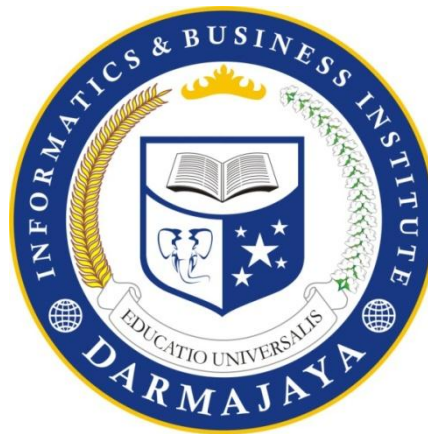


**RANCANG BANGUN SISTEM PENDETAAN KERUSAKAN  
KAMERA MENGGUNAKAN METODE CASE BASED  
REASONING**

**SKRIPSI**



Disusun Oleh:

**FERIYANTO**

09010013

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
INFORMATICS AND BUSINESS INSTITUTE DARMAJAYA  
BANDAR LAMPUNG  
2016**

**RANCANG BANGUN SISTEM PENDETAAN KERUSAKAN  
KAMERA MENGGUNAKAN METODE CASE BASED  
REASONING**

**SKRIPSI**

**Sebagai Salah Satu Syarat Untuk Mencapai Gelar  
SARJANA KOMPUTER  
Pada Jurusan Teknik Informatika**



Disusun Oleh:

**FERIYANTO**

NPM. 09010013

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
INFORMATICS AND BUSINESS INSTITUTE DARMAJAYA  
BANDAR LAMPUNG  
2016**



## PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi yang saya ajukan ini adalah hasil karya saya sendiri, tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi atau karya yang pernah ditulis atau diterbitkan orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka. Karya ini adalah milik saya dan pertanggung jawaban sepenuhnya berada di pundak saya.

Bandar Lampung, 14 September 2016



**FERIYANTO**

NPM. 09010013

**HALAMAN PERSETUJUAN**

Judul Skripsi : **RANCANG BANGUN SISTEM PENDAIGNOSA  
KERUSAKAN KAMERA MENGGUNAKAN  
METODE CASE BASED REASONING**

Nama Mahasiswa : **FERIYANTO**

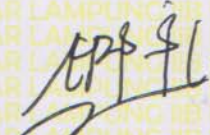
No. Pokok Mahasiswa : 09010013

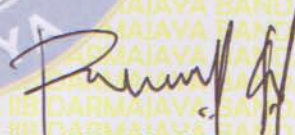
Jurusan : Teknik Informatika



Dosen Pembimbing

Ketua Jurusan Teknik Informatika

  
**Sri Lestari, S.Kom., M.Cs**  
NIK 01261005

  
**Rionaldi Ali, S.Kom., M.T.I**  
NIK 12710212

**HALAMAN PENGESAHAN**

Telah Diuji dan Dipertahankan Didepan Tim Penguji Skripsi  
Jurusan Teknik Informatika Informatics & Bussines Institute Darmajaya  
Bandar Lampung dan Dinyatakan Diterima untuk  
Memenuhi Syarat Guna Memperoleh Gelar  
Sarjana Komputer

**Mengesahkan**

1. Tim Penguji

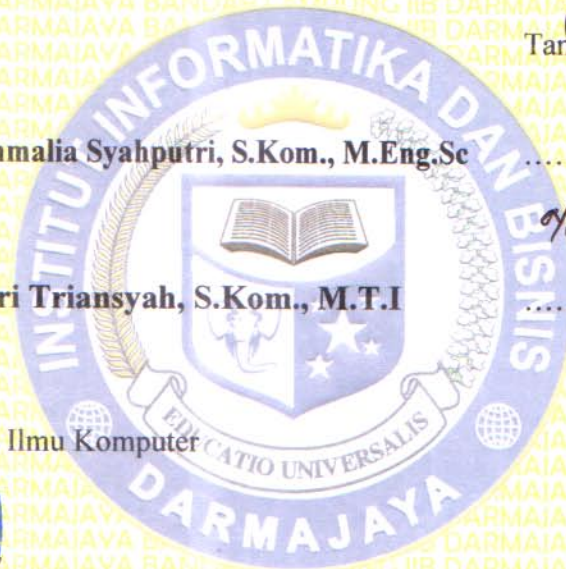
Tanda Tangan

Ketua : **Rahmalia Syahputri, S.Kom., M.Eng.Sc**

Anggota : **Apri Triansyah, S.Kom., M.T.I**

Dekan Fakultas Ilmu Komputer

**Dr. R. Z. Abdul Aziz, M.T**  
NIK 01050904



Tanggal Lulus Ujian Skripsi : 15 Agustus 2016

## **RIWAYAT HIDUP**

### **I. IDENTITAS**

- a. Nama : FERİYANTO
- b. Npm : 09010013
- c. Tempat/Tanggal Lahir : Pesawaran, 10 Juli 1990
- d. Agama : Islam
- e. Alamat : Kresnowidodo Kecamatan Tegineneng  
Kabupaten Pesawaran
- f. Suku : Jawa
- g. Kewarganegaraan : Indonesia
- h. No. Telp / Hp : 082282267210
- i. E-mail : andesfery@gmail.com

### **II. Pendidikan yang pernah ditempuh oleh penulis, antara lain :**

1. Pendidikan Sekolah Dasar (SD) Negeri 01 Kresnowidodo tahun 2003.
2. Sekolah Lanjut Tingkat Pertama (SLTP) Negeri 01 Tegineneng 2006.
3. Sekolah Menengah Kejuruan (SMK) Muhamdiah 2 Metro 2009.
4. Pada tahun 2009 Penulis diterima di IBI Darmajaya Jurusan S-1 Teknik Informatika.

## **PERSEMBAHAN**

Kupersembahkan karya ku ini untuk :

1. Bapak dan Ibuku tercinta, salam hormat Ananda, terima kasih atas do'a, kasih sayang, nasehat, dana, bimbingan dan semangat serta ketabahan yang telah diberikan untuk anak-anaknya agar menjadi orang yang berguna bagi bangsa, negara dan agama.
2. Saudara-saudaraku, dan seluruh keluarga tercinta yang selalu mendukung dan mengharapkan keberhasilanku.
3. Teman-teman seperjuangan dan sahabat – sahabatku serta semua teman mahasiswa Darmajaya khususnya untuk Deni Setiawan, Beny Andrian, Alfian Nugroho, Wahyu Widodo, Vanda Sari Septayogi dan Sapri dwiguna.
4. Almamaterku, dan IBI Darmajaya yang telah memberikanku banyak ilmu.

## **MOTTO**

Apa yang sudah kita mulai harus kita selesaikan.

Menjadi lebih baik dari satu detik yang lalu.

Kesuksesan bukan diukur dari seberapa banyak kita mendapatkan sesuatu  
Tetapi kesuksesan itu bagaimana cara kita mampu menyelesaikan sesuatu.



## **ABSTRAK**

### **RANCANG BANGUN SISTEM PENDIAKNOSA KERUSAKAN KAMERA MENGUNAKAN METODE CASE BASED REASONING**

*Oleh:*

**FERIYANTO**  
NPM. 09010013

*Case Base Reasoning* (CBR) merupakan salah satu cabang penalaran komputer berbasis kasus, metode ini bertujuan untuk menyelesaikan suatu kasus baru dengan cara mengadaptasi solusi-solusi yang terdapat pada kasus-kasus sebelumnya yang mirip dengan kasus yang baru. Ide dasar dari *Case base Reasoning* (CBR) adalah bahwa manusia seringkali merujuk kepada pengalaman sebelumnya jika ada suatu masalah yang dituangkan ke dalam komputer untuk memberi solusi pemecahan berdasarkan pengalaman-pengalaman yang ada. CBR dapat diimplementasi di berbagai bidang, salah satunya dalam komputer dalam hal ini sistem pendeteksi kerusakan Kamera digital.

Berdasarkan permasalahan tersebut diatas, penulis ingin merancang sebuah aplikasi berbasis web, sehingga bisa diaplikasikan dan diakses masyarakat khususnya orang-orang yang sedang mengalami masalah atau membutuhkan informasi tentang kerusakan kamera digital secara online.

Dari hasil prediksi dapat digunakan oleh masyarakat luas sebagai tahap awal pendiagnosa kerusakan kamera digital dan dapat digunakan sebagai bahan untuk kemajuan teknologi dalam bidang system pakar.

**Kata kunci : Case Based Reasoning (CBR),Kamera, prediksi kerusakan.**

## **ABSTRACT**

### **PENDIAKNOSA SYSTEM DESIGN USING ANY CASE BASED CAMERA REASONING**

**BY**

**FERIYANTO**  
NPM. 09010013

The Fuzzy Logic is one of the methods to analyze the uncertain system. The Fuzzy Logic imitates the human way of thinking which is called reasoning, which can explain and identify automatically. This research uses inference method of Fuzzy Mamdani system. The design of the system is to reach the output is done in several steps such as (a) Fuzzy group creation, (b) implication function application, (c) rule composition, (d) confirmation (Defuzzyfication).

The flour tapioca production is one of the process that should be done by the flour tapioca factory before trading it. The prediction of flour tapioca production by using FIS Mamdani method is able to be used as the recommendation for the flour tapioca company in preparing the flour tapioca production.

From the result of flour tapioca production prediction by using Mamdani method in January 2013 as of 3279,20 tons, February 2013 as 3304,95 tons, and March 2013 as 3227.19 tons. From the test, there a difference in error between the real production and the Mamdani prediction which are in January 2013 as 250,8 tons, February 2013 as 345,05 tons, March 2013 as 122,81 tons.

**Key Words : Fuzzy Mamdani, flour tapioca Production Prediction Amount**

## **PRAKATA**

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, Karena telah memberikan limpahan rahmat serta karunianya sehingga penulis dapat menyelesaikan penulisan skripsi ini yang berjudul “ RANCANG BANGUN SISTEM PENDIAKNOSA KERUSAKAN KAMERA MENGGUNAKAN METODE CASE BASED REASONING ”.

Skripsi ini di susun sebagai syarat untuk mencapai gelar sarjana strata satu (S1) pada jurusan Teknik Informatika IBI Darmajaya Bandar Lampung. Dalam penulisan skripsi ini saya banyak memperoleh bimbingan, dorongan dan bantuan dari berbagai pihak. Untuk itu saya mengucapkan terima kasih setulus-tulusnya kepada:

1. Bapak Alfian Husein, SH., Selaku Ketua Yayasan Alfian Husein Informatics and Business Institute Darmajaya Bandar Lampung.
2. Bapak Dr. Andi Desfiandi, S.E., M.A., Selaku Rektor Informatics and Business Institute Darmajaya Bandar Lampung.
3. Bapak Envemy Vem, M.Sc., Selaku Wakil Rektor I Bidang Akademik dan Kemahasiswaan Informatics and Business Institute Darmajaya.
4. Bapak Rionaldi Ali, S.Kom Selaku Ketua Jurusan Teknik Informatika yang telah memberikan petunjuk sehingga skripsi ini dapat diselesaikan dengan baik .
5. Bapak Sri Lestari, S.Kom., M.Cs Selaku Dosen pembimbing penyusunan skripsi yang telah berkenan membimbing dan membantu dalam menyelesaikan skripsi ini.
6. Ayah dan Ibu tercinta yang selalu memberikan motivasi, doa dan mencukupi segala keperluan untuk mendukung ku.
7. Bapak Wawan Selaku Teknisi Kamera yang telah membantu memberikan informasi tentang materi skripsi saya.

8. Para Dosen, Staf dan karyawan Informatics And Business Institute Darmajaya Bandar Lampung yang telah memberi bantuan baik langsung maupun tidak langsung selama saya menjadi mahasiswa.
9. Semua pihak yang telah memberikan bantuan dan petunjuk sehingga saya dapat lebih mudah dalam menyusun skripsi ini.
10. Almamaterku Darmajaya Tercinta.

Saya menyadari bahwa skripsi ini masih jauh dari kesempurnaan, baik dalam pembahasan materi maupun dalam penyajiannya, oleh karena itu kritik dan saran yang sifatnya membangun merupakan masukan yang sangat berarti bagi penyempurnaan dimasa yang akan datang.

Semoga skripsi ini bermanfaat dan dapat dijadikan bahan pertimbangan informasi bagi pihak yang berkepentingan.

Bandar Lampung, 14 September 2016

FERIYANTO  
NPM.09010013

## DAFTAR ISI

	<b>Halaman</b>
<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>HALAMAN JUDUL DALAM .....</b>	<b>ii</b>
<b>PERNYATAAN.....</b>	<b>iii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iv</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>v</b>
<b>ABSTRAK .....</b>	<b>vi</b>
<b>ABSTRACT .....</b>	<b>vii</b>
<b>RIWAYAT HIDUP.....</b>	<b>viii</b>
<b>PERSEMBAHAN.....</b>	<b>ix</b>
<b>MOTTO .....</b>	<b>x</b>
<b>PRAKATA .....</b>	<b>xi</b>
<b>DAFTAR ISI.....</b>	<b>xiii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvi</b>
<b>DAFTAR TABEL .....</b>	<b>xvii</b>
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	4
1.5 Manfaat Penelitian .....	5
1.6 Sistematika Penulisan .....	6
<b>BAB II LANDASAAN TEORI</b>	
2.1 Case Based Reasoning .....	7
2.2 Algoritma Nearest Neighbor .....	8
2.3 Perancangan Sistem .....	9

2.3.1 Diagram Konteks .....	10
2.3.2 DFD (Data Flow Diagram) .....	11
2.1.5 Metode Pengembangan Perangkat Lunak Menggunakan Meto	
12	
2.4. Perancangan Sistem .....	13
2.4.1 DFD (Data Flow Diagram) .....	14
2.6 Metode Pengembangan Perangkat Lunak Menggunakan Metode	
System Development Life Cycle Model Waterfal .....	15
2.7 Basis Dta .....	16
2.8 <i>MySQL</i> .....	17
2.9 HTML (Hyper Text Markup Language) .....	18
2.10 PHP (Hypertext Preprocessor) .....	19
2.10 Sejarah.....	20
2.10 Pengertian PHP .....	21

### **BAB III METODE PENELITIAN**

3.1 Perepan Metode Case Base Reasoning .....	22
3.1.1 Proses Retrive.....	23
3.1.2 Proses reuse .....	24
3.2.1 Proses Revise .....	25
3.2.1 Proses Retain.....	26
3.2.1 Perancangan .....	27
3.2 Rancangan Struktur Database .....	28
3.2.1 Relasi antar Tabel.....	29
3.2.2 Perancangan Antar Muka.....	30
3.2.2.1 Halaman Index .....	31
3.2.2.2 Halaman Administrator.....	32
3.2.2.2 Halaman Ganti Login.....	33
3.2.2.2 Halaman Indikasi .....	34
3.2.2.2 Halaman Kerusakan .....	35

## **BAB IV HASIL DAN PEMBAHASAN**

4.1 Implementasi Program .....	36
4.2 Hasil Pembahasan Program .....	37
4.3 Tampilan Program.....	38
4.3.1 Halaman Utama.....	39
4.3.2 Halaman Regristasi .....	40
4.3.3 Halaman Deteksi Kerusakan .....	41
4.3.4 Halaman Hasil Deteksi Kerusakan .....	42
4.3.5 Halaman Cetak Hasil Deteksi Kerusakan .....	43
4.3.6 Halaman Login Administrator .....	44
4.3.7 Halaman administrator .....	45
4.4 Halaman Ganti Login administrator.....	46
4.3.7 Halaman Atribut.....	47
4.3.7 Halaman input Atribut .....	48
4.3.7 Halaman Edit Atribut .....	49
4.3.7 Halaman Indikasi .....	50
4.3.7 Halaman Input Indikasi .....	51
4.3.7 Halaman Edit Indikasi.....	52
4.3.7 Halaman Kerusakan .....	53
4.3.7 Halaman Input Kerusakan .....	54
4.3.7 Halaman Edit Kerusakan .....	55
4.3.7 Halaman Sub Atribut .....	56
4.3.7 Halaman Sub Atribut .....	57
4.3.7 Halaman Nilai Kedekatan .....	58
4.3.7 Halaman Input Nilai Kedekatan.....	59
4.3.7 Halaman Hasil Deteksi (admin).....	60
4.3.7 Halaman Detail Hasil Deteksi.....	61
4.3.7 Testing Pengujian.....	62
4.3.7 Hasil Uji .....	63

**BAB V SIMPULAN DAN SARAN**

5.1 Simpulan .....	64
5.2 Saran.....	65

**DAFTAR PUSTAKA**

**LAMPIRAN**



## DAFTAR GAMBAR

	<b>Halaman</b>
Gambar 2.1 CBR Cycle (Aamondt dan Plaza, 1994.....	6
Gambar 2.2 Metode Pengembangan Model <i>Waterfall</i> .....	11
Gambar 2.3 Kamera DSLR Canon EOS 700D .....	24
Gambar 2.4 Kamera DSLR Canon EOS 70D .....	25
Gambar 2.54 Kamera DSLR Canon EOS 1100D .....	26
Gambar 3.4 Rancangan Tampilan Gambar Index .....	34
Gambar 3.5 Rancangan Tampilan Home Administrator.....	34
Gambar 3.7 Rancangan Tampilan Daftar Indikasi.....	35
Gambar 3.8 Rancangan Tampilan input indikasi .....	36
Gambar 3.9 Rancangan tampilan Edit indikasi .....	36
Gambar 3.10 Rancangan Tampilan Daftar Kerusakan .....	37
Gambar 3.11 Rancangan input kerusakan .....	37
Gambar 3.5 Rancangan Tampilan Edit Kerusakan .....	38
Gambar 4.1 Halaman utama website .....	33
Gambar 4.2 Halaman Registrasi .....	33
Gambar 4.3 Halaman deteksi kerusakan .....	34
Gambar 4.4 Halaman hasil deteksi kerusakan .....	34
Gambar 4.5 Halaman Cetak hasil kerusakan .....	35
Gambar 4.6 Halaaman Login Administrator.....	35
Gambar 4.7 Halaman Administrator.....	36
Gambar 4.8 Halaman Ganti Login administrator.....	36
Gambar 4.9 Halaman Atribut.....	37
Gambar 4.10 Halaman Input Atribut .....	37
Gambar 4.11 Halaman Edit Atribut .....	38
Gambar 4.12 Halaman Indikasi.....	38
Gambar 4.13 Halaman Input Indikasi .....	39
Gambar 4.4 Halaman Edit Indikasi.....	39
Gambar 4.15 Halaman Kerusakan .....	40

Gambar 4.16 Halaman Input Kerusakan .....	40
Gambar 4.17 Halaman Edit Kerusakan .....	41
Gambar 4.18 Halaman Sub Atribut.....	41
Gambar 4.19 Halaman Input Sub Atribut .....	42
Gambar 4.20 Halaman Nilai Kedekatan .....	42
Gambar 4.21 Halaman Nilai Input Kedekatan.....	43
Gambar 4.22 Halaman Hasil Deteksi (Admin).....	43
Gambar 4.23 Halaman Detail Hasil Deteksi .....	44

## DAFTAR TABEL

	<b>Halaman</b>
Tabel 2.1 Simbol diagram Konteks.....	8
Tabel 2.2 Simbol-simbol <i>Data Flow Diagram</i> (DFD).....	9
Table 2.3 Simbol ERD.....	14
Tabel 3.1 Sub Atribut Lensa .....	18
Tabel 3.2 Sub Atribut Shutter Block.....	18
Tabel 3.3 Sub Atribut Blitz (Flash).....	19
Tabel 3.4 Sub Atribut LCD .....	19
Tabel 3.5 Sub Atribut Battery .....	19
Tabel 3.6 Kasus Lama.....	19
Tabel 3.7 Bobot Atribut .....	20
Tabel 3.8 Kedekatan Nilai Atribut Lensa .....	20
Tabel 3.9 Kedekatan Nilai Atribut Shutter Block.....	21
Tabel 3.10 Kedekatan Nilai Atribut Blitz (Flash).....	21
Tabel 3.11 Kedekatan Nilai Atribut LCD .....	21
Tabel 3.12 Kedekatan Nilai Atribut Battery .....	21
Tabel 3.15 Rancangan Struktur Tabel admin.....	28
Tabel 3.16 Rancangan Struktur Kerusakan .....	29
Tabel 4.2 Rancangan Struktur Tabel Indikasi.....	29
Tabel 3.17 Rancangan Struktur Tabel Similarity kasus.....	29
Tabel 3.18 Rancangan Struktur Tabel Atribut.....	30
Tabel 3.19 Rancangan Struktur Tabel Reg .....	30
Tabel 3.20 Rancangan Struktur Tabel Kedekatan.....	31
Tabel 3.21 Rancangan Tabel Kasus laama .....	31
Tabel 3.22 Rancangan Tabel Detail kasus lama .....	32
Tabel 3.23 Rancangan Struktur Tabel Similitari .....	32
Tabel 3.24 Rancangan Struktur Tabel Sub atribut .....	32
Tabel 3.5 Relasi Antar Tabel .....	33



# BAB I PENDAHULUAN

## 1.1 Latar Belakang Masalah

Dewasa ini ilmu pengetahuan dan teknologi berkembang dengan cepat, terutama dalam bidang teknologi informatika. Teknologi komputer banyak diminati oleh masyarakat untuk memenuhi kebutuhan mereka, dengan menggunakan komputer para pengguna (*user*) dapat menyelesaikan urusan lebih cepat dan lebih efisien.

Kamera digital khususnya yang berjenis SLR (Single Lens Reflex) makin digemari oleh kalangan anak-anak muda jaman sekarang. Kamera digital banyak yang menyebar di masyarakat maka semakin banyak pula masalah tentang Kamera digital yang muncul, terutama kerusakan yang terjadi pada Kamera digital, misalnya hasil foto pada gambar bergaris, hasil foto berbayang, hasil foto tidak fokus, hasil foto tidak tersimpan di memory, dan lain-lain. Sebagian besar masyarakat masih awam tentang kerusakan Kamera digital, untuk itu dibutuhkan suatu sistem yang dapat membantu mendiagnosa kerusakan Kamera digital tersebut.

Salah satu rancangan sistem yang dapat mendiagnosa kamera digital dengan menggunakan metode *Case Base Reasoning* (CBR). *Case Base Reasoning* (CBR) merupakan salah satu cabang penalaran komputer berbasis kasus, metode ini bertujuan untuk menyelesaikan suatu kasus baru dengan cara mengadaptasi solusi-solusi yang terdapat pada kasus-kasus sebelumnya yang mirip dengan kasus yang baru. Ide dasar dari *Case base Reasoning* (CBR) adalah bahwa manusia seringkali merujuk kepada pengalaman sebelumnya jika ada suatu masalah yang dituangkan ke dalam komputer untuk memberi solusi pemecahan berdasarkan pengalaman-pengalaman yang ada. CBR dapat diimplementasi di berbagai bidang, salah satunya dalam komputer dalam hal ini sistem pendeteksi kerusakan Kamera digital.

Berdasarkan latar belakang permasalahan tersebut diatas, penulis ingin merancang sebuah aplikasi berbasis web, sehingga bisa diaplikasikan dan diakses masyarakat khususnya orang-orang yang sedang mengalami masalah atau membutuhkan informasi tentang kerusakan kamera digital secara online. Penulis memberi judul skripsi ini “Rancang Bangun Sistem Pendiagnosa Kerusakan Pada Kamera Menggunakan Metode *Case Based Reasoning (CBR)*”.

### **1.2 Perumusan Masalah**

Berdasarkan uraian latar belakang diatas, ditemukan perumusan masalah yaitu Bagaimana mengembangkan sistem pendiagnosa untuk mendiagnosa kerusakan kamera digital menggunakan metode *Case Based Reasoning (CBR)*.

### **1.3 Batasan Masalah**

Agar pembahasan dalam laporan tugas akhir ini dapat lebih terarah maka penulisan memberikan batasan masalah yaitu :

- a. Aplikasi ini hanya mendiagnosa kerusakan kamera digital DSLR merk Canon dengan Tipe (Canon EOS 700D, Canon EOS 70D, Canon EOS 1100D).
- b. Sistem pendiagnosa menggunakan metode *Case Based Reasoning (CBR)* dengan perhitungan Algoritma Nearest Neighbour.
- c. Pembangunan sistem menggunakan Macromedia Dreamweaver CS5, menggunakan skrip PHP dan *database* yang digunakan adalah MySQL.

### **1.4 Tujuan**

Adapun tujuan dalam penelitian ini adalah sebagai berikut :

1. Membangun dan menghasilkan suatu sistem pendiagnosa yang dapat mendiagnosa kerusakan kamera digital.
2. Menerapkan metode *Case Based Reasoning (CBR)* pada sistem pendiagnosa yang dapat mendiagnosa kerusakan kamera digital.

## 1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut :

1. Dapat digunakan oleh masyarakat luas sebagai tahap awal pendiagnosa kerusakan kamera digital.
2. Dapat digunakan sebagai bahan untuk kemajuan teknologi dalam bidang sistem pakar.

## 1.6 Sistematika Penulisan

### **BAB I            PENDAHULUAN**

Bab ini menjelaskan Latar Belakang, Perumusan Masalah, Batasan Masalah, Tujuan dan Manfaat dan Sistematika Penulisan.

### **BAB II           LANDASAN TEORI**

Bab ini membahas tentang tinjauan terhadap objek yang diteliti dan berisi tentang teori-teori tentang sistem pakar, serta metode-metode yang digunakan yang berkaitan dengan topik penelitian.

### **BAB III          METODOLOGI PENELITIAN**

Menjelaskan diagram alir pemecahan masalah, metode pengumpulan data, metode pengembangan perangkat lunak, analisis sistem yang di usulkan, desain global sistem yang diusulkan, konteks diagram, rancangan *database*, dan rancangan *interface*.

### **BAB IV          HASIL DAN PEMBAHASAN**

Bab ini menyajikan hasil tampilan program, penjelasan dan pembahasan.

### **BAB V           KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan pembahasan tentang hasil yang telah diperoleh dan saran-saran yang memungkinkan untuk pengembangan skripsi ini.

## **DAFTAR PUSTAKA**

## **LAMPIRAN**

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 *Case Based Reasoning***

*Case Based Reasoning* (CBR), adalah proses pemecahan masalah baru berdasarkan solusi dari masalah masa lalu yang sama.

*Case Based Reasoning* (CBR) adalah sebuah pendekatan yang menggunakan kasus-kasus lama / pengalaman untuk memahami dan memecahkan masalah baru. Pendekatan CBR terdiri dari menciptakan pengetahuan dasar (atau database) berisi kasus-kasus masa lalu (produk). Mendefinisikan kasus baru (konsep), mengambil kasus serupa dengan kasus baru, dan menyesuaikan solusi dari kasus diambil untuk kasus baru. (Muzid : 2008)

Sedangkan menurut Eri Irawan (2009) *Case Based Reasoning* (CBR) adalah salah satu metode untuk membangun sistem pakar dengan pengambilan keputusan dari kasus yang baru dengan berdasarkan solusi dari kasus-kasus sebelumnya. Konsep dari metode case based reasoning ditemukan dari ide untuk menggunakan pengalaman-pengalaman yang terdokumentasi untuk menyelesaikan masalah yang baru. Para decision maker kebanyakan menggunakan pengalaman-pengalaman dari problem solving terdahulu untuk menyelesaikan masalah yang dihadapi sekarang.

*Case Based Reasoning* (CBR) merupakan metode yang digunakan untuk mengimplementasikan sistem diagnosa komputer ke dalam aplikasi di dunia nyata. CBR juga dapat digunakan untuk menganalisa suatu masalah sesuai dengan kasus yang dihadapi dan untuk selanjutnya mengklasifikasikan kasus tersebut berdasarkan pada pengalaman masa lalu pengklasifikasian. Kelebihan dari CBR yaitu memungkinkan penggunaan contoh kasus masa lalu untuk mengakuisisi pengetahuan dan akhirnya diketahui pokok permasalahannya. Selain itu CBR juga dapat mencari solusi dari permasalahan tersebut berdasarkan dari pengalaman kasus masa lalu sehingga segala permasalahan dapat diselesaikan untuk selanjutnya kasus serta solusinya disimpan untuk kemudian dapat digunakan

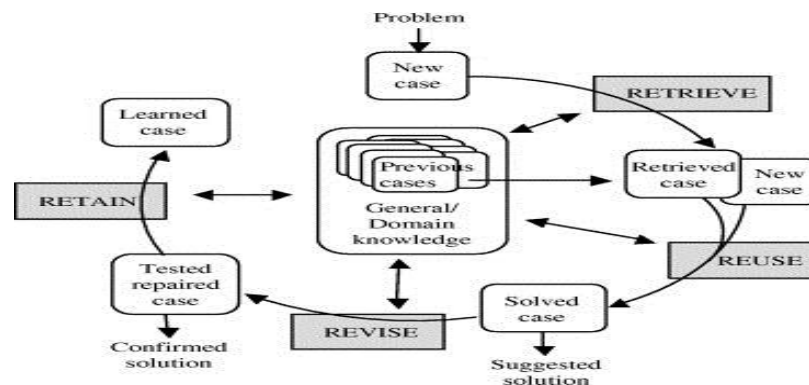


kembali untuk memecahkan kasus baru, jika kasus tersebut hampir sama atau mungkin sama dengan kasus terdahulu.

CBR Cycle menurut Aamont & Plaza (1994), secara keseluruhan model CBR Cycle dapat digambarkan dengan proses sebagai berikut :

- RETRIEVE, merupakan proses untuk mendapatkan kembali kasus terdahulu yang serupa dengan kasus yang sedang dihadapi.
- REUSE, merupakan proses untuk menggunakan kembali informasi dan pengetahuan dalam kasus terdahulu untuk menyelesaikan masalah yang dihadapi.
- REVISE, merupakan proses memperbaiki solusi yang telah ada sebelumnya.
- RETAIN, merupakan proses penyimpanan kasus baru dan solusinya untuk digunakan dalam menyelesaikan kasus berikutnya.

Keempat proses di atas akan terus dilakukan ketika menghadapi kasus baru. Model CBR tersebut dapat disajikan pada Gambar 2.1.



Gambar 2.1 CBR Cycle (Aamondt & Plaza, 1994)

## 2.2 Algoritma Nearest Neighbor

*Algoritma nearest neighbor* adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur-fitur yang ada. (Kusrini dan Lutfi, 2009).

Adapun rumus untuk melakukan perhitungan kedekatan antara dua kasus adalah sebagai berikut :

$$\text{Similarity (T,S)} = \frac{\sum_{i=1}^n f(T_i, S_i) \times W_i}{W_i}$$

Keterangan :

T = kasus baru

S = kasus yang ada dalam penyimpanan

N = jumlah atribut dalam setiap kasus

I = atribut individu antara 1 s.d. n

f = fungsi similarity untuk fitur I dalam kasus T dan kasus S

w = bobot yang diberikan pada atribut ke-i

## 2.3 Perancangan Sistem

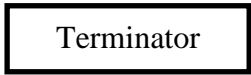


### 2.3.1 Diagram Konteks

Diagram Konteks merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data-aliran data utama menuju dan dari sistem. Diagram tersebut tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan, begitu entitas-entitas eksternal serta aliran data-aliran data menuju dan dari sistem diketahui penganalisis dari wawancara dengan *user* dan sebagai hasil analisis dokumen (Dhamidin, 2008).

Diagram Konteks menggarisbawahi sejumlah karakteristik penting dari suatu sistem (Dhamidin, 2008):

1. Kelompok pemakai, pihak yang akan memberikan data ke sistem
2. Data, apa saja yang diterima/dihasilkan sistem dari/ke dunia luar
3. Penyimpanan data, tempat sistem harus memberi informasi atau laporan
4. Batasan, yang membedakan antara sistem dan lingkungan

Tabel 2.1. Simbol Diagram Konteks

Simbol Diagram Konteks	Keterangan
	Pihak-pihak yang berada di luar sistem, tetapi secara langsung berhubungan dengan sistem dalam hal memberi data atau menerima informasi
	Didalam diagram konteks, berisi mengenai sistem yang akan dibuat
	Berisi data atau informasi yang mengalir dari satu pihak ke sistem dan sebaliknya

### 2.3.2 DFD (*Data Flow Diagram*)

*Data Flow Diagram* (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi (Roger S.Pressman, 2002).

Pada saat informasi mengalir melalui perangkat lunak, dimodifikasi oleh suatu deretan transformasi. Diagram alir data/ DFD (*data flow diagram*) adalah sebuah teknis grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari *input* menjadi *output* (Roger S.Pressman, 2002).


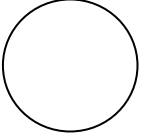
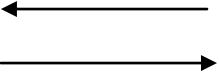
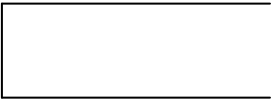
Berikut ini adalah aturan-aturan pembuatan (DFD) :

1. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan antara *external entity* dengan *entity* lainnya secara langsung.
2. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan *data store* yang satu dengan *data store* yang lainnya secara langsung.
3. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan *data store* dengan *external entity* secara langsung.

4. Setiap proses harus memiliki data *flow* yang masuk dan juga data *flow* yang keluar.

Berikut adalah simbol-simbol DFD dalam Ayu (2010) :

Tabel 2.2 Simbol-simbol *Data Flow Diagram* (DFD)

Simbol	Keterangan
	<i>External Entity</i> Simbol ini digunakan untuk menggambarkan asal atau tujuan data
	Proses Simbol ini menunjukan proses pengolahan atau tranformasi data
	<i>Data Flow</i> Simbol ini digunakan untuk menggambarkan aliran data yang berjalan
	<i>Data Store</i> Simbol ini menggambarkan <i>data Flow</i> yang sudah disimpan atau diarsipkan

Jenis-jenis DFD (*Data Flow Diagram*) dalam Ayu (2010) :

1. *Context Diagram* (CD)

Jenis pertama *Context Diagram*, adalah *data flow diagram* tingkat atas (DFD *Top Level*), yaitu diagram yang paling tidak detail, dari sebuah sistem informasi yang menggambarkan aliran-aliran data ke dalam dan ke luar sistem dan ke dalam dan ke luar entitas-entitas eksternal. (CD menggambarkan sistem dalam satu lingkaran dan hubungan dengan entitas luar. Lingkaran tersebut menggambarkan keseluruhan proses dalam sistem).

Beberapa hal yang harus diperhatikan dalam menggambar CD :

- a. Terminologi sistem
- b. Batas Sistem adalah batas antara “daerah kepentingan sistem”

- c. Lingkungan Sistem adalah segala sesuatu yang berhubungan atau mempengaruhi sistem tersebut
- d. *Interface* adalah aliran yang menghubungkan sebuah sistem dengan lingkungan sistem tersebut

## 2. Diagram Level $n$ / *Data Flow Diagram Levelled*

Dalam diagram  $n$  DFD dapat digunakan untuk menggambarkan diagram fisik maupun diagram diagram logis. Dimana Diagram Level  $n$  merupakan hasil pengembangan dari *Context Diagram* ke dalam komponen yang lebih detail tersebut disebut dengan *top-down partitioning*. Jika kita melakukan pengembangan dengan benar, kita akan mendapatkan DFD-DFD yang seimbang.

### a. DFD Fisik

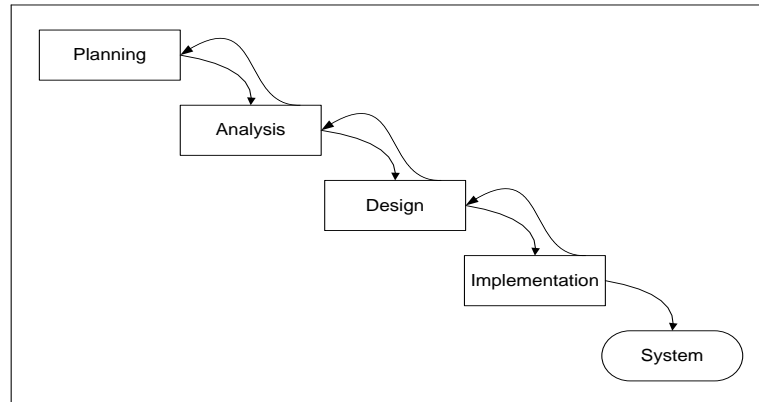
Adalah representasi grafik dari sebuah sistem yang menunjukkan entitas-entitas internal dan eksternal dari sistem tersebut, dan aliran-aliran data ke dalam dan keluar dari entitas-entitas tersebut.

### b. DFD Logis

Adalah representasi grafik dari sebuah sistem yang menunjukkan proses-proses dalam sistem tersebut dan aliran-aliran data ke dalam dan ke luar dari proses-proses tersebut. Kita menggunakan DFD logis untuk membuat dokumentasi sebuah sistem informasi karena DFD logis dapat mewakili logika tersebut, yaitu apa yang dilakukan oleh sistem tersebut, tanpa perlu menspesifikasi dimana, bagaimana, dan oleh siapa proses-proses dalam sistem tersebut dilakukan.

## 2.4 Metode Pengembangan Perangkat Lunak Menggunakan Metode *System Development Life Cycle model Waterfall*

Pada metode penelitian ini dilakukan rekayasa perangkat lunak yang digunakan adalah model *Waterfall* seperti pada gambar berikut ini:



(Sumber : Alan Dennis, Barbara H Wixom. 2003)

Gambar 2.2 Metode Pengembangan Model *Waterfall*

Keterangan:

1. *Planning (Perencanaan)*

Tahap perencanaan merupakan proses penting untuk mengetahui mengapa sistem harus dibuat dan menentukan bagaimana cara membangun sistem tersebut. Langkah pertama dari proses tersebut adalah dengan mengidentifikasi peluang apakah dapat memberikan kemungkinan biaya rendah tetapi menghasilkan keuntungan.

2. *Analysis (Analisis)*

Analisis sistem dilakukan untuk memberikan jawaban pertanyaan siapa yang akan menggunakan sistem. Apa yang akan dilakukan oleh sistem, dimana dan kapan sistem tersebut digunakan. Pada tahap ini pembuat sistem akan melakukan observasi dan pengamatan terhadap sistem yang lama, kemudian mengidentifikasi, memanfaatkan dan mengembangkan peluang, dan membangun konsep untuk sebuah sistem baru.

3. *Design (perancangan)*

Tahap perancangan dilakukan untuk menetapkan bagaimana sistem akan dioperasikan. Hal ini berkaitan dengan menentukan perangkat keras, perangkat lunak, jaringan, tampilan program, *form* dan laporan yang akan dipakai. Selain itu perlu juga menspesifikasi program, database dan file yang dibutuhkan.

#### 4. *Implementation*

Merupakan tahap berikutnya untuk menerjemahkan data atau pemecahan masalah yang telah dirancang ke dalam bahasa pemrograman komputer yang telah ditentukan. Semua tahap ini desain perangkat lunak sebagai sebuah program lengkap atau unit program.

#### 5. *System*

Tahapan ini, merupakan hasil sistem yang telah dibuat dalam bentuk perangkat lunak yang telah dipasang dan digunakan, termasuk didalamnya proses pemeliharaan dan perbaikan kesalahan. Perangkat lunak yang telah selesai dibuat dapat mengalami perubahan-perubahan atau penambahan sesuai dengan permintaan *user* atau perubahan sistem.

### 2.5 Basis Data

*Database* adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu. (Harianto Kristanto, 2003: 3)

Dengan memahami pengertian di atas, maka istilah basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (kalaupun ada maka kerangkapan data maka kerangkapan data tersebut harus seminimal mungkin dan terkontrol (*controlled redudancy*), data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali, data dapat digunakan oleh satu atau lebih program-program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan sedemikian rupa sehingga proses penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol (Edhy Sutanta, 2004).

Istilah-istilah yang digunakan dalam basis data:

- 1) *File* : merupakan kumpulan dari atribut *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda-beda dalam data *value*-nya.
- 2) *Record* : merupakan kumpulan dari elemen-elemen yang saling berhubungan atau berkaitan menginformasikan tentang *entry* secara lengkap.
- 3) *Field* : merupakan sekumpulan tanda-tanda yang berbentuk kesatuan tersendiri, merupakan bagian terkecil dari *record* dan bentuknya unik dijadikan *field* kunci yang dapat mewakili *record*-nya.
- 4) *Entity* : merupakan tempat kejadian atau konsep yang informasikan direkam.

Sutanta (2004), mengatakan bahwa perancangan basis data dan model data secara umum dapat dibagi menjadi beberapa kelompok, yaitu:

1. Model data berbasis objek
2. Model data berbasis *record*
3. Model data fisik
4. Model data konseptual

Dan dari model-model data tersebut, model data yang sering digunakan pada umumnya adalah model data berbasis objek dan model data berbasis *record*.

#### 1. Model Data Berbasis Objek (*Object Based Logical Model*)

Model data berbasis objek menggunakan konsep entitas, atribut dan hubungan antar entitas. Terdiri dari :


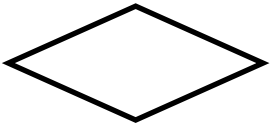
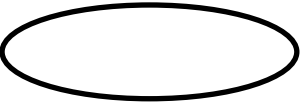
- a. *Entity Relationship Model*
- b. *Binary Model*
- c. *Semantic Data Model*
- d. *Infological Model*

Jenis model data ini yang paling populer dan sering digunakan adalah *Entity Relationship Model*, yaitu : Model Relasi-Entitas yang pada hakekatnya perwujudan dari model relasional dalam bentuk diagram, yaitu E-R Diagram. *Domain* data disebut juga sebagai himpunan entitas, diwakili oleh diagram



kotak. *Field* data atau atribut diwakili oleh diagram lingkaran atau ellips. Hubungan atau relasi antar domain diwakili oleh jajaran-genjang.

Tabel 2.3. Simbol ERD

Simbol ERD	Nama Simbol
	Domain Data
	Relasi antara domain data
	Atribut dari domain data

## 2. Model Data Berbasis *Record* (*Record-Based Logical Models*)

Model ini berdasarkan pada *record* untuk menjelaskan kepada *user* tentang hubungan *logic* antar data dalam basis data. Terdiri dari :

1. Model Relasional (*Relational Model*)
2. Model Hirarkis (*Hierarchical Model*)
3. Model Jaringan (*Network Model*)

Jenis model data ini yang paling populer dan sering digunakan adalah *Relational Model*, yaitu: Model data yang diciptakan berdasarkan teori-relasional seperti *relational algebra*, dan *relational calculus*. Salah seorang pencetus awal dari basis data relasional adalah E.F.Codd yang juga telah menciptakan serangkaian operasi matematika relasional terhadap model data relasional.

Sutanta (2004), mengatakan bahwa pada prinsipnya model data relasional dapat di-representasikan dalam bentuk tabel data, dimana:

- a. satu tabel mewakili satu “domain” data atau *entity*, bila direkam merupakan satu file yang hanya memiliki satu tipe *record* saja, setiap *record* adalah baris
- b. setiap *record* terdiri atas beberapa *field* (atribut) atau *tuple*, atau kolom
- c. jumlah *tuple* / *field* pada setiap *record* sama

- d. setiap *record* memiliki atribut kunci utama (*primary key*) yang unik dan dapat dipakai untuk mengenali satu *record*
- e. *record* dapat diurutkan menurut kunci utama
- f. hubungan antara *domain* dinyatakan dalam bentuk relasi, ada tiga kemungkinan relasi antar *domain* yaitu: relasi satu-satu (*one-to-one relation*), relasi satu-banyak (*one-to-many relation*), relasi banyak-banyak (*many-to-many relation*)

## 2.6 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. 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.

## 2.7 HTML ( Hyper Text Markup Language )

HTML merupakan kepanjangan dari *Hyper Text Markup Language* adalah suatu bahasa yang digunakan untuk membuat halaman-halaman hypertext (hypertext page) pada internet. Dengan konsep hypertext ini, untuk membaca suatu dokumen anda tidak harus melakukannya secara urut, baris demi baris, atau halaman demi halaman. Tetapi anda tidak dapat dengan mudah melompat dari satu topik ke topik lainnya yang anda sukai, seperti halnya jika anda melakukan pada online Help

dari suatu aplikasi Windows. HTML dirancang untuk digunakan tanpa tergantung pada suatu platform tertentu (platform independent).

## **2.8 PHP ( Hypertext Preprocessor )**

PHP diciptakan oleh Rasmus Lerdorf, seorang pemrogram C yang handal. Semula PHP hanya digunakan untuk mencatat jumlah pengunjung pada homepagenya. Rasmus adalah seorang pendukung open source. Karena itulah ia mengeluarkan Personal Home Page Tools versi 1.0 secara gratis. Setelah mempelajari YACC dan GNU Bison, Rasmus menambah kemampuan PHP 1.0 dan menerbitkan PHP 2.0. PHP mudah dibuat dan cepat dijalankan, PHP dapat berjalan dalam web server yang berbeda dan dalam sistem operasi yang berbeda pula. PHP dapat berjalan di sistem operasi UNIX, Windows 98, Windows XP, Windows NT, dan Macintosh.

### **2.8.1 Sejarah**

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP bernama FI (Form Interpreted). Pada saat tersebut PHP adalah sekumpulan script yang digunakan untuk mengolah data form dari web. Perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut dan menamakannya PHP/FI, pada saat tersebut kepanjangan dari PHP/FI adalah Personal Home Page/Form Interpreter. Dengan pelepasan kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP. Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter sudah diimplementasikan dalam C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan. Pada tahun 1997, sebuah perusahaan bernama Zend, menulis ulang interpreter PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada Juni 1998 perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan nama rilis tersebut menjadi PHP 3.0. Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk

membangun aplikasi web kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi. Pada Juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi mutakhir dari PHP. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman ke arah pemrograman berorientasi objek.

### 2.8.2 Pengertian PHP

Menurut Luke Welling dan Laura Thomson(2001,p1), PHP adalah *server-side scripting language* yang didesain secara spesifik untuk web. Dalam *page HTML*, dapat dimasukkan *code* PHP yang akan dieksekusi setiap kali halaman dikunjungi. PHP *code* diterjemahkan di *web-server* dan dirubah menjadi HTML atau output lain yang akan dilihat oleh pengunjung halaman. PHP adalah bahasa pemrograman script yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software di belakang Wikipedia). PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Postnuke, Xaraya, dan lain-lain.

Kelebihan PHP :

1. PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahamanan, PHP adalah bahasa scripting yang paling mudah karena referensi yang banyak. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan

secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

Selain berbagai kelebihan PHP, PHP memiliki beberapa masalah atau kekurangan. Pertama, dari segi bahasa ia bukanlah bahasa yang ideal untuk pengembangan berskala besar. Kekurangan utama adalah tidak adanya namespace. Namespace merupakan sebuah cara untuk mengelompokkan nama variabel atau fungsi dalam susunan hirarkis.

## **2.9 Kamera Digital**

Sebuah kamera digital terdiri dari pixel-pixel berupa photodiode yang bertugas menangkap energi cahaya (photon) yang dikonversikan kedalam bentuk energi listrik. Energi listrik yang berbentuk voltase dan amplitudo diperkuat kedalam level tertentu yang kemudian diproses menjadi bentuk digital oleh perangkat yang dinamakan Analog to Digital Converter (ADC). Pixel photodiode sensitif terhadap cahaya dan mampu mengukur tingkat brightness dari cahaya itu sendiri. Karena photodiode adalah device monokrom, maka tidak mungkin mereka mengenali perbedaan dari setiap panjang gelombang cahaya yang diterima. Oleh karena itu, dibuatlah sebuah sistem penyaring warna berupa pattern-pattern mosaik yang disebut dengan Color Filter Mosaic (CFM). Saat ini terdapat banyak kamera digital profesional untuk para photographer (Giwanda, 2004:5).

Beberapa istilah penting yang umum ditemui pada kamera digital yaitu (Giwanda, 2004:10).:

1. Megapixels dan resolusi; jumlah maksimum pixel yang bisa ditangkap sebuah sensor. 1 megapixels sama saja dengan seribu pixel, makin banyak pixel yang dimiliki dapat berarti pula makin lengkap detail sebuah gambar yang terekam pada sensor. Hal ini tentu berpengaruh pada saat kita melihat pada monitor, menyimpannya maupun mencetaknya. Untuk mencetak gambar pada ukuran 40x60 inchi diperlukan 12 megapixelss atau lebih agar mendapat hasil yang optimal.

2. Bits; Didalam dunia komputer, bit (binary digit) adalah bilangan berbasis 2 yang memiliki nilai antara angka 0 atau 1 yang berkorespondensi dengan switch ON atau OFF yang menyatakan suatu nilai tertentu. Dalam 1 bit gambar, kita dapat mengatakan bahwa binary 0 menyatakan black atau gelap, atau binary 1 menyatakan white atau terang. Dalam 2 bit gambar, kita mendapatkan 4 (2) kemungkinan nilai, yaitu nilai yang mewakili warna (tone): 00 (black), 01(gray), 10 (gray) dan 11 (white). Dalam 8 bit gambar (1 byte), kita mendapatkan  $2^8 = 256$  kemungkinan nilai warna, yaitu antara range 00000000 (0) - 11111111 (255) . Gambar BMP sering disebut sebagai gambar 24 bit, sebab menyimpan informasi sebesar 8 bit pada setiap warna utama (R=256, G=256, B=256), sehingga sering disebut dengan istilah 16.7 million color (mendekati warna asli) yang didapat dari perkalian  $256 \times 256 \times 256$ .
3. Burst rate; beberapa kamera digital memiliki kemampuan menangkap beberapa frame dalam waktu yang singkat, sangat berguna pada saat mengambil gambar yang membutuhkan kecepatan tinggi seperti pertandingan sepak bola. Dipengaruhi beberapa hal seperti kecepatan processor, buffer, dan storage, bahkan lensa.
4. Shutter lag; jarak waktu jeda ketika pelepasan shutter dan selama sensor merekam gambar.
5. Buffer; setelah sebuah sensor mengexpose gambar, gambar tersebut akan diproses oleh kamera dan disimpan kedalam storage. Buffer dalam kamera merupakan jenis memory (RAM) yang secara sementara menyimpan gambar sebelum ditulis ke storage. Ini akan meningkatkan waktu antara ketika melakukan shot terutama ketika kita melakukan pemotretan dalam mode burst. Penggunaan buffer diperlukan karena kecepatan storage memory penyimpanan akhir yang lebih lambat tidak mampu mengimbangi kecepatan proses dari kamera. Saat ini hampir semua kamera digital memiliki buffer yang cukup besar yang memungkinkan kamera beroperasi dengan cepat seperti kamera film konvensional, artinya kamera dapat menyimpan gambar kedalam storage dalam sebuah background process. Lokasi buffer dalam kamera tidak selalu terspesifikasikan dengan jelas, tetapi pasti ada karena

berpengaruh terhadap kecepatan saat melakukan pemotretan terutama pada mode burst continuous. Lokasi buffer sebuah kamera pada umumnya bekerja pada saat sebelum atau sesudah image processing.

6. After Image Processing Buffer

Dengan metoda ini, gambar akan diproses terlebih dahulu baru kemudian masuk ke buffer sebelum disimpan kedalam storage. Konsekuensinya, banyaknya jumlah gambar yang disimpan dalam buffer dipengaruhi jenis format gambar yang dipilih.

7. Before Image Processing Buffer

Dalam metoda ini tidak ada pemrosesan gambar ketika gambar akan dimasukkan ke dalam buffer. Gambar yang telah diexpose oleh sensor akan langsung masuk ke buffer tanpa diproses terlebih dahulu, artinya pemilihan jenis format sebuah gambar tidak akan berpengaruh langsung terhadap kapasitas sebuah buffer.

8. Smart Buffering

Mengkombinasikan kedua metoda diatas. Seperti halnya "Before Image Processing Buffer", gambar yang baru di expose akan langsung masuk ke buffer (1), kemudian gambar akan diproses oleh image processor (2) dan dikonversi ke dalam bentuk JPEG, TIFF atau RAW. Perbedaanya, daripada menulis gambar yang telah diproses langsung ke storage, gambar hasil pemrosesan/ konversi ditulis kembali ke buffer (3). Dengan demikian image processing tidak akan terpengaruh oleh bottleneck yang ada terjadi didalam storage card. Lebih dari itu setelah gambar diproses oleh image processor (2), space yang dipakai dalam buffer akan langsung dikosongkan untuk hasil pemrosesan tahap 3, yang artinya akan memberi ruang lebih terutama untuk pemrosesan gambar yang dikonversi sebagai format JPEG. Kemudian setelah diproses di tahap 3, gambar baru akan ditulis di storage, dan tentunya akan dihapus dari buffer

### 2.9.1 Jenis-jenis Kamera

#### 1. Compact Digital

Compact digital atau Digital Pocket Camera adalah salah satu dari jenis-jenis kamera fotografi yang mudah dibawa kemana-mana dan menggunakan bentuk penyimpanan data digital dan pengambilan gambar dengan ukuran yang kecil. Hal ini merupakan kegunaan dari kamera tersebut yang didesain sepraktis mungkin dan otomatis. Sesuai dengan namanya, kamera saku mempunyai bentuk dan ukuran yang muat untuk disimpan dalam kantong baju atau celana. Hal-hal diatas adalah salah satu kelebihan pada kamera saku. Tapi yang perlu diketahui untuk kamera saku secara umum yaitu keterbatasan untuk berkreasi secara profesional, dan fungsi foto yang monoton dan sederhana.

#### 2. Prosumer

Istilah "prosumer" merupakan gabungan PROfesional dan conSUMER. Bila sebuah camera disebut sebagai model prosumer biasanya ditandai kemampuan point and shoot tapi memiliki fitur lebih canggih dibanding pocket camera antara lain seperti dimilikinya kemampuan pemakaian secara manual untuk pengaturan exposure, ISO, tersedianya format RAW image capture. Prosumer camera ditargetkan untuk konsumen yang antusias pada fotografi. Beberapa contoh prosumer camera adalah Canon PowerShot S90, PowerShot G11, Panasonic Lumix DMC-LX3 dan Nikon Coolpix 8700.

#### 3. Bridge Camera

Jenis kamera ini merupakan level yang lebih tinggi daripada kamera pocket. Perbedaanya dengan kamera saku adalah Bridge camera ini tidak full otomatis, pengguna bisa mensetting secara manual exposure / Lensa kamera bridge ini tidak bisa digonta ganti layaknya kamera DSLR. Biasanya bentuk dan ukurannya lebih besar daripada kamera pocket. Contoh jenis kamera ini adalah Fujifilm Finepix S4600, Canon PowerShot G15 dsb

#### 4. DSLR

Kamera DSLR atau Digital Single Lens Reflex consumer ini merupakan kamera yang sangat populer dan laris manis di pasaran beberapa tahun belakangan ini. Popularitas tersebut, selain dikarenakan semakin banyaknya



orang yang menyukai fotografi, fasilitas yang ditawarkan kamera jenis ini juga melebihi kamera-kamera jenis standar lainnya. Rata-rata, ketika membeli baru, maka penggunaannya akan mendapatkan lensa tambahan atau lensa kit 18-55mm yang merupakan standar dari lensa DSLR jenis consumer. Perusahaan pembuat kamera jenis ini sudah melengkapinya dengan berbagai fasilitas mulai dari pengaturan auto atau juga manual sampai dengan fitur layar LCD di bagian belakangnya yang beberapa merk dapat diputar-putar untuk mencari viewfinder dan juga telah mengadopsi teknologi touchscreen. Tentunya bagi pemula yang benar-benar ingin mendalami dunia fotografi, DSLR consumer ini sangat cocok digunakan sebagai media belajar. Sayangnya, harga yang ditawarkan untuk kamera jenis ini tergolong tinggi.

#### 5. Mirrorless

Sesuai dengan namanya kamera mirrorless (Mirrorless interchangeable lens camera) ini tidak mempunyai cermin (mirror) itulah yang menjadi perbedaan dengan kamera DSLR. Sistem cermin yang ada di kamera DSLR digunakan untuk optical viewfinder. Sehingga kamera mirrorless tidak mempunyai fitur tersebut. Kamera ini hanya mengandalkan digital viewfinder (Layar LCD / LED) untuk mengetahui obyek yang kita bidik. Itulah yang menyebabkan mirrorless camera ini ukurannya lebih kecil dari DSLR. Selain itu juga lensa kamera ini bisa diganti ganti. Pengaturan kamera ini layaknya DSLR yang bisa diatur full manual ataupun otomatis. Kualitas foto yang dihasilkan sama dengan DSLR. Contoh kamera mirrorless adalah Canon EOS-MB1, Samsung NX1000.

### 2.9.2 Kamera DSLR

Kamera DSLR merupakan kamera digital dengan format mengadopsi kamera slr film, yaitu memiliki lensa yang bisa dilepas, cermin mekanik, dan pentaprisma untuk mengarahkan sinar yang melewati lensa menuju jendela bidik atau *view finder* (Giwanda, 2004:15).

Komponen Utama Kamera DSLR dan Fungsinya Dalam kamera dslr terdapat beberapa komponen penting, diantaranya adalah (Sugiarto, 2006:18):

### 1. Body

Body disebut juga sebagai badan kamera yang merupakan bagian utama dari kamera untuk membuat gambar. Di dalam body, terdapat berbagai macam fitur yang dapat menentukan kualitas gambar. Di dalam body terdapat elemen-elemen penting pada kamera, semisal Range Finder sebagai pengukur ketajaman sebuah objek. Lalu terdapat juga View finder atau jendela bidik untuk melihat objek secara langsung yang sedang dibidik. Badan kamera juga berfungsi melakukan proses pencahayaan yang selanjutnya akan memproduksi gambar. Oleh karena itu, body kamera tidak boleh dimasuki oleh cahaya. Jika terjadi kebocoran cahaya maka akan merusak gambar yang akan dipotret.

### 2. Shutter (Rana)

Shutter atau disebut juga rana, merupakan pintu masuknya cahaya ke dalam kamera sehingga dapat mengenai sensor untuk direkam. Kecepatan shutter dalam membuka dan menutup kembali akan mempengaruhi gambar yang dihasilkan.

### 3. Lensa

Lensa adalah alat yang paling vital pada kamera. Tanpa lensa, kamera tidak dapat menangkap atau merekam gambar. Dalam fotografi, lensa berfungsi untuk memfokuskan cahaya dan mengantarkannya ke dalam badan kamera. Di bagian luar lensa biasanya terdapat tiga cincin, yaitu cincin panjang focus (untuk lensa jenis variabel), cincin diafragma, dan cincin fokus. Pada permukaan lensa juga dilengkapi sebuah lapisan yang dibuat dari uap logam (coating). Lapisan coating berfungsi untuk menghilangkan efek flare yang didapat ketika melawan matahari. Sehingga para fotografer tidak takut memandangi matahari melalui kameranya. Coating juga berguna untuk menghilangkan efek kabur yang didapat dalam sebuah foto.

### 4. View Finder (Jendela Bidik)

Jendela bidik merupakan tempat melihat bayangan objek yang akan dipotret. Dalam jendela bidik tercantum banyak informasi dalam pemotretan, seperti penentu jarak (range finder), pilihan diafragma, shutter speed dan pencahayaan (exposure). Kamera SLR sesuai dengan namanya (Single Lens

Reflex) menggunakan sistem bidikan melalui lensa tunggal (reflex type). Mata fotografer melihat objek melalui lensa tanpa terjadi parallax. Parallax ialah keadaan dimana fotografer tidak melihat secara akurat indikasi keberadaan objek melalui lensa, sehingga ada bagian yang hilang ketika foto dicetak.

### **2.9.2.1 Canon EOS 700D**

Kamera Canon EOS 700D dirancang dengan tampilan yang elegan serta mudah dipegang maka kamera ini cocok untuk semua orang. Baik anda penggelut di dunia fotografi atau hanya sebagai hobi.

EOS 700D menawarkan performa dasar yang penuh dan kokoh yang jelas merupakan salah satu yang terbaik di setiap tingkat awal DSLR dengan kualitas gambarnya yang tinggi, fungsi yang bervariasi dari AFLive View dan perekaman film. LCD II Tampilan Jernih Sudut yang Bervariasi dengan kemampuan layar sentuh kapasitif serta desain baru dengan Pemutar Mode yang dapat berotasi 360 derajat dan Penyaring Kreatif akan menambah inspirasi dan ekspresi yang kreatif.

Spesifikasi kamera dslr ini mempunyai resolusi 18 megapiksel, 9 titik AF semua tipe silang, Kecepatan pemotretan bersambungan (Sekitar 5fps), Monitor LCD II Tampilan Jernih dengan Sudut yang Bervariasi, Full HD 1080, serta LSD layar sentuh membuat anda bisa semaksimal mungkin menyetel dan melihat hasil-hasil yang sudah anda ambil



Gambar 2.3 Kamera DSLR Canon EOS 700D

### 2.9.2.2 Canon EOS 70D

70D adalah DSLR termutakhir yang memudahkan pengguna mengubah suasana sehari-hari menjadi foto yang ekspresif. Dilengkapi dengan AF CMOS Piksel Ganda dengan deteksi fasa sensor teknologi AF yang memungkinkan kecepatan pemfokusan cepat yang impresif selama pemotretan foto dan film Live View.

Canon EOS 70D dengan Spesifikasi : AF CMOS Piksel Ganda, Pemrosesan gambar DIGIC 5+ & sensor CMOS APS-C 20.2M, AF tipe silang 19 titik & 7.0fps Pemotretan Rentetan Bersambungan. Kamera DSLR Canon saat ini telah berkembang dengan baik di Indonesia. Banyaknya para fotografi muda yang bermunculan dan para master fotografi yang ingin mengembangkan bakatnya dalam membidik banyak objek, memberikan banyak perubahan drastis pada seri-seri terbaru kamera DSLR Canon. Salah satunya adalah pada kamera Canon EOS 70D, seri ini sangat cocok untuk anda para master fotografi yang ingin lebih berkembang lagi karena kamera ini telah dilengkapi dengan resolusi 20.2MP APS-C CMOS Sensor dan dibantu dengan adanya sistem baru yaitu DIGIC 5 Image Processor. Tak hanya itu pengembangan sistem lain juga akan memberikan kenyamanan saat memotret anda seperti adanya Dual Pixel CMOS AF with Live View yang memberikan akses lebih mudah pada tampilan layar saat setelah memotret. Kualitas Full HD 1080p Video tak kalah menariknya pada kamera Canon EOS 70D yang dapat memberikan hasil maksimal dan jernih untuk sebuah perekaman gambar.



Gambar 2.4 Kamera DSLR Canon EOS 70D

### 2.9.2.3 Canon EOS 1100D

Canon menerapkan strategi yang agak unik dalam melakukan segmentasi produk DSLR mereka. Bila dulu kita menganggap EOS tiga digit (350D, 400D, 450D dst) adalah kamera pemula, maka kini EOS empat digitlah yang jadi DSLR kelas basic dari Canon yang sesungguhnya. Bila tiga tahun lalu EOS 1000D didesain begitu basic dan sederhana, kini pada penerusnya yaitu 1100D terdapat sejumlah peningkatan seperti resolusi sensor, jumlah titik AF, modul metering, prosesor dan yang terpenting adalah HD movie. Beberapa fitur dari 1100D pun tampak overlap dengan 550D/600D dalam arti banyak kemiripan fitur antara kamera kelas basic (empat digit) dengan Canon kelas tiga digit. Maka itu wajar kalau saya memprediksi EOS 1100D bakal mengulang sukses dengan meraih penjualan yang tinggi, terutama bila kita tidak terlalu membutuhkan segala kelebihan yang ada di EOS tiga digit.

Spesifikasi Canon EOS 1100D : sensor CMOS 12 MP, prosesor Digic IV, kemampuan merekam HD movie 720p, kemampuan metering dengan 63 zone (fokus, warna dan luminance), memakai modul AF dengan 9 titik (satu yang ditengah cross type), mencapai ISO 6400, kecepatan burst 3 fps, LCD 2,7 inci, resolusi 230 ribu piksel, HDMI out.



Gambar 2.54 Kamera DSLR Canon EOS 1100D

## **BAB III METODE PENELITIAN**

### **3.1 Penerapan Metode Case Base Reasoning**

Metode case based reasoning merupakan metode yang menerapkan 4 tahapan proses, yaitu retrieve, reuse, revise, dan retain. Cara kerja sistem secara umum berpedoman pada basis pengetahuan yang dimiliki oleh sistem yang bersumber dari kasus-kasus yang pernah ditangani oleh seorang pakar. Kemudian dihitung tingkat kemiripannya dengan kasus baru yang dimasukkan pengguna. Berdasarkan tingkat kemiripan kasus inilah sistem akan mengeluarkan kesimpulan Paket yang diambil.

#### **3.1.1 Proses Retrive**

Proses Retrive merupakan proses pencarian kemiripan kasus baru dengan kasus yang lama. Pencarian kemiripan antara kasus baru dengan kasus lama dilakukan dengan cara mencocokkan fakta yang diinputkan oleh pengguna dengan fakta yang ada pada basis pengetahuan. Pada proses retrieve ini akan dilakukan pembobotan dengan menggunakan metode Nearest Neighbour.

##### **a) Pemberian data range**

- 1) Data sub atribut Lensa dapat dilihat pada tabel berikut:

Tabel 3.1 Sub Atribut Lensa

No	Nama Range
1.	Auto fokus Error
2.	Lensa buram
3.	Lensa Normal
4.	Tidak bisa mengatur bukaan Diafragma

- 2) Data sub atribut Shutter Block dapat dilihat pada tabel berikut:

Tabel 3.2 Sub Atribut Shutter Block

No	Nama Range
1.	Shutter Block Normal
2.	Shutter Block Rusak

3) Data sub atribut Blitz (Flash) dapat dilihat pada tabel berikut:

Tabel 3.3 sub atribut Blitz (Flash)

No	Nama Range
1.	Blitz (Flash) normal
2.	Blitz (Flash) tidak nyala

4) Data sub atribut LCD dapat dilihat pada tabel berikut:

Tabel 3.4 sub atribut LCD

No	Nama Range
1.	LCD/Layar Bergaris
2.	LCD/Layar Mati
3.	LCD/Layar Normal

5) Data sub atribut Battery dapat dilihat pada tabel berikut:

Tabel 3.5 sub atribut Battery

No	Nama Range
1.	Battery cepat habis
2.	Battery normal

#### b) Kasus Lama

Tabel 3.6 Kasus Lama

No	Nama Kasus	Lensa	Shutter Block	Blitz (Flash)	LCD	Battery
1	Kasus 1 (Battery Rusak)	Lensa Normal	Shutter Block Normal	Blitz (Flash) Normal	LCD/Layar Normal	Battery Cepat Habis
2	Kasus 2 (Blitz (Flash) Rusak)	Lensa Normal	Shutter Block Normal	Blitz (Flash) Tidak Nyala	LCD/Layar Normal	Battery Normal
3	Kasus 3 (LCD/Layar Rusak)	Lensa Normal	Shutter Block Normal	Blitz (Flash) Normal	LCD/Layar Bergaris	Battery Normal
4	Kasus 4 (Lensa Rusak)	Tidak bisa mengatur bukaan Diafragma	Shutter Block Normal	Blitz (Flash) Normal	LCD/Layar Normal	Battery Normal
5	Kasus 5 (Shutter Block rusak)	Lensa Normal	Shutter Block rusak	Blitz (Flash) Normal	LCD/Layar Normal	Battery Normal

### c) Bobot Atribut

Bobot antara satu atribut dengan atribut yang lainnya pada atribut tujuan dapat didefinisikan dengan nilai berbeda. Didefinisikan bobot untuk tiap-tiap atribut seperti tampak pada tabel 3.7 dibawah ini.

Tabel 3.7 Bobot atribut

Atribut	Bobot
Keypad	1
Layar	1
Touchscreen	1
Speaker	1
Flexible	1

### d) Kedekatan Nilai Antar Atribut

Kedekatan antara nilai-nilai dalam atribut juga perlu didefinisikan. Lebih jelasnya dapat dilihat pada urutan tabel dibawah ini.

#### 1) Kedekatan nilai atribut Lensa

Tabel 3.8 Kedekatan Nilai Atribut Lensa

Nilai1	Nilai2	Kedekatan
Auto fokus Error	Auto fokus Error	1
Auto fokus Error	Lensa Buram	0.9
Auto fokus Error	Lensa Normal	0.5
Auto fokus Error	Tidak bisa mengatur bukaan Diafragma	0.6
Lensa Buram	Auto fokus Error	0.7
Lensa Buram	Lensa Buram	1
Lensa Buram	Lensa Normal	0.5
Lensa Buram	Tidak bisa mengatur bukaan Diafragma	0.7
Lensa Normal	Auto fokus Error	0.9
Lensa Normal	Lensa Buram	0.5
Lensa Normal	Lensa Normal	1
Lensa Normal	Tidak bisa mengatur bukaan Diafragma	0.6
Tidak bisa mengatur bukaan Diafragma	Auto fokus Error	0.5
Tidak bisa mengatur bukaan Diafragma	Lensa Buram	0.5
Tidak bisa mengatur bukaan Diafragma	Lensa Normal	0.3
Tidak bisa mengatur bukaan Diafragma	Tidak bisa mengatur bukaan Diafragma	1



## 2) Kedekatan nilai atribut Shutter Block

Tabel 3.9 Kedekatan Nilai Atribut Shutter Block

Nilai1	Nilai2	Kedekatan
Shutter Block Normal	Shutter Block Normal	1
Shutter Block Normal	Shutter Block Rusak	0.5
Shutter Block Rusak	Shutter Block Normal	0.5
Shutter Block Rusak	Shutter Block Rusak	1

## 3) Kedekatan nilai atribut Blitz (Flash)

Tabel 3.10 Kedekatan Nilai Atribut Blitz (Flash)

Nilai1	Nilai2	Kedekatan
Blitz (Flash) Normal	Blitz (Flash) Normal	1
Blitz (Flash) Normal	Blitz (Flash) Tidak Nyala	0.5
Blitz (Flash) Tidak Nyala	Blitz (Flash) Normal	0.5
Blitz (Flash) Tidak Nyala	Blitz (Flash) Tidak Nyala	1

## 4) Kedekatan nilai atribut LCD

Tabel 3.11 Kedekatan Nilai Atribut LCD

Nilai1	Nilai2	Kedekatan
LCD/Layar Bergaris	LCD/Layar Bergaris	1
LCD/Layar Bergaris	LCD/Layar Mati	0.6
LCD/Layar Bergaris	LCD/Layar Normal	0.3
LCD/Layar Mati	LCD/Layar Bergaris	0.9
LCD/Layar Mati	LCD/Layar Mati	1
LCD/Layar Mati	LCD/Layar Normal	0.5
LCD/Layar Normal	LCD/Layar Bergaris	0.4
LCD/Layar Normal	LCD/Layar Mati	0.3
LCD/Layar Normal	LCD/Layar Normal	1

## 5) Kedekatan nilai atribut Battery

Tabel 3.12 Kedekatan Nilai Atribut Battery

Nilai1	Nilai2	Kedekatan
Battery Cepat Habis	Battery Cepat Habis	1
Battery Cepat Habis	Battery Cepat Normal	0.5
Battery Cepat Normal	Battery Cepat Habis	0.5
Battery Cepat Normal	Battery Cepat Normal	1

## Contoh Kasus

Lensa	: Auto fokus Error
Shutter Block	: Shutter Block Normal
Blitz (Flash)	: Blitz (Flash) Normal
LCD	: LCD/Layar Normal
Battery	: Battery Normal

## 1. Menghitung kedekatan kasus baru dengan kasus 1 Diketahui :

- a. Kedekatan nilai atribut Lensa (Auto fokus Error Dengan Lensa Normal) : 0.5
- b. Bobot atribut Lensa: 1
- c. Kedekatan nilai atribut Shutter Block ( Shutter Block Normal Dengan Shutter Block Normal) : 1
- d. Bobot atribut Lensa: 1
- e. Kedekatan nilai atribut Blitz (Flash) (Blitz (Flash) Normal Dengan Blitz (Flash) Normal) : 1
- f. Bobot atribut Blitz (Flash): 1
- g. Kedekatan nilai atribut LCD (LCD/Layar Normal Dengan LCD/Layar Normal) : 1
- h. Bobot atribut LCD: 1
- i. Kedekatan nilai atribut Battery (Battery Normal Dengan Battery Cepat Habis) : 0.5
- j. Bobot atribut Battery: 1
- k. Dihitung :

$$\text{Jarak} = \frac{(a+b)+(c+d)+(e+f)+(g+h)+(i+j)}{b+d+f+h+j}$$

$$\text{Jarak} = \frac{(0,5+1)+(1+1)+(1+1)+(1+1)+(0,5+1)}{1+1+1+1+1}$$

$$\text{Jarak} = \frac{4}{5}$$

$$\text{Jarak} = 0.8$$

## 2. Menghitung kedekatan kasus baru dengan kasus 2 Diketahui :

- a. Kedekatan nilai atribut Lensa (Auto fokus Error Dengan Lensa Normal) : 0.5

- b. Bobot atribut Lensa: 1
- c. Kedekatan nilai atribut Shutter Block ( Shutter Block Normal Dengan Shutter Block Normal) : 1
- d. Bobot atribut Lensa: 1
- e. Kedekatan nilai atribut Blitz (Flash) (Blitz (Flash) Normal Dengan Blitz (Flash) Tidak Nyala) : 0.5
- f. Bobot atribut Blitz (Flash): 1
- g. Kedekatan nilai atribut LCD (LCD/Layar Normal Dengan LCD/Layar Normal) : 1
- h. Bobot atribut LCD: 1
- i. Kedekatan nilai atribut Battery (Battery Normal Dengan Battery Normal) : 1
- j. Bobot atribut Battery: 1
- k. Dihitung :

$$\text{Jarak} = \frac{(a+b)+(c+d)+(e+f)+(g+h)+(i+j)}{b+d+f+h+j}$$

$$\text{Jarak} = \frac{(0.5+1)+(1+1)+(0.5+1)+(1+1)+(1+1)}{1+1+1+1+1}$$

$$\text{Jarak} = \frac{4}{5}$$

$$\text{Jarak} = 0.8$$

- 3. Menghitung kedekatan kasus baru dengan kasus 3 Diketahui :
  - a. Kedekatan nilai atribut Lensa (Auto fokus Error Dengan Lensa Normal) : 0.5
  - b. Bobot atribut Lensa: 1
  - c. Kedekatan nilai atribut Shutter Block ( Shutter Block Normal Dengan Shutter Block Normal) : 1
  - d. Bobot atribut Lensa: 1
  - e. Kedekatan nilai atribut Blitz (Flash) (Blitz (Flash) Normal Dengan Blitz (Flash) Normal) : 1
  - f. Bobot atribut Blitz (Flash): 1
  - g. Kedekatan nilai atribut LCD (LCD/Layar Normal Dengan LCD/Layar Bergaris) : 0.4

- h. Bobot atribut LCD: 1
- i. Kedekatan nilai atribut Battery (Battery Normal Dengan Battery Normal) : 1
- j. Bobot atribut Battery: 1
- k. Dihitung :

$$\text{Jarak} = \frac{(a+b)+(c+d)+(e+f)+(g+h)+(i+j)}{b+d+f+H+j}$$

$$\text{Jarak} = \frac{(0,5+1)+(1+1)+(1+1)+(0,4+1)+(1+1)}{1+1+1+1+1}$$

$$\text{Jarak} = \frac{3,9}{5}$$

$$\text{Jarak} = 0,78$$

- 4. Menghitung kedekatan kasus baru dengan kasus 4 Diketahui :
  - a. Kedekatan nilai atribut Lensa (Auto fokus Error Dengan Tidak bisa mengatur bukaan Diafragma) : 0.6
  - b. Bobot atribut Lensa: 1
  - c. Kedekatan nilai atribut Shutter Block ( Shutter Block Normal Dengan Shutter Block Normal) : 1
  - d. Bobot atribut Lensa: 1
  - e. Kedekatan nilai atribut Blitz (Flash) (Blitz (Flash) Normal Dengan Blitz (Flash) Normal) : 1
  - f. Bobot atribut Blitz (Flash): 1
  - g. Kedekatan nilai atribut LCD (LCD/Layar Normal Dengan LCD/Layar Normal) : 1
  - h. Bobot atribut LCD: 1
  - i. Kedekatan nilai atribut Battery (Battery Normal Dengan Battery Normal) : 1
  - j. Bobot atribut Battery: 1
  - k. Dihitung :

$$\text{Jarak} = \frac{(a+b)+(c+d)+(e+f)+(g+h)+(i+j)}{b+d+f+H+j}$$

$$\text{Jarak} = \frac{(0,6+1)+(1+1)+(1+1)+(1+1)+(1+1)}{1+1+1+1+1}$$

$$\text{Jarak} = \frac{4,6}{5}$$

$$\text{Jarak} = 0.92$$

5. Menghitung kedekatan kasus baru dengan kasus 5 Diketahui :
- a. Kedekatan nilai atribut Lensa (uto fokus Error Dengan Lensa Normal) : 0.5
  - b. Bobot atribut Lensa: 1
  - c. Kedekatan nilai atribut Shutter Block (Shutter Block Normal Dengan Shutter Block rusak) : 0.5
  - d. Bobot atribut Lensa: 1
  - e. Kedekatan nilai atribut Blitz (Flash) (Blitz (Flash) Normal Dengan Blitz (Flash) Normal) : 1
  - f. Bobot atribut Blitz (Flash): 1
  - g. Kedekatan nilai atribut LCD (LCD/Layar Normal Dengan LCD/Layar Normal) : 1
  - h. Bobot atribut LCD: 1
  - i. Kedekatan nilai atribut Battery (Battery Normal Dengan Battery Normal) : 1
  - j. Bobot atribut Battery: 1
  - k. Dihitung :

$$\text{Jarak} = \frac{(a+b)+(c+d)+(e+f)+(g+h)+(i+j)}{b+d+f+H+j}$$

$$\text{Jarak} = \frac{(0.5+1)+(0.5+1)+(1+1)+(1+1)+(1+1)}{1+1+1+1+1}$$

$$\text{Jarak} = \frac{4}{5}$$

$$\text{Jarak} = 0.8$$

Berdasarkan penelusuran dari kasus sebelumnya, kasus baru memiliki kemiripan dengan kasus 4 dengan nilai similary 0.92 dan kasus baru yang ditemukan merupakan kerusakan Lensa Rusak.

### **3.1.2 Proses Reuse**

Perhitungan diatas kasus yang memiliki bobot kemiripan paling tinggi adalah kasus 2 yaitu sebesar 0.92. Pada proses *reuse*, solusi yang diberikan adalah solusi dengan bobot kemiripan kasus lama dengan kasus baru yang paling tinggi. Hasil perhitungan dengan bobot menunjukkan tingkat kepercayaan kurang dari 80% jadi harus dilakukan proses *revise*.

### **3.1.3 Proses Revise**

Proses *revise* adalah proses peninjauan kembali kasus dan solusi yang diberikan jika pada proses *retrieve* sistem tidak dapat memberikan hasil diagnosa yang tepat. Tetapi jika ternyata setelah dilakukan proses perhitungan dan tidak ada kasus yang mirip dengan kasus baru tersebut maka dilakukan proses *revise*. Informasi berupa masukan fakta pada kasus baru yang tidak ditemukan kemiripannya dengan basis pengetahuan tersebut akan ditampung pada suatu tabel khusus (tabel *revise*) yang selanjutnya akan dievakuasi dan diperbaiki kembali oleh pakar untuk menemukan solusi yang tepat.

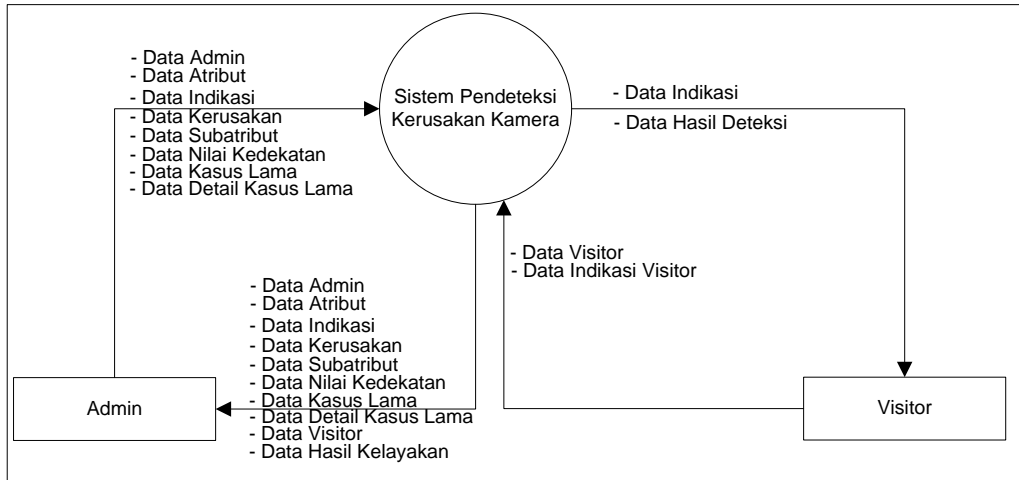
### **3.1.4 Proses Retain**

Setelah proses *revise* selesai dan sudah ditemukan solusi yang benar-benar tepat oleh pakar kerusakan handphone barulah admin mulai menambah aturan dengan memasukkan data kasus baru yang sudah ditemukan solusinya tersebut ke dalam basis pengetahuan yang nantinya dapat digunakan untuk kasus berikutnya yang memiliki permasalahan yang sama. Proses inilah yang disebut dengan proses *retain*.

## **3.2 Perancangan**

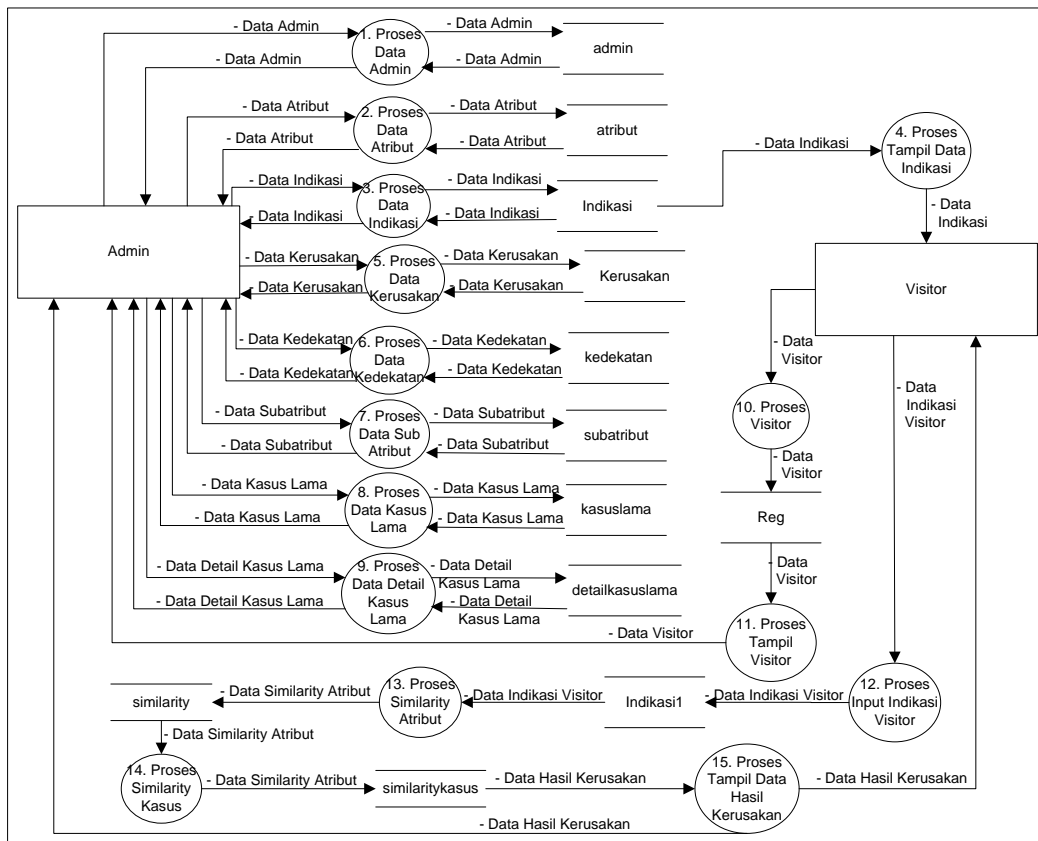
Pada tahapan ini proses desain sistem dibagi berdasarkan kebutuhan-kebutuhan berdasarkan analisis ke dalam bentuk yang mudah dimengerti pengguna. Perancangan dimulai dari penyusunan konteks diagram dan *data flow diagram* (DFD)

### 3.2.1 Diagram Konteks



Gambar 3.1 Diagram Konteks

### 3.2.2 Data flow diagram level 0



Gambar 3.2 Data Flow Diagram Level 0

### 3.4 Rancangan Struktur Database

Struktur *database* sistem yang diusulkan pada penelitian ini sebagai berikut:

a. Struktur Tabel Admin

Tabel ini digunakan untuk menyimpan data login admin administrator

Nama Database : cbr  
 Nama Tabel : admin  
 Kunci Utama : idadmin  
 Kunci Tamu : -  
 Media Penyimpanan : Harddisk

Tabel 3.13 Rancangan Struktur Tabel Admin

Field Name	Type Data	Size	Keterangan
idadmin*	Int	5	Id admin
username	Varchar	32	Username
password	Varchar	32	Password

b. Struktur Tabel Kerusakan

Tabel ini digunakan untuk menyimpan data kerusakan kamera

Nama Database : cbr  
 Nama Tabel : kerusakan  
 Kunci Utama : idkerusakan  
 Kunci Tamu : -  
 Media Penyimpanan : Harddisk

Tabel 3.14 Rancangan Struktur Tabel Kerusakan

Field Name	Type Data	Size	Keterangan
idkerusakan*	Int	5	Id kerusakan
kerusakan	Varchar	20	kerusakan
ket	Text		Keterangan kerusakan

c. Struktur Tabel Indikasi

Tabel ini digunakan untuk menyimpan data indikasi kerusakan kamera

Nama Database : cbr  
 Nama Tabel : indikasi



Kunci Utama : idindikasi  
 Kunci Tamu : -  
 Media Penyimpanan : Harddisk

Tabel 3.15 Rancangan Struktur Tabel Indikasi

Field Name	Type Data	Size	Keterangan
idindikasi*	Int	5	Id indikasi
indikasi	Varchar	30	indikasi kerusakan

## d. Struktur Tabel Indikasi1

bel ini digunakan untuk menyimpan data indikasi kerusakan kamera pengunjung.

Nama Database : cbr  
 Nama Tabel : indikasi1  
 Kunci Utama : idindikasi1  
 Kunci Tamu : idreg,idindikasi,idatribut  
 Media Penyimpanan : Harddisk

Tabel 3.16 Rancangan Struktur Tabel Indikasi1

Field Name	Type Data	Size	Keterangan
idindikasi1*	Int	5	Id indikasi pengunjung
idreg**	Int	5	Id pengunjung
idindikasi**	Int	5	Id indikasi
idatribut**	Int	5	Id atribut

## e. Struktur Tabel Similaritykasus

Tabel ini digunakan untuk menyimpan data kemiripan kasus

Nama Database : cbr  
 Nama Tabel : similaritykasus  
 Kunci Utama : -  
 Kunci Tamu : idreg,idkasus  
 Media Penyimpanan : Harddisk

Tabel 3.17 Rancangan Struktur Tabel Similaritykasus

Field Name	Type Data	Size	Keterangan
idreg**	Int	5	Id pengunjung

idkasus**	Int	5	Id kasus lama
similarity	Float		Hasil nilai similarity

#### f. Struktur Tabel Atribut

Tabel ini digunakan untuk menyimpan data atribut

Nama Database : cbr  
 Nama Tabel : atribut  
 Kunci Utama : idatribut  
 Kunci Tamu : -  
 Media Penyimpanan : Harddisk

Tabel 3.18 Rancangan Struktur Tabel Atribut

Field Name	Type Data	Size	Keterangan
idatribut*	Int	5	Id atribut
atribut	Varchar	20	Atribut
bobot	Varchar	5	Bobot atribut

#### g. Struktur Tabel Reg

Tabel ini digunakan untuk menyimpan data pengunjung.

Nama Database : cbr  
 Nama Tabel : reg  
 Kunci Utama : idreg  
 Kunci Tamu : -  
 Media Penyimpanan : Harddisk

Tabel 3.19 Rancangan Struktur Tabel Reg

Field Name	Type Data	Size	Keterangan
idreg*	Int	5	Id pengunjung
nama	Varchar	30	Nama pengunjung
alamat	Int	2	Alamat pengunjung

#### h. Struktur Tabel Kedekatan

Tabel ini digunakan untuk menyimpan data kedekatan antar indikasi

Nama Database : cbr

Nama Tabel : kedekatan  
 Kunci Utama : idkedekatan  
 Kunci Tamu : idindikasi  
 Media Penyimpanan : Harddisk

Tabel 3.20 Rancangan Struktur Tabel Kedekatan

Field Name	Type Data	Size	Keterangan
idkedekatan*	Int	5	Id kedekatan
idindikasi**	Int	5	Id indikasi
idindikasi1**	Int	5	Id indikasi
kedekatan	Varchar	5	Nilai kedekatan

## i. Struktur Tabel Kasuslama

Tabel ini digunakan untuk menyimpan data kasus lama

Nama Database : cbr  
 Nama Tabel : kasuslama  
 Kunci Utama : idkasus  
 Kunci Tamu : idkerusakan  
 Media Penyimpanan : Harddisk

Tabel 3.21 Rancangan Struktur Tabel Kasuslama

Field Name	Type Data	Size	Keterangan
idkasus*	Int	5	Id kasus lama
nama	Int	5	Nama kasus lama
alamat	Int	5	alamat kasus lama
idkerusakan**	Int	5	Id kerusakan

## j. Struktur Tabel Detailkasuslama

Tabel ini digunakan untuk menyimpan data detail indikasi kasus lama

Nama Database : cbr  
 Nama Tabel : detailkasuslama  
 Kunci Utama : iddetail  
 Kunci Tamu : idkasus, idindikasi, idatribut  
 Media Penyimpanan : Harddisk

Tabel 3.22 Rancangan Struktur Tabel Detailkasuslama

Field Name	Type Data	Size	Keterangan
iddetail*	Int	5	Primary key
idkasus**	Int	5	Id kasus lama
idindikasi**	Int	5	Id indikasi
idatribut**	Int	5	Id atribut

## k. Struktur Tabel Similarity

Tabel ini digunakan untuk menyimpan data nilai kedekatan atribut

Nama Database : cbr  
 Nama Tabel : similarity  
 Kunci Utama : -  
 Kunci Tamu : idreg, idkasus, idatribut  
 Media Penyimpanan : Harddisk

Tabel 3.23 Rancangan Struktur Tabel Similarity

Field Name	Type Data	Size	Keterangan
idreg**	Int	5	Id pengunjung
idkasus**	Int	5	Id Kasus lama
idatribut**	Int	5	Id atribut
similarity	Float		Nilai similarity

## l. Struktur Tabel Subatribut

Tabel ini digunakan untuk menyimpan data sub atribut

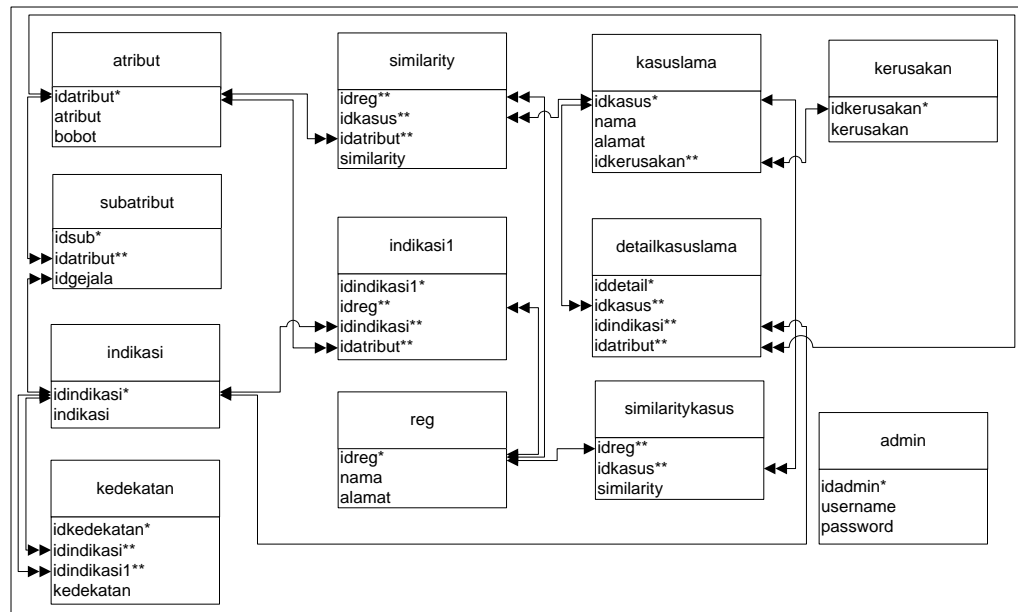
Nama Database : cbr  
 Nama Tabel : subatribut  
 Kunci Utama : idsub  
 Kunci Tamu : idatribut, idindikasi  
 Media Penyimpanan : Harddisk

Tabel 3.24 Rancangan Struktur Tabel Subatribut

Field Name	Type Data	Size	Keterangan
idsub*	Int	5	Id subatribut
idatribut**	Int	5	Id atribut
idindikasi**	Int	5	Id indikasi

### 3.5 Relasi Antar Tabel

Relasi Antar Tabel pada penelitian ini sebagai berikut :



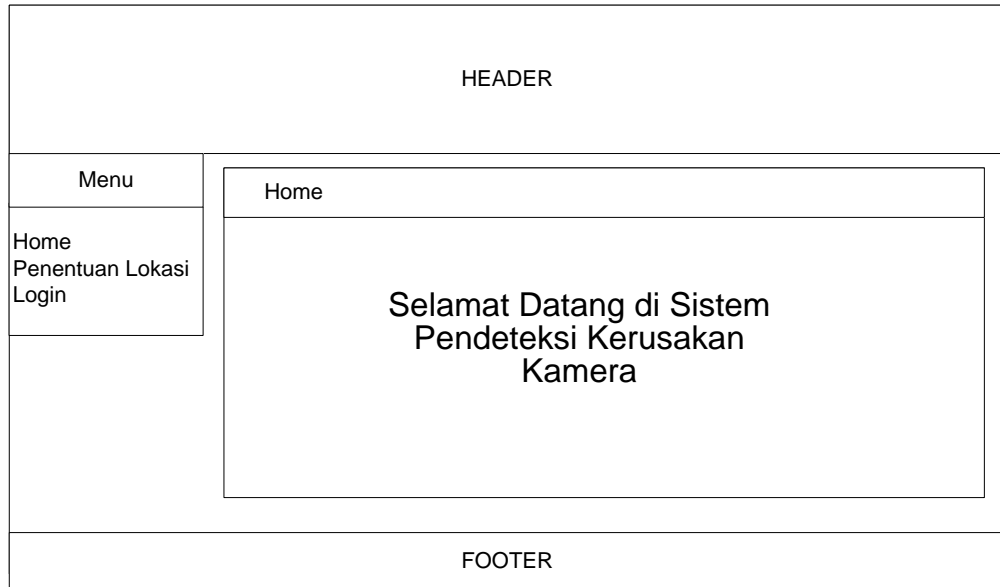
Gambar 3.3 Relasi Antar Tabel

### 3.6 Perancangan Antarmuka

Perancangan antarmuka sangat diperlukan untuk mempermudah *user* menggunakan sistem kerusakan kamera.

#### a. Halaman Index

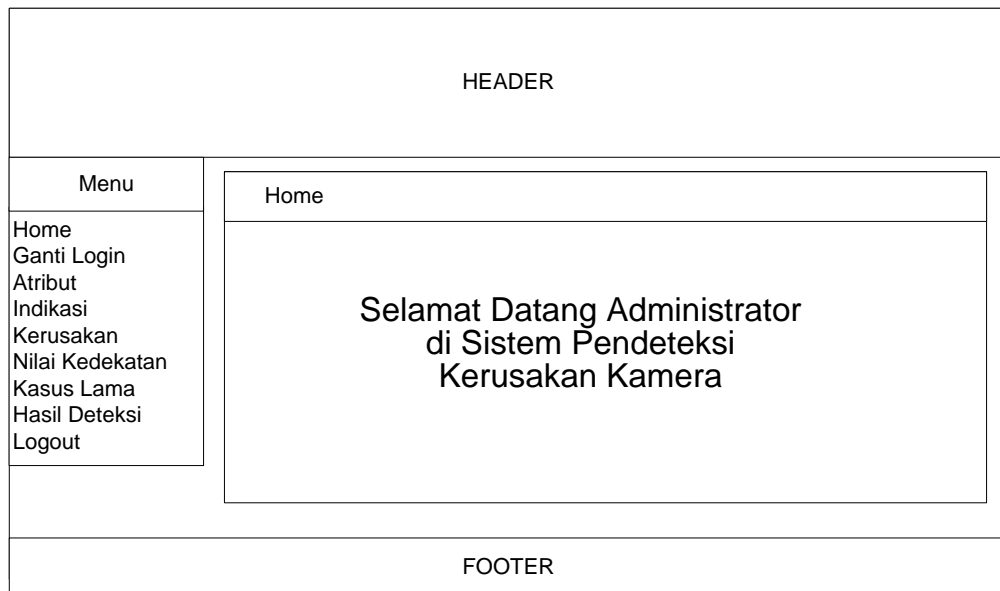
Menu *index* adalah halaman awal program. Pada halaman ini terdapat menu *home*, penentuan lokasi, dan *login*.



Gambar 3.4 Rancangan Tampilan Halaman Index

**b. Halaman Administrator**

Halaman ini akan tampil jika administrator berhasil login kedalam sistem. Dapat dilihat pada gambar berikut.



Gambar 3.5 Rancangan Tampilan Home Administrator

### c. Halaman Ganti Login

Halaman ini berisi form ganti login administrator. Dapat dilihat pada gambar berikut.

HEADER	
Menu Home Ganti Login Atribut Indikasi Kerusakan Nilai Kedekatan Kasus Lama Hasil Deteksi Logout	<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;">Ganti Login</p> <p>Username <input type="text" value="xxxxx"/></p> <p>Password <input type="text" value="xxxxx"/></p> <p style="text-align: center;"> <input type="button" value="Ganti"/> <input type="button" value="Reset"/> </p> </div>
FOOTER	

Gambar 3.6 Rancangan Tampilan Ganti Login Administrator

### d. Halaman Indikasi

Halaman ini berisi daftar indikasi kerusakan kamera. Pada halaman ini terdapat menu input indikasi, edit indikasi, dan hapus indikasi. Dapat dilihat pada gambar berikut.

HEADER													
Menu Home Ganti Login Atribut Indikasi Kerusakan Nilai Kedekatan Kasus Lama Hasil Deteksi Logout	<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;">Indikasi</p> <p style="text-align: center;"><u>Input Indikasi</u></p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>No.</th> <th>Indikasi</th> <th></th> </tr> </thead> <tbody> <tr> <td>1.</td> <td>Indikasi 1</td> <td>Edit    Hapus</td> </tr> <tr> <td>2.</td> <td>Indikasi 2</td> <td>Edit    Hapus</td> </tr> <tr> <td>3.</td> <td>Indikasi 3</td> <td>Edit    Hapus</td> </tr> </tbody> </table> </div>	No.	Indikasi		1.	Indikasi 1	Edit    Hapus	2.	Indikasi 2	Edit    Hapus	3.	Indikasi 3	Edit    Hapus
No.	Indikasi												
1.	Indikasi 1	Edit    Hapus											
2.	Indikasi 2	Edit    Hapus											
3.	Indikasi 3	Edit    Hapus											
FOOTER													

Gambar 3.7 Rancangan Tampilan Daftar Indikasi

Halaman dibawah ini berisi form input Indikasi.

HEADER	
Menu	Input Indikasi
Home Ganti Login Atribut Indikasi Kerusakan Nilai Kedekatan Kasus Lama Hasil Deteksi Logout	Indikasi <input type="text"/>  <input type="button" value="Simpan"/> <input type="button" value="Reset"/>
FOOTER	

Gambar 3.8 Rancangan Tampilan Input Indikasi

Halaman dibawah ini berisi form edit Indikasi.

HEADER	
Menu	Edit Indikasi
Home Ganti Login Atribut Indikasi Kerusakan Nilai Kedekatan Kasus Lama Hasil Deteksi Logout	Indikasi <input type="text" value="xxxxxx"/>  <input type="button" value="Simpan"/> <input type="button" value="Reset"/>
FOOTER	

Gambar 3.9 Rancangan Tampilan Edit Indikasi

**e. Halaman Kerusakan**

Halaman ini berisi daftar kerusakan kamera. Pada halaman ini terdapat menu input kerusakan, edit kerusakan, dan hapus kerusakan. Dapat dilihat pada gambar berikut.



HEADER															
<table border="1"> <thead> <tr> <th>Menu</th> </tr> </thead> <tbody> <tr> <td>Home</td> </tr> <tr> <td>Ganti Login</td> </tr> <tr> <td>Atribut</td> </tr> <tr> <td>Indikasi</td> </tr> <tr> <td>Kerusakan</td> </tr> <tr> <td>Nilai Kedekatan</td> </tr> <tr> <td>Kasus Lama</td> </tr> <tr> <td>Hasil Deteksi</td> </tr> <tr> <td>Logout</td> </tr> </tbody> </table>	Menu	Home	Ganti Login	Atribut	Indikasi	Kerusakan	Nilai Kedekatan	Kasus Lama	Hasil Deteksi	Logout	<table border="1"> <thead> <tr> <th>Kerusakan</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"><u>Input Kerusakan</u></td> </tr> <tr> <td>No. Kerusakan : xxxxxx Keterangan : xxxxx</td> </tr> <tr> <td style="text-align: right;"><a href="#">Edit</a>    <a href="#">Hapus</a></td> </tr> </tbody> </table>	Kerusakan	<u>Input Kerusakan</u>	No. Kerusakan : xxxxxx Keterangan : xxxxx	<a href="#">Edit</a>    <a href="#">Hapus</a>
Menu															
Home															
Ganti Login															
Atribut															
Indikasi															
Kerusakan															
Nilai Kedekatan															
Kasus Lama															
Hasil Deteksi															
Logout															
Kerusakan															
<u>Input Kerusakan</u>															
No. Kerusakan : xxxxxx Keterangan : xxxxx															
<a href="#">Edit</a>    <a href="#">Hapus</a>															
FOOTER															

Gambar 3.10 Rancangan Tampilan Daftar Kerusakan

Halaman dibawah ini berisi form input Kerusakan.

HEADER																			
<table border="1"> <thead> <tr> <th>Menu</th> </tr> </thead> <tbody> <tr> <td>Home</td> </tr> <tr> <td>Ganti Login</td> </tr> <tr> <td>Atribut</td> </tr> <tr> <td>Indikasi</td> </tr> <tr> <td>Kerusakan</td> </tr> <tr> <td>Nilai Kedekatan</td> </tr> <tr> <td>Kasus Lama</td> </tr> <tr> <td>Hasil Deteksi</td> </tr> <tr> <td>Logout</td> </tr> </tbody> </table>	Menu	Home	Ganti Login	Atribut	Indikasi	Kerusakan	Nilai Kedekatan	Kasus Lama	Hasil Deteksi	Logout	<table border="1"> <thead> <tr> <th colspan="2">Input Kerusakan</th> </tr> </thead> <tbody> <tr> <td>Kerusakan</td> <td><input type="text"/></td> </tr> <tr> <td>Keterangan</td> <td><input type="text"/></td> </tr> <tr> <td colspan="2" style="text-align: center;"> <input type="button" value="Simpan"/> <input type="button" value="Reset"/> </td> </tr> </tbody> </table>	Input Kerusakan		Kerusakan	<input type="text"/>	Keterangan	<input type="text"/>	<input type="button" value="Simpan"/> <input type="button" value="Reset"/>	
Menu																			
Home																			
Ganti Login																			
Atribut																			
Indikasi																			
Kerusakan																			
Nilai Kedekatan																			
Kasus Lama																			
Hasil Deteksi																			
Logout																			
Input Kerusakan																			
Kerusakan	<input type="text"/>																		
Keterangan	<input type="text"/>																		
<input type="button" value="Simpan"/> <input type="button" value="Reset"/>																			
FOOTER																			

Gambar 3.11 Rancangan Tampilan Input Kerusakan

Halaman dibawah ini berisi form edit Kerusakan.

HEADER							
Menu	Edit Kerusakan						
Home Ganti Login Atribut Indikasi Kerusakan Nilai Kedekatan Kasus Lama Hasil Deteksi Logout	<table><tr><td>Kerusakan</td><td><input type="text" value="xxxxxxx"/></td></tr><tr><td>Keterangan</td><td><input type="text" value="Xxxxxxx xxxxxxxx xxxxxxxx"/></td></tr><tr><td colspan="2" style="text-align: center;"><input type="button" value="Simpan"/> <input type="button" value="Reset"/></td></tr></table>	Kerusakan	<input type="text" value="xxxxxxx"/>	Keterangan	<input type="text" value="Xxxxxxx xxxxxxxx xxxxxxxx"/>	<input type="button" value="Simpan"/> <input type="button" value="Reset"/>	
Kerusakan	<input type="text" value="xxxxxxx"/>						
Keterangan	<input type="text" value="Xxxxxxx xxxxxxxx xxxxxxxx"/>						
<input type="button" value="Simpan"/> <input type="button" value="Reset"/>							
FOOTER							

Gambar 3.13 Rancangan Tampilan Edit Kerusakan

## **BAB IV HASIL DAN PEMBAHASAN**

### **4.1 Implementasi Program**

Implementasi program adalah menjelaskan bagaimana menjalankan program aplikasi yang telah dibuat pada komputer. Aplikasi ini bisa dijalankan pada komputer *stand alone* maupun *online*. Jika ingin menjalankan aplikasi ini pada komputer yang *stand alone*, maka pada komputer tersebut harus diinstall terlebih dulu program aplikasi *web server*. Tapi jika ingin menjalankan aplikasi ini secara *online*, maka harus memiliki domain situs dan web server.

Untuk penelitian ini, penulis hanya mengimplementasikan aplikasi ini pada komputer *stand alone*. Sebagai ganti domain situs, penulis menggunakan *localhost* yang bisa dijalankan pada komputer yang *stand alone*.

Langkah-langkah menjalankan aplikasi ini adalah dengan mengakses local domain pada komputer *stand alone* dengan menggunakan alamat <http://localhost/feri/> pada *web browser*. Dengan menggunakan halaman ini pengguna akan mendapatkan tampilan utama (*home page*) situs. Selanjutnya pengguna bisa menggunakan menu-menu yang telah disediakan disitus.

Berikut merupakan gambaran singkat tentang situs dengan mengakses semua menu dan *link navigasi* yang telah disediakan pada *home page*.

### **4.2 Hasil Pembahasan Program**

Hal yang perlu diperhatikan sebelum menjalankan website pada komputer PC adalah pastikan pada komputer tersebut *Apache* telah terinstal. Untuk menjalankan program ini dapat dilakukan dengan membuka program *Mozilla Firefox* yang diambil dari menu *start* kemudian pilih menu *Mozilla Firefox*, selanjutnya jendela isian alamat browser ketikkan <http://localhost> apabila kemudian tampilan jendela informasi mengenai *xampp*, maka *xampp* berjalan

dengan baik. Setelah *xampp* berjalan, lalu ketikkan alamat website dengan mengetikkan <http://localhost/feri/>

### 4.3 Tampilan Program

#### 4.3.1 Halaman Utama

Halaman ini merupakan halaman default yang akan ditampilkan pertama kali ketika visitor atau pengunjung membuka website. Dapat dilihat pada gambar 4.1 berikut ini.



Gambar 4.1 Halaman Utama Website

#### 4.3.2 Halaman Registrasi

Pada halaman ini terdapat form registrasi untuk Deteksi Kerusakan. Dapat dilihat pada gambar 4.2 berikut ini.



Gambar 4.2 Halaman Registrasi

### 4.3.3 Halaman Deteksi Kerusakan

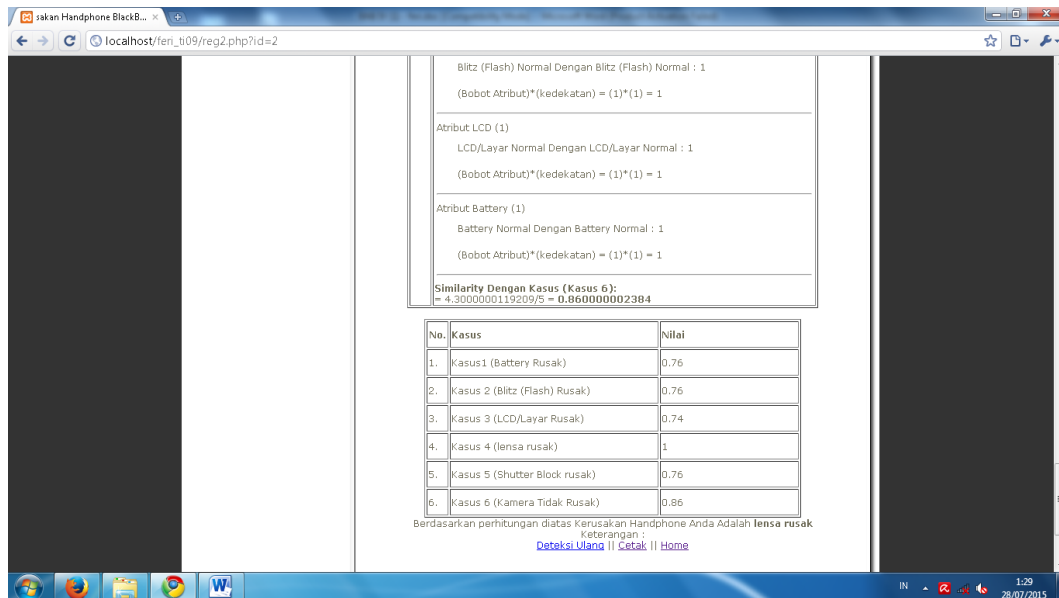
Pada halaman ini terdapat form input data Deteksi Kerusakan. Dapat dilihat pada gambar 4.3 berikut ini.



Gambar 4.3 Halaman Deteksi Kerusakan

### 4.3.4 Halaman Hasil Hasil Deteksi Kerusakan

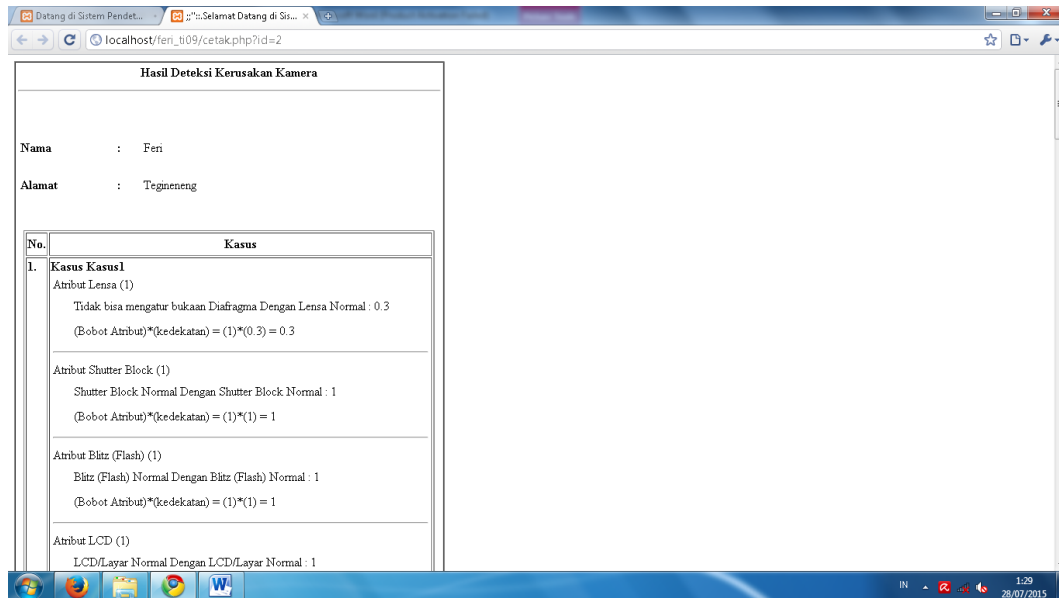
Pada halaman ini terdapat hasil Deteksi Kerusakan. Dapat dilihat pada gambar 4.4 berikut ini.



Gambar 4.4 Halaman Hasil Hasil Deteksi Kerusakan

### 4.3.5 Halaman Cetak Hasil Deteksi Kerusakan

Halaman ini digunakan untuk mencetak Hasil Deteksi Kerusakan. Dapat dilihat pada gambar 4.5 berikut ini.



Gambar 4.5 Halaman Cetak Hasil Deteksi Kerusakan

### 4.3.6 Halaman Login Administrator

Pada halaman ini terdapat form login administrator. Dapat dilihat pada gambar 4.6 berikut ini.



Gambar 4.6 Halaman Login Administrator

### 4.3.7 Halaman Administrator

Pada halaman ini dapat di akses setelah administrator berhasil login. Dapat dilihat pada gambar 4.7 berikut ini.



Gambar 4.7 Halaman Administrator

### 4.3.8 Halaman Ganti Login Administrator

Pada halaman ini terdapat form ganti login administrator. Dapat dilihat pada gambar 4.8 berikut ini.



Gambar 4.8 Halaman Ganti Login Administrator

### 4.3.9 Halaman Atribut

Pada halaman ini terdapat daftar Atribut. Dapat dilihat pada gambar 4.7 berikut ini.



Gambar 4.9 Halaman Atribut

### 4.3.10 Halaman Input Atribut

Pada halaman ini terdapat form Input Atribut. Dapat dilihat pada gambar 4.10 berikut ini.

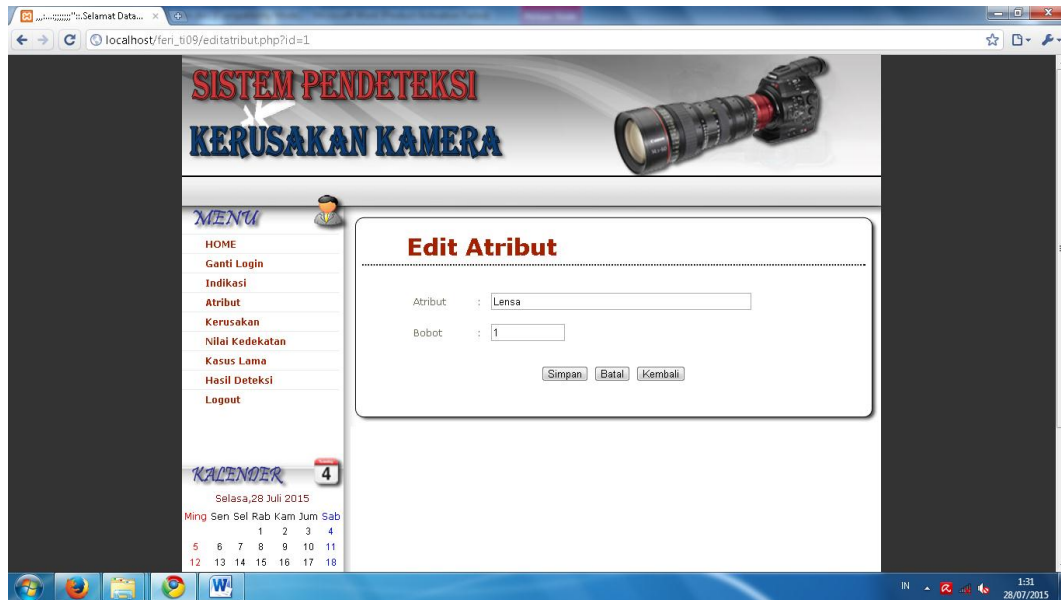


Gambar 4.10 Halaman Input Atribut



### 4.3.11 Halaman Edit Atribut

Pada halaman ini terdapat form edit atribut. Dapat dilihat pada gambar 4.11 berikut ini.



Gambar 4.11 Halaman Edit Atribut

### 4.3.12 Halaman Indikasi

Pada halaman ini terdapat daftar Indikasi. Dapat dilihat pada gambar 4.12 berikut ini.



Gambar 4.12 Halaman Indikasi

### 4.3.13 Halaman Input Indikasi

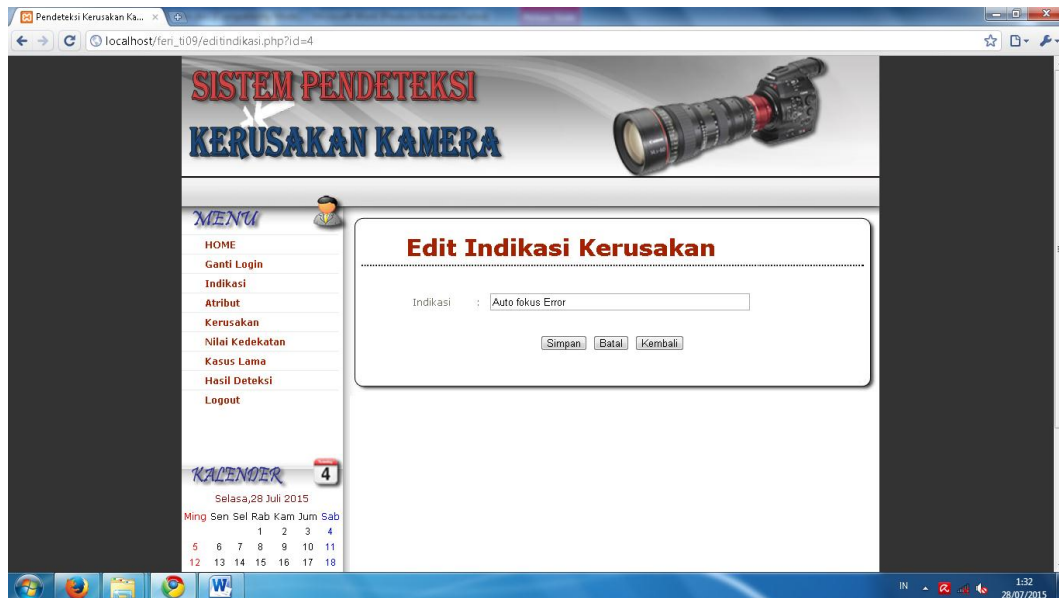
Pada halaman ini terdapat form input Indikasi. Dapat dilihat pada gambar 4.13 berikut ini.



Gambar 4.13 Halaman Input Indikasi

### 4.3.14 Halaman Edit Indikasi

Pada halaman ini terdapat form edit Indikasi. Dapat dilihat pada gambar 4.14 berikut ini.



Gambar 4.14 Halaman Edit Indikasi

### 4.3.15 Halaman Kerusakan

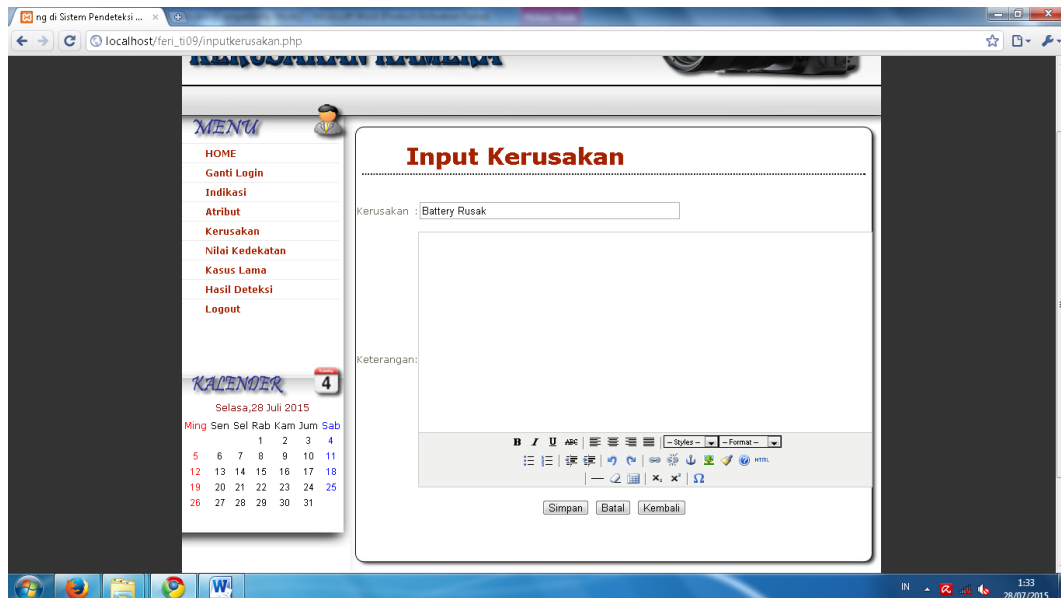
Pada halaman ini terdapat daftar Kerusakan Kamera. Dapat dilihat pada gambar 4.15 berikut ini.



Gambar 4.15 Halaman Kerusakan

### 4.3.16 Halaman Input Kerusakan

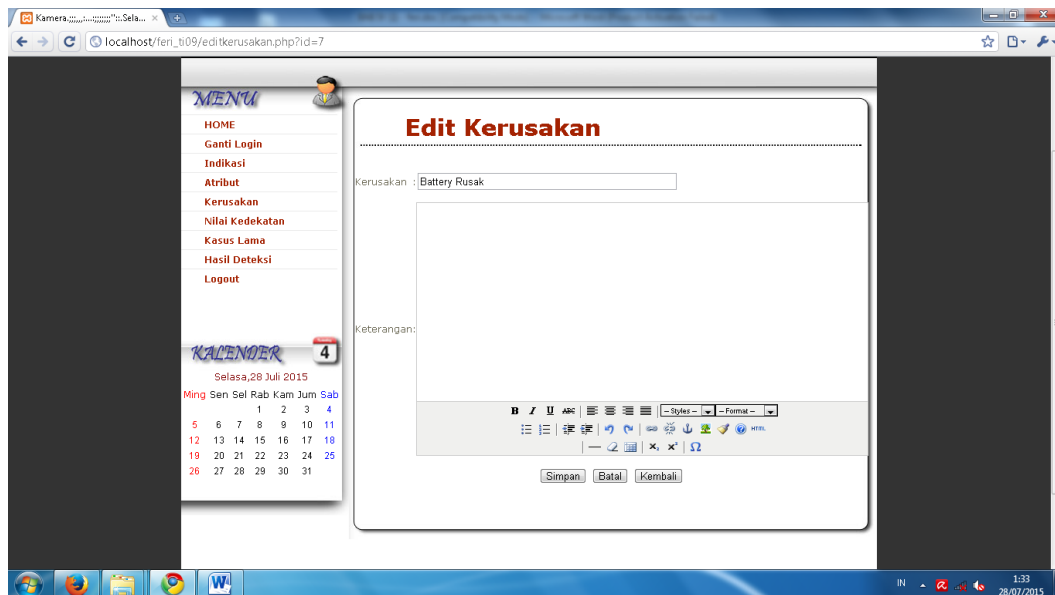
Pada halaman ini terdapat form input Kerusakan. Dapat dilihat pada gambar 4.16 berikut ini.



Gambar 4.16 Halaman Input Kerusakan

### 4.3.17 Halaman Edit Kerusakan

Pada halaman ini terdapat form edit Kerusakan. Dapat dilihat pada gambar 4.17 berikut ini.



Gambar 4.17 Halaman Edit Kerusakan

### 4.3.18 Halaman Sub Atribut

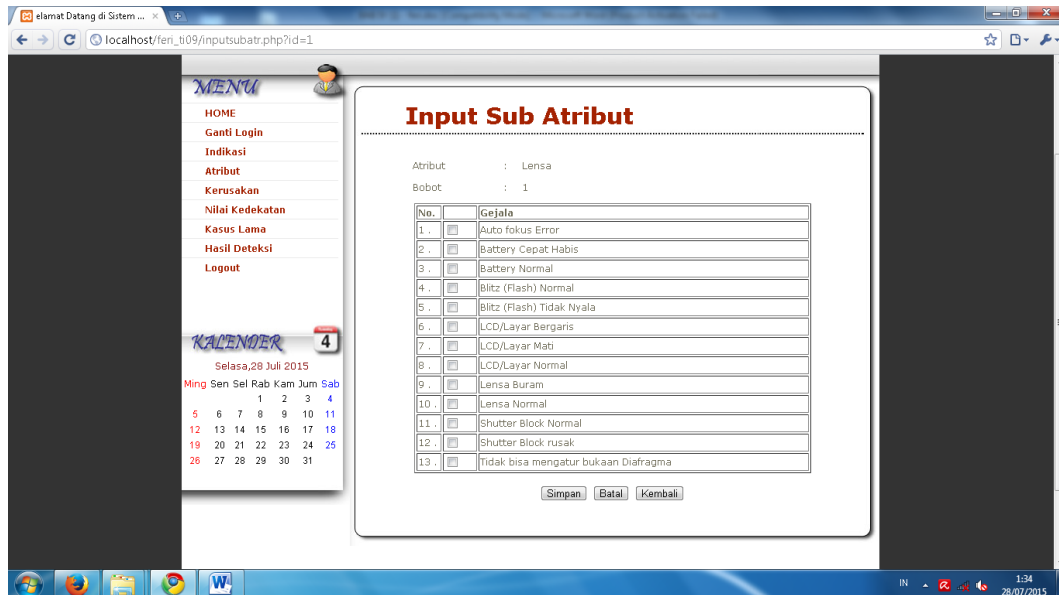
Pada halaman ini terdapat daftar Sub Atribut. Dapat dilihat pada gambar 4.18 berikut ini.



Gambar 4.18 Halaman Sub Atribut

### 4.3.19 Halaman Input Sub Atribut

Pada halaman ini terdapat form input Sub Atribut. Dapat dilihat pada gambar 4.19 berikut ini.



Gambar 4.19 Halaman Input Sub Atribut

### 4.3.20 Halaman Nilai Kedekatan

Pada halaman ini terdapat daftar nilai kedekatan antar gejala per atribut. Dapat dilihat pada gambar 4.20 berikut ini.



Gambar 4.20 Halaman Nilai Kedekatan

#### 4.3.21 Halaman Input Nilai Kedekatan

Pada halaman ini terdapat form input nilai kedekatan antar Sifat. Dapat dilihat pada gambar 4.21 berikut ini.



Gambar 4.21 Halaman Input Nilai Kedekatan

#### 4.3.22 Halaman Hasil Deteksi (Admin)

Pada halaman ini terdapat daftar hasil deteksi Deteksi Kerusakan yang menggunakan sistem ini. Dapat dilihat pada gambar 4.22 berikut ini.



Gambar 4.22 Halaman Hasil Deteksi (Admin)

### 4.3.23 Halaman Detail Hasil Deteksi

Pada halaman ini terdapat detail Hasil Deteksi. Dapat dilihat pada gambar 4.23 berikut ini.



Gambar 4.23 Halaman Detail Hasil Deteksi

## 4.4 Testing Pengujian

Pengujian perangkat lunak berfungsi untuk menentukan pencapaian tujuan berdasarkan kriteria. Dalam hal ini pengujian dilakukan dengan menghubungkan setiap entitas dari sistem sesuai dengan spesifikasi *hardware* dan *software*.

Pengujian kinerja aplikasi hanya dilakukan terhadap aplikasi yang berkaitan dengan pemrosesan *database* dalam program *dreamweaver 8* dan *database mysql*. Adapaun pengujian dilakukan dengan variasi banyaknya *database* yang harus dieksekusi. Selain itu kinerja dari perangkat lunak yang sangat terkait dengan kondisi konektivitas *database* yang digunakan. Setelah *Xampp* terinstal untuk sementara *server* yang digunakan adalah *localhost*. Pengaturan *database* dapat dilakukan melalui *phpmyadmin* yang berfungsi untuk membuat, merubah dan menghapus *database*. Dengan fasilitas ini akan memudahkan dalam pembuatan *database mysql* karena tidak menggunakan perintah (*syntax*) manual *sql*.

#### 4.5 Hasil Uji

Perangkat lunak yang diimplementasikan telah sesuai dengan spesifikasi kebutuhan pada bagian analisis dan perancangan. Hal ini dibuktikan dengan keberhasilan masing-masing subsistem melakukan apa yang menjadi spesifikasi seperti telah ditanyakan di bagian hasil pengujian diatas, sehingga proses yang terjadi telah dengan prosedur yang dispesifikasikan.

Banyaknya tabel tidak menjadi penghambat proses data. Tetapi sementara ini yang menjadi keterlambatan akses adalah waktu yang dibutuhkan untuk menyelesaikan proses semua aplikasi data selain itu dapat juga disebabkan oleh kualitas server yang digunakan. Untuk melakukan maintenance atau perawatan maka *website* ini harus memiliki *server* sendiri karena jika menggunakan *fresshosting* akan mengalami kesulitan, contohnya pada waktu membuka website akan membutuhkan waktu yang lama karena server tidak mendukung program yang dibuat. Dalam memiliki web hosting harus dipertimbangkan kapasitas yang disediakan bahasa pemrograman yang dapat mendukung *script PHP* dan *MySQL* serta mendukung *software* yang digunakan dan biaya untuk mendapatkan hosting tersebut



## **BAB V**

### **SIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil analisis masalah dan pembahasan, hasil kesimpulan dari penelitian ini adalah sebagai berikut:

- a. Sistem Pendeteksi Kerusakan Kamera Menggunakan metode Case Based Reasoning telah dilakukan secara terkomputerisasi dan dijalankan dengan sebuah website.
- b. Sistem Pendeteksi Kerusakan Kamera Menggunakan metode Case Based Reasoning berbasis Web membantu masyarakat luas dalam mendeteksi kerusakan kamera.

#### **5.2 Saran**

Sesuai dengan permasalahan yang ada pada skripsi ini, maka hanya dapat memberikan saran-saran:

- a. Sistem ini hanya memberikan tampilan sederhana, bagi yang akan mengembangkan program ini lebih lanjut diharapkan dapat dilengkapi dan disempurnakan tampilannya serta dapat memberikan tambahan fitur-fitur lainnya.
- b. Untuk menjaga suatu keamanan *database* yang ada maka disarankan agar membuat *database* tersendiri untuk *username* dan *password*.

## DAFTAR PUSTAKA

- Kusumadewi, S dan Hari Purnomo.2010.Aplikasi logika fuzzy untuk pendukung keputusan(edisi 2).Graha Ilmu, Yogyakarta.
- Jogiyanto, H.M. 2005. Analisi dan Desain Sistem Informasi, Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Edisi III. Andi Offset, Yogyakarta.
- Kadir, A.2004.Dasar aplikasi database mysql Delphi. Andi Offset, Yogyakarta.
- Sutanta, E.2004. Sistem Basis Data. Graha Ilmu, Yogyakarta
- Kristanto, A. 2004. Rekayasa Perangkat Lunak. Andi Offset, Yogyakarta.