. *Use Case Diagram*

Use Case Diagram digunakan untuk menggambarkan sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case diagram* lebih difokuskan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Sebuah *use case diagram* mempresentasikan sebuah interaksi antara aktor dengan sistem melalui sebuah cerita bagaimana sebuah sistem itu dipakai. Sedangkan *use case diagram* memfasilitasi komunikasi di antara analis dan pengguna serta antara analis dan *client*. Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan dan diharapkan pengguna.

a. Definisi Aktor

Definisi aktor adalah aktivitas yang bisa dilakukan oleh para aktor dalam menggunakan sistem pakar ini.

Tabel 3.5 Definisi Aktor

No	Aktor	Deskripsi
1	Admin	Aktor yang mengelola segala isi data dalam sistem pakar.
2	User	Aktor yang melakukan diagnosa kerusakan.

Sumber: Data Penelitian (2017)

b. Definisi *Use Case*

Definisi dalam *use case* diagram adalah kegiatan-kegiatan yang akan terjadi di dalam sistem antara para aktor dengan *use case*.

Tabel 3.6 Definisi *Use Case*

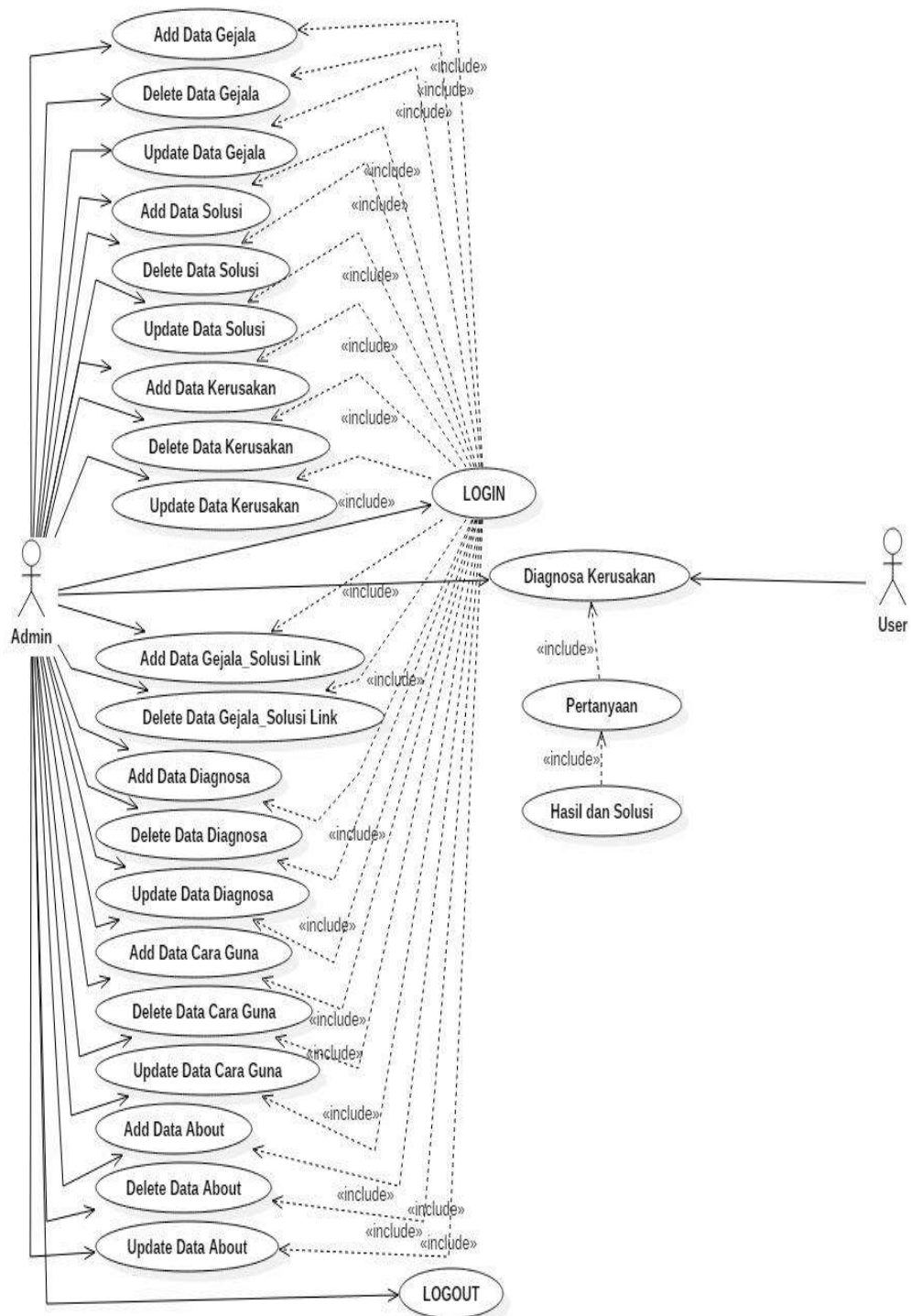
No	<i>Use Case</i>	Deskripsi	Aktor
1	<i>Add Data Gejala</i>	Admin dapat melakukan penambahan data gejala dalam <i>database</i> .	Admin
2	<i>Delete Data Gejala</i>	Admin dapat melakukan penghapusan data gejala dalam <i>database</i> .	Admin
3	<i>Update Data Gejala</i>	Admin dapat melakukan peng- <i>update</i> -an data gejala dalam <i>database</i> .	Admin
4	<i>Add Data Solusi</i>	Admin dapat melakukan penambahan data solusi dalam <i>database</i> .	Admin
5	<i>Delete Data Solusi</i>	Admin dapat melakukan penghapusan data solusi dalam <i>database</i> .	Admin
6	<i>Update Data Solusi</i>	Admin dapat melakukan peng- <i>update</i> -an data solusi dalam <i>database</i> .	Admin
7	<i>Add Data Kerusakan</i>	Admin dapat melakukan penambahan data kerusakan dalam <i>database</i> .	Admin

Tabel 3.7 Lanjutan

8	<i>Delete Data Kerusakan</i>	Admin dapat melakukan penghapusan data kerusakan dalam <i>database</i> .	Admin
9	<i>Update Data Kerusakan</i>	Admin dapat melakukan peng- <i>update</i> -an data kerusakan dalam <i>database</i> .	Admin
10	<i>Add Data Gejala > Solusi Link</i>	Admin dapat melakukan penambahan data gejala > solusi link dalam <i>database</i> .	Admin
11	<i>Delete Data Gejala > Solusi Link</i>	Admin dapat melakukan penghapusan data gejala > solusi link dalam <i>database</i> .	Admin
12	<i>Add Data Diagnosa</i>	Admin dapat melakukan penambahan data diagnosa dalam <i>database</i> .	Admin
13	<i>Delete Data Diagnosa</i>	Admin dapat melakukan penghapusan data diagnosa dalam <i>database</i> .	Admin
14	<i>Update Data Diagnosa</i>	Admin dapat melakukan peng- <i>update</i> -an data diagnosa dalam <i>database</i> .	Admin

15	<i>Add Data Cara Guna</i>	Admin dapat melakukan penambahan data cara guna dalam <i>database</i> .	Admin
16	<i>Delete Data Cara Guna</i>	Admin dapat melakukan penghapusan data cara guna dalam <i>database</i> .	Admin
17	<i>Update Data Cara Guna</i>	Admin dapat melakukan peng- <i>update</i> -an data cara guna dalam <i>database</i> .	Admin
18	<i>Add Data About</i>	Admin dapat melakukan penambahan data about dalam <i>database</i> .	Admin
19	<i>Delete Data About</i>	Admin dapat melakukan penghapusan data about dalam <i>database</i> .	Admin
20	<i>Update Data About</i>	Admin dapat melakukan peng- <i>update</i> -an data about dalam <i>database</i> .	Admin
21	<i>Login</i>	Proses untuk masuk ke dalam sistem pakar.	Admin
22	<i>Logout</i>	Proses untuk keluar dari sistem pakar.	Admin
23	Diagnosa Kerusakan	Tampilan untuk melakukan diagnosa kerusakan.	Admin <i>User</i>
24	Pertanyaan	Tampilan untuk menampilkan pertanyaan mengenai kerusakan mesin <i>fotocopy</i> .	Admin <i>User</i>
25	Hasil dan Solusi	Tampilan untuk menampilkan hasil dan solusi dari diagnosa kerusakan.	Admin <i>User</i>

Sumber: Data Penelitian (2017)

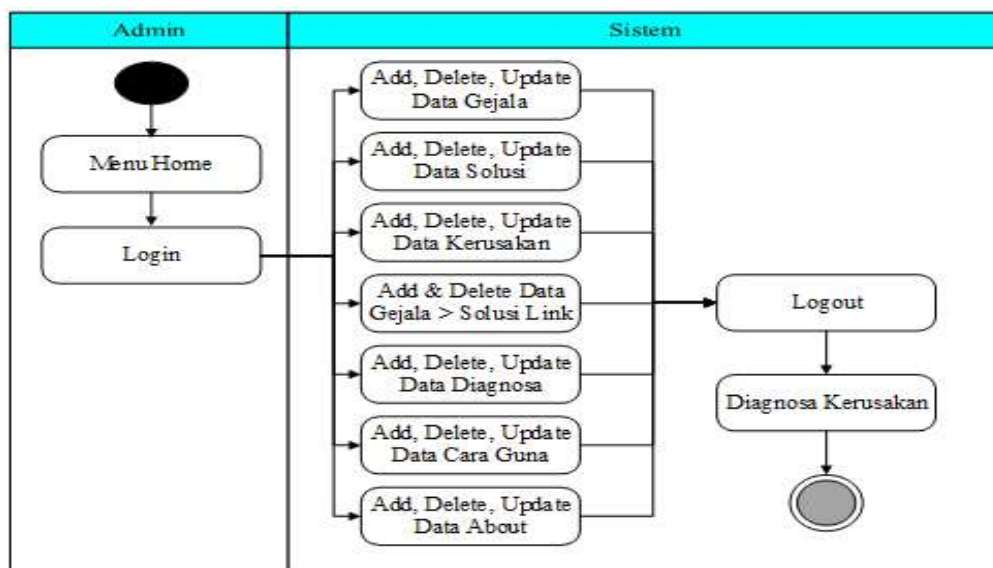


Gambar 3.3 Diagram Use Case Admin dan User
 Sumber: Data Penelitian (2017)

Dalam sistem ini, terdiri dari aktor berupa admin dan *user*. Fitur yang dapat diakses oleh admin, yaitu *login* untuk admin melakukan perintah *add*, *delete*, *update* pada data gejala, data solusi, data kerusakan, data diagnosa, data cara guna dan data *about*. Namun untuk data gejala > solusi link hanya bisa melakukan perintah *add* dan *delete*. Dan disini admin juga dapat mengakses menu diagnosa. Selain itu, fitur *logout* yang diakses admin untuk keluar dari *login* sistem. Berbeda dengan *user* yang hanya dapat mengakses menu diagnosa untuk mendeteksi kerusakan mesin *fotocopy* dengan menjawab pertanyaan yang terdapat dalam menu diagnosa dan kemudian akan menghasilkan solusi berupa cara memperbaiki.

2. Activity Diagram

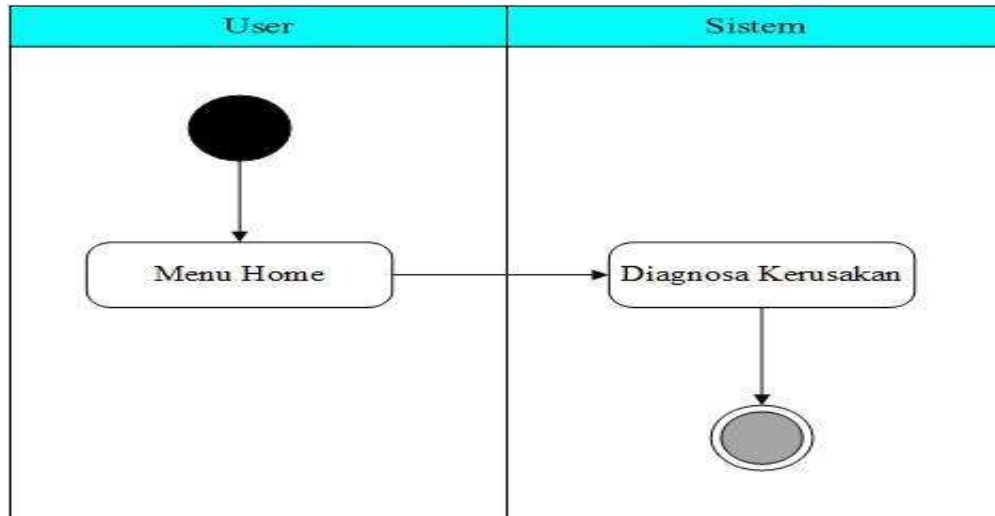
Berikut ini merupakan *activity diagram* dari Aplikasi Sistem pakar mendiagnosa kerusakan mesin fotocopy SHARP MX-M264N untuk *Admin* dan *User* :



Gambar 3.4 Activity Diagram Admin

Sumber: Data Penelitian (2017)

Berikut ini merupakan diagram *activity* untuk *user*:



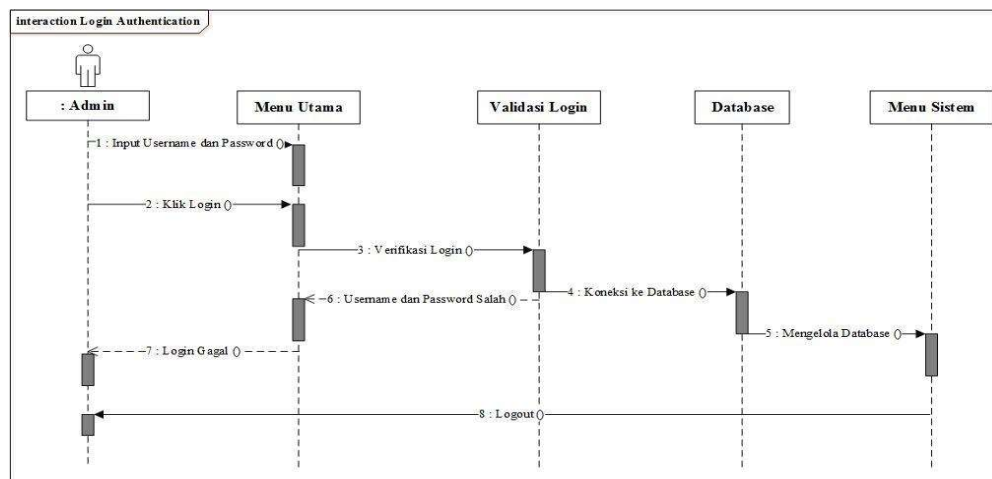
Gambar 3.5 Activity Diagram User

Sumber: Data Penelitian (2017)

3. Sequence Diagram

Berikut ini terdapat beberapa diagram sekuen yang ada di dalam sistem pakar mendiagnosa kerusakan mesin *fotocopy* SHARP MX-M264N:

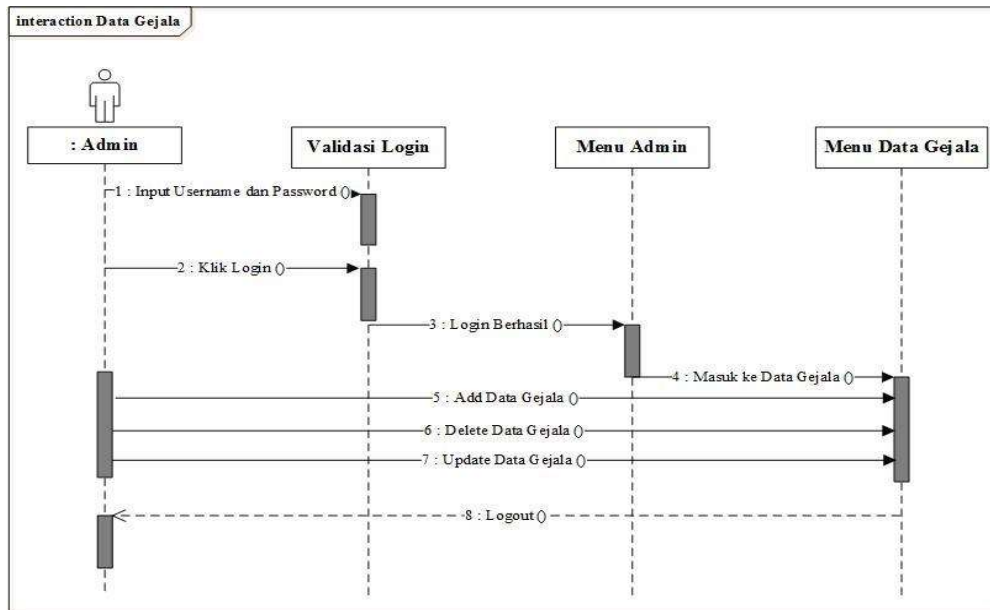
a. Sequence Diagram Login Admin



Gambar 3.6 Sequence Diagram Login Admin

Sumber: Data Penelitian (2017)

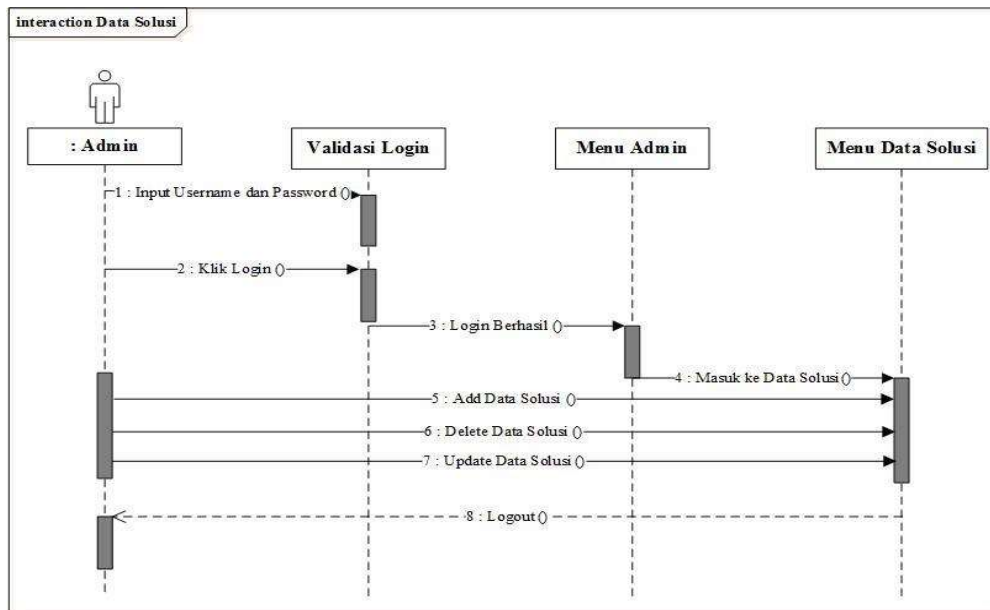
b. *Sequence Diagram Data Gejala*



Gambar 3.7 *Sequence Diagram Data Gejala*

Sumber: Data Penelitian (2017)

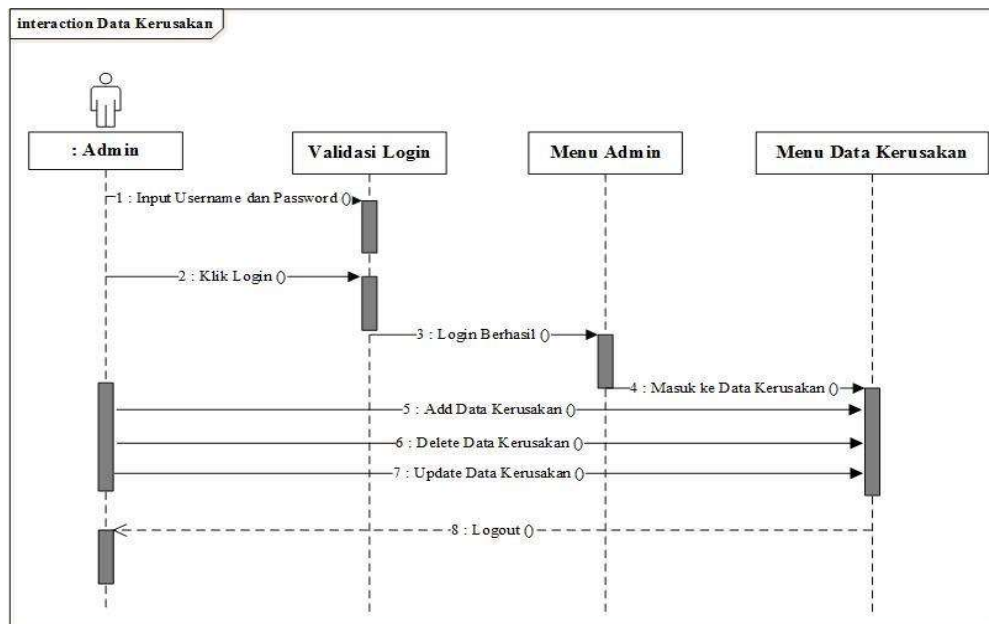
c. *Sequence Diagram Data Solusi*



Gambar 3.8 *Sequence Diagram Data Solusi*

Sumber: Data Penelitian (2017)

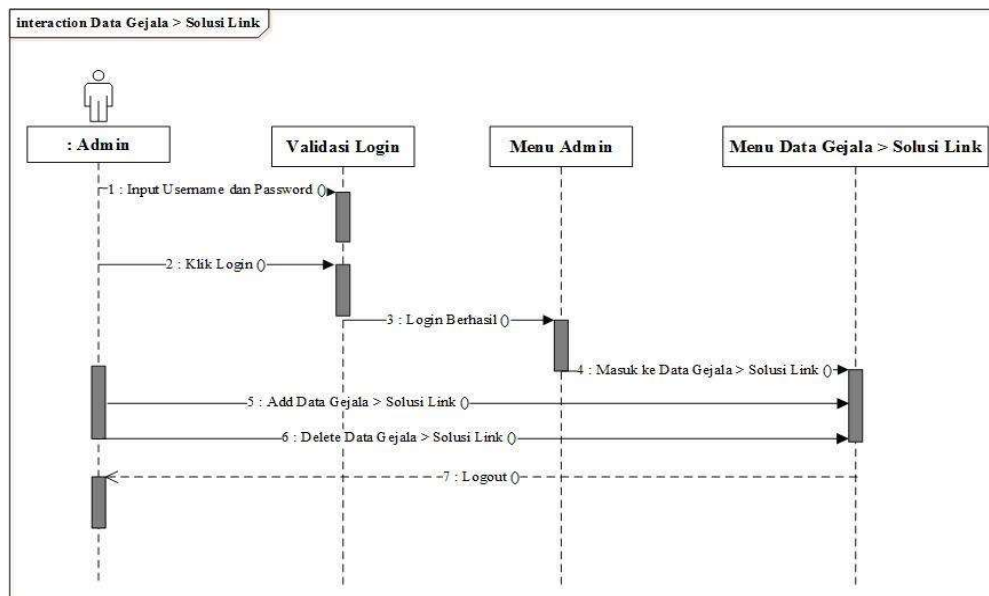
d. *Sequence Diagram Data Kerusakan*



Gambar 3.9 *Sequence Diagram Data Kerusakan*

Sumber: Data Penelitian (2017)

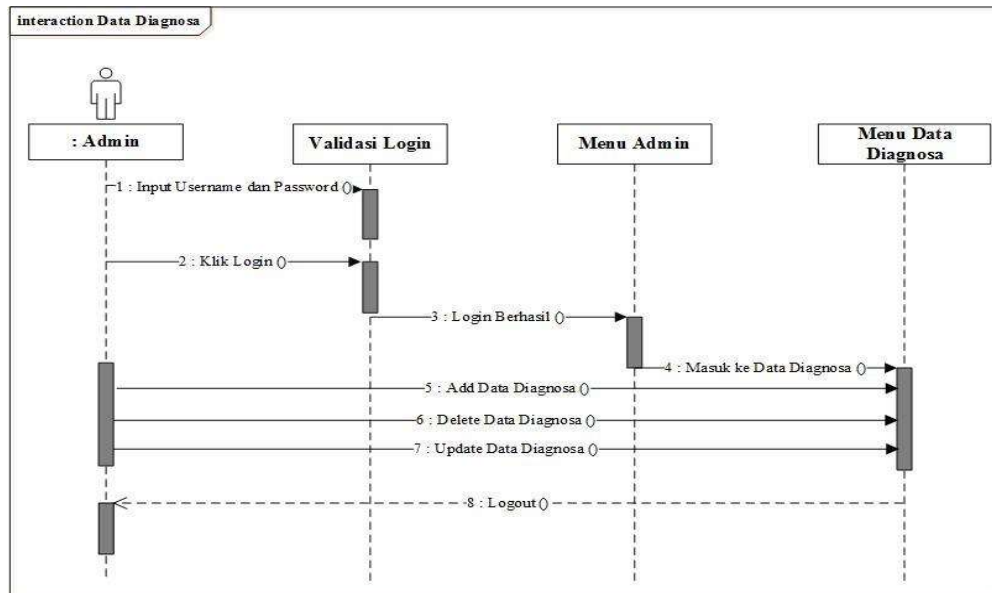
e. *Sequence Diagram Data Gejala > Solusi Link*



Gambar 3.10 *Sequence Diagram Data Gejala > Solusi Link*

Sumber: Data Penelitian (2017)

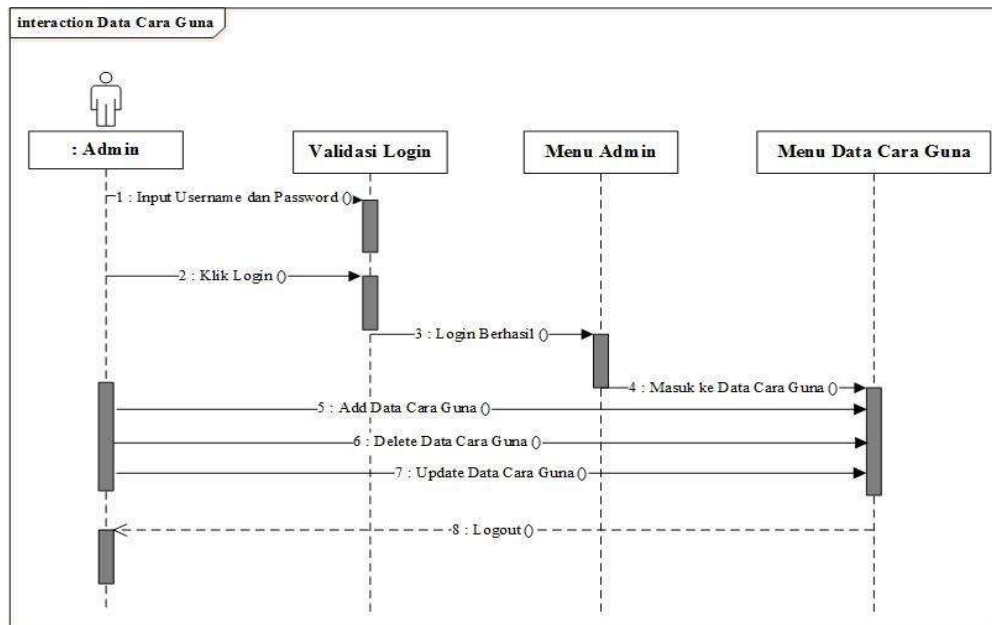
f. *Sequence Diagram Data Diagnosa*



Gambar 3.11 *Sequence Diagram Data Diagnosa*

Sumber: Data Penelitian (2017)

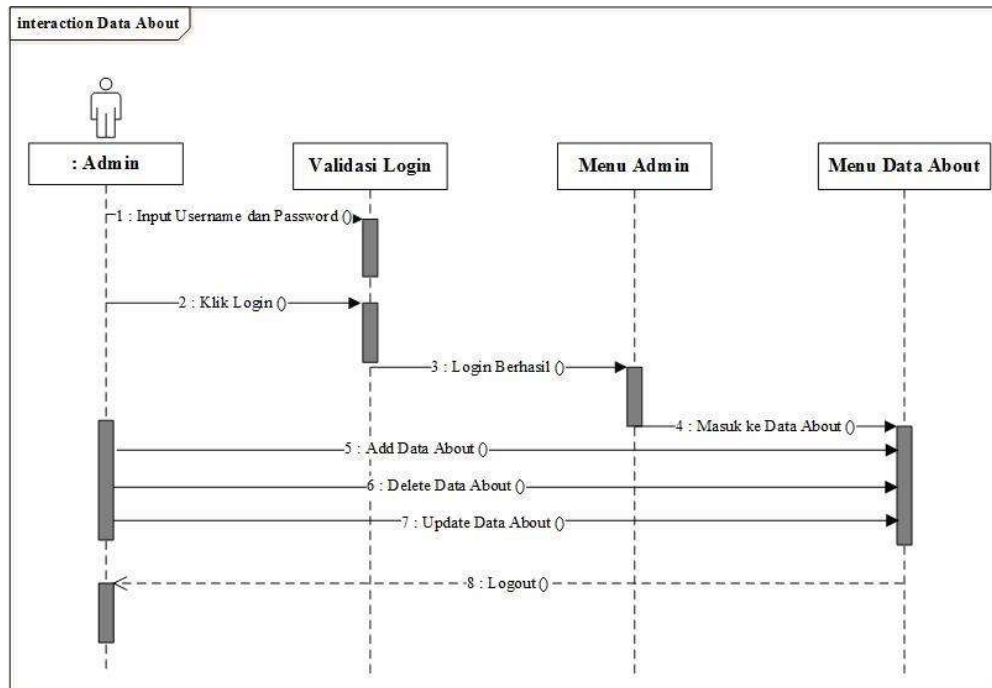
g. *Sequence Diagram Data Cara Guna*



Gambar 3.12 *Sequence Diagram Data Cara Guna*

Sumber: Data Penelitian (2017)

h. *Sequence Diagram Data About*



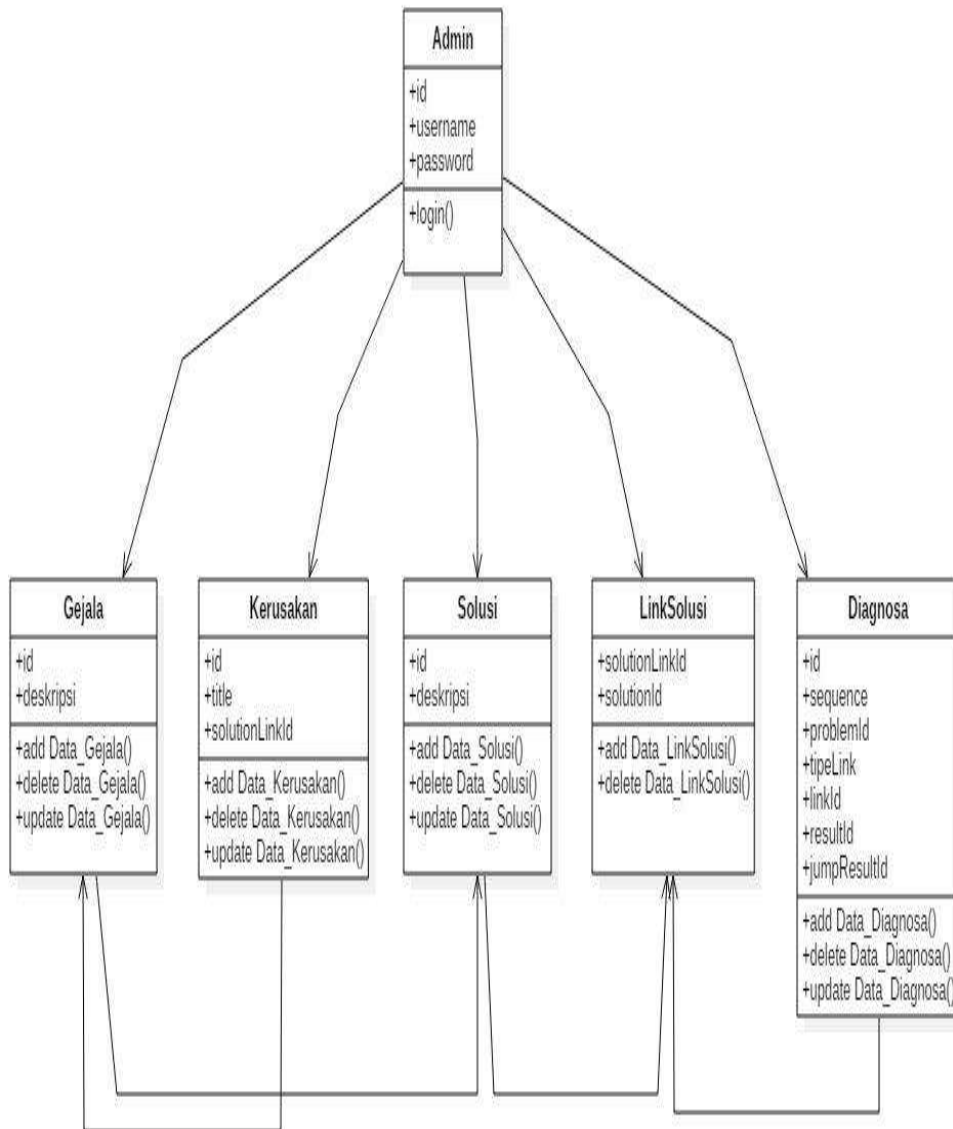
Gambar 3.13 *Sequence Diagram Data About*

Sumber: Data Penelitian (2017)

4. *Class Diagram*

Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam sistem perangkat lunak yang akan dikembangkan. *Class Diagram* menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

Berikut ini digambarkan *class diagram* dari aplikasi sistem pakar mendiagnosa kerusakan mesin *fotocopy* SHARP MX-M264N:



Gambar 3.14 *Class Diagram*

Sumber: Data Penelitian (2017)

8 3.4.3 Perancangan Database

Pada perancangan *database* aplikasi sistem pakar mendiagnosa kerusakan mesin *fotocopy* SHARP MX-M264N menggunakan *database MySql* yang dikombinasikan dengan Bahasa pemrograman *PHP*.

Berikut ini merupakan struktur *database* yang digunakan untuk sistem pakar kerusakan mesin *fotocopy* SHARP MX-M264N:

1. Tabel Admin

Tabel admin berfungsi untuk menyimpan data *login* sebagai admin yang terdiri dari *username* dan *password*. Struktur tabel admin dapat dilihat pada tabel 3.8.

Tabel 3.7 Tabel Admin

<i>Field</i>	Tipe	Panjang	Keterangan
Id	Int	10	<i>Primary Key</i>
Username	Varchar	255	
Password	Varchar	255	

Sumber: Data Penelitian (2017)

2. Tabel Gejala

Tabel gejala berfungsi untuk menyimpan data gejala kerusakan mesin *fotocopy* SHARP MX-M264N. Struktur tabel gejala dapat dilihat pada tabel 3.9.

Tabel 3.8 Tabel Gejala

<i>Field</i>	Tipe	Panjang	Keterangan
Id	Varchar	3	<i>Primary Key</i>
Deskripsi	Varchar	255	

Sumber: Data Penelitian (2017)

3. Tabel Kerusakan

Tabel kerusakan berfungsi untuk menyimpan data kerusakan mesin *fotocopy* SHARP MX-M264N. Struktur tabel kerusakan dapat dilihat pada tabel 3.10.

Tabel 3.9 Tabel Kerusakan

<i>Field</i>	Tipe	Panjang	Keterangan
Id	Varchar	3	<i>Primary Key</i>
Title	Varchar	255	
solutionLinkId	Varchar	3	

Sumber: Data Penelitian (2017)

4. Tabel Solusi

Tabel solusi berfungsi untuk menyimpan data solusi kerusakan mesin *fotocopy* SHARP MX-M264N. Struktur tabel solusi dapat dilihat pada tabel 3.11.

Tabel 3.10 Tabel Solusi

<i>Field</i>	Tipe	Panjang	Keterangan
Id	Varchar	3	<i>Primary Key</i>
Deskripsi	Varchar	255	

Sumber: Data Penelitian (2017)

5. Tabel LinkSolusi

Tabel linksolusi berfungsi untuk menghubungkan tabel kerusakan dan tabel solusi, sehingga menghasilkan solusi untuk suatu kerusakan tertentu yang terjadi pada mesin *fotocopy* SHARP MX-M264N.

Tabel 3.11 Tabel LinkSolusi

<i>Field</i>	Tipe	Panjang	Keterangan
SolutionLinkId	Varchar	3	<i>Primary Key</i>
SolutionId	Varchar	3	<i>Primary Key</i>

Sumber: Data Penelitian (2017)

6. Tabel Diagnosa

Tabel diagnosa berfungsi untuk menyimpan data hasil diagnosa berdasarkan pertanyaan yang dijawab oleh pengguna.

Tabel 3.12 Tabel Diagnosa

<i>Field</i>	Tipe	Panjang	Keterangan
Id	Int	10	<i>Primary Key</i>
Sequence	tinyInt	3	
problemId	Varchar	3	
tipeLink	Varchar	3	
linkId	Varchar	3	
resultId	Varchar	3	
Jump_ResultId	Varchar	3	

Sumber: Data Penelitian (2017)

9 3.4.4 *Prototype*

Menurut Rosa dan Shalahuddin (2011: 29) model prototipe dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Lalu dibuatlah program protipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi. Program ini biasanya menyediakan tampilan dengan simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi.

1. Form Halaman Utama

Form halaman utama merupakan tampilan awal ketika admin atau pengguna mengakses ke dalam *website* sistem pakar ini.

The image shows a web form layout. At the top is a box labeled "Header". Below it, on the left, is a login section with labels "Username" and "Password" above two input fields, and a "Login" button below. On the right is a vertical navigation menu with buttons for "Home", "Diagnosa", "Cara Guna", and "About", followed by a larger "LOGIN" button. At the bottom is a box labeled "Footer".

Gambar 3.15 Form Halaman Utama

Sumber: Data Penelitian (2017)

2. Form *Login Admin*

Form *login admin* merupakan tampilan bagi admin untuk menambah, menghapus dan meng-*update* data.

The image shows a login form for an administrator. It contains labels for "Username" and "Password" above two input fields. Below the input fields is the text "Login sebagai pakar untuk mengubah data" and a "Login" button.

Gambar 3.16 Form *Login Admin*

Sumber: Data Penelitian (2017)

3. Form Diagnosa

Form diagnosa merupakan tampilan untuk admin maupun pengguna untuk melakukan diagnosa kerusakan dengan menjawab pertanyaan yang telah disediakan.

Gambar 3.17 Form Diagnosa

Sumber: Data Penelitian (2017)

4. Form Cara Guna

Form cara guna merupakan tampilan yang berisi penjelasan untuk kegunaan-kegunaan dari menu yang terdapat pada sistem dan cara melakukan diagnosa.

Gambar 3.18 Form Cara Guna

Sumber: Data Penelitian (2017)

5. Form *About*

Form *about* merupakan tampilan yang berisi tentang informasi penulis.

Header	
Tentang Saya Terdiri dari Nama Peneliti, NPM, Bahasa Pemrograman yang digunakan dalam penelitian, Metode yang digunakan dalam penelitian, deskripsi web	Home Diagnosa Cara Guna About LOGIN
Footer	

Gambar 3.19 Form *About*

Sumber: Data Penelitian (2017)

3.5 Lokasi dan Jadwal Penelitian

10 3.5.1 Lokasi Penelitian

Penelitian dilaksanakan di PT. Syindo Mulia Perkasa yang beralamat di Ruko Permata Regency, Kota Batam.

11 3.5.2 Jadwal Penelitian

Penelitian dilakukan dan disesuaikan dengan jadwal dimulai sejak September 2016 dan berakhir pada bulan Januari 2017. Berikut dijabarkan jadwal penelitian lengkapnya dalam bentuk tabel:

Tabel 3.13 Jadwal Penelitian

No	Keterangan	2016												2017											
		Sept				Okt				Nov				Des				Jan				Feb			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2		
1	Pengajuan Judul Skripsi	■	■	■																					
2	Survei	■	■	■	■	■	■																		
3	Pengajuan Surat Penelitian						■	■																	
4	Wawancara							■	■	■	■	■	■	■											
5	Pengambilan Data										■	■	■	■											
6	Pengolahan Data												■	■	■										
7	Hasil Penelitian															■	■	■	■	■	■	■	■		

Sumber: Data Penelitian (2017)