

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN  
PROYEKTOR BERBASIS WEB MENGGUNAKAN  
METODE *FORWARD CHAINING* DAN *BACKWARD  
CHAINING***

**SKRIPSI**



**Oleh:**

**Martinus**

**130210003**

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

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN  
PROYEKTOR BERBASIS WEB MENGGUNAKAN  
METODE *FORWARD CHAINING* DAN *BACKWARD  
CHAINING***

**SKRIPSI**

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



**Oleh:**

**Martinus**

**130210003**

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

## **PERNYATAAN**

### **HALAMAN PERNYATAAN**

Dengan ini menyatakan bahwa:

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

Batam, 14 Februari 2017  
Yang membuat pernyataan,

@materai 6.000,-

Martinus  
NPM: 130210003

**SISTEM PAKAR MENDIAGNOSA KERUSAKAN  
PROYEKTOR BERBASIS WEB MENGGUNAKAN METODE  
*FORWARD CHAINING* DAN *BACKWARD CHAINING***

**HALAMAN PENGESAHAN**

**Oleh  
Martinus  
130210003**

**SKRIPSI**

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

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

**Batam, 14 Februari 2017**

**Alfannisa Annurrullah Fajrin, S.Kom., M.Kom.**

**Pembimbing**

## ABSTRAK

Proyektor merupakan alat media yang hampir seluruh tempat kerja dan wilayah kampus telah menggunakannya. Kerusakan proyektor tidak semua pengguna dapat mendapat solusi yang dimana pengguna langsung dibawa ke teknisi karena kurangnya pengetahuan atas kerusakan alat tersebut dimana akan menghabiskan banyak waktu, biaya dan jarak. Oleh karena itu dibutuhkan untuk membangun suatu aplikasi berbasis sistem pakar yang mempunyai pengetahuan untuk membantu menyelesaikan masalah ini secara cepat sehingga para pengguna tidak perlu datang ke teknisi. Metode yang akan digunakan dalam sistem pakar ini adalah metode *forward chaining* yaitu penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis, dan metode *backward chaining* yaitu penalaran dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut harus dicari fakta-fakta yang ada dalam basis pengetahuan. Sistem ini dibuat berbasis *Web* yang menggunakan bahasa pemrograman yang diterapkan dalam pembangunan aplikasi tersebut adalah bahasa pemrograman *PHP* dan basis data *MySQL* sehingga merupakan sistem yang mudah diakses para pengguna komputer. Hasil Penelitian menunjukkan bahwa aplikasi sistem pakar tersebut dapat membantu para pengguna untuk mencari solusi dan menambah pengetahuan atas kerusakan yang akan timbul pada perangkat keras tersebut. Kesimpulan yang bisa diambil adalah dengan melalui aplikasi ini, pengguna komputer dapat berkonsultasi dengan sistem layaknya berkonsultasi langsung dengan seorang pakar untuk mendeteksi suatu kerusakan yang terjadi pada perangkat keras tersebut serta menemukan solusi atas permasalahan yang dihadapi.

**Kata Kunci:** Sistem Pakar, Proyektor, *MySQL*, *PHP*, *Forward Chaining*, *Backward Chaining*

## **ABSTRACT**

*Projector is a device that almost every workplace and every campus used it. Not everyone do know the solution to the projector breakage where they have to bring it to the technician because the lack of knowledge to the breakage of the device where we lose a lot of time, cost and distance. It is why we need to build a expert system application that have the knowledge or information to help solve the problem so that the users do not need to go to technician. The method that is going to be used in this application will be forward chaining method that is the logic started from the fact of the first to test the truth of the hypothesis and backward chaining method that is the logic of starting from the first hypothesis and to test the truth of the hypothesis must search for the facts that exist in the knowledge base. This system will be made base on Web with the programming language that will be implemented to the application will be PHP programming language and MySQL database will be the database server so that it will be a easy to be access system to users. Result of the research shows that the expert system application can help the users to find the solution and knowledfe to the breakage that happens to the device. The conclusion of the research is that by using the application, user can easily consult with a system that is similar like consulting with an expert that help detect the breakage that happens to the device and find the solution to it.*

**Keyword:** *Expert System, Projector, MySQL, PHP, Forward Chaining, Backward 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 merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

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

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

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Teknik Informatika Universitas Putera Batam.
3. Alfannisa Annurrullah Fajrin, S.Kom., M.Kom. selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam
5. Kepada orang 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, 14 Februari 2017

Penulis



## DAFTAR ISI

<b>HALAMAN PERNYATAAN</b> .....	<b>i</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>ii</b>
<b>ABSTRAK</b> .....	<b>iii</b>
<b>ABSTRACT</b> .....	<b>iv</b>
<b>KATA PENGANTAR</b> .....	<b>v</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xii</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah .....	4
1.3 Pembatasan Masalah.....	4
1.4 Perumusan Masalah.....	5
1.5 Tujuan Penelitian.....	5
1.6 Manfaat Penelitian.....	6
<b>BAB II LANDASAN TEORI</b> .....	<b>7</b>
2.1 Teori Dasar .....	7
2.1.1 Kecerdasan Buatan atau <i>Artificial Intelligence</i> (AI).....	8
2.1.1.1 Logika Fuzzy .....	8
2.1.1.2 Jaringan Syaraf Tiruan.....	9
2.1.1.3 Sistem Pakar atau <i>Expert System</i> .....	10
2.1.2 Tabel Keputusan .....	16
2.1.3 Pohon Keputusan .....	17
2.1.4 UML ( <i>Unified Modeling Language</i> ).....	18
2.1.4.1 <i>Use Case Diagram</i> .....	18
2.1.4.2 <i>Activity Diagram</i> .....	20
2.1.4.3 <i>Sequence Diagram</i> .....	21
2.1.4.4 <i>Class Diagram</i> .....	23
2.2 Variabel .....	24
2.2.1 Proyektor.....	24
2.3 <i>Software</i> Pendukung.....	26
2.3.1 Bahasa Pemrograman <i>PHP</i> .....	27
2.3.2 Aplikasi <i>Adobe Dreamweaver</i> .....	27
2.3.3 <i>MySQL Database</i> .....	28
2.3.4 <i>WampServer</i> .....	29
2.3.5 <i>StarUML</i> .....	30
2.4 Penelitian Terdahulu.....	31
2.5 Kerangka Berpikir .....	32
<b>BAB III METODE PENELITIAN</b> .....	<b>33</b>
3.1 Desain Penelitian .....	33
3.2 Operasional Variabel .....	35
3.2.1 Metode <i>Forward Chaining</i> .....	35

3.2.2	Metode <i>Backward Chaining</i> .....	39
3.3	Teknik Pengumpulan Data .....	41
3.4	Perancangan Sistem .....	42
3.4.1	UML ( <i>Unified Modelling Language</i> ).....	42
3.4.2	Desain Basis Data .....	55
3.4.3	Desain Antarmuka .....	57
3.5	Lokasi Dan Jadwal Penelitian.....	65
<b>BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....</b>		<b>66</b>
4.1	Hasil Penelitian.....	66
4.1.1	Implementasi Antar Muka .....	66
4.2	Pembahasan .....	82
4.2.1	Pengujian Validasi .....	82
<b>BAB V SIMPULAN DAN SARAN.....</b>		<b>83</b>
5.1	Simpulan.....	83
5.2	Saran .....	83
<b>DAFTAR PUSTAKA .....</b>		<b>84</b>
<b>RIWAYAT HIDUP</b>		
<b>SURAT KETERANGAN PENELITIAN</b>		
<b>LAMPIRAN</b>		

## DAFTAR TABEL

<b>Tabel 2.1</b> Contoh Tabel Keputusan.....	16
<b>Tabel 2.2</b> Simbol <i>Use Case Diagram</i> .....	19
<b>Tabel 2.3</b> Simbol <i>Activity Diagram</i> .....	20
<b>Tabel 2.4</b> Simbol <i>Sequence Diagram</i> .....	22
<b>Tabel 2.5</b> Simbol <i>Class Diagram</i> .....	23
<b>Tabel 3.1</b> Indikator Kerusakan Proyektor .....	35
<b>Tabel 3.2</b> Kode Indikator Kerusakan Proyektor.....	36
<b>Tabel 3.3</b> Kode Penyebab Kerusakan Proyektor .....	36
<b>Tabel 3.4</b> Data Solusi Kerusakan Proyektor.....	37
<b>Tabel 3.5</b> Tabel Keputusan Kerusakan Proyektor (FC) .....	37
<b>Tabel 3.6</b> Tabel Keputusan (BC).....	39
<b>Tabel 3.7</b> Tabel Basis Data Admin.....	55
<b>Tabel 3.8</b> Tabel Basis Data Kerusakan/Penyebab .....	56
<b>Tabel 3.9</b> Tabel Basis Data Pertanyaan.....	56
<b>Tabel 3.10</b> Tabel Basis Data Solusi.....	56
<b>Tabel 4.1</b> Hasil Pengujian Validasi .....	82

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Gambar Struktur Sistem Pakar .....	12
<b>Gambar 2.2</b> Contoh Pohon Keputusan.....	17
<b>Gambar 2.3</b> Logo PHP .....	27
<b>Gambar 2.4</b> Tampilan Adobe Dreamweaver CC 2015 .....	28
<b>Gambar 2.5</b> Tampilan Database MySQL.....	29
<b>Gambar 2.6</b> Logo WampServer .....	29
<b>Gambar 2.7</b> Logo StarUML.....	30
<b>Gambar 2.8</b> Kerangka Berpikir Peneliti.....	32
<b>Gambar 3.1</b> Desain Penelitian .....	34
<b>Gambar 3.2</b> Pohon Keputusan (FC).....	38
<b>Gambar 3.3</b> Pohon Keputusan (BC) .....	40
<b>Gambar 3.4</b> <i>Use Case Diagram</i> .....	43
<b>Gambar 3.5</b> <i>Activity Diagram</i> Admin.....	44
<b>Gambar 3.6</b> <i>Activity Diagram</i> Pengguna .....	45
<b>Gambar 3.7</b> <i>Sequence Diagram</i> Login Admin.....	46
<b>Gambar 3.8</b> <i>Sequence Diagram</i> Input Data Kerusakan/Penyebab .....	47
<b>Gambar 3.9</b> <i>Sequence Diagram</i> Ubah Data Kerusakan/Penyebab .....	48
<b>Gambar 3.10</b> <i>Sequence Diagram</i> Input Data Pertanyaan/Penyebab.....	49
<b>Gambar 3.11</b> <i>Sequence Diagram</i> Ubah Data Pertanyaan Kerusakan/Penyebab.....	50
<b>Gambar 3.12</b> <i>Sequence Diagram</i> Input Data Solusi Kerusakan/Penyebab .....	51
<b>Gambar 3.13</b> <i>Sequence Diagram</i> Ubah Data Solusi Kerusakan/Penyebab .....	52
<b>Gambar 3.14</b> <i>Sequence Diagram</i> Ubah Password .....	53
<b>Gambar 3.15</b> <i>Sequence Diagram</i> User Access.....	54
<b>Gambar 3.16</b> <i>Class Diagram</i> .....	55
<b>Gambar 3.22</b> Form Login Aplikasi .....	57
<b>Gambar 3.23</b> Menu Utama Aplikasi .....	57
<b>Gambar 3.24</b> Menu Diagnosa Aplikasi.....	58
<b>Gambar 3.25</b> Menu Tanya Jawab Aplikasi .....	58
<b>Gambar 3.26</b> Menu Hasil Solusi Aplikasi .....	59
<b>Gambar 3.27</b> Menu Data Kerusakan Aplikasi .....	59
<b>Gambar 3.28</b> Menu Tambah Data Kerusakan/Penyebab Aplikasi .....	60
<b>Gambar 3.29</b> Menu Ubah Data Kerusakan/Penyebab Aplikasi.....	60
<b>Gambar 3.30</b> Menu Data Pertanyaan Aplikasi.....	61
<b>Gambar 3.31</b> Menu Tambah Data Pertanyaan Aplikasi.....	61
<b>Gambar 3.32</b> Menu Ubah Data Pertanyaan Aplikasi .....	62
<b>Gambar 3.33</b> Menu Data Solusi Aplikasi .....	62
<b>Gambar 3.34</b> Menu Tambah Data Solusi Aplikasi .....	63
<b>Gambar 3.35</b> Menu Ubah Data Solusi Aplikasi.....	63

<b>Gambar 3.36</b> Menu Ubah <i>Password</i> Aplikasi.....	64
<b>Gambar 4.1</b> Tampilan Halaman Menu Utama .....	66
<b>Gambar 4.2</b> Tampilan Form Login .....	67
<b>Gambar 4.3</b> Tampilan Halaman Menu Diagnosa Penyebab.....	67
<b>Gambar 4.4</b> Tampilan Halaman Menu Tanya Jawab Penyebab .....	68
<b>Gambar 4.5</b> Tampilan Halaman Menu Hasil Solusi Penyebab .....	68
<b>Gambar 4.6</b> Tampilan Halaman Menu Data Penyebab .....	69
<b>Gambar 4.7</b> Tampilan Halaman Tambah Data Penyebab .....	69
<b>Gambar 4.8</b> Tampilan Halaman Ubah Data Penyebab.....	70
<b>Gambar 4.9</b> Tampilan Halaman Data Pertanyaan Penyebab.....	71
<b>Gambar 4.10</b> Tampilan Halaman Tambah Data Pertanyaan Penyebab .....	71
<b>Gambar 4.11</b> Tampilan Halaman Ubah Data Pertanyaan Penyebab .....	72
<b>Gambar 4.12</b> Tampilan Halaman Menu Data Solusi Penyebab .....	72
<b>Gambar 4.13</b> Tampilan Halaman Tambah Data Solusi Penyebab .....	73
<b>Gambar 4.14</b> Tampilan Halaman Ubah Data Solusi Penyebab .....	74
<b>Gambar 4.15</b> Tampilan Halaman Menu Diagnosa Kerusakan .....	74
<b>Gambar 4.16</b> Tampilan Halaman Menu Tanya Jawab Kerusakan .....	75
<b>Gambar 4.17</b> Tampilan Halaman Menu Hasil Solusi Kerusakan.....	75
<b>Gambar 4.18</b> Tampilan Halaman Menu Data Kerusakan .....	76
<b>Gambar 4.19</b> Tampilan Halaman Tambah Data Kerusakan.....	76
<b>Gambar 4.20</b> Tampilan Halaman Ubah Data Kerusakan .....	77
<b>Gambar 4.21</b> Tampilan Halaman Data Pertanyaan Kerusakan .....	77
<b>Gambar 4.22</b> Tampilan Halaman Tambah Data Pertanyaan Kerusakan .....	78
<b>Gambar 4.23</b> Tampilan Halaman Ubah Data Pertanyaan Kerusakan.....	78
<b>Gambar 4.24</b> Tampilan Halaman Menu Data Solusi Kerusakan.....	79
<b>Gambar 4.25</b> Tampilan Halaman Tambah Data Solusi Kerusakan .....	80
<b>Gambar 4.26</b> Tampilan Halaman Ubah Data Solusi Kerusakan .....	80
<b>Gambar 4.27</b> Tampilan Halaman Ubah Password .....	81

## **DAFTAR LAMPIRAN**

**LAMPIRAN I SOURCE CODE**

**LAMPIRAN II SURAT PERNYATAAN PENELITIAN**

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Proyektor adalah perangkat keras yang digunakan untuk menampilkan layar secara visual dengan mengintegrasikan sumber cahaya, sistem optik, elektronik dan display. Proyektor ada berbagai fungsi, yakni sebagai alat presentasi, pemutar video ataupun sebagai media informasi. Proyektor adalah perangkat yang digunakan untuk membuat proyeksi. Proyektor sering dipakai didalam presentasi (Muhlisin, dkk., 2015, p. 3). Proyektor banyak ditemukan diberbagai tempat seperti perusahaan-perusahaan, kampus, dan lain lain. Pengetahuan atas kerusakan proyektor pada para pengguna masih terhitung kurang maka biasanya pengguna langsung diserahkan kepada teknisi dan juga dapat membantu para teknisi *junior* (non-pengalaman) untuk dapat memahami kerusakan-kerusakan proyektor tersebut tanpa harus meminta bantuan kepada teknisi yang berpengalaman saat mereka tidak ada ditempat.

Kecerdasan Buatan atau *Artificial Intelligence* merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industry dan rumah tangga (Budiharto & Suhartono, 2014, p. 2). Dengan adanya kemajuan teknologi tersebut dapat membantu para profesi dewasa dalam bekerja secara efisien. Khususnya pada

profesi di bidang Teknik yang erat berhubungannya dengan dunia Teknologi Informasi, baik level atas ataupun bawah. Sistem Pakar adalah salah satu bagian dari Kecerdasan Buatan yang mengandung pengetahuan dan pengalaman yang dimasukkan oleh banyak pakar ke dalam suatu area pengetahuan tertentu sehingga setiap orang dapat menggunakannya untuk memecahkan berbagai masalah.

Permasalahan yang ditangani oleh seorang pakar bukan mengandalkan algoritma namun terkadang ada permasalahan yang sulit dipahami. Oleh karena itu sistem pakar dibangun bukan berdasarkan algoritma tertentu tetapi berbasis pengetahuan dan aturan. Berdasarkan alur pelacakan solusi, Sistem pakar memiliki metode *forward chaining* dan *backward chaining*. Kedua metode yang disebutkan akan diterapkan oleh peneliti dalam membangun sebuah sistem pakar.

Penelitian yang akan dilakukan adalah terhadap kerusakan proyektor. Seperti yang dimaksudkan tadi, pengetahuan terhadap kerusakan proyektor masih terhitung kurang. Maka berdasarkan permasalahan diatas, dibutuhkan suatu aplikasi sistem pakar untuk mendeteksi terdahulu permasalahan yang ada didalam alat proyektor tersebut supaya para pengguna dapat melakukan reparasi secara tersendiri dengan mudah ataupun dapat membantu para teknisi datang memperbaiki alat tersebut sudah dengan persiapan alat yang sesuai dengan kerusakan proyektor tersebut.

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



Manoor Hara Pardede dan Budi Serasi Ginting mengenai Perancangan Sistem Pakar Deteksi Kerusakan Printer Canon Berbasis *Web* membahas mengenai suatu program sistem pakar untuk mendiagnosa kerusakan printer dengan menggunakan metode *Backward Chaining*, bahasa pemrograman *PHP* dan *MySQL* sebagai *database*. Penelitian Purwanto dan Bramuditya Adi Putra mengenai Sistem Pakar Diagnosa Kerusakan pada Televisi LED dengan menggunakan Metode *Forward Chaining* membahas mengenai sistem pakar untuk mendiagnosa kerusakan pada televisi LED dengan menggunakan metode *Forward Chaining* dan dibangun dengan bahasa pemrograman *Visual Basic 6.0* dan *Microsoft Access* sebagai *database*.

Pada penelitian ini, aplikasi sistem pakar yang dapat menghasilkan pengetahuan untuk menangani masalah yang timbul dari kerusakan komponen proyektor harus mampu menangani dalam perbaikan kerusakan proyektor tersebut. Hal inilah yang mendorong peneliti untuk melakukan penelitian dengan judul **“SISTEM PAKAR MENDIAGNOSA KERUSAKAN PROYEKTOR BERBASIS WEB MENGGUNAKAN METODE FORWARD CHAINING DAN BACKWARD CHAINING”**

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang masalah yang ada pada latar belakang, dapat diidentifikasi masalah sebagai berikut:

1. Teknisi memiliki pengetahuan yang kurang mengenai jenis kerusakan pada proyektor.
2. Teknisi memiliki pengetahuan yang kurang mengenai solusi terhadap kerusakan pada proyektor.

## 1.3 Pembatasan Masalah

Adapun batasan masalah pada tugas akhir ini adalah:

1. Peneliti hanya akan menganalisis kerusakan alat proyektor khususnya pada proyektor merk Infocus IN2124.
2. Sistem pakar tersebut dibuat dengan menggunakan metode *Forward Chaining* dan *Backward Chaining*.
3. Sistem pakar dirancang dengan menggunakan bahasa pemrograman *PHP* dengan *database MySQL Server*.
4. Data akan diambil dengan melakukan wawancara dengan teknisi dan mengambil data dari referensi buku, jurnal dan internet.

## **1.4 Perumusan Masalah**

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

1. Bagaimana implementasi sistem pakar mendiagnosa kerusakan proyektor dalam memberikan solusi yang berguna untuk para penggunanya?

## **1.5 Tujuan Penelitian**

Berdasarkan rumusan masalah diatas maka tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Untuk menghasilkan suatu sistem pakar yang efektif dan dapat digunakan dalam mendiagnosa kerusakan pada proyektor sehingga pengguna dapat mengetahui tindakan yang harus dilakukan serta dapat melakukan pencegahan secara dini.

## 1.6 Manfaat Penelitian

Berdasarkan uraian di atas, manfaat penelitian di bagi menjadi dua sub poin terpenting yang diberikan dari hasil perancangan sistem yaitu:

a. Aspek Teoritis

Untuk mengetahui secara dalam tentang sistem pakar dengan metode *Forward Chaining* dan *Backward Chaining* dan cara sistem pakar dengan metode *Forward Chaining* dan *Backward Chaining* mendiagnosa kerusakan proyektor.

b. Aspek Praktis

Hasil penelitian bisa dijadikan media informasi untuk masyarakat dan menambah wawasan tentang kerusakan proyektor serta solusi untuk dapat memperbaikinya.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Teori Dasar**

Landasan teori perlu ditegakkan agar penelitian mempunyai dasar yang kokoh, dan bukan sekedar perbuatan coba-coba (*trial and error*). Adanya landasan teori merupakan ciri bahwa penelitian merupakan cara ilmiah untuk mendapatkan data. Kerlinger mengatakan bahwa teori adalah seperangkat konstruk (konsep), definisi, dan proposisi yang berfungsi untuk melihat fenomena secara sistematis melalui spesifikasi hubungan antarvariabel sehingga dapat berguna untuk menjelaskan dan meramalkan fenomena. Dengan kata lain, teori adalah generalisasi atau kumpulan generalisasi yang dapat digunakan untuk menjelaskan berbagai fenomena secara sistematis. (Sudaryono, 2015, p. 13)

Teori dasar ini bertujuan untuk mengkaji tentang teori-teori yang berhubungan dengan penelitian yang sedang dikerjakan. Teori adalah serangkaian bagian atau variabel, definisi, dan dalil yang saling berhubungan yang menghadirkan sebuah pandangan sistematis mengenai fenomena dengan menentukan hubungan antar variabel, dengan menentukan hubungan antar variabel, dengan maksud menjelaskan fenomena alamiah.

### 2.1.1 Kecerdasan Buatan atau *Artificial Intelligence* (AI)

Kecerdasan Buatan atau *Artificial Intelligence* (AI) merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industry dan rumah tangga. AI mencakup bidang yang cukup besar. Mulai dari yang paling umum hingga yang khusus. Dari *learning* atau *perception* hingga pada permainan catur, pembuktian teori matematika, menulis puisi, mengemudi mobil, dan melakukan diagnosis penyakit. AI relevan dengan berbagai macam *task* kecerdasan, AI merupakan sebuah ilmu yang universal (Budiharto & Suhartono, 2014, p. 2). Ada beberapa bidang ilmu AI yaitu:

#### 2.1.1.1 Logika Fuzzy

Logika fuzzy merupakan salah satu cabang dari bidang *soft computing*. Logika fuzzy merupakan suatu teori himpunan logika yang dikembangkan untuk mengatasi konsep nilai yang terdapat diantara kebenaran (*true*) dan kesalahan (*false*) (Irwansyah & Faisal, 2015, p. 31).

Ada 3 metode dalam logika *fuzzy*, yaitu:

### 1. Metode Mamdani

Sering dikenal sebagai metode *max-min*. Untuk mendapatkan *output* diperlukan beberapa tahapan, yaitu pembentukan himpunan *fuzzy*, aplikasi fungsi implikasi (aturan), komposisi aturan (Pusadan, 2014, p. 14).

### 2. Metode Sugeno

Metode Sugeno adalah penalaran dengan metode *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear (Rofiq, 2013, p. 6).

### 3. Metode Tsukamoto

Pada metode ini, setiap aturan direpresentasikan menggunakan himpunan-himpunan *fuzzy*, dengan fungsi keanggotaan yang monoton. Untuk menentukan nilai *output crisp* atau hasil yang tegas, dicari dengan cara mengubah input menjadi suatu bilangan pada domain himpunan *fuzzy* tersebut (Sholihin, dkk. 2013, p. 501).

#### **2.1.1.2 Jaringan Syaraf Tiruan**

Jaringan syaraf tiruan (JST) sistem pemrosesan informasi yang memiliki karakteristik serupa dengan jaringan syaraf biologis dan terdiri dari sejumlah besar elemen pemrosesan sederhana yang disebut neuron, unit, sel, atau node. Ada 3 hal yang diadopsi dari sistem syaraf otak manusia oleh JST adalah Neuron, Topologi, Toleransi Kesalahan (*Fault Tolerant*) (Irwansyah & Faisal, 2015, p. 42). Menurut Russel dan Norvig (2003), JST memiliki proses belajar yang berbeda dengan

metode yang lainnya, sehingga *Neural Network* memiliki struktur yang unik dimana metode pelatihannya dibagi ke dalam dua kelompok, yakni sebagai berikut:

1. *Feed-Forward Neural Network* (FFNN)

FFNN adalah jaringan syaraf tiruan dimana hubungan antara sinyal informasinya bergerak hanya satu arah saja dalam mengasosiasikan *input* dengan *output* yang ekstensif digunakan dalam pengenalan pola.

2. *Feed-Back Neural Network* (FBNN)

FBNN merupakan jaringan syaraf tiruan yang memiliki sinyal informasi yang bergerak dari kedua arah pada *layer* jarignannya.

### **2.1.1.3 Sistem Pakar atau *Expert System***

Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar. Pakar yang dimaksud di sini adalah orang yang mmepunyai keahlian khusus yang dapat menyelesaikan masalah yang tidak dapat diselesaikan oleh orang awam (Kusrini, 2008, p. 3).

Sistem pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek kemampuan pengambilan keputusan seorang pakar. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah (Rosnelly, 2012, p. 3).



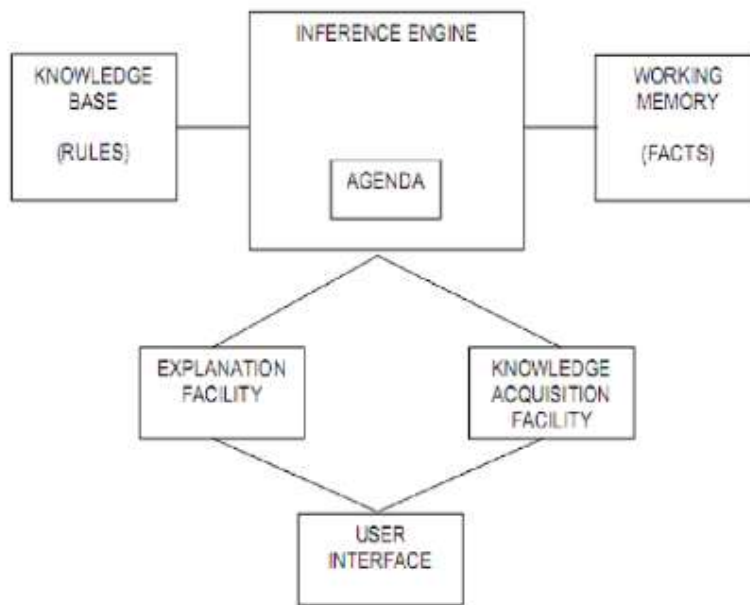
Pakar atau ahli didefinisikan sebagai seorang yang memiliki pengetahuan atau keahlian khusus yang tidak dimiliki oleh kebanyakan orang. Seorang pakar dapat memecahkan masalah yang tidak mampu dipecahkan kebanyakan orang. Dengan kata lain, dapat memecahkan suatu masalah dengan lebih efisien namun bukan berarti lebih murah. Pengetahuan yang dimuat ke dalam sistem pakar dapat berasal dari seorang pakar atau pun pengetahuan yang berasal dari buku, jurnal, majalah, dan dokumentasi yang dipublikasi lainnya, serta orang yang memiliki pengetahuan meskipun bukan ahli. Istilah sistem pakar sering disinonimkan dengan sistem berbasis pengetahuan.

#### **A. Kelebihan Sistem Pakar**

Sistem pakar memiliki beberapa fitur menarik yang merupakan kelebihannya, seperti meningkatkan ketersediaan, mengurangi biaya, permanen, keahlian multipel, meningkatkan kehandalan, penjelasan, respon yang cepat, stabil, pembimbing pintar dan basis data cerdas.

## B. Struktur Sistem Pakar

Adapun struktur sistem pakar dapat dilihat pada gambar berikut.



**Gambar 2.1** Gambar Struktur Sistem Pakar  
**Sumber:** Rosnelly (2012: 13)

Komponen yang terdapat dalam struktur sistem pakar ini adalah *knowledge base (rules)*, *inference engine*, *working memory*, *explanation facility*, *knowledge acquisition facility*, *user interface*.

### 1. *Knowledge Base* (Base Pengetahuan)

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi dan dua elemen dasar, yaitu fakta dan aturan. Fakta merupakan informasi tentang objek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi

tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui. Pada struktur sistem pakar diatas, *knowledge base* disini untuk menyimpan pengetahuan dari pakar berupa *rule*/aturan.

## 2. *Inference Engine* (Mesin Inferensi)

Mesin inferensi merupakan otak dari sebuah sistem pakar dan dikenal juga dengan sebutan *control structure* atau *rule interpreter*. Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi disini adalah processor pada sistem pakar yang mencocokkan bagian kondisi dari rule yang tersimpan didalam *knowledge base* dengan fakta yang tersimpan di *working memory*.

## 3. *Working Memory*

Berguna untuk menyimpan fakta yang dihasilkan oleh *inference engine* dengan penambahan parameter berupa derajat kepercayaan atau dapat juga dikatakan sebagai global *database* dari fakta yang digunakan oleh *rule-rule* yang ada.

## 4. *Explanation Facility*

Menyediakan kebenaran dari solusi yang dihasilkan kepada user (*reasoning chain*).

5. *Knowledge Acquisition Facility*

Meliputi proses pengumpulan, pemindahan dan perubahan dari kemampuan pemecahan masalah seorang pakar atau sumber pengetahuan terdokumentasi ke program computer, yang bertujuan untuk memperbaiki atau mengembangkan basis pengetahuan.

6. *User Interface*

Mekanisme untuk memberi kesepakatan kepada *user* dan sistem pakar untuk berkomunikasi. Antar muka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

**C. *Inference Engine* atau Mesin Inferensi**

Inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Inferensi adalah konklusi logis (*logical conclusion*) atau implikasi berdasarkan informasi yang tersedia. Dalam sistem pakar proses inferensi dilakukan dalam suatu modul yang disebut *Inference Engine* (Mesin Inferensi) (Kusrini, 2008, p. 8).

Ada dua metode inferensi yang penting dalam sistem pakar yaitu:

### 1. Runut Maju (*Forward Chaining*)

Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil (Wilson, 1998).

Metode inferensi runut maju cocok digunakan untuk menangani masalah pengendalian (*controlling*) dan peramalan (*prognosis*) (Giarattano dan Riley, 1994).

### 2. Runut Balik (*Backward Chaining*)

Runut balik merupakan metode penalaran kebalikan dari runut maju. Dalam runut balik penalaran dimulai dengan tujuan kemudian merunut balik ke jalur yang akan mengarahkan ke tujuan tersebut (Giarattano dan Riley, 1994). Runut balik disebut juga sebagai *goal-driven reasoning*, merupakan cara yang efisien untuk memecahkan masalah yang dimodelkan sebagai masalah pemilihan terstruktur.

Tujuan inferensi adalah mengambil pilihan terbaik dari banyak kemungkinan. Metode inferensi runut balik ini cocok digunakan untuk memecahkan masalah diagnosi (Schnupp, 1989).

#### D. Karakteristik Sistem Pakar

Sistem pakar umumnya dirancang untuk memenuhi beberapa karakteristik umum, yaitu kinerja sangat baik, waktu respon yang baik, dapat diandalkan, dapat dipahami dan fleksibel.

##### 2.1.2 Tabel Keputusan

Dalam proses pengambilan keputusan, semua informasi yang diperlukan disusun dalam bentuk ringkasan hasil yang disebut sebagai tabel hasil atau tabel keputusan. Tabel ini merupakan suatu matriks yang terdiri dari baris yang menunjukkan berbagai alternatif pilihan/keputusan dan kolom yang menunjukkan nilai harapan untuk setiap alternatif pilihan/keputusan pada berbagai keadaan atau situasi yang mungkin terjadi. (Herjanto & Herfan, 2008, p. 26).

**Tabel 2.1** Contoh Tabel Keputusan

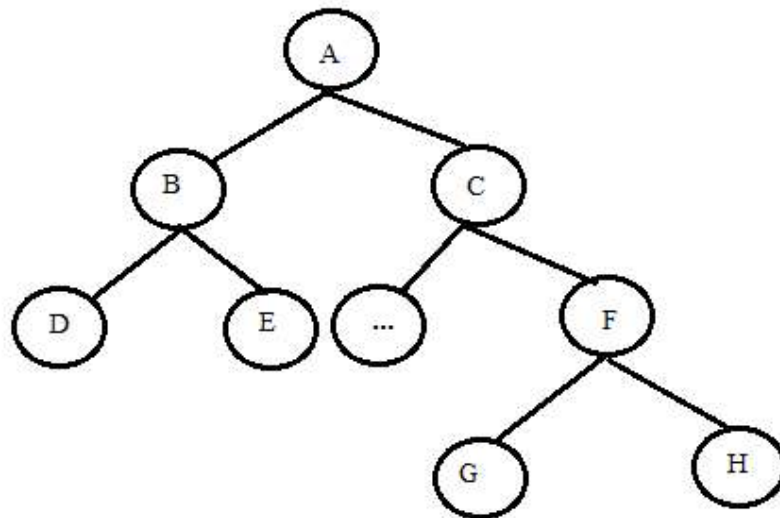
<b>Hipotesa</b> <b>Indikator</b>	<b>Hipotesa1</b>	<b>Hipotesa2</b>	<b>Hipotesa3</b>	<b>Hipotesa4</b>
<b>Indikator A</b>	Ya	Ya	Tidak	Ya
<b>Indikator B</b>	Ya	Tidak	Ya	Tidak
<b>Indikator C</b>	Ya	Tidak	Ya	Tidak
<b>Indikator D</b>	Tidak	Ya	Tidak	Ya

**Sumber:** Data Penelitian (2017)

### 2.1.3 Pohon Keputusan

Sebuah pohon keputusan adalah sebuah struktur yang dapat digunakan untuk membagi kumpulan data yang besar menjadi himpunan-himpunan *record* yang lebih kecil dengan menerapkan serangkaian aturan keputusan. Dengan masing-masing rangkaian pembagian, anggota himpunan hasil menjadi mirip satu dengan yang lain (Berry & Linoff, 2004 dalam buku (Kusrini & Luthfi, 2009, p. 14)).

Sebuah pohon keputusan mungkin dibangun dengan saksama secara manual atau dapat tumbuh secara otomatis dengan menerapkan salah satu atau beberapa algoritma pohon keputusan untuk memodelkan himpunan data yang belum terklasifikasi.



**Gambar 2.2** Contoh Pohon Keputusan  
**Sumber:** Data Penelitian (2017)

#### 2.1.4 UML (*Unified Modeling Language*)

Salah satu pemodelan yang saat ini paling banyak digunakan adalah UML. UML (*Unified Modeling Language*) adalah salah standar bahasa yang banyak digunakan di dunia industry untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa & Shalahuddin, 2011, p. 113)

Berikut adalah beberapa jenis diagram UML:

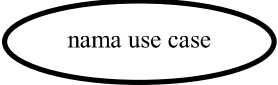


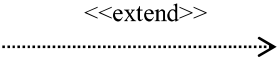
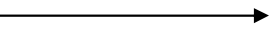
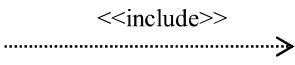
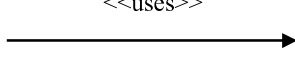
##### 2.1.4.1 *Use Case Diagram*

*Use case diagram* merupakan pemodelan untuk kelakuan 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 di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Berikut adalah simbol-simbol yang ada pada diagram *use case*:



Tabel 2.2 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use Case</i></p>  <p>nama use case</p>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
<p>Aktor / <i>actor</i></p>  <p>nama aktor</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang; tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
<p>Asosiasi / <i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> yang memiliki interaksi dengan aktor
<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; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan
<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 lainnya
<p>Menggunakan / <i>include</i> / <i>uses</i></p>  	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.






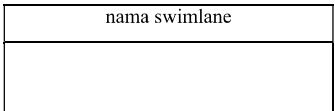

Sumber: Rosa, A. & Shalahuddin, M. (2011)

### 2.1.4.2 Activity Diagram

*Activity diagram* menggambarkan 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.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

**Tabel 2.3** Simbol Activity Diagram

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
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
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi



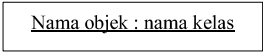

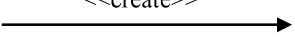
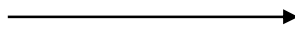
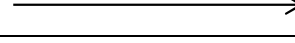
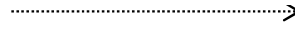
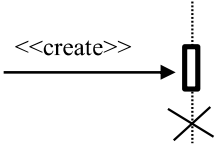
**Sumber:** Rosa, A. & Shalahuddin, M. (2011)

#### 2.1.4.3 *Sequence Diagram*

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

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.4 Simbol *Sequence Diagram*

Simbol	Deskripsi
<p>Aktor / <i>actor</i></p>  <p>nama aktor</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang; tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek
<p>Objek</p> 	Menyatakan objek yang berinteraksi pesan
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi dengan pesan
<p>Pesan tipe <i>create</i></p> 	Objek yang lain, arah panah mengarah pada objek yang dibuat
<p>Pesan tipe <i>call</i></p> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
<p>Pesan tipe <i>send</i></p> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
<p>Pesan tipe <i>return</i></p> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>

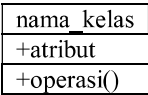


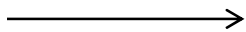
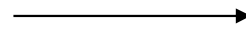


Sumber: Rosa, A. & Shalahuddin, M. (2011)

#### 2.1.4.4 Class Diagram

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

**Tabel 2.5** Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian ( <i>whole part</i> )

**Sumber:** Rosa, A. & Shalahuddin, M. (2011)

## 2.2 Variabel

Variabel penelitian juga menjadi suatu bagian penting dalam penelitian. Kemampuan peneliti untuk memahami variabel penelitian sangat tergantung pada penguasaan konsep tentang penelitian terutama variabel penelitian. Selain itu, pengalaman peneliti dalam melaksanakan penelitian maupun menyusun proposal penelitian juga dapat menambah pemahaman tentang kemampuan mengidentifikasi variabel penelitian. Variabel adalah sebuah konsep yang dioperasionalkan. Lebih tepatnya, operasional properti dari sebuah objek agar dapat dioperasionalkan, diaplikasikan, dan menjadi properti dari objek (Swarjana, 2015, p. 42).

Variabel yang akan digunakan oleh peneliti dalam penelitiannya ada sebagai berikut:

### 2.2.1 Proyektor

Proyektor adalah alat yang digunakan untuk memproyeksikan tampilan dari komputer atau alat lain agar diperoleh gambar yang lebih besar. Jenis-jenis proyektor di antaranya proyektor CRT, proyektor LCD, proyektor DLP, proyektor *slide*, proyektor *film*, proyektor *overhead*, dan lain-lain (Priyatno, 2012, p. 16).

Fungsi-fungsi dari sebuah alat proyektor ada sebagai berikut:

**a. Sebagai Alat Presentasi**

Projector dapat membuat sebuah presentasi menjadi lebih hidup, karena dengan tampilan gambar atau tulisan itu kita dapat memberikan presentasi yang lebih dinamis dan atraktif.

**b. Sebagai Pemutar Video (*Home Theater*)**

Dengan Proyektor kita dapat menikmati bioskop di dalam rumah. Ini dikarenakan proses tampilan yang terjadi di bioskop bisa kita tampilkan di rumah, yaitu dengan proyeksi.

**c. Sebagai Media Informasi**

Karena Proyektor dapat menampilkan tampilan dengan layar besar, maka proyektor sangat efektif untuk dijadikan sebagai media informasi.

Proyektor yang diteliti oleh peneliti adalah proyektor Infocus IN2124 dimana jenis proyektor DLP (*Digital Light Processing*). Proyektor tersebut telah banyak digunakan oleh umum baik di perorangan maupun di perkantoran yang mana telah dijadikan alasan mengapa peneliti mengambil proyektor jenis tersebut. Pemrosesan Cahaya Digital adalah sebuah teknologi yang digunakan dalam proyektor dan televisi proyeksi. DLP awalnya dikembangkan oleh Texas Instruments, dan mereka tetap pembuat satu-satunya teknologi ini, meskipun banyak produk pasar berlisensi menggunakan chipset mereka (Utami, 2007, p. 73).

Dalam proyektor DLP, gambar diciptakan oleh kaca kecil mikroskopis yang disusun dalam sebuah *matrix* di atas chip semikonduktor, dikenal sebagai *Digital*

*Micromirror Device* (DMD). Setiap kaca mewakili satu *pixel* dalam gambar yang diproyeksikan. Jumlah kaca sama dengan resolusi gambar yang diproyeksikan: 800x600, 1024x768, dan 1280x720. *Matrix* adalah beberapa ukuran DMD yang umum. Kaca-kaca ini dapat diubah posisinya dengan cepat untuk merefleksikan cahaya melalui lensa atau ke sebuah *heatsink* (pembuangan cahaya dalam terminologi Barco).

Penyusunan posisi kaca-kaca ini pergantian antara 'on' dan 'off' secara cepat membuat DMD mengatur intensitas cahaya yang direfleksikan melalui lensa, menciptakan efek abu-abu bertingkat sebagai tambahan untuk putih (kaca dalam posisi 'on'), dan hitam (kaca dalam posisi 'off'). Ada dua metode primer dimana sistem proyeksi DLP menciptakan sebuah gambar berwarna, yaitu dengan menggunakan proyektor DLP chip-tunggal dan proyektor tiga-chip.

### **2.3 Software Pendukung**

Program pendukung yang digunakan untuk menjalankan sistem aplikasi tersebut adalah:



### 2.3.1 Bahasa Pemrograman PHP

PHP merupakan bahasa pemrograman berbasis web yang dibuat secara khusus untuk membangun aplikasi berbasis web. Selain tersedia secara gratis, PHP juga mudah dipelajari oleh siapapun. Akan lebih mudah lagi, jika kita sudah pernah mempelajari Bahasa C atau C++. Banyak perintah PHP yang diturunkan atau mengadopsi perintah-perintah di Bahasa C (Solichin, 2016, p. 23).



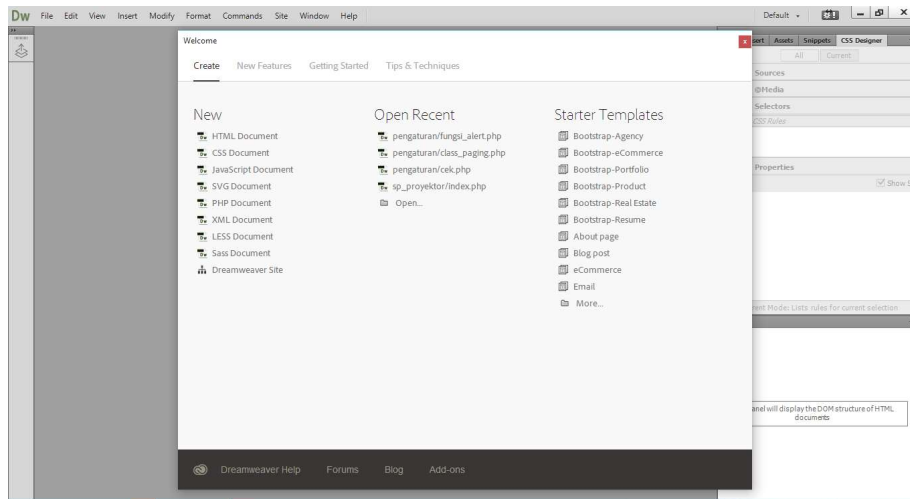
**Gambar 2.3** Logo PHP

**Sumber:** <http://php.net/>

### 2.3.2 Aplikasi Adobe Dreamweaver

Adobe Dreamweaver merupakan salah satu program aplikasi yang digunakan untuk membangun sebuah *website*, baik secara grafis maupun dengan menuliskan kode sumber secara langsung (Wahana Komputer n.d., p. 2).

Adobe Dreamweaver memudahkan pengembang website untuk mengelola halaman-halaman *website* dan asset-asetnya, baik gambar(*image*), animasi *flash*, video, suara dan lain sebagainya.

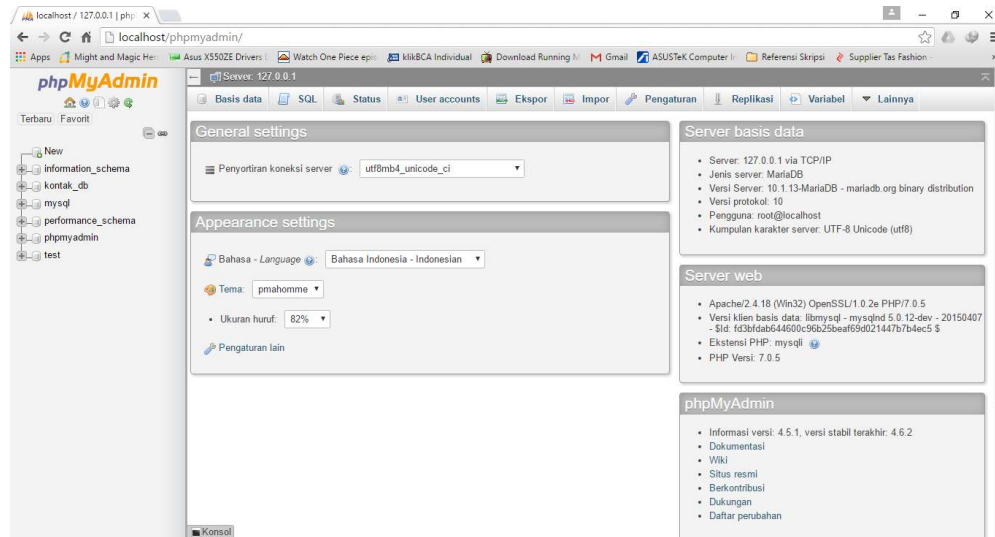


**Gambar 2.4** Tampilan Adobe Dreamweaver CC 2015  
**Sumber:** Data Penelitian (2017)

### 2.3.3 MySQL Database

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*database management system*), *database* ini *multithread*, *multi-user* (Miftakhul Huda, 2010, p. 181).

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.



**Gambar 2.5** Tampilan Database MySQL  
**Sumber:** Data Penelitian (201)

### 2.3.4 WampServer

WampServer adalah singkatan dari Windows, Apache, MySQL dan PHP. Wampserver adalah aplikasi yang menggabungkan antara Apache, Mysql, dan PHP. Sehingga, Wampserver akan mencukupi semua persyaratan yang diminta oleh Joomla. Dengan demikian, tidak perlu lagi meng-*install* masing-masing aplikasi Apache, Mysql, dan PHP (Meissa, 2009, p. 17).



**Gambar 2.6** Logo *WampServer*  
**Sumber:** <http://www.wampserver.com/en/>

### 2.3.5 StarUML

StarUML adalah *platform* pemodelan perangkat lunak yang mendukung UML (*Unified Modeling Language*). StarUML yang berbasiskan pada UML versi 1.4, menyediakan sebelas jenis *Diagram* yang berbeda, dan mendukung notasi UML 2.0 StarUML juga secara aktif mendukung pendekatan MDA (*Model Driven Architercture*) dengan mendukung konsep UML. StarUML unggul dalam hal kustomisasi lingkungan kerja pengguna, dan memiliki ekstensibilitas tinggi pada fungsionalitasnya. (Triandini & Suardika, 2012, p. 1)



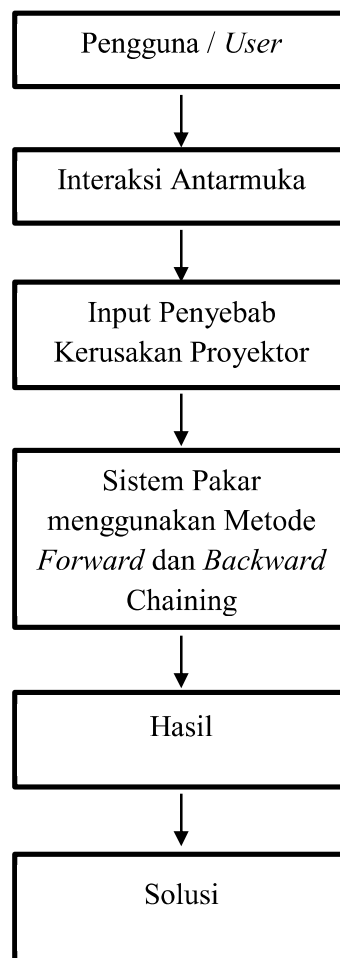
**Gambar 2.7** Logo *StarUML*  
**Sumber:** <http://staruml.io/>

## 2.4 Penelitian Terdahulu

<b>Penulis</b>	<b>Judul</b>	<b>Metode</b>	<b>Hasil</b>
Manaor dan Serasi (2013)	Perancangan Sistem Pakar Deteksi Kerusakan Printer Canon Berbasis <i>Web</i>	<i>Backward Chaining</i>	Kesimpulan dari Hasil Penelitian adalah metode tersebut layak digunakan.
Purwanto dan Putra (2016)	Sistem Pakar Diagnosa Kerusakan Pada Televisi LED Dengan Menggunakan Metode <i>Forward Chaining</i>	<i>Forward Chaining</i>	Kesimpulan dari Hasil Penelitian adalah metode tersebut layak digunakan.
Hartanto dan Herlawati (2015)	Sistem Pakar Pendeteksian Permasalahan Komputer Pada PT. Pasifik Satelit Nusantara Cikarang	<i>Forward Chaining</i>	Kesimpulan dari Hasil Penelitian adalah metode tersebut layak digunakan.
Saputra, dkk (2016)	Aplikasi Analisa Masalah Mesin Motor Bebek Menggunakan Metode <i>Backward Chaining</i>	<i>Backward Chaining</i>	Kesimpulan dari Hasil Penelitian adalah metode tersebut layak digunakan.
Iriani (2015)	Penerapan Metode <i>Backward Chaining</i> pada Sistem Pakar Diagnosa Penyakit Tulang Manusia	<i>Backward Chaining</i>	Kesimpulan dari Hasil Penelitian adalah metode tersebut layak digunakan.

## 2.5 Kerangka Berpikir

Penjelasan sementara terhadap suatu gejala yang menjadi objek permasalahan peneliti. Kerangka berpikir ini disusun dengan berdasarkan pada tinjauan pustaka dan hasil penelitian yang relevan atau terkait.



**Gambar 2.8** Kerangka Berpikir Peneliti  
**Sumber:** Data Penelitian (2017)

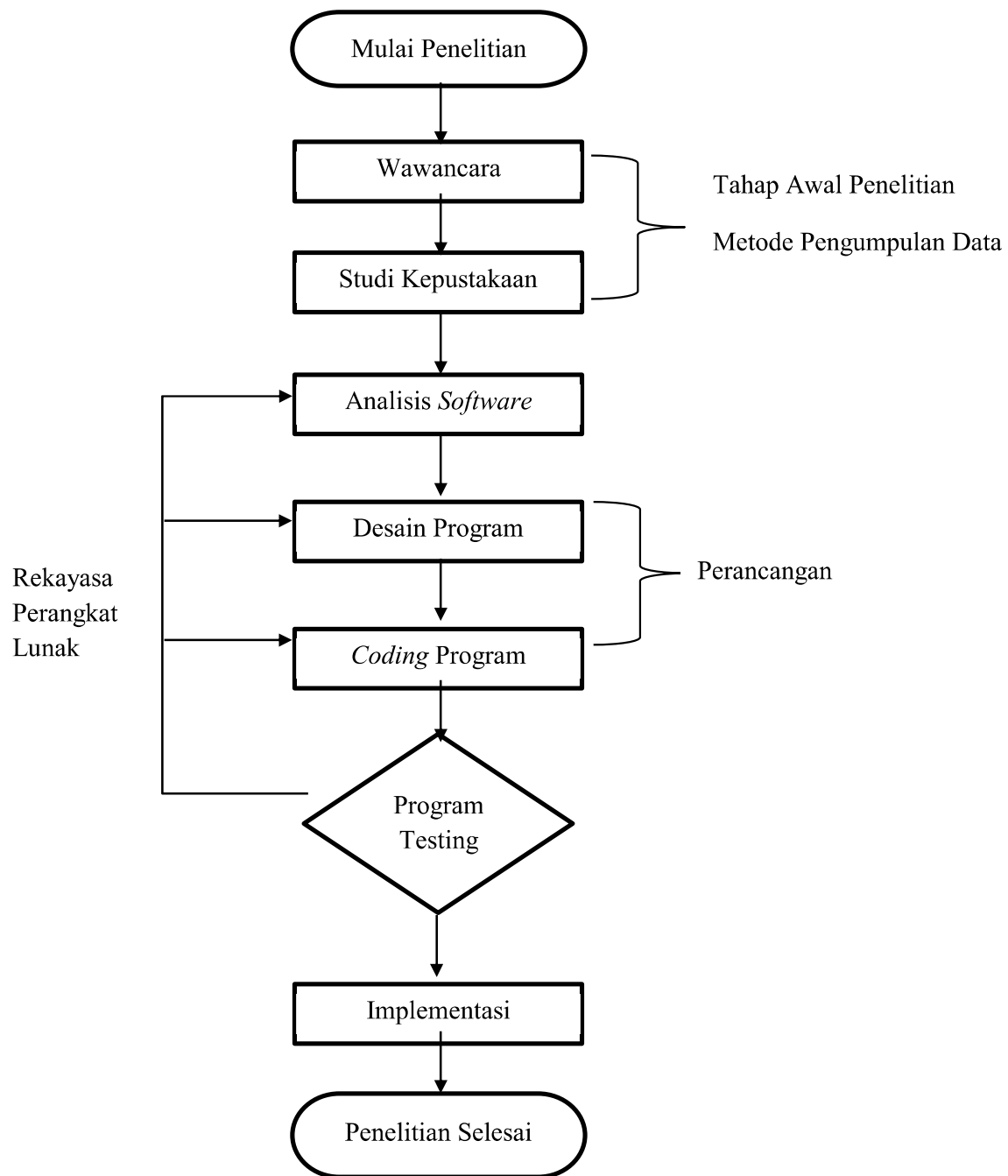
## **BAB III METODE PENELITIAN**

### **3.1 Desain Penelitian**

Desain penelitian ini bertujuan untuk melaksanakan penelitian sehingga diperoleh suatu logika dalam pengujian untuk memperoleh hasil penelitian dan dalam membuat kesimpulan sesuai dengan fokus penelitian.

Desain penelitian adalah semua proses yang diperlukan dalam perencanaan dan pelaksanaan penelitian. Desain penelitian adalah kerangka atau cetak biru dalam melaksanakan suatu proyek riset. Desain penelitian adalah penjelasan mengenai berbagai komponen yang akan digunakan peneliti dan kegiatan yang akan dilakukan selama proses penelitian (Martono, dkk., 2010, p. 131).

Untuk menyelesaikan masalah didalam pengembangan perangkat lunak, penulis memilih menggunakan metode pengembangan sistem model sekuensial linier. Model ini sering juga disebut dengan siklus kehidupan klasik atau model air terjun.



**Gambar 3.1** Desain Penelitian  
**Sumber:** Data Penelitian (2017)



## 3.2 Operasional Variabel

### 3.2.1 Metode *Forward Chaining*

Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil (Wilson, 1998).

Metode inferensi runut maju cocok digunakan untuk menangani masalah pengendalian (*controlling*) dan peramalan (*prognosis*) (Giarattano dan Riley, 1994). Berikut adalah operasional variabel menggunakan metode inferensi *Forward Chaining*.

**Tabel 3.1** Indikator Kerusakan Proyektor

Variabel	Indikator
Kerusakan Komponen Proyektor	1. Masalah pada Lampu
	2. Masalah pada <i>Power Supply</i>
	3. Masalah pada <i>DMD Chip</i>

**Sumber:** Data Penelitian (2017)

**Tabel 3.2** Kode Indikator Kerusakan Proyektor

<b>Kode</b>	<b>Indikator</b>	<b>Penyebab</b>
FK01	Masalah pada Lampu	<ul style="list-style-type: none"> <li>• Proyektor tidak bekerja ketika dinyalakan,</li> <li>• Ujung bola lampu pecah,</li> <li>• Kipas menyala kencang saat lampu alarm menyala,</li> <li>• Lampu LED berkedip ketika dinyalakan</li> </ul>
FK02	Masalah pada <i>Power Supply</i>	<ul style="list-style-type: none"> <li>• Mati Total, tidak bisa dinyalakan sama sekali,</li> <li>• <i>Premature Shutdown</i> (hidup sebentar terus mati),</li> <li>• Nyala namun lampu tidak mau nyala</li> </ul>
FK03	Masalah pada <i>DMD Chip</i>	<ul style="list-style-type: none"> <li>• Tampilan layar muncul bintik-bintik,</li> <li>• Tampilan layar muncul garis vertikal</li> </ul>

**Sumber:** Data Penelitian (2017)

**Tabel 3.3** Kode Penyebab Kerusakan Proyektor

<b>Kode Penyebab</b>	<b>Nama Penyebab</b>
FP01	Proyektor tidak bekerja ketika dinyalakan
FP02	Ujung bola lampu pecah
FP03	Kipas menyala kencang saat lampu alarm menyala
FP04	Lampu LED berkedip ketika dinyalakan
FP05	Mati Total (tidak bisa dinyalakan sama sekali)
FP06	<i>Premature Shutdown</i> (hidup sebentar terus mati)
FP07	Nyala namun lampu tidak mau nyala
FP08	Tampilan layar muncul bintik-bintik
FP09	Tampilan layar muncul garis vertikal

**Sumber:** Data Penelitian (2017)

**Tabel 3.4** Data Solusi Kerusakan Proyektor

Indikator	Penyebab	Solusi
Masalah pada Lampu	<ul style="list-style-type: none"> <li>• Proyektor tidak bekerja ketika dinyalakan,</li> <li>• Ujung bola lampu pecah,</li> <li>• Kipas menyala kencang saat lampu alarm menyala,</li> <li>• Lampu LED berkedip ketika dinyalakan</li> </ul>	Masa Pemakaian Lampu telah kadaluarsa, wajib digantikan yang baru
Masalah pada <i>Power Supply</i>	<ul style="list-style-type: none"> <li>• Mati Total (tidak bisa dinyalakan sama sekali),</li> <li>• <i>Premature Shutdown</i> (hidup sebentar terus mati),</li> <li>• Nyala namun lampu tidak mau nyala</li> </ul>	Kemungkinan <i>IC Regulator</i> rusak, bisa ganti <i>IC Regulator</i> kalau masih bermasalah butuh digantikan power supply
Masalah pada <i>DMD Chip</i>	<ul style="list-style-type: none"> <li>• Tampilan layar muncul bintik-bintik,</li> <li>• Tampilan layar muncul garis vertikal</li> </ul>	Lakukan Servis <i>Cleaning Overhaul</i>

Sumber: Data Penelitian (2017)

**Tabel 3.5** Tabel Keputusan Kerusakan Proyektor (FC)

Kode Kerusakan	Kode Penyebab								
	FP01	FP02	FP03	FP04	FP05	FP06	FP07	FP08	FP09
<b>FK01</b>	√	√	√	√					
<b>FK02</b>					√	√	√		
<b>FK03</b>								√	√

Sumber: Data Penelitian (2017)

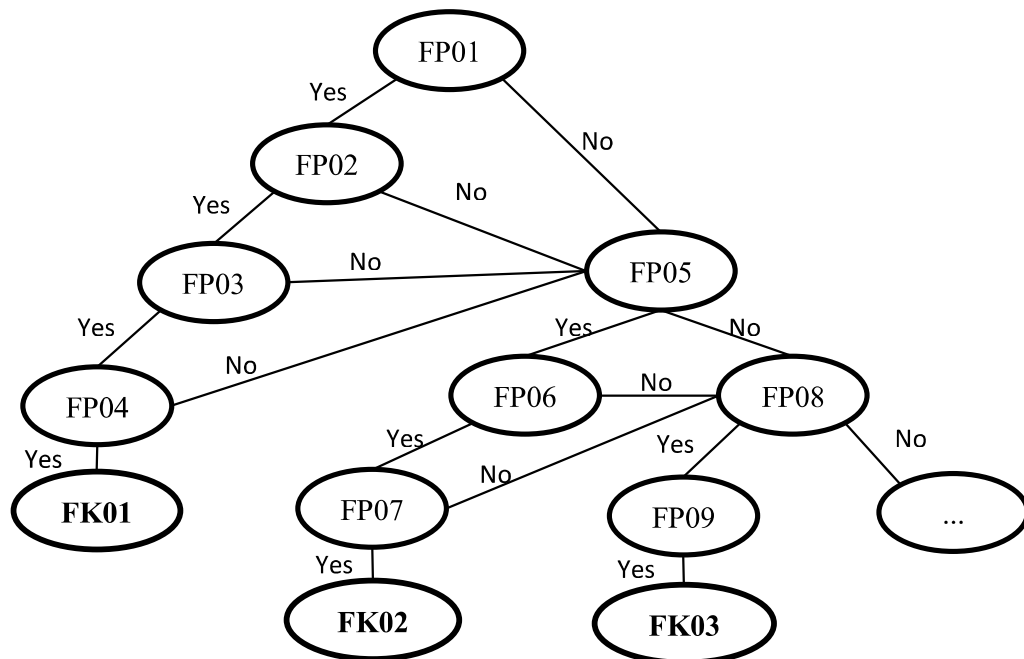
Kaidah berdasarkan dari tabel diatas:

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

Penjelasan dari kaidah diatas:

1. Jika Proyektor tidak bekerja ketika dinyalakan dan Ujung bola lampu pecah dan Kipas menyala kencang saat lampu alarm menyala dan Lampu LED berkedip ketika dinyalakan maka Masalah pada lampu.
2. Jika Mati Total (tidak bisa dinyalakan sama sekali) dan *Premature Shutdown* (hidup sebentar terus mati) dan Nyala namun lampu tidak mau nyala maka masalah pada *Power Supply*.
3. Jika Tampilan layar muncul bintik-bintik dan Tampilan layar muncul garis vertikal maka masalah pada *DMD Chip*.

Berdasarkan tabel keputusan diatas maka pohon keputusannya adalah sebagai berikut:



**Gambar 3.2** Pohon Keputusan (FC)

**Sumber:** Data Penelitian (2017)

### 3.2.2 Metode *Backward Chaining*

Runut balik merupakan metode penalaran kebalikan dari runut maju. Dalam runut balik penalaran dimulai dengan tujuan kemudian merunut balik ke jalur yang akan mengarahkan ke tujuan tersebut (Giarattano dan Riley, 1994). Runut balik disebut juga sebagai *goal-driven reasoning*, merupakan cara yang efisien untuk memecahkan masalah yang dimodelkan sebagai masalah pemilihan terstruktur.

Tujuan inferensi adalah mengambil pilihan terbaik dari banyak kemungkinan. Metode inferensi runut balik ini cocok digunakan untuk memecahkan masalah diagnosis (Schnupp, 1989).

**Tabel 3.6** Tabel Keputusan (BC)

Kode Penyebab	Kode Kerusakan		
	FK01	FK02	FK03
FP01	√		
FP02	√		
FP03	√		
FP04	√		
FP05		√	
FP06		√	
FP07		√	
FP08			√
FP09			√

**Sumber:** Data Penelitian (2017)

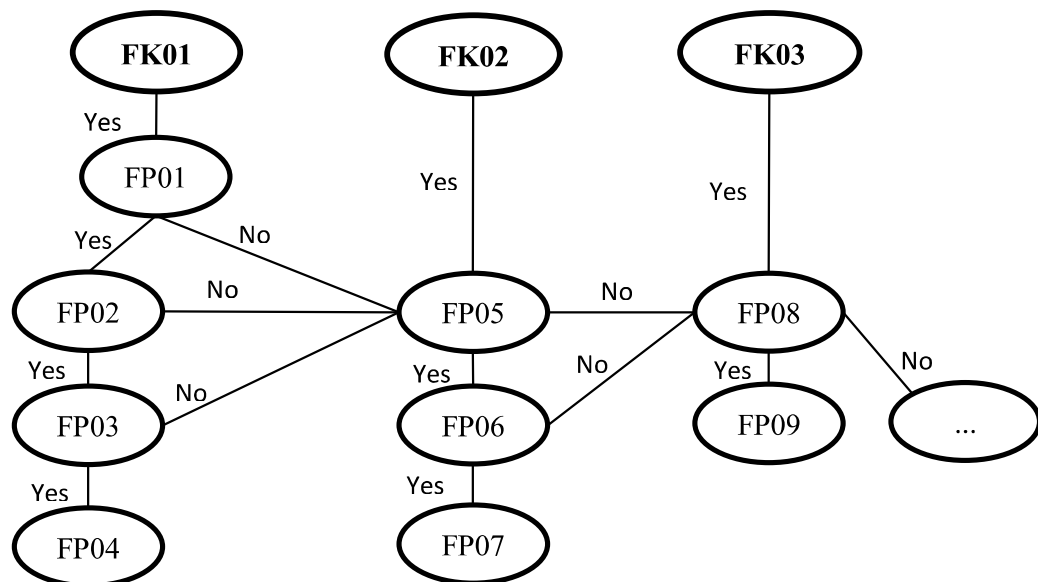
Kaidah berdasarkan dari tabel diatas:

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

Penjelasan dari kaidah diatas:

1. Jika Masalah pada lampu maka Proyektor tidak bekerja ketika dinyalakan dan Ujung bola lampu pecah dan Kipas menyala kencang saat lampu alarm menyala dan Lampu LED berkedip ketika dinyalakan.
2. Jika masalah pada *Power Supply* maka Mati Total (tidak bisa dinyalakan sama sekali) dan *Premature Shutdown* (hidup sebentar terus mati) dan Nyala namun lampu tidak mau nyala.
3. Jika masalah pada *DMD Chip* maka Tampilan layar muncul bintik-bintik dan Tampilan layar muncul garis vertikal.

Berdasarkan tabel keputusan diatas maka pohon keputusannya adalah sebagai berikut:



**Gambar 3.3** Pohon Keputusan (BC)  
**Sumber:** Data Penelitian (2017)

### **3.3 Teknik Pengumpulan Data**

Data yang dibutuhkan dalam analisis ini adalah gejala dan solusi. Dari data yang diperoleh dapat dilakukan analisis terhadap kebutuhan sistem, yang selanjutnya dijadikan sebagai acuan untuk menerjemahkan ke dalam bahasa pemrograman.

Cara pengumpulan data dalam penelitian ini yaitu:

#### **a. Teknik Wawancara**

Merupakan metode yang dilakukan dengan cara melakukan diskusi serta tanya jawab dengan sumber yang dianggap memiliki pengetahuan lebih mengenai permasalahan yang dijadikan objek penelitian. Alat bantu yang peneliti gunakan dalam wawancara ini yaitu alat bantu berupa alat tulis untuk mencatat hasil wawancara.

#### **b. Teknik Studi Kepustakaan**

Merupakan metode yang dilakukan dengan cara mencari bahan yang mendukung dalam pendelisian masalah melalui buku-buku yang erat kaitannya dengan objek permasalahannya.

### **3.4 Perancangan Sistem**

Untuk merancang suatu sistem tentang kerusakan proyektor dalam langkah-langkah operasi dalam proses pengolahan data dan prosedur untuk mendukung operasi sistem.

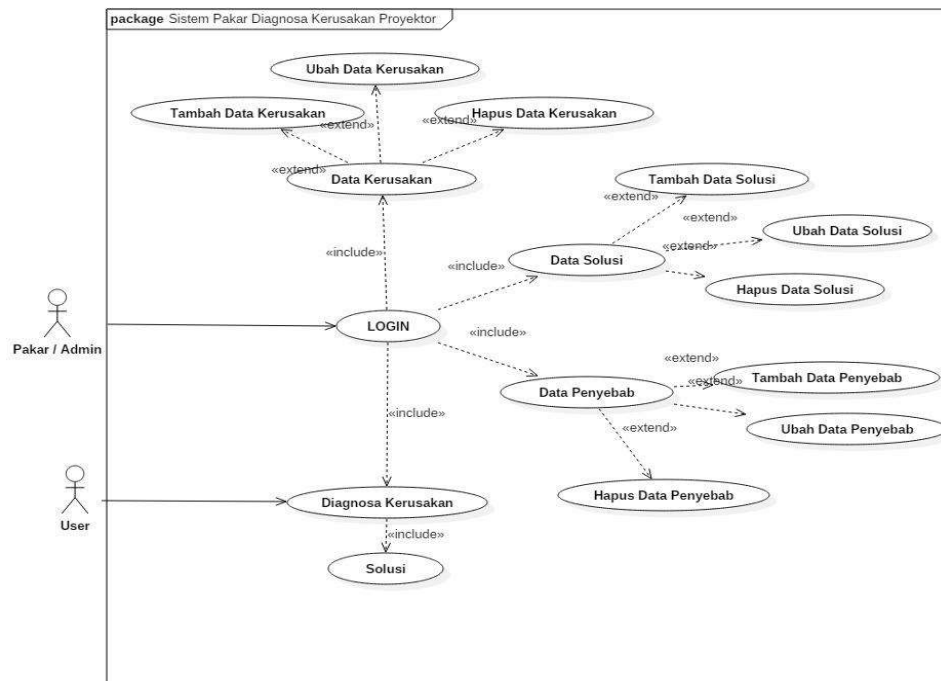
#### **3.4.1 UML (*Unified Modelling Language*)**

Suatu metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi UML yaitu sebagai suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem *software*.

##### **A. *Use Case Diagram***

*Use Case* diagram, yaitu menggambarkan scenario atau interaksi yang dapat dilakukan oleh seorang aktor, aktor disini berupa admin dan *user*. Adapun *use case* diagram tersebut yaitu:





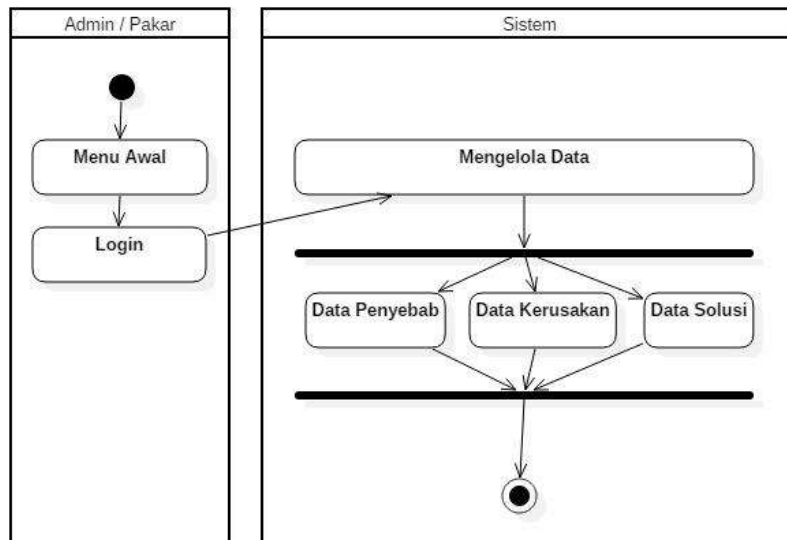
**Gambar 3.4** Use Case Diagram  
**Sumber:** Data Penelitian (2017)

## B. Activity Diagram

Activity diagram atau diagram aktivitas yaitu salah satu jenis diagram pada UML yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.

i. *Activity Diagram Admin*

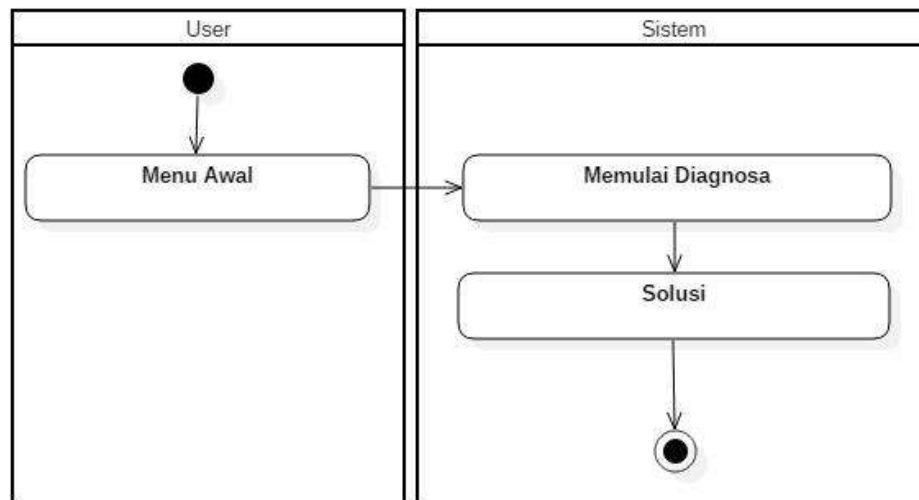
Proses dimana admin masuk ke menu awal, melakukan *login*, mengelola data termasuk menambah/mengubah/menghapus data.



**Gambar 3.5** *Activity Diagram Admin*  
**Sumber:** Data Penelitian (2017)

ii. *Activity Diagram* Pengguna

Proses dimana pengguna melakukan diagnosa termasuk menjawab pertanyaan dan mendapatkan solusi.



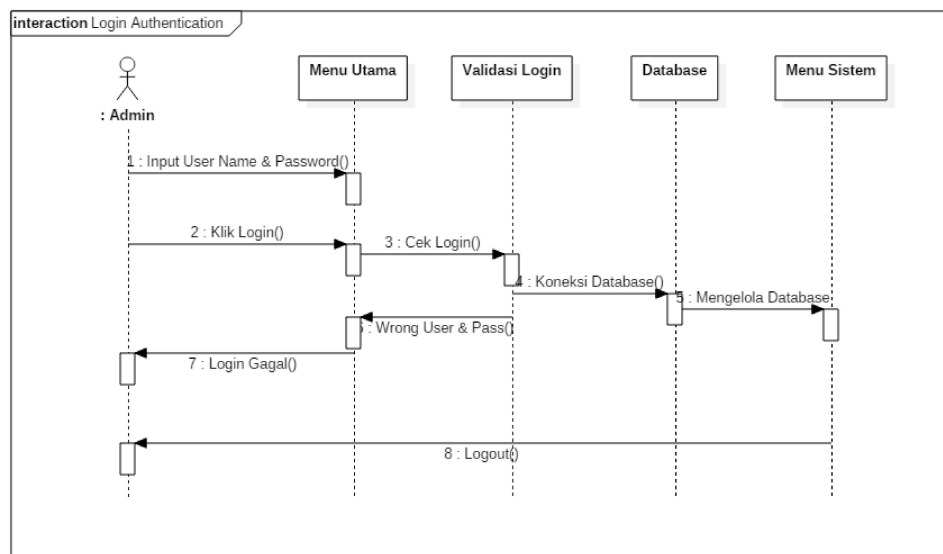
**Gambar 3.6** *Activity Diagram* Pengguna

**Sumber:** Data Penelitian (2017)

### C. *Sequence Diagram*

#### i. *Sequence Diagram Login Admin*

Admin menginput ID dan Password, dilanjutkan dengan klik Login kemudian validasi login, jika valid admin akan masuk ke database dan mengelola data jika invalid atau salah ID/Password maka Login gagal, setelah mengelola data admin logout.

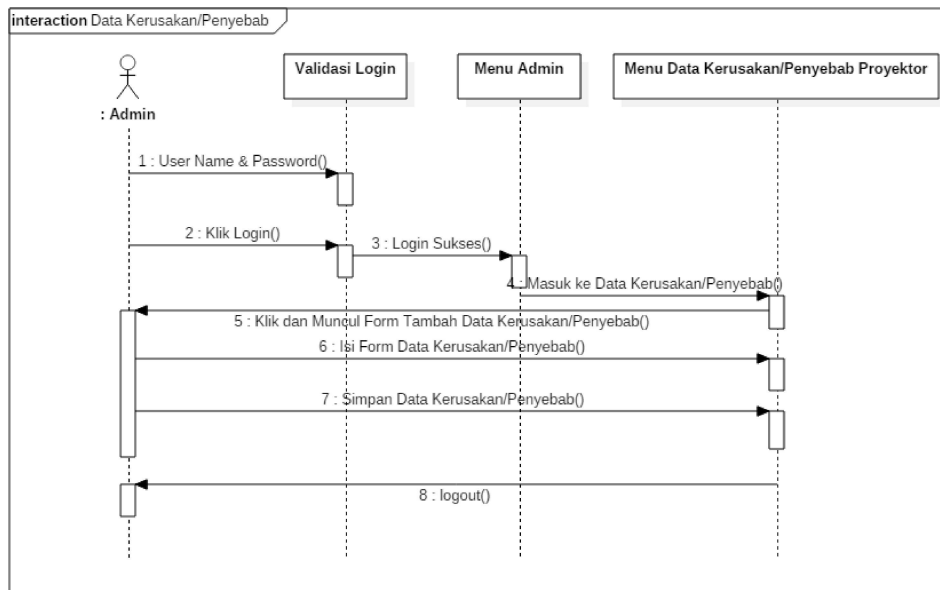


**Gambar 3.7** *Sequence Diagram Login Admin*

**Sumber:** Data Penelitian (2017)

ii. *Sequence Diagram Input Data Kerusakan/Penyebab*

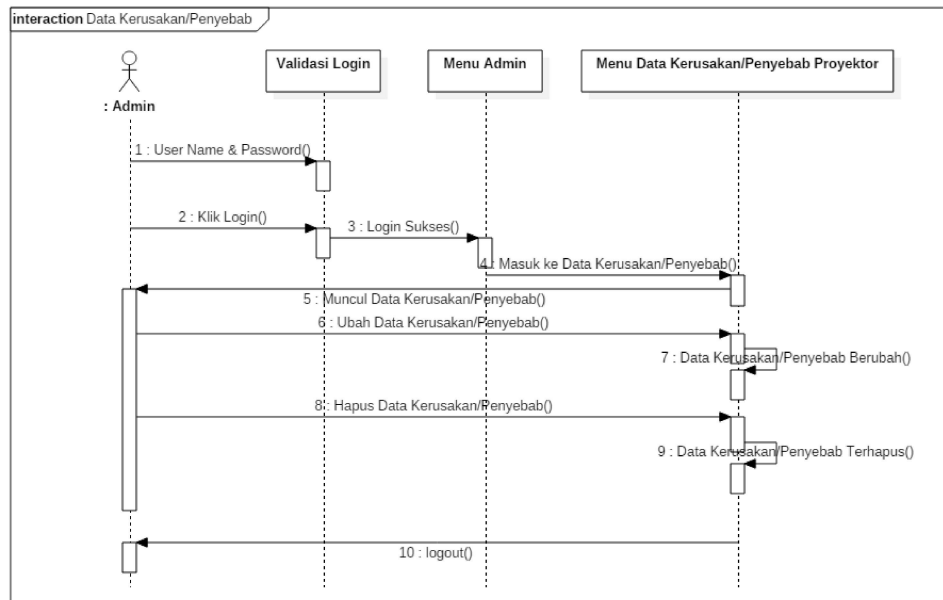
Admin melakukan login, klik login jika login sukses admin akan masuk ke data kerusakan atau penyebab, disini admin muncul form tambah data kerusakan atau penyebab, selanjutnya admin akan mengisi form data kerusakan atau penyebab dan menyimpannya.



**Gambar 3.8** *Sequence Diagram Input Data Kerusakan/Penyebab*  
**Sumber:** Data Penelitian (2017)

iii. *Sequence Diagram* Ubah Data Kerusakan/Penyebab

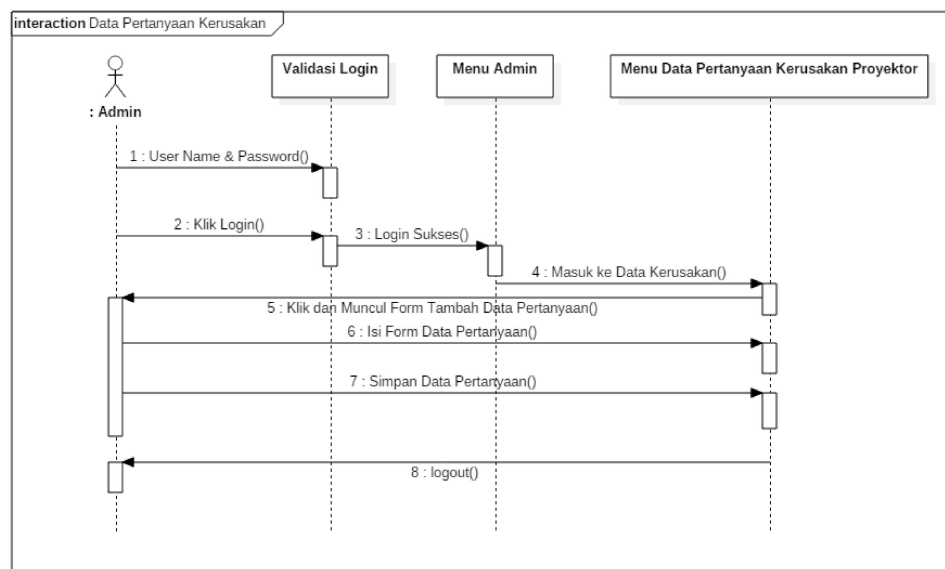
Admin melakukan login, klik login jika login sukses admin akan masuk ke data kerusakan atau penyebab, disini admin muncul form edit dan hapus data kerusakan atau penyebab, jika admin mengedit, data kerusakan atau penyebab akan mengubah, jika admin menghapus, maka data tersebut akan terhapus



**Gambar 3.9** *Sequence Diagram* Ubah Data Kerusakan/Penyebab  
**Sumber:** Data Penelitian (2017)

iv. *Sequence Diagram Input Data Pertanyaan*

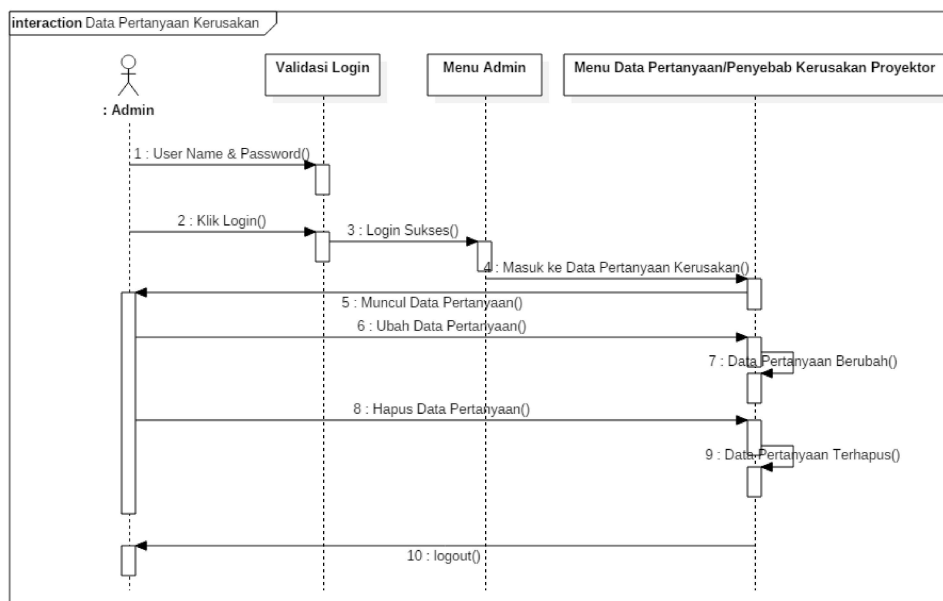
Admin melakukan login, klik login jika login sukses admin akan masuk ke data pertanyaan, disini admin muncul form tambah data pertanyaan, selanjutnya admin akan mengisi form data pertanyaan dan menyimpannya.



**Gambar 3.10** *Sequence Diagram Input Data Pertanyaan/Penyebab*  
**Sumber:** Data Penelitian (2017)

v. *Sequence Diagram* Ubah Data Pertanyaan

Admin melakukan login, klik login jika login sukses admin akan masuk ke data pertanyaan, disini admin muncul form edit dan hapus data pertanyaan, jika admin mengedit, data pertanyaan akan mengubah, jika admin menghapus, maka data pertanyaan akan terhapus.

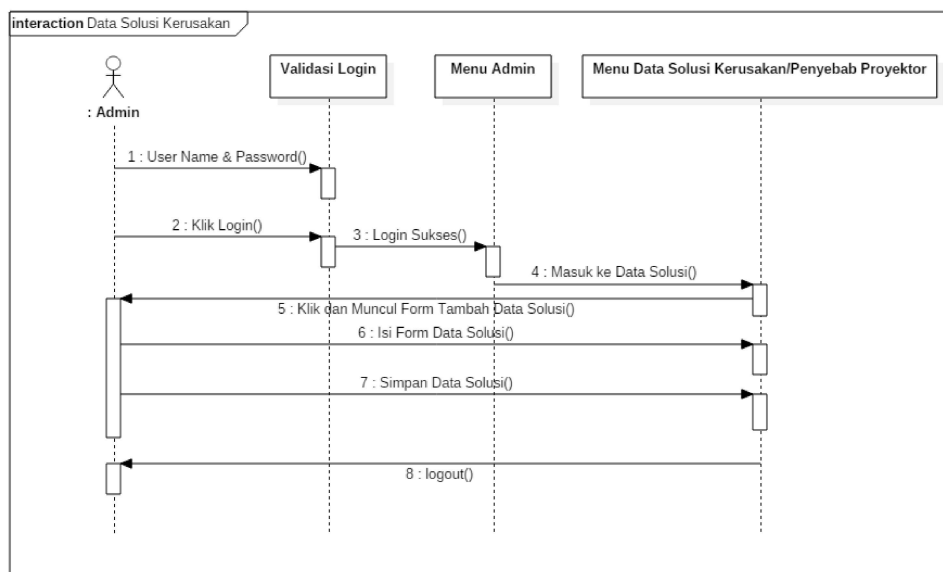


**Gambar 3.11** *Sequence Diagram* Ubah Data Pertanyaan Kerusakan/Penyebab  
**Sumber:** Data Penelitian (2017)



vi. *Sequence Diagram Input Data Solusi*

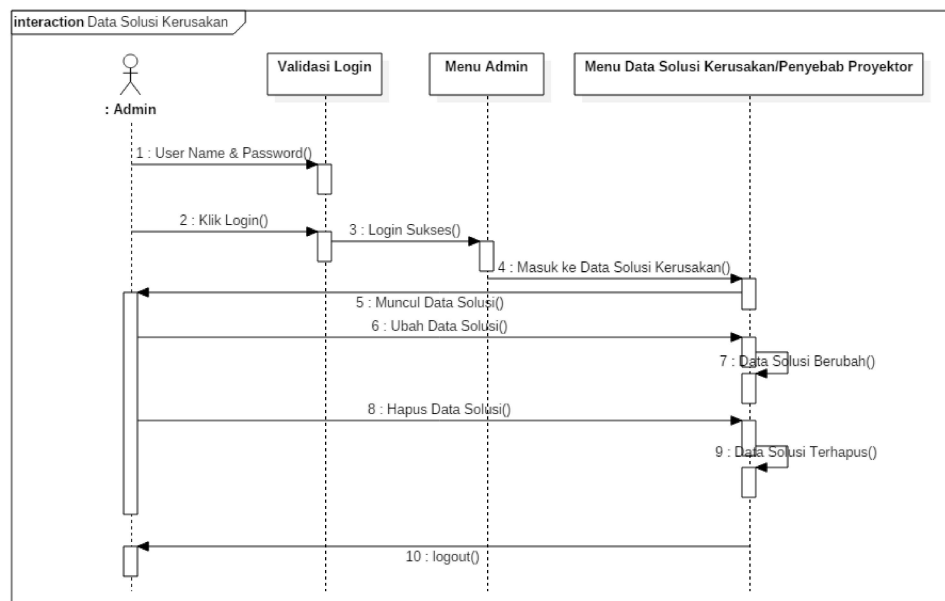
Admin melakukan login, klik login jika login sukses admin akan masuk ke data solusi, disini admin muncul form tambah data solusi, selanjutnya admin akan mengisi form data solusi dan menyimpannya.



**Gambar 3.12** *Sequence Diagram Input Data Solusi Kerusakan/Penyebab*  
**Sumber:** Data Penelitian (2017)

vii. *Sequence Diagram* Ubah Data Solusi

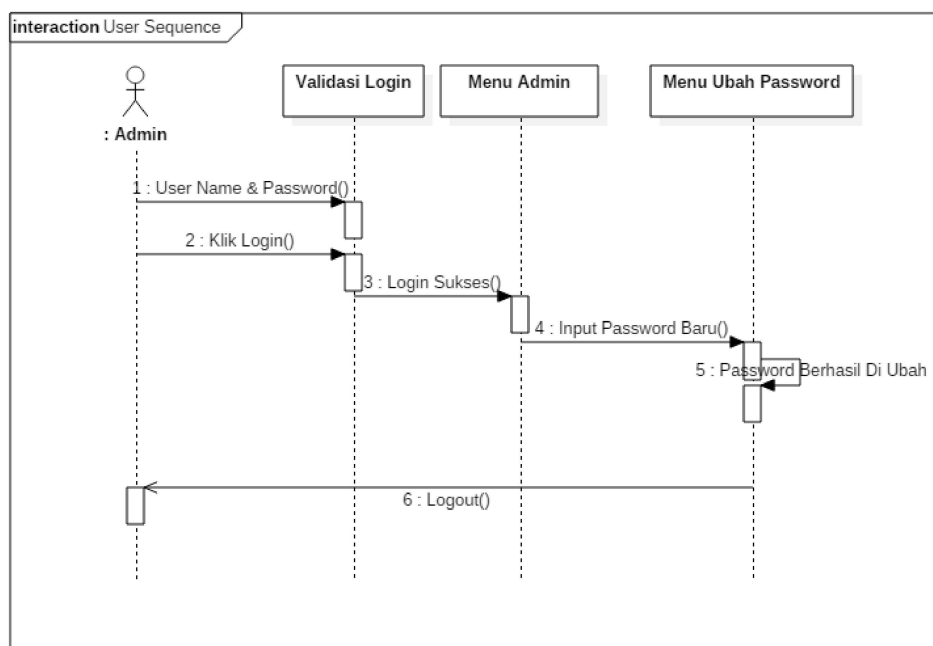
Admin melakukan login, klik login jika login sukses admin akan masuk ke data solusi, disini admin muncul form edit dan hapus data solusi, jika admin mengedit, data solusi akan mengubah, jika admin menghapus, maka data solusi akan terhapus.



**Gambar 3.13** *Sequence Diagram* Ubah Data Solusi Kerusakan/Penyebab  
**Sumber:** Data Penelitian (2017)

viii. *Sequence Diagram Ubah Password*

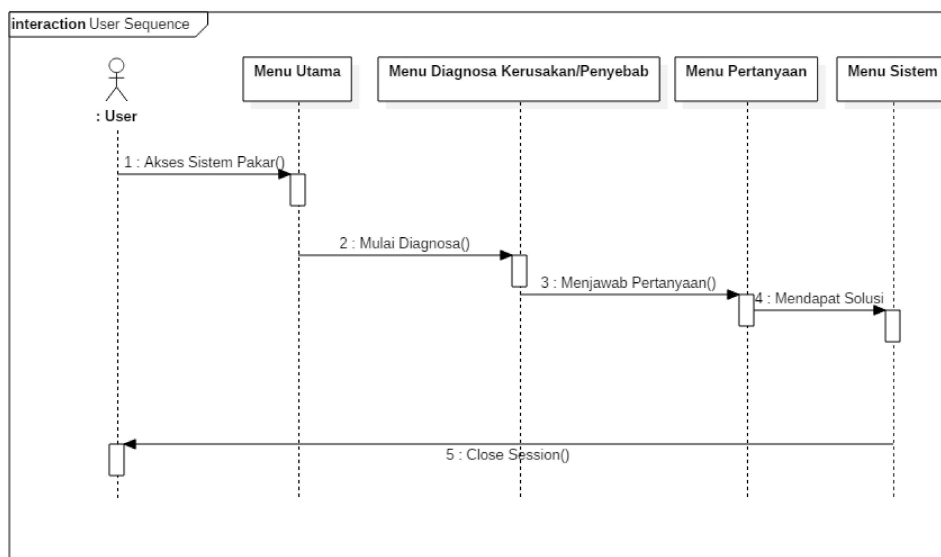
Admin melakukan login, klik login jika login sukses admin maka akan muncul menu khusus admin dimana terdapat salah satu menu ubah password yang dapat mengubah *password* admin.



**Gambar 3.14** *Sequence Diagram Ubah Password*  
**Sumber:** Data Penelitian (2017)

ix. *Sequence Diagram User Access*

*User* atau pengguna akan mengakses sistem pakar tersebut dan melakukan diagnosa seterusnya dilanjutkan dengan menjawab pertanyaan-pertanyaan serta menampilkan solusi sesuai dengan jawaban yang telah dipilih.

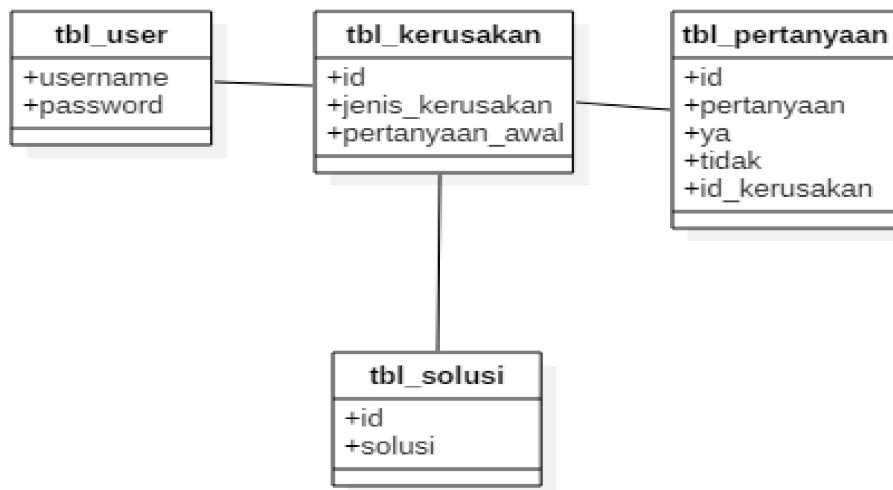


**Gambar 3.15** *Sequence Diagram User Access*

**Sumber:** Data Penelitian (2017)

#### D. *Class Diagram*

Diagram berikut adalah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut.



**Gambar 3.16** *Class Diagram*  
**Sumber:** Data Penelitian (2017)

#### 3.4.2 Desain Basis Data

##### 1. Tabel Basis Data Admin

**Tabel 3.7** Tabel Basis Data Admin

<i>Field</i>	Tipe	Panjang
Username	Varchar	20
Password	Varchar	50

**Sumber:** Data Penelitian (2017)

## 2. Tabel Basis Data Kerusakan/Penyebab

**Tabel 3.8** Tabel Basis Data Kerusakan/Penyebab

<i>Field</i>	Tipe	Panjang
Id	Varchar	5
Jenis_kerusakan	Text	
Pertanyaan_awal	Varchar	5

**Sumber:** Data Penelitian (2017)

## 3. Tabel Basis Data Pertanyaan

**Tabel 3.9** Tabel Basis Data Pertanyaan

<i>Field</i>	Tipe	Panjang
Id	Varchar	5
pertanyaan	Text	
ya	Varchar	5
tidak	Varchar	5
Id_kerusakan	Varchar	5

**Sumber:** Data Penelitian (2017)

## 4. Tabel Basis Data Solusi

**Tabel 3.10** Tabel Basis Data Solusi

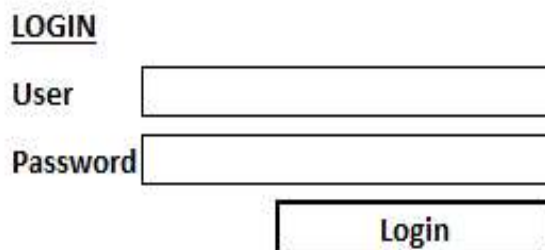
<i>Field</i>	Tipe	Panjang
Id	Varchar	5
Solusi	Text	

**Sumber:** Data Penelitian (2017)

### 3.4.3 Desain Antarmuka

#### 1. Menu Form *Login*

Form ini menampilkan menu login khusus untuk *administrator* untuk dapat akses buat mengelola *database* aplikasi sistem pakar tersebut.



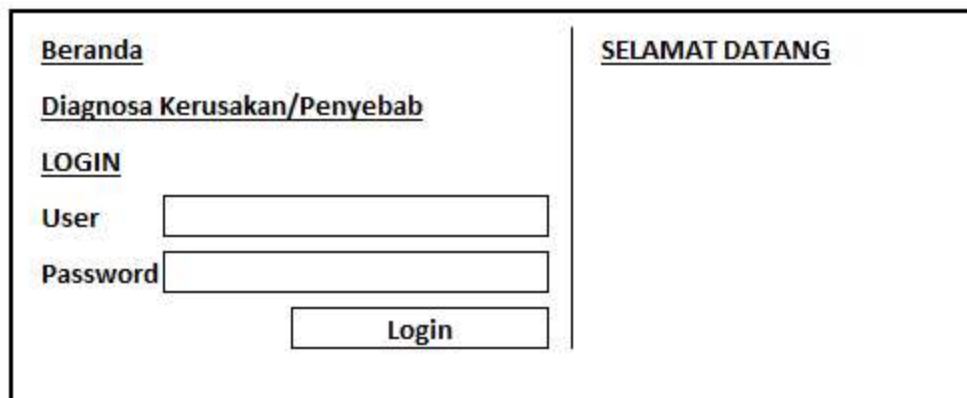
The image shows a login form with the following elements:

- The word **LOGIN** is underlined at the top left.
- Below it, the label **User** is followed by a rectangular input field.
- Below that, the label **Password** is followed by another rectangular input field.
- At the bottom center, there is a rectangular button labeled **Login**.

**Gambar 3.17** Form *Login* Aplikasi  
**Sumber:** Data Penelitian (2017)

#### 2. Menu Utama

Pada menu ini menampilkan menu utama dari aplikasi sistem pakar tersebut.



The image shows a main menu interface with the following elements:

- On the left side, there is a vertical list of menu items: **Beranda**, **Diagnosa Kerusakan/Penyebab**, and **LOGIN**. Each item is underlined.
- Below the **LOGIN** item, there is a sub-form with the labels **User** and **Password**, each followed by a rectangular input field, and a **Login** button at the bottom.
- On the right side, the text **SELAMAT DATANG** is underlined.

**Gambar 3.18** Menu Utama Aplikasi  
**Sumber:** Data Penelitian (2017)

### 3. Menu Diagnosa Kerusakan/Penyebab

Pada menu ini menampilkan menu diagnosa yang akan digunakan oleh para pengguna untuk memulai mendiagnosa masalah pada proyektor.

<p><u>Beranda</u></p> <p><u>Diagnosa Kerusakan/Penyebab</u></p> <p><u>LOGIN</u></p> <p>User <input type="text"/></p> <p>Password <input type="text"/></p> <p style="text-align: center;"><input type="button" value="Login"/></p>	<p><u>PILIH JENIS KERUSAKAN/PENYEBAB</u></p> <p><i>kerusakan/penyebab 1</i></p> <p><i>kerusakan/penyebab 2</i></p>
---	--

**Gambar 3.19** Menu Diagnosa Aplikasi

**Sumber:** Data Penelitian (2017)

### 4. Menu Tanya Jawab

Pada menu ini menampilkan menu tanya jawab ketika memilih kerusakan atau penyebab yang dipilih saat di menu diagnosa.

<p><u>Beranda</u></p> <p><u>Diagnosa Kerusakan/Penyebab</u></p> <p><u>LOGIN</u></p> <p>User <input type="text"/></p> <p>Password <input type="text"/></p> <p style="text-align: center;"><input type="button" value="Login"/></p>	<table border="1"> <tr> <td style="text-align: center;"><u>Pertanyaan</u></td> <td colspan="2" style="text-align: center;"><i>pertanyaan 1</i></td> </tr> <tr> <td></td> <td style="text-align: center;">Ya</td> <td style="text-align: center;">Tidak</td> </tr> </table>	<u>Pertanyaan</u>	<i>pertanyaan 1</i>			Ya	Tidak
<u>Pertanyaan</u>	<i>pertanyaan 1</i>						
	Ya	Tidak					

**Gambar 3.20** Menu Tanya Jawab Aplikasi

**Sumber:** Data Penelitian (2017)



### 5. Menu Hasil Solusi

Pada menu ini menampilkan menu hasil solusi setelah menjawab dari pertanyaan sistem pada menu sebelumnya.

<u>Beranda</u>	<u>Solusi</u>	solusi 1
<u>Diagnosa Kerusakan/Penyebab</u>		
<u>LOGIN</u>		
User	<input type="text"/>	
Password	<input type="text"/>	
	<input type="button" value="Login"/>	

**Gambar 3.21** Menu Hasil Solusi Aplikasi  
**Sumber:** Data Penelitian (2017)

### 6. Menu Data Kerusakan/Penyebab

Pada menu ini menampilkan menu data kerusakan yang dimana akan digunakan hanya oleh pihak *administrator* untuk dapat menambah, mengubah ataupun menghapus data kerusakan sistem pakar.

<u>Beranda</u>	<u>DATA KERUSAKAN/PENYEBAB</u>	
<u>Diagnosa Kerusakan/Penyebab</u>	<input type="button" value="INPUT"/>	
<u>Data Kerusakan/Penyebab</u>	1. kerusakan/penyebab 1	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
<u>Data Solusi</u>	2. kerusakan/penyebab 2	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
<u>Data Pertanyaan</u>		
<u>Ubah Password</u>		
<u>Logout</u>		

**Gambar 3.22** Menu Data Kerusakan Aplikasi  
**Sumber:** Data Penelitian (2017)

### 7. Menu Tambah Data Kerusakan/Penyebab

Pada menu ini berfungsi untuk pihak *administrator* dapat menambah data-data kerusakan atau penyebab baru ke dalam basis data aplikasi sistem pakar tersebut.

<p><u><a href="#">Beranda</a></u></p> <p><u><a href="#">Diagnosa Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Solusi</a></u></p> <p><u><a href="#">Data Pertanyaan</a></u></p> <p><u><a href="#">Ubah Password</a></u></p> <p><u><a href="#">Logout</a></u></p>	<p><b>INPUT DATA KERUSAKAN/PENYEBAB</b></p> <p>ID : <input type="text"/></p> <p>Kerusakan : <input type="text"/></p> <p>Pertanyaan_awal : <input type="text"/></p> <p><input type="button" value="Simpan"/> <input type="button" value="Batal"/></p>
--	--

**Gambar 3.23** Menu Tambah Data Kerusakan/Penyebab Aplikasi  
**Sumber:** Data Penelitian (2017)

### 8. Menu Ubah Data Kerusakan/Penyebab

Pada menu ini berfungsi untuk pihak *administrator* dapat mengubah data-data kerusakan atau penyebab yang ada di dalam basis data aplikasi sistem pakar tersebut.

<p><u><a href="#">Beranda</a></u></p> <p><u><a href="#">Diagnosa Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Solusi</a></u></p> <p><u><a href="#">Data Pertanyaan</a></u></p> <p><u><a href="#">Ubah Password</a></u></p> <p><u><a href="#">Logout</a></u></p>	<p><b>UBAH DATA KERUSAKAN/PENYEBAB</b></p> <p>ID : <input type="text" value="01"/></p> <p>Kerusakan : <input type="text" value="kerusakan/penyebab 1"/></p> <p>Pertanyaan_awal : <input type="text" value="pertanyaan 1"/></p> <p><input type="button" value="Simpan"/> <input type="button" value="Batal"/></p>
--	--

**Gambar 3.24** Menu Ubah Data Kerusakan/Penyebab Aplikasi  
**Sumber:** Data Penelitian (2017)

### 9. Menu Data Pertanyaan

Pada menu ini menampilkan menu data kerusakan yang dimana akan digunakan hanya oleh pihak *administrator* untuk menambah, mengubah ataupun menghapus data pertanyaan sistem pakar.

<p><u><a href="#">Beranda</a></u></p> <p><u><a href="#">Diagnosa Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Solusi</a></u></p> <p><u><a href="#">Data Pertanyaan</a></u></p> <p><u><a href="#">Ubah Password</a></u></p> <p><u><a href="#">Logout</a></u></p>	<p><b>DATA PERTANYAAN</b></p> <p><input type="button" value="INPUT"/></p> <p>1. pertanyaan 1 <input type="button" value="Edit"/> <input type="button" value="Hapus"/></p> <p>2. pertanyaan 2 <input type="button" value="Edit"/> <input type="button" value="Hapus"/></p>
--	---

**Gambar 3.25** Menu Data Pertanyaan Aplikasi

**Sumber:** Data Penelitian (2017)

### 10. Menu Tambah Data Pertanyaan

Pada menu ini berfungsi untuk pihak *administrator* dapat menambah data-data pertanyaan yang baru ke dalam basis data aplikasi sistem pakar tersebut.

<p><u><a href="#">Beranda</a></u></p> <p><u><a href="#">Diagnosa Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Kerusakan/Penyebab</a></u></p> <p><u><a href="#">Data Solusi</a></u></p> <p><u><a href="#">Data Pertanyaan</a></u></p> <p><u><a href="#">Ubah Password</a></u></p> <p><u><a href="#">Logout</a></u></p>	<p><b>INPUT DATA PERTANYAAN</b></p> <p>ID Pertanyaan : <input type="text"/></p> <p>Pertanyaan : <input type="text"/></p> <p>Iya : <input type="text"/></p> <p>Tidak : <input type="text"/></p> <p>Jenis Kerusakan : <input type="text"/></p> <p><input type="button" value="Simpan"/> <input type="button" value="Batal"/></p>
--	--

**Gambar 3.26** Menu Tambah Data Pertanyaan Aplikasi

**Sumber:** Data Penelitian (2017)

### 11. Menu Ubah Data Pertanyaan

Pada menu ini berfungsi untuk pihak *administrator* dapat mengubah data-data pertanyaan yang ada di dalam basis data aplikasi sistem pakar tersebut.

<p><u>Beranda</u></p> <p><u>Diagnosa Kerusakan/Penyebab</u></p> <p><u>Data Kerusakan/Penyebab</u></p> <p><u>Data Solusi</u></p> <p><u>Data Pertanyaan</u></p> <p><u>Ubah Password</u></p> <p><u>Logout</u></p>	<p><b>UBAH DATA PERTANYAAN</b></p> <p>ID Pertanyaan : <input type="text" value="01"/></p> <p>Pertanyaan : <input type="text" value="pertanyaan 1"/></p> <p>Iya : <input type="text" value="02"/></p> <p>Tidak : <input type="text" value="03"/></p> <p>Jenis Kerusakan : <input type="text" value="kerusakan/penyebab 1"/></p> <p><input type="button" value="Simpan"/> <input type="button" value="Batal"/></p>
--	--

**Gambar 3.27** Menu Ubah Data Pertanyaan Aplikasi  
**Sumber:** Data Penelitian (2017)

### 12. Menu Data Solusi

Pada menu ini menampilkan menu data kerusakan yang dimana akan digunakan hanya oleh pihak *administrator* untuk menambah, mengubah ataupun menghapus data solusi sistem pakar.

<p><u>Beranda</u></p> <p><u>Diagnosa Kerusakan/Penyebab</u></p> <p><u>Data Kerusakan/Penyebab</u></p> <p><u>Data Solusi</u></p> <p><u>Data Pertanyaan</u></p> <p><u>Ubah Password</u></p> <p><u>Logout</u></p>	<p><b>DATA SOLUSI</b></p> <p><input type="button" value="INPUT"/></p> <p>1. solusi 1 <input type="button" value="Edit"/> <input type="button" value="Hapus"/></p> <p>2. solusi 2 <input type="button" value="Edit"/> <input type="button" value="Hapus"/></p>
--	---

**Gambar 3.28** Menu Data Solusi Aplikasi  
**Sumber:** Data Penelitian (2017)

### 13. Menu Tambah Data Solusi

Pada menu ini berfungsi untuk pihak *administrator* dapat menambah data-data solusi yang baru ke dalam basis data aplikasi sistem pakar tersebut.

**Gambar 3.29** Menu Tambah Data Solusi Aplikasi  
**Sumber:** Data Penelitian (2017)

### 14. Menu Ubah Data Solusi

Pada menu ini berfungsi untuk pihak *administrator* dapat mengubah data-data solusi yang ada di dalam basis data aplikasi sistem pakar tersebut.

**Gambar 3.30** Menu Ubah Data Solusi Aplikasi  
**Sumber:** Data Penelitian (2017)

### 15. Menu Ubah *Password*

Pada menu ini berfungsi untuk pihak *administrator* dapat mengubah *password administrator* sistem aplikasi tersebut.

<u>Beranda</u>	<u>UBAH PASSWORD</u>
<u>Diagnosa Kerusakan/Penyebab</u>	Password Lama : <input type="text"/>
<u>Data Kerusakan/Penyebab</u>	Password Baru : <input type="text"/>
<u>Data Solusi</u>	Ulang Password Baru : <input type="text"/>
<u>Data Pertanyaan</u>	
<u>Ubah Password</u>	
<u>Logout</u>	
	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>

**Gambar 3.31** Menu Ubah *Password* Aplikasi  
**Sumber:** Data Penelitian (2017)

### 3.5 Lokasi Dan Jadwal Penelitian

Penelitian di laksanakan di, penelitian akan dilaksanakan sesudah seminar proposal.

Kegiatan \ Jadwal	Oktober				November				Desember				Januari				Februari			
	2016				2016				2016				2017				2017			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Identifikasi Masalah	■																			
Pengumpulan Data	■																			
Analisa Data	■	■	■	■																
Perancangan Sistem					■	■	■	■												
Pengkodean									■	■	■	■	■	■	■	■				
Uji Coba																	■	■	■	
Penulisan Skripsi	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	