

**SISTEM PAKAR MENDIAGNOSA PENYAKIT IKAN PATIN DENGAN METODE
FORWARD CHAINING BERBASIS WEB**

SKRIPSI



**Oleh:
Mesran Fernando Simamora
130210369**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU SOSIAL DAN HUKUM
UNIVERSITAS PUTERA BATAM
2018**

**SISTEM PAKAR MENDIAGNOSA PENYAKIT IKAN PATIN DENGAN METODE
FORWARD CHAINING BERBASIS WEB**

SKRIPSI



**Oleh:
Mesran Fernando Simamora
130210369**

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

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU SOSIAL DAN HUKUM
UNIVERSITAS PUTERA BATAM
2018**

PERNYATAAN

Dengan ini saya menyatakan bahwa:

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

Batam, 08 Febuari 2018
Yang membuat pernyataan,

Mesran Fernando simamora

HALAMAN PENGESAHAN
SISTEM PAKAR MENDIAGNOSA PENYAKIT IKAN PATIN DENGAN
METODE FORWARD CHAINING BERBASIS WEB

Mesran Fernando Simamora
130210369

SKRIPSI
Untuk memenuhi salah satu syarat guna
memperoleh gelar Sarjana

Telah disetujui oleh Pembimbing pada tanggal
seperti tertera di bawah ini

Batam, 08 febuari 2018

Algifanri Maulana, S.Si., M.MSI.

ABSTRAK

Kebutuhan akan informasi tentang penyakit ikan Patin saat ini dibutuhkan, hal ini biasanya terjadi pada petani ikan yang berlokasi pedesaan yang jauh dari perkembangan teknologi, Kurangnya informasi pengetahuan dan pemahaman dalam pengenalan suatu penyakit dapat mengakibatkan kesalahan diagnosa penyakit pada ikan Patin. Oleh sebab itu dibutuhkan peran seorang pakar dibidang perikanan sebagai tempat konsultasi. Pakar perikanan juga diharapkan dapat memberikan informasi mengenai penyakit, cara penanggulangan, pengobatan, dan solusi mengatasinya. Berdasarkan hal tersebut dikembangkan suatu sistem pakar tentang penyakit ikan patin, sehingga dapat memberikan solusi untuk menanggulangi penyakit ikan. penelitian dengan *forward chaining* ini merupakan metode penyesuaian fakta atau pernyataan yang dimulai dari bagian sebelah kiri (*if*), Dengan kata lain, Penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis. Dengan menggunakan metode ini juga mempermudah untuk mendeteksi penyakit dari ikan patin tersebut. dan oleh karena itu peneliti akan mendiagnosa penyakit ikan patin terlebih dahulu baru disimpulkan ikan patin tersebut terkena penyakit apa saja. Dengan fitur yang berbasis *web* yang dimiiki, sistem pakar untuk mendiagnosa penyakit ikan patin ini dapat diakses oleh pembudidaya ikan patin maupun masyarakat umum tanpa harus konsultasi kedokter atau pun ke seorang pakar. Pgroses ini dilanjutkan sampai dengan tujuan mencapai goal dan untuk membantu pembudidaya ikan patin dalam mendiagnosa penyakit tersebut dan saran-saran pengobatannya.

Kata kunci: Sistem Pakar, Diagnosa Penyakit, Ikan Patin, *Forward Chaining*,

Web

ABSTRACT

The need for information on fish disease Patin is currently required, this usually occurs in rural fish farmers who are far away from technological developments, Lack of information knowledge and understanding in the introduction of a disease can lead to errors in disease diagnosis Patin fish. Therefore, an expert in the field of fisheries as a consultation place. Fisheries experts are also expected to provide information on the disease, how to overcome, treatment, and solutions to overcome them. Based on this case developed an expert system about catfish disease, so it can provide solutions to overcome fish diseases. research with forward chaining is a method of adjusting facts or statements beginning at the left (if), In other words, Reasoning begins with facts first to test the truth of the hypothesis. Using this method also makes it easier to detect the disease from the catfish. and therefore researchers will diagnose catfish disease first newly concluded catfish are affected by any disease. With the web-based feature in question, expert systems to diagnose catfish disease can be accessed by catfish farmers and the general public, this process continues until the goal of achieving goals and to assist catfish farmers in diagnosing the disease and its suggestion of treatment.

Keywords: Expert System, Diagnosis of Disease, Patin Fish, Forward Chaining, Web

KATA PENGANTAR

Segala puji dan syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

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

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

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Teknik Informatika Universitas Putera Batam, Bapak Andi Maslan, S.T, M.SI.
3. Bapak Algifanri Maulana, S.SI., M.MSI, selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam.
5. Bapak Drh. Achmad Bahtiar Rifai selaku dokter hewan di Stasiun Karantina Ikan, Pengendalian Mutu dan Keamanan Hasil Perikanan Kelas 1 Batam dan

sebagai narasumber yang telah rela meluangkan banyak waktunya untuk mendukung penelitian ini.

6. Keluarga yang selalu memberikan doa dan motivasi yang baik
7. Rekan-rekan mahasiswa/i Universitas Putera Batam yang turut memberikandoa dan dukungannya.
8. Mitra kerja yang selalu memberikan masukan yang berguna untuk penelitian ini.
9. Serta pihak-pihak lain yang tidak dapat disebutkan satu per satu.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan selalu mencurahkan hidayah serta taufikNya, Amin.

Batam, 8 Februari 2017

Mesran Fernando Simamora

DAFTAR ISI

PERNYATAAN	iii
HALAMAN PENGESAHAN	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Penelitian	1
1.2. Identifikasi Masalah	4
1.3 Pembatasan Masalah	4
1.4 Perumusan Masalah.....	5
1.5 Tujuan Penelitian.....	6
BAB II TINJAUAN PUSTAKA	8
2.1. Teori Dasar	8
2.1.1. Kecerdasan Buatan atau <i>Artificial Intelligence (AI)</i>	8
2.1.1.1. Logika <i>Fuzzy (Fuzzy Logic)</i>	9
2.1.1.2 Jaringan Saraf Tiruan (<i>Artificial Neural Network</i>).....	12
2.1.1.3 Sistem Pakar (<i>Expert System</i>).....	17
2.1.2 <i>Web</i>	31
2.1.3 Basis data (<i>Database</i>).....	31

2.1.4	Validasi Sistem	32
2.2	Variabel Penelitian	33
2.2.1	Bintik putih (<i>white spot</i>)	34
2.2.2.	<i>Saprolegnia</i>	35
2.3	<i>Software</i> Pendukung	36
2.3.1	<i>Xampp</i>	36
2.3.2.	<i>Hypertext Preprocessor (PHP)</i>	36
2.3.3	HTML (Hyper Text Markup Language)	37
2.3.4	CSS(Cascading Style sheet)	38
2.3.5	JavaScript	39
2.3.6	MySQL dan SQL.....	40
2.3.7	<i>StarUML</i>	41
2.4	Penelitian Terdahulu.....	49
2.5	Kerangka Pemikiran	54
BAB III METODE PENELITIAN		54
3.1.	Desain Penelitian	54
3.2.	Teknik Pengumpulan Data	56
3.3.	Operasional Variabel	57
3.4.	Perancangan Sistem.....	58
3.4.1.	Desain Basis Pengetahuan.....	59
3.4.2.	Struktur Kontrol (Mesin inferensi).....	66
3.4.3.	<i>DesainUML (Unified Modeling Language)</i>	69
3.3.4	Desain <i>Database</i>	78
3.4.5.	Prototype.....	79
3.5.	Lokasi dan Jadwal penelitian.....	85
3.5.1.	Lokasi	85
3.5.2.	Jadwal Penelitian	86

BAB IV	88
HASIL PENELITIAN DAN PEMBAHASAN	88
4.1 Hasil Penelitian.....	88
4.2. Pembahasan	96
4.2.1. Pengujian Validasi Sistem.....	96
BAB V SIMPULAN DAN SARAN	102
5.1. Simpulan.....	102
5. 2. Saran.....	103
DAFTAR PUSTAKA.....	104
DAFTAR RIWAYAT HIDUP	
SURAT KETERANGAN PENELITIAN	
LAMPIRAN	

DAFTAR TABEL

Tabel 2. 1 Tabel Keputusan	26
Tabel 2. 2 Alternatif Tabel Keputusan	28
Tabel 2. 3 Simbol <i>Use Case Diagram</i>	44
Tabel 2. 4 Simbol <i>Activity Diagram</i>	45
Tabel 2. 5 Simbol <i>Sequence Diagram</i>	46
Tabel 2. 6 Simbol <i>class diagram</i>	48
Tabel 3. 1 Variabel dan indikator.....	58
Tabel 3. 2 Tabel data indikator	59
Tabel 3. 3 Tabel indikator penyakit ikan Patin	60
Tabel 3. 4 Tabel gejala penyakit ikan Patin	62
Tabel 3. 5 Tabel data gejala <i>White spot</i>	63
Tabel 3. 6 Tabel data gejala <i>Saprolegnia</i>	63
Tabel 3. 7 Tabel indikator dan gejala penyakit ikan Patin.....	64
Tabel 3. 8 Tabel Keputusan	64
Tabel 3. 9 Jadwal Penelitian.....	86
Tabel 4. 1 Pengujian Menu <i>Home</i>	96
Tabel 4. 2 Pengujian Menu <i>Diagnosa</i>	97
Tabel 4. 3 Pengujian Menu <i>Hasil Diagnosa</i>	97
Tabel 4. 4 Pengujian menu <i>log out</i>	98
Tabel 4. 5 Pengujian Menu <i>log In</i>	98
Tabel 4. 6 Pengujian Menu <i>Beranda</i>	98
Tabel 4. 7 Pengujian Menu <i>Mengelola User</i>	99
Tabel 4. 8 Pengujian Menu <i>Relasi</i>	100

Tabel 4. 9 Pengujian Menu Mengelolah User.....	100
Tabel 4. 10 Pengujian Menu Mengelolah Penyakit	101
Tabel 4. 11 Pengujian Menu Mengelolah Gejala.....	102
Tabel 4. 12 Pengujian Menu <i>log Out</i>	102

DAFTAR GAMBAR

Gambar2.2 Komponen-komponen dalam sebuah sisitem pakar	21
Gambar2.3 Contoh Jaringan Semantik	24
Gambar2.4 Pohon Keputusan	27
Gambar2.5 Alternatif Pohon Keputusan	29
Gambar 2.6 Ikan Patin	33
Gambar 2.7 Bintik putih	34
Gambar 2.8 <i>Saprolegnia</i>	35
Gambar 2.9 Kerangka Pemikiran	54
Gambar 3.1 Desain penelitian	54
Gambar 3.2 Pohon keputusan.....	65
Gambar 3.3 Flowchart	68
Gambar 3.4 <i>Use Case Diagram</i>	70
Gambar 3.5 <i>Diagram activity admin login</i>	71
Gambar 3.6 <i>Diagram activity admin lupa password</i>	71
Gambar 3.7 <i>Diagram activity admin ubah password</i>	72
Gambar 3.8 <i>Diagram activity admin daftar penyakit</i>	72
Gambar 3.9 <i>Diagram activity admin daftar gejala</i>	73
Gambar 3.10 <i>Diagram activity admin relasi</i>	73
Gambar 3.11 <i>Diagram activity admin mengelola bobot gejala</i>	74
Gambar 3.12 <i>Diagram activity admin log out</i>	74
Gambar 3.13 <i>Diagram activity user registrasi</i>	75
Gambar 3.14 <i>Diagram activity user log in</i>	75

Gambar 3.15 <i>Diagram activity user ubah password</i>	76
Gambar 3.16 <i>Diagram activity user diagnosa</i>	76
Gambar 3.17 <i>Diagram activity user hasil diagnosa</i>	77
Gambar 3.18 <i>Rancangan Database</i>	78
Gambar 3.19 <i>Tampilan menu daftar</i>	79
Gambar 3.20 <i>Tampilan user login</i>	79
Gambar 3.21 <i>Tampilan Menu Home dan Help</i>	80
Gambar 3.22 <i>Tampilan Menu Pertanyaan</i>	80
Gambar 3.23 <i>Tampilan Menu Hasil</i>	81
Gambar 3.24 <i>Tampilan Menu Profil</i>	81
Gambar 3.25 <i>Tampilan Menu Login</i>	82
Gambar 3.26 <i>Tampilan Menu Home</i>	82
Gambar 3.27 <i>Tampilan Menu info</i>	83
Gambar 3.28 <i>Tampilan Menu input Gejala</i>	83
Gambar 3.29 <i>Tampilan Menu Input Daftar Penyakit</i>	84
Gambar 3.30 <i>Tampilan Admin Pengguna</i>	84
Gambar 3.31 <i>Alamat Karentina dari Google Map</i>	85
Gambar 4.1 <i>Home</i>	89
Gambar 4.2 <i>Diagnosa</i>	90
Gambar 4.3 <i>Diagnosa</i>	91
Gambar 4.4 <i>Log in admin</i>	92
Gambar 4.5 <i>Home Admin</i>	92
Gambar 4.6 <i>Laporan Penyakit</i>	93
Gambar 4.7 <i>Relasi</i>	93
Gambar 4.8 <i>Mengelola user</i>	94
Gambar 4.9 <i>Mengelola Penyakit</i>	94
Gambar 4.10 <i>Mengelola Gejala</i>	95

DAFTAR LAMPIRAN

LAMPIRAN FORM WAWANCARA

LAMPIRAN PHOTO WAWANCARA

LAMPIRAN DATA KASUS PENYAKIT

LAMPIRAN KODING PROGRAM

BAB I

PENDAHULUAN

1.1. Latar Belakang Penelitian

Indonesia adalah negara yang mempunyai perairan laut dan perairan darat yang sangat luas dibandingkan negara asean lainnya sumber daya alam ini salah satunya menghasilkan ikan dan perikanan lainnya. Oleh karena itu, akhir-akhir ini pemerintah sangat mengintensifikan usaha penangkapan ikan dan budidaya ikan dalam upaya mendapatkan pemasukan devisa yang lebih besar (juni 2003) Budidaya ikan patin lokal indonesia sudah mulai dirintis sejak tahun 1985, setelah Balai Penelitian Perikanan Air tawar berhasil mengembangkan belum tersebar luaskan kepada masyarakat.

Ikan patin (*pangesius sp.*) merupakan nilai yang memiliki ekonomis. Hal ini diperkuat dengan pernyataan yang dikemukakan oleh departemen pertanian FAO (*Food And Agriculture organization*) yang menyatakan bahwa ikan patin menempati urutan keempat setelah udang, salmon, dan nila dalam hal produksi perikanan akuakultur.

Dalam proses budidaya ikan patin, pembudidaya ikan patin mengalami beberapa kendala, salah satu kendala yang dimaksud yaitu terjangkitnya penyakit pada ikan patin yang dibudidayakan. Untuk mengatasi kendala tersebut maka pembudidaya membutuhkan suatu pengetahuan tentang informasi penyakit,

gejala, dan penanganan untuk penyakit tersebut. Oleh sebab itu dibutuhkan peran seorang pakar dibidang perikanan sebagai tempat konsultasi. Pakar perikanan juga diharapkan dapat memberikan informasi mengenai penyakit, cara penanggulangan, pengobatan, dan solusi mengatasinya. Berdasarkan hal tersebut dikembangkan suatu sistem pakar tentang penyakit ikan patin, sehingga dapat memberikan solusi untuk menanggulangi penyakit ikan. Oleh sebab itu penelitian dilakukan di “STASIUN KARANTINA IKAN PENGENDALIAN MUTU DAN KEAMANAN HASIL PERIKANAN KELAS 1 BATAM”

Stasiun Karantina ikan pengendalian mutu dan keamanan hasil perikanan kelas 1 Batam adalah Balai perikanan tempat kegiatan pemeriksaan dan identifikasi penyakit ikan pada kegiatan pemantauan untuk keluar masuk nya ikan kota Batam yang dilaksanakan di laboratorium. Kurangnya pengetahuan karyawan pembudidayaan dan pembesaran ikan patin, di stasiun karentina pengendalian mutu dan keamanan hasil perikanan kelas 1 Batam untuk mengetahui tentang jenis-jenis penyakit dan gejala penyakit ikan patin yang akan menghambat atau memperlambat pekerjaan mereka terjadi sesuatu penyakit atau gejala penyakit. Dan kurangnya informasi pengetahuan dan pemahaman dalam pengenalan suatu penyakit dapat mengakibatkan kesalahan diagnosa penyakit pada ikan patin. Serta sulitnya masyarakat umum yang membudidaya ikan patin untuk mendiagnosa penyakit ikan patin yang menyebabkan keterlambatan atas penanggulangan penyakit dan sehingga terjadi kematian ikan patin.

Dengan ini peneliti menggunakan metode *forward chaining*. Metode ini terbukti merupakan cara paling tepat dalam mendeteksi suatu penyakit ikan,

sehingga *user* tidak perlu menebak penyakit ikan patin mereka karena sistem akan memberikan jawaban berdasarkan fakta yang ada. *forward chaining* ini merupakan metode penyesuaian fakta atau pernyataan yang dimulai dari bagian sebelah kiri(*if*), Dengan kata lain, Penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis. Dengan menggunakan metode ini juga mempermudah untuk mendeteksi penyakit dari ikan patin tersebut. dan oleh karena itu peneliti akan mendiagnosa penyakit ikan patin terlebih dahulu baru disimpulkan ikan patin tersebut terkena penyakit apa saja.

Hal ini yang menjadi dasar pertimbangan peneliti dalam membuat sistem pakar untuk mendeteksi penyakit ikan patin ini. Dimana pengertian dari sistem pakar adalah merupakan bagian dari ilmu pengetahuan komputer dimana sistem pakar yang dirancang ini dapat menggantikan seorang pakar atau dokter yang bisa mendiagnosa penyakit ikan patin tersebut dengan hanya menggunakan sistem pakar tersebut di komputer. Dan setiap orang pun bisa menggunakannya tanpa harus konsultasi kedokter atau pun ke seorang pakar.

Berdasarkan uraian di atas, maka dikembangkan sebuah sistem pakar untuk menangani indentifikasi penyakit pada ikan patin bedasarkan gejalahnya yang diharapkan bisa memberi suatu informasi tentang penyakit yang di derita oleh ikan patin dan cara penanggulangannya, dengan demikian maka dibuatlah suatu sistem tersebut, diangkat sebuah judul yaitu **“SISTEM PAKAR MENDIAGNOSA PENYAKIT IKAN PATIN DENGAN METODE FORWARD CHAINING BERBASIS WEB”**

1.2. Identifikasi Masalah

Berdasarkan penjelasan dari latar belakang penelitian di atas, identifikasi masalah dirumuskan sebagai berikut:

1. Kurangnya informasi pengetahuan dan pemahaman dalam pengenalan suatu penyakit dapat mengakibatkan kesalahan diagnosa penyakit pada ikan Patin.
2. Kurangnya pengetahuan karyawan pembudidayaan dan pembesaran ikan Patin, di stasiun Karantina pengendalian mutu dan keamanan hasil perikanan kelas 1 batam untuk mengetahui tentang jenis-jenis penyakit dan gejala penyakit ikan patin yang akan menghambat atau memperlambat pekerjaan mereka jika terjadi sesuatu penyakit atau gejala penyakit.
3. Sulitnya masyarakat umum yang membudidaya ikan patin untuk mendiagnosa penyakit ikan patin yang menyebabkan keterlambatan atas penanggulangan penyakit dan sehingga terjadi kematian ikan patin.

1.3 Pembatasan Masalah

Agar penelitian ini terarah dan tujuan dari penulisan ini dapat tercapai sesuai dengan yang diharapkan, maka dalam penulisan ini dibatasi terhadap masalah yang dibahas. Adapun batasan masalah tersebut adalah:

1. Pada penelitian ini hanya membahas penyakit Bintik putih (*white spot*) dan *Saprolegnia*.

2. Sistem pakar ini dirancang menggunakan pemograman *PHP* dan *database MySQL*.
3. Penelusuran penyakit berdasarkan eksternal dan juga menurut pengetahuan dokter hewan di stasiun Karantina pengendalian mutu dan keamanan hasil perikanan kelas 1 batam.

1.4 Perumusan Masalah

Adapun rumusan masalah yang akan dikaji pada penelitian ini sebagai Berikut :

1. Bagaimana cara mengidentifikasi penyakit ikan patin ini dengan menggunakan metode *forward chaining*?
2. Bagaimana merancang suatu sistem pakar yang dapat digunakan untuk mendiagnosa suatu jenis penyakit berdasarkan gejala yang ditemukan pengguna sehingga pengguna menemukan solusi atas permasalahan yang dihadapi?
3. Bagaimana implementasi sistem pakar ini berbasis *Web*?

1.5 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini adalah:

1. Berdasarkan gejala-gejala yang muncul maka kita dapat mengetahui penyakit yang ada pada ikan patin tersebut.
2. Untuk merancang suatu sistem pakar yang dapat digunakan untuk mendiagnosa Suatu jenis penyakit berdasarkan gejala yang ada pada ikan patin, sehingga pengguna dapat menemukan solusi yang dihadapinya.
3. Melalui tahapan yang dilalui seperti penerapan metode, merancang sistem dan menemukan solusi terhadap terhadap masalah yang dihadapi dengan hanya mengakses di *web browser*.

1.6 Manfaat Penelitian

1.6.1 Aspek Teoritis

1. Untuk mengembangkan ilmu dibidang *programming web*.
2. Untuk menambah kajian ilmiah dalam media pembelajaran.

1.6.2 Aspek Praktis

1. Bagi peneliti
 - a. Memberikan informasi mengenai penyakit ikan patin sebagai acuan dalam implementasi atau perancangan suatu aplikasi atau system.
2. Bagi pembudaya ikan Patin

- a. Memberikan informasi maupun solusi pada pengguna untuk mendiagnosa penyakit ikan patin dengan lebih cepat, akurat, praktis, dan efisien dengan menggunakan sistem pakar ini.
- b. Mempermudah pengguna untuk mengetahui jenis gejala dan penyakit yang di derita ikan patin.

3 Bagi masyarakat umum

- a. Memberikan solusi awal untuk mengetahui penyakit penyakit ikan patin tanpa harus menjumpai seorang pakar maupun dokter hewan (ikan).

BAB II

TINJAUAN PUSTAKA

2.1. Teori Dasar

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

Pada penelitian ini akan dijelaskan tentang beberapa teori dasar antara lain kecerdasan buatan atau *Artificial Intelligence (AI)* dan beberapa subdisiplin ilmunya seperti logika *fuzzy (fuzzy logic)*, jaringan saraf tiruan (*artificial neural network*), dan sistem pakar (*expert system*); *web*, basis data, dan validitas sistem.

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

Menurut (Sutojo, Mulyanto, & Suhartono, 2011) kecerdasan buatan berasal dari bahasa Inggris “*Artificial Intelligence*” atau disingkat AI, yaitu *Intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud disini merujuk pada mesin yang mampu berfikir, menimbang tindakan yang akan diambil, dan mampu mengambil

keputusan seperti yang dilakukan oleh manusia. Berikut adalah beberapa definisi kecerdasan buatan yang telah didefinisikan oleh para ahli.

Rich and Knight (1991):

“Kecerdasan buatan (AI) merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia” ([http://catalog.ebay.com/Artificial-Intelligence-Elaine-Rich-Kevin-Knight-\(1991-Hardcover-Subsequent-Edition-/34475:\)](http://catalog.ebay.com/Artificial-Intelligence-Elaine-Rich-Kevin-Knight-(1991-Hardcover-Subsequent-Edition-/34475:))).

Menurut Wiston dan Prendergast (1984), tujuan kecerdasan buatan adalah:

1. Membuat mesin lebih pintar (tujuan utama)
2. Memahami apa itu kecerdasan buatan (tujuan ilmiah)
3. Membuat mesin lebih bermanfaat (tujuan *enterpreneurial*)

Bedasarkan definisi ini, maka kecerdasan buatan menawarkan media maupun uji teori tentang kecerdasan. Teori-teori ini nantinya dapat dinyatakan dalam bahasa pemrograman dan eksekusinya dapat dibuktikan pada komputer nyata.

2.1.1.1. Logika Fuzzy (*Fuzzy Logic*)

Logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah, yang sesuai untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan komputer, *multi-channel* atau *workstation* berbasis akuisisi data, dan sistem kontrol. Dalam logika *fuzzy* memungkinkan nilai keanggotaan berada di antara 0 dan 1.

Artinya suatu keadaan memungkinkan mempunyai dua nilai “Ya” dan “Tidak” secara bersamaan, namun besar nilainya tergantung pada bobot keanggotaan yang dimilikinya. Logika *fuzzy* dapat digunakan di berbagai bidang seperti pada sistem diagnosis penyakit (dalam bidang kedokteran); pemodelan sistem pemasaran, riset operasi (dalam bidang ekonomi); kendali kualitas air, prediksi adanya gempa bumi, klasifikasi dan pencocokan pola (dalam bidang teknik)(Sutojo et al., 2011).

Ada beberapa keuntungan yang dapat diambil ketika menggunakan logika *fuzzy* untuk memecahkan suatu masalah yaitu :

1. Konsep *fuzzy logic* mudah dimengerti.
2. *Fuzzy logic* sangat fleksibel.
3. *Fuzzy logic* memiliki toleransi terhadap data-data yang tidak tepat.
4. *Fuzzy logic* mampu memodelkan fungsi-fungsi non linear yang sangat kompleks.
5. *Fuzzy logic* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar serta langsung tanpa harus melalui pelatihan.
6. *Fuzzy logic* dapat bekerja sama dengan teknik-teknik kendali serta konvensional.
7. *Fuzzy logic* didasarkan pada bahasa alami.

Sistem inferensi *fuzzy* adalah cara memetakan ruang *input* menuju ruang *output* menggunakan logika *fuzzy*. Untuk memahami cara kerja *fuzzy*, ada 4 struktur elemen dasar sistem inferensi *fuzzy*.(Sutojo et al., 2011)

Basis pengetahuan *fuzzy*, yaitu kumpulan aturan (*rule*) *fuzzy* dalam bentuk pernyataan *IF...THEN*.

1. Fuzzifikasi, yaitu proses untuk mengubah *input* sistem yang mempunyai nilai tegas menjadi variabel linguistik menggunakan fungsi keanggotaan yang disimpan dalam basis pengetahuan *fuzzy*.
2. Mesin inferensi, yaitu proses untuk mengubah *input fuzzy* menjadi *output fuzzy* dengan cara mengikuti aturan-aturan yang telah ditetapkan pada basis pengetahuan *fuzzy*.
Defuzzifikasi, yaitu mengubah *output fuzzy* yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan fuzzifikasi.

Beberapa metode yang digunakan dalam sistem inferensi *fuzzy* adalah (Sutojo.T, Mulyanto.edy, dkk 2011: 233-237):

1. Metode Tsukamoto

Dalam inferensinya, metode Tsukamoto menggunakan tahapan sebagai berikut:

- a. *Fuzzyfikasi*
- b. Pembentukan basis pengetahuan *fuzzy* (*rule* dalam bentuk *IF...THEN*)
- c. Mesin inferensi menggunakan fungsi implikasi *MIN* (*Minimum*)
- d. *Defuzzifikasi* menggunakan metode Rata-rata (*Average*)

2 Metode Mamdani

Metode ini sering digunakan dalam karena strukturnya yang sederhana. Pada metode ini, untuk mendapatkan *output* diperlukan 4 tahapan sebagai berikut:

- a. *Fuzzyfikasi*

- b. Pembentukan basis pengetahuan *fuzzy* (*rule* dalam bentuk *IF...THEN*)
- c. Aplikasi fungsi implikasi menggunakan fungsi *MIN* (*Minimum*) dan komposisi antar-*rule* menggunakan fungsi *MAX*(*Maximum*) dengan menghasilkan himpunan *fuzzy* baru
- d. *Defuzzyfikasi* menggunakan metode *Centroid* (Titik Tengah)

3. Metode Sugeno

Metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985. Dalam metode ini, *output* sistem berupa konstanta atau persamaan linier. Dalam inferensinya, metode Sugeno menggunakan tahapan sebagai berikut:

- a. Fuzzyfikasi
- b. Pembentukan basis pengetahuan *fuzzy* (*rule* dalam bentuk *IF...THEN*)
- c. Mesin inferensi menggunakan fungsi implikasi *MIN* (*Minimum*)
- d. Defuzzyfikasi menggunakan metode Rata-rata (*Average*)

2.1.1.2 Jaringan Saraf Tiruan (*Artificial Neural Network*)

Jaringan saraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak manusia. Elemen kunci dari paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah elemen pemrosesan yang saling berhubungan (*neuron*), cara kerja jaringan saraf tiruan seperti cara kerja manusia, yaitu belajar melalui contoh. Seperti pengenalan pola atau klasifikasi data, melalui proses pembelajaran. (Sutojo et al., 2011) Beberapa kelebihan yang dimiliki jaringan saraf

tiruan antara lain:

1. Belajar adaptif, yaitu kemampuan untuk mempelajari bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan atau pengalaman awal.
2. *Self-Organisation*, Sebuah jaringan saraf tiruan dapat membuat organisasi sendiri atau representasi dari informasi yang diterimanya selama sewaktu belajar
3. *Real Time Operation*, yaitu perhitungan jaringan saraf tiruan yang dapat dilakukan secara paralel sehingga perangkat keras yang dirancang dan diproduksi secara khusus dapat mengambil keuntungan dari kemampuan ini.
4. Selain mempunyai beberapa kelebihan, jaringan saraf tiruan juga mempunyai kelemahan-kelemahan antara lain :
5. Tidak efektif jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.
6. Tidak efisien jika digunakan untuk melakukan operasi algoritma aritmatika, operasi logika, dan simbolis.
7. Untuk beroperasi JST butuh pelatihan sehingga bila jumlah datanya besar, waktu yang digunakan untuk proses pelatihan sangat lama.

Adapun beberapa aplikasi dalam jaringan saraf tiruan antara lain :(Sutojo et al., 2011)

1. Jaringan saraf tiruan dalam kehidupan sehari-hari

Jaringan saraf tiruan telah banyak digunakan untuk mengidentifikasi pola atau tren dalam data sehingga jaringan saraf tiruan baik digunakan untuk peramalan kebutuhan , termasuk peramalan penjualan, kontrol proses industri,

penelitian pelanggan, validasi data, manajemen resiko, dan target pemasaran.

2. Jaringan saraf tiruan dalam dunia kedokteran

Jaringan saraf tiruan ini diyakini akan menerima aplikasi yang sangat luas pada sistem biomedis dalam beberapa tahun mendatang. Pada saat ini, sebagian besar penelitian diarahkan pada pemodelan bagian tubuh manusia dan pengenalan penyakit dari berbagai scan (misalnya cardiograms, CAT scann, scan *ultrasonik*, dan lain-lain).

3. Saraf tiruan dalam bisnis.

Hampir semua aplikasi jaringan saraf tiruan masuk dalam suatu bidang bisnis atau analisis keuangan.ada beberapa potensi menggunakan jaringan saraf tiruan untuk tujuan bisnis, termasuk alokasi sumber daya dan penjadwalan Berdasarkan cara memodifikasi bobotnya, pelatihan jaringan saraf tiruan dibagi menjadi dua, yaitu:(Sutojo et al., 2011)

1. Algoritma pembelajaran dengan Supervisi (pembimbing)

Dalam pelatihan ini, jaringan dipandu oleh sejumlah pasangan data (masukan dan target) yang berfungsi sebagai pembimbing untuk melatih jaringan hingga diperoleh bobot yang terbaik. Algoritma yang termasuk dalam pelatihan dengan supervisi antara lain:

a. *Hebb-Rule*

Model ini diperkenalkan oleh D.O. Hebb yang menggunakan cara menghitung bobot dan bias secara iteratif dengan memanfaatkan model pembelajaran dengan supervisi sehingga bobot dan bias dapat dihitung secara otomatis tanpa harus melakukan cara coba-coba. Arsitektur jaringan ini terdiri dari

beberapa unit *input* dihubungkan langsung dengan sebuah unit *output*, ditambah dengan sebuah bias.

b. *Perceptron*

Model ini ditemukan oleh Rosenblatt (1962) dan Minsky – Papert (1969). Model jaringan ini merupakan model yang terbaik pada saat itu. Algoritma pelatihan *perceptron* digunakan baik untuk *input* biner maupun bipolar, dengan tertentu.

c. *Delta-Rule*

Selama pelatihan pola, *Delta-Rule* akan mengubah bobot dengan cara meminimalkan *error* antara *output* jaringan dengan target.

d. *Backpropagation*

Backpropagation adalah metode penurunan gradien untuk meminimalkan kuadrat *error* keluaran. Pelatihan jaringan ini terdiri dari 3 tahap, yaitu tahap perambatan maju (*forward propagation*), tahap perambatan balik, dan tahap perubahan bobot dan bias. Arsitektur jaringan ini terdiri dari *input layer*, *hidden layer*, dan *output layer*.

e. *Heteroassociative Memory*

Jaringan saraf *heteroassociative memory* adalah jaringan yang dapat menyimpan kumpulan pengelompokan pola dengan cara menentukan bobot-bobotnya sedemikian rupa. Algoritma pelatihan yang biasa digunakan adalah *Hebb-Rule*.

f. *Bidirectional Associative Memory (BAM)*

Bidirectional Associative Memory (BAM) adalah model jaringan saraf yang

memiliki 2 lapisan, yaitu lapisan *input* dan lapisan *output* yang mempunyai hubungan timbal balik antara keduanya (bersifat *bidirectional*). Arsitektur jaringan ini terdiri dari 3 *neuron* pada lapisan *input* dan 2 *neuron* pada lapisan *output*. Model jaringan ini terbagi menjadi 2 jenis yaitu *BAM* Diskrit dan *BAM* Kontinu.

g. *Learning Vector Quantization (LVQ)*

Learning Vector Quantization (LVQ) adalah suatu model pelatihan pada lapisan kompetitif terawasi yang akan belajar secara otomatis untuk mengklasifikasikan vektor-vektor *input* ke dalam kelas-kelas tertentu.

2. Pembelajaran tanpa Supervisi (jaringan kohonen).

Dalam pelatihan ini, tidak ada pembimbing yang digunakan untuk memandu proses pelatihan. Jaringan hanya diberi *input* tetapi tidak mendapatkan target yang diinginkan sehingga modifikasi bobot pada jaringan dilakukan menurut parameter tertentu. Model jaringan yang termasuk dalam pelatihan tanpa supervisi adalah jaringan kohonen yang diperkenalkan oleh Prof. Teuvo Kohonen pada tahun 1982.

Pada jaringan kohonen, *neuron-neuron* pada suatu lapisan data akan menyusun dirinya sendiri berdasarkan *input* nilai tertentu dalam suatu *cluster*. *Cluster* yang dipilih sebagai pemenang adalah *cluster* yang mempunyai vektor bobot paling cocok dengan pola *input*, yaitu *cluster* yang memiliki jarak yang paling dekat.

2.1.1.3 Sistem Pakar (*Expert System*)

Menurut (Sutojo et al., 2011) Sistem pakar mulai dikembangkan pada pertengahan 1960, ditandai dengan lahirnya sistem pakar pertama bernama *General-purpose Problem Solver (GPS)* yang dikembangkan oleh Newell dan Simon. Kemudian bermunculan sistem pakar lain di berbagai bidang seperti *MYCIN* untuk diagnosis penyakit, *DENDRAL* untuk mengidentifikasi struktur molekul campuran yang tak dikenal, *XCON & XSEL* untuk membantu konfigurasi sistem komputer besar, *SOPHIE* untuk analisis sirkuit elektronik, *Prospector* digunakan di bidang geologi untuk membantu mencari dan menemukan deposit, *FOLIO* digunakan untuk membantu memberikan keputusan bagi seorang manajer dalam masalah stok dan investasi, *DELTA* dipakai untuk pemeliharaan lokomotif listrik diesel, dan sebagainya.

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

Sistem pakar memiliki ciri-ciri sebagai berikut:

1. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.

2. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami.
3. Bekerja berdasarkan kaidah/rile tertentu.
4. Mudah dimodifikasi.
5. Basis pengetahuan dan mekanisme inferensi terpisah.
6. Keluarannya bersifat anjuran.
7. Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

Menurut (Sutojo et al., 2011) Konsep dasar dari sistem pakar, di antaranya:

1. Kepakaran (*Expertise*)

Kepakaran (*Expertise*), merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Memungkinkan para ahli dapat mengambil keputusan lebih cepat, dan lebih daripada seorang yang bukan pakar. Kepakaran itu sendiri meliputi pengetahuan tentang :

- a. Fakta-fakta tentang bidang permasalahan tertentu.
- b. Teori-teori tentang bidang permasalahan tertentu.
- c. Aturan-aturan prosedur-prosedur menurut bidang permasalahan umumnya.
- d. Aturan *heuristic* yang harus dikerjakan dalam situasi tertentu.
- e. Strategi global untuk memecahkan permasalahan.
- f. Pengetahuan tentang pengetahuan (*meta knowledge*).

2. Pakar (*Expert*)

Pakar (*Expert*) adalah seorang yang mempunyai pengetahuan, pengalaman, dan metode khusus, serta mampu menerapkan untuk memecahkan masalah atau

memberi nasihat. Jadi seorang pakar harus mampu melakukan kegiatan-kegiatan berikut:

- a. Mengenali dan memformulasikan permasalahan.
 - b. Memecahkan permasalahan secara cepat dan tepat.
 - c. Menerangkan pemecahannya.
 - d. Belajar dari pengalaman.
 - e. Merestrukturisasi pengetahuan.
 - f. Memecahkan aturan-aturan.
 - g. Menentukan relevansi.
3. Pemindehan Kepakaran (*Transferring Expertise*)

Pemindehan Kepakaran (*Transferring Expertise*) adalah memindehan keparakan dari seseorang ke dalam komputer, kemudian ditransfer kepada orang lain yang buakn pakar.

- a. Akuisisi pengetahuan (dari pakar atau sumber lain)
- b. Epresentasi pengetahuan (pada komputer)
- c. Inferensi pengetahuan
- d. Pemindehan pengetahuan ke pengguna.

4. Inferensi (*Inferencing*)

Inferensi (*Inferencing*) adalah sebuah prosedur (program) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampikan pada suatu komponen disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Tujuan mesin inferensi adalah mengambil kesimpulan berdasarkan basis pengetahuan yang dimiliki.

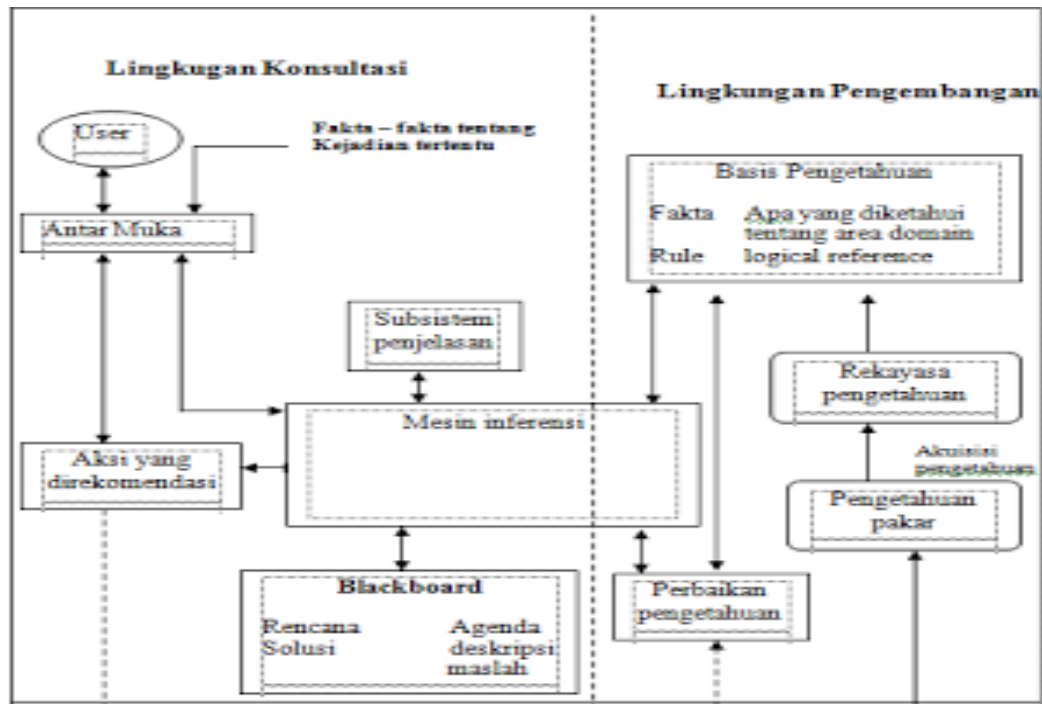
5. Aturan-aturan (*rule*)

Aturan-aturan (*rule*) kebanyakan software sistem pakar komersial adalah sistem yang berbasis *rule* (*rule-based system*), yaitu pengetahuan disimpan terutama dalam bentuk *rule*, sebagai prosedur-prosedur pemecahan masalah.

6. Kemampuan menjelaskan (*Explanation capability*)

Kemampuan menjelaskan (*Explanation capability*) adalah kemampuan untuk menjelaskan saran atau rekomendasi yang diberikan. Sistem ini memungkinkan untuk memeriksa penalaran yang dibuatnya sendiri dan menjelaskan operasi-operasinya.

Menurut (Sutojo et al., 2011) sistem pakar terdapat dua bagian penting, yaitu lingkungan pengembang (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembang digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan ke dalam *knowledge base* (basis pengetahuan). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar.



Gambar 2.1 Komponen-komponen dalam sebuah sistem pakar

Sumber: (Sutojo et al., 2011)

1. Akuisisi Pengetahuan

Menurut (Sutojo et al., 2011) Subsistem ini digunakan untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan meletakkannya ke dalam basis pengetahuan dengan format tertentu (dalam bentuk representasi pengetahuan). Sumber-sumber pengetahuan bisa diperoleh dari pakar, buku, dokumen, multimedia, basis data, laporan riset khusus, dan informasi yang terdapat di web.

2. Basis Pengetahuan (*Knowledge base*)

Menurut (Sutojo et al., 2011) Basis pengetahuan berisi pengetahuan yang diperlukan untuk memahami, memformulasikan dan menyelesaikan masalah. Basis

pengetahuan terdiri dari dua elemen dasar yaitu fakta dan *rule* atau aturan. Basis pengetahuan terdiri dari dua elemen dasar, yaitu:

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

3. Mesin Inferensi (*Inference Engine*)

Menurut (Sutojo et al., 2011) Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan.

4. Daerah Kerja (*Blackboard*)

Menurut (Sutojo et al., 2011) untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang terjadi, Sistem pakar membutuhkan *Blackboard*, yaitu area pada memori yang berfungsi sebagai basis data. Tiga keputusan yang dapat direkam pada *blackboard*, yaitu:

- a. rencana : bagaimana menghadapi masalah.
- b. agenda, aksi-aksi potensial yang sedang menunggu untuk eksekusi.
- c. solusi : calon aksi yang akan dibangkitkan.

5. Antarmuka Pengguna (*User Interface*)

Menurut (Sutojo et al., 2011) digunakan sebagai media komunikasi antara pengguna dan sistem pakar. Komunikasi ini paling bagus bila disajikan dalam

bahasa alami (*natural language*) dan dilengkapi dengan grafik, menu, dan formulir elektronik.

6. Subsistem penjelasan (*Explanation Subsystem*)

Berfungsi memberi penjelasan kepada user, bagaimana suatu kesimpulan dapat diambil.

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

Kemampuan memperbaiki pengetahuan dari seorang pakar diperlukan untuk menganalisis pengetahuan, belajar dari kesalahan masa lalu, kemudian memperbaiki pengetahuannya sehingga dapat dipakai pada masa mendatang.

8. Pengguna (*User*)

Menurut (Sutojo et al., 2011) Pengguna sistem pakar bukanlah seorang pakar (*non-expert*) yang membutuhkan solusi, saran, atau pelatihan (*training*) dari berbagai permasalahan yang ada.

Menurut(Kusrini, 2008) Representasi pengetahuan merupakan metode yang digunakan untuk mengkodekan pengetahuan dalam sebuah sistem pakar. Representasi dimaksudkan untuk menangkap sifat-sifat penting masalah dan membuat informasi itu dapat diakses oleh prosedur pemecahan masalah.

Adapun karakteristik representasi pengetahuan adalah:

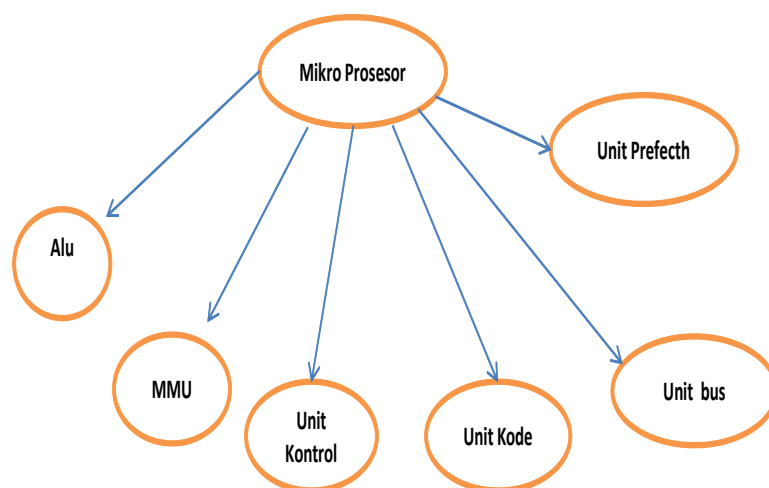
1. Harus bisa diprogramkan dengan bahasa pemrograman atau dengan *shells* dan hasilnya disimpan dalam memori.
2. Dirancang sedemikian sehingga isinya dapat digunakan untuk proses penalaran.

3. Model representasi pengetahuan merupakan sebuah struktur data yang dapat dimanipulasi oleh mesin inferensi dan pencarian untuk aktivasi pencocokkan pola.

Menurut (Hartati & Iswanti, 2008b) Beberapa model representasi pengetahuan yang penting:

1. Jaringan Semantik

Jaringan semantik adalah teknik representasi pengetahuan yang digunakan untuk informasi proposional, sedangkan yang dimaksud informasi proposional adalah pernyataan yang mempunyai nilai benar dan salah. Contoh: sebuah bujur sangkar mempunyai empat sisi. Informasi proposional merupakan bahasa deklaratif karena menyatakan fakta. Komponen untuk mempresentasikan pengetahuan dalam bentuk jaringan semantik adalah simpul (*node*) dan penghubung (*link*).



Gambar 2.2 Contoh Jaringan Semantik

Sumber: (Hartati, 208:23)

2. Bingkai (Frame)

Bingkai merupakan kumpulan *slot-slot* yang berisi atribut untuk mendeskripsikan pengetahuan. Pengetahuan yang termuat dalam *slot* dapat berupa kejadian, lokasi, situasi ataupun *elemen-elemen* lainnya. Bingkai digunakan untuk representasi pengetahuan deklaratif. Bingkai memuat deskripsi sebuah objek dengan menggunakan tabulasi informasi yang berhubungan dengan objek. Representasikan pengetahuan menggunakan bingkai sesuai untuk jenis pengetahuan yang memiliki subjek sempit, lebih bersifat pasti dan jarang berubah-ubah isinya kecuali terdapat kondisi khusus.

3. Kaidah Produksi

Kaidah menyediakan cara formal untuk merepresentasikan rekomendasi, arahan, atau strategi. Kaidah reproduksi dituliskan dalam bentuk jika-maka (*if-then*). Kaidah *if-then* menghubungkan atesenden (*antecedent*) dengan konsekuensi yang diakibatkannya. Berbagai struktur kaidah *if-then* yang menghubungkan obyek atau atribut sebagai berikut :

IF premis *THEN* konklusi

IF masukan *THEN* keluaran

IF kondisi *THEN* tindakan

IF atesenden *THEN* konsekuen

IF data *THEN* hasil

IF tindakan *THEN* tujuan

IF aksi *THEN* reaksi

IF sebab *THEN* akibat

IF gejala *THEN* diagnosa

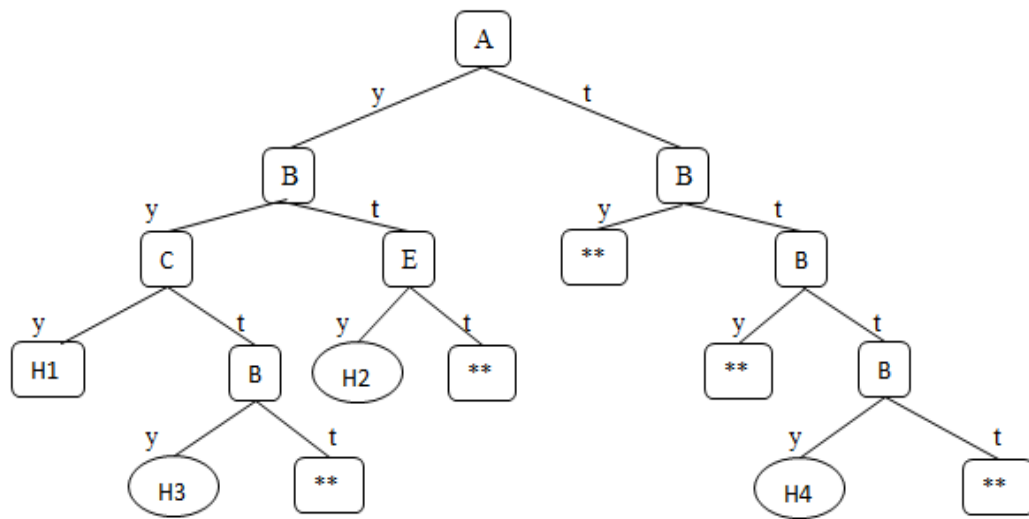
Premis mengacu pada fakta yang harus benar sebelum konklusi tertentu dapat diperoleh. *Masukan* mengacu pada data yang harus tersedia sebelum keluaran dapat diperoleh. *Kondisi* mengacu pada keadaan yang harus berlaku sebelum tindakan dapat diambil. *Antesenden* mengacu situasi yang terjadi sebelum konsekuensi dapat diamati. *Data* mengacu pada informasi yang harus tersedia sehingga sebuah hasil dapat diperoleh. *Tindakan* mengacu pada kegiatan yang harus dilakukan sebelum hasil dapat diharapkan. *Aksi* mengacu pada kegiatan yang menyebabkan munculnya efek dari tindakan tersebut. *Gejala* mengacu pada keadaan yang menyebabkan adanya kerusakan atau keadaan tertentu yang mendorong adanya pemeriksaan.

Menurut (Hartati & Iswanti, 2008a) Tabel keputusan merupakan suatu cara untuk mendokumentasikan pengetahuan. Tabel keputusan merupakan matriks kondisi yang dipertimbangkan dalam pendeskripsian kaidah. Berikut penyajian dalam bentuk tabel keputusan dan pohon keputusan:

Tabel 2.1 Tabel Keputusan

Hipotesa Evidence	Hipotesa 1	Hipotesa 2	Hipotesa 3	Hipotesa 4
<i>Evidence A</i>	Ya	Ya	Ya	Tidak
<i>Evidence B</i>	Ya	Tidak	Ya	Ya
<i>Evidence C</i>	Ya	Tidak	Tidak	Ya
<i>Evidence D</i>	Tidak	Tidak	Tidak	Ya
<i>Evidence E</i>	Tidak	Ya	Ya	Tidak

Sumber:(Hartati & Iswanti, 2008a)



Gambar 2.3 Pohon Keputusan

Sumber: (Hartati & Iswanti, 2008a)

Keterangan:

A = *evidence* A, H1 = hipotesa 1, y = ya
 B = *evidence* B, H2 = hipotesa 2, t = tidak
 C = *evidence* C, H3 = hipotesa 3, ** = tidak menghasilkan hipotesa tertentu
 D = *evidence* D, H4 = hipotesa 4

Menurut (Hartati & Iswanti, 2008a) Dengan melihat pohon keputusan yang dihasilkan, dapat diketahui hipotesa H1 terpenuhi jika memenuhi *evidence* A, B, dan C. Hipotesa H2 terpenuhi jika memiliki *evidence* A dan *evidence* E. Hipotesa H3 akan terpenuhi jika memiliki *evidence* A, B, dan E. Hipotesa H4 akan dihasilkan jika memenuhi *evidence* B, C, dan D. Notasi “y” mengandung arti memenuhi *node* (*evidence*) di atasnya, notasi “t” artinya tidak memenuhi. Dapat dicontohkan untuk menghasilkan hipotesis H2 kalau diurutkan dari node A adalah memiliki *evidence* A, tidak memiliki *evidence* B, dan memiliki *evidence* E. Dalam hal ini jika tidak memenuhi *evidence* tertentu, maka diabaikan, sehingga hasil

yang didapatkan adalah hiptesis H2 terpenuhi jika memiliki evidence A dan E saja.

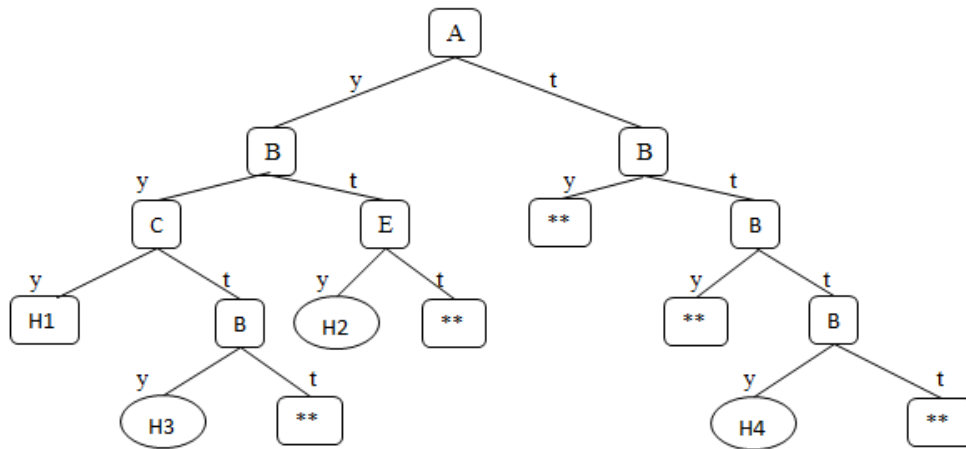
Dalam implementasi sistem pakar terutama dalam sesi konsultasi, *node-node* yang mewakili *evidence* biasanya akan menjadi pertanyaan yang diajukan oleh sistem. Dengan melihat pohon keputusan pada gambar 2.4 permasalahan dapat saja terjadi pada awal sesi konsultasi yaitu pada saat sistem pakar menanyakan “apakah memiliki *evidence* A?”. Permasalahannya adalah apapun jawaban pengguna baik “ya” atau “tidak” maka sistem akan menanyakan *evidence* B. dengan kata lain apapun jawaban pengguna maka tidak akan mempengaruhi sistem. Hal ini dapat diatasi, salah satu caranya adalah dengan mengubah urutan pada tabel keputusan. Alternatif keputusan yang lain seperti terlihat pada tabel 2.5.

Tabel 2.2 Alternatif Tabel Keputusan

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

Sumber: (Hartati & Iswanti, 2008a)

Berdasarkan tabel 2.2 dapat dihasilkan pohon keputusan sebagai berikut:



Gambar 2.4 Alternatif Pohon Keputusan

Sumber: (Hartati & Iswanti, 2008a)

Keterangan:

A = <i>evidence</i> A,	H1 = hipotesa 1,	y = ya
B = <i>evidence</i> B,	H2 = hipotesa 2,	t = tidak
C = <i>evidence</i> C,	H3 = hipotesa 3,	** = tidak menghasilkan hipotesa tertentu
D = <i>evidence</i> D,	H4 = hipotesa 4	

Telihat dari pohon keputusan gambar , masing-masing node yang mewakili *evidence* tertentu kondisi “y” dan “t” sudah tidak mengarah pada *evidence* yang sama. Dalam sesi konsultasi hal yang mengandung arti jawaban pengguna yang berbeda, akan mengarah pada pertanyaan yang berbeda pula.

Proses akuisisi pengetahuan dalam sistem pakar bukanlah hal yang mudah dan sederhana. Pengetahuan yang berhasil didapatkan dan disajikan dalam salah satu bentuk representasi pengetahuan misalnya disajikan dalam bentuk tabel keputusan dan dapat diubah susunannya seperti contoh kasus diatas. Secara teori sudah benar tetapi pada pengetahuan domain pengetahuan tertentu, perubahan

yang seperti itu terkadang tidak diperkenalkan. Dengan kata lain, pengetahuan yang akan disajikan dalam bentuk representasi tertentu harus mengikuti kaidah-kaidah kepakaran dalam domainnya dan tidak boleh bertentangan.

4. Logika Predikat

Logika predikat berdasarkan pada kebenaran dan kaidah inferensi untuk merepresentasikan symbol-simbol dan hubungan satu dengan yang lainnya. Logika predikat selain digunakan untuk menentukan kebenaran (truthfulness) atau kesalahan (falsity) sebuah pernyataan, juga dapat digunakan untuk merepresentasikan pernyataan tentang objek tertentu.

Menurut (Sutojo et al., 2011) Pada sistem pakar berbasis *rule*, domain pengetahuan direpresentasikan dalam sebuah kumpulan *rule* berbentuk *IF-THEN*, sedangkan data direpresentasikan dalam sebuah kumpulan fakta-fakta tentang kejadian saat ini. Mesin inferensi membandingkan masing-masing *rule* yang tersimpan dalam basis pengetahuan dengan fakta-fakta yang terdapat dalam database. Jika dibagian *IF* (kondisi) dari *rule* cocok dengan fakta, maka *rule* dieksekusi dan bagian *THEN* (aksi) diletakkan dalam *database* sebagai fakta baru yang ditambahkan.

2.1.2 Web

Menurut (Prasetyo, 2008) *Browser* merupakan suatu perangkat lunak yang berada dikomputer *client* yang mempunyai tugas untuk menerjemahkan informasi yang diterima dari *server web* dan menampilkannya pada *layer* komputer penerima. Contoh: *Internet Explorer, Netscape, Navigator, Opera, Mozilla, Lynx*, dan lain-lain.

Server Web adalah suatu perangkat lunak yang khusus yang bertugas melayani permintaan-permintaan dari browser akan dokumen-dokumen yang tersimpan didalamnya.

2.1.3 Basis data (*Database*)

Menurut (Pahlevi, 2013) Basis data adalah sekumpulan data yang saling berhubungan secara logis beserta deskripsinya, yang digunakan secara bersama-sama dan dirancang untuk memenuhi kebutuhan informasi di suatu tempat.

Menurut (Prasetyo, 2008) menyimpan data dalam file biasa memiliki banyak keterbatasan. Semakin besar ukuran file, pencarian data lebih sulit. File biasa juga tidak memiliki kemampuan untuk mengolah data, misalnya menghitung total nilai, rata-rata, dan lain sebagainya. Operasi dalam basisdata umumnya mengikuti pola yang sama, yaitu melalui rangkaian langkah sebagai berikut :

1. Membuka sambungan basis data server.
2. Memilih dan membuka basis data yang diinginkan.

3. Mengirim perintah untuk mengambil/mengubah/menghapus data.
4. Mengakses pengambilan data.
5. Mengakhiri sambungan.

2.1.4 Validasi Sistem

Menurut (Rosa A & Shalahuddin, 2015) Pengujian intregrasi sebaiknya dilakukan secara berharap, tidak dilakukan secarasatu tahap langsung diakhir untuk menghindari kesulitan penulsuran jika terjadi kesalahan (*error*) Beberapa pendekatan dalam melakukan pengujian untuk validasi sistem antara lain.

1. *Black-Box Testing* (pengujian kotak hitam)

Pendekatan ini dilakukan dengan menguji sistem atau perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Tujuannya untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari sistem atau perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan menggunakan sistem atau perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan *black-box testing* harus dibuat dengan kasus benar dan kasus salah.

2. *White-Box Testing* (pengujian kotak putih)

Pendekatan ini dilakukan dengan menguji sistem atau perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi masukan, dan keluaran yang sesuai dengan spesifikasi yang dibutuhkan. *White-*

box testing dilakukan dengan memeriksa logika dari kode program. Pembuatan kasus uji dapat mengikuti standar pengujian dari standar pemrograman yang ada.

2.2 Variabel Penelitian

Variabel penelitian pada dasarnya adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi dan kesimpulannya (Sugiyono, 2012). Pada penelitian ini yang menjadi variabel adalah penyakit ikan patin, penyakit ikan patin juga mempunyai beberapa jenis yaitu Bintik putih (*white spot*).



Gambar 2.5 Ikan Patin

Sumber: (Rukmana & Yudirachman, 2016)

2.2.1 Bintik putih (*white spot*)

Menurut (Rukmana & Yudirachman, 2016) Bintik putih disebabkan oleh parasit dari bangsa *protozoa* jenis *ichthyoptirus multifilis foquet*. Parasit ini dapat menimbulkan kematian benih secara massal, terutama yang berumur 1-2 bulan. Penyakit ini dapat membunuh ikan patin dalam waktu singkat. Organisme ini akan menempel pada tubuh ikan secara bergerombolan sampai jumlahnya ratusan, sehingga akan terlihat seperti bintik-bintik putih. Tempat ini yang disukai penyakit ini adalah di bawah selaput lendir, sekaligus merusak tempat tersebut. Mencegah penyakit infeksi bintik putih dapat dilakukan menggunakan metil biru atau methilene blue konsentrasi 1% (satu gram betil biru dalam 100 cc air). Caranya, ikan yang sakit dimasukkan dalam larutan tersebut. Ikan patin dibiarkan berada di dalam larutan selama 24 jam..



Gambar 2.6 Bintik putih

Sumber: (Rukmana & Yudirachman, 2016)

2.2.2. *Saprolegnia*

Menurut *Saprolegnia* akibat merupakan salah satu jamur patogen yang sering menyerang ikan. Penyakit yang ditimbulkan akibat serangannya adalah saprolegniasis. Jamur saprolegnis memiliki bentuk menyerupai benang. *Saprolegnia* dapat menyebabkan stres pada ikan.

Menurut (Samadi, 2016) *Saprolegnia* merupakan patogen yang tergolong dalam kelompok tumbuhan jamur (cendawan). Infeksi cendawan tersebut pada ikan menimbulkan penyakit yang disebut dengan penyakit yang disebut sebagai penyakit "*Mycosis*". Gejala penyakit *mycosis* mirip seperti bercak putih yang disebabkan oleh parasit *Ichthyophthirius multifiliis*.



Gambar 2.7 *Saprolegnia*

Sumber : (Rukmana & Yudirachman, 2016)

2.3 *Software Pendukung*

2.3.1 *Xampp*

Menurut (Aditya, 2011) XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MY SQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

2.3.2. *Hypertext Preprocessor (PHP)*

Menurut (Aditya, 2011) *PHP* adalah bahasa skrip yang dapat ditanamkan atau disisipkan kedalam *HTML*. *PHP* banyak dipakai untuk memprogram situs web dinamis. *PHP* dapat digunakan untuk membangun sebuah *CMS*.

Pada awalnya *PHP* merupakan kependekan dari *Personal Home Page* (Situs personal). *PHP* pertama kali dibuat oleh *Rasmus Lerdor* pada tahun 1995. Pada waktu itu *PHP* masih bernama *Form interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

2.3.3 HTML (Hyper Text Markup Language)

Menurut (Saputra, 2012b) *HTML* singkatan dari *Hyper text markup language*. *HTML* bisa disebut bahasa paling dasar dan penting yang digunakan untuk menampilkan dan mengelolah tampilan pada halaman *website*. Menurut sumber yang penulis kutip dari *Wikipedia*, *HTML* digunakan untuk menampilkan berbagai informasi didalam sebuah penjelajah *web internet* dan *formatting hypertext* sederhana yang ditulis ke dalam berkas format *ASCII* agar dapat \

Menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata, disimpan ke dalam format *ASCII* normal sehingga menjadi *homepage* dengan perintah-perintah *HTML*. *HTML* menggunakan 2 macam ekstensi *file* yaitu *.htm* dan *.html*. Format ekstensi berformat *.htm* awalnya hanyalah untuk mengakomodasi pengguna html dalam operasi *DOS*.

Menurut (Raharjo, 2011a) *HTML* adalah singkatan dari *Hyper Text Markup Language*, yaitu bahasa (aturan) yang digunakan untuk menampilkan teks, gambar, *video* dan *audio* ke dalam halaman web. *HTML* merupakan *file* teks yang tersusun atas elemen-elemen yang disebut dengan *tag*. *Tag HTML* diapit dengan tanda lebih kecil (<) dan tanda lebih besar (>), misalnya: <*html*>, <*head*>, <*body*>, <*p*>, dan lain-lain. *Tag HTML* ada yang memiliki pasangan, maka *tag* penutup akan disertai dengan tanda *slash (/)*, misalnya: </*html*>, </*head*>, </*body*>, </*p*>, dan lain-lain. Dokumen atau *file HTML* dapat dibuat menggunakan aplikasi *Text Editor*, dan disimpan dengan ekstensi *.html* atau *.htm*.

Text Editor yang secara khusus dirancang untuk menulis kode-kode HTML. Saat ini, sudah banyak *software text editor* untuk HTML yang ditawarkan secara gratis di *internet*.

2.3.4 CSS(Cascading Style sheet)

Menurut (Saputra, 2012a, pp. 27–29) *Cascading Style Sheet (CSS)* merupakan bahasa pemrograman web yang didesain khusus untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web lebih rapih, terstruktur, dan seragam. CSS merupakan salah satu program wajib disamping html yang harus dikuasai oleh para setiap program web, terlebih lagi itu adalah Web Designer. Tujuan utama CSS adalah untuk memisahkan konten utama dengan tampilan dokumen lainnya (html dan sejenisnya). Dengan adanya pemisah ini, akses konten pada web meningkat. Web yang menggunakan CSS akan lebih ringan dan mudah untuk dibuka dibandingkan dengan web yang tidak menggunakan CSS. Perbedaan ini akan semakin terasa ketika web yang anda buka mempunyai data yang banyak.

Tujuan lainnya adalah untuk mempercepat pembuatan halaman web. Hanya perlu membuat satu property dan property tersebut dapat digunakan pada halaman lainnya, tidak perlu menulis ulang kode program yang digunakan berulang kali. CSS ini dikembangkan oleh *World Wide Web Consortium* atau yang biasa lebih dikenal dengan istilah W3C. Sehingga CSS menjadi bahasa standard dalam pembuatan web. CSS bukan menggantikan kode html, tetapi hanya difungsikan

sebagai penopang atau pendukung (pelengkap) dari file html yang berperan dalam penataan kerangka dan layout.

Dengan menggunakan CSS, akan banyak keuntungan yang dapat kita peroleh, diantaranya:

- a) Memisahkan pembuatan dokumen (CSS dan HTML).
- b) Mempermudah dan mempersingkat pembuatan dan pemeliharaan dokumen web.
- c) Akses web lebih pat saat di-loading (mempercepat pembacaan HTML)
- d) Flaksibel, interaktif, tampilan lebih menarik dan nyaman dipandang.
- e) Lebih kecil ukuran file sehingga bandwidth yang digunakan juga otomatis lebih kecil.
- f) Dapat digunakan pada semua browser.

2.3.5 JavaScript

Menurut (Raharjo, 2011b) *Javascript* adalah bahasa yang berfungsi untuk membuat skrip-skrip program yang dapat dikenal dan dieksekusi oleh *web browser* dengan tujuan untuk menjadikan halaman *web* lebih bersifat interaktif. Meskipun banyak fitur dan bahasa *java* yang diadopsi oleh *javascript*, namun *javascript* dikembangkan secara terpisah dan independen. Jadi, perlu diketahui bahwa *Javascript* itu bukan *Java*. *Javascript* dan *Java* merupakan dua hal yang konsepnya sangat berbeda, meskipun ada kemiripan dalam penulisan sintaksnya. *Javascript* dikembangkan oleh *Netscape* dan merupakan bahasa yang bersifat

terbuka (*open*). Sehingga setiap orang dapat menggunakannya tanpa harus membeli lisensi.

Beberapa contoh penggunaan *Javascript* yang sering dijumpai dalam halaman *web* adalah:

- a. Untuk menampilkan pesan peringatan atau konfirmasi ke *user*
- b. Untuk menampilkan *popup window*
- c. Untuk membuat menu *dropdown*
- d. Untuk melakukan validasi pada saat *user* memasukkan data dalam suatu *form*
- e. Untuk menampilkan tanggal dan waktu

2.3.6 MySQL dan SQL

Menurut (Saputra, 2012b) MySQL merupakan salah satu *database* kelas dunia yang sangat cocok bila dipadukan dengan bahasa pemrograman *PHP*. *MySQL* bekerja menggunakan bahasa *SQL* (*Structure Query Language*) yang merupakan bahasa standar yang digunakan untuk manipulasi *database*. Perintah yang sering digunakan dalam *MySQL* adalah *SELECT* (mengambil), *INSERT* (menambah), *UPDATE* (mengubah), dan *DELETE* (menghapus). Selain itu, *SQL* juga menyediakan perintah untuk membuat *database*, *field*, ataupun *index* untuk menambah atau menghapus data.

Ada beberapa alasan yang menjadikan *database MySQL* sangat diminati oleh para *programmer*, diantaranya :

1. Bersifat open source.

2. Menggunakan bahasa SQL (*Structure Query language*), yang merupakan standar bahasa dalam pengolahan data
3. *Performance* dan *reliable*, pemrosesan *database*-nya sangat cepat dan stabil
4. Sangat mudah dipelajari (*ease of use*).
5. Memiliki dukungan (*group*) pengguna *MySQL*.
6. *Lintas Platform*, dapat digunakan pada berbagai sistem operasi berbeda.
7. *Multiuser*, dimana *MySQL* dapat digunakan oleh banyak *user* dalam waktu yang bersamaan tanpa mengalami konflik.
8. Dan masih banyak lagi.

2.3.7 StarUML

Menurut (A.S & Shalahuddin, 2011) salah satu pemodelan yang saat ini paling banyak digunakan adalah *UML*. *UML (Unified Modeling Language)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisa dan *desain*.

Menurut (A.S & Shalahuddin, 2011) pada *UML 2.3* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori dan macam-macam diagram. Berikut penjelasan singkat dari pembagian kategori diagram tersebut.

1. *Structure diagrams*

Structure diagrams merupakan kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Macam-macam diagram dari kategori ini antara lain, *class diagram*, *object diagram*, *component diagram*, *composite structure diagram*, *package diagram*, dan *deployment diagram*.

2. *Behaviour diagrams*

Behaviour diagrams merupakan kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Macam-macam diagram dari kategori ini antara lain *use case diagram*, *activity diagram*, dan *state machine diagram*.

3. *Interaction diagrams*

Interaction diagrams merupakan kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. Macam-macam diagram dari kategori ini antara lain *sequence diagram*, *communication diagram*, *timing diagram*, dan *interaction overview diagram*.

Menurut (Rosa A & Shalahuddin, 2015) Dalam penelitian ini, diagram yang akan digunakan untuk *desain* sistem yaitu:

1. *Use case diagram*

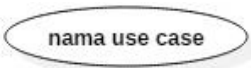


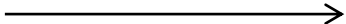
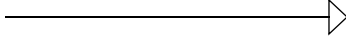

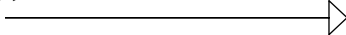
Menurut (Rosa A & Shalahuddin, 2015) *Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu sistem atau lebih aktor dengan

sistem informasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Ada 2 hal utama pada *use case* yaitu pendefinisian apa yang dimaksud actor dan use case.

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri. Jadi walaupun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukaran pesan antarunit atau aktor.

Menurut (Rosa A & Shalahuddin, 2015) Berikut ini adalah simbol-simbol yang digunakan dalam *use case*:

Tabel 2.3 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor</p>
<p>asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi/<i>extend</i></p> <p><<extend>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu. Arah panah mengarah pada <i>use case</i> yang ditambahkan.</p>
<p>generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara 2 buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari fungsi lainnya. Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)</p>
<p>Menggunakan/<i>include/uses</i></p> <p><<include>></p>  <p><<uses>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankannya <i>use case</i> ini. Arah panah mengarah pada <i>use case</i> yang ditambahkan</p>






Sumber : (A.S & Shalahuddin, 2011)

2. Activity Diagram

Menurut (A.S & Shalahuddin, 2011) Diagram aktifitas (activity diagram) menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah system atau proses bisnis . diagram aktivitas menggambarkan aktivitas sistem apa yang akan dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Dibawah ini akan menjelaskan simbol-simbol *Activity Diagram* yaitu:

Tabel 2.4 Simbol *Activity Diagram*

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

Sumber : (A.S & Shalahuddin, 2011)



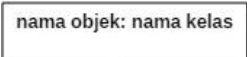

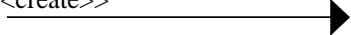
3. Sequence Diagram

Menurut (A.S & Shalahuddin, 2011) Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan

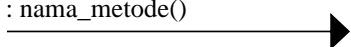
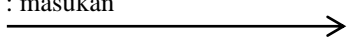
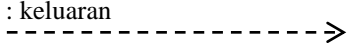
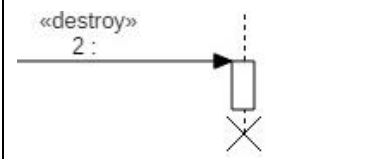
message yang dikirimkan dan diterima antarobjek.oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

simbol-simbol *Sequence Diagram* yaitu:

Tabel 2.5 Simbol *Sequence Diagram*

Simbol	Deskripsi
<p>Aktor/<i>actor</i></p>  <p>nama aktor</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama actor
<p>Garis hidup/<i>lifeline</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
<p>Objek</p>  <p>nama objek: nama kelas</p>	Menyatakan objek yang berinteraksi pesan
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. Aktor tidak memiliki waktu aktif
<p>Pesan tipe <i>create</i></p> <p><<create>></p> 	Menyatakan suatu objek membuat objek yang lain. Arah panah mengarah pada objek yang dibuat

Tabel 2.5 Lanjutan Simbol *Sequence Diagram*

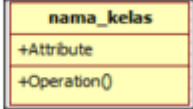

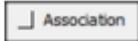
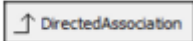
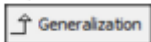
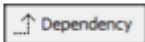
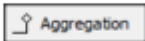
Simbol	Deskripsi
<p>pesan tipe <i>call</i></p> <p>1 : nama_metode()</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi/metode.</p>
<p>Pesan tipe <i>send</i></p> <p>1 : masukan</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya. Arah panah mengarah pada objek yang dituju</p>
<p>pesan tipe <i>return</i></p> <p>1 : keluaran</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu. Arah panah mengarah pada objek penerima</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek lain. Arah panah mengarah pada objek yang diakhiri</p>

Sumber:(A.S & Shalahuddin, 2011)

4. *Class Diagram*

Menurut (A.S & Shalahuddin, 2011) *Class diagram* adalah menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Tabel 2.6 Simbol *class diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem.
Antarmuka (Interface) 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
Asosiasi (Association) 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga di sertai dengan multiplicity.
Asosiasi berarah (Directed Association) 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi berarah biasanya juga disertai dengan multiplicity.
Generalisasi (Generalization) 	Relasi antar kelas dengan makna generalisasi-spesialisasi (Umum-khusus)
Kebergantungan (Dependency) 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi (Aggregation) 	Relasi antar kelas dengan makna semua-bagian (Whole-part)

Sumber: (A.S & Shalahuddin, 2011)

2.4 Penelitian Terdahulu

Untuk mendukung teori yang berkaitan dengan penelitian, peneliti mencantumkan beberapa penelitian terdahulu di bidang sistem pakar.

1. (Lisa Novalia*, 2013) dalam penelitian mereka tentang “**Pengaruh Metil Metsulfuron terhadap jaringan insang patin siam (*Pangasius hypophthalmus*)**” diperoleh kesimpulan: Patin siam (*pangisius hypophthalmus*) merupakan jenis ikan tawar yang dapat dibudidayakan pada kolam tanah dari lahan sawah. Patin siam yang dipelihara dengan cara tersebut kemungkinan dapat terpapar senyawa kimia akibat penggunaan herbisida yang tidak terkontrol Metil metsulfuron merupakan bahan aktif herbisida yang sistemik untuk gulma. Penelitian bertujuan untuk mengetahui pengaruh berbagai tingkat konsentrasi metil metsulfuron (0;15,6 dan 39 ppm) terhadap perubahan jaringan insang patin siam. Hasil pengamatan terhadap jaringan insang menunjukkan terjadi kerusakan berupa hiperplasia lamella insang, deskuamasi, kongesti dan hemoragik. Konsentrasi metil metsufuron yang semakin tinggi menyebabkan tingkat kerusakan jaringan insang yang beragam.
2. (Dewi, Lestari, & Lestari, 2015) dalam penelitian mereka tentang “ **Sistem pakar Diagnosis penyakit Ikan Koi dengan Metode *Bayes***” diperoleh kesimpulan: Penanganan penyakit pada ikan koi sejak dini dapat menghindari penularan penyakit ke ikan koi lain dalam satu kawanan dan dapat menyelamatkan ikan koi dari kematian. Terbatasnya jumlah pakar,

serta kurangnya penyebaran pengetahuan, menyebabkan diperlukannya sistem pakar untuk diagnosis penyakit ikan koi. Sistem pakar diagnosis penyakit ikan koi dibangun dengan bahasa pemrograman web PHP dan database MySQL. Representasi pengetahuan menggunakan kaidah produksi, proses inferensi menggunakan forward chaining dan proses perhitungan nilai kepastian terjadinya penyakit dilakukan menggunakan metode bayes. Para peternak dapat mendiagnosis penyakit yang terjadi pada ikan koi dan mengetahui cara penanganan penyakit dengan menjawab pertanyaan-pertanyaan yang diajukan sistem.

3. (Elfani, 2013) dalam penelitian mereka tentang **“Sistem Pakar Mendiagnosa Penyakit pada Ikan konsumsi air tawar Berbasis Website”** diperoleh kesimpulan usaha peternakan ikan merupakan salah satu peluang bisnis yang prospeknya menjanjikan khususnya ikan konsumsi air tawar. Namun masih banyaknya kendala yang ditemui oleh para pengusaha petenak ikan, seperti penyakit ikan, yang mengakibatkan peternak ikan panen tidak maksimal dan kurangnya jumlah pakar ikan sebagai tempat konsultasi. Sehingga perlu adanya media bantu berupa sistem yang dapat memberi solusi kapan saja. Dengan demikian peternak ikan dapat mengetahui penyakit yang menyerang ikan ternaknya lebih dini. Pada penelitian ini akan dibangun sebuah media konsultasi dengan pendekatan system pakar, dengan menggunakan metode penelusuran fakta forward chaining dan metode kepastiannya menggunakan theorema bayes yaitu metode untuk menghitung nilai kepastian suatu penyakit. Tahap

pengembangan aplikasi diawali dengan tahapan analisis system yaitu analisis data dan deskripsi kebutuhan sistem, membangun basis pengetahuan, pembuatan Diagram Konteks, Diagram Alir Data, Entity Relationship Diagram, dan membuat struktur tabel, perancangan mapping table, dan perancangan menu antarmuka. Setelah tahap perancangan selesai maka dilanjutkan pada tahap implementasi menggunakan PHP dengan Framework Codeigniter sebagai bahasa pemrograman dan MySQL sebagai database. Terakhir pengujian sistem menggunakan Black Box Tes dan Alpha Test. Dari penelitian yang dilakukan menghasilkan perangkat lunak “Sistem Pakar Mendiagnosa Penyakit Pada Ikan Konsumsi Air Tawar Berbasis Website” yang dapat bekerja seperti layaknya seorang pakar perikanan. Informasi yang dihasilkan adalah nama penyakit, definisi, penyebab, gejala – gejala yang menyertai, pengendalian dan nilai bayes sebagai total dari perhitungan probabilitas penyakit yang diderita.

4. *(Zhang, Pridgeon, & Klesius, 2013) dalam penelitian mereka tentang “Expression and activity of recombinant proaerolysin derived from Aeromonas hydrophila cultured from diseased channel catfish” diperoleh kesimpulan: Aerolysin is one of the putative toxins in extracellular products (ECP) produced by Aeromonas hydrophila, an important pathogen of catfish. To better understand the molecular mechanism and mode of action of this toxin, proaerolysin-coding gene was cloned from the genomic DNA of an A. Hydrophila strain, cultured from diseased channel catfish, and heterologously expressed in E. coli. Functional recombinant proaerolysin*

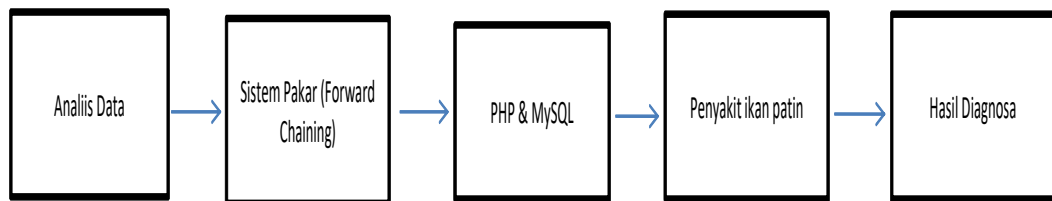
was obtained, revealing some unique properties. The purified recombinant proaerolysin was inactive but could be activated by treatment with furin, trypsin, and ECP although different treatments produced different cleavage profiles and resulted in differential hemolytic and cytotoxic activities. The highest activity was observed from aerolysin processed by furin while treatment of proaerolysin with trypsin and ECP resulted in reduced activities. The unprocessed proaerolysin, though not hemolytic *in vitro*, had the same cytopathic effect on cultured walking catfish gill cells as the furin-processed had. *In vivo* assay, the recombinant proaerolysin was found to be lethal to catfish when injected via intraperitoneal (IP) route. The lethal toxicity was acute and dose-dependent, as observed in IP Injection of live *A. hydrophila*. This is the first recombinant proaerolysin confirmed to be a virulence factor; the recombinant protein could be used to further evaluate virulence, pathogenicity and antigenicity associated with *A. Hydrophila* infection.

5. (Baldissera et al., 2017) dalam penelitian mereka **“Melaleuca alternifolia essential oil prevents oxidative stress and ameliorates the antioxidant system in the liver of silver m catfish (*Rhamdia quelen*) naturally infected with *Ichthyophthirius multifiliis*”** diperoleh kesimpulan: Oxidative stress has been linked to a pathological mechanism that contributes to the initiation and progression of hepatic injury during *Ichthyophthirius multifiliis* infection. Thus, natural compounds with antioxidant and free radical scavenger properties, such as the *Melaleuca alternifolia* essential

oil (TTO), might help to prevent or reduce hepatic damage. Thus, the aim of this study was to investigate whether TTO is capable of preventing or reducing hepatic oxidative damage in silver cat fish naturally infected with I. multifilii. After a TTO bath treatment of 1 h/day for 4 days, the number of trophonts on the skin of cat fish was reduced by 94.87% compared to the control group. Hepatic samples from fish infected by I. Multifiliis showed increased thiobarbituric acidreactive substances (TBARS) and protein carbonylation levels, while catalase (CAT) activity decreased compared to uninfected and untreated animals. Moreover, the histopathology analyses demonstrated the presence of ne-crosis, nuclear lateralization and cariorrexis, as well as swelling and vacuolization of cytoplasm. Treatment with TTO reduced hepatic TBARS and protein carbonylation levels and ameliorated the antioxidant system of infected fish, as well as preserve the hepatocytes morphology, but not prevented the occurrence of necrosis. Based on our findings, we propose that TTO is an alternative therapeutic option for silver catfish infected with I. Multifiliis Moreover, TTO treatment is a useful approach for avoiding or minimizing the hepatic oxidative stress caused by I. multifiliis, thus improving the hepatic antioxidant system.

2.5 Kerangka Pemikiran

Berdasarkan kerangka pemikiran diketahui sebagai suatu gambaran menjelaskan garis besar dari kerangka pemikiran. Berikut gambaran kerangka pemikiran dari penelitian.



Gambar 2.8 Kerangka Pemikiran

Sumber : (Pengolahan Data Peneliti, 2017)

Data-data yang dibutuhkan berkaitan dengan penyakit ikan Ikan Patin dianalisis terlebih dahulu agar lebih sederhana atau mudah dilakukan proses pengolahan datanya. Data-datanya tersebut kemudian diolah menggunakan sistem pakar menggunakan metode *forward chaining*. Sistem pakar yang menggunakan metode *forward chaining* ini menggunakan bahasa pemograman *PHP* dan *databaseMySQL* yang dapat digunakan untuk mendiagnosa penyakit ikan Patin dan menghasilkan *output* (hasil diagnosa).

BAB III METODE PENELITIAN

3.1. Desain Penelitian

Desain penelitian diketahui sebagai gambaran tentang proses yang dilakukan dalam penelitian. Berikut merupakan desain penelitian dalam penelitian ini:



Gambar 3.1 Desain penelitian

Sumber : (Data Penelitian, 2017)

Berikut ini adalah penjelasan dari tahap-tahap desain penelitian pada gambar dihalaman sebelumnya:

1. Mengidentifikasi masalah

Penelitian diawali dengan melakukan studi pendahuluan untuk mengidentifikasi permasalahan yang berkaitan dengan topik penelitian agar peneliti mendapatkan apa sesungguhnya menjadi masalah untuk dipecahkan. Pada tahap mengidentifikasi masalah adalah untuk menunjukkan bahwa kurangnya wawasan masyarakat umum akan penyakit ikan Patin.

2. Analisis masalah

Setelah mengidentifikasi masalah maka selanjutnya yang dilakukan adalah menganalisis masalah dimana setelah menentukan variabel yang akan di teliti, maka perlu adanya menganalisis variabel tersebut apakah layak untuk dilakukannya penelitian pada masalah tersebut.

3. Pengumpulan data

Pengumpulan data dilakukan guna untuk mendapatkan rincian tentang variabel yang di ambil untuk di teliti supaya memperlengkap informasi yang di kumpulkan, dan dalam penelitian ini teknik pengumpulan data menggunakan wawancara sebagai alat untuk mendapatkan data yang diperlukan

4. Menganalisis data

Menganalisis data adalah dimana setelah melakukan pengumpulan data akan didatakannya data-data yang diperlukan, maka pada tahap menganalisis

data adalah menyusun secara berurut mengenai data yang telah ditepkan agar lebih mudah dipahami dan di terapkan dalam sistem pakar.

5. *Forward chaining*

Forward chaining adalah metode dari penganalisan data, yaitu setelah data yang di analisis maka akan dikerjakan sesuai dengan metode yang di pilih, guna mencapai tujuan dalam pengolahan data yang telah ada.

6. Perancangan sistem

Perancangan system merupakan bagian dari merancang web dari sistem pakar yang menggunakan metode *forward chaining* agar dapat menentukan penyakit ikan patin sesuai dengan *rule* yang sudah ada.

7. Implementasi sistem

Implementasi sistem merupakan tahap akhir dari kerangka kerja penelitian yaitu dimana sistem yang sudah di buat dan di rancang dapat di uji cara kerjanya, untuk mengetahui apakah aplikasi tersebut berjalan dengan baik dan dapat membantu masyarakat untuk mendiagnosa penyakit ikan patin berbasis *web*.

3.2. Teknik Pengumpulan Data

Teknik pengumpulan data merupakan cara-cara yang digunakan peneliti dalam mendapatkan data-data yang berkaitan dengan pokok bahasan dalam rangka untuk mendukung penelitian yang sedang dilakukan. Teknik pengumpulan data yang digunakan dalam penelitian ini adalah:

1. Wawancara

Penelitian ini mendapatkan data-data penyakit dengan cara wawancara langsung pada Dokter di Stasiun Karantina Ikan, Pengendalian Mutu dan Keamanan Hasil Perikanan Kelas 1 Batam. Dalam metode wawancara, pembicaraan selama proses wawancara dilakukan Suatu bentuk komunikasi verbal, semacam percakapan yang bertujuan untuk memperoleh informasi (Sudaryono, 2015). permasalahan yang akan ditanyakan yaitu hal-hal yang berkaitan dengan penyakit ikan patin.

2. Studi literatur

Peneliti melakukan studi *literatur* dengan mengumpulkan, membaca, dan memahami referensi teoritis yang berasal dari buku-buku teori, buku elektronik (*e-book*), jurnal-jurnal penelitian, dan sumber pustaka otentik lainnya yang berkaitan dengan penelitian.

3.3. Operasional Variabel

Variabel harus didefinisikan secara operasional agar lebih mudah dicari hubungannya antara satu variabel dengan lainnya dan pengukurannya. Adapun manfaat operasionalisasi variabel antara lain: untuk mengidentifikasi kriteria yang dapat diobservasi yang sedang didefinisikan, menunjukkan bahwa suatu konsep atau objek mungkin mempunyai lebih dari satu definisi operasional, dan untuk mengetahui bahwa definisi operasional bersifat unik dalam situasi dimana definisi tersebut harus digunakan. Definisi operasional adalah suatu definisi yang

didasarkan pada karakteristik yang dapat diobservasi dari apa yang sedang didefinisikan .

Variabel yang digunakan dalam penelitian ini adalah penyakit bintik putih (*white spot*) dan *saprolegnia*. Terdapat beberapa gejala umum penyakit ikan Patin sebagai berikut:

Tabel 3.1 Variabel dan indikator

Variabel	Indikator
Penyakit ikan Patin	Bintik putih (<i>White Spot</i>) <i>Saprolegnia</i>

(Sumber : Data Penelitian, 2017)

3.4. Perancangan Sistem

Menurut (A.S & Shalahuddin, 2011) perancangan sistem merupakan upaya untuk mengkontruksi sebuah sistem yang memberikan kepuasan akan spesifikasi kebutuhan fungsional, memenuhi target,memenuhi kebutuhan secara implisit atau eksplisit dari segi performa maupun penggunaan sumber daya, kepuasan batasan pada proses desain dari segi biaya, waktu dan perangkat.

3.4.1. Desain Basis Pengetahuan

Sebelum melakukan desain basis pengetahuan, peneliti telah melakukan proses akuisi pengetahuan dengan mengumpulkan pengetahuan dan fakta dari sumber-sumber yang tersedia. Sumber pengetahuan dan fakta diperoleh melalui wawancara dengan dokter ikan dan studi literatur tentang materi yang berkaitan dengan penyakit ikan patin. Sumber pengetahuan dan fakta yang didapat berupa data-data yang berhubungan dengan penyakit ikan patin, gejala penyakit dan juga solusi mengatasinya. Pengetahuan dan fakta tersebut ditampilkan dengan tabel bagian indikator (Tabel 3.2), tabel indikator penyakit ikan patin (Tabel 3.3), tabel gejala (Tabel 3.4) dan tabel indikator dan gejala penyakit ikan patin (Tabel 3.5).

Tabel 3.2 Data indikator

Kode	Nama indikator
IND001	Bintik Putih(<i>White Spot</i>)
IND002	<i>Saprolegnia</i>

Sumber : (Data Penelitian, 2017)

Tabel 3.3 Indikator penyakit ikan Patin

Indikator	Gejala	Solusi
<i>White Spot</i>	<ol style="list-style-type: none"> 1. Ikan akan tampak lemah dan terapung di permukaan. 2. Tubuh berubah menjadi pucat dan mengelupas. 3. Perubahan fisik seperti timbul bintik-bintik putih pada tubuh. 4. Sirip tampak robek-robek dan ingang memucat. 5. Terjadinya kerusakan pada kulit dan insang memicu ikan mengalami stres. 6. Ikan sering mengosokkan tubuh pada dasar dan dinding kolam 	<ol style="list-style-type: none"> 1. Siapkan wadah berupa bak untuk mengobati ikan yang sakit. 2. Buat larutan baku dengan mencampurkan 1 gram Methylene Blue ke dalam 100 ml air bersih. 3. Teteskan larutan baku tadi ke dalam bak sebanyak 2-4 ml untuk setiap 4 liter air. 4. Rendam ikan yang sakit selama 24 jam dan ulangi pengobatan sebanyak 3-5 kali. dalam selang waktu satu hari.

(Sumber : Data Penelitian, 2017)

Tabel 3.3 Lanjutan Indikator penyakit ikan Patin

<i>Saprolegnia</i>	<ol style="list-style-type: none"> 1. Menyerang ikan yang luka dan lemah. 2. Menyerang telur. 3. Kepala, tutup insang, dan sirip ditumbuhi benang halus seperti kapas. 4. Baik telur maupun ikan yang terinfeksi terdapat bercak putih kecoklatan. 5. Luka-luka pada bagian badan akibat gigitan hama. 	<ol style="list-style-type: none"> 1. Rendamkan ikan patin didalam larutan <i>Malachite green oxalate</i> dengan dosis 2,5-3 g air selama 30menit. 2. Oles bagian tubuh ikan yn terluka dengan obat merah yang telah diencerkan terlebih dahulu, oleskan sebanyak 10 kali. 3. Telur rendam malachyte green oxalate 0,1-0.2 ppm selama 1 jam atau 5-10 ppm selama 15 menit.
--------------------	---	---

Tabel 3. 4 Gejala penyakit ikan Patin

Kode	Nama gejala
GJL001	Ikan akan tampak lemah dan terapung di permukaan.
GJL002	Tubuh berubah menjadi pucat dan mengelupas.
GJL003	Perubahan fisik seperti timbul bintik-bintik putih pada tubuh.
GJL004	Sirip tampak robek-robek dan ingang memucat.
GJL005	Terjadinya kerusakan pada kulit dan insang memicu ikan mengalami stres.
GJL006	Ikan sering mengosokkan tubuh pada dasar dan dinding kolam.
GJL007	Menyerang ikan yang luka dan lemah.
GJL008	Menyerang telur.
GJL009	Kepala, tutup insang, dan sirip ditumbuhi benang halus seperti kapas.
GJL010	Baik telur maupun ikan yang terinfeksi terdapat bercak putih kecoklatan
GJL011	Luka-luka pada bagian badan akibat gigitan hama

(Sumber : Data Penelitian, 2017)

Tabel 3. 5 Data gejala *White spot*

Kode	Nama Gejala
GJL001	Ikan akan tampak lemah dan terapung di permukaan.
GJL003	Tubuh berubah menjadi pucat dan mengelupas.
GJL004	Perubahan fisik seperti timbul bintik-bintik putih pada tubuh.
GJL005	Sirip tampak robek-robek dan insang memucat.
GJL006	Terjadinya kerusakan pada kulit dan insang memicu ikan mengalami stres.
GJL007	Ikan sering mengosokkan tubuh pada dasar dan dinding kolam.

Sumber : (Data Penelitian, 2017)

Tabel 3.6 Data gejala *Saprolegnia*

Kode	Nama Gejala
GJL002	Menyerang ikan yang luka dan lemah
GJL008	Menyerang telur.
GJL009	Kepala, tutup insang, dan sirip ditumbuhi benang halus seperti kapas.
GJL010	Baik telur maupun ikan yang terinfeksi terdapat bercak putih kecoklatan
GJL011	Luka-luka pada bagian badan akibat gigitan hama

Sumber : (Data Penelitian, 2017)

Tabel 3.7 Tabel indikator dan gejala penyakit ikan Patin

Kode penyakit	Kode gejala
IND001	GJL001,GJL003,GJL004,GJL005, GJL006,GJL007.
IND002	GJL002,GJL008.GJL009,GJL010,GJL011.

Sumber : (Data Penelitian, 2017)

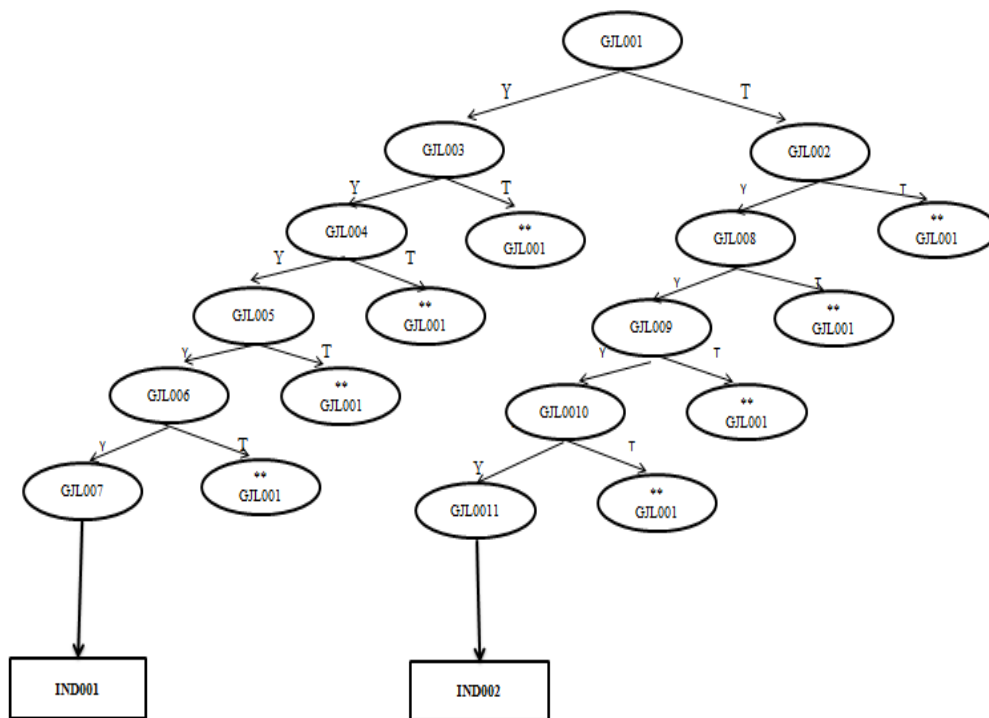
Berdasarkan data aturan yang telah disusun, maka kaidah (*rule*) yang akan digunakan dalam sistem pakar adalah sebagai berikut:

1. Kaidah 1: IF GJL001 AND GJL003 AND GJL004 AND GJL005 AND GJL006 AND GJL007 THEN IND001.
2. Kaidah 2: IF GJL002 AND GJL008,GJL009,GJL010,GJL011 THEN IND002.

Tabel 3.8 Keputusan

Penyakit Gejala	IND 001	IND 002
GJL001	√	
GJL002		√
GJL003	√	
GJL004	√	
GJL005	√	
GJL006	√	
GJL007	√	
GJL008		√
GJL009		√
GJL010		√
GJL011		√

Sumber : (Data Penelitian, 2017)



Gambar 3.2 Pohon keputusan

(Sumber : Data Penelitian, 2017)

Keterangan:

GJL001 = Gejala 001 GJL006 = Gejala 007

IND001 = Indikator 001

GJL002 = Gejala 002 GJL007 = Gejala 008

IND002 = Indikator 002

GJL003 = Gejala 003 GJL008 = Gejala 009

Y = ya

GJL004 = Gejala 004 GJL009 = Gejala 010

T = tidak

GJL005 = Gejala 005 GJL010 = Gejala 011

** GJL001 = kembali kegejala G01

GJL006 = Gejala 006

Data gejala ditentukan sebagai keadaan awal dalam sistem saat melakukan penelusuran sebelum diperoleh sebuah kesimpulan. Pohon keputusan pada gambar 3.2 digunakan untuk memperlihatkan hubungan terkait antar gejala yang ada. Arah penelusuran pada pohon keputusan tersebut dimulai dari simpul akar (yang paling

atas) ke bawah. Berdasarkan gambar di atas, *user* akan diberikan urutan pertanyaan-pertanyaan tentang gejala-gejala yang dialami. *User* diminta untuk menjawab “ya” apabila gejala yang dialami sesuai dan “tidak” apabila gejala tersebut tidak dialami. Alur penelusuran sistem pakar ini dimulai dari GJL001, yaitu Ikan akan tampak lemah dan terapung di permukaan. Gejala ini dipilih sebagai keadaan awal dalam penelusuran karena gejala ini adalah gejala yang paling mudah di diagnosa atau diperiksa.

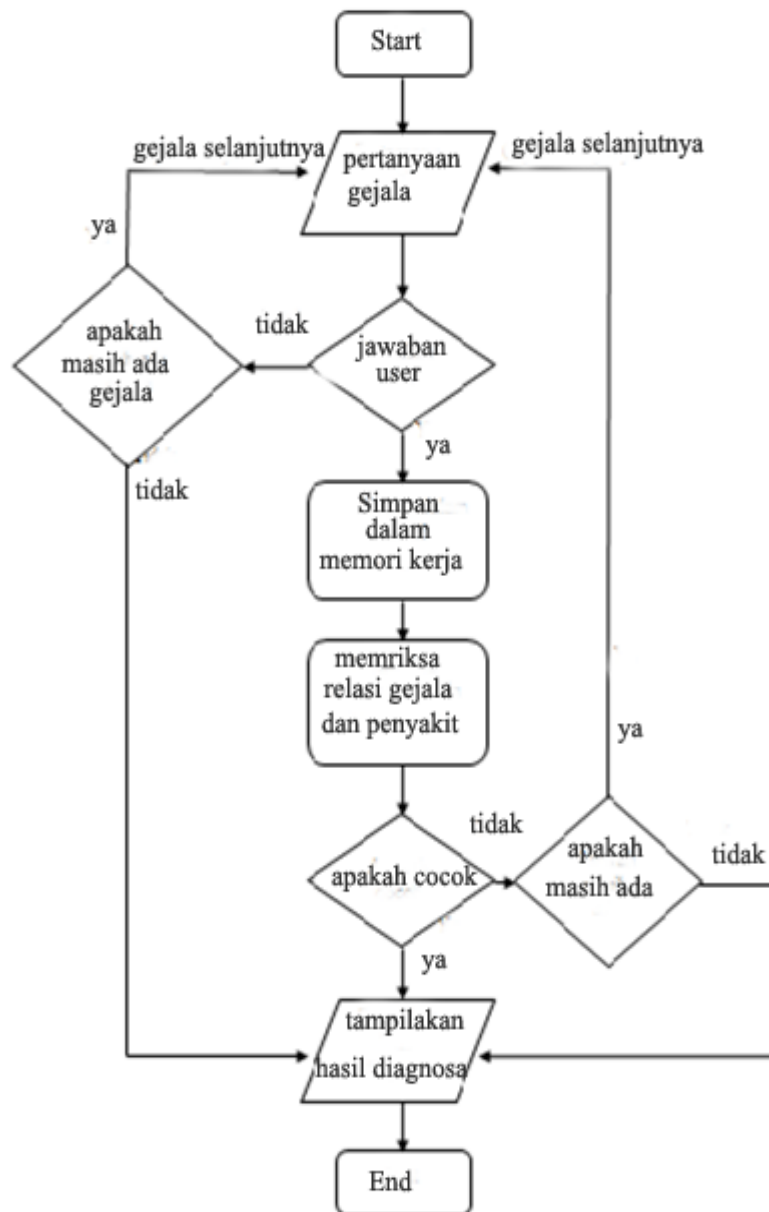
Proses penelusuran selanjutnya tergantung bagaimana jawaban yang diberikan pengguna. Jika pengguna memberikan jawaban “ya”, maka penelusuran menuju simpul kiri pada level berikutnya GJL003 dan jika pengguna memberikan jawaban “tidak”, maka penelusuran menuju simpul kanan pada level berikutnya GJL002. Begitu seterusnya sampai penelusuran menemukan simpul IND atau simpul *. Simpul IND tersebut merupakan bagian dari Indikator. misalnya GJL001 yaitu indikator berada di IND 001, yaitu *white spot*. Simpul ** berarti tidak menghasilkan kesimpulan tertentu. Pada sistem pakar ini, jika penelusuran menemukan simpul** maka sistem akan kembali melakukan penelusuran mulai dari keadaan awal (simpul GJL001).

3.4.2. Struktur Kontrol (Mesin inferensi)

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

1. Mengajukan pertanyaan tentang gejala x penyakit kepada pengguna. Jika jawaban pengguna “Ya” maka sistem akan melakukan langkah 3. Jika jawaban pengguna “Tidak” maka sistem akan melakukan langkah 4.
1. Menyimpan gejala dalam memori kerja lalu memeriksa relasi gejala dengan penyakit yang telah dibuat. Jika ada relasi yang cocok maka sistem akan melakukan langkah 5. Jika tidak ada aturan yang cocok maka sistem akan melakukan langkah 4.
2. Memeriksa apakah masih ada gejala lain yang belum ditanyakan. Jika masih ada, maka sistem akan mengajukan pertanyaan tentang gejala penyakit selanjutnya kepada pengguna dan ulangi langkah 2 sampai dengan 4. Jika tidak ada, maka sistem akan melakukan langkah 5.
3. Menampilkan hasil diagnosa.

Berikut ini adalah gambar *flowchart* mesin inferensi yang digunakan dalam sistem pakar ini.



Gambar 3.2 Flowchart

(Sumber : Data Penelitian, 2017)

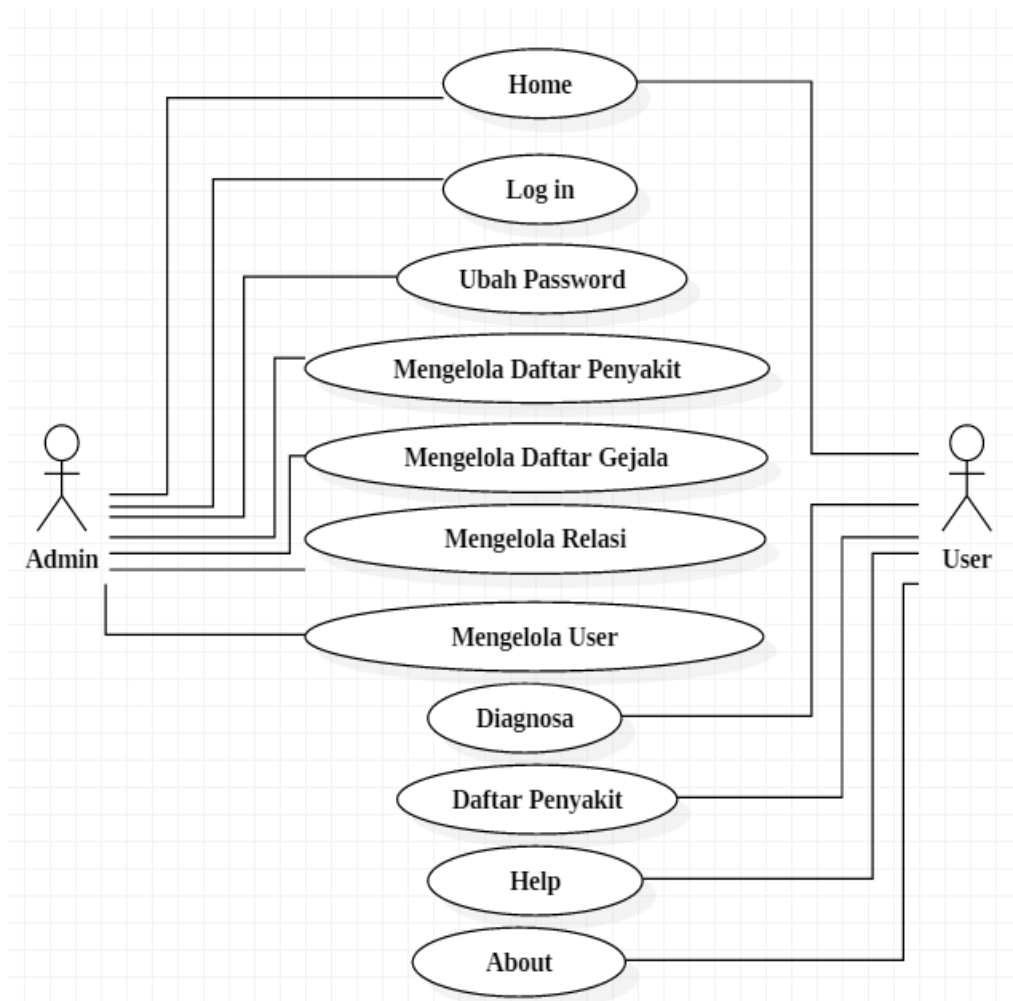
3.4.3. *DesainUML (Unified Modeling Language)*

Desain sistem pada penelitian ini menggunakan bahasa pemodelan *Unified Modelling Language (UML)* yang digambarkan dengan bantuan aplikasi *StarUML* versi 2.5.1. Diagram *UML* yang akan digunakan dalam penelitian ini antara lain:

1. *Use Case Diagram*

Aktor yang digunakan dalam sistem pakar ini terdiri dari 2 orang yaitu admin dan *user*. Dalam sistem pakar ini yang berperan sebagai admin adalah peneliti sendiri sedangkan *user* adalah masyarakat umum yang ingin menangani permasalahan yang berkaitan dengan penyakit ikan kerapu *tiger*. *Use case* yang terdapat dalam sistem antara lain *registrasi*, lupa *password*, *log in*, *home*, ubah *password*, mengelola daftar penyakit, mengelola daftar gejala, mengelola relasi, mengelola bobot gejala, profil, diagnosa, hasil diagnosa dan *log out*.

Use casediagram yang dirancang untuk sistem pakar dalam penelitian ini dapat dilihat pada gambar dibawah ini.



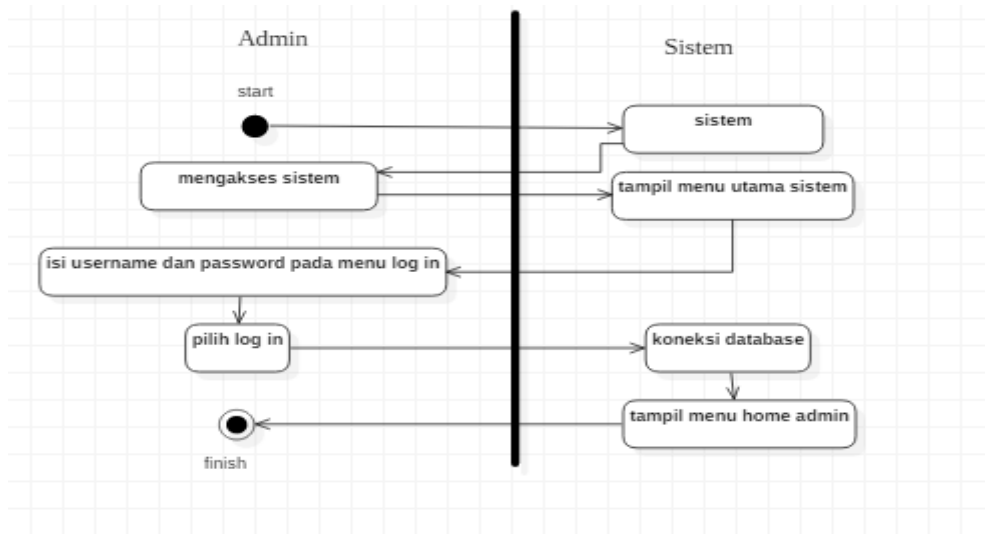
Gambar 3.3 *Use Case Diagram*

Sumber : (Data Penelitian, 2017)

2. *Activity diagram*

Activity diagram menggambarkan aktifitas yang dapat dilakukan oleh sistem atau menu yang ada pada perangkat lunak, bukan apa yang dilakukan oleh aktor (A.S. dan Shalahuddin, 2013: 161). *Activity diagram* yang dirancang untuk sistem pakar dalam penelitian ini akan ditunjukkan melalui gambar-gambar dibawah ini.

a. *Activity admin login*

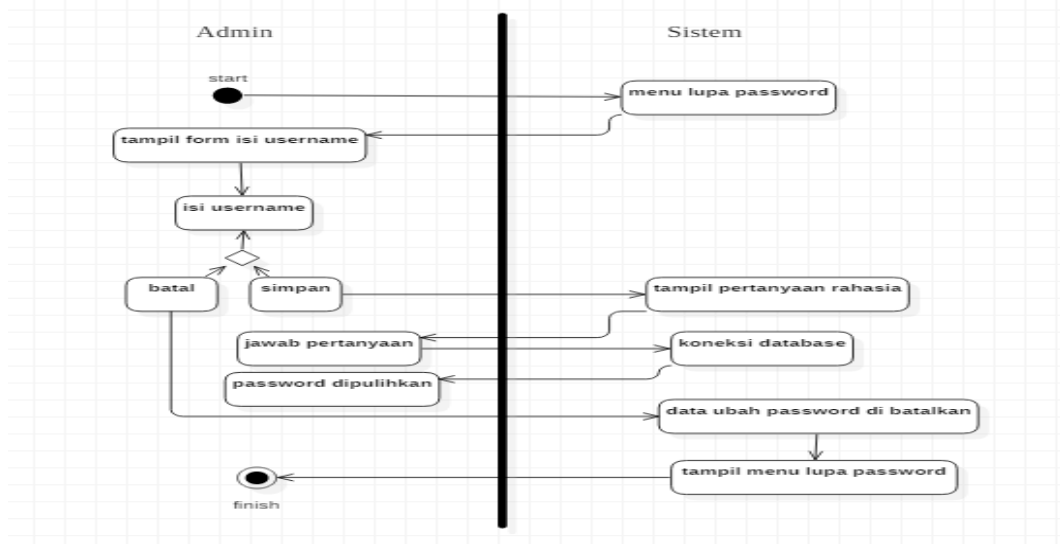


Gambar 3.4 *Diagram activity admin login*

Sumber : (Data Penelitian, 2017)

b. *Activity admin lupa password*

Untuk menulupa *password admin sama user activity diagramnya sama.*

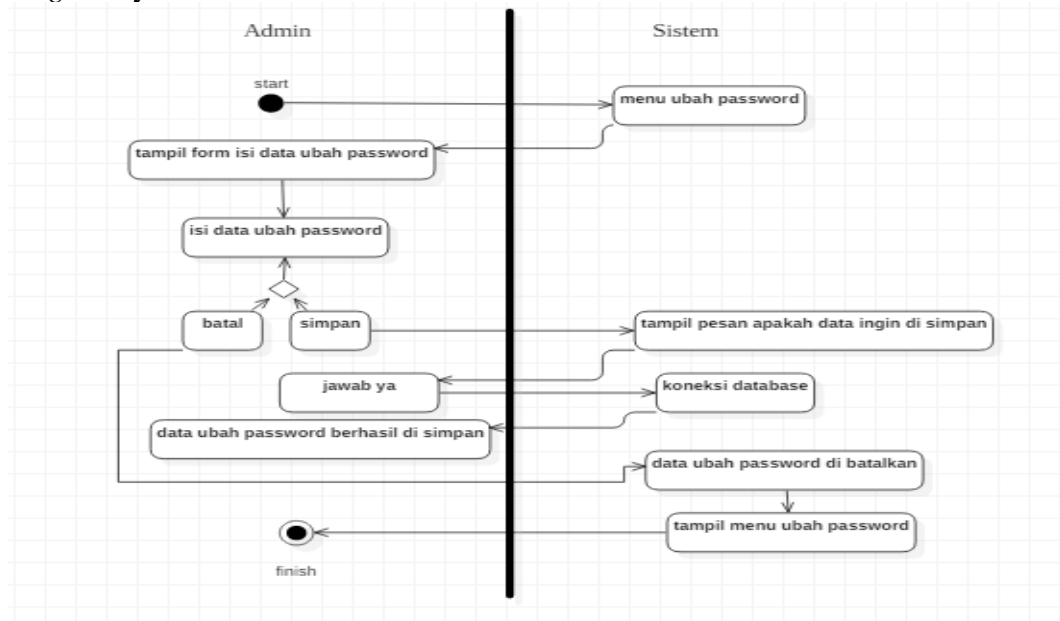


Gambar 3.5 *Diagram activity adminlupa password*

Sumber : (Data Penelitian, 2017)

c. *Activity admin ubah password*

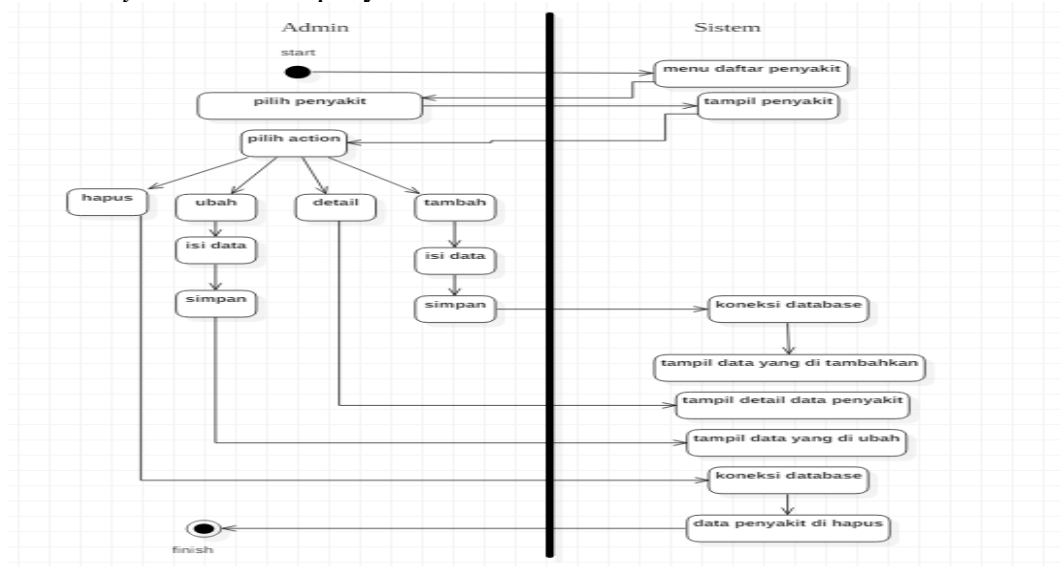
Untuk menuubah *password admin* sama *user activity diagramnya* sama.



Gambar 3.6 *Diagram activity admin ubah password*

Sumber : (Data Penelitian, 2017)

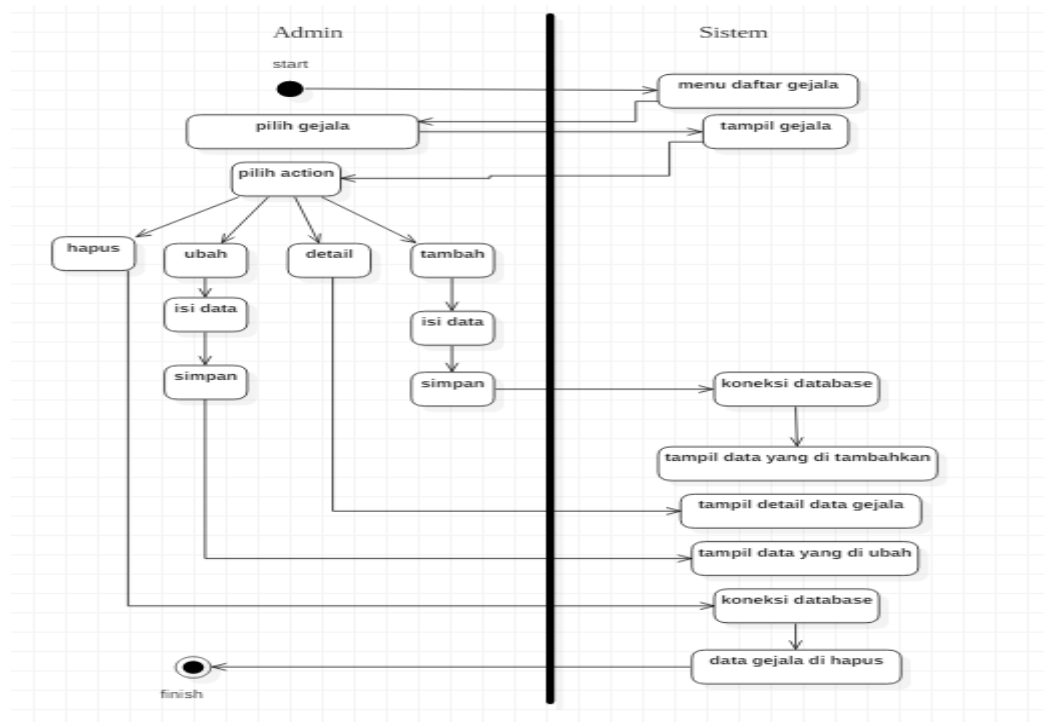
d. *Activity admin daftar penyakit*



Gambar 3.7 *Diagram activity admin daftar penyakit*

Sumber : (Data Penelitian, 2017)

e. *Activity admin* daftar gejala



Gambar 3.8 *Diagram activity admin* daftar gejala

Sumber : (Data Penelitian, 2017)

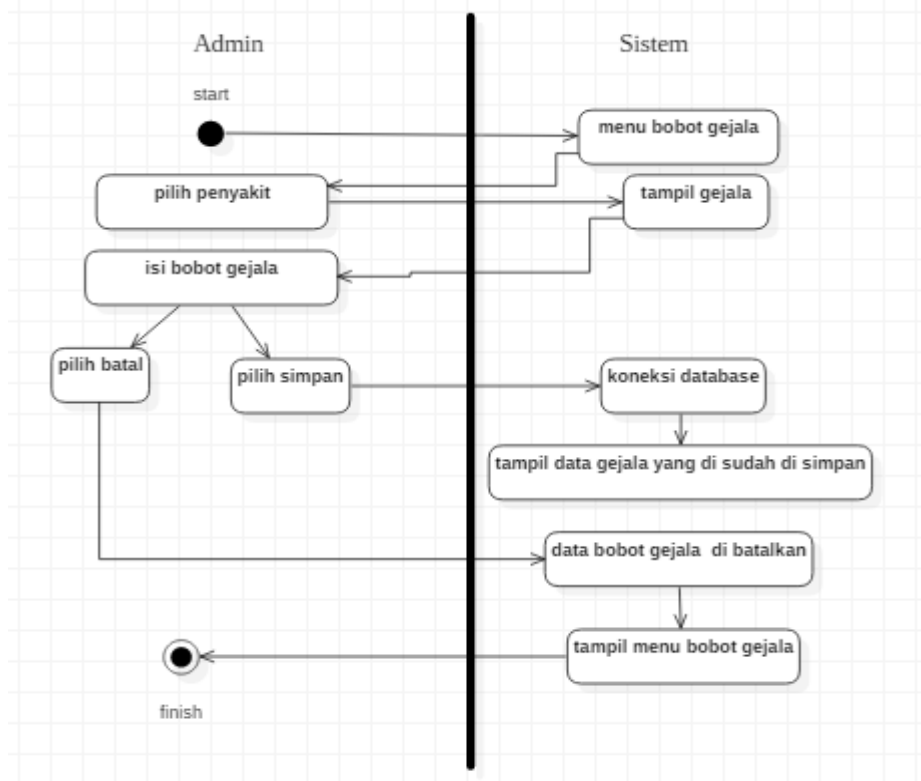
f. *Activity admin* mengelola relasi



Gambar 3.9 *Diagram activity admin* relasi

Sumber : (Data Penelitian, 2017)

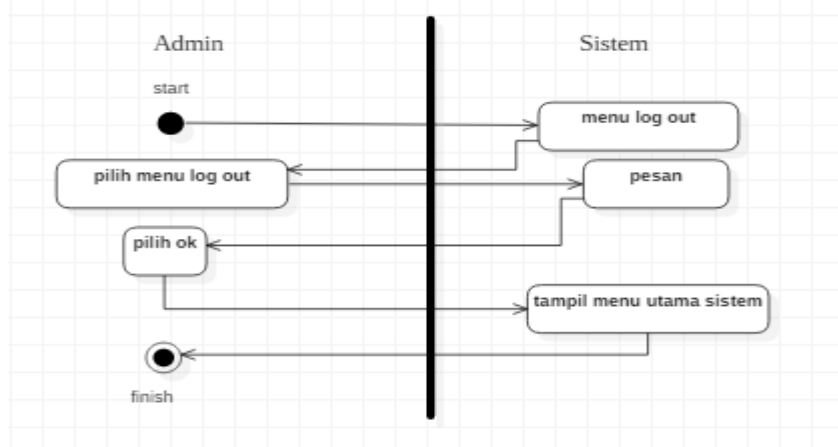
g. *Activity admin* mengelola bobot gejala



Gambar 3.10 *Diagram activity admin* mengelola bobot gejala

Sumber : (Data Penelitian, 2017)

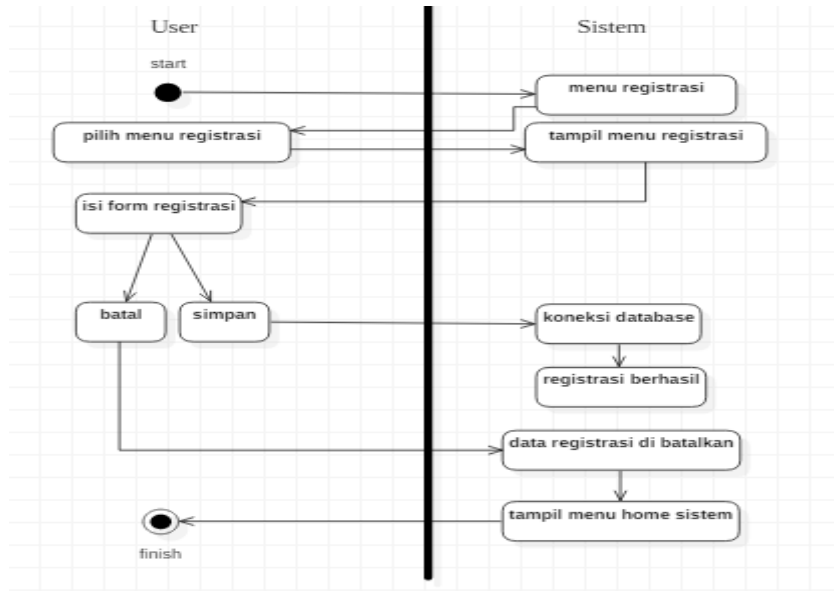
h. *Activity admin* log out



Gambar 3.11 *Diagram activity admin* log out

Sumber : (Data Penelitian, 2017)

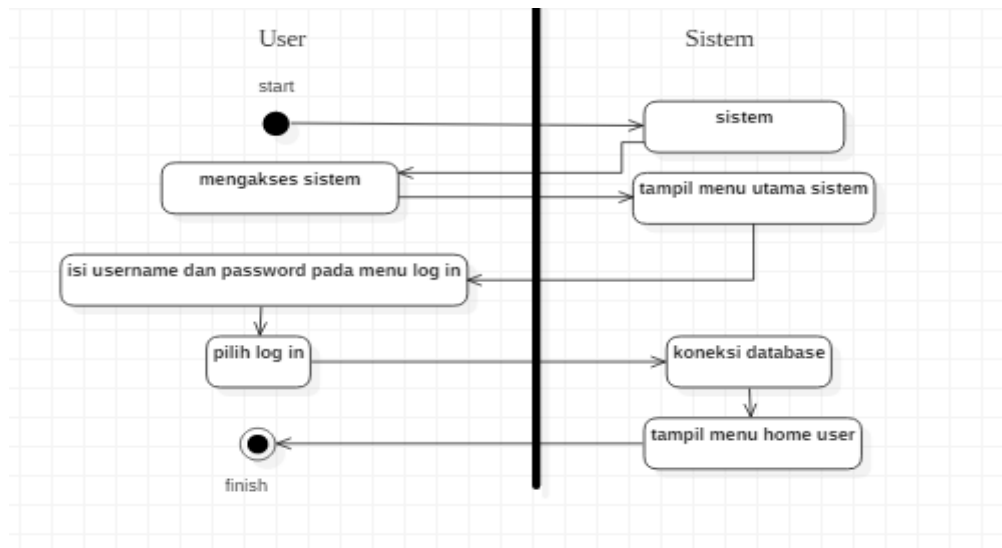
i. Activity user registrasi



Gambar 3.12 Diagram activity user registrasi

Sumber : (Data Penelitian, 2017)

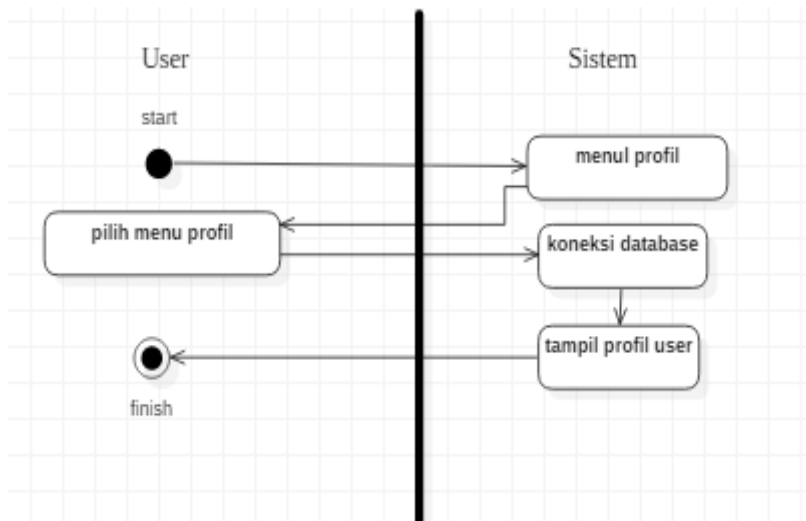
j. Activity user log in



Gambar 3.13 Diagram activity user log in

Sumber : (Data Penelitian, 2017)

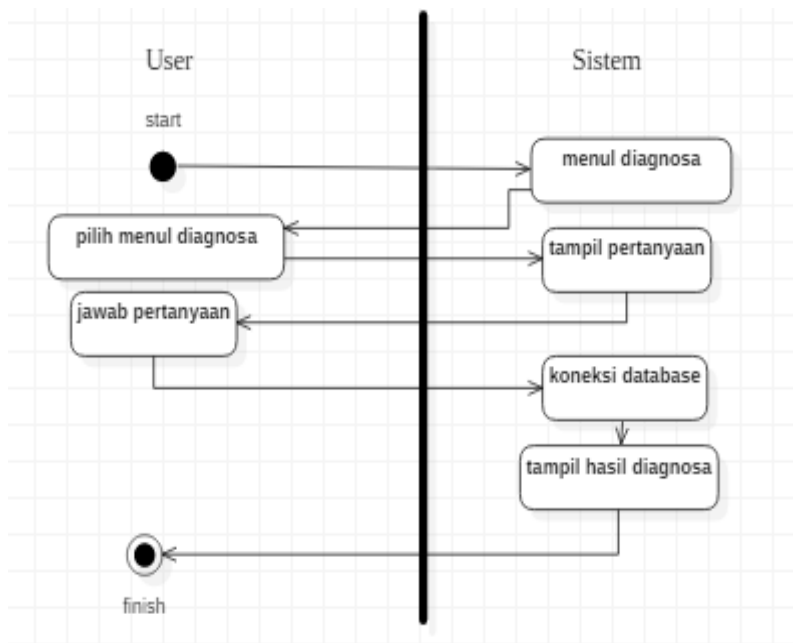
k. *Activity user profil*



Gambar 3.14 *Diagram activity user ubah password*

Sumber : (Data Penelitian, 2017)

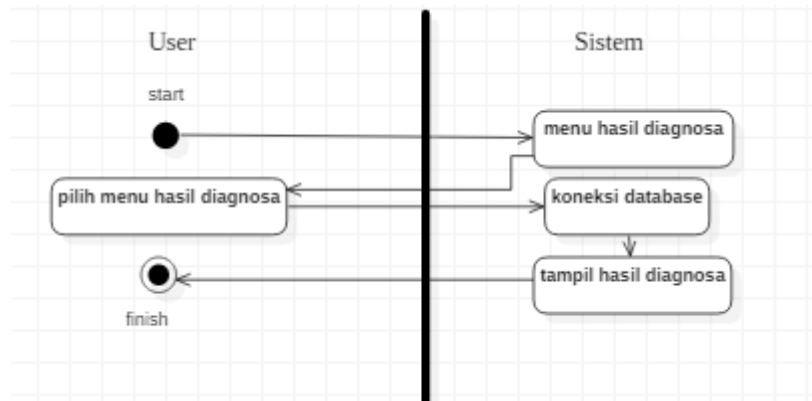
l. *Activity user diagnosa.*



Gambar 3.15 *Diagram activity user diagnosa*

Sumber : (Data Penelitian, 2017)

m. *Activity user* hasil diagnosa.

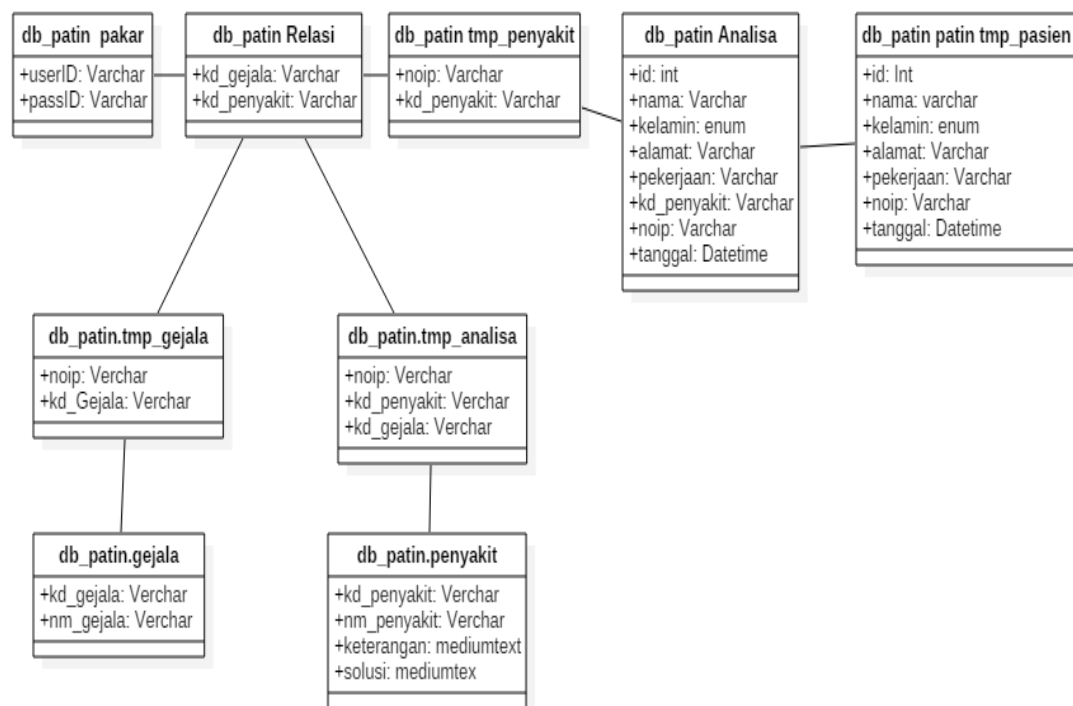


Gambar 3.16 *Diagram activity user* hasil diagnosa

Sumber : (Data Penelitian, 2017)

3.3.4 Desain Database

Dalam penelitian ini, peneliti membuat desain *database* menggunakan teknik pemodelan *Physical Data Model (PDM)* atau model relasional. Berikut ini adalah gambar model relasional yang digunakan dalam sistem pakar ini:



Gambar 3.17 Rancangan Database

Sumber : (Data Penelitian, 2017)

3.4.5. Prototype

Berikut ini adalah desain tampilan sistem pakar untuk mendiagnosa penyakit ikan Patin:

1. Menu user

a. Tampilan Menu daftar

Header	
Selamat datang di sistem pakar mendiagnosa penyakit pada ikan mas	
Nama	<input type="text"/>
Alamat	<input type="text"/>
No. Tlpn	<input type="text"/>
E_mail	<input type="text"/>
Username	<input type="text"/> (Untuk Login)
Password	<input type="text"/>
Footer	

Gambar 3.18 Tampilan menu daftar

Sumber: (Data Penelitian, 2017)

b. Tampilan User Login

Header								
Home	Info	Gejala	Penyakit	Pengguna	Keluar			
Input Pengguna								
ID User	Nama	Alamat	Telp	Email	Username	Level	Kontrol	
							Edit	Hapus
Footer								

Gambar 3.19 Tampilan user login

Sumber: (Data Penelitian, 2017)

c. Tampilan Menu Home dan Help

Header				
Home	Help	Konsultasi	Profil	Keluar
CONTENT				
Footer				

Gambar 3.20 Tampilan Menu Home dan Help**Sumber:** (Data Penelitian, 2017)

d. Tampilan Menu Pertanyaan

Header
Silahkan Jawab pertanyaan-pertanyaan berikut ini sesuai gejala yang dialami ikan
Apakah Ikan Mas Anda?
<input type="radio"/> Ya <input type="radio"/> Tidak
<input type="button" value="Simpan"/>
Footer

Gambar 3.21 Tampilan Menu Pertanyaan**Sumber:** (Data Penelitian, 2017)

e. Tampilan Menu Hasil

Header	
HASIL KEMUNGKINAN DIAGNOSA PENYAKIT IKAN PATIN	
<div style="border: 1px solid black; padding: 2px; width: fit-content; margin-bottom: 5px;">Data User</div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">CONTENT DATA USER</div>	
Hasil Analisa Terakhir Penyakit : Gejala : Keterangan : Solusi :	
<div style="border: 1px solid black; padding: 2px; width: fit-content;">Cetak</div>	
Footer	

Gambar 3.22 Tampilan Menu Hasil

Sumber: (Data Penelitian, 2017)

f. Tampilan Menu profil

Header				
Home	Help	Konsultasi	Profil	Keluar
Nama : Nama User				
Alamat : Alamat User				
No. Telpn : No. Telpn User				
E_mail : E_mail User				
Username : Nama Untuk Login				
Footer				

Gambar 3.23 Tampilan Menu Profil

Sumber: (Data Penelitian, 2017)

2. Menu Admin

a. Tampilan Menu login

Header	
Login Admin	
Username :	<input type="text"/>
Password :	<input type="password"/>
	<input type="button" value="Login"/>
Footer	

Gambar 3.24 Tampilan Menu Login

Sumber: (Data Penelitian, 2017)

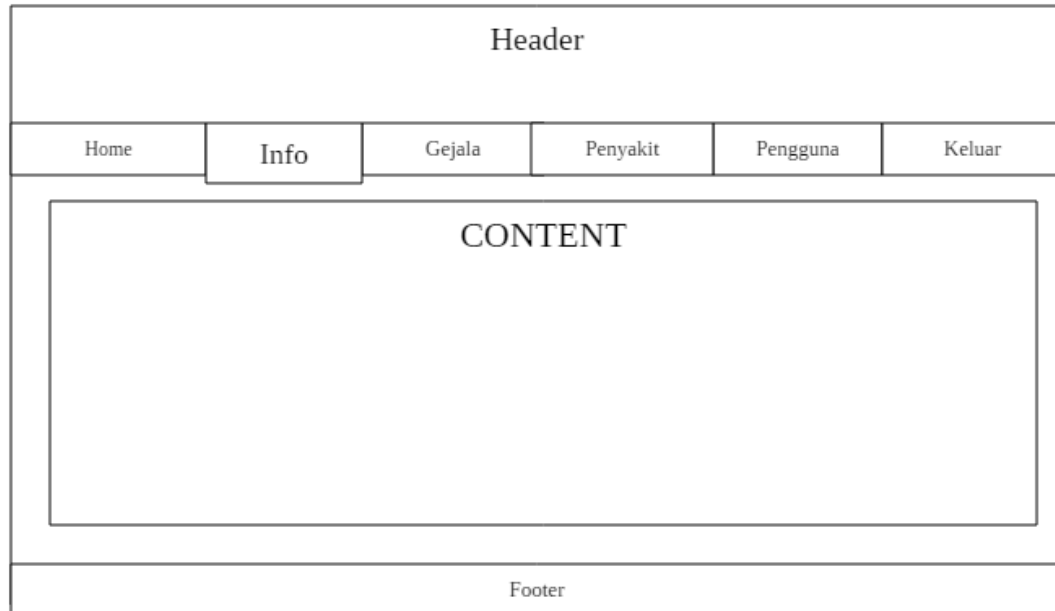
b. Tampilan Menu home

Header					
Home	Info	Gejala	Penyakit	Pengguna	Keluar
CONTENT					
Footer					

Gambar 3.25 Tampilan Menu Home

Sumber: (Data Penelitian, 2017)

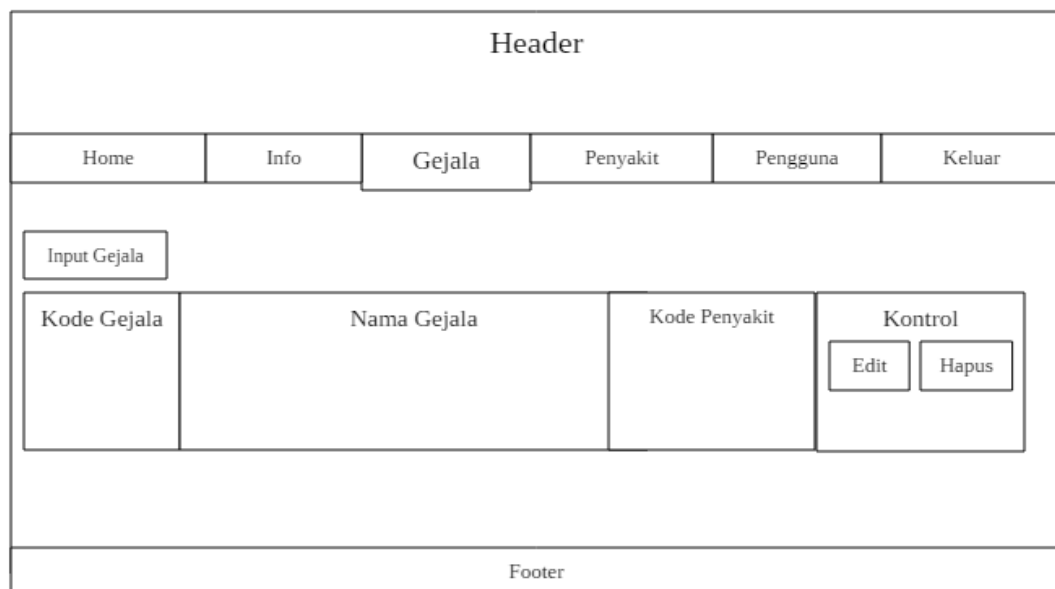
c. Tampilan Menu Info



Gambar 3.26 Tampilan Menu info

Sumber: (Data Penelitian, 2017)

d. Tampilan Menu input Gejala



Gambar 3.27 Tampilan Menu input Gejala

Sumber: (Data Penelitian, 2017)

e. Tampilan Menu Input Daftar Penyakit

Header					
Home	Info	Gejala	Penyakit	Pengguna	Keluar
Input Gejala					
Kode Penyakit	Nama Penyakit	Keterangan	Solusi	Kontrol Edit Hapus	
Footer					

Gambar 3.28 Tampilan Menu Input Daftar Penyakit

Sumber: (Data Penelitian, 2017)

f. Tampilan Admin Pengguna

Header							
Home	Info	Gejala	Penyakit	Pengguna	Keluar		
Input Pengguna							
ID User	Nama	Alamat	Telp	Email	Username	Level	Kontrol Edit Hapus
Footer							

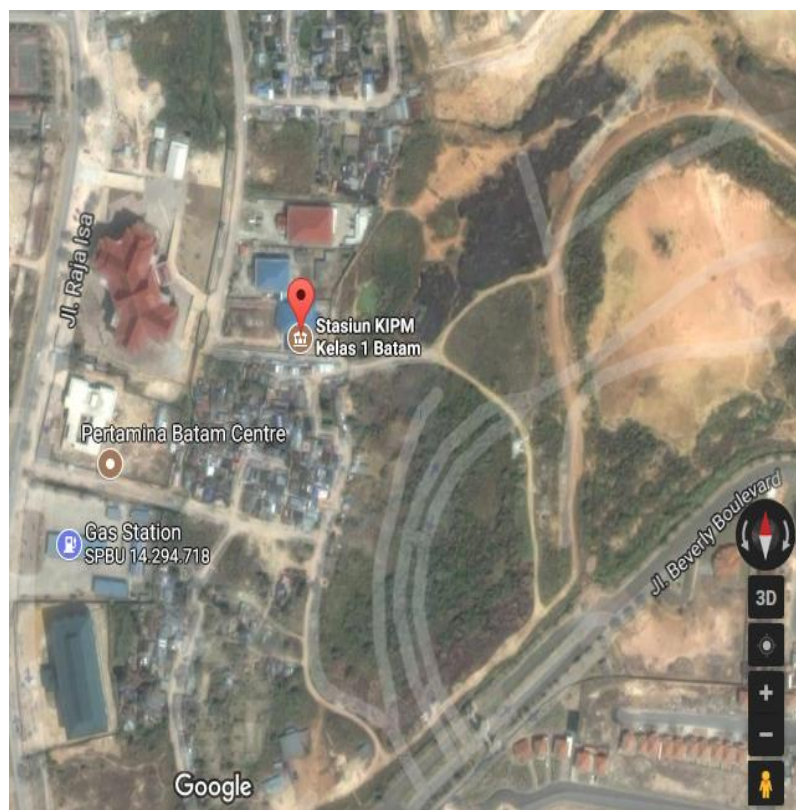
Gambar 3.29 Tampilan Admin Pengguna

Sumber: (Data Penelitian, 2017)

3.5. Lokasi dan Jadwal penelitian

3.5.1. Lokasi

Lokasi penelitian sebagai tempat pengambilan data dilakukan di Stasiun Karantina Ikan, Pengendalian Mutu dan Keamanan Hasil Perikanan Kelas 1 Batam yang berlokasi di JL. M. Nahar. Batam Center. Kota Batam.



Gambar 3.30 Alamat Karentina dari Google Map

Sumber: (Data Penelitian, 2017)

3.5.2. Jadwal Penelitian

Setiap rancangan penelitian perlu dilengkapi dengan jadwal penelitian yang dilaksanakan yang berisi jadwal kegiatan apa saja yang akan dilakukan selama penelitian (Sugiyono, 2014: 286). Berikut ini adalah tabel jadwal kegiatan yang dilakukan selama penelitian berlangsung.

Tabel 3.9 Jadwal Penelitian

NO	KEGIATAN	Tahun 2017																					
		Mar'2016				Apr'2016				Mei'2016				Jun'2016				Jul'2017				Agt'2017	
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2
1	Pengajuan Judul Penelitian	■	■	■																			
2	Pengajuan Surat Izin Penelitian				■	■	■																
3	Penyusunan Bab I				■	■	■	■															
4	Penyusunan Bab II								■	■	■	■											
5	Penyusunan Bab III													■	■	■	■						
6	Penyusunan Bab IV																	■	■	■	■		
7	Penyusunan Bab V, Daftar Pustaka, Lampiran																				■	■	

Sumber: Data Penelitian, 2017)