

**SISTEM PAKAR MENDETEKSI KERUSAKAN MESIN
MANUFAKTUR BERBASIS DESKTOP**



Oleh:

Kelbin

140210191

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS PUTERA BATAM
2018-2019**

**SISTEM PAKAR MENDETEKSI KERUSAKAN MESIN
MANUFAKTUR BERBASIS DESKTOP**

SKRIPSI
Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana



Oleh:

Kelbin

140210191

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS PUTERA BATAM
2018-2019**

PERNYATAAN

Dengan ini menyatakan bahwa:

1. Skripsi ini berupa 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 berupa 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 individu 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 saksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi.

Batam, 11 Februari 2019
Yang membuat pernyataan,

Kelbin
NPM: 140210191

**SISTEM PAKAR MENDETEKSI KERUSAKAN MESIN
MANUFAKTUR BERBASIS DESKTOP**

HALAMAN PENGESAHAN

**Oleh
Kelbin
140210191**

SKRIPSI

**Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana**

**Telah disetujui oleh Pembimbing pada tanggal
seperti tertera dibawah ini**

Batam, 11 Februari 2019

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

Pembimbing

ABSTRAK

Penelitian ini dilatar belakangi oleh permasalahan seringkali terjadi gangguan dalam bagian produksi apabila terjadi masalah pada mesin manufaktur. Hal ini dikarenakan para *operator* tidak mengenal kerusakan yang dialami oleh mesin yang rusak, mereka harus menunggu Seorang teknisi untuk memeriksa serta memperbaiki mesin tersebut. Tujuan penelitian ini berupa untuk mengoptimisasi pekerjaan di dalam industri dengan merancang sebuah sistem pakar deteksi kerusakan mesin manufaktur yang dapat menggantikan peran Seorang teknisi di dalam industri tersebut. Sistem ini memanfaatkan metode *forward chaining*. Sistem ini dibuat dengan memanfaatkan Bahasa pemograman *Vb.NET*, serta *database SQLSERVER* untuk penyimpanan data. Hasil penelitian menunjukkan bahwa aplikasi sistem pakar tersebut dapat membantu para pengguna mendeteksi kerusakan yang dialami mesin manufaktur. Kesimpulannya adalah *operator* manufaktur dapat berkonsultasi dengan sistem layaknya berkonsultasi langsung dengan Seorang pakar untuk mendeteksi kerusakan yang terjadi pada mesin manufaktur berdasarkan gejala-gejala yang dialami mesin.

Kata Kunci: Sistem Pakar, Mesin Manufaktur, *SQLSERVER*, *VB.NET*, *Forward Chaining*

ABSTRACT

This research is motivated by the problem of frequent disturbances in the production section if there is a problem with the manufacturing machine. This is because operators do not know the problem in a damaged machine, they have to wait for a technician to check and repair the machine. The purpose of this study is to optimize work within the company by designing an expert system for manufacturing engine damage detection that can replace the role of a technician within the company. This system uses the forward chaining method. This system is created using the Vb.NET programming language, and SQLSERVER database for data storage. The results of the study show that the application of expert systems can help users detect the damage experienced by manufacturing machines. The conclusion is that manufacturing machine users can consult with a system like consulting directly with an expert to detect damage that occurs in manufacturing machines based on the symptoms experienced by the engine.

Keyword: *Expert System, Manufacturing Machine, SQLSERVER, Vb.NET, Forward Chaining*

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang adalah 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. 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. Kepada individu tua penulis, yang terus mendoakan keberhasilan penulis menyelesaikan skripsi ini.
6. Teman-teman seperjuangan yang juga selalu memberikan motivasi baik berupa *sharing* pendapat, motivasi dan hal-hal lainnya dalam rangka pembuatan skripsi ini.
7. Serta semua pihak yang tak dapat penulis sebutkan satu persatu yang telah membantu dalam penyusunan skripsi ini.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan selalu mencurahkan hidayah serta taufik-Nya, Amin.

Batam, 11 Februari 2019

Kelbin

DAFTAR ISI

PERNYATAAN	iii
HALAMAN PERNYATAAN	iii
HALAMAN PENGESAHAN	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Identifikasi Masalah.....	2
1.3. Batasan Masalah	2
1.4. Rumusan Masalah.....	3
1.5. Tujuan Penelitian.....	3
1.6. Manfaat Penelitian.....	4
BAB II LANDASAN TEORI	5
2.1. Teori Dasar	5
2.1.1 Kecerdasan Buatan (<i>Artificial Intelligence</i>)	5
2.1.1.1. Sistem Pakar atau Expert System.....	5
2.1.1.2. Jaringan Syaraf Tiruan	10
2.1.1.3. Logika Fuzzy.....	11
2.1.2 Tabel Keputusan	12
2.1.3 Pohon Keputusan.....	13
2.2. Variabel Penelitian.....	14
2.2.1 Mesin Manufaktur	15
2.3. <i>Software</i> Pendukung	15
2.3.1 UML (<i>Unified Modeling Language</i>)	15
2.3.1.1. Use Case Diagram.....	16

2.3.1.2. Activity Diagram	17
2.3.1.3. Sequence Diagram	19
2.3.1.4. Class Diagram	20
2.3.2 Visual Studio	23
2.3.3 Database SQL <i>Server</i>	24
2.4. Penelitian Terdahulu	24
2.5. Kerangka Pemikiran	27
2.5.1 Pakar / Ahli	28
2.5.2 <i>User</i>	29
2.5.3 Input <i>Rule / Data</i>	29
2.5.4 Pilih Mesin.....	29
2.5.5 Input Masalah Mesin	29
2.5.6 Hasil Deteksi.....	29
BAB III METODE PENELITIAN	30
3.1. Desain Penelitian	30
3.2. Operasional Variabel	32
3.2.1. Metode <i>Forward Chaining</i>	32
3.3. Teknik Pengumpulan Data	34
3.3.1. Metode Studi Pustaka	35
3.3.2. Metode Wawancara	36
3.4. Perancangan Sistem.....	36
3.4.1. UML (<i>Unified Modeling Language</i>)	36
3.4.2. Desain Basis Data.....	52
BAB IV HASIL DAN PEMBAHASAN.....	56
4.1. Hasil Penelitian.....	56
4.1.1. Implementasi Antar Muka	56
4.2. Pembahasan	61
4.2.1. Rencana Pengujian.....	61
4.2.2. Hasil Pengujian.....	62
BAB V KESIMPULAN DAN SARAN	67
5.1. Kesimpulan.....	67

5.2. Saran	67
DAFTAR PUSTAKA	68
DAFTAR RIWAYAT HIDUP	70
SURAT KETERANGAN PENELITIAN.....	71
LAMPIRAN SOUCE CODE	72
LAMPIRAN HASIL TURNITIN	102

DATFTAR GAMBAR

Gambar 2.1 Struktur Sistem Pakar.....	7
Gambar 2.2 Pohon Keputusan.....	14
Gambar 2.3 Logo StarUML	22
Gambar 2.4 Logo Visual Studio 2010.....	23
Gambar 2.5 Logo SQL Server	24
Gambar 2.6 Kerangka Pemikiran Penelitian.....	28
Gambar 3. 1 Desain Penelitian.....	30
Gambar 3. 2 Pohon Keputusan.....	35
Gambar 3. 3 Use Case Diagram.....	37
Gambar 3. 4 Class Diagram	38
Gambar 3. 5 Activity Diagram Login	39
Gambar 3. 6 Activity Diagram Tambah Data Mesin	40
Gambar 3. 7 Activity Diagram Ubah Mesin	41
Gambar 3. 8 Activity Diagram Tambah User	42
Gambar 3. 9 Activity Diagram Ubah User.....	43
Gambar 3. 10 Activity Diagram Tambah Rule	44
Gambar 3. 11 Activity Diagram Ubah Rule.....	45
Gambar 3. 12 Activity Diagram Tambah Kerusakan.....	46
Gambar 3. 13 Activity Diagram Ubah Kerusakan	47
Gambar 3. 14 Activity Diagram Deteksi.....	48
Gambar 3. 15 Activity Diagram Lihat Laporan Kerusakan.....	48
Gambar 3. 16 Sequence Diagram Login	49
Gambar 3. 17 Sequence Diagram Mesin.....	49
Gambar 3. 18 Sequence Diagram Kerusakkan	50
Gambar 3. 19 Sequence Diagram Rule	50
Gambar 3. 20 Sequence Diagram User	51
Gambar 3. 21 Sequence Diagram Deteksi Kerusakan	51
Gambar 3. 22 Sequence Diagram Lihat Laporan Kerusakkan.....	52
Gambar 4. 1 Form Login.....	56
Gambar 4. 2 Form Menu Utama	57
Gambar 4. 3 Form Mengelola Data Mesin.....	57
Gambar 4. 4 Form Mengelola Data Kerusakan	58
Gambar 4. 5 Form Mengelola Data Rule	59
Gambar 4. 6 Form Mengelola Data User	60
Gambar 4. 7 Form Deteksi Kerusakan.....	61

DAFTAR TABEL

Tabel 2.1 Tabel Keputusan	13
Tabel 2.2 Simbol Use Case Diagram	16
Tabel 2.3 Simbol Activity Diagram	18
Tabel 2.4 Simbol Sequence Diagram.....	19
Tabel 2.5 Simbol Class Diagram	21
Tabel 3. 1 Indikator Kerusakan Mesin Manufaktur	33
Tabel 3. 2 Kode Indikator Kerusakan Mesin Manufaktur	33
Tabel 3. 3 Kode Gejala Kerusakan Mesin Manufaktur.....	33
Tabel 3. 4 Tabel KeputusanS Kerusakan Mesin Manufaktur	34
Tabel 3. 5 Desain Tabel User.....	52
Tabel 3. 6 Desain Tabel Mesin	53
Tabel 3. 7 Desain Tabel Kerusakan	53
Tabel 3. 8 Desain Tabel Pertanyaan.....	53
Tabel 3. 9 Desain Tabel Deteksi	54
Tabel 4. 1 Tabel Rencana Pengujian.....	62
Tabel 4. 2 Tabel Pengujian Login	62
Tabel 4. 3 Tabel Pengujian Pengolahan Data Mesin	63
Tabel 4. 4 Tabel Pengujian Pengolahan Data Kerusakan	64
Tabel 4. 5 Tabel Pengujian Pengolahan Data Rule.....	64
Tabel 4. 6 Tabel Pengujian Pengolahan Data <i>User</i>	65
Tabel 4. 7 Tabel Pengujian Pilih Mesin Deteksi Kerusakan.....	66

BAB I

PENDAHULUAN

1.1. Latar Belakang

PT Celindo Pratama adalah sebuah industri yang telah berdiri sejak 19 Oktober 2013 dan bergerak di bidang aspek memproduksi serta mengolah logam. Untuk dapat mengolah bahan logam dengan cepat dan presisi tinggi, maka perlu memanfaatkan mesin.

Meskipun untuk mengolah bahan logam tersebut memerlukan mesin, tetapi mesti juga memerlukan tenaga manusia untuk mengoperasikan mesin tersebut, biasanya daya tahan sebuah mesin akan menyusut dan akan mengalami kerusakan jika sudah dipakai kurang lebih 5 tahun, Apabila mesin mengalami kerusakan, operator harus menunggu seorang teknisi untuk memperbaiki atau mengeceknya, hal ini dikarenakan kurangnya pengetahuan pada *operatornya*. Hal ini sangat merugikan industri karena produksi oleh mesin tersebut harus dihentikan hingga selesai di perbaiki dan biayanya juga sangat tinggi.

Untuk membantu mengurangi waktu perbaikan mesin dan mengurangi biaya perbaikan, penulis akan merancang sebuah sistem yang menggunakan *método* sistem pakar sehingga para *operator* bisa mengetahui kerusakan pada mesin dengan dibantu oleh sistem tanpa harus menunggu kehadiran teknisi.

Maka penulis ingin melakukan penelitian dengan judul “**SISTEM PAKAR MENDETEKSI KERUSAKAN MESIN MANUFAKTUR BERBASIS DESKTOP**”

1.2. Identifikasi Masalah

Berdasarkan uraian diatas, maka permasalahan yang ada adalah sebagai berikut:

- a. Minimnya pengetahuan tentang kerusakan yang ada pada mesin-mesin manufaktur.
- b. Kurangnya sistem yang bisa mendeteksi kerusakan yang ada pada mesin-mesin manufaktur
- c. Tidak ada riwayat untuk kejadian kerusakan yang terjadi pada mesin-mesin.

1.3. Batasan Masalah

Batasan masalah yang ada adalah sebagai berikut:

- a. Penulis melakukan analisi kerusakan mesin pada PT Celindo Pratama, tidak untuk PT yang lain.
- b. Metode yang digunakan dalam sistem pakar ini adalah metode *Forward Chaining*.
- c. Bahasa Pemograman yang digunakan peneliti dalam sistem pakar ini adalah bahasa pemograman *VB.NET* dan *SQL Server* sebagai basis data.

- d. Data didapatkan melalui wawancara dengan teknisi dan data dari referensi internet, buku dan jurnal.

1.4. Rumusan Masalah

Berdasarkan identifikasi masalah di atas, maka rumusan masalah dalam penelitian ini berupa:

- a. Bagaimana metode *Forward Chaining* bisa mendeteksi jenis kerusakan yang terjadi pada mesin manufaktur?
- b. Bagaimana metode *Forward Chaining* dan bahasa pemrograman VB.NET dapat mendeteksi kerusakan pada mesin manufaktur?
- c. Bagaimana sebuah sistem pakar dapat menyimpan dan menampilkan riwayat kerusakan yang pernah terjadi pada mesin tersebut?

1.5. Tujuan Penelitian

Berdasarkan rumusan masalah diatas, maka tujuan penelitian pada penelitian ini berupa:

- a. Membuat sistem yang bisa membantu *operator* untuk mengetahui kerusakan yang sedang dialami oleh mesin.
- b. Menjelaskan kerusakan pada mesin manufaktur dengan memanfaatkan metode *Forward Chaining*.

- c. Memakai metode *Forward Chaining* dan bahasa pemrograman VB.NET untuk mendeteksi masalah pada mesin manufaktur.

1.6. Manfaat Penelitian

Berdasarkan penelitian ini, dapat di harapkan memberikan dua manfaat bagi peneliti sendiri dan objek yang diteliti, yaitu:

- a. Manfaat Teoritis

Untuk mengetahui secara detail tentang sistem pakar dengan metode *Forward Chaining* dan sistem pakar metode *Forward Chaining* mendeteksi kerusakan mesin manufaktur.

- b. Manfaat Praktis

Hasil penelitian ini bisa dijadikan sebagai media informasi untuk masyarakat serta menjadi alat deteksi kerusakan mesin manufaktur pada PT Celindo Pratama.

BAB II

LANDASAN TEORI

2.1. Teori Dasar

2.1.1 Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan Buatan atau *Artificial Intelligence* (AI), *Intelligence* yang berarti kecerdasan, *Artificial* berarti buatan. Kecerdasan buatan yang dimaksud di sini mengacu pada mesin yang mampu berpikir, mengukur tindakan yang diambil dan membuat keputusan.

Kecerdasan Buatan adalah kecerdasan yang ditambahkan pada suatu sistem yang bisa diatur dalam konteks ilmiah atau bisa disebut juga intelegensi artifisial, *Artificial Intelligence* (AI) dalam Bahasa Inggris. Kecerdasan Buatan juga merupakan kemampuan sistem untuk menafsirkan data eksternal dengan benar, untuk belajar dari data tersebut, dan menggunakan pembelajaran tersebut guna mencapai tujuan dengan melalui adaptasi yang fleksibel. (Kaplan & Haenlein, 2019). Berikut berupa metode-metode Kecerdasan Buatan (AI):

2.1.1.1. Sistem Pakar atau *Expert System*

Sistem Pakar atau *Expert System* adalah sistem komputer yang mengemulasi kemampuan pengambilan keputusan seorang ahli manusia, Sistem pakar dirancang untuk memecahkan masalah yang kompleks dengan bernalar melalui kumpulan pengetahuan, yang diwakili terutama seolah-olah memerintah daripada melalui kode prosedural konvensional, sistem pakar yang *modern* dapat

ditingkatkan dengan penambahan basis pengetahuan atau set aturan seperti permainan catur dan sistem diagnosis medis (Nwigbo Stella & Agbo Okechuku Chuks, 2015).

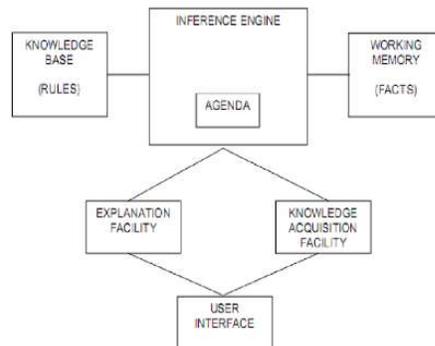
a. **Keuntungan Sistem Pakar**

Secara garis besar, banyak manfaat yang dapat di ambil dengan adanya sistem pakar, antara lain (Kusumadewi, 2003:110):

1. Memungkinkan orang awam untuk melakukan pekerjaan para ahli
2. Menyimpan pengetahuan dan keahlian pakar
3. Meningkatkan output dan produktivitas
4. Meningkatkan kualitas
5. Memiliki realiabilitas
6. Meningkatkan kapabilitas sistem computer
7. Sebagai media pelengkap dalam pelatihan
8. Permanen
9. Memiliki kemampuan untuk mengakses pengetahuan

b. **Struktur Sistem Pakar**

Adapun struktur sistem pakar dapat dilihat pada gambar berikut (Rosnelly, 2012).



Gambar 2.1 Struktur Sistem Pakar

Berikut berupa komponen yang terdapat dalam struktur sistem pakar:

1. *Knowledge Base* (Base Pengetahuan)

Adalah program komputer yang beralasan dan menggunakan basis pengetahuan untuk memecahkan masalah yang kompleks. Istilah ini luas dan mengacu pada berbagai jenis sistem. Satu tema umum yang menyatukan semua sistem berbasis pengetahuan adalah upaya untuk mewakili pengetahuan secara eksplisit dan sistem penalaran yang memungkinkannya untuk memperoleh pengetahuan baru.

2. *Inference Engine* (Mesin Inferensi)

adalah komponen sistem yang menerapkan aturan logis ke basis pengetahuan untuk menyimpulkan informasi baru. Mesin inferensi pertama adalah komponen sistem pakar. Mesin inferensi menerapkan aturan logis ke basis pengetahuan dan menyimpulkan pengetahuan baru. Proses ini akan berulang karena setiap fakta

baru dalam basis pengetahuan dapat memicu aturan tambahan dalam mesin inferensi. Mesin inferensi bekerja terutama di salah satu dari dua mode baik aturan khusus atau fakta, yaitu *forward chaining* dan *backward chaining*.

3. *Working Memory*

Untuk menyimpan data dari *inference engine*.

4. *Explanation Facility*

Modul yang memungkinkan sistem pakar untuk memberikan penjelasan kepada pengguna tentang mengapa ia mengajukan pertanyaan dan bagaimana hal itu mencapai beberapa kesimpulan.

5. *Knowledge Acquisition Facility*

Mencakup proses memasukan informasi-informasi seorang pakar/ahli ke dalam database program, untuk memperbaiki ataupun mengembangkan basis pengetahuan.

6. *User Interface*

Antarmuka untuk dilihat dan di pakai oleh pengguna sistem dan mengubahnya menjadi data yang dapat diterima oleh sistem. Antarmuka juga menerima informasi dari sistem dan muncul dalam wujud yang pengguna bisa mengerti.

c. ***Inference Engine* atau Mesin Inferensi**

Mesin inferensi adalah komponen sistem yang menerapkan aturan logis ke basis pengetahuan untuk menyimpulkan informasi baru. Mesin inferensi pertama

adalah komponen sistem pakar. Sistem pakar yang khas terdiri dari basis pengetahuan dan mesin inferensi. Basis pengetahuan menyimpan fakta tentang dunia. Mesin inferensi menerapkan aturan logis ke basis pengetahuan dan menyimpulkan pengetahuan baru. Proses ini akan berulang karena setiap fakta baru dalam basis pengetahuan dapat memicu aturan tambahan dalam mesin inferensi. Mesin inferensi bekerja terutama di salah satu dari dua mode baik aturan khusus atau fakta: forward chaining dan backward chaining. Forward chaining dimulai dengan fakta yang diketahui dan menegaskan fakta baru. Rantai mundur dimulai dengan tujuan, dan bekerja mundur untuk menentukan fakta apa yang harus ditegaskan agar tujuan dapat dicapai (Lenat D.B. & Weyer, 2016).

Ada dua cara inferensi di dalam sistem pakar yaitu:

1. Rumut Maju (*Forward Chaining*)

Forward chaining adalah pencocokan data atau pernyataan mulai dari sisi kiri (*IF* dulu). Penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis (Octavina & Fadlil, 2014).

2. Rumut Balik (*Backward Chaining*)

Backward chaining adalah adalah pencocokan data atau pernyataan mulai dari sisi kiri (*THEN* dulu). Penalaran dimulai dari hipotesa terlebih dahulu. Dan untuk menguji fakta hipotesis harus ditemukan di basis pengetahuan (Octavina & Fadlil, 2014).

d. **Karakteristik Sistem Pakar.**

Sistem pakar memiliki beberapa karakteristik umum, berupa kinerja yang baik, waktu respons yang cepat, sangat andal, mudah dipahami, dan fleksibel.

2.1.1.2. Jaringan Syaraf Tiruan

adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan sistem saraf manusia. JST merupakan sistem adaptif yang dapat mengubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. Oleh karena sifatnya yang adaptif, JST juga sering disebut dengan jaringan adaptif (Irwansyah & Faisal, 2015).

Secara sederhana, JST adalah sebuah alat pemodelan data statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data. Menurut suatu teorema yang disebut "teorema penaksiran universal", JST dengan minimal sebuah lapis tersembunyi dengan fungsi aktivasi non-linear dapat memodelkan seluruh fungsi terukur Boreal apapun dari suatu dimensi ke dimensi lainnya (Irwansyah & Faisal, 2015). Proses belajarnya JST berbeda dengan metode yang lain, dimana *Neural Network* memiliki struktur yang distingtif, berikut adalah kelompok metode pelatihannya:

a. *Feed Forward Neural Network (FFNN)*

Feed Forward Neural adalah jenis spesifik dari jaringan saraf tiruan awal yang dikenal karena kesederhanaan desainnya. Jaringan neural feedforward memiliki lapisan input, lapisan tersembunyi dan lapisan output. Informasi selalu bergerak dalam satu arah, dari lapisan input ke lapisan output dan tidak pernah bergerak ke belakang (Irwansyah & Faisal, 2015).

b. *Feed Back Neural Network (FBNN)*

Feed Back Neural Network adalah kelas jaringan saraf tiruan di mana koneksi antara node membentuk grafik diarahkan sepanjang urutan. Ini memungkinkannya untuk menunjukkan perilaku dinamis temporal untuk urutan waktu. Tidak seperti jaringan neural feedforward, RNNs dapat menggunakan status internal (memori) untuk memproses urutan input. Ini membuatnya dapat diterapkan untuk tugas-tugas seperti pengakuan tulisan tangan yang tidak tersegmentasi, terhubung atau pengenalan ucapan (Irwansyah & Faisal, 2015).

2.1.1.3. Logika Fuzzy

Logika *Fuzzy* adalah cabang dari *soft computing*. Logika *Fuzzy* adalah suatu teori kompilasi logika yang terus dikembangkan supaya dapat menggantikan konsep nilai yang ada diantara benar dan salah (Irwansyah & Faisal, 2015). Logika *Fuzzy* terdapat 3 metode, yaitu:

a. Metode Mamdani

Metode Mamdani sering juga dikenal dengan nama metode MIN-MAX.

berikut beberapa tahap untuk mendapatkan output (Pusadan, 2014) :

1. Penyusunan Himpunan *fuzzy*
2. Aplikasi Fungsi Implikasi
3. Komposisi Aturan
4. Defuzzifikasi

b. Metode Sugeno

Berupa penalaran Bersama cara *output* sistem, bukan himpunan *fuzzy*, tetapi konstanta atau persamaan linier (Rofiq, 2013).

c. Metode Tsukamoto

Semua *rule* menggunakan himpunan-himpunan *fuzzy*, dengan fungsi keanggotaan yang menonton, Untuk mendapatkan hasil yang benar, itu dicari dengan memodifikasi input penggunaan menjadi angka di *domain* yang ditetapkan oleh *fuzzy* (Sholihin, Fuad, & Khamiliyah, 2013).

2.1.2 Tabel Keputusan

Tabel yang digunakan sebagai alat bantu untuk menyelesaikan logika dalam program. Algoritma yang berisi keputusan bertingkat yang banyak sekali sangat sulit untuk digambarkan langsung dengan structured English atau pseudocode dan dapat dibuat terlebih dahulu dengan menggunakan tabel keputusan. Dengan

demikian tabel keputusan efektif digunakan bilamana kondisi yang akan diseleksi didalam program jumlahnya cukup banyak dan rumit. (Herjanto & Herfan, 2008).

Tabel 2.1 Tabel Keputusan

	Hipotesa 1	Hipotesa 2	Hipotesa 3	Hipotesa 4
Indikator A	Y	Y	N	Y
Indikator B	Y	N	Y	N
Indikator C	Y	N	Y	N
Indikator D	N	Y	N	Y

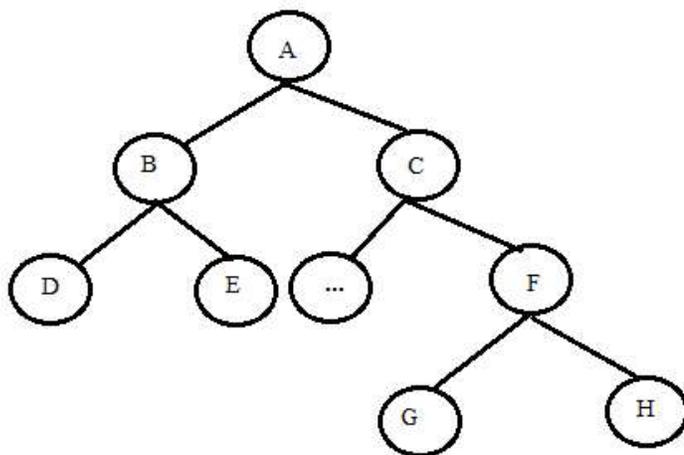
2.1.3 Pohon Keputusan

Pohon keputusan adalah salah satu metode klasifikasi yang paling populer karena mudah untuk di interpretasi oleh manusia. Pohon keputusan adalah model prediksi menggunakan struktur pohon atau struktur berhirarki. Pohon keputusan adalah salah satu metode klasifikasi yang paling populer karena mudah untuk di interpretasi oleh manusia. Pohon keputusan adalah model prediksi menggunakan struktur pohon atau struktur berhirarki (Kusrini & Luthfi, 2009).

Manfaat utama dari menggunakan pohon keputusan adalah kemampuan mereka untuk memecah proses pengambilan keputusan yang kompleks menjadi lebih sederhana sehingga pengambil keputusan akan lebih baik menafsirkan solusi untuk masalah tersebut. Decision Tree juga berguna untuk mengeksplorasi data,

menemukan hubungan tersembunyi antara sejumlah variabel input prospektif dengan variabel target (Kusrini & Luthfi, 2009).

Pohon keputusan harus dirancang hati-hati dengan cara manual atau secara otomatis dengan memakai satu atau lebih algoritme pohon keputusan untuk membuat kumpulan data tanpa diklasifikasikan.



Gambar 2.2 Pohon Keputusan

2.2. Variabel Penelitian

Variabel penelitian merupakan bagian krusial dari penelitian. Kemampuan peneliti dalam mengerti variable penelitian sangat tergantung pada pemahannya konsep dalam penelitian, terpenting adalah variable penelitian. Pengalaman peneliti dalam melakukan penelitian dan menyiapkan proposal penelitian juga dapat membantu pengetahuan tentang kemampuan untuk menentukan variable penelitian. Variable merupakan rancangan yang dioperasionalkan. (Swarjana, 2015).

Variabel yang digunakan peneliti adalah sebagai berikut:

2.2.1 Mesin Manufaktur

Indikator kerusakan yang terdapat pada mesin manufaktur adalah sebagai berikut:

1. Pompa air pendingin

Adalah tipe pompa yang dipakai untuk mensirkulasi pendingin, biasanya dalam bentuk air (Singh, 2006:148).

2. Minyak Pelumas

Adalah sebuah zat yang dipakai untuk mengurangi gesekan antara dua permukaan yang saling bersentuhan. (Singh, 2006:290).

3. Sparepart

Adalah unsur yang bisa dipertukarkan dan disimpan dalam inventaris lalu dipakai untuk perbaikan atau penggantian unit gagal (Singh, 2006:467).

2.3. *Software* Pendukung

Dipakai untuk membantu peneliti dalam membangun sistem yang nanti akan menjadi sebuah aplikasi. Aplikasi pendukung untuk merancang sistem ini berupa:

2.3.1 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah standar yang sering digunakan untuk mendefinisikan persyaratan. Membuat analisis serta desain, dan

menggambarkan arsitektur dalam *Object Oriented Programming* (OOP) (Rosa & Shalahuddin, 2011).

Berikut adalah beberapa jenis diagram UML:

2.3.1.1. Use Case Diagram

Use case Diagram adalah gambaran graphical dari *actor*, *use case*, dan interaksi yang memperkenalkan suatu sistem. *Use Case* menggambarkan hubungan dari satu atau lebih actor dengan sistem informasi yang mau dibuat. *Use Case* digunakan untuk mencari tahu fungsi apa yang ada dalam sistem informasi dan siapa yang berhak memanfaatkan fungsi tersebut.

Use Case Diagram adalah permodelan untuk kelakuan sistem informasi yang mau dibuat, *Use Case* menggambarkan interaksi antara satu atau beberapa actor dengan sistem informasi yang mau dibuat. *Use Case* biasanya digunakan untuk mencari informasi tentang fungsi yang berada dalam sistem informasi dan siapa individu yang berhak memanfaatkan fungsi-fungsi tersebut. Ada dua poin utama pada *Use Case*, Yaitu Actor dan *Use Case* (Rosa & Shalahuddin, 2015).

Tabel 2.2 Simbol Use Case

SIMBOL	DESKRIPSI
<i>Use Case</i>	Fungsi disediakan di sistem sebagai unit yang bertukar pesan antara unit atau actor, biasanya diekspresikan dengan menggunakan kata kerja di mulainya frase <i>use case</i>

Aktor / <i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem
Asosiasi/ <i>association</i> 	Komunikasi antara aktor dan <i>Use Case</i> yang berpartisipasi dalam <i>Use Case</i>
Ekstensi / <i>Extension</i> 	Hubungan <i>use case</i> tambahan dengan <i>use case</i> dimana <i>case</i> yang ditambahkan dapat berdiri sendiri bahkan tanpa <i>use case</i> tambahan
Generalisasi / <i>Generalization</i> 	Generelasi hubungan dan spesialisasi antara dua <i>use case</i> dimana satu fungsi lebih umum dari pada yang lain.
Menggunakan / <i>Include</i> 	Hubungan <i>use case</i> tambahan dengan <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau syarat untuk memakai <i>use case</i> ini

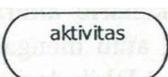
2.3.1.2. Activity Diagram

Activity Diagram visualkan alur kerja, yang perlu dipertimbangkan di sini berarti bahwa diagram yang menggambarkan aktivitas sistem bukanlah apa yang dilakukan actor, sehingga aktivitas dapat dilakukan oleh sistem.

Activity Diagram menggambarkan *workflow* atau aktivitas sistem atau proses bisnis atau menu yang ada di dalam *software*. Yang perlu telitikan di sini adalah *activity diagram* yang menggambarkan aktivitas sistem bukanlah yang dilakukan actor (Rosa & Shalahuddin, 2015). *Activity Diagram* juga sering dipakai untuk menentukan hal-hal berikut:

- a. Desain proses bisnis di mana setiap urutan kegiatan yang dijelaskan adalah sistem proses bisnis yang ditentukan
- b. Urutan atau pengelompokan tampilan antarmuka sistem di mana setiap aktivitas dianggap memiliki desain antarmuka layar
- c. Desain tes di mana setiap kegiatan dianggap membutuhkan tes yang perlu didefinisikan sebagai kasus uji
- d. Design menu ditampilkan oleh perangkat lunak

Tabel 2.3 Simbol Activity Diagram

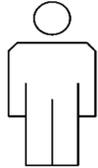
SIMBOL	DESKRIPSI
STATUS AWAL 	Status awal aktivitas sistem, setiap diagram aktivitas pasti memiliki status awal
Aktivitas / <i>Activity</i> 	Aktivitas dalam sistem, biasanya diawali dengan kata kerja
Percabangan / <i>Decision</i> 	Asosiasi percabangan dimana jika ada pemilihan

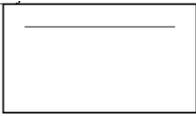
Penggabungan / <i>Join</i> 	Asosiasi penggabungan dimana beberapa aktivitas gabung menjadi satu
Status Akhir 	Status akhir akhir yang dilakukan sistem, setiap diagram aktivitas pasti memiliki status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.3.1.3. Sequence Diagram

Sequence Diagram merupakan gubahan dari *Diagram Class* dan *diagram Object* yang memiliki suatu gambaran model statis, ada juga yang bersifat dinamis, seperti *Diagram Interaction*. *Diagram sequence* merupakan salah satu diagram *Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan; message (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut (Rosa & Shalahuddin, 2015).

Tabel 2.4 Simbol *Sequence Diagram*

SIMBOL	DESKRIPSI
Aktor / Actor 	Menggambarkan seseorang yang berinteraksi dengan sistem, di mana hanya bisa menginputkan informasi dan menerima informasi dari sistem dan tidak memegang kendali pada use case
Garis hidup / lifeline 	Menyatakan <i>lifetime</i> satu objek

Objek 	Menyatakan objek yang berinteraksi pesan
Waktu Aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
Pesan tipe <i>create</i> <code><<create>></code> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan tipe <i>call</i> Nama_metode() 	Memanggil operasi / metode yang ada
Pesan tipe <i>send</i> 	Mengirim informasi ke objek lainnya
Pesan tipe <i>return</i>	Kembalikan satu data / informasi setelah telah menjalankan suatu operasi / metode
Pesan tipe <i>destroy</i>	Mengakhiri hidup objek yang lain

2.3.1.4. Class Diagram

Class diagram adalah model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class. Class diagram mirip ER-Diagram

pada perancangan database, bedanya pada ER-diagram tdk terdapat operasi/methode tapi hanya atribut (Rosa & Shalahuddin, 2015).

Kelas-kelas dalam struktur sistem harus bisa meneksekusi fungsi sebagai persyaratan sistem sehingga *developer* dapat membuat kelas dalam programnya seperti desain diagram kelas. Struktur-struktur kelas yang baik dalam diagram kelas harus memiliki jeni kelas-kelas berikut (Rosa & Shalahuddin, 2015):

1. *Main Class*

Kelas yang memiliki fungsi awal ketika dibuka

2. *View Class*

Kelas yang mengatur tampilan ke aktor.

3. *Controller Class*

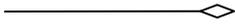
Kelas yang memproses fungsi-fungsi yang akan diambil dari definisi *use case*

4. *Model Class*

Kelas yang dipakai untuk memegang data menjadi suatu kesatuan yang diambil ataupun disimpan ke basis data

Tabel 2.5 Simbol *Class Diagram*

SIMBOL	DESKRIPSI
<p>Kelas / <i>Class</i></p> 	<p>Kelas pada struktur sistem</p>

<p>Antarmuka / <i>Interface</i></p> 	Tampilan / desain yang di tampilkan ke pemakai
<p>Asosiasi berarah / <i>direct association</i></p> 	Relasi antarkelas dengan makna umum
<p>Generalisasi / <i>Generalization</i></p>	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum-khusus)
<p>Gantungan / <i>Dependency</i></p> 	Relasi antar kelas dengan makna keberagantungan kelas
<p>Agresi / <i>aggregation</i></p> 	Relasi antar kelas dengan makna semua bagian

Biasanya untuk menggambarkan diagram-diagram (seperti diagram-diagram di atas), bisa menggunakan aplikasi yang bernama StarUML. StarUML adalah *platform* pemodelan *software* yang mendukung UML (*Unified Modeling Language*). StarUML berdasarkan UML versi 1.4, menyiapkan 11 jenis *diagram*, serta kompatibel dengan notasi UUM 2.0 StarUML dan juga secara aktif mendukung pendekatan MDA (*Model Driven Architercture*). StarUML lebih dikenal untuk



Gambar 2.3 Logo StarUML

keunggulannya dalam hal menyesuaikan lapangan kerja pengguna, dan memiliki ekstensi tinggi untuk fungsinya (Triandini & Suardika, 2012: 1).

2.3.2 Visual Studio

Adalah sebuah perangkat lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web.

Dengan Visual Studio ini, penulis menuliskan *coding* aplikasi memanfaatkan bahasa pemrograman sebagai berikut.



Gambar 2.4 *Visual Studio*

a. Bahasa Pemrograman Vb. Net

Adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak di atas sistem .NET Framework, dengan menggunakan bahasa BASIC. Dengan menggunakan alat ini, para *programmer* dapat membangun aplikasi Windows Forms, Aplikasi web berbasis ASP.NET, dan juga aplikasi *command-line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti *Microsoft Visual C++*, *Visual C#*, atau *Visual J#*), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio .NET.

2.3.3 Database SQL Server

Microsoft SQL Server adalah *software* untuk *manage* relasional basis data Microsoft (RDBMS). Bahasa dari query utama adalah Bahasa Transact - SQL, adalah penerapan SQL standar ANSI / ISO yang dipakai oleh *Microsoft* serta *Sybase*. SQL server bisa digunakan untuk pribadi maupun industri kecil, menengah, ataupun besar.



Gambar 2.5 Logo *SQL Server*

2.4. Penelitian Terdahulu

Penelitian ini dilakukan untuk meningkatkan hasil penelitian terdahulu yang sudah pernah diteliti, ada beberapa penelitian sebelumnya sebagai berikut:

Menurut Supyani, Bebas Widata, Wawan Laksito dengan judul “**APLIKASI DETEKSI KERUSAKAN MESIN SEPEDA MOTOR BEBEK 4 TAK DENGAN METODE *FORWARD CHAINING* (ISSN: 2338-4018, 2014)**”, Pesatnya pertumbuhan sepeda motor tentunya juga harus didukung oleh kesiapan mekaniknya, sebab semakin banyak jumlah sepeda motor yang digunakan individu akan semakin banyak pula timbulnya kerusakan mesin. Dalam prakteknya pabrik telah melatih mekaniknya untuk menyelesaikan masalah kerusakan sepeda motor yang muncul, karena banyaknya pengguna sepeda motor maka jumlah itu tidak cukup untuk menyelesaikan masalah kerusakan sepeda motor, untuk itu banyak

bengkel yang berdiri untuk dapat membantu menyelesaikan masalah kerusakan sepeda motor.

Menurut Kresna Amanda dengan judul “**PENERAPAN SISTEM PAKAR UNTUK MENDETEKSI PENYAKIT PADA KEHAMILAN (ISSN: 1978-1946, 2015)**”, Kehamilan adalah suatu fenomena fisiologis yang dimulai dengan pembuahan dan diakhiri dengan proses persalinan. Selama kehamilan, ibu dan janin adalah unit fungsi yang tak terpisahkan. Meskipun terlihat dengan kondisi kehamilan yang sehat bukan berarti ibu dan janin dalam keadaan baik – baik saja. Namun kurangnya informasi atau sosialisasi tentang penyakit kehamilan akan menyebabkan mereka baru mengetahui adanya penyakit yang menyertai kehamilannya setelah stadium lanjut.

Menurut Rosmawati Tamin dengan judul “**SISTEM PAKAR UNTUK DETEKSI KERUSAKAN PADA PRINTER MEMANFAATKAN METODE FORWARD CHAINING (SSN: 2442-4512, 2015)**”, Mengganti pengetahuan manusia ke dalam bentuk sistem dengan mengakomodasi kemampuan / keahlian Seorang ahli untuk mengolah analisis suatu masalah sehingga sistem tersebut dapat bekerja dengan sistem untuk memecahkan masalah seperti halnya manusia melakukannya dan memecahkan masalah. Kerusakan printer terkadang merupakan masalah besar ketika orang awam tidak tahu lokasi kesalahan printer, itu membutuhkan sistem yang mampu bekerja secara otomatis untuk memberikan solusi untuk merusak printer. Penelitian ini bertujuan untuk merancang sistem yang dapat digunakan untuk menangani kerusakan printer. Pengguna aplikasi ini tampaknya berhadapan langsung dengan para ahli di aspek perangkat keras,

terutama printer. Perencanaan sistem dilakukan dengan membuat basis pengetahuan menggunakan pohon keputusan dan jika-maka aturan sebagai representasi pengetahuan. Sistem ini dibuat dengan menggunakan metode forward chaining dan bahasa pemrograman Visual Basic. Hasil penelitian ini mengungkapkan jenis kerusakan yang terjadi pada printer dan penanganan kerusakan. Pengujian aplikasi juga dilakukan untuk menentukan akurasi dan variasi serta keramahan pengguna dan fleksibilitas sistem. Hasil dari keseluruhan pengujian ini dapat disimpulkan bahwa program ini cukup baik walaupun jenis kerusakan yang dihasilkan tidak lengkap karena sistem ini hanya mendeteksi 15 jenis kerusakan mesin secara umum.

Menurut Samy Salim Abu Nazer dan Mohammed Ibrahim Alhabbash dengan Judul “***MALE INFERTILITY EXPERT SYSTEM DIAGNOSES AND TREATMENT (ISSN : 2429-5396, 2016)***”, *Infertility affects up to 15% of couples of reproductive age all over the world, the prevalence of infertility is said to be increasing globally and Male infertility is the most prevalent. Infertility means no pregnancy after one year of marriage, 40% due to male factor and 40% due to female factor and 20% due to both factors. Many men do not know something about infertility.*

Menurut Samy Saling Abu Nazer dan Ali Osama Mahdi dengan judul “***A PROPOSED EXPERT SYSTEM FOR FOOT DISEASES DIAGNOSIS (ISSN : 2429-5396, 2008)***”, *Feet are complex structures with 26 bones, 33 joints, and many muscles, nerves, ligaments of various types. Any part of the foot may be affected. Some leg disorders may arise only with blunt pain, but on the other hand other leg*

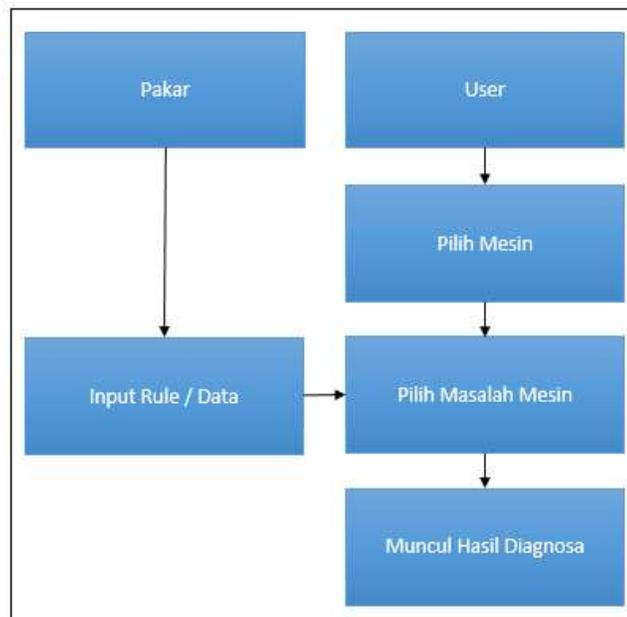
disorders can be very severe and limit the ability to walk or tolerate weight. The bulk of small foot pain cases are treated at home with special care. When severe pain is encountered, it requires several types of medical treatment. If proper treatment for leg pain is not done quickly, it can cause damage to the foot. Objective: The main objective of this expert system is to get the right diagnosis of the disease and the right treatment. Method: In this paper the design of the proposed Expert System is produced to help Podiatric doctors in diagnosing many foot diseases such as clubfoot, toes, stress fractures, ankle or ankle fractures, sprained legs, turtle legs, turtle feet, flat feet, Plantar Fasciitis, Warts, Bunion, Rheumatoid Arthritis, Gout, Heel Spurs, Athlete's Foot, Morton Neuroma, Cellulitis, Frostbite, and Gangrene. The proposed expert system provides an overview of the foot disease given, the cause of the disease described and treatment of the disease if possible is given. SL5 Language The Object Expert System is used to design and implement the proposed expert system. Results: The proposed expert system for diagnosing foot disease was evaluated by medical students and they were satisfied with their performance. Conclusion: The proposed expert system is very useful for Podiatric doctors, patients with foot problems and newly graduated doctors.

2.5. Kerangka Pemikiran

Identifikasi masalah dalam penelitian ini berupa banyaknya *operator* manufaktur yang tidak bisa memperbaiki mesin diri sendiri karena kurangnya

pengetahuan dan tidak ada sebuah sistem yang bisa membantu pengguna untuk mengecek masalah pada mesin manufaktur, waktu untuk menunggu kedatangan teknisi dapat menyebabkan tidak capainya target hasil produksi karena *operator* tidak dapat bekerja ketika mesin sedang ada masalah.

Kerangka pemikiran sebagai berikut:



Gambar 2.6 Kerangka Pemikiran Penelitian

Uraian dari poin-poin diatas adalah sebagai berikut:

2.5.1 Pakar / Ahli

Pemakai yang memiliki akses untuk seluruh halaman yang terdapat pada sistem.

2.5.2 User

Pengguna system atau *operator*, biasanya menggunakan sistem untuk melaporkan kerusakan pada mesin ke dalam sistem. Tidak mempunyai otoritas untuk mengisi atau menambah *rule* ataupun *user* lain.

2.5.3 Input Rule / Data

Tambahkan data *Rule* dan mesin ke sistem, sehingga Pengguna dapat laporkan mesin yang rusak.

2.5.4 Pilih Mesin

Pengguna/User memilih Mesin yang ingin di Deteksi.

2.5.5 Input Masalah Mesin

Apabila *operator* mengalami masalah saat sedang menggunakan mesin, maka dia daftar masalah tersebut ke dalam sistem, serta gejala-gejala yang terjadi.

2.5.6 Hasil Deteksi

Dari gejala-gejala yang telah diinput, sistem Akan menampilkan hasil deteksi kerusakan mesin.

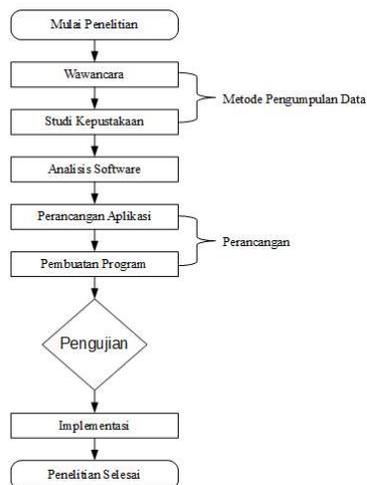
BAB III METODE PENELITIAN

3.1. Desain Penelitian

Desain penelitian ini sangat logis sehingga hasil penelitian dan kesimpulannya sesuai dengan focus penelitian.

Desain penelitian diperlukan dalam perencanaan dan pelaksanaan penelitian. Desain penelitian adalah kerangka kerja atau *blueprint* untuk melaksanakan proyek penelitian (Martono, 2013:131).

Dalam penelitian ini, peneliti menggunakan metode model sekuensial linier atau model air terjun, Alur Desain penelitian sebagai berikut:



Gambar 3. 1 Desain Penelitian

Desain penelitian untuk penelitian ini bisa dilihat pada gambar diatas, untuk penjelasan yang terkandung dalam penelitian ini sebagai berikut:

a. Mulai Penelitian

Langkat pertama untuk memulai sebuah penelitian adalah menentukan topik dan menentukan masalah yang diteliti. Penulis melakukan penelitian pada “Sistem Pakar Deteksi Kerusakan Pada Mesin Manufaktur”.

b. Wawancara

Setelah menentukan topik dan masalah pada penelitian, maka berikutnya adalah mengumpulkan data-data yang dibutuhkan oleh penelitian, bisa melalui wawancara dengan para ahli.

c. Studi Kepustakaan

Penulis juga mengumpulkan data dari buku dan jurnal dari internet.

d. Analisis *Software*

Setelah mendapatkan data yang dibutuhkan dalam penelitian, perangkat lunak apa yang harus ditentukan diperlukan dalam penelitian yang akan dilakukan. Di sini penulis memerlukan perangkat lunak SQL Server untuk database, *Microsoft Visual Studio* untuk bagian pengkodean.

e. Perancangan Aplikasi

Setelah perangkat lunak yang diperlukan ditentukan, selanjutnya harus mulai merancang aplikasi.

f. Pembuatan Program

Setelah perancangan aplikasi selesai, kita harus mulai melakukan pemograman atau *coding* aplikasi, bahasa pemograman yang digunakan oleh penulis berupa Vb.Net serta *software* pendukung lainnya.

g. Pengujian

Langkah selanjutnya adalah melakukan uji coba memanfaatkan mesin yang sebenarnya, untuk menentukan apakah aplikasi berjalan dengan benar dan sesuai dengan *target*. Pendekatan yang digunakan adalah pendekatan *Black-box*, yang hanya menguji apakah program sudah berjalan dengan benar atau sesuai *request*.

h. Implementasi

Setelah pengujian sudah selesai, maka harus memasang aplikasi di lapangan untuk dipakai *operator*.

i. Selesai

Sampai sini, aplikasi tetap harus dirawat dan dimonitor apakah ada masalah pada sistemnya.

3.2. Operasional Variabel

Konsep yang nilainya ingin diketahui oleh peneliti. Tidak sedikit variabel yang terlibat dalam suatu penelitian sifatnya abstrak, dalam arti tidak jelas wujud dan ukurannya, sehingga sulit juga ditentukan nilainya (Sugiyono, 2010: 58).

3.2.1. Metode *Forward Chaining*

Forward chaining adalah pencocokan data atau pernyataan mulai dari sisi kiri (*IF* dulu). Penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis (Octavina & Fadlil, 2014).

Tabel 3. 1 Indikator Kerusakan Mesin Manufaktur

VARIABEL	INDIKATOR
Mesin Manufaktur	<ol style="list-style-type: none"> 1. Masalah pada pompa air pendingin 2. Masalah pada minyak pelumas 3. Masalah pada sparepart internal mesin

Tabel 3. 2 Kode Indikator Kerusakan Mesin Manufaktur

KODE	INDIKATOR	GEJALA
FK01	Masalah pada pompa air pendingin	<ol style="list-style-type: none"> 1. Temperatur mesin tinggi 2. Pompa air pendingin tidak berputar 3. Pompa air pendingin bocor
FK02	Masalah pada minyak pelumas	<ol style="list-style-type: none"> 1. Pipa pengeluaran pelumas salah 2. Tekanan minyak pelumas terlalu tinggi 3. Jumlah minyak pelumas mesin dibawah minimum 4. Minyak pelumas bocor
FK03	Masalah pada sparepart mesin	<ol style="list-style-type: none"> 1. Getaran mesin tidak sesuai 2. Mesin mengeluarkan bunyi suara aneh / <i>abnormal</i>

Tabel 3. 3 Kode Gejala Kerusakan Mesin Manufaktur

Kode Gejala	Nama Gejala
FP01	Temperatur Mesin Tinggi
FP02	Pompa air pendingin tidak berputar
FP03	Pompa air pendingin bocor
FP04	Pipa pengeluaran pelumas salah
FP05	Tekanan minyak pelumas terlalu tinggi
FP06	Jumlah minyak pelumas mesin dibawah garis minimal
FP07	Minyak pelumas bocor

FP08	Mesin bergetar tak beraturan
FP09	Mesin mengeluarkan suara abnormal.

Tabel 3. 4 Tabel Keputusan Kerusakan Mesin Manufaktur

Kode Kerusakan	Kode Gejala								
	FP01	FP02	FP03	FP04	FP05	FP06	FP07	FP08	FP09
FK01	Y	Y	Y						
FK02				Y	Y	Y	Y		
FK03								Y	Y

Dari table di atas, maka *rul*nya berupa:

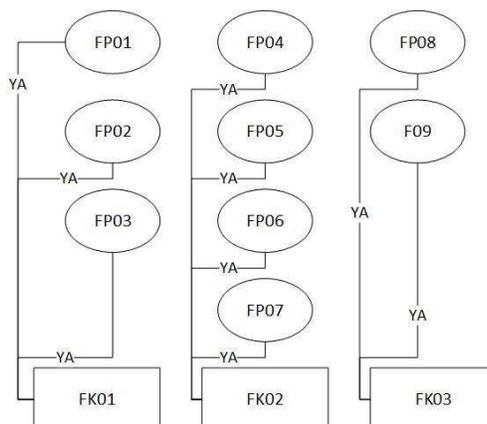
1. IF FK01 THEN FP01 AND FP02 AND FP03
2. IF FK02 THEN FP04 AND FP05 AND FP06 AND FP07
3. IF FK03 THEN FP08 AND FP09

Penjelasan dari kaidah diatas adalah:

1. Jika suhu mesin tinggi dan pompa air pendingin bocor dan pompa air pendingin tidak berputar, maka masalah dengan pompa air pendingin
2. Jika pipa pembuangan pelumas salah dan tekanan oli pelumas terlalu tinggi dan jumlah oli mesin di bawah garis min dan oli pelumas bocor, maka masalah dengan oli pelumas.
3. Jika mesin mengeluarkan suara abnormal dan bergetar tidak teratur, maka masalah dengan suku cadang mesin.

Berdasarkan table keputusan diatas, pohon keputusannya adalah sebagai berikut

3.3. Teknik Pengumpulan Data



Gambar 3. 2 Pohon Keputusan

Pengumpulan data dapat dilakukan dalam berbagai pengaturan, berbagai sumber, dan berbagai cara. Jika dilihat dari pengaturan, data dapat dikumpulkan dalam pengaturan alami, di laboratorium dengan metode eksperimental, di rumah, seminar, kantor dan lainnya. (Sugiyono, 2012: 137)

Jika peneliti ingin melakukan studi pendahuluan untuk menemukan masalah yang harus dipelajari, bisa menggunakan wawancara sebagai teknik pengumpulan data dan juga jika peneliti ingin memperoleh lebih banyak informasi dari responden lain. (Sugiyono, 2012:137)

Gejala dan Solusi diperlukan dalam analisis ini. Gejala dan Solusi yang didapatkan dapat dilakukan analisis persyaratan sistem, yang kemudian digunakan sebagai referensi.

Cara Pengumpulan data dalam penelitian sebagai berikut:

3.3.1. Metode Studi Pustaka

Adalah metode yang mempelajari berbagai buku atau jurnal referensi serta hasil penelitian sebelumnya yang sejenis yang berguna untuk mendapatkan landasan teori mengenai masalah yang akan diteliti

3.3.2. Metode Wawancara

Salah satu metode pengumpulan data melalui wawancara, yaitu memperoleh informasi dengan cara menanyakan langsung ke responden. Wawancara adalah salah satu bagian terpenting dari setiap survei. Tanpa wawancara, peneliti akan kehilangan informasi yang hanya bisa diperoleh dengan cara menanyakan langsung ke responden. Wawancara adalah tulang punggung dari survei penelitian.

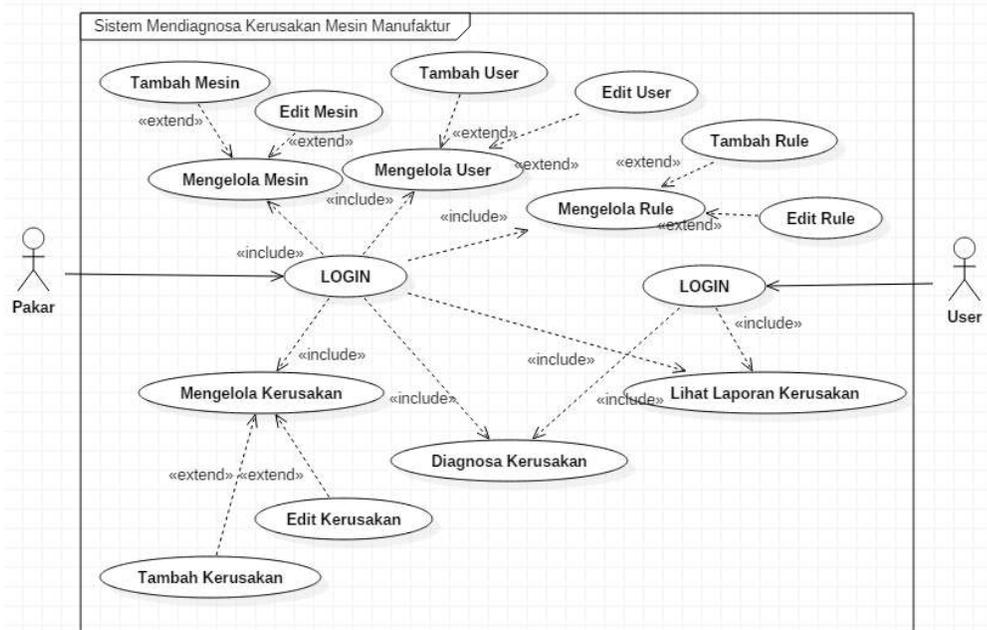
3.4. Perancangan Sistem

3.4.1. UML (*Unified Modeling Language*)

Beberapa diagram dimanfaatkan, berupa:

1. *Use Case Diagram*

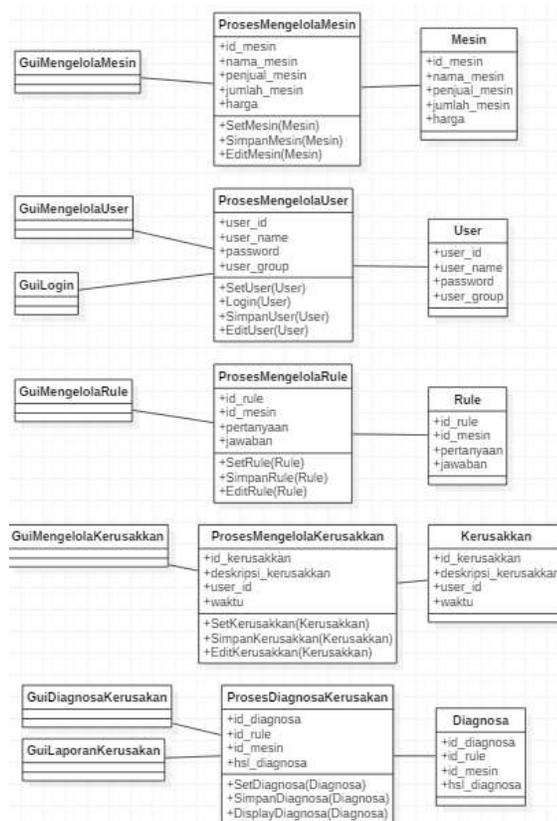
Use Case Diagram bertujuan untuk lebih memahami sistem dan evaluasi bahwa apa yang dilakukan sistem bisa membantu deteksi kerusakan pada mesin manufaktur oleh pengguna. Diagram kasus kerusakan sistem pakar mesin manufaktur dapat dilihat dari gambar dibawah.



Gambar 3.3 Use Case Diagram

2. Class Diagram

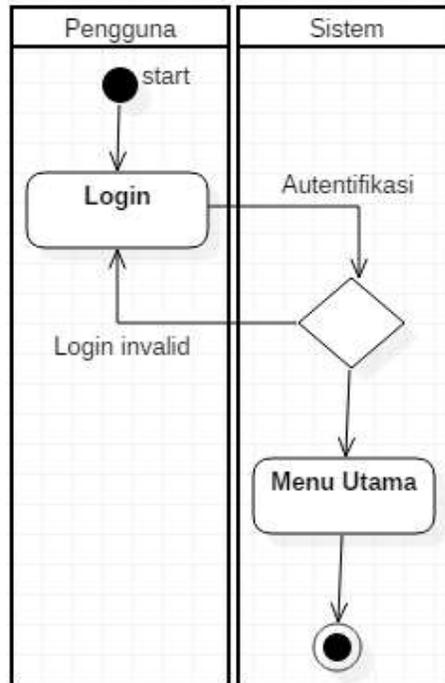
Class diagram adalah model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class. Class diagram mirip ER-Diagram pada perancangan database, bedanya pada ER-diagram tdk terdapat operasi/methode tapi hanya atribut (Rosa & Shalahuddin, 2015). *Class Diagram* sistema Pakar deteksi kerusakan mesin manufaktur seperti pada gambar dibawah.



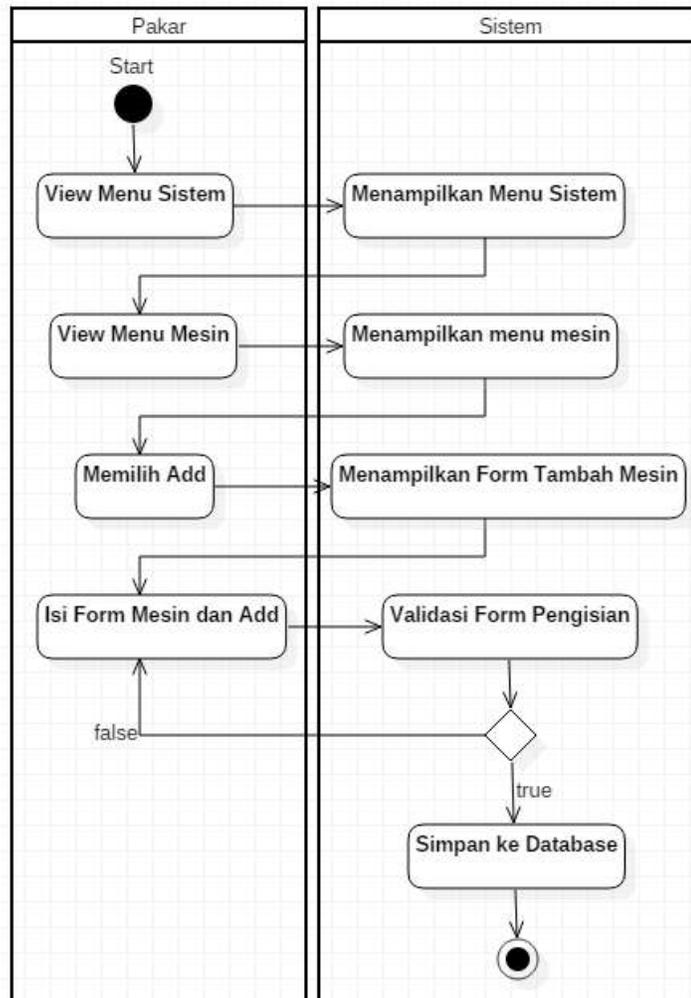
Gambar 3. 4 *Class Diagram*

3. *Activity Diagram*

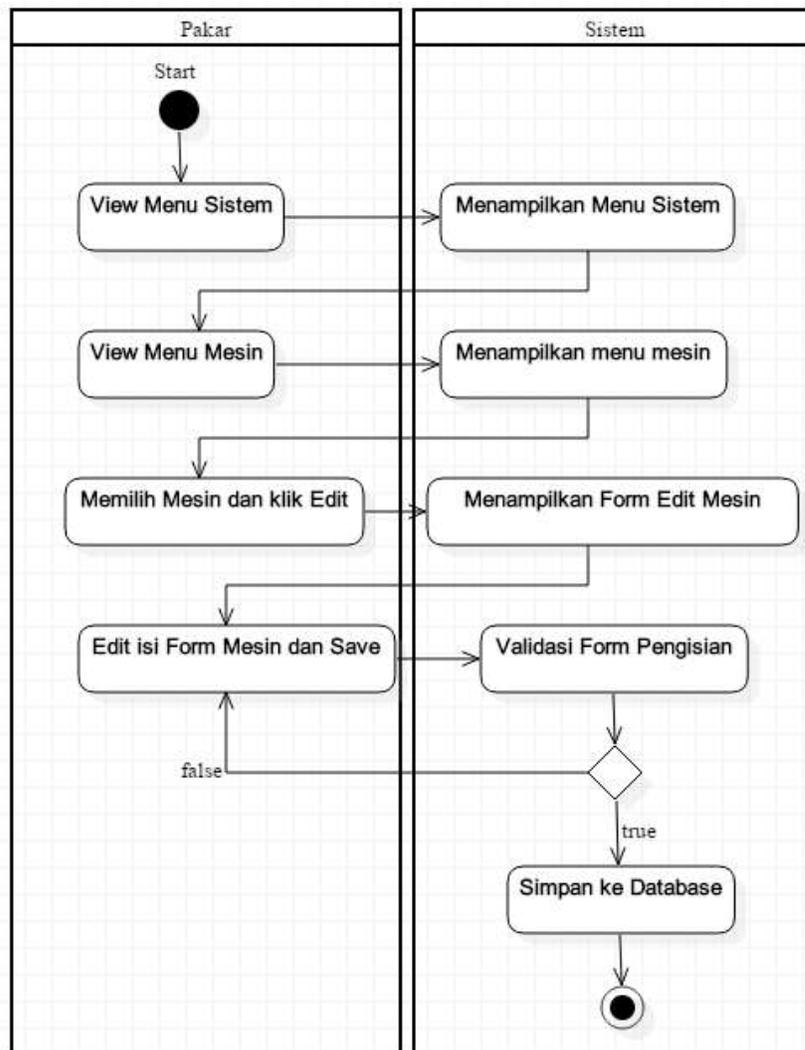
Activity Diagram berfungsi untuk menggambarkan aliran peristiwa dalam kasus penggunaan sistem dengan tujuan membuatnya mudah untuk mengkomunikasikan langkah-langkah ke aliran peristiwa. *Activity Diagram* untuk penelitian ini dapat dilihat sebagai berikut.



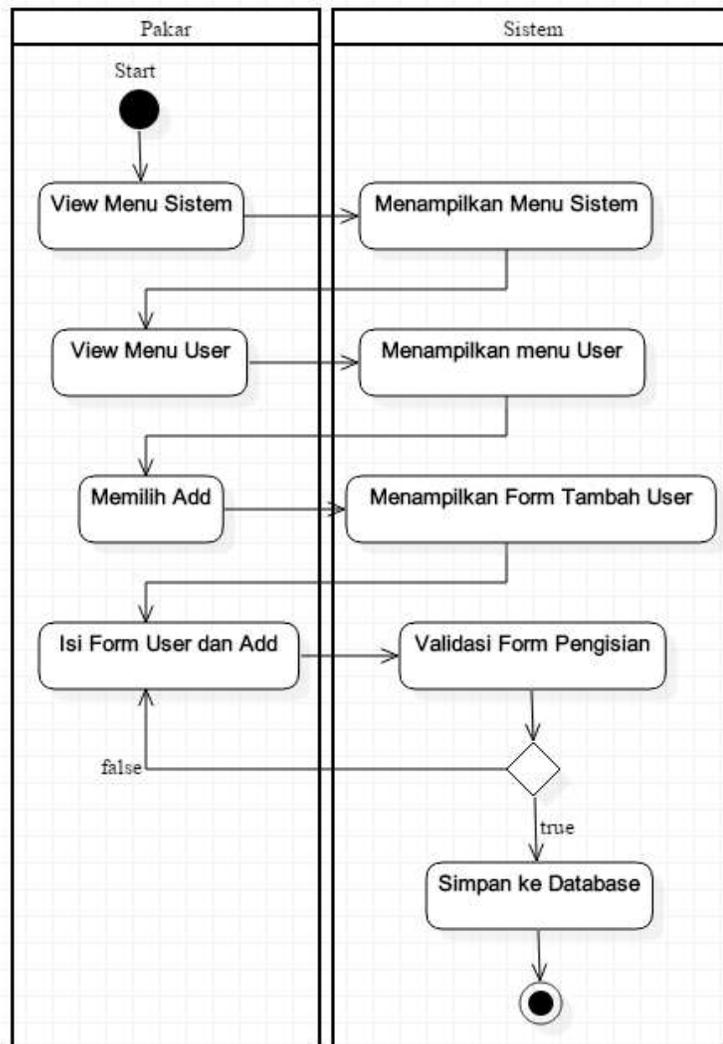
Gambar 3.5 *Activity Diagram Login*



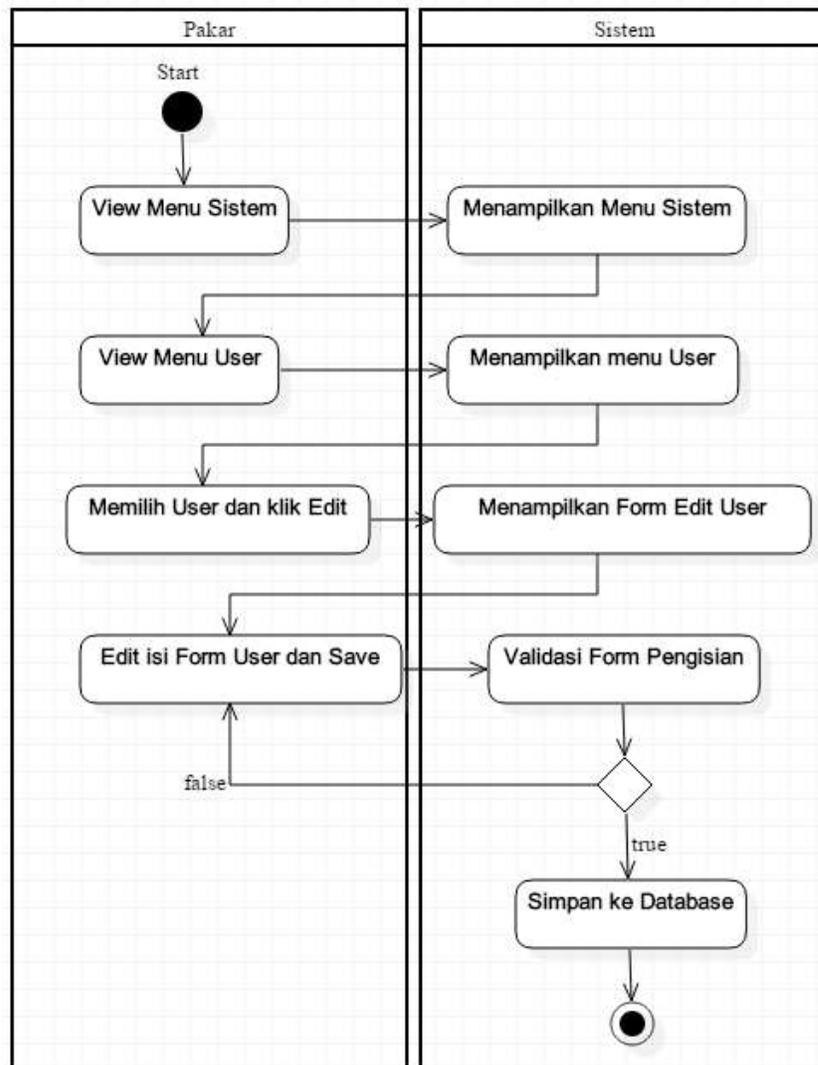
Gambar 3. 6 Activity Diagram Tambah Data Mesin



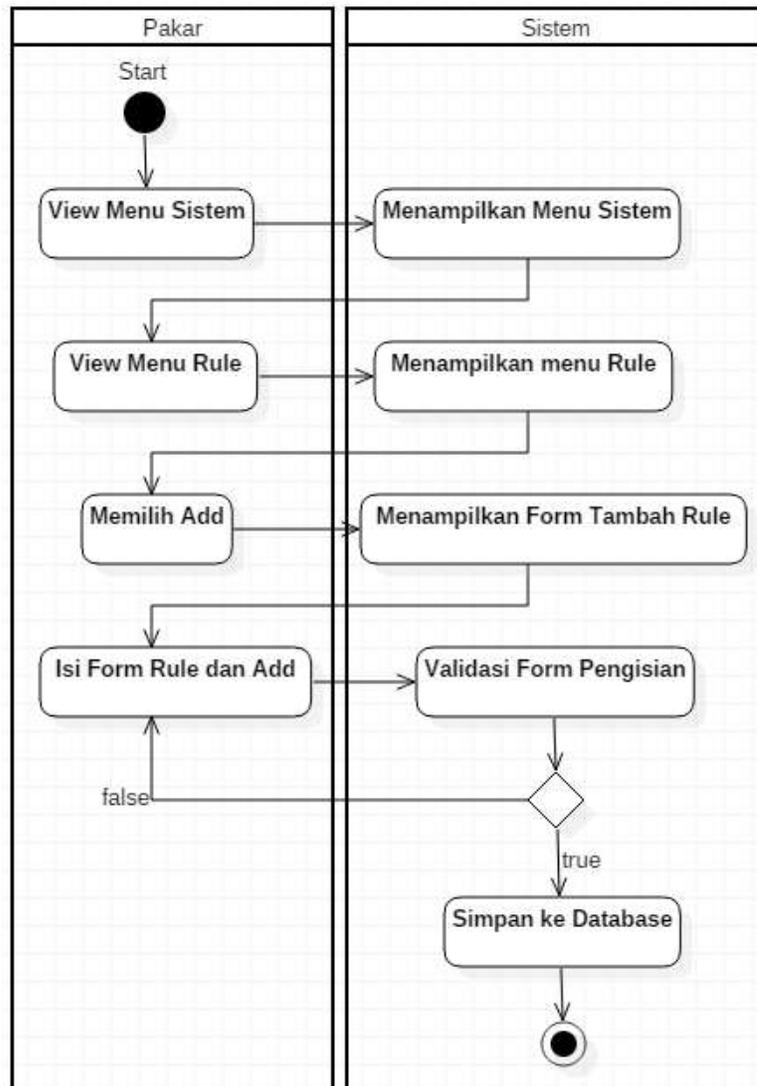
Gambar 3. 7 *Activity Diagram* Ubah Mesin



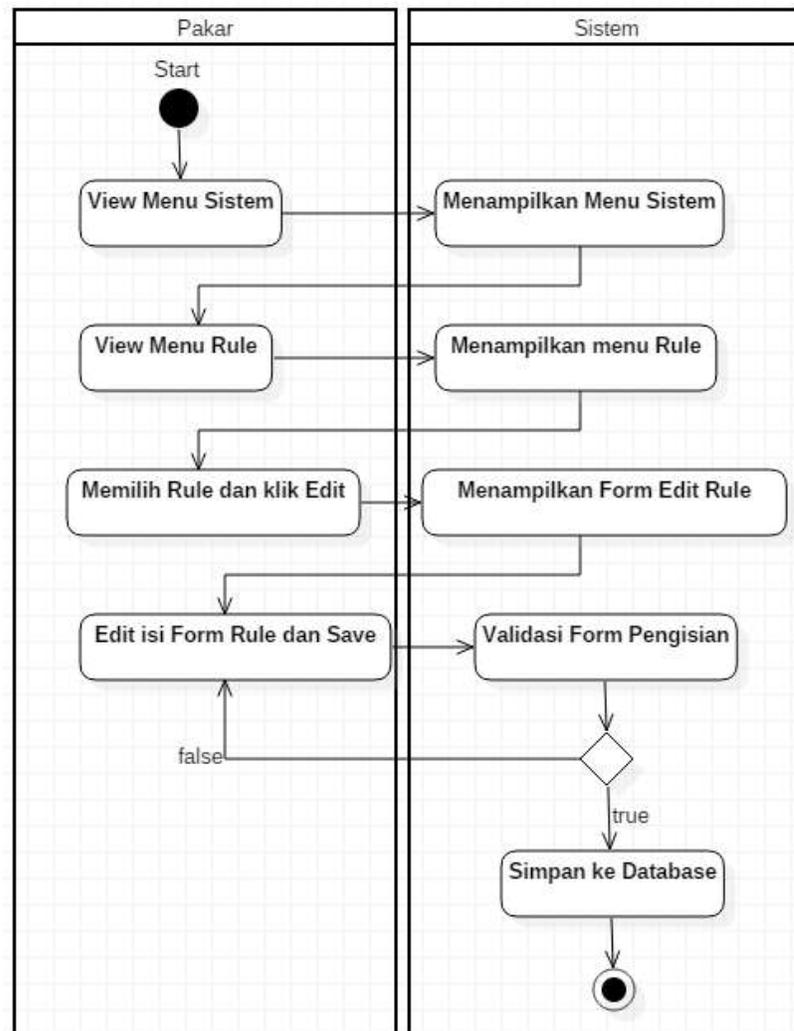
Gambar 3. 8 *Activity Diagram* Tambah User



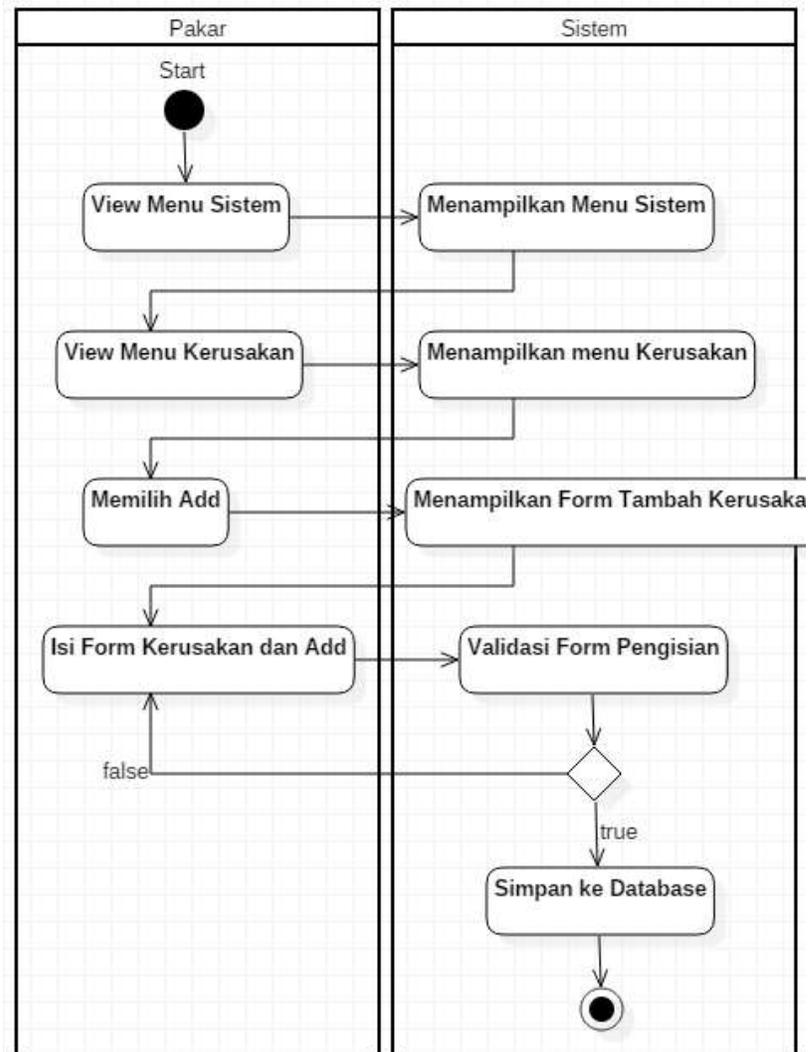
Gambar 3. 9 Activity Diagram Ubah User



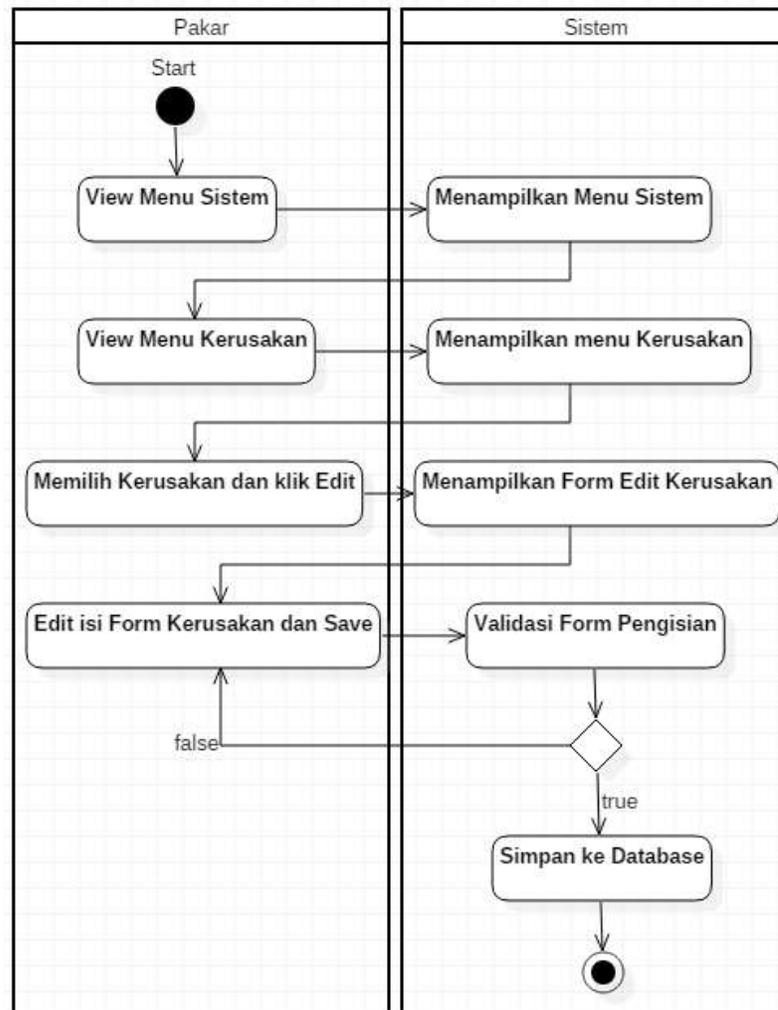
Gambar 3. 10 Activity Diagram Tambah Rule



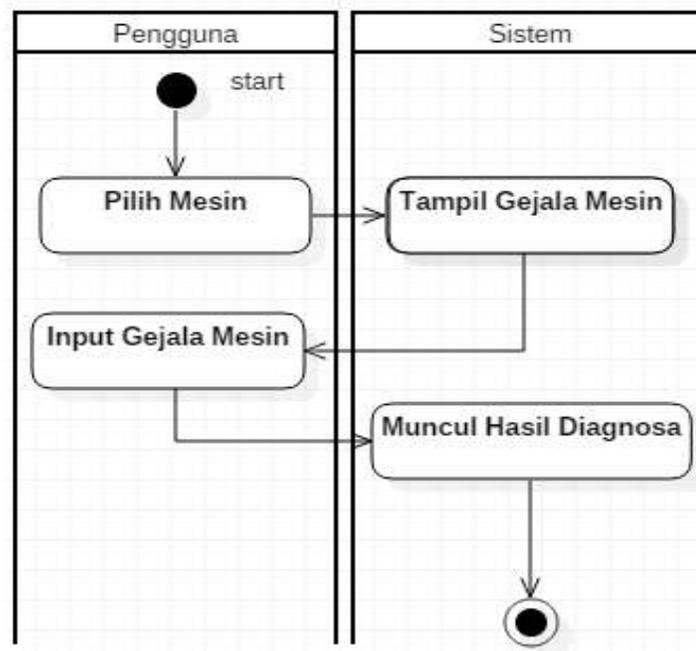
Gambar 3. 11 *Activity Diagram* Ubah Rule



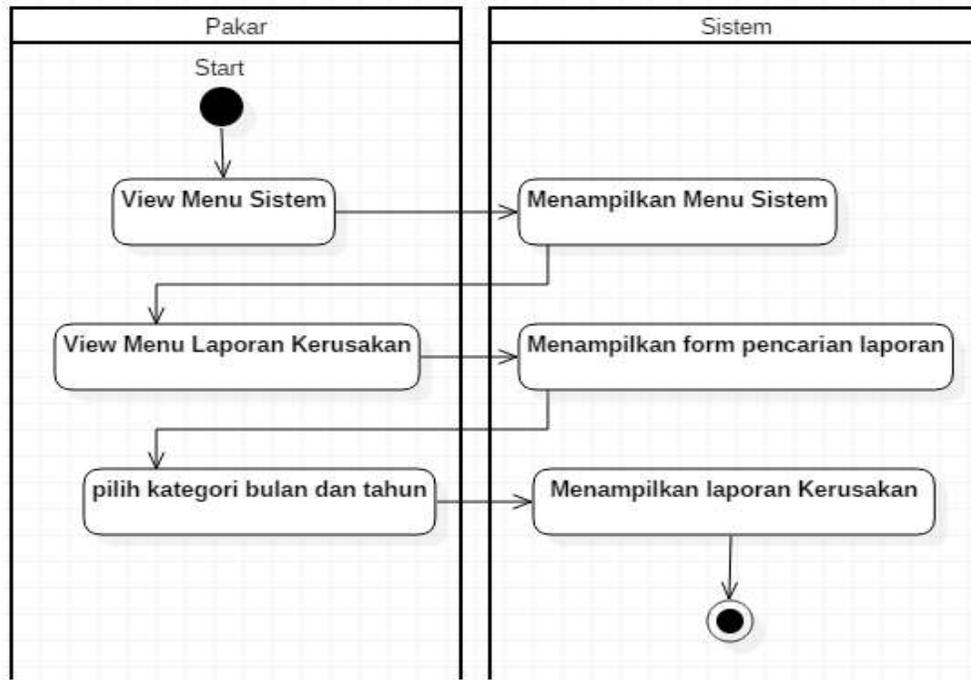
Gambar 3. 12 Activity Diagram Tambah Kerusakan



Gambar 3. 13 Activity Diagram Ubah Kerusakan



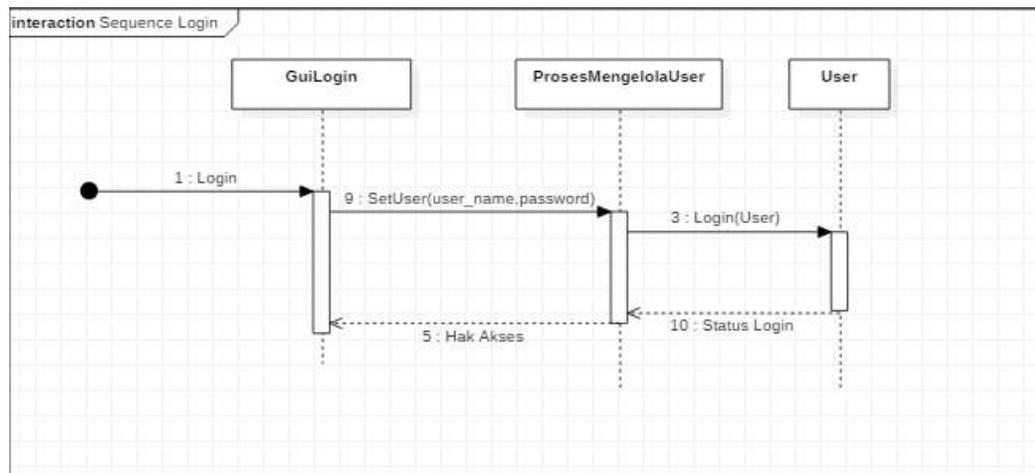
Gambar 3. 14 Activity Diagram Deteksi



Gambar 3. 15 Activity Diagram Lihat Laporan Kerusakan

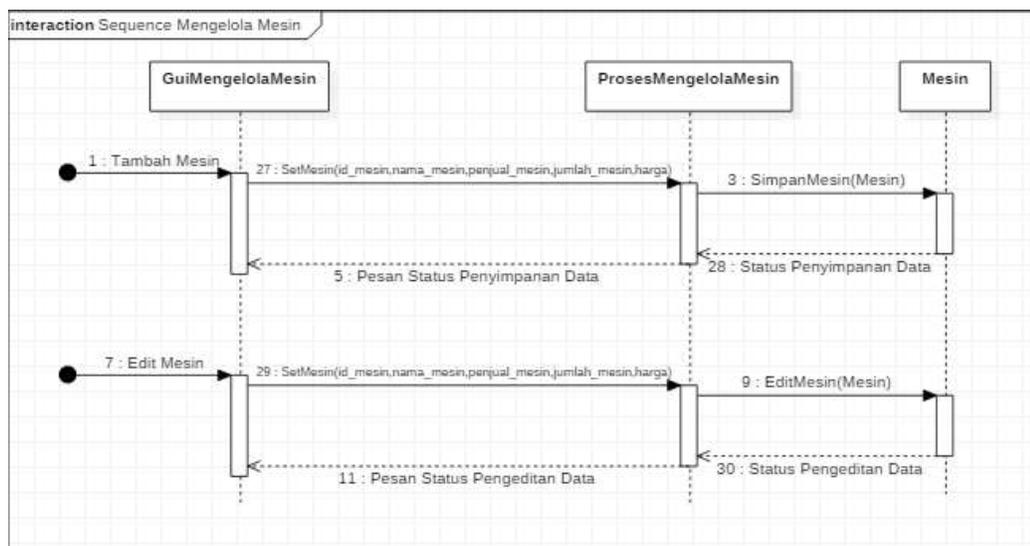
4. Sequence Diagram

a. Sequence Diagram Login



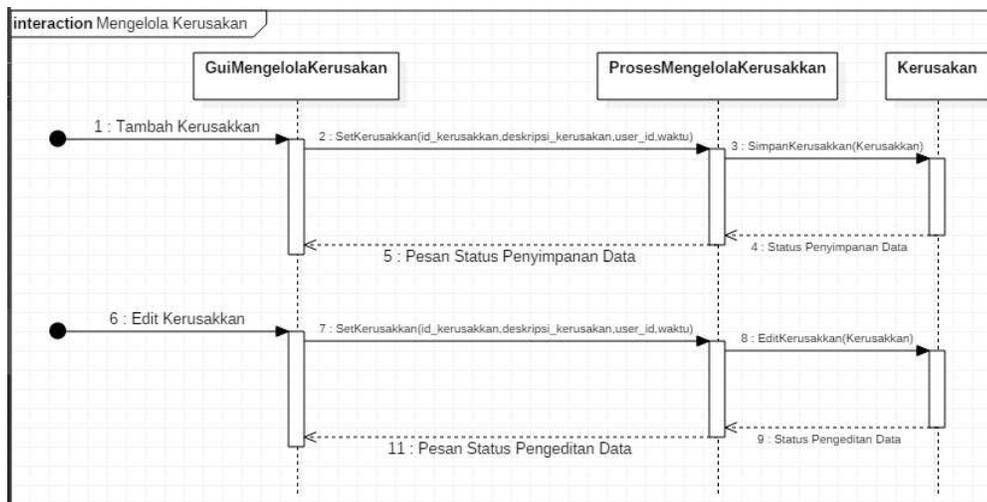
Gambar 3. 16 Sequence Diagram Login

b. Sequence Diagram Mesin



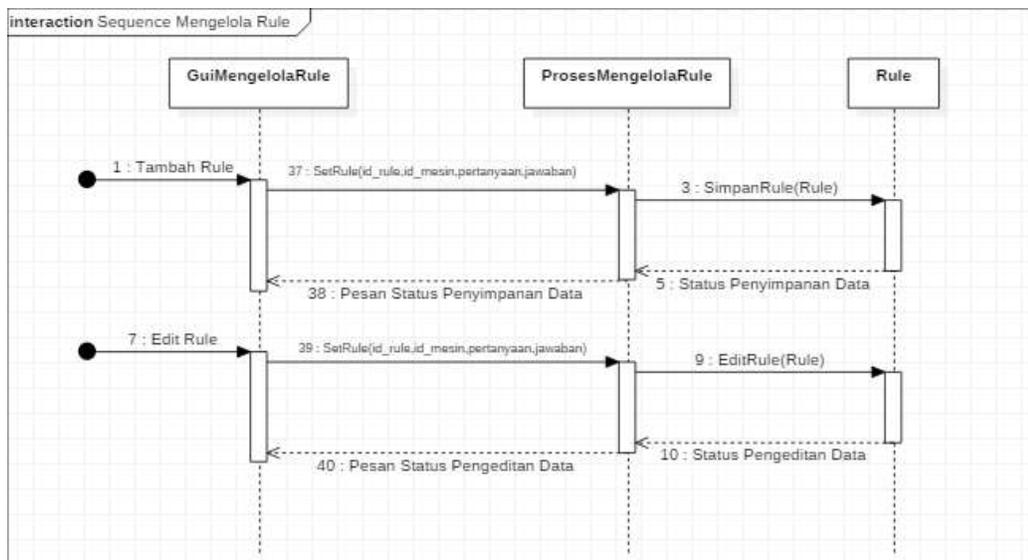
Gambar 3. 17 Sequence Diagram Mesin

c. *Sequence Diagram Kerusakan*



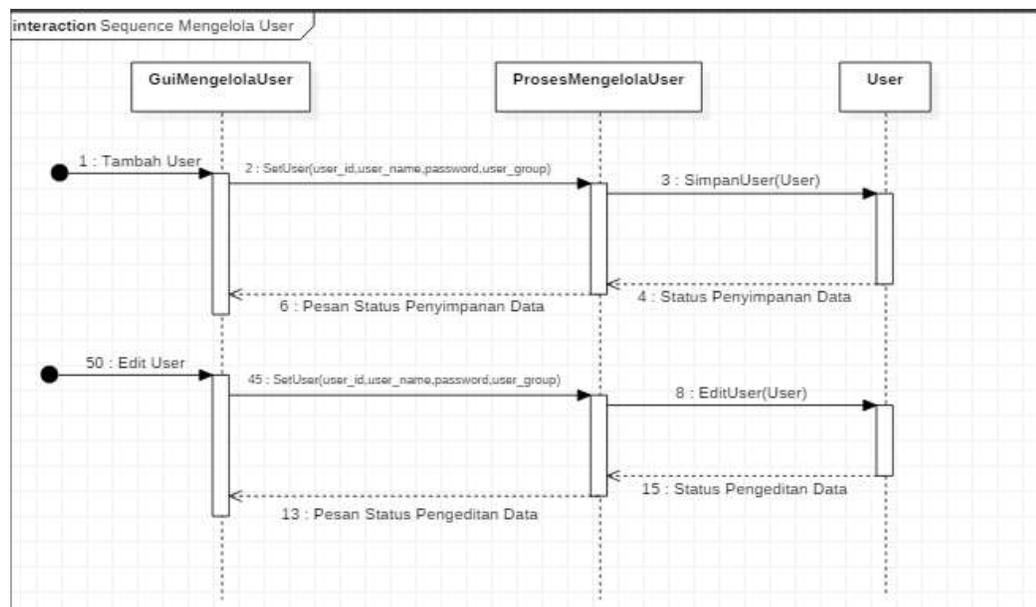
Gambar 3. 18 *Sequence Diagram* Kerusakan

d. *Sequence Diagram Rule*



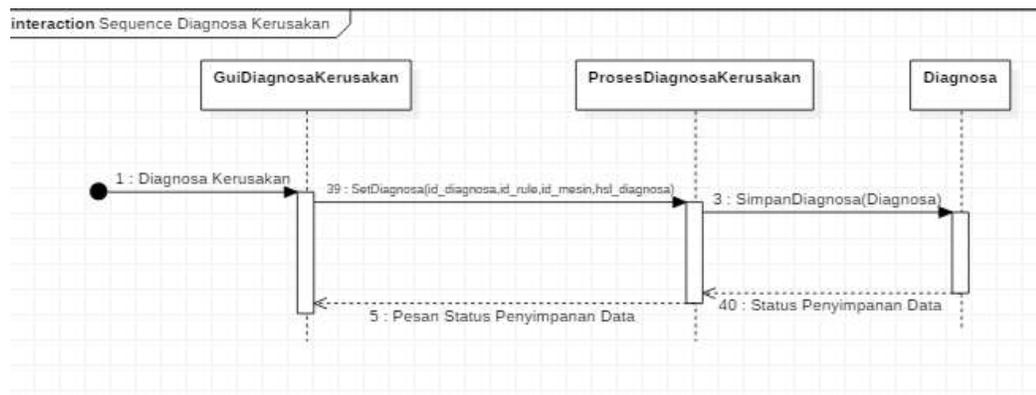
Gambar 3. 19 *Sequence Diagram* Rule

e. *Sequence Diagram User*

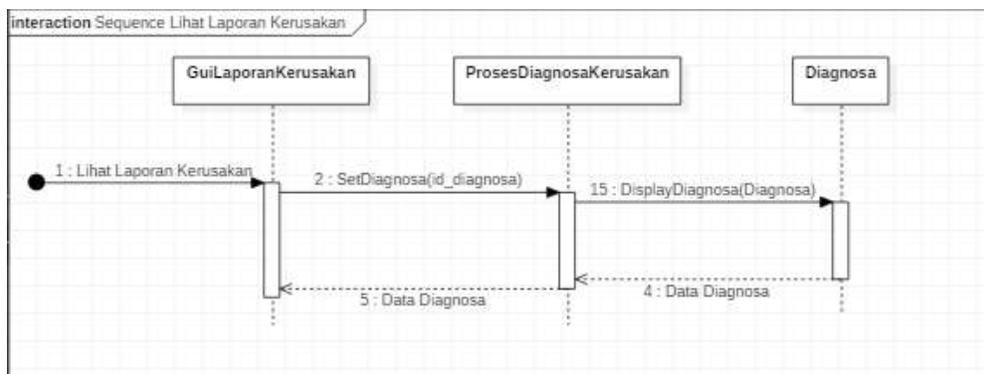


Gambar 3. 20 *Sequence Diagram User*

f. *Sequence Diagram Deteksi Kerusakan*



Gambar 3. 21 *Sequence Diagram Deteksi Kerusakan*

g. *Sequence Diagram* Laporan Riwayat Kerusakkan

Gambar 3. 22 *Sequence Diagram* Laporan Riwayat Kerusakkan

3.4.2. Desain Basis Data

Untuk penyimpanan data dalam penelitian ini, penulis memanfaatkan basis data SQL Server. Untuk detail tabel yang terdapat dalam sistem untuk mendeteksi kerusakan pada mesin pabrikan ini, Anda dapat melihat tabel di bawah ini.

1. Tabel *User*

Tabel 3. 5 Tabel User

TABEL USER	
Kolom	Tipe Data
UserID	nvarchar(20)
Username	nvarchar(50)
Password	nvarchar(20)
GroupID	nvarchar(20)
UpdatedBy	nvarchar(20)
UpdatedDate	datetime

2. Tabel Mesin

Tabel 3. 6 Tabel Mesin

TABEL MESIN	
Kolom	Tipe Data
MacID	nvarchar(20)
MacName	nvarchar(50)
VendorID	nvarchar(20)
PurchaseAmount	Int
Price	Bigint
UpdatedBy	nvarchar(20)
UpdatedOn	datetime

3. Tabel Kerusakan

Tabel 3. 7 Tabel Kerusakan

TABEL KERUSAKAN	
Kolom	Tipe Data
ProblemID	nvarchar(100)
ProblemDesc	nvarchar(100)
UpdatedBy	nvarchar(20)
UpdatedDate	Datetime

4. Tabel Pertanyaan

Tabel 3. 8 Tabel Pertanyaan

DESAIN TABEL PERTANYAAN	
Kolom	Tipe Data
MacID	Varchar(20)

ProblemID	Varchar(100)
KodePertanyaan	Int
Pertanyaan	Varchar(100)
UpdatedBy	Varchar(20)
UpdatedDate	Datetime

5. Tabel Deteksi

Tabel 3. 9 Tabel Deteksi

TABEL DETEKSI	
Kolom	Tipe Data
DeteksiID	nvarchar(20)
MacID	nvarchar(100)
KodePertanyaan	Int
Pertanyaan	nvarchar(100)
Pilihan	nvarchar(5)
Result	nvarchar(20)
CheckedBy	nvarchar(20)
CheckedDate	datetime

3.5. Lokasi dan Jadwal Penelitian

Berikut jadwal penelitian yang dilakukan pada PT. Celindo Pratama.

