

**SISTEM PAKAR BERBASIS WEB MENDETEKSI  
KERUSAKAN SPEAKER MENGGUNAKAN  
METODE *FORWARD CHAINING*  
(STUDI KASUS PT FOSTER)**

**SKRIPSI**



**Oleh:  
Harisyah Putra  
130210163**

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

**SISTEM PAKAR BERBASIS WEB MENDETEKSI  
KERUSAKAN SPEAKER MENGGUNAKAN  
METODE *FORWARD CHAINING*  
(STUDI KASUS PT FOSTER)**

**SKRIPSI**

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



**Oleh:  
Harisyah Putra  
130210163**

**PROGRAM 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.

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, 17 Februari 2017

Harisyah Putra  
130210163

**SISTEM PAKAR BERBASIS WEB MENDETEKSI  
KERUSAKAN SPEAKER MENGGUNAKAN  
METODE *FORWARD CHAINING*  
(STUDI KASUS PT FOSTER)**

**Oleh  
Harisyah Putra  
130210163**

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

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

**Batam, 17 Februari 2017**

**Yusli Yenni, S.kom., M.kom.  
Pembimbing**

## ABSTRAK

Sistem pakar adalah aplikasi berbasis web yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar, yang mencoba memecahkan sebuah masalah yang biasanya hanya dipecahkan oleh seorang pakar. Ada beberapa kerusakan penyebab kerusakan speaker seperti *No sound*, *Sound precision NG*. Penelitian ini bertujuan untuk menganalisis dan merancang sistem pakar. Sistem pakar ini dibangun untuk membantu pengguna untuk mendeteksi kerusakan speaker berdasarkan gejala yang di alami speaker. Sistem pakar ini menggunakan metode *forward chaining* dengan menggunakan bahasa pemrograman PHP dan *database* yang digunakan adalah MySQL dan implementasinya menggunakan web. Melalui sistem pakar ini pengguna dapat melakukan konsultasi dengan sistem layaknya berkonsultasi dengan seorang pakar sesungguhnya untuk mengetahui kerusakan apa yang terjadi pada sebuah speaker tersebut, serta memberikan solusi atas permasalahan apa yang di hadapi.

**Kata kunci : Sistem Pakar, Kerusakan Speaker, *Forward Chaining***

## ABSTRACT

*Expert system is a web-based application that is used to solve the problem, as is thought by experts, who are trying to solve a problem that is usually broken only by an expert. There are several causes of damage such as damage to the speakers No sound, Sound precision NG. This study aims to analyze and design expert system. This expert system developed to help users to detect damage to the speakers based on the symptoms experienced speakers. This expert system uses forward chaining method by using the PHP programming language and MySQL database choose the security and implementation using the web. Through this expert system users can consult with the system like a consultation with a real expert to know what damage occurred on a speaker, as well as provide solutions for problems faced what.*

***Keywords: Expert System, Speaker Damage, Forward Chaining***

## KATA PENGANTAR

Segala puji bagi Allah SWT 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 Putra Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa diterima 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 Yusli Yenni, S.Kom., M.Kom., selaku pembimbing skripsi pada program studi teknik Informatika Universitas Putera Batam.
4. Dosen dan staff Universitas Putera Batam.
5. Bapak Oman Al Imron selaku nara sumber yang rela meluangkan banyak waktunya untuk mendukung penelitian ini.
6. Bapak Gunawan selaku Manajer *HRD* PT FOSTER yang telah memberikan dukungannya.
7. Orang tua dan keluarga yang selalu memberikan doa dan motivasi yang baik.

8. Teman-teman mahasiswa/i Universitas Putera Batam yang turut memberikan doa dan dukungannya.

9. Serta pihak-pihak lain yang tidak dapat disebutkan satu persatu.

Semoga Allah SWT membalas kebaikan dan selalu mencurahkan taufik dan hidayahNya, Amin.

Batam, Februari 2017

Penulis



# DAFTAR ISI

PERNYATAAN.....	i
HALAMAN PENGESAHAN.....	ii
ABSTRAK.....	iii
<i>ABSTRACT</i> .....	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	x
DAFTAR LAMPIRAN.....	xi
<b>BAB IPENDAHULUAN</b>	
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah.....	4
1.3 Pembatasan Masalah.....	4
1.4 Perumusan Masalah.....	5
1.5 Tujuan Peneliti.....	5
1.6 Manfaat Peneliti.....	6
1.6.1 Aspek Praktis.....	6
1.6.2 Aspek Teoritis.....	7
<b>BAB IITINJAUAN PUSTAKA</b>	
2.1 Teori Dasar.....	8
2.1.1 Pengertian Artificial Intelligence.....	8
2.1.1.1 Logika <i>Fuzzy</i> .....	9
2.1.1.2 Jaringan saraf tiruan ( <i>Artificial neural network</i> ).....	10
2.1.2 Sistem pakar.....	11
2.1.3 Bentuk Sistem Pakar.....	12
2.1.4 Struktur Sistem Pakar.....	13
2.1.5 Manfaat Sistem Pakar.....	14
2.1.6 Komponen Sistem Pakar.....	14
2.1.7 Runut Maju ( <i>Forward Chaining</i> ).....	16
2.2 Variabel Penelitian.....	20
2.2.1 Speaker.....	21
2.2.2 Kerusakan Speaker.....	27
2.3 <i>Software</i> Pendukung.....	28
2.3.1 Unified Modeling Language (UML).....	28
2.3.1.1 Simbol-simbol Pada <i>Unified Modeling Language</i> (UML).....	29
2.3.3 Hypertext Preprocessor (PHP).....	36
2.3.3.1 Sejarah PHP.....	37
2.3.3.2 Keunggulan PHP.....	38
2.3.4 Adobe Dreamweaver.....	39

2.3.5 XAMPP .....	40
2.3.6 Domain name/Url-Uniform Resource Lacator.....	41
2.3.7 MySQL.....	42
2.4 Penelitian Terdahulu .....	43
2.5 Kerangka Pemikiran.....	45

### **BAB IIIMETODE PENELITIAN**

3.1 Desain Penelitian.....	47
3.2 Teknik Pengumpulan Data .....	50
3.3 Operasional Variabel.....	51
3.4 Metode Perancangan Sistem .....	52
3.4.1 Desain Basis Pengetahuan .....	53
3.4.2 Struktur kontrol (mesin inferensi).....	58
3.4.3 Desain UML (Unifield Modeling Languange) .....	58
3.4.4 Desain Antarmuka ( <i>prototype</i> ).....	65
3.5 Lokasi dan Jadwal Penelitian .....	69
3.5.1 Jadwal Penelitian.....	70

### **BAB IVHASIL DAN PEMBAHASAN**

4.1 Hasil Penelitian .....	69
4.2 Pembahasan.....	77
4.2.1 Pengujian validasi sistem .....	77

### **BAB VSIMPULAN DAN SARAN**

5.1 Simpulan .....	81
5.2 Saran.....	82

DAFTAR PUSTAKA

DAFTAR RIWAYAT HIDUP

SURAT KETERANGAN PENELITIAN

LAMPIRAN

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Struktur sistem pakar .....	13
<b>Gambar 2.3</b> Speaker .....	21
<b>Gambar 2.4</b> Frame .....	22
<b>Gambar 2.5</b> Spider.....	22
<b>Gambar 2.6</b> Magnet .....	23
<b>Gambar 2.7</b> Voice Coil.....	23
<b>Gambar 2.8</b> Dust cap .....	24
<b>Gambar 2.9</b> Diaphragm .....	24
<b>Gambar 2.10</b> Terminal.....	25
<b>Gambar 2.11</b> Suspension .....	25
<b>Gambar 2.12</b> Yoke.....	26
<b>Gambar 2.13</b> Plate .....	26
<b>Gambar 2.14</b> Uml (Unified Modeling Language).....	28
<b>Gambar 2.15</b> PHP .....	36
<b>Gambar 2.16</b> Adobe Dreamweaver .....	39
<b>Gambar 2.17</b> Xampp .....	40
<b>Gambar 2.18</b> Domain name/URL.....	41
<b>Gambar 2.19</b> MySQL .....	42
<b>Gambar 2.20</b> Kerangka pemikiran.....	46
<b>Gambar 3.1</b> Desain Penelitian .....	48
<b>Gambar 3.2</b> Pohon Keputusan .....	57
<b>Gambar 3.3</b> Diagram Use Case Admin dan User .....	59
<b>Gambar 3.4</b> Activity diagram admin .....	60
<b>Gambar 3.5</b> Activity diagram user .....	61
<b>Gambar 3.6</b> Class diagram.....	62
<b>Gambar 3.7</b> Sequence Diagram Admin.....	63
<b>Gambar 3.8</b> Sequence Diagram User .....	63
<b>Gambar 3.9</b> Flowchart.....	64
<b>Gambar 3.10</b> Rancangan halaman home .....	65
<b>Gambar 3.11</b> Rancangan halaman konsultasi sitem pakar .....	66
<b>Gambar 3.12</b> Rancangan halaman kritik & saran.....	67
<b>Gambar 3.13</b> Rancangan halaman CRUD admin.....	68
<b>Gambar 3.14</b> Rancangan halaman kritik & saran.....	69
<b>Gambar 3.15</b> Jadwal penelitian .....	70
<b>Gambar 4.1</b> Tampilan Halaman utama.....	69
<b>Gambar 4.2</b> Tampilan Halaman Login Admin.....	70
<b>Gambar 4.3</b> Tampilan Halaman Admin .....	71
<b>Gambar 4.4</b> Tampilan Halaman Komentar Pengguna.....	72
<b>Gambar 4.5</b> Tampilan Halaman Input Data .....	73
<b>Gambar 4.6</b> Tampilan Halaman Sistem Pakar .....	74
<b>Gambar 4.7</b> Tampilan Halaman Konsultasi .....	75
<b>Gambar 4.8</b> Tampilan Halaman Hasil Konsultasi.....	76
<b>Gambar 4.9</b> Tampilan Halaman Kritik dan Saran .....	77

## DAFTAR TABEL

<b>Tabel 2.1</b> Tabel Keputusan.....	18
<b>Tabel 2.2</b> Simbol pada diagram <i>use case</i> .....	30
<b>Tabel 2.3</b> Simbol pada <i>activity</i> diagram .....	32
<b>Tabel 2.4</b> Simbol pada <i>sequence</i> diagram .....	33
<b>Tabel 3.1</b> variabel dan indikator.....	52
<b>Tabel3.2</b> Indikator .....	53
<b>Tabel 3.3</b> Tabel Penyebab .....	53
<b>Tabel 3.4</b> Tabel Gejala .....	54
<b>Tabel 3.5</b> Tabel Aturan.....	55
<b>Tabel 3.6</b> Tabel Keputusan.....	56
<b>Tabel 4.1</b> Tabel pengujian menu <i>Home</i> .....	78
<b>Tabel 4.2</b> Tabel pengujian halaman <i>login</i> .....	78
<b>Tabel 4.3</b> Tabel pengujian halaman admin .....	78
<b>Tabel 4.4</b> Tabel pengujian halaman sistem pakar .....	79
<b>Tabel 4.5</b> Tabel pengujian halaman konsultasi .....	79
<b>Tabel 4.6</b> Tabel pengujian halaman hasil konsultasi.....	79
<b>Tabel 4.7</b> Tabel pengujian halaman hasil konsultasi.....	80
<b>Tabel 4.8</b> Tabel pengujian halaman komentrs prngguna.....	80
<b>Tabel 4.9</b> Tabel pengujian halaman input data.....	80

## **DAFTAR LAMPIRAN**

LAMPIRAN I FORM WAWANCARA  
LAMPIRAN II FOTO WAWANCARA  
LAMPIRAN III KODING PROGRAM

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Seiring perkembangan teknologi yang pesat, Hampir semua orang memiliki speaker sebagai alat bunyi suara, Sehingga speaker merupakan hal yang tak lazim lagi bagi masyarakat, dan masyarakat juga sudah banyak yang memiliki speaker di rumah, karena efisiensi dan waktu ini sangat penting dalam melengkapi suara barang elektronik miliknya. Namun tidak sedikit speaker mengalami masalah, baik dari masalah ringan maupun berat, dan untuk perbaikan tersebut memerlukan biaya yang cukup mahal. Masyarakat awam pada saat ini masih minim pengetahuan dan kesulitan jika menghadapi masalah kerusakan speaker.

Salah satu komponen dari peralatan audio adalah pengeras suara atau speaker. Saat ini speaker merupakan salah satu alat yang sering digunakan dalam kehidupan sehari-hari. Kita dapat menikmatinya ketika mendengarkan ceramah di masjid, bersantai di rumah mendengarkan musik pada saat berkendara, *set home theater* ataupun ketika memasak sambil mendengarkan musik favorit. Setiap para pencinta audio sudah pasti akan lebih menyukai untuk menikmati musik yang sedapat mungkin sesuai dengan aslinya tanpa ada *IRC iron chip*. *Iron chip* memang tidaklah berbahaya akan tetapi mengganggu pendengaran seorang pendengar. Dari pengeras suara biasanya terdengar suara-suara yang tidak nyaman didengar seperti suara gemerisik, suara yang pecah, dengungan dan

sebagainya. Padahal pengeras suara yang baik adalah yang dapat menghasilkan kualitas suara seperti aslinya tanpa ada gangguan sehingga nyaman untuk didengarkan oleh telinga si pendengar.

Bagi para pemilik speaker tentunya menghendaki speaker dengan kualitas suara yang bagus, jernih dan tidak mudah rusak. Pada *amplifier* saat listik pertama kali dinyalakan terjadi perubahan tegangan *output* yang diakibatkan kesalahan pada rangkain *output amplifier* atau tegangan. Hal ini menyebabkan timbulnya derau letup pada speaker bila ini dibiarkan terus menerus maka dapat menimbulkan kerusakan pada speaker terutama bila tegangan catu daya yang digunakan cukup besar sedangkan speaker mempunyai kemampuan daya yang terbatas. Sebagai gambaran untuk amplifier yang menggunakan catu daya sebesar  $\pm 55V_{dc}$  untuk beban speaker  $8 \Omega$  akan dilepaskan daya sebesar 200 watt (daya arus searah) pada speaker tersebut. Untuk menghindari terjadinya hal-hal diatas, salah satu solusinya adalah menggunakan rangkain pelindung speaker. Penggunaan rangkaian ini dapat mereduksi derau letup pada saat menghidupkan dan mematikan amplifier serta peka terhadap tegangan searah yang harus dihindari oleh speaker.

PT FOSTER sebuah perusahaan yang bergerak dibidang elektronik, membuat beragam *customemade* (suara). Memulai kecerdikan teknologi dan akumulasi pengetahuan, memenuhi kebutuhan audio luar dari klien. Yang memproduksi loudspeaker, *Speaker system*.

Loudspeaker adalah *transduser* yang mengubah sinyal listrik menjadi sinyal suara. Ketika loudspeaker mengubah sinyal listrik ke sinyal suara, ada derau (*noise*) yang mengganggu sinyal suara yang dihasilkan loudspeaker, Sehingga suara yang terdengar dari loudspeaker tidak jernih (kualitas suara berkurang). Pola derau yang muncul pada sinyal loudspeaker selalu berubah-ubah sepanjang waktu.

Salah satu yang dibutuhkan dalam permasalahan mendeteksi kerusakan speaker ini, kurangnya pengetahuan dari masyarakat umum dan teknisi dalam penanganan metode *forward chaining* dengan harapan dapat mempermudah dalam membantu teknisi atau orang biasa dalam mendeteksi kerusakan speaker. Masalah-masalah yang ditimbulkan kerusakan speaker yang sebenarnya lebih mudah untuk diperbaiki dengan membangun aplikasi dengan memanfaatkan teknologi berbasis *web* yang sedang maraknya berkembang sekarang ini. Pada penelitian ini akan menerapkan oleh speaker kadangkala merupakan masalah kecil yang tidak memerlukan tingkat pengetahuan yang tinggi. Untuk menyelesaikan hal itu, mungkin bisa diselesaikan oleh seorang yang mempunyai pengetahuan sangat dasar tentang speaker. Tetapi terkadang masalah-masalah tersebut juga membutuhkan tingkat kemampuan yang tinggi akan komponen-komponennya sehingga memerlukan seorang teknisi khusus atau pakar untuk memperbaikinya.



Pakar diharapkan bisa mempercepat dalam mendeteksi suatu jenis kerusakan pada speaker, sehingga dapat dengan mudah mengetahui jenis kerusakan yang terjadi, dari uraian diatas penulis tertarik untuk mengangkat judul “ SISTEM PAKAR BERBASIS WEB MENDETEKSI KERUSAKAN SPEAKER MENGGUNAKAN METODE FORWARD CHAINING (STUDI KASUS PT FOSTER)”

## **1.2 Identifikasi Masalah**

Bedasarkan latar belakang yang telah diuraikan maka peneliti mengidentifikasi masalah sebagai berikut :

1. Kurangnya pengetahuan masyarakat dalam mengetahui kerusakan-kerusakan yang terjadi pada speaker.
2. Susahnya mencari teknisi untuk menyampaikan informasi tentang kerusakan speaker.
3. Banyaknya masalah kerusakan yang kecil, mengakibatkan malasnya masyarakat untuk memperbaiki.

## **1.3 Pembatasan Masalah**

Pembatasan masalah yang akan dibahas dalam penelitian ini adalah sebagai berikut :

1. Jenis speaker yang di teliti berupa speaker tv, mobil dan *home teater*

2. Penelitian ini memakai sistem pakar dengan metode *forward chaining*.
3. Aplikasi pembuatan *web* menggunakan Bahasa pemrograman *PHP* dan database *MySQL*.

#### **1.4 Perumusan Masalah**

Adapun rumusan dari penelitian ini adalah:

1. Bagaimana menerapkan sistem pakar metode *forward chaining* pada kerusakan speaker?
2. Bagaimana implementasi sistem pakar berbasis *web* yang bermanfaat dan berguna bagi operator dan masyarakat?

#### **1.5 Tujuan Penelitian**

Adapun tujuan penelitian ini adalah :

1. Dengan pencegahan lebih awal dari masyarakat agar tidak terjadinya rusaknya speaker.
2. Dengan adanya aplikasi sistem pakar berbasis *web* bermanfaat dan berguna bagi operator dan masyarakat.

## **1.6 Manfaat Peneliti**

Manfaat dari penelitian ini adalah sebagai tambahan materi dan referensi dalam pengembangan ilmu dan pengetahuan teknologi khususnya pada sistem pakar berbasis *web* mendeteksi kerusakan speaker. Adapun manfaat peneliti ini dapat dirasakan dari berbagai segi yang berbeda :

### **1.6.1 Aspek Praktis**

Ada beberapa penerapan untuk aspek praktis untuk mempermudah pemahaman tentang kerusakan speaker :

1. Secara praktis penelitian ini diharapkan masyarakat dapat mendeteksi kerusakan speaker secara langsung tanpa harus bertemu dengan pakar atau teknisi
2. Diharapkan mampu memberikan tambahan informasi bagi masyarakat umum dalam mengambil tindakan perbaikan untuk penanganan kerusakan speaker mereka.
3. Dapat berguna untuk masyarakat umum agar bisa memberikan penanganan terlebih dahulu sebelum terjadi kerusakan pada speaker.

### **1.6.2 Aspek Teoritis**

Beberapa aspek teoritis yang bisa menjadi bahan masukan bagi semua orang yang memerlukan :

1. Secara teoritis diharapkan penelitian ini dapat meningkatkan dan mengembangkan ilmu tentang sistem pakar
2. Serta menjadi tambahan dan masukan teori disamping teori-teori yang sudah ada sebelumnya.
3. Sebagai masukan kedepannya kepada masyarakat umum yang belum paham tentang speaker yang mereka miliki.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teori Dasar**

Deskripsi teori paling tidak berisi tentang penjelasan terhadap variabel-variabel yang diteliti melalui pendefinisian, dan uraian yang lengkap dan mendalam dari berbagai referensi, sehingga ruang lingkup, kedudukan dan prediksi terhadap hubungan antara variabel yang akan diteliti menjadi lebih jelas dan terarah (Sugiyono, 2012:58)

##### **2.1.1 Pengertian *Artificial Intelligence***

Kecerdasan buatan berasal dari bahasa Inggris "*Artificial Intelligence*" atau disingkat AI, yaitu *Intelligence* adalah kata sifat yang berarti cerdas, sedangkan *Artificial* artinya buatan. Kecerdasan yang dimaksud disini merujuk kepada mesin yang mampu berfikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia.

Sebagian kalangan menerjemahkan *Artificial Intelligence* sebagai kecerdasan buatan, kecerdasan artifisial, inteligensia artifisial, atau intelegensi buatan. Pada buku ini, istilah *Artificial Intelligence* sengaja tidak diterjemahkan ke bahasa Indonesia karena istilah tersebut sudah sangat akrab bagi orang Indonesia. Begitu

juga dengan singkatan istilah tersebut, yaitu AI, sudah sangat melekat di berbagai media ilmiah maupun non-ilmiah. (Suyanto, 2014:3)

### 2.1.1.1 Logika *Fuzzy*

Dua buah *logic* yang kita bahas di atas adalah untuk masalah-masalah yang pasti. Bagaimana merepresentasikan masalah yang mengandung ketidakpastian ke dalam suatu bahasa formal yang dipahami komputer. Untuk masalah seperti ini, kita bisa menggunakan *fuzzy logic*. Beberapa kalangan menerjemahkan istilah ini sebagai logika samar. Teori tentang *fuzzy logic set* atau himpunan samar pertama kali dikemukakan oleh Lotfi Zadeh sekitar tahun 1965 pada sebuah makalah yang berjudul *fuzzy set* setelah itu, sejak pertengahan 1970-an, para peneliti Jepang berhasil mengaplikasikan teori ini ke dalam berbagai permasalahan praktis. (Suyanto, 2014:99)

#### 1. Metode *Tsukamoto*

Pada metode *Tsukamoto*, setiap konsekuensi pada aturan yang berbentuk *IF-Then* harus direpresentasikan dengan suatu himpunan *fuzzy* dengan fungsi keanggotaan yang monoton, sebagai hasilnya, *output* hasil inferensi dari tiap-tiap aturan diberikan secara tegas (*crisp*) berdasarkan  $\alpha$ -predikat (*fire strength*). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot.

## 2. Metode *Mamdani*

Metode ini sering juga dikenal dengan nama metode *Max-Min*. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan *output*, diperlukan 4 tahapan :

1. Pembentukan himpunan *fuzzy*
2. Aplikasi fungsi implikasi (aturan)
3. Komposisi aturan
4. Penegasan (*defuzzy*)

## 3. Metode *Sugeno*

Penalaran dengan metode *sugeno* hampir sama dengan penalaran *mamdani*, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau permasalahan *linier*. Metode ini diperkenalkan oleh *Takagi-sugeno kang* pada tahun 1985.

### **2.1.1.2 Jaringan saraf tiruan (*Artificial neural network*)**

Jaringan Saraf Tiruan (JST) adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf tiruan ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran. Jst adalah proses tersebar paralel (*paralel distributed processor*) yang sangat besar yang memiliki kecenderungan untuk menyimpan pengetahuan

yang bersifat pengalaman dan membuatnya siap digunakan. Jst menyerupai otak manusia dalam dua hal, yaitu: pengetahuan diperoleh jaringan melalui proses belajar; kekuatan hubungan antar sel syaraf (*neuron*) yang dikenal sebagai bobot-bobot sinaptik digunakan untuk menyimpan pengetahuan.

JST mempunyai struktur tersebar parallel yang sangat besar dan mempunyai kemampuan belajar, sehingga bisa melakukan *generalization* atau diterjemahkan sebagai generalisasi, yaitu bisa menghasilkan *output* yang benar untuk *input* yang belum pernah dilatihkan. Dengan kemampuan kedua pemrosesan informasi ini, Jst mampu menyelesaikan masalah-masalah yang sangat kompleks. (Suyanto, 2014:170)

### **2.1.2 Sistem pakar**

Sri Hartati dan Sari Ismawanti (2008:1) Sistem pakar (*Expert System*) adalah sistem yang berusaha mengadopsi pengetahuan manusia kekomputer, agar komputer dapat menyelesaikan masalah yang seperti dilakukan manusia.

Sri Kusumadewi (2007:109) Sistem pakar adalah suatu sistem yang bisa menyamai atau meniru kemampuan seorang pakar.

Pengertian lain menurut Sri Kusumadewi (2007:109) Sistem pakar adalah modal prosedur yang berkaitan dalam suatu domain tertentu, yang mana tingkat keahlian dapat dibandingkan dengan keahlian seorang pakar. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja para ahli. Dengan sistem pakar, orang awampun dapat menyelesaikan masalah yang cukup rumit, yang sebenarnya hanya dapat



diselesaikan dengan buatan oleh para ahli, sistem pakar ini juga akan membantu aktifitasnya sebagai asisten yang sangat berpengalaman.

Sistem pakar adalah suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna. Dengan bantuan sistem pakar seseorang yang bukan pakar/ahli dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar. (Sutojo.T, dkk, 2011:13)

### **2.1.3 Bentuk Sistem Pakar**

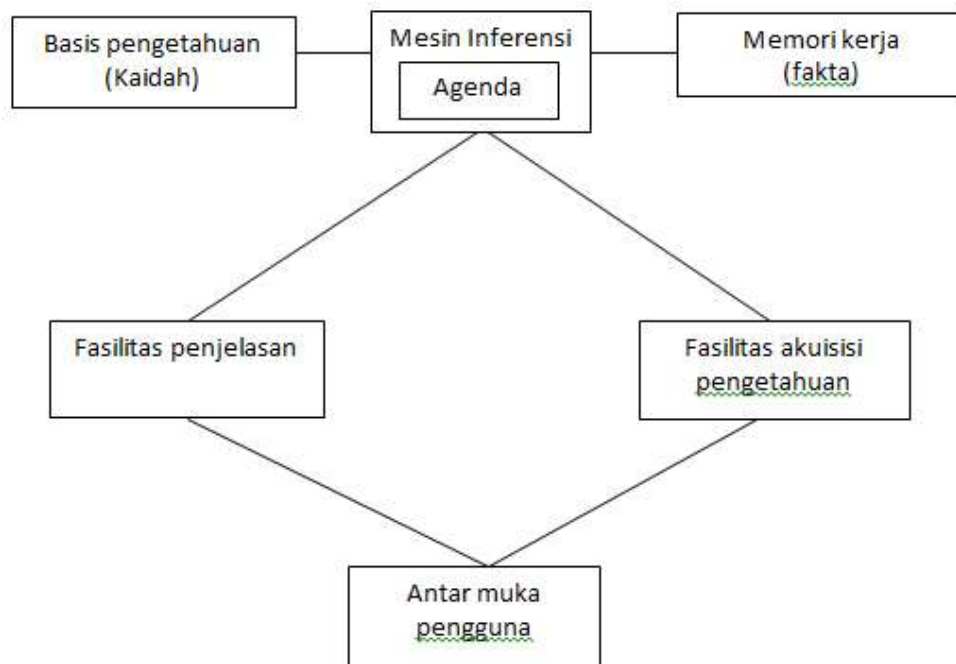
Menurut ada empat bentuk sistem pakar yaitu sebagai berikut:

1. Berdiri sendiri. Sistem pakar jenis ini merupakan *software* yang lainnya.
2. Tergabung. Sistem pakar jenis ini merupakan bagian program yang terkandung di dalam suatu algoritma (konvensional), atau merupakan program dimana didalamnya memanggil algoritma subrutin lain (konvensional)
3. Menghubungkan ke *software* lain. Bentuk ini biasanya merupakan sistem pakar yang menghubungkan ke suatu paket program tertentu, misalnya dengan DBMS
4. Sistem mengabdikan. Sistem pakar merupakan bagian dari komputer khusus yang dihubungkan dengan suatu fungsi tertentu. Misalnya sistem pakar yang digunakan untuk membantu menganalisis data radar.

### 2.1.4 Struktur Sistem Pakar

Sistem pakar disusun oleh dua bagian utama, yaitu:

- a. Lingkungan pengembangan (*Development Environment*), digunakan untuk memasukkan pengetahuan pakar kedalam lingkungan sistem pakar.
- b. Lingkungan konsultasi (*Consultation Environment*), digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar.



**Gambar 2.1** Struktur sistem pakar  
(Sumber: Hartiti, Sari Iswanti 2008:4)

### **2.1.5 Manfaat Sistem Pakar**

Menurut Sutojo T, dkk(2011:160) Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang diberikannya, diantaranya:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat dari pada 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 dilingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.

### **2.1.6 Komponen Sistem Pakar**

Hartono Milawati, Eko Nur Irsyad (2016:101) Menyatakan tentang komponen sistem pakar adalah:

1. Antar muka pengguna (*User Interface*)

Antar muka pengguna merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi.

2. Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi dan penyelesaian masalah.

### 3. Akuisisi Pengetahuan

Akuisisi pengetahuan adalah akumulasi, transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan kedalam program komputer.

### 4. Mesin Inferensi

Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace* untuk memformulasikan kesimpulan. Dalam perancangan sistem pakar ini penulis menggunakan metode runut maju (*forward chaining*). Merupakan proses peruntutan yang dimulai dengan menampilkan kumpulan data atau fakta yang menyakinkan menuju kesimpulan akhir.

### 5. Memori Kerja

Merupakan bagian dari sistem pakar yang menyimpan fakta-fakta yang diperoleh saat dilakukan proses konsultasi.

### 6. Fasilitas Penjelasan

Fasilitas penjelasan adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar.

### 7. Perbaikan Pengetahuan

Pakar memiliki kemampuan untuk menganalisis, meningkatkan kinerja dan kemampuan yang semua tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang akan dialaminya.

### 2.1.7 Runut Maju (*Forward Chaining*)

Tutik, dkk (2009) Metode *Forward Chaining* adalah metode pencarian atau teknik pelacakan kedepan yang dimulai dengan informasi yang ada berupa data *real* lalu bergerak maju melalui premis-premis dan penggabungan *rule* untuk menghasilkan suatu kesimpulan (*botton up reasoning*) atau tujuan. *Forward chaining* bekerja dari suatu masalah menu kepada solusinya (dalam Rosmawati Tamin, 2015:41).

*Forward chaining* merupakan metode inferensi yang melakukan penalaran dari suatu masalah kesolusinya. Jika klausa premis sesuai dengan situasi (bernilai *TRUE*), maka proses menyatakan konklusi. *Forward chaining* adalah *data-driven* karena inferensi dimulai dengan informasi yang tersedia dan baru konklusi diperoleh. Jika suatu aplikasi menghasilkan *tree* yang lebar dan tidak dalam, maka gunakan *forward chaining*.

Sistem yang dapat dicari dengan metode *forward chaining*

1. Sistem yang direpresentasikan dengan satu atau beberapa kondisi.
2. Untuk setiap kondisi, sistem mencari *rule-rule* yang berkorespondensi dengan kondisi bagian *IF*
3. Setiap *rule* dapat menghasilkan kondisi baru dari konklusi yang diminta pada bagian *THEN*. Kondisi ini ditambahkan ke kondisi lain yang sudah ada.
4. Setiap kondisi yang ditambah ke sistem akan diproses. Jika ditemui suatu kondisi dari konklusi yang diminta, sistem akan kembali ke langkah 2 dan

mencari *rule-rule* dalam *knowledge base* kembali, jika tidak ada konklusi baru, sesi ini berakhir.

Runut maju merupakan proses perunutan yang dimulai dengan menampilkan kumpulan data atau fakta yang meyakinkan menuju konklusi akhir. Runut maju bisa juga disebut sebagai penalaran *forward (forward reasoning)* atau pencarian yang dimotori data (*data driven search*). Jadi dimulai dari premis-premis atau informasi masukan (*if*) dahulu kemudian menuju konklusi atau *drived information (then)* atau dapat dimodelkan sebagai berikut:

*IF* (Informasi masukan)

*THEN* (Konklusi)

Informasi masukan dapat berupa data, bukti, temuan, atau pengamatan. Sedangkan konklusi dapat berupa tujuan, hipotesa, penjelasan, atau diagnosis. Sehingga jalannya penalaran runut maju dapat dimulai dari data menuju tujuan, dari bukti menuju hipotesa, dari temuan menuju penjelasan, atau pengamatan menuju diagnosa

Hartati, dan Sari Iswanti, (2008:25) Menjelaskan bahwa kaidah menyediakan cara formal untuk merepresentasikan rekomendasi, arahan, atau strategi. Kaidah produksi dituliskan dalam bentuk jika-maka (*if-then*) menghubungkan antesenden (*antecedent*) dengan konsekuensi yang diakibatkannya. Berbagai struktur kaidah *if-then* yang menghubungkan obyek atau atribut sebagai berikut:

1. *IF* premis *THEN* konklusi
2. *IF* masukan *THEN* keluaran

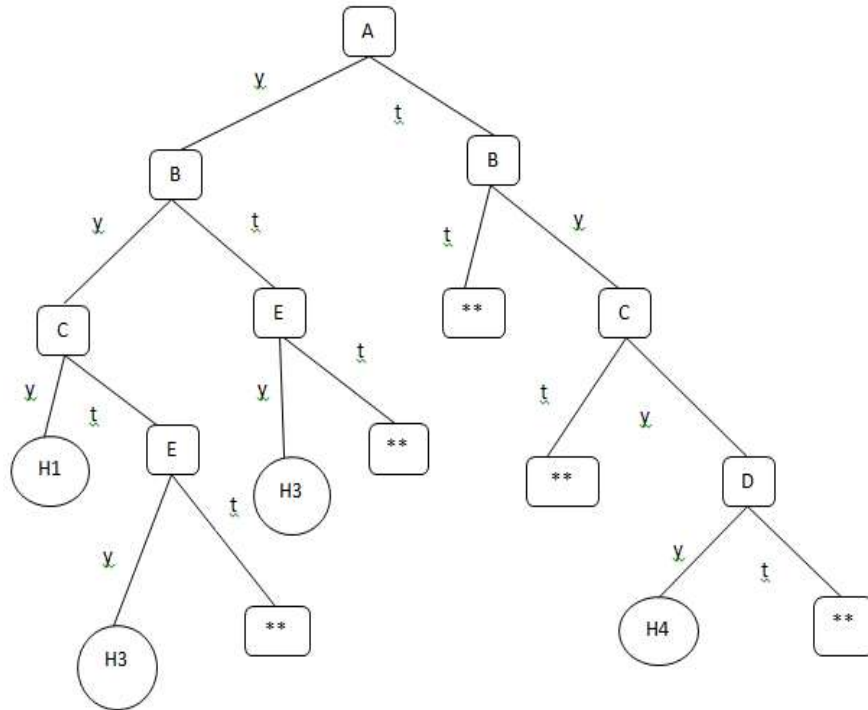
3. *IF* kondisi *THEN* tindakan
4. *IF* antesenden *THEN* konsekuen
5. *IF* data *THEN* hasil
6. *IF* tindakan *THEN* tujuan
7. *IF* aksi *THEN* reaksi
8. *IF* sebab *THEN* akibat
9. *IF* gejala *THEN* diagnosa

Sebelum sampai pada bentuk kaidah produksi, pengetahuan yng berhasil didapatkan dari domain tertentu disajikan dalam bentuk table keputusan kemudian dibuat pohon keputusannya. Berikut ini adalah contoh penyajian dalam bentuk table keputusan dan pohon keputusan(Hartati, dan Sari Iswanti, 2008:26)

**Tabel 2.1** Tabel Keputusan

<b>Hipotesa</b> <i>Evidence</i>	<b>Hipotesa</b> <b>1</b>	<b>Hipotesa</b> <b>2</b>	<b>Hipotesa</b> <b>3</b>	<b>Hipotesa</b> <b>4</b>
<i>Evidence A</i>	ya	ya	ya	tidak
<i>Evidence B</i>	ya	tidak	ya	ya
<i>Evidence C</i>	ya	tidak	tidak	ya
<i>Evidence D</i>	tidak	tidak	tidak	ya
<i>Evidence E</i>	tidak	ya	ya	tidak

Sumber: Hartati, dan Sari Iswanti (2008:32)



**Gambar 2.2** Pohon Keputusan  
(Sumber: Hartati, dan Sri Iswanti, 2008:33)

Keterangan:

A = *evidence* A,      H1 = hipotesa 1, y = ya

B = *evidence* B,      H2 = hipotesa 2, t = tidak

C = *evidence* C,      H3 = hipotesa 3, \*\* = tidak menghasilkan hipotesa tertentu

D = *evidence* D,      H4 = hipotesa 4,

Dilihat dari pohon keputusan gambar 2.2, masing-masing *node* yang mewakili *evidence* tertentu untuk kondisi “y” dan “t” sudah tidak mengarah pada



*evidence* yang sama. Dalam sesi konsultasi hal ini mengandung arti jawaban pengguna yang berbeda, akan mengarah pada pertanyaan yang berbeda pula. Kaidah yang dapat dihasilkan dengan mengacu pohon keputusan gambar 2.2 adalah sebagai berikut:

1. Kaidah 1 : *IF A and B and C THEN H1*
2. Kaidah 2 : *IF A and B and E THEN H3*
3. Kaidah 3 : *IF A and E then H2*
4. Kaidah 4 : *IF D and B and C then H4*

Model representasi pengetahuan kaidah produksi banyak digunakan pada aplikasi sistem pakar karena model representasi ini mudah dipahami dan bersifat deklaratif sesuai dengan jalan pikiran manusia dalam menyelesaikan suatu masalah, dan mudah diinterpretasikan.

## **2.2 Variabel Penelitian**

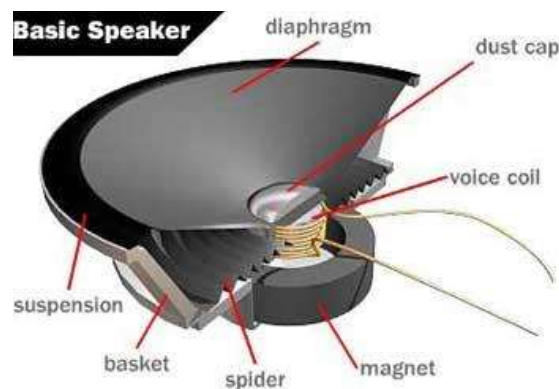
Variabel penelitian merupakan segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi tentang hal tersebut, kemudian ditarik kesimpulan. (Sugiyono, 2012:38). Objek yang digunakan pada penelitian ini adalah Speaker dan variabel yang ditetapkan yaitu kerusakan Speaker.

## 2.2.1 Speaker

PT FOTER, (2016: 1) Speaker adalah Suatu alat atau *transduser* yang digunakan untuk merubah gelombang listrik menjadi gelombang getaran *electro magnetic* yang akan menghasilkan gelombang suara yang bisa di dengar oleh gendang telinga.

### 1. Komponen utama

Komponen utama merupakan komponen-komponen penting yang harus ada dalam sebuah perangkat speaker. Tanpa komponen ini. Perangkat speaker tidak dapat bekerja menjalankan fungsinya. Adapun beberapa komponen dari speaker dan fungsinya antara lain adalah:



**Gambar 2.3** Speaker

(Sumber: PT FOSTER, 2016:1)

1. *Basket* atau Frame adalah kompoenen speaker yang berfungsi untuk melindungi seluruh bagian dalam speaker. Bahan yang digunakan biasanya terbuat dari Besi logam, plastik atau bahan campuran lainnya.



**Gambar 2.4**Frame  
(Sumber: PT FOSTER, 2016:1)

2. *Spider* adalah komponen yang berfungsi untuk menghasilkan gelombang tekanan yang diakibatkan oleh gerakan udara di sekitarnya dari pergerakan *voice coil*.



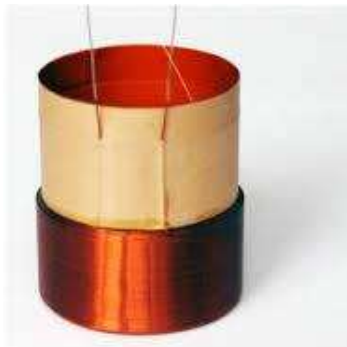
**Gambar 2.5**Spider  
(Sumber: PT FOSTER, 2016:1)

3. Magnet adalah komponen yang berfungsi melakukan induksi sehingga menghasilkan medan magnet.



**Gambar 2.6**Magnet  
(Sumber: PT FOSTER, 2016:2)

4. *Voice Coil* adalah komponen yang berfungsi mengubah dan mengalirkan energi listrik menjadi energi gerak menuju ke cone paper dan spider, perubahan yang terjadi dalam medan magnet speaker menyebabkan geraknya kumparan yang diakibatkan oleh interaksi antara kumparan dengan medan magnet.



**Gambar 2.7**Voice Coil  
(Sumber: PT FOSTER,2016:2)

5. *Dust cap* atau *Center Cap* adalah komponen yang digunakan untuk menutup lobang yang ada pada *voice coil* sehingga benda – benda tidak dapat masuk kedalam rongga *voice coil*.



**Gambar 2.8**Dust cap  
(Sumber: PT FOSTER,2016:2)

6. *Diaphragm* atau *Cone paper* adalah komponen yang berfungsi untuk menerima proses induksi dari magnet yang kemudian menghasilkan bunyi yang diakibatkan oleh getarannya atau induksi.



**Gambar 2.9**Diaphragm  
(Sumber: PT FOSTER,2016:2)

7. Terminal adalah komponen yang digunakan untuk penghubung kabel listrik dan biasanya terminal terdiri dari posisi terminal positif dan terminal negatif.



**Gambar 2.10**Terminal  
(Sumber: PT FOSTER,2016:2)

8. *Suspension* atau gasket adalah penahan dari cone paper agar tidak bergeser selama terjadi getaran.



**Gambar 2.11**Suspension  
(Sumber: PT FOSTER, 2016:2)

9. *Yoke* berguna untuk tempat berkumpulnya medan magnet dan membuat Voice coil bergerak.



**Gambar 2.12**Yoke  
(Sumber: PT FOSTER, 2016:3)

10. *Plate* untuk merekatkan antara Magnet dengan *Frame*



**Gambar 2.13**Plate  
(Sumber: PT FOSTER, 2016:3)

### 2.2.2 Kerusakan Speaker

PT FOSTER, (2016:9) Speaker dapat menjalankan fungsinya sebagai alat keluaran suara karena di dalamnya terdapat banyak komponen yang membutuhkan perawatan dan pemeliharaan yang baik.

#### 1. *No sound*

Lilitan *voice coil* ada yang putus, proses solder yang kurang bagus, terminal dengan solder ada yang terputus, magnet yang tidak diberikan medan magnet.

Solusi, ganti *voice coil*

#### 2. *Sound Precision NG* (Tekanan suara di beberapa frequency problem)

Pada permasalahan ini, proses pemberian medan magnet pada magnet tidak optimal, posisi *voice coil* agak miring (kurang center), bahan perekat atau lem yang digunakan kurang cepat kering. Salah menggunakan komponen.

Solusi, penambahan lapisan *flugs* pada lapisan terminal

#### 3. *Abnormal Sound* (perbedaan suara)

Permasalahan ini dapat disebabkan oleh adanya magnet yang pecah dan masuk ke dalam rongga *voice coil*, *voice coil* tidak bulat, posisi *yoke* bergeser, spider bersentuhan dengan *frame* dan posisi *cone paper* miring.

Solusi, *recharger magnet*.



#### 4. *Appearance Defect* (Problem secara visual)

Permasalahn ini dapat saja terjadi secara visual, *frame* pecah, *cone paper* atau *center cap* berjamur, magnet pecah, perketat menempel pada bagian speaker yang menyebabkan kotor, bekas solder menempel dan *frame* bengkok

Solusi, perbaikan metode penyolderan.(PT FOSTER, 2016)

### 2.3 Software Pendukung

*Software* pendukung merupakan beberapa perangkat lunak yang digunakan untuk mendukung pembuatan sistem pakar dalam peelitian ini. Perangkat lunak tersebut antara lain: *UML(Unifield Modelling Languange)*, *PHP*, *Adobe Dreamweaver*, *XAMPP*, *Domain name Url*, *MySQL*

#### 2.3.1 Unified Modeling Languange (UML)



**Gambar 2.14 Uml** (*Unified Modeling Language*)  
(Sumber:<http://staruml.sourceforge.net/image/staruml-logo.jpg>)

*Unified Modeling Language* (UML) adalah salah standar bahasa yang banyak digunakan didunia industri untuk mendefenisikan *requirement*, membuat analisis dan desain (Rosa A.S, M. Shalahuddin, 2011:117)

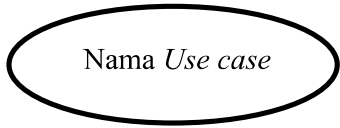

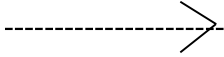
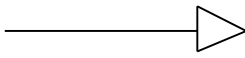
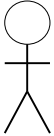
Paling tidak ada tiga karekter penting yang melekat pada UML, yaitu sketsa, cetak biru dan bahasa pemrograman. Sebagai sebuah sketsa, UML bisa berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem. UML bisa juga berfungsi sebagai cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi penting tentang *coding* program (*forward engeneering*) atau bahkan membaca program dan menginterpretasikannya kembali ke dalam diagram (*reverse engeneering*), sebagai bahasa pemrograman. UML dapat menterjemahkan diagram yang ada di UML menjadi code program yang siap untuk dijalankan.

### **2.3.1.1 Simbol-simbol Pada *Unified Modeling Language* (UML)**

#### *1. Use Case Diagram*

*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 actor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. (Rosa A.S, M. Shalahuddin, 2011:130)

Tabel 2.2 Simbol pada diagram *use case*

Simbol	Deskripsi
<p><i>Use case</i></p> 	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau actor
<p>Asosiasi / <i>Association</i></p> 	Komunikasi antara actor dengan <i>use case</i> yang berpartisipasi pada <i>use case</i>
<p>Ekstensi / <i>Extend</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu
<p>Generalisasi / <i>Generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari fungsi lainnya
<p>Aktor / <i>Actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibuat, jadi walaupun simbol dari actor adalah gambar orang. Biasanya dinyatakan menggunakan kata dinyatakan menggunakan kata benda diawal frase nama actor

Sumber: Rosa A. S, M. Shlahudin (2011:131)




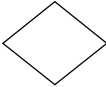

## 2. Activity Diagram

Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sistem.

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

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah penguin yang perlu didefinisikan kasus ujinya. (Rosa A.S, M. Shalahuddin, 2011:134)

**Tabel 2.3** Simbol pada *activity* diagram

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

Sumber: Rosa A. S, M. Shlahudin (2011:134)


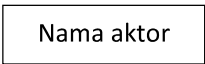
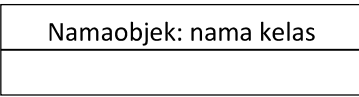


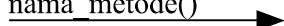
### 3. *Sequence* Diagram

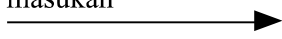
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Banyak diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use*

*case* yang telah didefinisikan interaksinya jalannya pesan sudah dicakup pada diagram urutan sehingga semakin banyak *use case* yang didefinisikan maka diagram urutan yang harus dibuat juga semakin banyak. (Rosa A.S, M. Shalahuddin, 2011:137)

**Tabel 2.4** Simbol pada *sequence* diagram

Simbol	Deskripsi
Garis hidup / <i>Lifeline</i> 	Menyatakan kehidupan suatu objek
Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem yang akan dibuat itu sendiri
Objek 	Menyatakan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
Pesan tipe <i>create</i> <<create>> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan tipe <i>call</i> 1: nama_metode() 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri

Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
Pesan tipe <i>return</i>	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

**Tabel 2.4**Lanjutan

Sumber: Rosa A. S, M. Shlahudin (2011:138)

#### 4. *Class* diagram

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

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


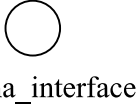
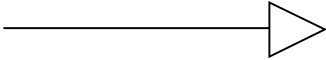


Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

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*

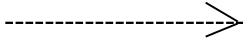
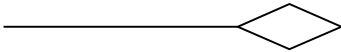
Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*

4. Kelas yang diambil dari pendefinisian data
5. Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kestuan yang diambil maupun akan disimpan kebasis data. (Rosa A.S, M. Shalahuddin, 2011:122)

**Tabel 2.5** Simbol pada *class* diagram

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>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
<p>Asosiasi berarah</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain
<p>Asosiasi</p> 	Relasi antar kelas dengan makna umum



Kebergantungan / <i>Dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>Agregation</i> 	Relasi antar kelas dengan makna semua-bagian ( <i>whole-part</i> )

**Tabel 2.5** Lanjutan

Sumber: Rosa A. S, M. Shlahudin (2011:123)

### 2.3.3 Hypertext Preprocessor (PHP)



**Gambar 2.15**PHP

(Sumber: <https://www.php.net/download-logos.php>)

PHP adalah kepanjangan dari *Hypertext Preprocessor*, merupakan sebuah bahasa pemrograman yang berupa kode atau *script* yang bisa diambahkan kedalam bahasa pemrograman HTML. PHP sendiri sering digunakan untuk hal merancang, membuat dan memprogram sebuah website. (Milawati Hartono, Eko Nur M Irsyad, 2016:51)

### 2.3.3.1 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari *personal homepage* (situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *form interpreted*(F1), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakanya PHP/F1. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk mengembangkan PHP.

Pada November 1997, dirilis PHP/F1 2.0. Pada rilis 2.0 ini, *interpreter* PHP sudah diimplementasikan dalam program C. Didalam rilis ini juga ikut disertakan modul-modul ekstensi yang meningkatkan kemampuan PHP/F1 secara signifikan. Kemudian pada juni 1998, sebuah perusahaan bernama Zend merilis *interpreter* terbaru untuk PHP dan meresmikan perilsan tersebut sebagai PHP 3.0 dan singkatan PHP menjadi akronim berulang *Hypertext Preprocessing*.

Pada pertengahan tahun 1999, Zend merilis *interpreter* PHP baru dan rilis tersebut dikenal dengan PHP 4.0. Versi ini banyak dipakai karena kemampuannya untuk membangun aplikasi *web* tkompleks tetapi memiliki kecepatan dan stabilitas yang tinggi.

Pada juni 2004, Zend merilis PHP 5.0, dalam versi ini, inti dari *interpreter* PHP mengalami perubahan yang besar, versi ini juga memasukkan model

pemrograman berorientasi objek kedalam PHP untuk menawab perkembangan bahasa pemrograman kearah pemrograman berorientasi objek.

### 2.3.3.2 Keunggulan PHP

PHP berbasis *server side scripting*. PHP sendiri dapat melakukan tugas-tugas yang dilakukan dengan mekanisme CGI seperti mengambil, mengumpulkan data dari *database*, meng-*generate* halaman dinamis, atau bahkan menerima dan mengirim *cookie*. Untuk lebih jelasnya berikut adalah keunggulan PHP:

1. PHP berbasis *server side scripting*

*Server side scripting* pada PHP dapat bekerja jika ada komponen berikut: PHP *parser* (CGI atau *server* modul), *web server* contohnya XAMPP, dan *web browser*.

2. *Command Line Scripting Pada PHP*

Menggunkan PHP dengan dan tanpa *web server* atau *browser*. Dengan pemanfaatan PHP pada *command line*.

3. PHP dapat membuat aplikasi dekstop

PHP mungkin bukan baasah yang ukup baik untuk suatu aplikasi di dekstop karena untuk bagian tamnpilan (*user interface*) agak sedikit sulit diimplementasikan dengan PHP saja, sedangkan *userinterface* merupakan salah satu point utama ketika kita membuat suatu aplikasi dekstop.

4. Digunakan untuk berbagai macam *platform OS*

Linux, Unix (including HP-UX, Solaris and Open BSD), Microsoft Windows, Mac OS X, RISC OS, dan lainnya. (Hidayatullah Priyanto, Jauhari Khairul Kawistara, 2015:234)

### 2.3.4 Adobe Dreamweaver



**Gambar 2.16** Adobe Dreamweaver

(Sumber: <http://www.adobe.com/products/dreamweaver.html>)

*Adobe dreamweaver* adalah aplikasi desain pengembangan *web* yang menyediakan halaman visual (bahasa sehari-hari yang disebut sebagai *design view*), maksudnya adalah tampilan hasil akhir *web* kita nanti akan sama pada saat tampilan proses perancangan halaman *web*.

Aplikasi ini menggunakan kode editor dengan fitur standar seperti *syntaxhighlighting*, *code completion*, dan *codecollapsing* serta fitur canggih seperti

*real-time syntax checking* dan *code introspection* untuk menghasilkan petunjuk kode untuk membantu pengguna dalam menulis kode.

*Dreamweaver* memiliki fitur dari *browser* yang terintegrasi untuk melihat halaman *web* yang dikembangkan di jendela pratinjau program sendiri agar konten memungkinkan untuk terbuka di *web browser* yang sedang telah terinstall.

Aplikasi ini menyediakan transfer dan fitur sinkronisasi, kemampuan untuk mencari dan mengganti baris teks atau kode untuk mencari kata dan kalimat biasa diseluruh situs.

*Adobe dreamweaver* merupakan program penyunting halaman *web* *adobe system*, yang dulu dikenal sebagai *Macromedia Dreamweaver* keluaran *macromedia*. Program ini banyak digunakan oleh pengembang *web* karena fitur-fiturnya yang menarik dan kemudahan penggunaannya. (Maudi Meiska Firstiar, dkk, 2014:102)

### 2.3.5 XAMPP



**Gambar 2.17**Xampp  
(Sumber:<https://brandeps.com/logo/X/Xampp-10>)

XAMPP adalah sebuah *software web server apache* yang didalamnya sudah tersedia *database Mysql* dan dapat mendukung pemrograman *Php*. Xampp merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di *linux* dan *windows*. Keuntungan lainnya adalah cuman menginstall satu kali sudah tersedia *Apache web server, Mysql database server, php support* (PHP 4 dan PHP 5) dan beberapa modul lainnya. (Februarianti Herny, Eri Zuliarso, 2012:129)

XAMPP adalah salah satu paket instalasi *apache*, *PHP*, dan *Mysql* secara instant yang dapat masuk kedalam halaman administrator. *User* (pengguna) hanya memiliki hak akses, seperti melihat koleksi buku-buku digital yang ada, mendownload buku digital, mendaftar sebagai member baru, dan menuliskan pesan kepada administrator melalui menu “*Contact Us*” (Wahana,2009:30).

### 2.3.6 Domain name/Url-Uniform Resource Lacator



**Gambar 2.18** Domain name/URL  
(Sumber: <https://www.hostinger.co.id/>)

Pengertian nama domain atau biasa disebut *Domain Name* atau *Url* adalah alamat unik didunia internet yang digunakan untuk mengidentifikasi sebuah *website*,

atau dengan kata lain *domain name* adalah alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet. Contoh <http://www.detik.com>, [www.unsrat.ac.id](http://www.unsrat.ac.id). Nama domain diperjual belikan secara bebas di internet dengan status sewa tahunan.(Pangerapan Octavia Ritha, 2013:765)

### 2.3.7MySQL



**Gambar 2.19**MySQL

(Sumber:<https://mysql.com/about/legal/logos.html>)

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relational (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License).Setiap pengguna dapat secara bebas menggunakan Mysql, Namun perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.Mysql sebnarnya turunan salah satu konsep utama dalam basisdata yang ada sebelumnya. (Februarianti Herny, Eri Zuliarso, 2012:128)

*Database Management System* (DBMS) adalah aplikasi yang dipakai untuk mengelola basis data. DBMS biasanya menawarkan beberapa kemampuan yang terintegrasi seperti:

1. Membuat, menghapus, menambah, dan memodifikasi basis data
2. Pada beberapa DBMS pengelolanya berbasis windows (berbentuk jendela-jemdela) sehingga lebih mudah digunakan
3. Tidak semua orang bisa megakses basis data yang ada sehingga memberikan keamanan bagi data
4. Kemmpuan berkomunikasi dengan program aplikasi yang lain. Misalnya dimungkinkan untuk mengakses basis data MySQL menggunakan aplikasi yang dibuat menggunakan PHP
5. Kemampuan pengaksesan melalui komunikasi antar komputer (*Client server*)

MySQL adalah sebuah perangkat lunak manajemen berbasis data *Sql DatabaseManagement System* atau DBMS dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL dan lainnya.

## **2.4 Penelitian Terdahulu**

Pada penelitian sebelumnya yang sudah dilakukan yang digunakan sebagai dasar dan pertimbangan dalam pembuatan penelitian ini antara lain:

**RIVO YOUFLY RURUT, 2016, RANCANG BANGUN ALAT PENGONTROL POLARIS LOUD SPEAKER BERBASIS ATMEGA8535.** Pada perancangan sistem tester Polaris audio komponen-komponen dirangkai pada satu papan PCB agar lebih efesien dan praktis. Komponen utama dari rangkaian yang



pertama *Microcontroler* ATmega 8535. Dengan alat ini dapat memeriksa *polaritis* dari speaker, apakah itu pertama *feedback* atau non *feedback* (seiring keliru disebut *fase*, yang juga waktu dan frekuensi yang terkait).

**AKIM MANAOR HARA PARDEDE, 2013, PERANCANGAN SISTEM PAKAR DETEKSI KERUSAKAN PRINTER CANON BERBASIS WEB.** Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya dapat dipecahkan oleh orang pkr dalam hal bidang tertentu. Aplikasi sistem pakar ini dibuat untuk menentukan jenis kerusakan mesin printer canon jenis ink jet dan intensitas yang tepat bagi penggunaannya menghasilkan keluaran berupa saran jenis kerusakan dan intensitas solusi yang tepat bagi penggunaannya berdasarkan gejala yang diinput oleh user. Data dianalisa dengan menggunakan metode *backward chaining*, dengan bahasa pemrograman PHP dan database MySQL.

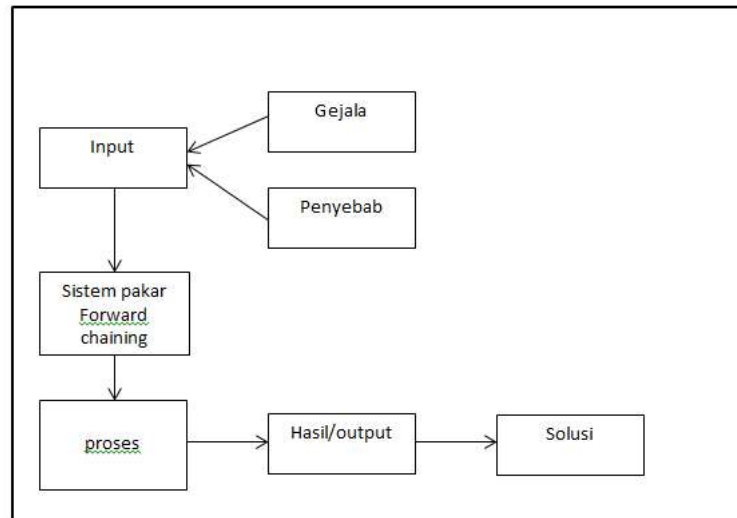
**ROSMAWATI TAMIN, 2015, SISTEM PAKAR UNTUK DIAGNOSA KERUSAKAN PADA PRINTER MENGGUNAKAN METODE FORWARD CHAINING.** Perancangan sistem dengan membuat *knowledge base* menggunakan *decision tree* dan aturan *if-then* sebagai representasi pengetahuan. Sistem dibuat dengan metode *forward chaining* dan bahasa pemrograman visual basic. Pengujian aplikasi juga dilakukan untuk mengetahui kurasi dan variasi serta *user friendly* dan fleksibilitas sistem. Dari keseluruhan pengujian ini dapat disimpulkan bahwa program sudah cukup baik walaupun jenis kerusakannya yang dihasilkan belum lengkap karena pada sistem ini hanya mendeteksi 15 jenis kerusakan secara umum.

**SARTINI, 2015, SISTEM PAKAR IDENTIFIKSI KERUSAKAN HARDWARE HANDPHONE DENGAN MENGGUNAKAN METODE FORWARD CHAINING.**Basis pengetahuan untuk membuat sistem pakar identifikasi gejala mulai dari data penginputan data pertanyaan, data jawaban, pendaftaran pengguna, hingga proses konsultasi. Metodologi yang digunakan adalah membuat aturan (*rule*) dengan teknik pelacakan kedepan (*forward chaining*) dan menggunakan topology pencarian pertama yang terbaik. Dengan *website* sistem pakar ini diharapkan dapat menyediakan informasi tentang gejala kerusakan *handphone*.

**ACHMAD KARIDI, BUDI RAHMAN, SYAHIB NATARSYAH, 2014, SISTEM PAKAR DETEKSI KERUSAKAN PLC (POWER LINE CARRIER) DENGAN METODE FORWARD CHAINING.**Sistem pakar yang dibangun adalah untuk mendeteksi kerusakan PLC dengan menerapkan metode *forward chaining*. Sistem ini menggunakan bahasa pemrograman *Delphi* yang *databasenya* dengan *Microsoft acces*. Pada penelitian yang telah dilakukan, gejala kerusakan PLC (Power Line Carrier) dicoba untuk dirangkum dan dianalisis.

## **2.5 Kerangka Pemikiran**

Menurut Sugiono, (2012:63) Setelah sintesa atau kesimpulan sementara dapat dirumuskan maka selanjutnya disusun kerangka berfikir. Kerangka berfikir yang dihasilkan dapat berupa kerangka berfikir yang *assosiatif* atau hubungan maupun *komparatif* atau perbandingan.



**Gambar 2.20** Kerangka pemikiran  
(Sumber: Data Penelitian, 2017)

Dengan gambar ini diatas menjelaskan tentang prosesnya sistem pakar mendeteksi kerusakan pada speaker. Kemudian sistem pakar akan memprosesnya menggunakan metode *forward chaining* berbasis *web*. Dengan menggunakan aplikasi berbasis *web* maka akan mempermudah masyarakat umum untuk mencari informasi yang diperlukan hanya dengan mengakses melalui situs *web* setelah proses berjalan dan selesai maka aplikasi *web* akan menghasilkan analisis dan solusi.

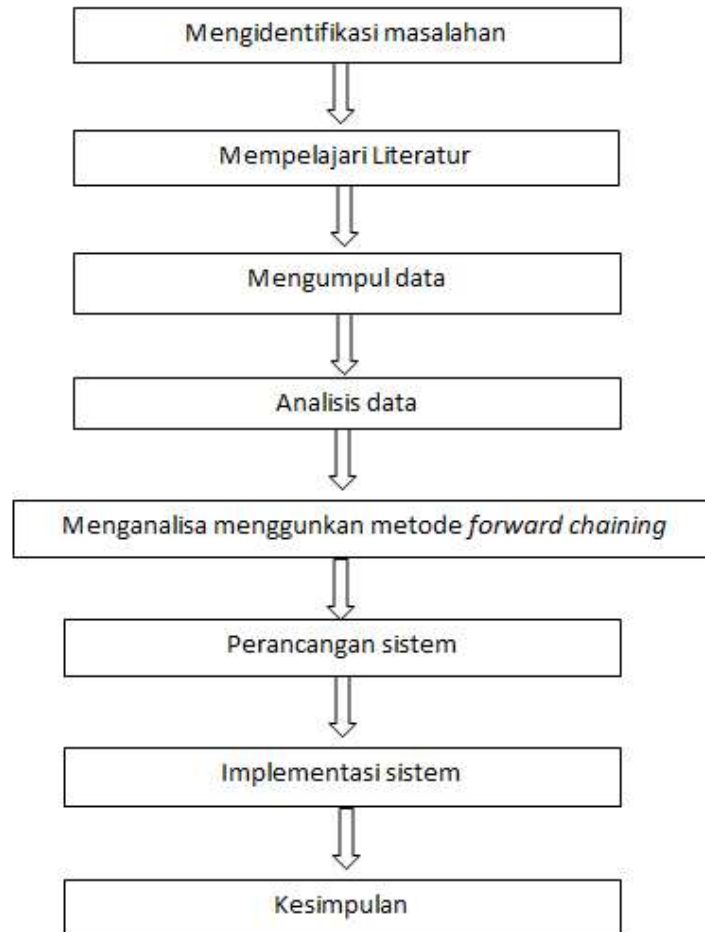
## **BAB III METODE PENELITIAN**

### **3.1 Desain Penelitian**

Menurut Sugiyono (2012: 297) Metode penelitian dan pengembangan atau dalam bahasa Inggrisnya *Research and Development* adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut.

Metode penelitian dan pengembangan telah banyak digunakan pada bidang-bidang Ilmu Alam dan Teknik. Hampir semua produk teknologi, seperti alat-alat elektronik, kendaraan bermotor, pesawat terbang, kapal laut, alat-alat kedokteran, bangunan gedung bertingkat dan alat-alat rumah tangga yang modern diproduksi dan dikembangkan melalui penelitian dan pengembangan. Namun demikian metode penelitian dan pengembangan bisa juga digunakan dalam bidang ilmu-ilmu sosial seperti psikologi, sosiologi, pendidikan, manajemen, dan lain-lain.

Penelitian ini menggunakan desain penelitian dengan beberapa tahap proses penelitian seperti yang terlihat pada gambar dibawah ini.



**Gambar 3.1** Desain Penelitian  
(Sumber: Pengolahan data penulis, 2017)

Berikut ini adalah penjelasan dari desain penelitian yang terdapat pada gambar tersebut :

1. Mengidentifikasi masalah

Penelitian diawali dengan melakukan studi pendahuluan untuk mengidentifikasi permasalahan yang berkaitan dengan topik penelitian agar peneliti mendapatkan apa yang sesungguhnya menjadi masalah untuk dapat dipecahkan.

## 2. Mempelajari Literatur

Untuk mendukung jalannya penelitian, peneliti mencari dan mempelajari sumber-sumber pengetahuan berupa buku-buku teori, jurnal-jurnal penelitian, dan sumber pustaka otentik lainnya yang berkaitan dengan penelitian, diantaranya yaitu kecerdasan buatan, sistem pakar, speaker, PHP, Mysql, dan UML. Peneliti merumuskan masalah yang telah didapatkan secara lebih spesifik agar masalah tersebut dapat dijawab dengan baik melalui penelitian tersebut

## 3. Pengumpulan data

Pengumpulan data dapat diambil dari buku maupun dari jurnal yang berhubungan dengan kerusakan pada speaker. Peneliti mendapatkan data-data tentang kerusakan speaker melalui wawancara

## 4. Analisi 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 gejala-gejala kerusakan pada speaker ataupun dari buku yang berhubungan langsung dengan kerusakan tersebut.

## 5. Mengolah data dengan metode *forward chaining*

Setelah penulis mengumpulkan data-data dari hasil wawancara dengan seorang pakar tentang gejala-gejala kerusakan speaker dan solusinya, lalu peneliti mengolah data tersebut agar dapat dijadikan sebuah sistem pakardengan metode *forward chaining*

#### 6. Perancangan sistem

Tahap selanjutnya adalah perancangan sistem. Perancangan sistem ini dibuat untuk mendeteksi kerusakan pada speaker berbasis web. Perancangan dari model sistem, perancangan sistem input dan merancang rule-rule yang akan digunakan dalam mendeteksi kerusakan pada speaker berdasarkan data yang ada.

#### 7. Implementasi sistem

Setelah perancangan sistem, diharapkan sistem tersebut mampu memecahkan masalah kerusakan pada speaker dan mempermudah user untuk mengetahui informasi serta melakukan perbaikan jika terjadi kerusakan pada speaker

#### 8. Kesimpulan

Tahap terakhir dalam penelitian ini yaitu menyimpulkan hasil penelitian yang berisi jawaban singkat terhadap rumusan masalah berdasarkan data-data yang ada. Peneliti juga memberikan saran yang penting untuk membantu dalam memecahkan permasalahan yang ada.

### **3.2 Teknik Pengumpulan Data**

Teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Tanpa tau mengatahui teknik pengumpulan data, maka peneliti tidak akan mendapatkan data yang memenuhi syandar data yang ditetapkan. Teknik pengumpulan data yang digunakan dalam penelitian ini adalah:

## 1. Wawancara

Wawancara adalah pengumpulan data yang digunakan untuk memperoleh informasi langsung dari sumbernya. Untuk mendapatkan informasi dan data-data yang berkaitan dengan penelitian, peneliti melakukan wawancara langsung dengan bapak Oman Al Imron yang berkerja sebagai QC Specialis di perusahaan PT FOSTER dalam metode wawancara, alat bantu perekam suara yang digunakan untuk proses wawancara. Pedoman wawancara yang digunakan berupa garis-garis besar permasalahan yang akan ditanya yaitu hal-hal yang berkaitan dengan speaker dan kerusakan yang terjadi pada speaker serta solusi.

## 2. Studi Literatur

Peneliti melakukan studi literature dengan mengumpulkan, membaca, dan memahami referensi teoritis yang bersal dari buku-buku teoti, buku-buku elektronik (*e-book*), jurnal-jurnal penelitian, dan sumber pustaka lainya yang berkaitan dengan penelitian. Studi literatur bertujuan untuk menemukan variable yang akan diteliti, membedakan hal-hal yang sudah dilakukan dan menentukan hal yang perlu dilakukan, melakukan sintesa dan memperoleh prespektif baru, dan menentukan makna dan hubungan antara variabel.

### **3.3 Operasional Variabel**

Variabel penelitian pada dasarnya adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi dan kesimpulannya (Sudaryono, 2015:16). Variabel yang digunakan dalam penelitian ini adalah kerusakan speaker. Kerusakan speaker dapat didefinisikan



sebagai berkurangnya fungsi kerja pada sebuah perangkat speaker termasuk komponen-komponen yang terdapat didalamnya dari komponen pendukung dan komponen utama sehingga tidak dapat berfungsi semestinya. Terdapat beberapa bagian yang mempengaruhi kinerja sebuah speaker sekaligus menjadi indikator kerusakan speaker. Bagian-bagian tersebut adalah:

1. *No sound*
2. *Sound precision NG* (Tekanan suara di beberapa Frequency problem)
3. *Abnormal sound* (Perbedaan suara)
4. *Appearance defect* (Problem secara visual)

Variabel dan indikator kerusakan speaker disajikan dalam table dibawah ini.

**Tabel 3.1** variabel dan indikator

<b>Variabel</b>	<b>Indikator</b>
Kerusakan speaker	<i>No sound</i>
	<i>Sound precision NG</i>
	<i>Abnormal sound</i>
	<i>Appearance defect</i>

(Sumber: Pengolahan data penulis, 2017)

### **3.4 Metode Perancangan Sistem**

Perancangan sistem bisa kita gambarkan dalam suatu bagan alir yang menjelaskan keseluruhan proses yang kita lakukan. Jika ada data ril yang kita ambil. Jelaskan kapan, dimana, dan bagaimana data kita ambil.

### 3.4.1 Desain Basis Pengetahuan

Setelah mengumpulkan data-data dari sumber yang tersedia dan fakta-fakta yang ada sehingga dibuat sebagai tabel yang berhubungan dengan kerusakan speaker, gejala kerusakan speaker, penyebab kerusakan dan solusi memperbaikinya. Di sajikan dalam bentuk sebuah tabel.

**Tabel 3.2** Indikator

<b>Kode</b>	<b>Nama Bagian</b>
IND01	<i>No sound</i>
IND02	<i>Sound precision NG</i>
IND03	<i>Abnormal sound</i>
IND04	<i>Appearance defect</i>

(Sumber: Pengolahan data penulis, 2017)

**Tabel 3.3** Tabel Penyebab

<b>Kode Penyebab</b>	<b>Nama Penyebab</b>	<b>Solusi</b>
PYB01	Lilitan <i>voice coil</i> ada yang putus	Ganti <i>voice coil</i> dengan yang baru
PYB02	Proses pemberian medan magnet tidak optimal	Penetralkan magnet dan <i>recharge</i> ulang magnet
PYB03	Adanya magnet yang pecah dan masuk kedalam rongga <i>voice coil</i>	Pemberishan magnet sebelum pemasukan ke <i>yoke</i>
PYB04	<i>Cone paper</i> atau <i>center cap</i> berjamur	Pergantian <i>center cap</i> dan <i>cone paper</i> dengan yang baru

(Sumber: Pengolahan data penulis, 2017)

Sistem pakar yang menggunakan metode *forward chaining* pada penelitian ini digunakan untuk melakukan mendeteksi kerusakan speaker, sehingga data

solusi tidak diberikan kode. Data solusi hanya sebagai keterangan tambahan yang digabungkan kedalam tabel.

**Tabel 3.4** Tabel Gejala

<b>Gejala</b>	<b>Nama Gejala</b>
Gjl01	Suara naik turun secara derastis
Gjl02	Suara antara hidup dan mati
Gjl03	Suara mati total
Gjl04	Suara sangat kecil
Gjl05	Suara berdengung dan frekuensi kecil
Gjl06	Pergerakan <i>voice coil</i> lambat
Gjl07	Suara kasar dan tidak halus
Gjl08	Adanya suara gesekan
Gjl09	Suara getaran yang keras pada sisi speaker
Gjl10	<i>Visual</i> /penampakan yang tidak sesuai dengan spesifikasi yang ditetapkan
Gjl11	<i>Cone paper</i> sangat rapuh
Gjl12	<i>Center cap</i> berlubang

(Sumber: Pengolahan data penulis, 2017)

Data aturan merupakan data yang berisi relasi antara data-data bagian kerusakan, penyebab kerusakan dan gejala kerusakan yang telah diberi kode sebelumnya. Relasi antar data tersebut disusun berdasarkan sumber pengetahuan

dan fakta yang telah didapatkan. Data aturan ini disusun untuk memudahkan peneliti dalam menyusun kaidah yang akan digunakan sebagai basis pengetahuan dalam sistem pakar pada penelitian ini. Susunan data aturan yang digunakan dapat dilihat pada tabel dibawah ini.

**Tabel 3.5** Tabel Aturan

<b>Kode Indikator</b>	<b>Kode Penyebab</b>	<b>Kode Gejala</b>
IND01	PYB01	GJL01,GJL02,GJL03, GJL04
IND02	PYB02	GLJ05,GJL06,GJL07,GJL08
IND03	PYB03	GJL07,GJL08,GJL09
IND04	PYB04	GJL10,GJL11,GJL12

(Sumber: Pengolahan data penulis, 2017)

Berdasarkan data aturan yang telah disusun, maka kaidah yang akan digunakan dalam sistem pakar dan tabel keputusanya adalah sebagai berikut:

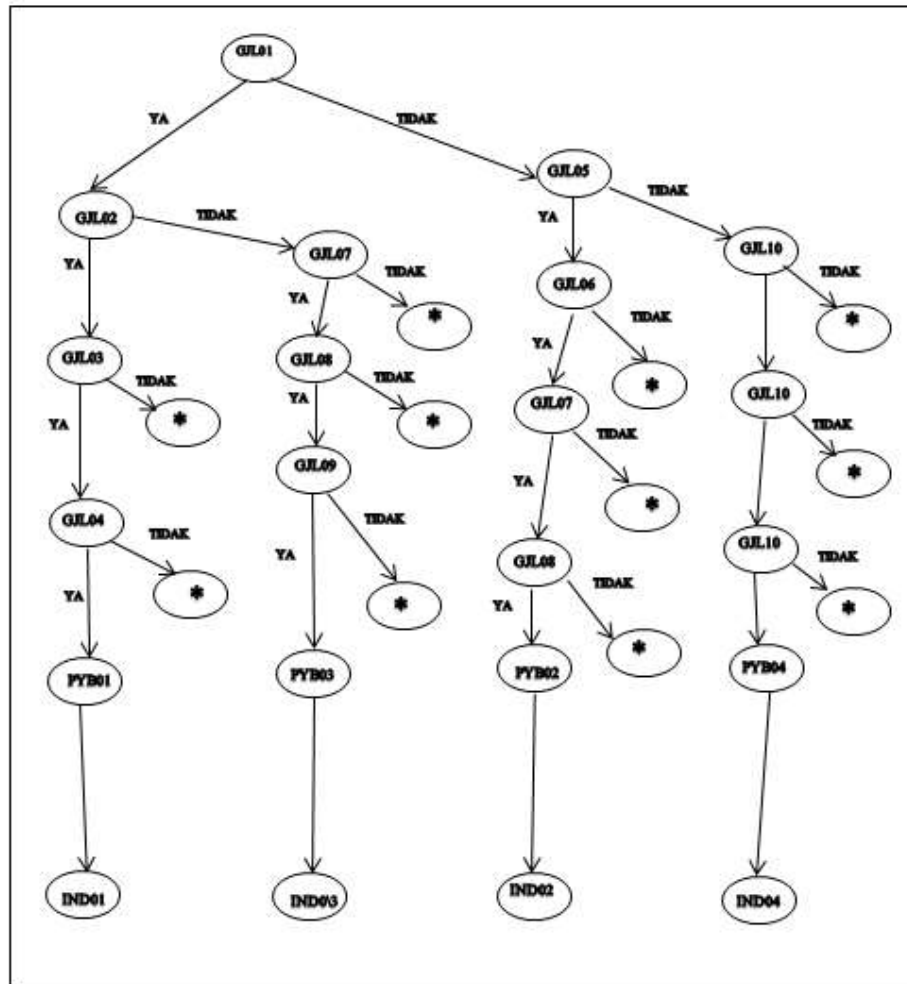
1. Kaidah 1: *IF GJL01 AND GJL02 AND GJL03 AND GJL04 THEN PYB01*
2. Kaidah 2: *IF GJL05 AND GJL06 AND GJL07 ANDGJL08THEN PYB02*
3. Kaidah 3: *IF GJL07 AND GJL08 AND GJL09 THEN PYB03*
4. Kaidah 4: *IF GJL10 AND GJL11 AND GJL12 THEN PYB04*

Tabel 3.6 Tabel Keputusan

BAGIAN	IND01	IND02	IND03	IND04
PENYEBAB GEJALA	PYB 01	PYB 02	PYB 03	PYB 04
GJL01	√			
GJL02	√			
GJL03	√			
GJL04	√			
GJL05		√		
GJL06		√		
GJL07		√	√	
GJL08		√	√	
GJL09			√	
GJL10				√
GJL11				√
GJL12				√

(Sumber: Pengolahan data penulis, 2017)

Berdasarkan tabel keputusan tersebut maka pohon keputusan adalah sebagai berikut:



**Gambar 3.2** Pohon Keputusan  
(Sumber: Pengolahan data penulis, 2017)

Pohon keputusan dibuat digunakan untuk memperlihatkan hubungna yang terkait dengan gejala-gejala yang ada. Pohon keputusan tersebut dimulai dari yang paling atas menuju yang paling bawah. Alur dimulai dari GJL01 dan dilanjutkan ke GJL02 jika pengguna memberikan jawaban “ya” dan jika “tidak” maka kembali ke gejala awal yaitu GJL01, maka sistem akan melakukan penelusuran ulang dan sampai menemukan hasil yang berupa IND01. Begitu juga dengan seterusnya.

### 3.4.2 Struktur kontrol (mesin inferensi)

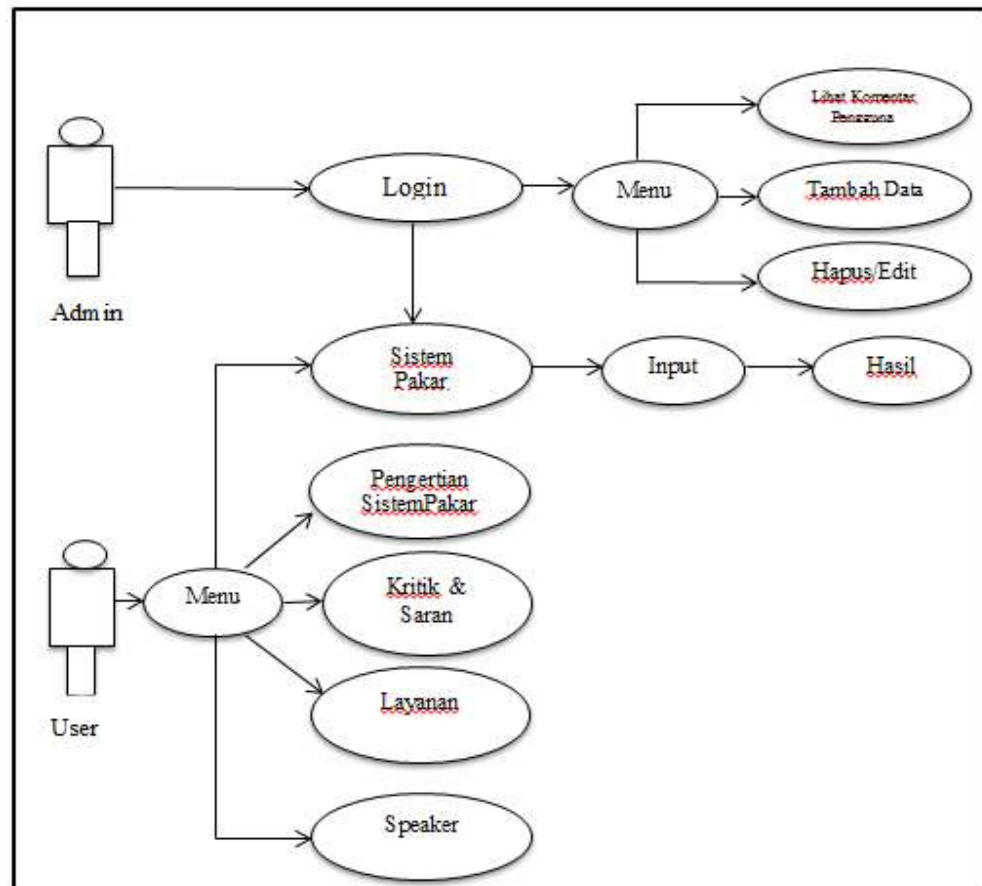
Mesin inferensi dalam sistem pakar ini menggunakan metode penelusuran *forward chaining*. Langkah-langkah yang digunakan dalam proses penelusurannya adalah sebagai berikut:

1. Mengajukan pertanyaan tentang gejala kerusakan kepada pengguna.
2. Menyimpan sementara jawaban pengguna tentang gejala kerusakan dan kemungkinan penyebab kerusakan ke dalam memori sementara
3. Memeriksa gejala-gejala yang ada dengan aturan yang telah dibuat, jika ada konklusi yang cocok maka simpan hasil kedalam memori tetap (table hasil *database*), jika belum memenuhi konklusi apapun, ulangi langkah 1 sampai 3.
4. Menampilkan hasil diagnosa kerusakan speaker

### 3.4.3 Desain UML (*Unified Modeling Language*)

#### 1. *Use caseDiagram*

Use case diagram menjelaskan siapa saja aktor-aktor yang terlibat dengan perangkat lunak yang dibangun beserta proses-proses di dalamnya. Berikut ini adalah gambar *user case diagram admin* dan *user* :



**Gambar 3.3**Diagram Use Case Admin dan User  
(Sumber :Pengolahan data penulis, 2017)

1. Admin

Admin disini dapat melakukan login masuk ke menu kemudian dari menu admin dapat melihat dan mengubah dari data yang ada pada data *base*.

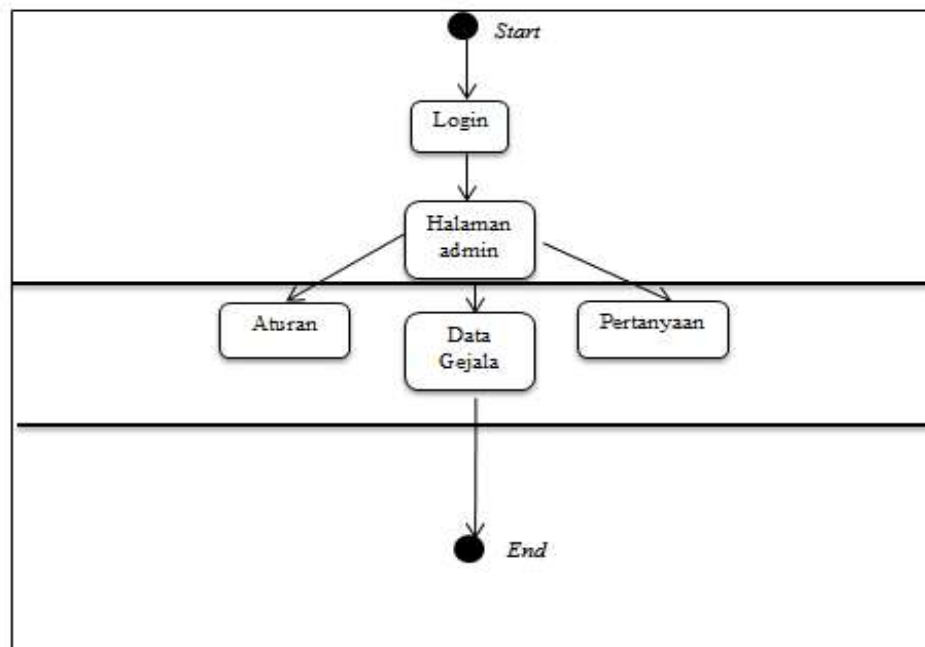
2. User

User bisa memulai masuk dari menu kemudian memilih beberapa opsi yang ingin dilakukan oleh user, dan juga dapat konsultasi langsung sehingga dapat menghasilkan *output*.



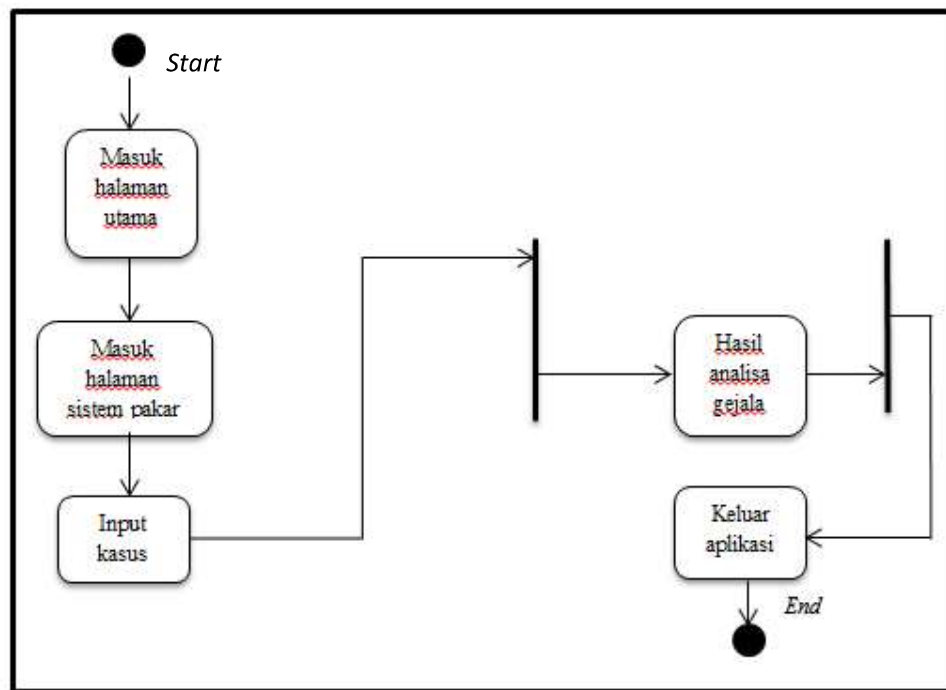
## 2. Activity Diagram

*Activity diagram* *Activity diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem, yang perlu diperhatikan gambar dibawah ini adalah bahwa *activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor.



**Gambar 3.4** *Activity diagram* admin  
(Sumber: Pengolahan data penulis, 2017)

Alur kerja dari pada *activity diagram* admin dimulai dengan *Start* kemudian login admin masuk kedalam halaman admin lalu bias memilih tiga opsi untuk mana yang ingin digunakan, setelah selesai diakhiri dengan *End*.

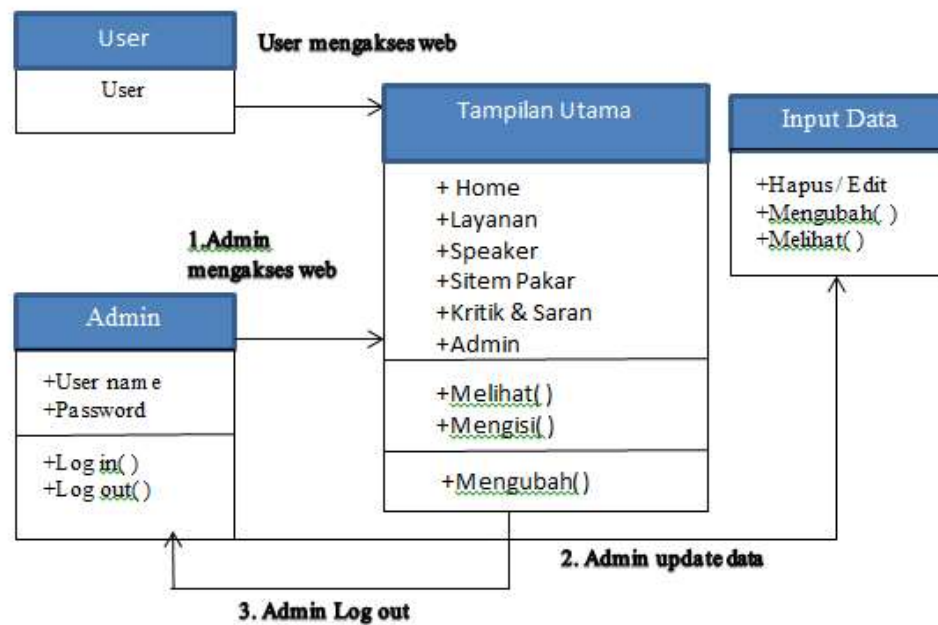


**Gambar 3.5** Activity diagram user  
(Sumber: Pengolahan data penulis, 2017)

Alur kerja Activity diagram user dimulai dari Start masuk ke halaman utama kemudian masuk ke halaman sistem pakar kemudian input dari pada kasus atau pertanyaan, kemudian dihasilkan analisis data yang telah di input kemudian keluar aplikasi dan End.

### 3. ClassDiagram

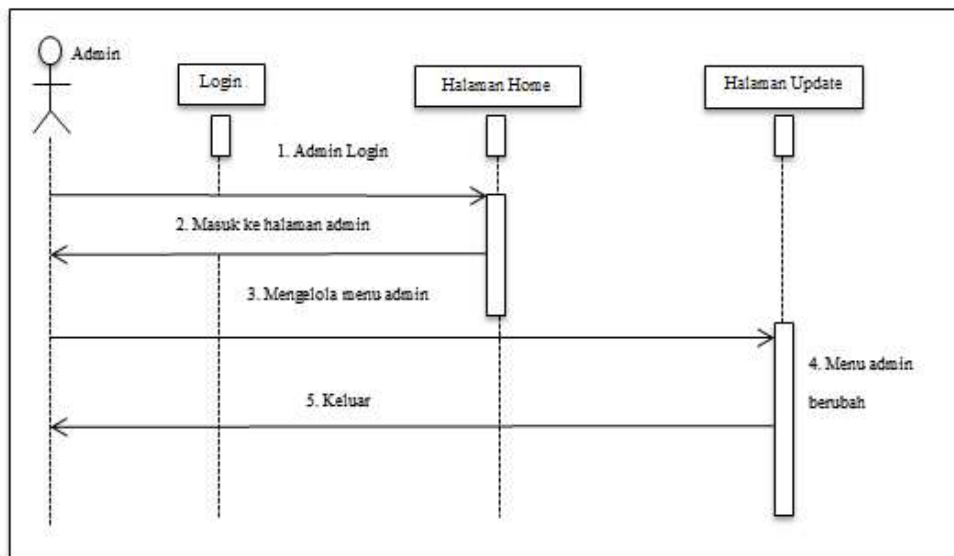
*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut ini adalah gambar *class diagram* :



**Gambar 3.6** Class diagram  
(Sumber: Pengolahan data penulis, 2017)

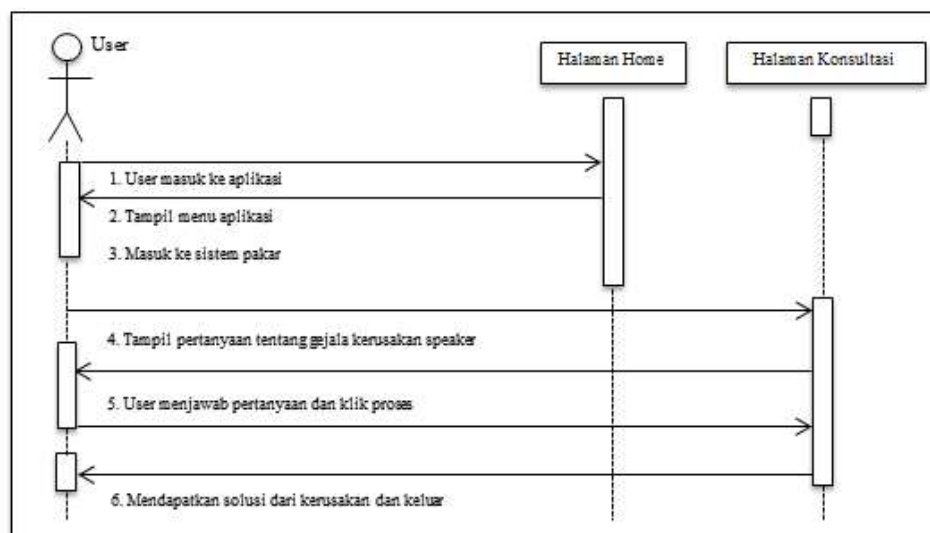
### 4. Sequence Diagram

*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek tersebut. Berikut sequendiagram admin dan *user* pada aplikasi sistem pakar ini:



**Gambar 3.7** *Sequence Diagram Admin*  
(Sumber: Pengolahan data penulis, 2017)

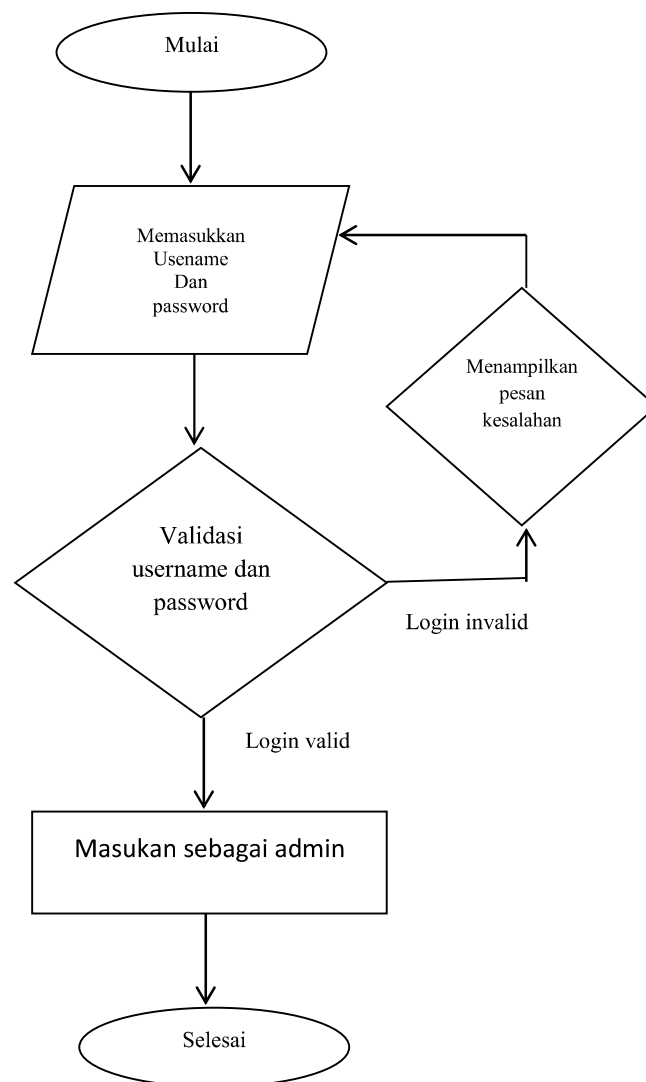
*Sequence diagram admin* di atas menggambarkan interaksi admin antar objek dan mengidinkasikan diantara bojek tersebut di sistem pakar dalam penelitian ini.



**Gambar 3.8** *Sequence Diagram User*  
(Sumber: Pengolahan data penulis, 2017)

Sequence diagram user diatas yang menggambarkan interaksi user antar objek dan mengidikasikan diantara bojek tersebut di sistem pakar dalam penelitian ini.

## 5. Flowchart



**Gambar 3.9**Flowchart

(Sumber: Pengolahan data penulis, 2017)

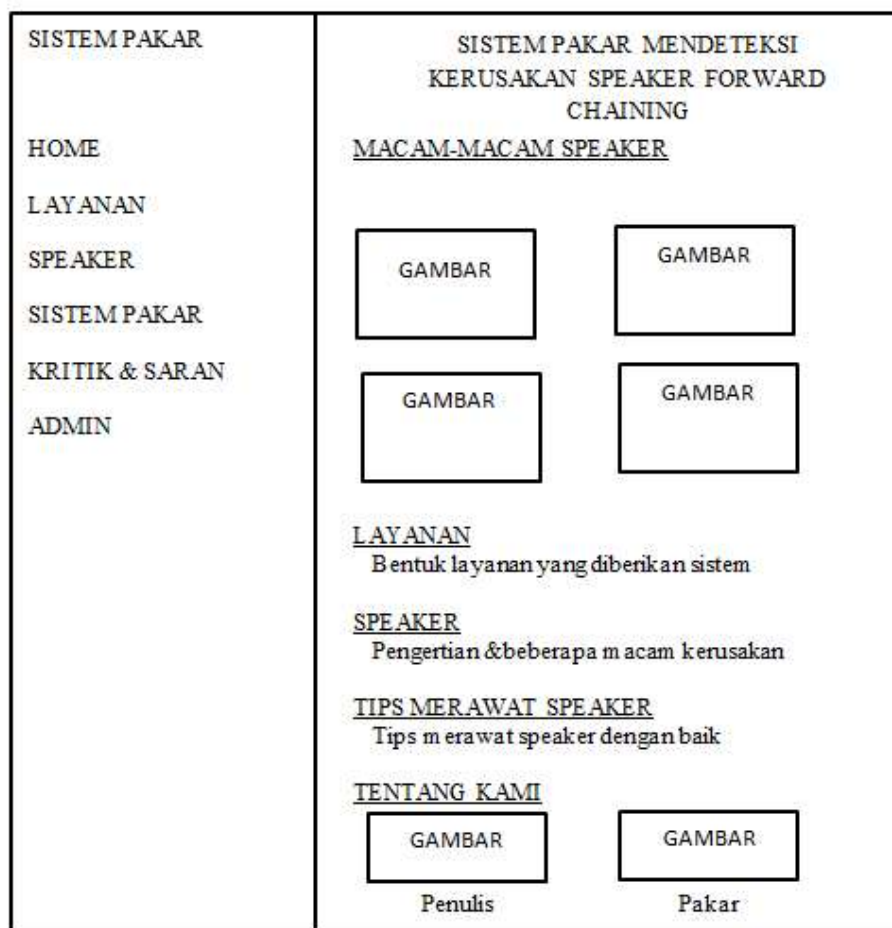
Flowchart menggambarkan sistem kerja dari pada sistem pakar yang dibuat untuk menentukan hasil dan sebuah solusi bagi pengguna.

### 3.4.4 Desain Antarmuka (*prototype*)

#### A. Halaman user

##### 1. Halaman home

Halaman ini merupakan tampak halaman awal dari pada bentuk sisitem pakar yang dibuat. Yang terdiri dari beberapa opsi yang bisa dilihat dari pada halaman.



**Gambar 3.10** Rancangan halaman home  
(Sumber: Pengolahan data penulis, 2017)

## 2. Halaman konsultasi sistem pakar

<p>SITEM PAKAR</p>  <p>HOME</p> <p>KRITIK &amp; SARAN</p> <p>ADMIN</p>	<p>SISTEM PAKAR MENDETEKSI KERUSAKAN SPEAKER FORWARD CHAINING</p> <div data-bbox="831 506 1131 596" style="border: 1px solid black; text-align: center; padding: 5px;">GAMBAR</div> <p>KONSULTASI PERTANYAAN</p> <p><u>1. PERTANYAAN</u></p> <p><input type="radio"/> YA    <input type="radio"/> TIDAK</p> <div data-bbox="756 806 894 863" style="border: 1px solid black; text-align: center; padding: 5px;">LANJUT</div>
--	--

**Gambar 3.11** Rancangan halaman konsultasi sitem pakar  
(Sumber: Pengolahan data penulis, 2017)

Pada halaman ini user dapat melakukan konsultasi terhadap masalah yang mereka hadapi tentang kerusakan dari pada speaker tersebut. User dimudahkan dengan pilihan “YA” & “TIDAK” untuk menjawab kerusakan dari pada speaker yang rusak.

### 3. Halaman kritik & saran

SITEM PAKAR	NAMA <input type="text"/>
HOME	EMAIL <input type="text"/>
KRITIK & SARAN	KRITIK DAN SARAN <input type="text"/>
ADMIN	

**Gambar 3.12**Rancangan halaman kritik & saran  
(Sumber: Pengolahan data penulis, 2017)

Kemudian pada halaman ini, user dapat memberikan masukan berupa kritik dan saran untuk memperbaiki dari mana kurangnya sebuah sistem pakar yang dibuat.



B. Halaman admin

1. Halaman CRUD admin

SELAMAT DATANG DIHALAMAN ADMIN				
TAMBAHKAN DATA		KELUAR		
NO	Pertanyaan & solusi	Bila benar	Bila salah	Opsi
1				Edit Hapus
2				Edit Hapus
3				
4				

**Gambar 3.13**Rancangan halaman CRUD admin  
(Sumber: Pengolahan data penulis, 2017)

Halaman ini berupa form admin yang gunannya dari pada untuk menambah kekurangan yang ingin ditambahkan dari sistem pakar ini. Dari berupa pertanyaan & solusi,

## 1. Halaman kritik &amp; saran

<div style="border: 1px solid black; display: inline-block; padding: 2px 10px;">KEMBALI KE CRUD</div>	
NAMA	KRITIK & SARAN

**Gambar 3.14**Rancangan halaman kritik & saran  
(Sumber: Pengolahan data penulis, 2017)

Halaman ini sebuah halaman list dari pada user yang telah mengirimkan pesan kritik & saran mereka akan tersusun rapi dari mulai atas sampai bawah.

### 3.5 Lokasi dan Jadwal Penelitian

Penelitian ini dilakukan di PT FOSTER yang beralamt di Jl. Beringin Lot 3 Muka Kuning Batam. Alasan peneliti memilih perusahaan ini sebagai lokasi penelitian adalah:

1. Ketersedian data
2. Mudah mendapatkan data
3. Efisiensi biaya dan waktu

### 3.5.1 Jadwal Penelitian

Penelitian ini dilakukan selama 5 bulan yang dimulai dari bulan September 2016 samapa dengan januari 2017, dengan kegiatan dimulai dari input judul, penyusunan dari BAB I, BAB II, BAB III, BAB IV ,BAB V, laludilanjutkan dengan penyempurnaan skripsi, dan pengumpulan skripsi. Berikut ini adalah table jadwal kegiatan yang dilakukan selama penelitian berlangsung

No	Kegiatan	Tahun 2016/2017																			
		September				Oktober				November				Desember				Januari			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pengajuan judul	■	■	■																	
2	BAB I				■	■	■														
3	BAB II							■	■	■	■	■									
4	BAB III												■	■	■	■	■				
5	BAB IV																■	■	■		
6	BAB V																	■	■		
7	Penyempurnaan skripsi																		■	■	
8	Pengumpulan skripsi																			■	■

**Gambar 3.15**Jadwal penelitian  
(Sumber: Pengolahan data penulis, 2017)