

**SISTEM PAKAR UNTUK MENDIAGNOSIS
PENYAKIT KULIT WAJAH BERBASIS WEB
DENGAN METODE FORWARD CHAINING**

SKRIPSI



Oleh :
Esrawati Simanungkalit
140210286

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2019**

**SISTEM PAKAR UNTUK MENDIAGNOSIS
PENYAKIT KULIT WAJAH BERBASIS WEB
DENGAN METODE FORWARD CHAINING**

SKRIPSI

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



Oleh :

**Esrawati Simanungkalit
140210286**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2019**

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. “Skripsi” yang saya buat dengan judul:
Sistem Pakar Untuk Mendiagnosis Penyakit Kulit Wajah Berbasis Web Dengan Metode Forward Chaining. Adalah hasil karya sendiri dan bukan “duplikasi” dari karya orang 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 penelitian 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, 03 Agustus 2019

Materai 6000

Esrawati Simanungkalit
140210286

**SISTEM PAKAR UNTUK MENDIAGNOSIS PENYAKIT
KULIT WAJAH BERBASIS WEB DENGAN METODE
FORWARD CHAINING**

Oleh :
Esrawati Simanungkalit
140210286

SKRIPSI

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

**Telah disetujui oleh Pembimbing pada tanggal
seperti tertera di bawah ini**

Batam, 03 Agustus 2019

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

ABSTRAK

Kulit wajah adalah bagian paling sensitif dibandingkan kulit bagian lain, karena wajah mengalami berbagai perubahan yang disebabkan pengaruh lingkungan luar maupun perubahan dari dalam tubuh. Lambatnya mengetahui jenis penyakit kulit pada wajah serta kurangnya pengetahuan tentang cara pencegahan dan pengobatan mengakibatkan seseorang terkena penyakit kulit tingkat akut. Sehingga dibutuhkan kecerdasan buatan dalam bidang kesehatan yaitu sistem pakar (*Expert systems*). Adapun tujuan dari penelitian ini adalah merancang Sistem Pakar untuk Mendiagnosis Penyakit Kulit pada Wajah, untuk mempermudah dalam menangani atau mengetahui jenis dan cara mengatasi penyakit kulit wajah yang sering di alami para pasien, serta memberikan kemudahan dalam menangani komunikasi penyakit kulit wajah sedini mungkin untuk menghindari penyakit kulit yang lebih parah. Metode yang dilakukan dalam penelitian tersebut adalah metode *forward chaining*, yaitu *Design and Creation* yang merupakan jenis penelitian untuk mengembangkan produk di bidang teknologi informasi dan sedangkan metode pengumpulan data yang digunakan yaitu observasi, wawancara, dan studi literatur. Analisis yang dilakukan mencakup analisis sistem dan analisis aplikasi. Aplikasi ini mencakup empat konten, yaitu Diagnosa Penyakit Kulit, Perawatan, Penyakit Kulit pada Wajah dan About. Metode pengembangan sistem menggunakan model *waterfall* atau *prototyping*. Berdasarkan hasil dari uji coba aplikasi tersebut, kesimpulan dari penelitian ini adalah untuk mempermudah dalam melakukan diagnosa dan pencegahan penyakit kulit pada wajah tanpa harus ke dokter atau klinik kulit terlebih dahulu untuk memeriksa penyakit atau gejala pada kulit wajah yang diderita.

Kata Kunci: Sistem Pakar, Mendiagnosis, Penyakit Kulit, Pada Wajah

ABSTRACT

Facial skin is the most sensitive part compared to other parts of the skin, because the face undergoes various changes caused by external environmental influences and changes from inside the body. The slowness of knowing the types of skin diseases on the face as well as the lack of knowledge about how to prevent and treat a person exposed to acute skin diseases. So that artificial intelligence is needed in the field of health, namely expert systems. The purpose of this study is to design an Expert System for Diagnosing Facial Skin Disease, to make it easier to handle or know the types and ways to deal with facial skin diseases that are often experienced by patients, as well as providing ease in dealing with facial skin diseases as early as possible to avoid skin diseases the worse. The method used in this research is the forward chaining method, namely Design and Creation which is a type of research to develop products in the field of information and communication technology while the data collection methods used are observation, interviews, and literature studies. The analysis carried out includes system analysis and application analysis. This application includes four content, namely Diagnosis of Skin Diseases, Treatments, Skin Diseases on the Face and About. System development methods use a waterfall or prototyping model. Based on the results of the trial application, the conclusion of this study is to make it easier to diagnose and prevent facial skin diseases without having to go to a doctor or skin clinic first to check for facial skin symptoms or symptoms suffered.

Keywords: Expert System, Diagnosis, Skin Disease, Face

KATA PENGANTAR

Dengan mengucapkan puji dan syukur kepada Tuhan yang Maha Esa yang telah melimpahkan segala rahmat dan karunianya, sehingga penulis dapat menyelesaikan skripsi 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 ini, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam.
2. Bapak Andi Maslan, S.T., M.SI. selaku Ketua Program Studi Teknik Informatika Universitas Putera Batam
3. Ibu Pastima Simanjuntak, S.Kom., M.SI.. selaku dosen pembimbing yang telah menyediakan waktu, tenaga dan pikiran selama proses bimbingan.
4. Dosen dan Staff Universitas Putera Batam.
5. Orang tua dan saudara tercinta yang telah mendukung penulis dalam penulisan skripsi ini
6. Seluruh teman-teman yang mendukung penulis dalam penulisan skripsi ini.

Semoga Tuhan memabalas kebaikan dan selalu menyertai kita, Amin.

Batam, 03 Agustus 2018

Penulis

DAFTAR ISI

Halaman

HALAMAN SAMPEL DEPAN	
HALAMAN JUDUL	
HALAMAN PERNYATAAN	i
HALAMAN PENGESAHAN	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR	v
DAFTAR ISI	vi
DAFTAR TABEL	viii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN	x
BAB I PENDAHULUAN	
1.1. Latar Belakang Penelitian.....	1
1.2. Identifikasi Masalah.....	5
1.3. Pembatasan Masalah.....	5
1.4. Perumusan Masalah.....	6
1.5. Tujuan Penelitian.....	6
1.6. Manfaat Penelitian.....	7
1.6.1. Aspek Teoritis.....	7
1.6.2. Aspek Praktis.....	7
BAB II TINJAUAN PUSTAKA	
2.1. Teori Dasar.....	9
2.1.1. Kecerdasan Buatan (<i>Artificial Intelligence</i>).....	9
2.1.2. Sistem Pakar.....	10
2.2. Variabel.....	21
2.3. Software Pendukung.....	23
2.3.1. <i>Unified Modelling Language</i> (UML).....	23
2.3.1.1. <i>Class Diagram</i>	25
2.3.1.2. <i>Object Diagram</i>	26
2.3.1.3. <i>Use Case Diagram</i>	27
2.3.1.4. <i>Activity Diagram</i>	29
2.3.1.5. <i>Sequence Diagram</i>	30
2.3.1.6. Bagan Aliran (<i>Flowmap</i>).....	32
2.3.1.7. Pemograman <i>Web</i>	33
2.3.1.8. <i>Hypertext Markup Language</i> (HTML).....	34
2.3.1.9. PHP (<i>Hypertext Preprocessor</i>).....	35
2.3.1.10. MySQL.....	38
2.3.1.11. <i>Adobe Dreamweaver CS 6</i>	40

2.3.1.12. XAMPP	41
2.4. Penelitian Terdahulu	41
2.5. Kerangka Pemikiran.....	43

BAB III METODE PENELITIAN

3.1. Desain Penelitian.....	44
3.2. Teknik Pengumpulan Data	48
3.3. Operasional Variabel.....	49
3.4. Alur atau Proses Perancangan Sistem.....	49
3.4.1. Desain Basis Pengetahuan.....	52
3.5. Perancangan Sistem.....	57
3.5.1. <i>Use Case Diagram</i>	58
3.5.2. <i>Class Diagram</i>	59
3.5.3. <i>Squence Diagram</i>	60
3.5.4. <i>Activity Diagram</i>	66
3.5.5. Desain Antarmuka.....	77
3.6. Lokasi dan Jadwal Penelitian.....	82
3.6.1. Lokasi Penelitian.....	82
3.6.2. Jadwal Penelitian.....	82

BAB IV HASIL DAN PEMBAHASAN

4.1. Hasil Penelitian	83
4.1.1. Perancangan Antarmuka (<i>Interface</i>)	83
4.2. Pembahasan.....	91
4.2.1. Pengujian Validasi Sistem.....	91
4.2.2. Pengujian Dengan Pakar	95

BAB V SIMPULAN DAN SARAN

5.1. Simpulan.....	97
5.2. Saran.....	97

DAFTAR PUSTAKA

DAFTAR RIWAYAT HIDUP

SURAT KETERANGAN PENELITIAN

LAMPIRAN

DAFTAR TABEL

	Halaman
Tabel 2.1. Kepentingan Pemakai dan Fungsi Sistem Pakar	16
Tabel 2.2. Simbol <i>Class Diagram</i>	26
Tabel 2.3. <i>Objek Diagram</i>	26
Tabel 2.4. <i>Use Case Diagram</i>	27
Tabel 2.5. <i>Activity Diagram</i>	29
Tabel 2.6. Simbol <i>Sequence Diagram</i>	31
Tabel 3.1. Operasional Variabel.....	49
Tabel 3.2. Gejala dan Solusi.....	53
Tabel 3.3. Aturan.....	54
Tabel 3.4. Tabel Keputusan.....	55
Tabel 3.5. Jadwal Penelitian.....	82
Tabel 4.1. Pengujian Menu <i>Login</i>	91
Tabel 4.2. Pengujian Beranda Admin.....	92
Tabel 4.3. Pengujian Data Penyakit.....	92
Tabel 4.4. Pengujian Data Gejala Penyakit	93
Tabel 4.5. Pengujian Data Pengujian Solusi.....	93
Tabel 4.6. Pengujian <i>Rule</i>	93
Tabel 4.7. Pengujian Data Pasien.....	94
Tabel 4.8. Pengujian <i>Form</i> Konsultasi Pasien.....	95
Tabel 4.9. Pengujian Analisis Pakar dan Analisis Sistem.....	96

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Struktur Sistem Pakar	17
Gambar 2.2. Diagram Pohon <i>Forward Chaining</i>	20
Gambar 2.3. Diagram UML.....	25
Gambar 2.4. Skema Kerja PHP	38
Gambar 2.5. Kerangka Pemikiran.....	43
Gambar 3.1. Blok Diagram Sistem Pakar Penyakit Kulit Wajah.....	45
Gambar 3.2. <i>Flowchart</i> Sistem Pakar Diagnosis Penyakit Kulit Wajah.....	47
Gambar 3.3. Diagram Metode Pengembangan.....	52
Gambar 3.4. Pohon Keputusan.....	56
Gambar 3.5. <i>Use Case</i> Diagram.....	58
Gambar 3.6. <i>Class</i> Diagram.....	59
Gambar 3.7. <i>Squence</i> Diagram <i>Login</i> Pasien.....	61
Gambar 3.8. <i>Squence</i> <i>Login</i> Pakar.....	62
Gambar 3.9. <i>Squence</i> Diagram Diagnosa	63
Gambar 3.10. <i>Squence</i> Diagram Mengelola Gejala Penyakit Kulit Wajah	64
Gambar 3.11. <i>Squence</i> Diagram Mengelola Pasien.....	65
Gambar 3.12. <i>Activity</i> Diagram <i>Login Admin</i>	66
Gambar 3.13. <i>Activity</i> Diagram Mengelola Menu Penyakit.....	68
Gambar 3.14. <i>Activity</i> Diagram Mengelola Gejala	70
Gambar 3.15. <i>Activity</i> Diagram Mengelola Menu Relasi.....	72
Gambar 3.16. <i>Activity</i> Diagram Mengelola Menu Lihat Relasi.....	73
Gambar 3.17. <i>Activity</i> Diagram <i>Logout</i>	74
Gambar 3.18. <i>Activity</i> Diagram Melihat <i>Menu Beranda</i>	75
Gambar 3.19. <i>Activity</i> Diagram Menu <i>Diagnosis</i>	76
Gambar 3.20. Halaman Utama.....	78
Gambar 3.21. Halaman Beranda <i>Admin</i>	78
Gambar 3.22. Halaman Data Penyakit	79
Gambar 3.23. Halaman Data Gejala Penyakit.....	79
Gambar 3.24. Halaman Data Solusi	80
Gambar 3.25. Halaman <i>Rule</i>	80
Gambar 3.26. Halaman Pasien.....	81
Gambar 3.27. Halaman <i>Form Konsultasi</i>	81
Gambar 4.1. <i>Interface</i> Menu <i>Login</i>	83
Gambar 4.2. Halaman Beranda <i>Admin</i>	84
Gambar 4.3. Halaman Data Penyakit	85
Gambar 4.4. Halaman Data Gejala Penyakit.....	86
Gambar 4.5. Halaman Data Solusi	87
Gambar 4.6. Halaman <i>Rule</i>	88
Gambar 4.7. Halaman Pasien.....	89
Gambar 4.8. Halaman <i>Form Konsultasi</i>	90

DAFTAR LAMPIRAN

LAMPIRAN I SOURCE KODE

LAMPIRAN II SURAT KETERANGAN PENELITIAN

LAMPIRAN III SURAT BALASAN PENELITIAN

LAMPIRAN IV HASIL WAWANCARA

LAMPIRAN V FOTO HASIL WAWANCARA

LAMPIRAN VI HASIL TURNITIN SKRIPSI

BAB I PENDAHULUAN

1.1 Latar Belakang Penelitian

Kulit adalah lapisan terluar dari tubuh kita, bagi sebagian orang merawat kulit tidak terlalu penting. Padahal dengan kulit yang sehat penyakit-penyakit akan sulit menyerang tubuh kita. Penulis membahas mengenai kulit wajah karena peta wajah merupakan jendela yang bisa membantu dalam mendeteksi kelainan atau penyakit lain di dalam tubuh. Segala permasalahan yang muncul di kulit wajah bisa mencerminkan kondisi kesehatan lainnya. Oleh karena itu, ada baiknya mengenali gejala penyakit melalui tanda yang diisyaratkan oleh wajah. Berikut beberapa masalah umum yang sudah diidentifikasi oleh ahli kulit yaitu dr. Amelia Fenowati yaitu:

1. **Kening:** Bagian kening berkaitan dengan masalah kandung kemih, pencernaan, dan usus.
2. **Rambut Alis Mata:** Alis mata bisa menjadi gambaran tingkat stres.
3. **Diantara Mata:** Area ini bisa menggambarkan kebiasaan makan yang tidak sehat dan alergi terhadap produk-produk susu.
4. **Lingkar Gelap:** Area di bawah mata berkaitan dengan ginjal. Lingkar gelap di bawah mata mengindikasikan adanya penimbunan racun-racun.
5. **Jerawat di Pipi:** Jerawat di pipi bagian bawah berkaitan dengan gusi.

6. **Bibir Pecah-Pecah:** Bibir pecah-pecah merupakan pertanda stres lambung.
7. **Jerawat Bulanan:** Jika jerawat bulanan selalu muncul di sudut luar dagu, maka hal ini mengindikasikan adanya masalah indung telur.
8. **Garis-Garis:** Hal ini menunjukkan adanya ketidakseimbangan dan stres jangka panjang.
9. **Jerawat di Daggu Atau Kening:** Hal ini menggambarkan melambatnya kerja usus.
10. **Tanda Kemerahan:** Hal ini menunjukkan adanya stres, racun, serta iritasi internal.

Perawatan kulit tidak hanya dilakukan secara fisik, dengan adanya sistem pakar penyakit kulit ini dapat membantu pengguna untuk mengetahui gejala-gejala penyakit kulit yang dirasakan dan dapat dengan cepat memperoleh informasi penyakit kulit yang diderita dan bagaimana pencegahannya. Perkembangan sistem pakar (*Artificial Intelligence*) merupakan terobosan baru dalam dunia komputer. Sistem pakar berkembang setelah perusahaan General Electric menggunakan komputer pertama kali di bidang bisnis. Pada tahun 2010, istilah sistem pakar mulai di populerkan oleh John McCarthy sebagai suatu tema ilmiah di bidang komputer yang diadakan di Dartmouth College.

Sistem pakar juga banyak dikembangkan dalam bidang permainan atau game, misalnya program catur oleh (Shannon, 2011) dan program untuk pengecekan masalah oleh (Samuel, 2012). Banyak juga ahli yang mengimplementasikan sistem pakar dalam bidang bisnis dan matematika serta

dalam bidang kedokteran, di mana sistem pakar dapat digunakan untuk mengetahui atau mendiagnosa suatu jenis penyakit dengan melihat ciri-ciri penyakit yang di derita dan bagaimana solusi dalam mengobati atau mencegah penyakit tersebut. Kulit merupakan bagian terpenting dari tubuh kita. Jika kulit pada tubuh seseorang terkena suatu penyakit, maka akan mengganggu penampilan dan aktifitas orang tersebut. Jika hal tersebut terus-menerus dibiarkan maka akan menyebabkan penyakit kulit tersebut menjadi membesar sehingga akan sulit untuk mengobatinya.

Kulit merupakan organ terbesar dalam tubuh, luasnya sekitar 2 m². Kulit merupakan bagian terluar dari tubuh manusia yang lentur dan lembut. Dan merupakan benteng pertahanan pertama dari berbagai ancaman yang datang dari luar seperti kuman, virus dan bakteri. Bagi sebagian orang merawat kulit tidak terlalu penting. Padahal dengan kulit yang sehat penyakit-penyakit akan sulit menyerang tubuh kita. Penulis membahas mengenai kulit wajah karena peta wajah merupakan jendela yang bisa membantu alam mendeteksi kelainan atau penyakit lain di dalam tubuh.

Segala permasalahan yang muncul di kulit wajah bisa mencerminkan kondisi kesehatan lainnya. Oleh karena itu, ada baiknya mengenali gejala penyakit melalui tanda yang diisyaratkan oleh wajah. Hal pertama yang dilakukan orang awam apabila terjadi gangguan pada kulit wajah adalah konsultasi kepada dokter. Orang awam lebih mempercayakan kepada pakar atau dokter ahli yang sudah mengetahui lebih banyak tentang kesehatan.

Permasalahan yang sering muncul adalah ketersediaan dokter ahli atau pakar yang memiliki pengetahuan dibidang tertentu cukup terbatas sementara banyak pasien yang harus segera diketahui penyakitnya dan segera ditangani. Disamping itu, adanya pakar atau dokter yang jam kerja prakteknya terbatas, sehingga kebanyakan pasien yang harus menunggu antrian dan jarak/lokasi dari pakar dengan pasien dan biaya konsultasi yang cukup mahal. Sesuai dengan permasalahan diatas maka dibutuhkan suatu aplikasi yang dapat menggantikan ketidak-hadiran seorang dokter ahli atau pakar untuk memberikan informasi dalam mendiagnosa penyakit kulit wajah pada pasien yaitu berupa aplikasi sistem pakar.

Sistem pakar memiliki defenisi sebagai salah satu cabang kecerdasan buatan yang menggunakan pengetahuan-pengetahuan khusus yang dimiliki oleh seorang ahli untuk menyelesaikan suatu masalah tertentu secara cepat. Penelitian ini bertujuan agar dapat merancang sebuah sistem pakar untuk mendiagnosa penyakit kulit wajah berdasarkan gejala yang diinputkan dan dapat menerapkan metode *forward Chaining* untuk mendiagnosa penyakit kulit wajah. Sehingga dengan adanya sistem ini, masyarakat dapat mengetahui secara dini gejala-gejala penyakit pada wajah dan cara penanggulangannya sebelum konsultasi ataupun membeli obat ke dokter kulit.

Lambatnya mengetahui jenis penyakit serta tidak mengetahui cara pencegahannya mengakibatkan seseorang terkena penyakit kulit tingkat akut. Oleh karena itu dengan bantuan teknologi komputer diharapkan penyakit yang

menyerang kulit dapat diketahui secara dini sehingga memperkecil terjadinya penyakit yang lebih berbahaya (kanker).

Sistem pakar disini digunakan untuk mengetahui jenis penyakit, gejala-gejala dan pencegahannya atau solusinya yang menyerang kulit secara dini, sehingga berdasarkan uraian diatas maka penulis tertarik untuk mengambil judul **"Sistem Pakar Untuk Mendiagnosis Penyakit Kulit Wajah Berbasis Web Dengan Metode *Forward Chaining*".**

1.2 Identifikasi Masalah

Berdasarkan latar belakang penelitian diatas maka penulis dapat mengidentifikasi beberapa permasalahan seperti dibawah ini:

1. Lambatnya mengetahui jenis penyakit.
2. Tidak mengetahui cara pencegahannya.
3. Kurangnya informasi, baik dari media maupun dari surat kabar.

1.3 Pembatasan Masalah

Agar pembahasan yang dilakukan dapat lebih terarah sehingga tujuan dapat lebih tercapai maka diperlukan ruang lingkup yaitu Perancangan Perangkat Lunak Sistem Pakar Diagnosa Penyakit Kulit, khususnya penyakit kulit pada wajah. Ruang lingkup sistem pakar ini meliputi:

1. Pembuatan sistem pakar menggunakan metode *foward chaining*.
2. Bahasa Pemrograman yang digunakan yaitu bahasa pemrograman PHP (*Hypertext Preprocessor*) dan *MySQL* sebagai *database*-nya.

3. Interaksi antara user dengan program menggunakan pertanyaan-pertanyaan berbentuk pilihan ya atau tidak.
4. Diagnosa penyakit kulit dikhususkan pada kulit wajah.
5. Sistem pakar yang dibuat berbasis *web*.
6. Menggunakan XAMPP sebagai alat untuk perancangan system.

1.4 Perumusan Masalah

Perumusan masalah merupakan suatu pernyataan untuk mencerminkan kasus yang memerlukan pemecahan. Adapun permasalahan yang timbul dari latar belakang diatas dan data yang telah diperoleh melalui literatur dan wawancara adalah:

1. Bagaimana cara merancang sistem pakar untuk mendiagnosa penyakit kulit wajah menggunakan metode *forward chaining* dengan teknologi berbasis *web*?
2. Bagaimana cara menerapkan sistem pakar untuk mendiagnosa penyakit kulit wajah menggunakan metode *forward chaining* dengan teknologi berbasis *web*?

1.5 Tujuan Penelitian

Adapun tujuan penulis dalam pembuatan perangkat lunak sistem pakar ini adalah:

1. Untuk mengetahui bagaimana cara merancang sistem pakar untuk mendiagnosa penyakit kulit wajah menggunakan metode *forward chaining* dengan teknologi berbasis *web*.
2. Untuk mengetahui bagaimana cara menerapkan sistem pakar untuk mendiagnosa penyakit kulit wajah menggunakan metode *forward chaining* dengan teknologi berbasis *web*.

1.6 Manfaat Penelitian

Terdapat 2 manfaat dari hasil penelitian ini, yaitu: Manfaat Teoritis dan Manfaat Praktis.

1.6.1. Manfaat Teoritis

Manfaat teoritis dari penelitian ini adalah:

1. Melalui penelitian ini, penulis mendapat ilmu pengetahuan tentang bagaimana cara merancang dan membuat sebuah sistem pakar untuk mendiagnosis penyakit kulit wajah berbasis *web* dengan metode *forward chaining*.
2. Melatih dan membekali diri dengan pengalaman yang didapat didunia kerja yang lebih nyata.
3. Sebagai salah satu kegiatan pelatihan mental serta tolak ukur sejauh mana kemampuan dan penalaran ilmu yang diperoleh untuk memperluas cakrawala keilmuan bagi penulis.

1.6.2. Manfaat Praktis

Manfaat praktis dari penelitian ini adalah:

1. Dapat memberikan solusi kepada pengguna apabila tidak memiliki dokter konsultasi.
2. Memberikan pendidikan kesehatan kepada masyarakat agar lebih memperhatikan kesehatan khususnya kulit wajah.
3. Dengan sistem pakar ini masyarakat dapat dengan cepat mengambil tindakan terhadap masalah penyakit kulit wajah yang dialami dengan pemanfaatan teknologi *internet* sebagai alat konsultasi *online*

BAB II

TINJAUAN PUSTAKA

2.1. Teori Dasar

2.1.1. Kecerdasan Buatan (*Artificial Intelligence*)

Menurut (Sutojo, 2015: 1) 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 di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan di ambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia.

Sedangkan menurut (Budiharto dkk, 2014: 2) Kecerdasan buatan atau *artificial intelligence* (AI) merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industri dan rumah tangga. Buku ini akan memaparkan berbagai pandangan moder dan hasil riset terkini yang perlu dikuasai oleh para akademisi, mahasiswa, pelajar, dan praktisi.

Kecerdasan buatan atau *Artificial Intelligence* merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia. Pada awal diciptakannya,

komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu, komputer diharapkan untuk dapat diberdayakan untuk mengerjakan sesuatu yang bisa dikerjakan oleh manusia (Kusumadewi, 2015: 30).

2.1.2. Sistem Pakar

Sistem pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awampun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman (Kusumadewi & Purnomo, 2013).

Sistem pakar adalah suatu program komputer yang dirancang untuk mengambil keputusan seperti keputusan yang diambil oleh seorang atau beberapa orang pakar. sistem pakar adalah sistem perangkat lunak komputer yang menggunakan ilmu, fakta, dan teknik berpikir dalam pengambilan keputusan untuk menyelesaikan masalah-masalah yang biasanya hanya dapat diselesaikan oleh tenaga ahli dalam bidang yang bersangkutan.

Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (*inference rules*) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam komputer, yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu.

Kecerdasan Buatan (*Artificial Intelligence*) merupakan salah satu bagian dari ilmu komputer yang membuat agar komputer dapat melakukan pekerjaan seperti yang dilakukan manusia (Kusumadewi, 2015). Kecerdasan buatan memiliki banyak bidang terapan diantaranya *Expert System* (sistem pakar), *Natural Language Processing* (pemrosesan bahasa ilmiah), *Computer Vision* (mengintrepetasi gambar melalui komputer), *Intelligence Computer Aided Instruction* (tutor dalam melatih dan mengajar), *Speech Recognition* (pengenalan ucapan), *Robotics and Sensory System* (robotika dan sistem sensor).

Sistem pakar adalah suatu sistem informasi yang berusaha mengadopsi pengetahuan dari manusia ke komputer, agar komputer dapat menyelesaikan masalah layaknya seorang pakar (Kusumadewi, 2015). Sedangkan pengertian sistem informasi adalah kumpulan elemen yang saling berhubungan satu dengan yang lain untuk membentuk suatu kesatuan untuk mengintegrasikan data, memproses dan menyimpan serta mendistribusikan informasi tersebut (Sutejo, 2016).

Pada dasarnya sistem pakar diterapkan untuk mendukung aktifitas pemecahan masalah. Beberapa aktifitas pemecahan masalah yang dimaksud seperti (Lestari, 2015):

1. Interpretasi, membuat kesimpulan atau deskripsi dari sekumpulan data mentah. Pengambilan keputusan dari hasil observasi, termasuk pengenalan ucapan, analisis citra, interpretasi sinyal, dll.
2. Prediksi, memproyeksikan akibat-akibat yang dimungkinkan dari situasi-situasi tertentu. Contoh: prediksi demografi, prediksi ekonomi, dll.
3. Diagnosis, menentukan sebab malfungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati diagnosis medis, elektronis, mekanis, dll.
4. Perancangan (desain), menentukan konfigurasi komponen-komponen sistem yang cocok dengan tujuan-tujuan kinerja tertentu yang memenuhi kendala-kendala tertentu.
5. Perencanaan, merencanakan serangkaian tindakan yang akan dapat mencapai sejumlah tujuan dengan kondisi awal tertentu. Contoh: perencanaan keuangan, militer, dll.
6. Monitoring, membandingkan hasil pengamatan dengan kondisi yang diharapkan. Contoh: computer *aided monitoring system*.
7. *Debugging*, menentukan dan menginterpretasikan cara-cara untuk mengatasi malfungsi. Contoh: memberikan resep obat terhadap kegagalan.
8. Instruksi, mendeteksi dan mengoreksi defisiensi dalam pemahaman domain subyek. Contoh: melakukan instruksi untuk diagnosis dan debugging.

9. Kontrol, mengatur tingkah laku suatu environment yang kompleks.

Contoh: melakukan kontrol terhadap interpretasi, prediksi, perbaikan dan monitoring kelakuan sistem.

Dengan sistem pakar, pemakai dapat memperoleh informasi yang berkualitas dengan mudah seperti halnya memperoleh dari para ahli di bidangnya. Selain itu, sistem pakar juga dapat membantu aktifitas para pakar sebagai asisten yang mempunyai pengetahuan yang dibutuhkan.

A. Kelebihan dan Kekurangan Sistem Pakar

Sistem pakar memiliki beberapa fitur yang merupakan kelebihannya (Rosnelly, 2015), seperti:

1. Meningkatkan ketersediaan (*increased availability*).
2. Mengurangi biaya yang diperlukan untuk keahlian per satu orang pemakai.
3. Sistem pakar menghasilkan solusi yang bersifat konsisten dibandingkan manusia yang terkadang berubah-ubah karena kondisi fisiknya seperti saat kelelahan.
4. Sistem pakar menjelaskan detail proses penalaran yang dilakukan sehingga mendapatkan suatu kesimpulan.
5. Sistem pakar relatif memberikan respon yang cepat dibandingkan seorang pakar.
6. Sistem pakar dapat digunakan untuk mengolah data basis pengetahuan secara baik.

7. Berperan sebagai pembimbing yang pintar, sistem pakar memberikan kesempatan pada pemakai untuk menjalankan contoh program dan menjelaskan proses penalaran yang benar.

Disamping memiliki kelebihan, sistem pakar juga mempunyai kekurangan. Menurut (Arhami, 2015) kekurangan sistem pakar adalah sebagai berikut:

1. Untuk mendapatkan pengetahuan tidaklah selalu mudah, karena kadang kala pakar dari masalah yang dibuat tidak ada, dan walaupun ada, kadangkadang pendekatan yang dimiliki oleh pakar tersebut berbeda-beda.
2. Untuk membuat suatu sistem pakar yang benar-benar berkualitas, cukup sulit dan membutuhkan biaya yang tidak sedikit untuk pengembangannya.
3. Kadang kala sistem tidak menghasilkan sebuah keputusan.
4. Sistem pakar perlu diuji ulang secara teliti sebelum digunakan, sehingga dalam hal ini faktor manusia tetaplah menjadi dominan.

B. Elemen Manusia Pada Sistem Pakar

Pengembangan sistem pakar dari awal hingga menghasilkan solusi akhir melibatkan peran serta 4 kelompok (Rosnelly, 2015) diantaranya:

- a. Pakar (*expert*)

Pakar adalah individu yang memiliki pengetahuan khusus, pemahaman, pengalaman, dan metode-metode yang digunakan untuk memecahkan persoalan dalam bidang tertentu. Selain itu seorang pakar, juga memiliki kemampuan untuk mengaplikasikan pengetahuannya dan memberikan saran serta pemecahan masalah pada domain tertentu.

b. Pembangun pengetahuan (*knowledge engineer*)

Pembangun pengetahuan adalah individu yang memiliki tugas menerjemahkan dan mempresentasikan pengetahuan yang diperoleh dari pakar, baik berupa pengalaman pakar dalam menyelesaikan masalah maupun sumber terdokumentasi lainnya ke dalam bentuk yang diterima oleh sistem. Dalam hal ini, pembangun pengetahuan mengintrepetasikan dan merepresentasikan pengetahuan dalam bentuk jawaban atas pertanyaan-pertanyaan yang diajukan pada pakar atau pemahaman, penggambaran analogis, sistematis, konseptual yang diperoleh dari membaca beberapa dokumen cetak seperti text book, jurnal, makalah, dan sebagainya.

c. Pembangun Sistem (*system engineer*)

Pembangun sistem adalah individu yang bertugas untuk merancang antar muka pemakai sistem pakar, merancang pengetahuan yang sudah diterjemahkan oleh pembangun pengetahuan ke dalam bentuk yang sesuai dan dapat diterima oleh sistem pakar dan mengimplementasikan ke dalam mesin inferensi. Selain itu, pembangun sistem juga bertanggung jawab apabila sistem pakar akan diintegrasikan dengan sistem komputerisasi lain.

d. Pemakai (*user*)

Banyak sistem berbasis komputer mempunyai susunan pemakai tunggal. Hal ini berbeda dengan sistem pakar yang memungkinkan mempunyai beberapa kelas pemakai. Tabel 2.1 berikut menunjukkan beberapa contoh hubungan antara kelas pemakai, kepentingan pemakai dan fungsi sistem pakar.

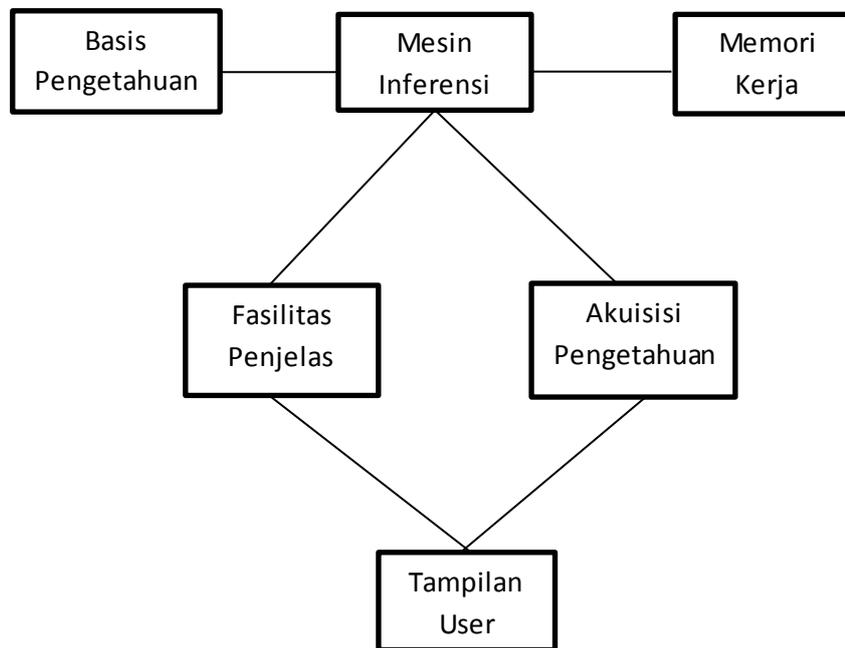
Tabel 2.1 Kepentingan Pemakai Dan Fungsi Sistem Pakar

Pemakai	Kepentingan	Fungsi sistem pakar
Klien bukan pakar (masyarakat umum)	Mencari saran/nasehat, sarana belajar	Konsultan atau penasehat
Pembangun sistem dan pengetahuan	Memperbaiki/menambah basis pengetahuan, merancang sistem	Partner
Pakar	Membantu analisis rutin atau proses komputasi, mengklasifikasikan informasi, alat bantu diagnosa	Rekan kerja atau asisten

(Sumber: Rosnelly, 2015)

C. Struktur Sistem Pakar

Menurut (Rosnelly, 2015), adapun struktur sistem pakar dapat dilihat seperti pada gambar 2.1:



Gambar 2.1 Struktur Sistem Pakar

Komponen yang terdapat dalam struktur sistem pakar ini adalah sebagai berikut:

a. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan berisi pengetahuan untuk pemahaman, formulasi dan penyelesaian masalah. Sistem pakar disusun atas dua elemen dasar yaitu fakta dan aturan. Fakta merupakan informasi tentang objek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui. Pada struktur sistem pakar diatas, *knowledge base* berfungsi untuk menyimpan pengetahuan dari pakar berupa rule/aturan (*if <kondisi> then <aksi>* atau dapat juga disebut *condition-action rules*).

b. Mesin Inferensi (*Inference Engine*)

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

Ada beberapa teknik penalaran yang dapat digunakan salah satunya adalah *Forward Chaining*. *Forward chaining* merupakan pencocokan fakta atau pernyataan dimulai dari bagian kiri (*IF*) atau dengan kata lain penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran (Kusumadewi, 2015). Metode ini sering disebut data-driven karena mesin inferensi menggunakan informasi yang ditentukan oleh pemakai untuk memindahkan ke seluruh jaringan dari logika "AND" dan "OR" sampai sebuah terminal ditentukan sebagai objek. Bila mesin inferensi tidak dapat menentukan objek maka akan meminta informasi lain. Aturan (*Rule*) dimana menentukan objek, membentuk lintasan (*path*) yang mengarah ke objek. Oleh karena itu, hanya satu cara untuk mencapai suatu objek adalah dengan memenuhi semua aturan.

Ada dua pendapat mengenai pelaksanaan metode ini. Pertama dengan cara membawa seluruh data yang didapat ke dalam sistem pakar. Kedua dengan membawa bagian penting-penting saja dari data yang didapat ke dalam sistem pakar. Cara pertama akan baik digunakan jika sistem pakar terhubung dengan proses otomatis dan dapat menerima seluruh data dari basis data. Namun cara

kedua lebih efisien karena menghemat biaya dan waktu dengan mengambil data-data yang penting saja.

Contoh:

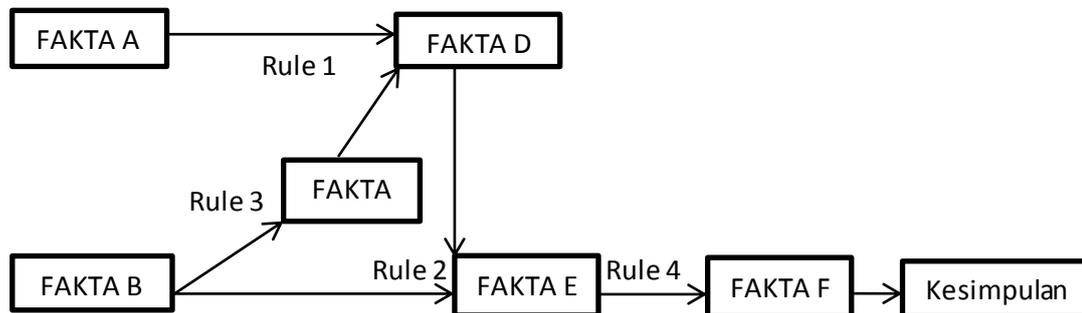
1. R1: IF A and C, THEN E
2. R2: IF D and C, THEN F
3. R3: IF B and E, THEN F
4. R4: IF B, THEN C
5. R5: IF F, THEN G

Fakta nya adalah: A benar dan B benar

Langkah –langkahnya:

1. Dimulai dari R1 Karena C dan E tidak diketahui maka tidak diambil kesimpulan. Pencocokan lanjut di R2, ternyata di R2 juga tidak diambil kesimpulan. Lakukan hal sama pada R3, kemudian pada R4 bernilai benar karena B diketahui benar.
2. Selanjutnya ke R5, namun tidak dapat diambil kesimpulan. Lalu kembali lagi ke atas, R1 bernilai benar karena A benar dan C benar maka F benar.
3. Lalu ke R2, karena D belum diketahui kebenarannya makanya tidak dapat diambil kesimpulan.
4. Lanjut pada R3, karena B dan F benar maka E juga benar.
5. Selanjutnya pada R5, karena F benar maka G benar. Sehingga dengan demikian G adalah kesimpulannya.

Diagram pohon *foward chaining* dapat dilihat pada Gambar 2.2



Gambar 2.2 Diagram Pohon *Foward Chaining*

c. Memori Kerja (*Working Memory*)

Berguna untuk menyimpan fakta yang dihasilkan oleh mesin inferensi dengan penambahan parameter berupa derajat kepercayaan atau dapat juga dikatakan sebagai global database dari fakta yang digunakan oleh aturanaturan yang ada.

d. Fasilitas penjelasan (*Explanation facility*)

Menyediakan kebenaran dari solusi yang dihasilkan kepada pemakai.

e. Akuisisi pengetahuan (*Knowledge acquisition facility*)

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

f. Tampilan pemakai (*User Interface*)

Mekanisme untuk memberi kesempatan kepada pemakai dan sistem pakar untuk berkomunikasi antar muka yaitu dengan menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu

antar muka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

2.2. Variabel

Variabel adalah objek penelitian, atau apa saja yang menjadi fokus didalam suatu penelitian untuk di peajari dan diperoleh kesimpulannya. (Sudaryono, 2014, p. 16).

Pada penelitian ini, peneliti hanya berfokus pada bagian penyakit kulit wajah. Peneliti mengangkat gejala penyakit kulit wajah yang terjadi di area wajah. Beberapa diantaranya:

1. Jerawat

Jerawat adalah suatu keadaan dimana pori-pori kulit tersumbat sehingga menimbulkan kantung nanah yang meradang. Jerawat adalah penyakit kulit yang cukup besar jumlah penderitanya. Berikut adalah gejala-gejala yang terjadi pada kulit wajah:

1. Produksi minyak berlebihan.
2. Adanya sumbatan lapisan kulit mati pada pori-pori.
3. Adanya benjolan dipermukaan kulit.
4. Kulit terasa kasar.

Cara mengatasi terjadinya jerawat pada kulit wajah adalah Kulit wajah harus selalu bersih saat istirahat dirumah, Jangan memecahkan jerawat , jangan sering disentuh, hindari pemakaian kosmetika rias saat tidur.

2. Komedo

Komedo adalah ruam kulit berupa bintik-bintik hitam yang timbul akibat proses oksidasi udara terhadap sekresi kelenjar sebacea di permukaan kulit, seperti agne. Berikut adalah gejala-gejala yang terjadi pada kulit wajah:

1. Produksi minyak berlebihan.
2. Jumlah bakteri pada permukaan kulit meningkat.
3. Perubahan hormone.
4. Iritasi pori-pori kulit.

Cara mengatasi terjadinya komedo pada kulit wajah adalah Komedo bisa diangkat dengan bantuan plester, Memencetnya supaya keluar menggunakan alat khusus, Dengan cara laser.

3. Milia

Milia adalah penonjolan diatas permukaan kulit yang berwarna putih, yang ditimbulkan oleh penyumbatan saluran kelenjar sebacea, seperti pada akne sistika. Berikut adalah gejala-gejala yang terjadi pada kulit wajah:

1. Merasakan sedikit demam.
2. Adanya benjolan putih kecil di wajah.
3. Benjolan yang muncul di sepanjang pipi, hidung, dan dagu.

4. Benjolan putih seperti mutiara pada gusi atau langit-langit mulut.

Cara mengatasi terjadinya milia pada kulit wajah adalah Parafin cair, Minyak paraffin, dan Cairan petrolatum. Ini semua adalah jenis minyak mineral yang dapat menyebabkan benjolan kecil putih ada di wajah.

2.3. Software Pendukung

2.3.1. *Unified Modelling Language (UML)*

UML (*Unified Modelling Language*) adalah sebuah alat bantu yang sangat handal di dunia pengembangan sistem berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. (Novita, 2015). Dalam UML terdiri dari 13 macam diagram yang terbagi dalam tiga kategori yaitu: (Rosa & Shalahuddin, 2011:121)

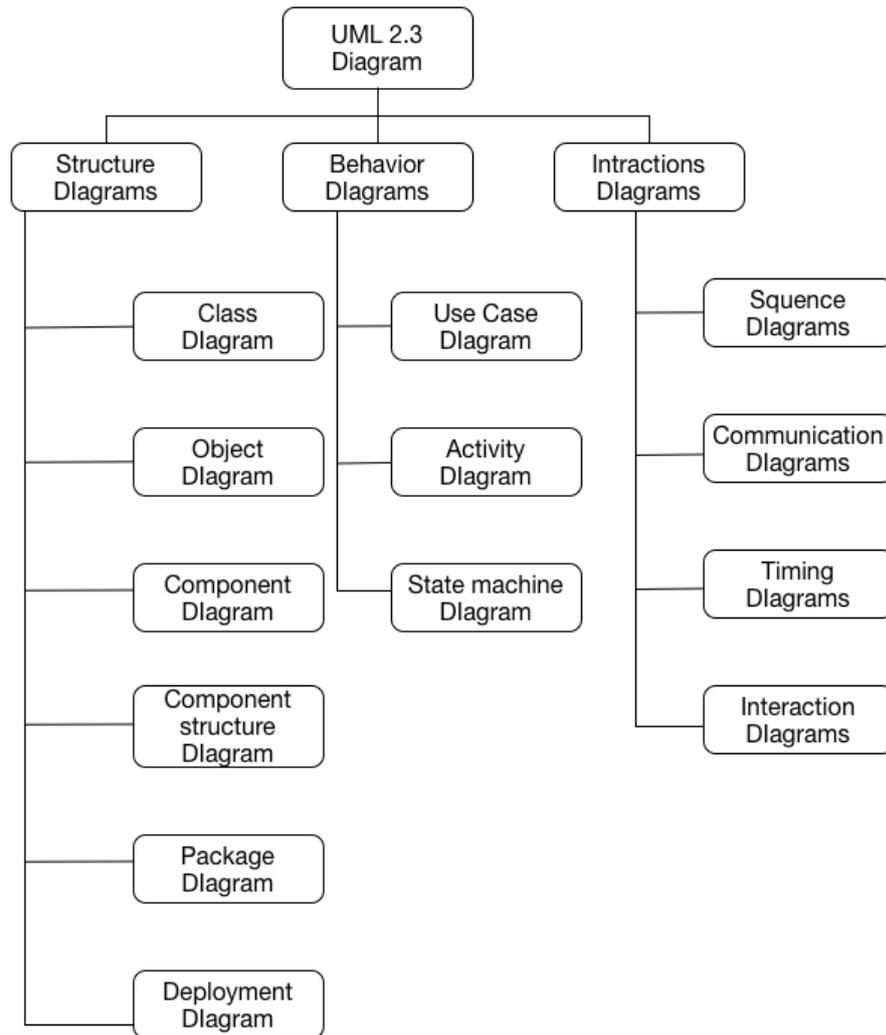
1. *Structure Diagram* : kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior Diagram* : kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

3. *Interaction diagram* : kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

Menurut (Rosa & Shalahuddin, 2011) UML merupakan sintesis dari tiga metode analisis dan perancangan berbasis objek serta ditambah keunggulan metode-metode berorientasi objek lainnya (*Fision, Shlaer-Mellon, Coad-Yurdon*) yang juga disintesakan dalam UML menawarkan pendekatan yang cukup baik serta digunakan cukup luas digunakan di industri perangkat lunak. Secara umum, UML merupakan 'bahasa' untuk visualisasi, spesifikasi, konstruksi serta dokumentasi.

Menurut (Rosa & Shalahuddin, 2011) Pemodelan adalah gambaran dari realita yang simple dan dituangkan dalam bentuk pemetaan dengan aturan tertentu". Pemodelan sistem dilakukan untuk menggambarkan realita yang simple sehingga dapat mempermudah pengembangan sistem tersebut kedepannya karena lebih terencana dan terstruktur. Ada beberapa perangkat pemodelan sistem yang bisa digunakan dalam pemodelan sistem, diantaranya adalah *Unified Modeling Language (UML)*.

Diagram Unified Modeling Language (UML) adalah kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. *Diagram UML 2.3* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar 2.3 dibawah ini:

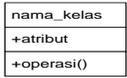
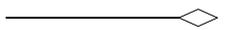


Gambar 2.3 Diagram UML

2.3.1.1. *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut ini adalah simbol-simbol yang ada pada *class diagram*: (Rosa & Shalahuddin, 2011:122)

Tabel 2.2 Simbol *Class Diagram*

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

Sumber : Rosa & Shalahuddin (2011:123-124)

2.3.1.2. *Object Diagram*

Object Diagram menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Berikut adalah simbol-simbol yang ada pada diagram objek: (Rosa & Shalahuddin, 2011:124)

Tabel 2.3 Simbol *Object Diagram*

Simbol	Deskripsi
Objek	Objek dari kelas yang berjalan saat sistem dijalankan

<table border="1"> <tr> <td>nama objek : nama kelas</td> </tr> <tr> <td>Atribut = nilai</td> </tr> </table>	nama objek : nama kelas	Atribut = nilai	
nama objek : nama kelas			
Atribut = nilai			
Link 	Relasi antar objek		

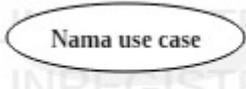
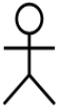
(Sumber : Rosa & Shalahuddin 2011:124)

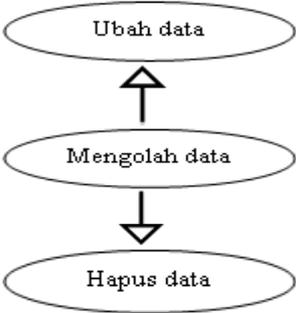
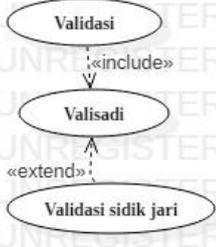
2.3.1.3. Use Case Diagram

Use Case Diagram merupakan pemodelan kelakuan (*behavior*) sistem informasi yang akan dibuat. Dan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol-simbol yang ada pada *use case diagram*:

(Rosa & Shalahuddin, 2011:130)

Tabel 2.4 Simbol *Use Case*

Simbol	Deskripsi
<i>Use Case</i> 	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>
Aktor/ <i>actor</i>  nama aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi <i>actor</i> belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang

Simbol	Deskripsi
	<p>lebih umum dari lainnya, misalnya</p>  <pre> graph BT Mengolah([Mengolah data]) --> Ubah([Ubah data]) Hapus([Hapus data]) --> Mengolah </pre> <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasi (umum)</p>
<p>Asosiasi / <i>association</i></p> <hr/>	<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 walau tanpa <i>use case</i> tambahan yaitu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan sama dengan <i>use case</i> yang ditambahkan, missal</p>  <pre> graph TD Validasi([Validasi]) -.-> <<include>> Valisadi([Valisadi]) ValidasiSidik([Validasi sidik jari]) -.-> <<extend>> Valisadi </pre> <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan</p>

Simbol	Deskripsi
<p>Menggunakan / <i>include</i> / <i>uses</i></p> <p style="text-align: center;"><<include>></p>  <p style="text-align: center;"><<uses>></p> 	<p>Relasi <i>use case</i> tambah ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i>. Ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ul style="list-style-type: none"> • <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, missal pada kasus berikut: • <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan. Misal pada kasus: <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan</p>

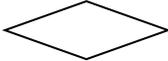
(Sumber : Rosa & Shalahuddin 2011:131-133)

2.3.1.4. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Berikut adalah simbol-simbol yang ada pada *activity diagram*: (Rosa & Shalahuddin, 2011:134)

Tabel 2.5 Simbol *Activity Diagram*

Simbol	Deskripsi
<p>Status awal</p> 	<p>Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status</p>

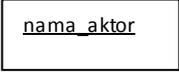
Simbol	Deskripsi
	awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  Atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

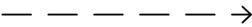
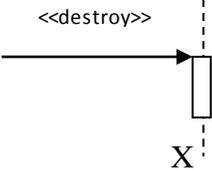
Sumber : Rosa & Shalahuddin 2011:134-135)

2.3.1.5. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Berikut adalah simbol-simbol yang ada pada *sequence diagram*: (Rosa & Shalahuddin, 2011:137)

Tabel 2.6 Simbol *Sequence Diagram*

Simbol	Deskripsi
<p>Aktor</p>  <p>Atau</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor</p>
<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe <i>create</i></p> <p><<create>></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<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,</p>

Simbol	Deskripsi
	Arah panah mengarah pada objek yang memiliki operasi/metode karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram, kelas sesuai dengan kelas objek yang berinteraksi
Pesan tipe <i>send</i> 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/ masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
Pesan tipe <i>return</i> 1 : keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>

Sumber : Rosa & Shalahuddin (2011:138-139)

2.3.1.6. Bagan Aliran (*Flowmap*)

Bagan alir adalah bagan yang menunjukkan alir di dalam program atau prosedur sistem secara logika, bagan alir digunakan sebagai alat Bantu komunikasi dan untuk dokumentasi. Bagan alir meliputi: (Rinawati & Candrawati, 2013)

1. Bagan alir dokumen atau *document flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusannya.
2. Bagan alir sistem atau *flowchart* merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem, bagan alir sistem menunjukkan apa yang dikerjakan sistem.
3. Bagan alir sistematis merupakan alir yang mirip dengan bagan alir sistem yaitu untuk menggambarkan prosedur dalam sistem perbedaannya bagan alir sistematis selain menggunakan *symbol* bagan alir juga menggunakan gambar- gambar *computer* juga gambar- gambar lain yang digunakan hal ini untuk memudahkan pengkomunikasian.
4. Bagan alir program merupakan bagan alir yang menjelaskan secara rinci langkah-langkah proses program. Bagan alir proses merupakan bagan alir yang banyak digunakan di teknik industri namun berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur serta menggunakan dalam suatu *symbol* tersendiri.

2.3.1.7. Pemograman Web

Website (situs *Web*) merupakan kumpulan dari halaman-halaman *Web* yang berhubungan dengan *file-file* lain yang terkait. Didalam sebuah *Website* terdapat suatu halaman yang dikenal dengan sebutan *Home Page*. *Home Page*

adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi *Website*. Dari *home page*, pengunjung dapat mengklik *hyperlink* untuk pindah ke halaman lain yang terdapat dalam *Website* tersebut (Hendrianto, 2014).

2.3.1.8. Hypertext Markup Language (HTML)

HTML (*Hypertext Markup Language*) adalah suatu bahasa yang digunakan untuk menulis halaman *Web*. HTML dirancang untuk digunakan tanpa tergantung pada suatu *platform tertentu* atau *platform independent* (Rahmad *et al*, 2013). HTML (*Hypertext Markup Language*) adalah sebuah bahasa pemrograman berbentuk skrip-skrip yang berguna untuk membuat sebuah halaman *Web*. HTML dapat dibaca oleh berbagai *platform* seperti Windows, Linux, Macintosh. Kata “*Markup Language*” pada HTML menunjukkan fasilitas yang berupa tanda tertentu dalam skrip HTML dimana kita bisa mengatur judul, garis, tabel, gambar dan lain-lain dengan perintah yang telah ditentukan pada elemen HTML. HTML sendiri dikeluarkan oleh W3C (*World Wide Web Consortin*), setiap terjadi perkembangan level HTML harus dievaluasi ketat dan disetujui oleh W3C (Hariadi, 2013).

HTML yang mempunyai kepanjangan dari *Hyper Text Markup Language* yang merupakan pondasi awal akan terciptanya suatu aplikasi *Web*, karena HTML memiliki fungsi untuk membangun kerangka ataupun format *Web* yang digunakan untuk menampilkan suatu informasi kedalam halaman *Web* atau *browser* (Saputra, 2012).

Menurut (Djaelangara, 2015) *Hypertext Markup Language* (HTML) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman *Web*, menampilkan berbagai informasi di dalam sebuah penjelajah *Web internet* dan pemformatan *hiperteks* sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman *Web* dengan perintah-perintah HTML.

HTML merupakan pengembangan dari pemformatan dokumen teks yaitu *Standard Generalized Markup Language* (SGML). HTML sebenarnya adalah dokumen ASCII atau teks biasa yang dirancang untuk tidak tergantung pada suatu *system* operasi tertentu. HTML dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN dan dipopulerkan pertama kali oleh *browser* Mosaic. Selama awal tahun 1990 HTML mengalami perkembangan yang sangat pesat. Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya (Purbadian, 2015).

2.3.1.9. PHP (*Hypertext Preprocessor*)

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *Web* yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang akan diberikan akan sepenuhnya akan dijalankan di *server* disertakan pada dokumen HTML. Pembuatan *Web* ini merupakan kombinasi antara PHP sendiri sebagai bahasa pemrograman dan

HTML sebagai pembangun halaman *Web* (Hendrianto, 2014). PHP merupakan bahasa pemrograman berbentuk *script* yang ditempatkan dalam *server* yang pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Hasil dari pengolahan akan dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus PHP dirancang untuk membentuk *Web* dinamis. Artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya bisa menampilkan isi basis data ke halaman *Web* (Renatha *et al.*, 2016).

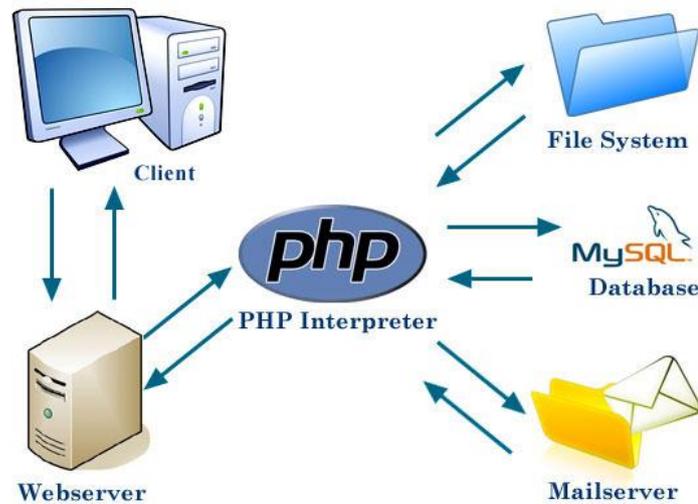
PHP merupakan singkatan dari PHP (*hypertext preprocessor*). Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya yang dikirim ke klien, tempat pemakai menggunakan *browser*. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi *tool* yang disebut "*personal home page*". Paket inilah yang menjadi cikal bakal PHP (Rahmad & Purnama, 2013). Selain dapat digunakan untuk berbagai sistem operasi dan bersifat *open source*. PHP ternyata memiliki banyak kelebihan yang tidak dimiliki oleh bahasa *script* sejenis, yaitu:

1. PHP dapat digunakan semua jenis sistem operasi seperti *Windows*, *Linux*, dan lain-lain.
2. PHP memiliki kemampuan untuk mengolah keluarga gambar, *file* berformat pdf, dan *Movies Flash*.
3. PHP didukung oleh banyak *Database Management System* (DBMS) seperti MySQL, Oracle, dan lain-lain. (Renatha *et al.*, 2016)

PHP yang mempunyai kepanjangan dari *PHP Hypertext Preprocessor* merupakan bahasa pemrograman berbentuk skrip yang ditempatkan disisi *server*, sehingga php disebut juga sebagai bahasa *Server Side Scripting*, artinya bahwa dalam menjalankan php selalu membutuhkan *Web server*, dan untuk melihat hasilnya menggunakan *Web browser* (Purbadian , 2015).

PHP dirancang untuk membentuk tampilan *Web* yang dinamis artinya php dapat membentuk suatu tampilan berdasarkan permintaan *user*, misalnya dapat mengakses *database* dan menampilkannya pada halaman *Web*. php menyatu dengan kode html, namun beda kondisinya. Maksudnya adalah kode yang dibuat menggunakan html dirancang untuk membangun suatu pondasi awal dari kerangka layout *Web*, sedangkan php digunakan untuk memproses data dari sisi *server*, sehingga terciptalah suatu tampilan *Web* yang dinamis.

Prinsip kerja PHP mirip dengan kode HTML, hanya saja ketika berkas PHP yang diminta didapatkan oleh *Web server*, isinya segera dikirim ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya berupa kode HTML ke *Web server* dan selanjutnya *Web server* menyampaikan ke *client*.



Gambar 2.4 Skema Kerja PHP

2.3.1.10. MySQL

MySQL (*MyStructure Query Language*) adalah aplikasi atau sistem untuk mengelola *database* atau manajemen data (Rahmad & Purnama, 2013). Untuk menyimpan segala informasi ke komputer menggunakan data. MySQL bertugas mengatur dan mengelola data-data pada *database*, selain itu MySQL dikenal sebagai sistem yang efisien dan *reliable*, proses *query* cepat dan mudah, sehingga cocok untuk aplikasi berbasis *Web* (Gunawan, 2012; Rahmad & Purnama, 2013). MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data *Database Management System* atau DBMS dari sekian banyak DBMS seperti Oracle, MS SQL, Postgre SQL dan lainya (Prayitno & Safitri, 2015).

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*,

multiuser, dengan sekitar 6 juta instalasi di seluruh dunia. *MySQL* AB membuat *MySQL* tersedia sebagai perangkat lunak gratis dibawah lisensi *GNU General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Watung, 2014).

MySQL (dibaca : mai se kyu el) adalah *Multiuser database* yang menggunakan bahasa *Structured Query Language* (SQL). *MySQL* merupakan *software* sistem manajemen *database* (*Database Management system – DBMS*). *MySQL* awalnya dibuat oleh oleh perusahaan konsultan bernama TcX yang berlokasi di Swedia. Saat ini, pengembangan *MySQL* berada dibawah naungan perusahaan *MySQL* AB (Purbadian , 2015).

Menurut (Purbadian, 2015) tinjauan luas mengenai *MySQL*, antara lain:

1. *MySQL* merupakan suatu *Database Management system* (DBMS)
2. *MySQL* adalah suatu *Relational Database Management system* (RDBMS)
3. Perangkat lunak *MySQL* didistribusikan secara *open source*
4. *Database MySQL Server* sangat cepat, dapat dipercaya dan mudah digunakan
5. *Database Server MySQL* bekerja dalam lingkungan *client/server*
6. Dukungan terhadap perangkat lunak *MySQL* tersebar luas dan mudah ditemukan
7. *MySQL* mendukung penggunaan *database* dengan berbagai kriteria pengaksesan. Maksudanya adalah untuk mengatur *user* tertentu agar bisa mengakses data yang bersifat rahasia
8. *MySQL* tersedia pada beberapa platform seperti Windows, Linux, Unix, dan lain-lain.

9. *MySQL* dapat mendeteksi pesan kesalahan pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.
10. *MySQL* mampu menangani basis data dalam skala besar.

2.3.1.11. Adobe Dreamweaver CS 6

Adobe Dreamweaver CS6 merupakan salah satu program aplikasi yang digunakan untuk membangun sebuah *website*, baik secara grafis maupun dengan menuliskan kode sumber secara langsung. (Team, 2010; Rahmad & Purnama, 2013).

Menurut (Prabantini, 2013) *Dreamweaver* adalah sebuah *HTML* editor profesional untuk mendesain *web* secara visual dan mengelola situs atau halaman *web*. *Dreamweaver CS6* memiliki kemampuan untuk mendesain *web*, menyunting kode, serta pembuatan aplikasi *web* dengan menggunakan berbagai bahasa pemrograman *web*, antara lain: *JPS*, *PHP*, *ASP* dan *ColdFusion*.

Menurut (Prabantini, 2013) komponen-komponen yang disediakan oleh ruang kerja *Dreamweaver CS6* antara lain adalah *InsertBar*, *Document Toolbar*, Jendela Dokumen, *Panel Group*, *Tag Selector*, *Property Inspector*, dan *Site Panel*.

2.3.1.12. XAMPP

Menurut (Prabantini, 2013) XAMPP merupakan pengembangan dari LAMP (Linux Apache, MySQL, PHP and PERL), XAMPP ini merupakan project non-profit yang di kembangkan oleh Apache friends yang didirikan Kai Oswald Seidler dan Kay Vogelgesang pada tahun 2002, project mereka ini bertujuan mempromosikan penggunaan apache web server.

Menurut (Prabantini, 2013) XAMPP merupakan salah satu paket instalasi Apache, PHP dan MySQL instant yang dapat kita gunakan untuk membantu proses instalasi ketiga produk tersebut.

2.4. Penelitian Terdahulu

Pada tinjauan penelitian sebelumnya akan dibahas secara lengkap jurnal dan artikel yang mendukung sebagai dasar pembahasan intepretasi penelitian pada bahan sebelumnya. Penelitian terdahulu dari penelitian ini adalah sebagai berikut:

1. **Deddy Kusbianto, Rizky Ardiansyah & Dzaki Alwan Hamadi (2017)**, Jurnal Informatika Polinema, Volume 4, Edisi 1, November 2017, ISSN: 2407-070X, dengan judul penelitian “Implementasi Sistem Pakar *Forward Chaining* Untuk Identifikasi Dan Tindakan Perawatan Jerawat Wajah”. Hasil pengujian fungsional dan sistem menunjukkan bahwa metode *Forward Chaining* dapat digunakan untuk mengembangkan sistem pakar

identifikasi dan tindakan perawatan jerawat wajah yang diimplementasikan berbasis *dekstop*.

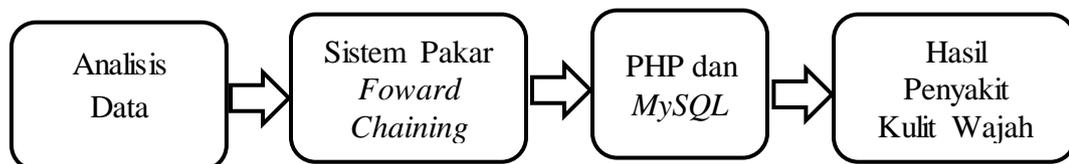
2. **Fitri Nuraeni, Yoga Handoko Agustin & Endah Nirwani Yusup (2016)**. Seminar Nasional Teknologi Informasi dan Multimedia 2016. STMIK AMIKOM Yogyakarta, 6-7 Februari 2016. ISSN : 2302-3805, dengan judul penelitian “Aplikasi Pakar Untuk Diagnosa Penyakit Kulit Menggunakan Metode *Forward Chaining* Di Al Arif Skin Care Kabupaten Ciamis”. Penelitian ini menghasilkan aplikasi sistem pakar yang dapat digunakan untuk mendiagnosa penyakit kulit dan memberikan informasi serta solusi penanganan terhadap penyakit yang diderita sehingga pelayanan terhadap pasien penyakit kulit dapat tetap berjalan tanpa harus menunggu kehadiran dokter ahli untuk menangani.
3. **Fristi Riandari (2017)**. Jurnal Mantik Penusa, Vol. 1, No. 2, Desember 2017, pp.85-89. e-ISSN 2580-9741, p-ISSN 2088-3943, dengan judul penelitian “Sistem Pakar Mendiagnosa Penyakit Kulit Wajah Menggunakan Metode *Certainty Factor*”. Dengan adanya aplikasi ini dapat membantu memudahkan masyarakat khususnya para penderita penyakit kulit wajah untuk menegetahui jenis penyakit kulit wajah yang di derita dan dapat segera di obati.
4. **Joko S Dwi Raharjo, Damdam Damiyana & Supardi (2016)**. JURNAL SISFOTEK GLOBAL ISSN: 2088 – 1762 Vol. 6 No. 1 / Maret 2016, dengan judul penelitian “Sistem Pakar Diagnosa Penyakit Kulit Dengan Menggunakan Metode *Forward Chaining* Berbasis *Web Mobile* (Studi

Kasus di PT. Kimia Farma Senen Tbk.)”, Hasil rancangan sistem ini dalam bentuk aplikasi *mobile Web Mobile* yang dapat digunakan oleh semua orang yang ingin mengetahui tentang gejala penyakit kulit yang disebabkan oleh virus. Perancangan sistem pakar ini menggunakan bahasa pemrograman *Web Mobile Studio*.

5. **M. Ramaddan Julianti, Agus Budiman & Intan Ardinda Pramanova (2018).** JURNAL SISFOTEK GLOBAL ISSN : 2088 – 1762 Vol. 8 No. 2, September 2018, dengan judul penelitian “Sistem Pakar Diagnosa Penyakit Kulit Berbasis *Web* dengan Metode *Forward Chaining*”, Hasil penelitian ini didapat yaitu dengan sistem pakar ini diharapkan dapat membantu orang mengetahui tentang penyakit kulit wajah dan cara mengatasinya.

2.5. Kerangka Pemikiran

Berdasarkan teori-teori yang telah diperoleh dan dijelaskan, maka kerangka berpikir dari penelitian, digambarkan pada kerangka pemikiran yang disajikan pada gambar di bawah ini:



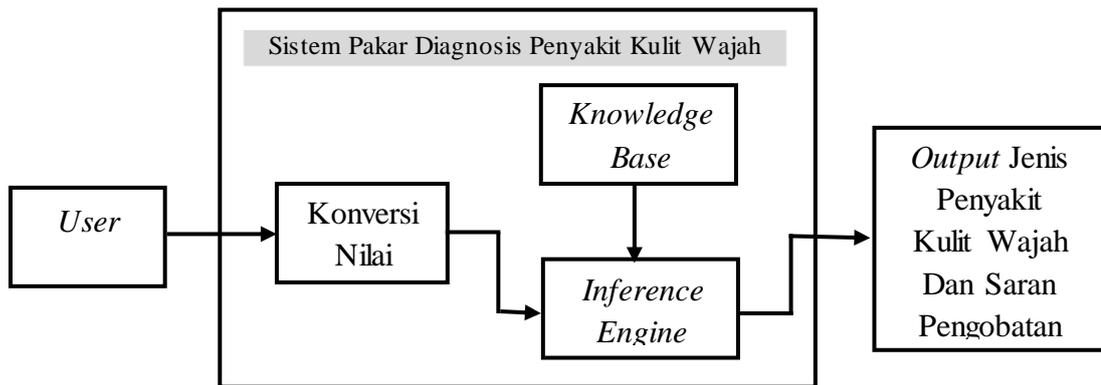
Gambar 2.5 Kerangka Pemikiran

BAB III METODE PENELITIAN

Metode penelitian adalah cara ilmiah untuk mendapatkan data yang digunakan untuk tujuan dan kegunaan tertentu, seperti untuk penemuan, pengembangan, dan pembuktian. Data nantinya diperoleh melalui kriteria penelitian yang valid, yaitu untuk menunjukkan derajat ketetapan antara data yang sesungguhnya terjadi pada objek dengan data yang dapat dikumpulkan oleh penelitian.

3.1. Desain Penelitian

Desain dari penelitian adalah semua proses yang diperlukan dalam perencanaan dan pelaksanaan penelitian. Dalam pengertian yang lebih sempit, desain penelitian hanya mengenai pengumpulan dan analisis data saja. Dalam merencanakan penelitian, desain dimulai dengan mengadakan penyelidikan dan evaluasi terhadap penelitian yang sudah dikerjakan dan diketahui, dalam memecahkan masalah. Dari sini pula dapat dicari beberapa petunjuk tentang desain yang akan dibuat untuk penelitian yang akan dikembangkan. Desain penelitian dapat dilihat pada Gambar 3.1 yang menggambarkan hubungan antara elemen-elemen utama dari sistem pakar diagnosis penyakit kulit wajah yang akan dibuat.



Gambar 3.1. Blok Diagram Sistem Pakar Penyakit Kulit Wajah

Penjelasan dari desain arsitektur untuk sistem pakar diagnosis penyakit kulit pada wajah adalah sebagai berikut:

1. Pasien

Pasien dalam sistem pakar diagnosis penyakit kulit pada wajah ini merupakan orang yang berperan dalam memasukkan jawaban dari pertanyaan konsultasi berupa fakta-fakta gejala yang terjadi pada wajah. Nilai dari jawaban tersebut nantinya akan diolah untuk mendapatkan suatu kesimpulan.

2. Konversi Nilai

Proses konversi nilai merupakan proses dalam melakukan konversi jawaban pertanyaan konsultasi dari Pasien menjadi sebuah nilai tertentu yang nantinya akan diolah dalam proses inferensi.

3. *Knowledge Base*

Knowledge base berisi kumpulan dari fakta-fakta mengenai situasi, kondisi atau permasalahan yang ada; dan aturan-aturan yang digunakan sebagai acuan dalam menggunakan pengetahuan untuk menyelesaikan masalah yang ada. Dalam sistem pakar diagnosis penyakit kulit pada wajah ini, fakta dan aturan yang ada

telah di desain berupa data gejala penyakit kulit wajah, data penyakit kulit wajah, dan data saran pengobatan terhadap penyakit kulit wajah.

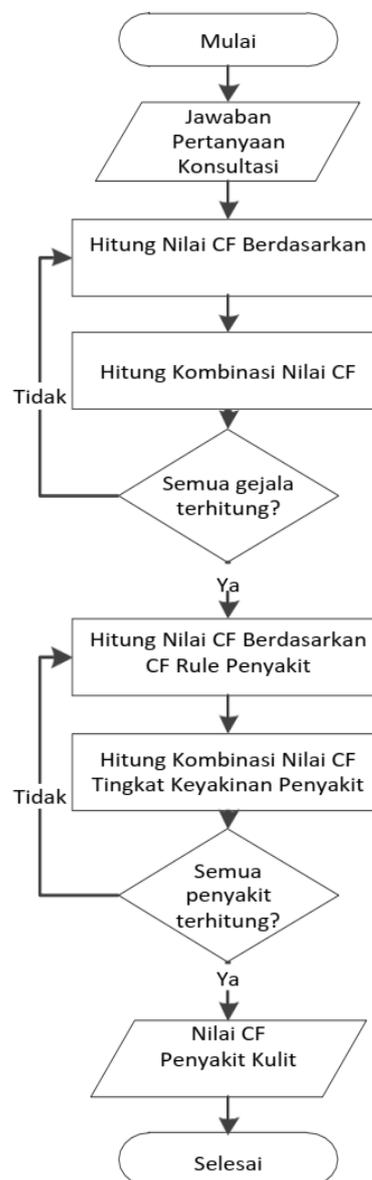
4. *Inference Engine*

Mesin Inferensi adalah sebuah program untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan *rule*, model, dan fakta yang disimpan dalam *knowledge base* untuk mencapai solusi atau kesimpulan. Dalam sistem pakar diagnosis penyakit kulit pada wajah ini dapat dilihat dalam perhitungan *Certainty Factor (CF)*.

Proses dalam mesin inferensi ini dimulai dengan inputan jawaban pertanyaan gejala-gejala yang dialami wajah. Setelah semua pertanyaan dijawab, maka jawaban tersebut akan dikonversi dari sebuah "*uncertain term*" menjadi sebuah nilai CF. Nilai hasil konversi nantinya akan dikalikan dengan nilai CF *rule* gejala. Setelah mendapatkan hasil perkalian nilai CF gejala, maka akan dilakukan verifikasi apakah nilai tersebut harus dikombinasi untuk menghasilkan sebuah nilai atau tidak.

Setelah semua hasil perhitungan nilai CF gejala didapatkan, maka proses selanjutnya melakukan perkalian dengan nilai CF *rule* dari setiap penyakit kulit. Nilai CF penyakit pertama diperoleh dari perhitungan nilai CF berdasarkan CF *rule* penyakit pertama. Proses perhitungan nilai CF dari penyakit diawali dengan mencari nilai CF dari gejala pertama dan gejala kedua, setelah itu kedua nilai CF tersebut akan dikombinasikan. Hasil dari nilai CF kombinasi pertama nantinya akan dikombinasikan dengan nilai CF *rule* dari gejala berikutnya. Proses perhitungan nilai CF kombinasi akan diulang sampai dengan gejala terakhir dari

penyakit pertama. Setelah perhitungan nilai CF dari penyakit pertama selesai, maka proses perhitungan akan dilanjutkan sampai dengan penyakit terakhir. Apabila nilai semua penyakit ditemukan, maka proses perhitungan telah selesai dan akan menampilkan nilai CF dari semua penyakit kulit wajah beserta saran pengobatan yang harus dilakukan. Gambar 3.2 dibawah ini merupakan *flowchart* dari mesin inferensi sistem pakar diagnosis penyakit kulit wajah.



Gambar 3.2. *Flowchart* Sistem Pakar Diagnosis Penyakit Kulit Wajah

5. Output

Output merupakan hasil kesimpulan dari sistem yang menunjukkan jawaban dari gejala atau fakta mengenai penyakit kulit yang telah di *input*-kan. *Output* yang dihasilkan sistem pakar ini merupakan hasil diagnosis penyakit kulit wajah beserta saran pengobatan yang harus dilakukan.

3.2. Teknik Pengumpulan Data

Metode yang digunakan dalam proses pengumpulan data sebagai bahan pembuatan sistem adalah:

1. Metode Wawancara

Teknik wawancara dilakukan dengan cara berkonsultasi langsung dengan dr. Amelia Fenowati di Klinik OASE BEUTY dan SKINCARE Kota Batam sebagai pakar penyakit kulit wajah untuk mendapatkan gambaran mengenai jenis-jenis penyakit kulit wajah, gejala-gejala yang disebabkan oleh penyakit kulit wajah, maupun solusi terhadap penyakit kulit wajah.

2. Studi Pustaka

Metode studi pustaka merupakan teknik pencarian dengan melakukan pencarian data lewat literatur-literatur yang terkait misalnya buku-buku referensi, artikel tentang masalah sistem pakar, penyakit kulit wajah dan pemograman PHP dan *MySQL*.

3.3. Operasional Variabel

Menurut (Hasibuan, 2017:130) Variabel adalah sesuatu yang akan menjadi objek atau sering juga sebagai faktor yang berperan dalam peristiwa atau gejala yang akan diteliti. Ada beberapa variabel-variabel yang berkaitan dengan penelitian yang disajikan pada table 3.1 sebagai berikut:

Tabel 3.1. Operasional Variabel

Variabel	Kode Penyakit	Gejala Penyakit
Gejala penyakit kulit, pada kulit wajah	P01	Jerawat
	P02	Komedo
	P03	Milia

(Sumber : Data Penelitian, 2019)

3.4. Alur Atau Proses Perancangan Sistem

Metode perancangan adalah suatu cara atau tahapan yang dilakukan dalam sebuah proses perancangan, metode ini dibutuhkan untuk memudahkan perancang dalam mengembangkan ide rancangan. Metode yang dilakukan oleh seseorang berbeda-beda berdasarkan kebutuhannya, dalam sistem pakar untuk mendiagnosis penyakit kulit wajah berbasis *web* menggunakan metode *forward chaining*. Ada 6 tahap dalam pengembangan sistem pakar berikut ini adalah:

a. Identifikasi

Tahap ini merupakan tahap penentuan hal-hal penting sebagai dasar dari permasalahan yang akan dianalisis. Tahap ini merupakan tahap untuk mengkaji dan membatasi masalah yang akan diimplementasikan dalam

sistem. Setiap masalah yang diidentifikasi harus dicari solusi, fasilitas yang akan dikembangkan, penentuan jenis bahasa pemrograman dan tujuan yang ingin dicapai dari proses pengembangan tersebut. Apabila proses identifikasi masalah dilakukan dengan benar maka akan dicapai hasil yang optimal.

b. Konseptualisasi

Hasil identifikasi masalah dikonseptualisasikan dalam bentuk relasi antar data, hubungan antar pengetahuan dan konsep-konsep penting dan ideal yang akan diterapkan dalam sistem. Konseptualisasi juga menganalisis data-data penting yang harus dialami bersama dengan pakar di bidang permasalahan tersebut. Hal ini dilakukan untuk memperoleh konfirmasi hasil wawancara dan observasi sehingga hasilnya dapat memberikan jawaban pasti bahwa sasaran permasalahan tepat, benar dan sudah sesuai.

c. Tahap Formalisasi

Apabila tahap konseptualisasi telah selesai dilakukan, maka di tahap formalisasi konsep tersebut diimplementasikan secara formal, misalnya memberikan kategori sistem yang akan dibangun, mempertimbangkan beberapa faktor pengambilan keputusan seperti keahlian manusia, kesulitan dan tingkat kesulitan yang mungkin terjadi, dokumentasi kerja, dan sebagainya.

d. Implementasi

Apabila pengetahuan sudah diformalisasikan secara lengkap, maka tahap implementasi dapat dimulai dengan membuat garis besar masalah kemudian

memecahkan masalah ke dalam modul-modul. Untuk memudahkan maka harus diidentifikasi:

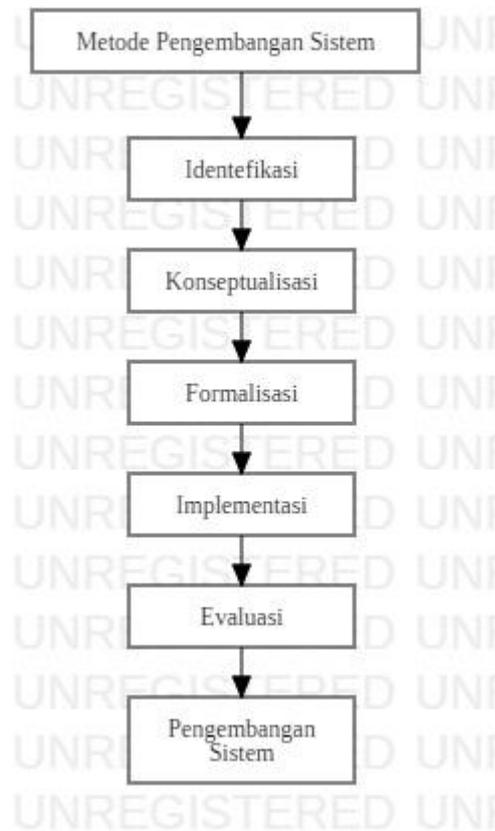
- 1) Apa saja yang menjadi inputan
- 2) Prosesnya digambarkan dalam bagan alur dan basis aturannya.
- 3) Apa saja yang menjadi output atau hasil dan kesimpulannya.
- 4) Sesudah itu semuanya diubah dalam bahasa yang mudah dimengerti oleh komputer dengan menggunakan tahapan fase seperti gambaran fase pengembangan sistem pakar.

e. Evaluasi

Sistem pakar yang selesai dibangun, perlu untuk dievaluasi untuk menguji dan memunculkan kesalahannya. Hal ini merupakan hal yang umum dilakukan karena suatu sistem belum tentu sempurna setelah selesai pembuatannya sehingga proses evaluasi diperlukan untuk penyempurnaannya. Dalam evaluasi akan ditemukan bagian-bagian yang harus di koreksi untuk menyamakan permasalahan dan tujuan dari pembuatan sistem.

f. Pengembangan sistem

Pengembangan sistem diperlukan sehingga sistem yang dibangun tidak menjadi usang dan investasi sistem tidak sia-sia. Hal pengembangan sistem yang paling berguna adalah proses dokumentasi sistem di mana di dalamnya tersimpan semua hal penting yang dapat menjadi tolak ukur pengembangan sistem di masa mendatang termasuk di dalamnya adalah kamus pengetahuan masalah yang diselesaikan.



Gambar 3.3. Diagram Metode Pengembangan Sistem
(Sumber: Kecerdasan Buatan, Kristanto, 2014).

3.4.1. Desain Basis Pengetahuan

Sumber pengetahuan dan fakta yang diperoleh melalui wawancara dengan studi kasus tentang penyakit yang berkaitan dengan kulit wajah, sumber pengetahuan dan fakta diperoleh dari gejala-gejala yang terjadi pada kulit wajah, dengan adanya penyakit kulit wajah dan cara mengatasinya. Pengetahuan dan fakta tersebut dapat dilihat dalam tabel indikator, tabel penyebab dan solusi, table gejala dan table aturan.

Tabel 3.2 Gejala dan Solusi

Kode Penyakit	Kode Gejala	Gejala	Solusi
P01	G001	Produksi minyak berlebihan	Kulit wajah harus selalu bersih saat istirahat dirumah, Jangan memecahkan jerawat , jangan sering disentuh, hindari pemakaian kosmetika rias saat tidur.
	G002	Adanya sumbatan lapisan kulit mati pada pori-pori	
	G003	Adanya benjolan dipermukaan kulit	
	G004	Kulit terasa kasar	
P02	G001	Produksi minyak berlebihan	Komedo bisa diangkat dengan bantuan plester, Memencetnya supaya keluar menggunakan alat khusus, Dengan cara laser.
	G005	Jumlah bakteri pada permukaan kulit meningkat.	
	G006	Perubahan hormone	
	G007	Iritasi pori-pori kulit	
P03	G008	Merasakan sedikit demam	Parafin cair, Minyak paraffin, dan Cairan petrolatum. Ini semua adalah jenis minyak mineral yang dapat menyebabkan benjolan kecil putih ada diwajah.
	G009	Adanya benjolan putih kecil di wajah	
	G0010	Benjolan yang muncul di sepanjang pipi, hidung dan dagu	
	G0011	Benjolan putih seperti mutiara pada gusi atau langit-langit mulut	

Sumber: Data Penelitian (2019)

Sistem pakar yang menggunakan metode *forward chaining* pada penelitian ini digunakan untuk mengetahui penyakit kulit wajah sehingga data solusi tidak diberikan kode. Data solusi hanya sebagai keterangan tambahan yang digabungkan kedalam tabel penyakit dan solusi (table 3.3). Dalam Tabel 3.2 menjelaskan nama kode penyakit, kode gejala, dan solusinya untuk memudahkan dalam pengolahan data.

Data aturan merupakan data yang berisi data-data indikator, penyebab penyakit dan gejala penyakit telah diberi kode sebelumnya. Data aturan ini disusun untuk memudahkan peneliti dalam menyusun kaidah yang akan digunakan sebagai basis pengetahuan dalam sistem pakar pada penelitian ini. Susunan data aturan yang digunakan dapat dilihat pada table dibawah ini.

Tabel 3.3 Aturan

Kode Penyakit	Kode Gejala
P01	G001, G002, G003, G004
P02	G001, G005, G006, G007
P03	G008, G009, G0010, G0011

Sumber: Data penelitian (2019)

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

1. Kaidah 1 : *IF G001 AND G002 AND G003 AND G004 THEN P01*
2. Kaidah 2 : *IF G001 AND G005 AND G006 AND G007 THEN P02*
3. Kaidah 3 : *IF G008 AND G009 AND G0010 AND G0011 THEN P03*

Berdasarkan kaidah (*rule*) tersebut, maka dapat dijelaskan bahwa:

1. Jika gejala penyakit yang timbul pada kulit wajah adalah Produksi minyak berlebihan, Adanya sumbatan lapisan kulit mati pada pori-pori, Adanya benjolan dipermukaan kulit, Kulit terasa kasar . Penyebabnya adalah jerawat.
2. Jika gejala penyakit yang timbul pada kulit wajah adalah Produksi minyak berlebihan, Jumlah bakteri pada permukaan kulit meningkat, Perubahan hormone, Iritasi pori-pori kulit. Penyebabnya adalah Komedo.

3. Jika gejala penyakit yang timbul pada kulit wajah adalah Merasakan sedikit demam, Adanya benjolan putih kecil di wajah, Benjolan yang muncul di sepanjang pipi, hidung dan dagu, Benjolan putih seperti mutiara pada gusi atau langit-langit mulut. Penyebabnya adalah Milia.

Berdasarkan kaidah yang telah dibuat tersebut maka table keputusannya adalah sebagai berikut:

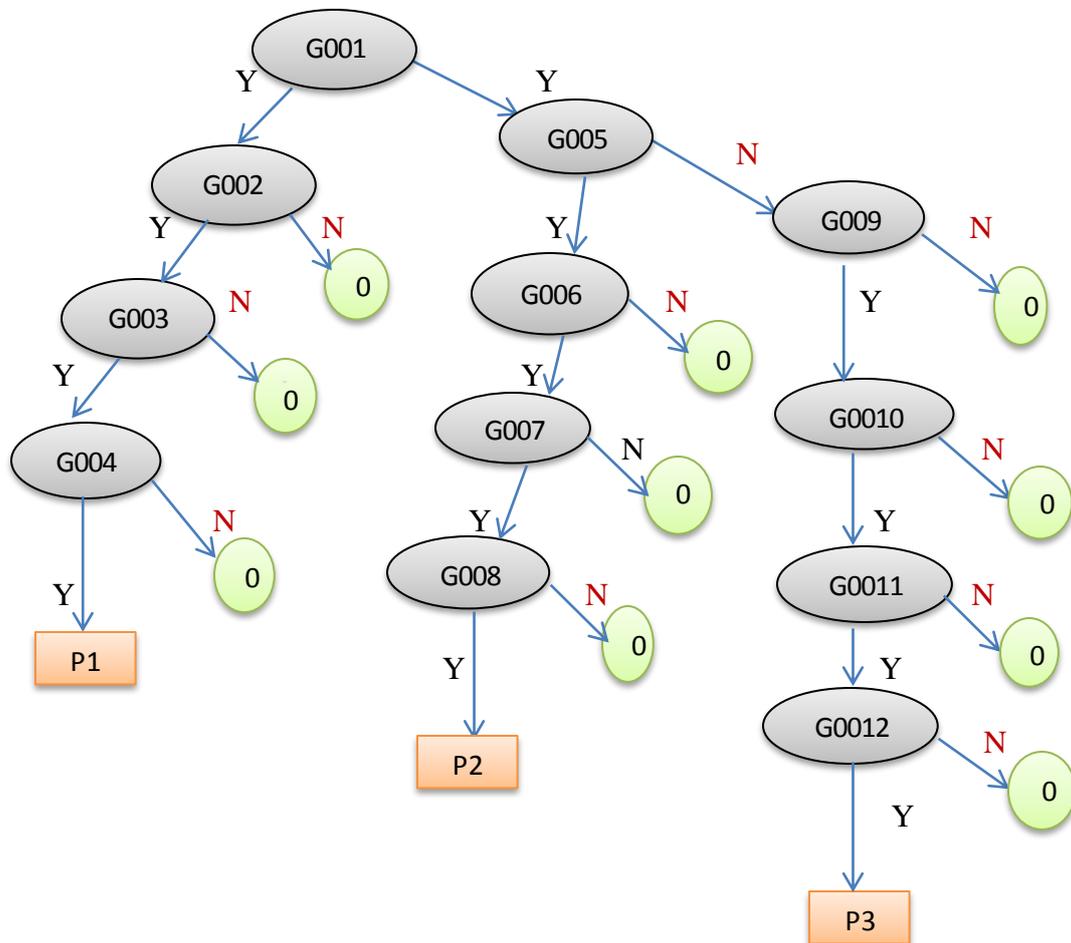
Tabel 3.4 Tabel Keputusan

Penyakit Gejala	P01	P02	P03
G001	✓	✓	
G002	✓		
G003	✓		
G004	✓		
G005		✓	
G006		✓	
G007		✓	
G008			✓
G009			✓
G0010			✓
G0011			✓

Sumber: Data Penelitian (2019)

Pada **Tabel 3.4** diatas, baris gejala diberi tanda centang untuk kolom kode penyakit yang memenuhi aturan dari masing-masing gejala. Hal ini dibuat untuk mempermudah dalam menyusun aturan kaidah produksi sistem pakar yang akan

dibuat. Setelah disusun tabel keputusan (Tabel 3.4) diatas maka dapat dibuat pohon keputusan berikut ini:



Gambar 3.4. Pohon Keputusan

Menurut Suparman (2017:74) pohon pelacakan adalah cabang satu *node* menuju ke *node successor* yang mewakili dua jalan alternatif menuju satu atau lebih tujuan antara salah satu jalan atau lainnya menuju tujuan akhir (*goal*). Sebagai *node OR* karena cabang *OR* lainnya merupakan jalan menuju tujuan.

Penjelasan dari Gambar 3.4 adalah sebagai berikut:

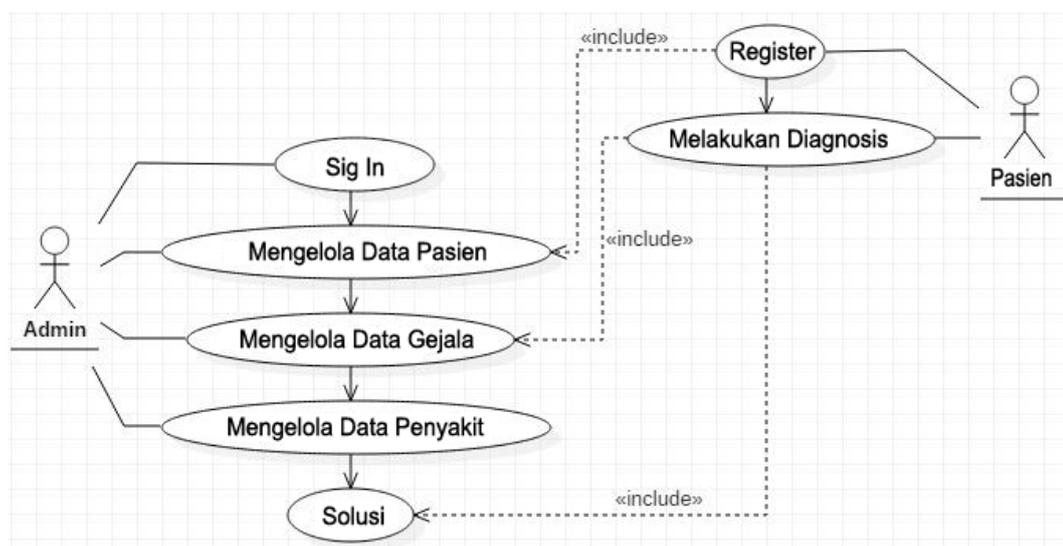
1. G001, G002, G003, G004 gejala penyakit, dimana G001 sampai G004 terdapat di penyakit satu (P1).
2. G005, G006, G007, G008 gejala penyakit, dimana G005 sampai G008 terdapat di penyakit dua (P2).
3. G009, G0010, G0011, G0012 gejala penyakit, dimana G009 sampai G0012 terdapat di penyakit (P3).
4. Y sebagai proses berjalannya sebuah pohon keputusan.
5. N berhenti. Proses kerja pohon keputusan selesai.

3.5. Perancangan Sistem

Sebagaimana proses pembuatan sebuah perangkat lunak, pembuatan sebuah sistem informasi juga memerlukan beberapa buah tahapan. Tahapan-tahapan tersebut dimulai dari desain, perancangan, hingga implementasi dan pengujian. Pembuatan sebuah sistem informasi diawali dengan kebutuhan pengguna berdasarkan permasalahan yang terjadi. Berdasarkan sudut pandang keilmuan informatika, langkah-langkah tersebut termasuk dalam kajian penelitian. *UML (Unified Modelling Language)* adalah standarisasi internasional untuk notasi dalam bentuk grafik, yang menjelaskan tentang analisis dan desain perangkat lunak yang dikembangkan dengan pemrograman berorientasi objek (Pratama, 2014).

3.5.1. Use Case Diagram

Use Case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sistem dan merepresentasikan interaksi antara aktor dengan sistem.



Gambar 3.5. *Use Case Diagram*
(Sumber: Pengolahan Data Penelitian 2019)

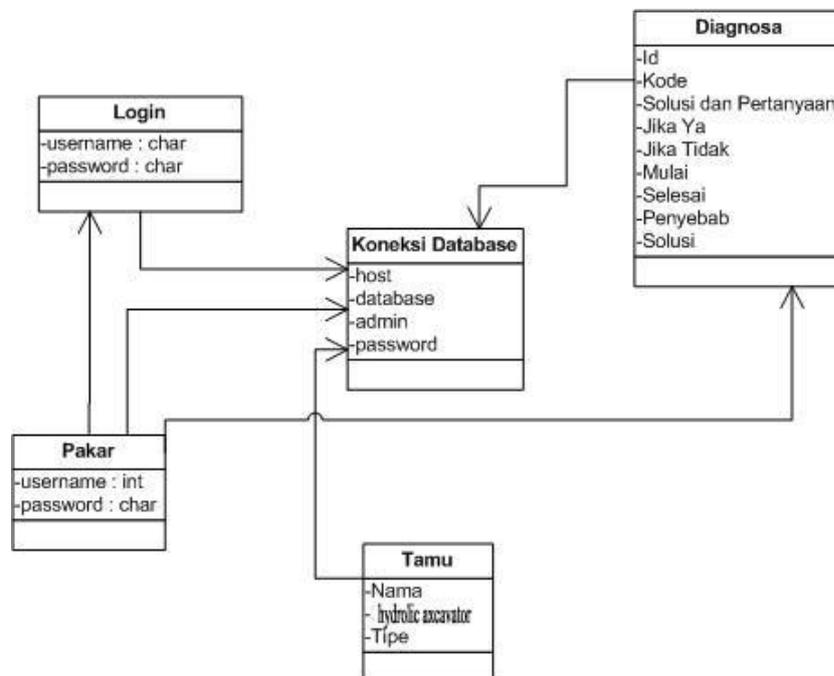
Penjelasan dari Gambar 3.5 *Use Case Diagram*, aplikasi sistem pakar sebagai berikut:

- 1) Pakar adalah aktor yang menjadi sumber pengetahuan penyakit kulit wajah, dan juga menjadi *admin* dari sistem aplikasi yang dibangun.
- 2) Pasien adalah aktor yang menggunakan aplikasi sistem pakar.
- 3) *Login* adalah pintu masuk untuk pakar sebagai *administrator*.
- 4) Isi data adalah pendaftaran pengguna untuk menjadi pasien dari aplikasi sistem pakar.

- 5) *Logout* adalah pintu keluar untuk pakar sebagai *administrator*
- 6) Mengelola Diagnosa adalah pengetahuan pakar mendiagnosis penyakit kulit wajah.
- 7) Mengelola Gejala adalah gejala-gejala dari setiap jenis penyakit kulit wajah.
- 8) Mengelola Solusi adalah solusi atas penyakit kulit wajah.
- 9) Melakukan Diagnosa adalah pengguna menggunakan fungsi diagnosa penyakit kulit wajah.
- 10) Mendapatkan Solusi adalah hasil dari diagnosa penyakit kulit wajah.

3.5.2. Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat atribut dan membangun sistem.



Gambar 3.6 *Class Diagram*
(Sumber: Pengolahan Data Penelitian 2019)

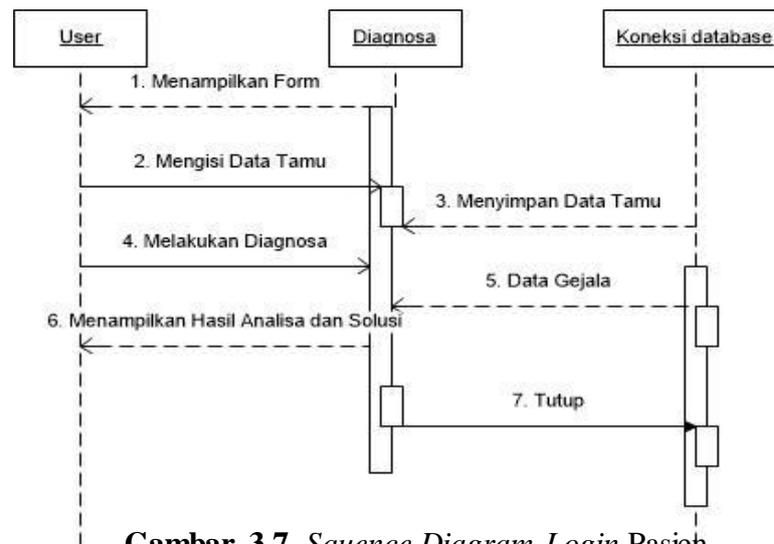
Penjelasan dari Gambar 3.6 *Class diagram*, aplikasi sistem pakar sebagai berikut:

- 1) *Login* adalah kelas proses yang diambil dari pendefinisian *use case login* untuk pintu masuk pakar *administrator* ke sistem aplikasi pakar.
- 2) *Tamu* adalah kelas proses yang diambil dari pendefinisian *use case* untuk aktor yang menggunakan aplikasi sistem pakar.
- 3) *Kelola Diagnosa* adalah kelas proses yang diambil dari pendefinisian *use case* mengelola diagnosa yang didalamnya menangani proses gejala dan proses solusi.
- 4) *Kelola Gejala* adalah kelas proses yang diambil dari pendefinisian *use case* mengelola gejala penyakit kulit wajah yang didalamnya menangani proses pertanyaan gejala untuk setiap jenis penyakit kulit wajah.
- 5) *Kelola Solusi* adalah kelas proses yang diambil dari pendefinisian *use case* mengelola solusi yang didalamnya menangani proses kesimpulan atas proses mengelola diagnosa dan proses mengelola gejala penyakit kulit wajah.
- 6) *Pakar* adalah kelas data yang digunakan untuk memproses segala pengaksesan terhadap proses mengelola diagnosa, proses mengelola gejala penyakit kulit wajah, dan proses mengelola solusi.
- 7) *Koneksi database* adalah kelas utilitas untuk koneksi ke *database*.

3.5.3. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

1. *Sequence Diagram* Pasien



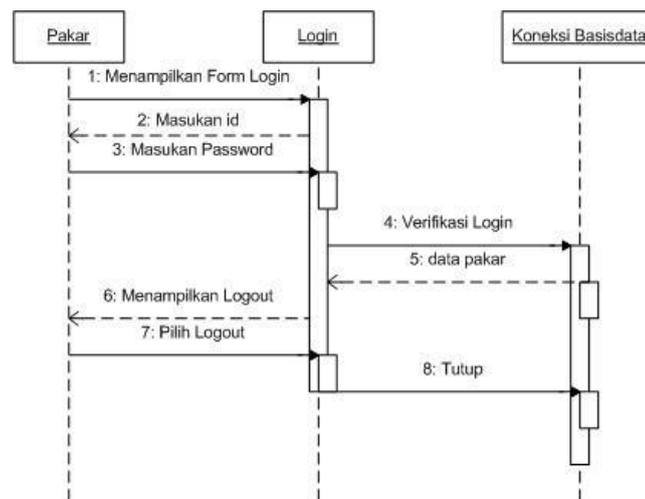
Gambar 3.7. *Sequence Diagram Login Pasien*
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.7. *Sequence Diagram Pasien*, aplikasi sistem pakar sebagai berikut:

- 1) Untuk melakukan diagnosa pasien akan mengklik menu diagnosa pada sistem aplikasi pakar dan seterusnya akan tampil form untuk di isi.
- 2) *Diagnosa* menampilkan *form* yang berisi nama, jenis penyakit dan *type*. Pasien akan mengisi kolom jenis penyakit dan *type* berupa angka atau huruf serta mengisi nama pengguna pada kolom nama.
- 3) Setelah tombol Mulai diklik, maka data Pasien akan disimpan ke dalam *database*.
- 4) Proses diagnosa akan dijalankan dengan menjawab pertanyaan.
- 5) Pertanyaan diagnosa akan muncul satu persatu.

- 6) Menampilkan hasil diagnosa berdasarkan jawaban pertanyaan.
- 7) Proses pasien selesai.

2. *Sequence Diagram Login Pakar*



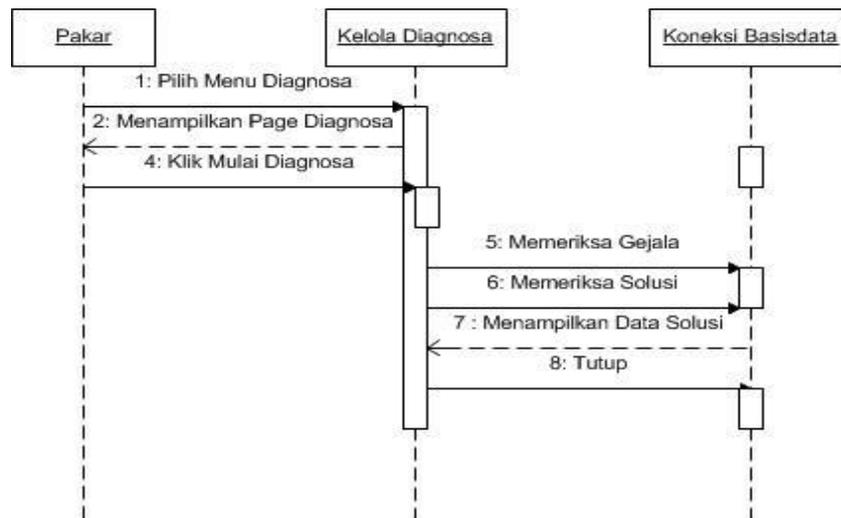
Gambar 3.8. *Sequence Diagram Login Pakar*
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.8 *Sequence Diagram Login Pakar*, aplikasi sistem pakar sebagai berikut:

- 1) Pakar memasukan *link* ke *address bar* di *browser* untuk masuk ke *form login* sistem aplikasi pakar.
- 2) *Login* menampilkan *form* yang berisi *pasienname* dan *password*.
- 3) Pakar memasukan *pasienname* yang berupa angka atau huruf.
- 4) Pakar memasukan *password* yang berupa angka atau huruf.
- 5) Setelah tombol *login* diklik, maka data Pakar diverifikasi ke dalam *database*.
- 6) Jika data *input* benar maka *login* pakar berhasil masuk ke dalam sistem.
- 7) Tombol *logout* tampil setelah berhasil *login*.

- 8) Jika ingin keluar dari sistem aplikasi maka silahkan klik tombol *logout*.
- 9) Proses *login* selesai.

3. *Sequence Diagram* Diagnosa

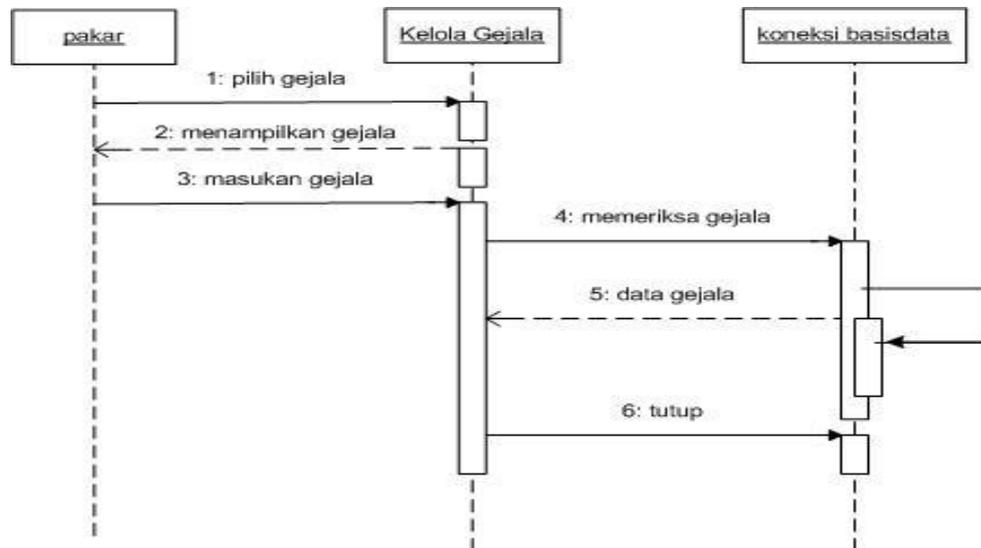


Gambar 3.9. *Sequence Diagram* Diagnosa
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.9. *Sequence Diagram* Diagnosa, aplikasi sistem pakar sebagai berikut:

- 1) Pakar memilih *menu* diagnosa untuk mendiagnosis penyakit kulit wajah.
- 2) Mengelola Diagnosa terkoneksi ke *database*.
- 3) Mengelola Diagnosa menampilkan halaman diagnosa.
- 4) Pakar memulai diagnosa.
- 5) Mengelola Diagnosa memeriksa gejala dengan mengajukan pertanyaan.
- 6) Mengelola Diagnosa memeriksa solusi berdasarkan jawaban atas pertanyaan gejala penyakit kulit wajah.
- 7) Koneksi basis data menampilkan solusi penyakit kulit wajah.
- 8) Proses Mengelola Diagnosa selesai.

4. *Sequence Diagram* Mengelola gejala penyakit kulit wajah

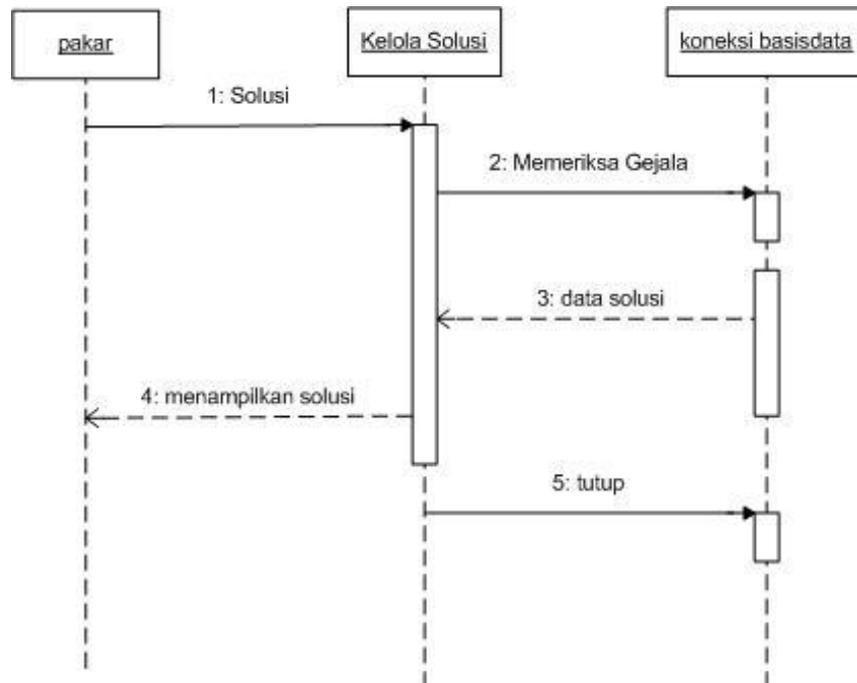


Gambar 3.10 *Sequence Diagram* Mengelola Gejala Penyakit Kulit Wajah
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.10 *Sequence Diagram* Mengelola gejala penyakit kulit wajah, aplikasi sistem pakar sebagai berikut:

- 1) Pakar menjawab/memilih gejala penyakit kulit wajah.
- 2) Mengelola gejala penyakit kulit wajah menampilkan pertanyaan gejala sesuai jenis penyakit kulit wajah.
- 3) Pakar memasukan/memilih gejala penyakit kulit wajah.
- 4) Mengelola gejala penyakit kulit wajah memeriksa setiap jawaban dari gejala terpilih.
- 5) Mengelola gejala penyakit kulit wajah memproses dan menampilkan gejala.
- 6) Proses Mengelola gejala penyakit kulit wajah selesai.

5. *Sequence Diagram* Mengelola Solusi



Gambar 3.11 *Sequence Diagram* Mengelola Solusi
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.11 *Sequence Diagram* Mengelola Solusi, sebagai berikut:

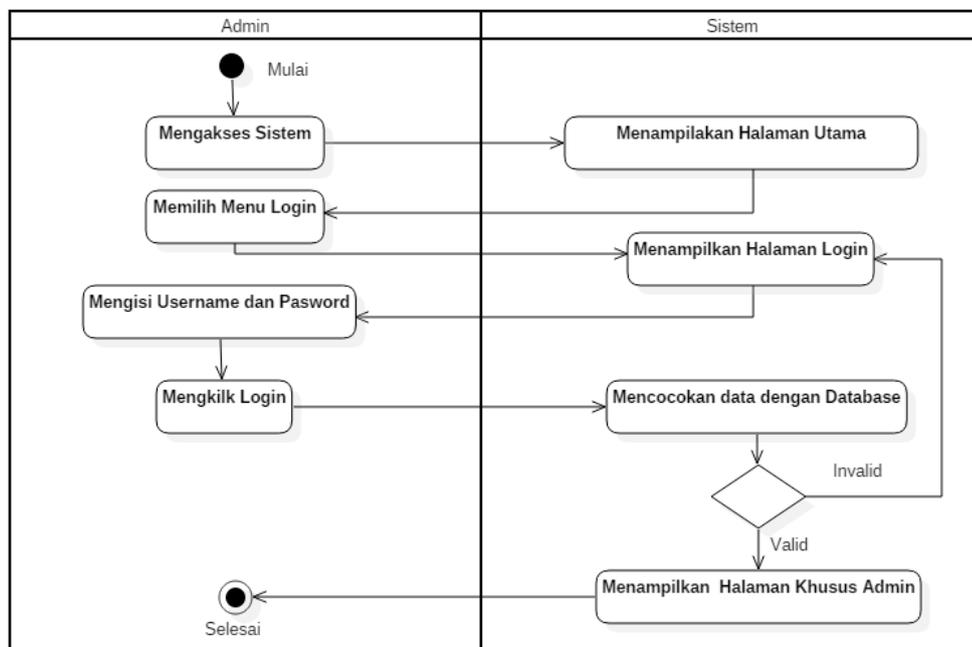
- 1) Pakar mengikuti proses untuk mendapatkan solusi.
- 2) Mengelola Solusi memilih gejala yang sudah dipilih pakar.
- 3) Mengelola Solusi memeriksa hasil pertanyaan gejala sudah dipilih.
- 4) Mengelola Solusi memproses analisa dari *database*.
- 5) Mengelola Solusi menampilkan solusi penyakit kulit wajah.
- 6) Proses Mengelola Solusi selesai.

3.5.4 Activity Diagram

Berikut ini adalah *activity diagram* yang dirancang dalam penelitian ini:

1. Activity Diagram Login Admin

Activity diagram Login admin merupakan *UML* yang menggambarkan kegiatan pengguna pada halaman khusus *admin*.



Gambar 3.12. Activity Diagram Login Admin
(Sumber: Pengolahan Data Penelitian 2019)

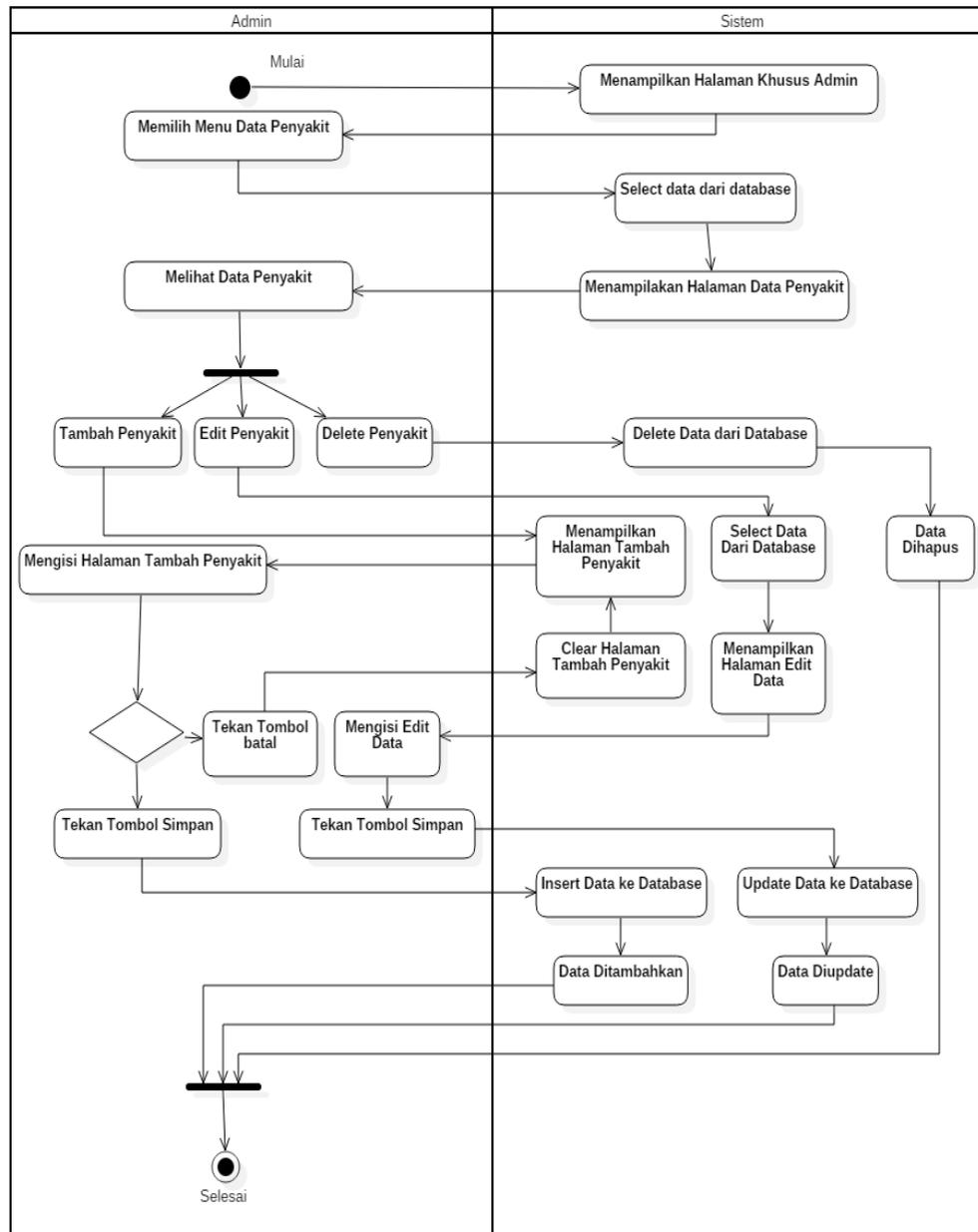
Penjelasan dari Gambar 3.12. *activity diagram login admin*, sebagai berikut:

- 1) Admin mengakses sistem, kemudian sistem akan melakukan halaman utama.
- 2) Kemudian admin akan memilih menu admin dan sistem akan menampilkan halaman khusus admin.

- 3) Admin akan mengisi username dan password pada menu Login, kemudian klik tombol Login.

2. *Activity diagram* Mengelola Menu Penyakit

Activity diagram mengelola menu penyakit merupakan kegiatan *admin* dalam mengelola data jenis penyakit dan solusi yang digunakan.



Gambar 3.13: Activity Diagram Mengelola Menu Penyakit

(Sumber: Pengolahan Data Penelitian 2019)

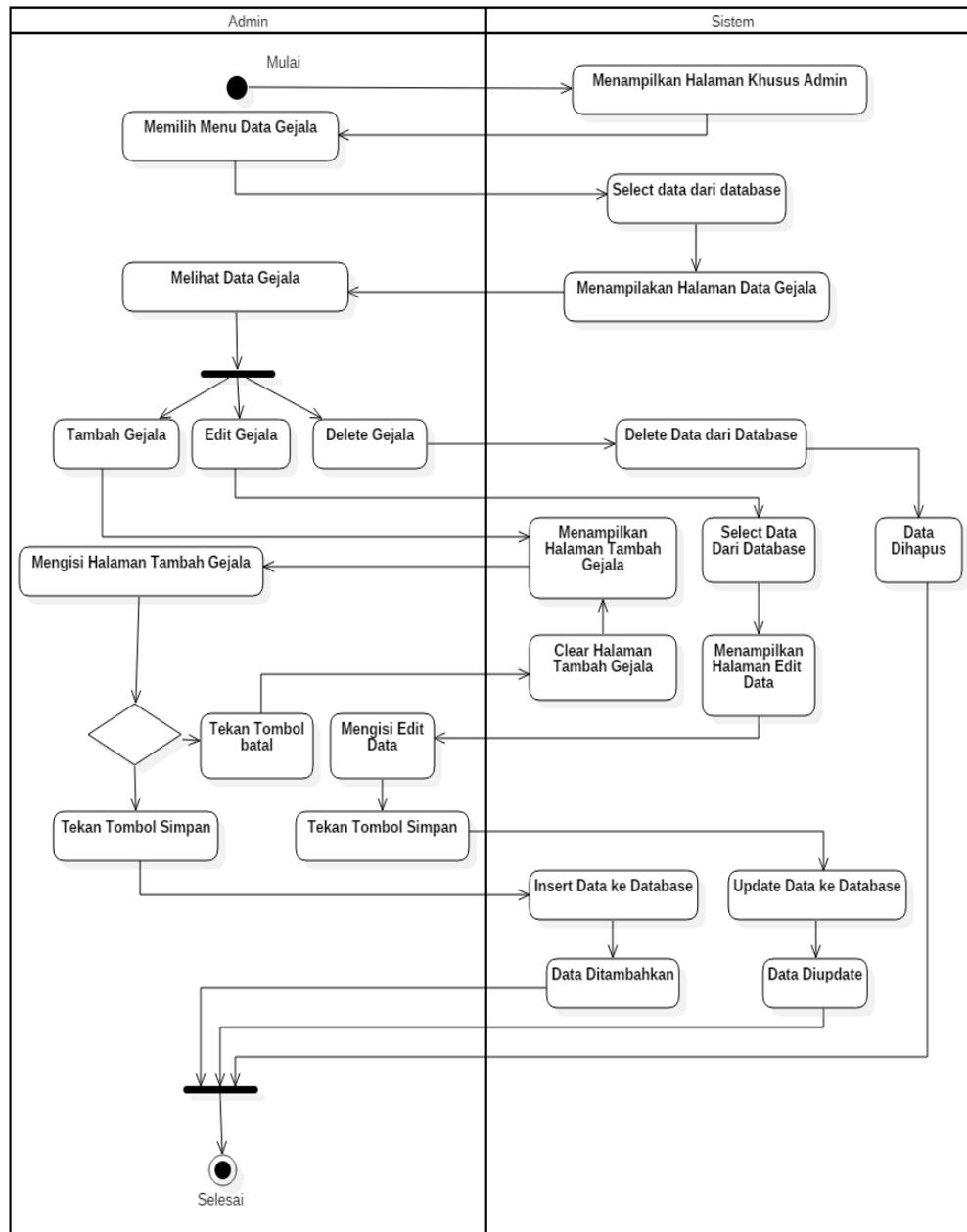
Penjelasan dari Gambar 3.13. activity diagram mengelola menu penyakit sebagai berikut:

- 1) Halaman khusus *admin* yang terbuka setelah *admin* melakukan *Login*.

- 2) Sistem akan menampilkan menu-menu pada halaman khusus *admin*, kemudian *admin* memilih menu penyakit.
- 3) Sistem memanggil data dari database dan menampilkan halaman data penyakit.
- 4) *Admin* melihat 3 pilihan yaitu tambah, *edit*, dan *delete*.
- 5) Jika *admin* mengklik tombol tambah maka sistem akan menampilkan halaman sistem penyakit.
- 6) Kemudian *admin* mengisi data penyakit dan solusi.
- 7) Kemudian memilih tombol simpan maka data akan dimasukkan ke *database* kemudian data ditambahkan di *database*, jika *admin* menekan tombol batal maka sistem akan membersihkan halaman tambah penyakit.
- 8) Jika *admin* memilih tombol *edit*, maka sistem akan mengambil data dari *database*.
- 9) *Admin* mengedit data kemudian klik tombol simpan. Sistem akan melakukan *update database*.
- 10) Jika *admin* menekan tombol *delete* maka data yang ada di *database* akan terhapus maka proses selesai.

3. Activity Diagram Mengelola Gejala

Activity diagram mengelola menu gejala merupakan kegiatan *admin* dalam mengelola gejala penyakit yang digunakan.



Gambar 3.14. Activity Diagram Mengelola Gejala
(Sumber: Pengolahan Data Penelitian 2019)

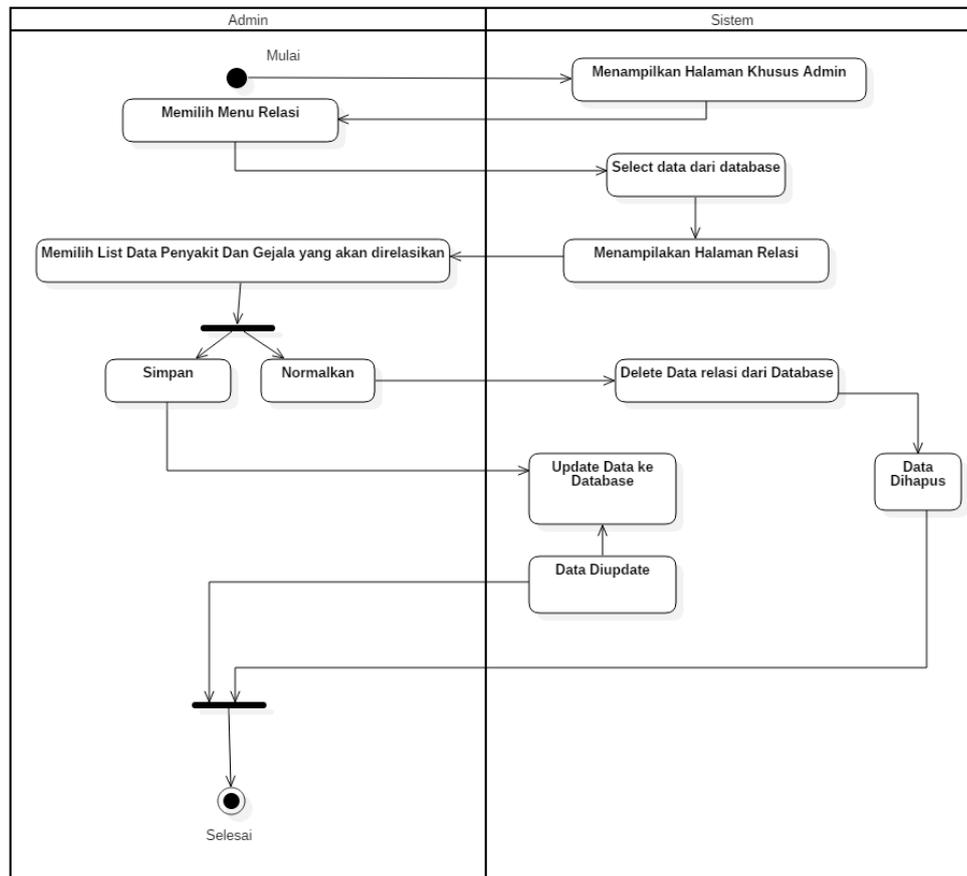
Penjelasan dari Gambar 3.14. *activity diagram* mengelola gejala sebagai berikut:

- 1) Halaman khusus *admin* yang terbuka setelah *admin* melakukan *Login*.

- 2) Sistem akan menampilkan menu-menu pada halaman khusus *admin*, kemudian *admin* memilih menu gejala.
- 3) Sistem memanggil data dari database dan menampilkan halaman data gejala.
- 4) Halaman tambah gejala, kemudian admin mengisi data gejala.
- 5) Kemudian memilih tombol simpan maka data akan dimasukkan ke *database* kemudian data ditambahkan di *database*.
- 6) Jika *admin* menekan tombol batal maka sistem akan membersihkan halaman tambah gejala.
- 7) Jika *admin* memilih tombol *edit*, maka sistem akan mengambil data dari *database*.
- 8) *Admin* mengedit data kemudian klik tombol simpan.
- 9) Sistem akan melakukan *update database*.
- 10) Jika *admin* menekan tombol *delete* maka data yang ada di *database* akan terhapus maka proses selesai.

4. Activity Diagram Mengelola Menu Relasi

Activity diagram mengelola menu relasi merupakan diagram *UML* yang menggambarkan kegiatan *admin* dalam mengelola data relasi sesuai dengan data aturan penelitian ini.



Gambar 3.15. Activity Diagram Mengelola Menu Relasi
(Sumber: Pengolahan Data Penelitian 2019)

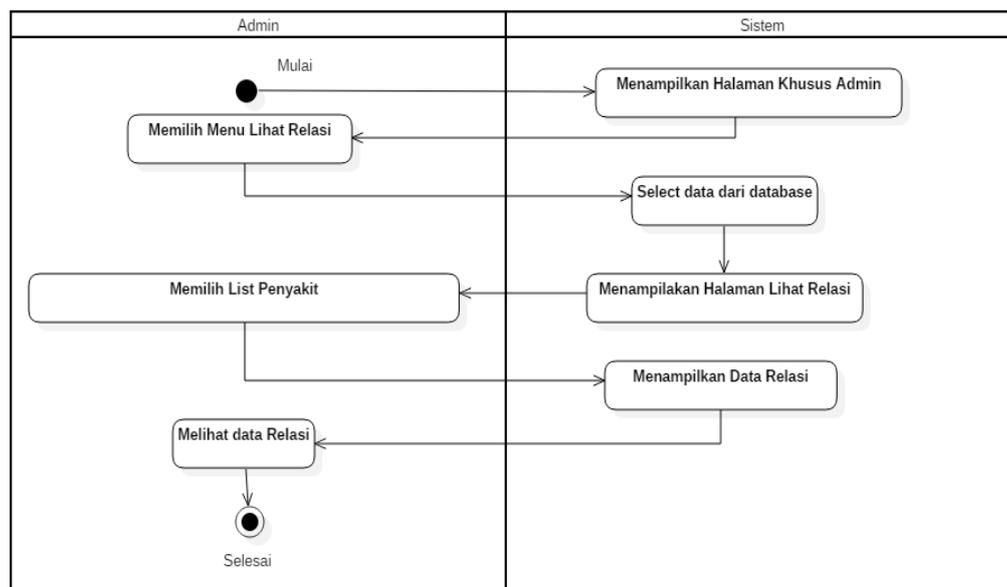
Penjelasan dari Gambar 3.15. *activity diagram* mengelola menu relasi sebagai berikut:

- 1) *Admin* mulai mengakses sistem dan sistem menampilkan halaman khusus admin.
- 2) *Admin* memilih menu relasi kemudian sistem menampilkan halaman menu relasi.
- 3) *Admin* dapat memilih jenis penyakit dan gejala yang akan direlasikan.
- 4) Jika *admin* menekan tombol simpan maka sistem akan mengupdate data relasi ke *database*.

- 5) Jika *admin* menekan tombol normalkan , maka sistem akan menghapus data relasi antar data penyakit dan gejala dan data relasi akan dihapus dari *database* maka proses selesai.

5. Activity Diagram Mengelola Menu Lihat Relasi

Activity diagram mengelola menu lihat relasi merupakan diagram *UML* yang menggambarkan kegiatan *admin* dalam data relasi atau aturan dari *rule forward chaining*.



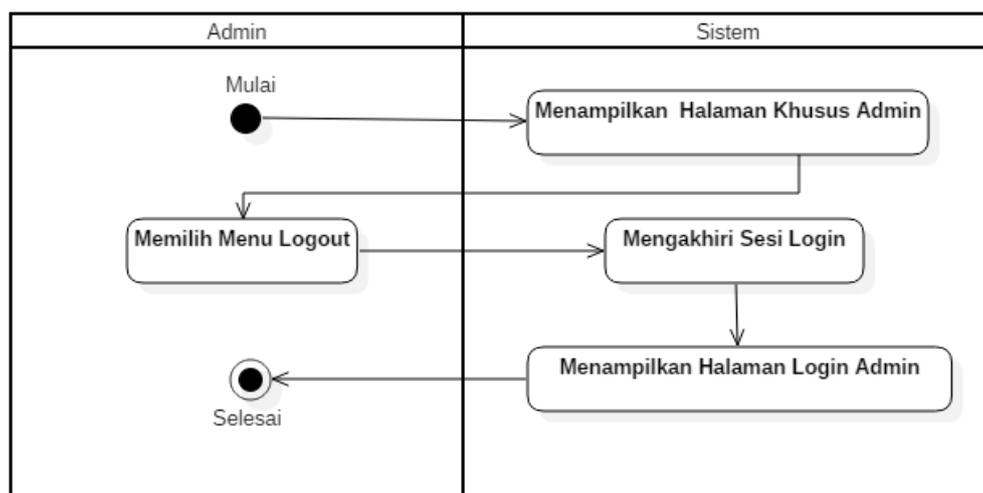
Gambar 3.16. Activity Diagram Mengelola Menu Lihat Relasi
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.16. activity diagram mengelola menu lihat relasi sebagai berikut:

- 1) *Admin* mulai dengan mengakses sistem dan sistem menampilkan halaman khusus admin.
- 2) *Admin* memilih menu lihat relasi kemudian sistem akan mengambil data dari *database* dan menampilkan halaman lihat relasi.
- 3) *Admin* dapat melihat list penyakit yang akan dilihat relasinya, maka sistem akan menampilkan halaman data relasi yang telah dibuat.
- 4) *Admin* akan melihat data relasi apakah telah sesuai dengan aturan penelitian ini, maka proses selesai.

6. Activity diagram menu *logout*

Activity diagram menu *logout* merupakan diagram *UML* yang menggambarkan kegiatan admin dalam menggunakan menu *logout*.



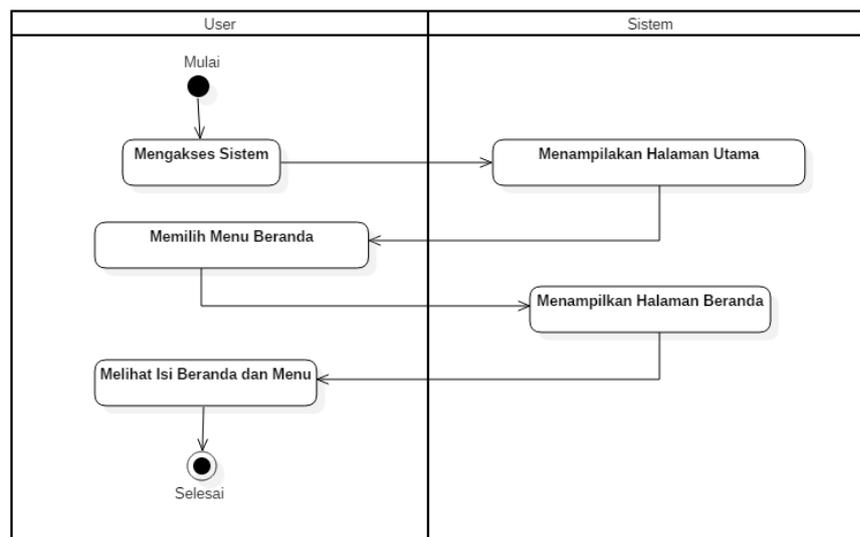
Gambar 3.17. Activity Diagram *Logout*
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.17. *activity diagram logout* sebagai berikut:

- 1) *Admin* memilih menu *logout* kemudian sistem akan mengakhiri sesi *Login admin* dan sistem akan menampilkan halaman menu *Login*.
- 2) Proses selesai.

7. Activity Diagram Melihat Menu Beranda

Activity diagram melihat menu beranda merupakan diagram *UML* yang menggambarkan kegiatan *user* melihat menu beranda.



Gambar 3.18. Activity Diagram Melihat Menu Beranda
(Sumber: Pengolahan Data Penelitian 2019)

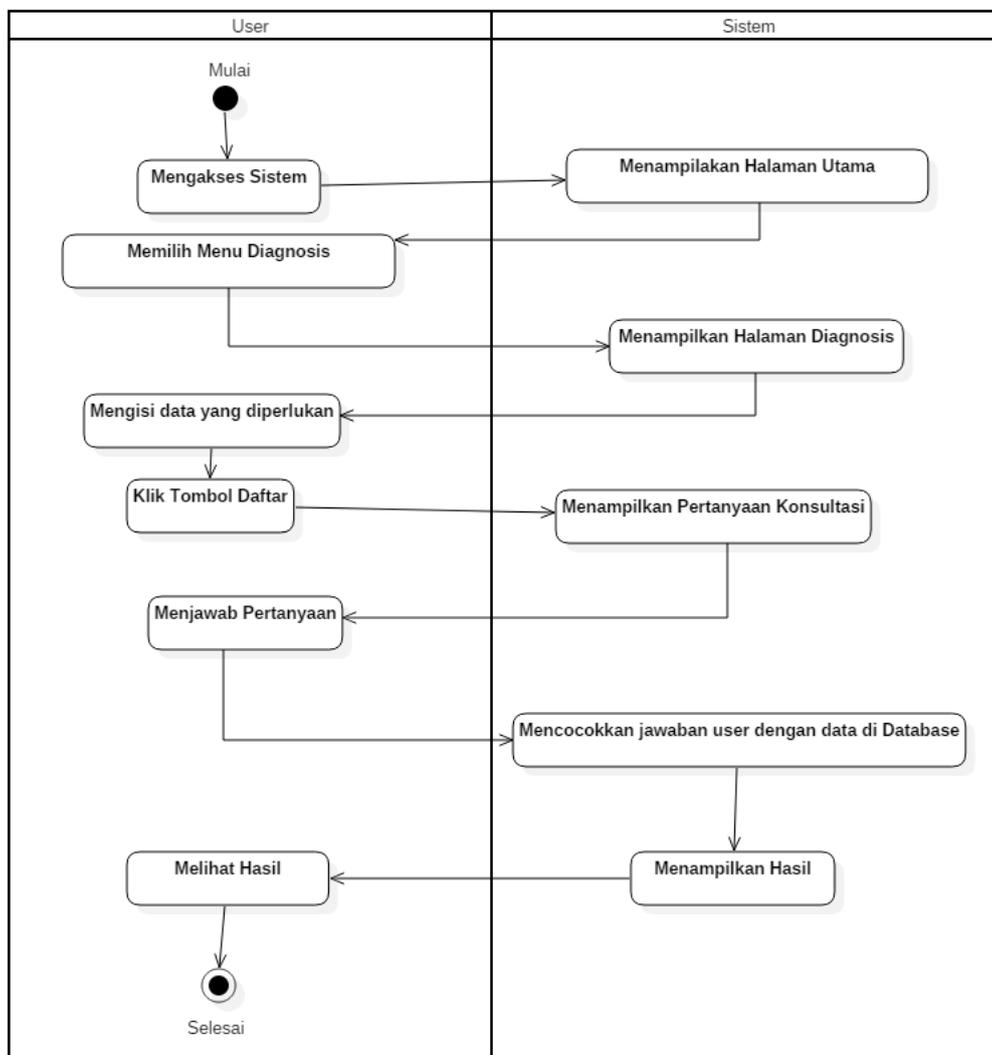
Penjelasan dari Gambar 3.18. *activity diagram* melihat menu beranda sebagai berikut:

- 1) *User* mulai dengan mengakses sistem dan sistem menampilkan halaman utama.
- 2) *User* memilih menu beranda kemudian sistem menampilkan halaman beranda.

- 3) *User* dapat melihat info mengenai penyakit kulit wajah yang bermanfaat di sistem maka proses melihat menu beranda selesai.

8. Activity diagram Menu Diagnosis

Activity diagram menu *diagnosis* merupakan diagram *UML* yang menggambarkan kegiatan *user* melakukan *diagnosis* penyakit kulit wajah.



Gambar 3.19. Activity Diagram Menu Diagnosis
(Sumber: Pengolahan Data Penelitian 2019)

Penjelasan dari Gambar 3.19. *activity diagram* menu *diagnosis* sebagai berikut:

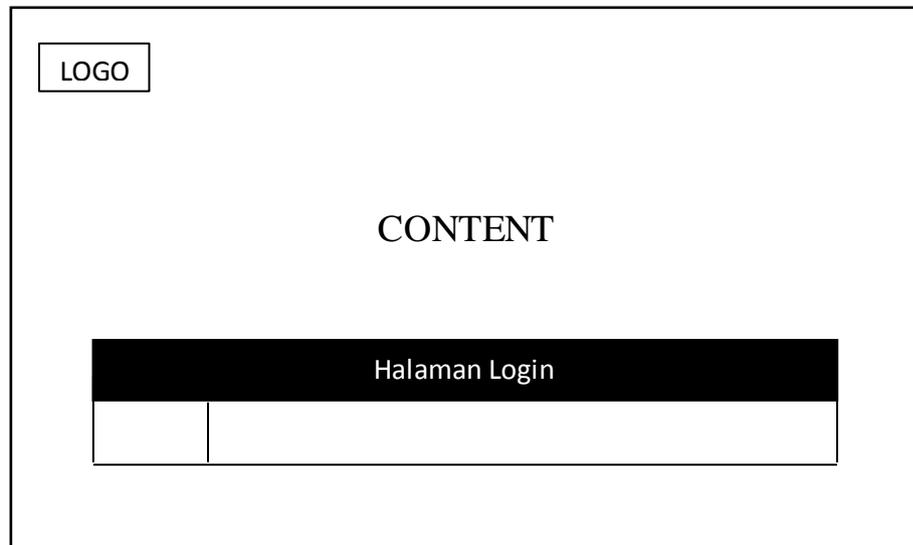
- 1) *User* mulai dengan mengakses sistem dan sistem menampilkan halaman utama.
- 2) *User* memilih menu *diagnosis* kemudian sistem menampilkan halaman menu *diagnosis*.
- 3) *User* akan mengisi data terlebih dahulu dan menekan tombol daftar, sistem akan menampilkan halaman pertanyaan *diagnosis*.
- 4) *User* akan menjawab pertanyaan sesuai dengan apa yang terjadi pada pasien.
- 5) Kemudian sistem akan mencocokkan jawaban dengan data yang ada didatabase lalu sistem akan menampilkan hasil atau solusinya.
- 6) *User* dapat melihat solusi dari gejala yang terjadi pada orang dewasa penyakit kulit wajah.
- 7) Maka proses melakukan *diagnosis* selesai.

3.5.5. Desain Antarmuka

Berikut ini adalah desain antarmuka yang akan dibuat pada aplikasi sistem pakar mendiagnosa penyakit kulit wajah:

1. Halaman Utama

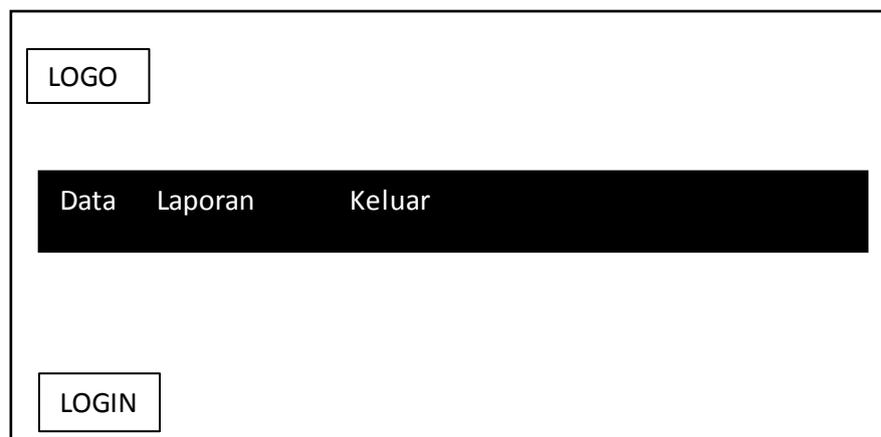
Halaman rancangan awal menampilkan halaman pengenalan sistem pakar yang telah dibuat oleh peneliti untuk mendiagnosa penyakit kulit wajah.



Gambar 3.20. Halaman Utama
(Sumber: Pengolahan Data Penelitian 2019)

2. Halaman Beranda *Admin*

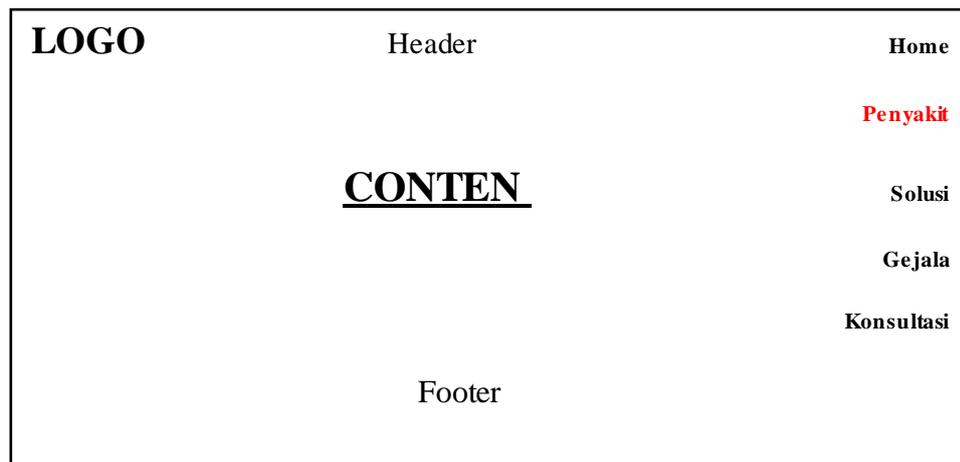
Halaman beranda admin, berisi beberapa menu yang ada di sistem pakar untuk mendiagnosa penyakit kulit wajah.



Gambar 3.21. Halaman Beranda *Admin*
(Sumber: Penolahan Data Penelitian 2019)

3. Halaman Data Penyakit

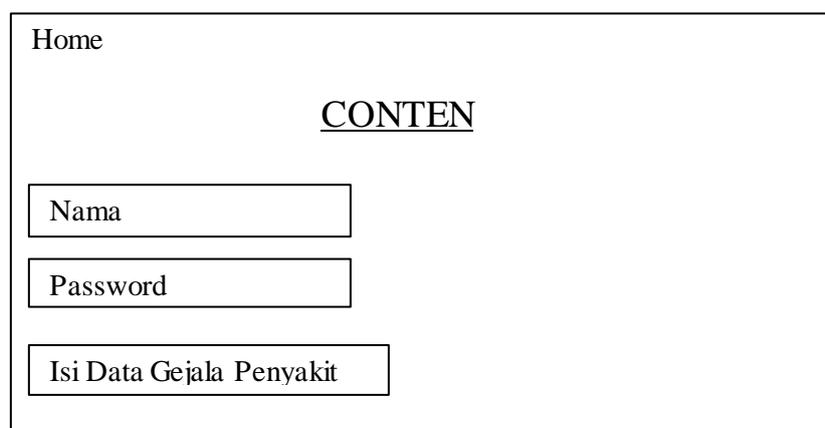
Sebelum user mengisi data penyakit, user diharuskan untuk mengisi form login. Setelah user mengisi nama dan password , klik tombol lanjut mengisi data penyakit.



Gambar 3.22. Halaman Data Penyakit
(Sumber: Pengolahan Data Penelitian 2019)

4. Halaman Data Gejala Penyakit

Halaman data penyakit ini digunakan oleh pasien atau user untuk melakukan pengisian data penyakit.



Gambar 3.23. Halaman Data Gejala Penyakit
(Sumber: Pengolahan Data Penelitian 2019)

5. Halaman Data Solusi

Sebelum mengisi data solusi, pasien diwajibkan mengisi data penyakit untuk mengetahui gejalanya.

Home Isi Data Penyakit Ulang

Solusi Penyakit

Ya Tidak

Keterangan

Lanjut

Gambar 3.24. Halaman Data Solusi
(Sumber: Pengolahan Data Penelitian 2019)

6. Halaman Rule

Halaman rule ini memungkinkan pasien untuk melihat penyakit, gejala, dan solusi.

Home Isi Data Penyakit Ulang

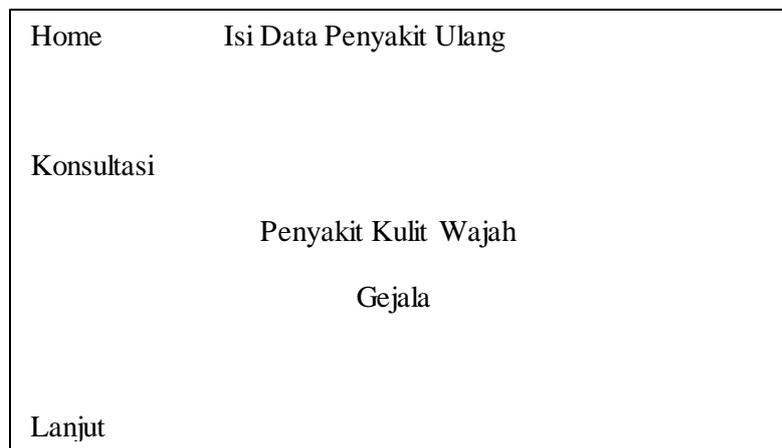
Nama Penyakit

Solusi

Gambar 3.25. Halaman *Rule*
(Sumber: Pengolahan Data Penelitian 2019)

7. Halaman Pasien

Halaman pasien tempat untuk konsultasi, dimana pasien bisa melihat gejala dari penyakit yang dialami.

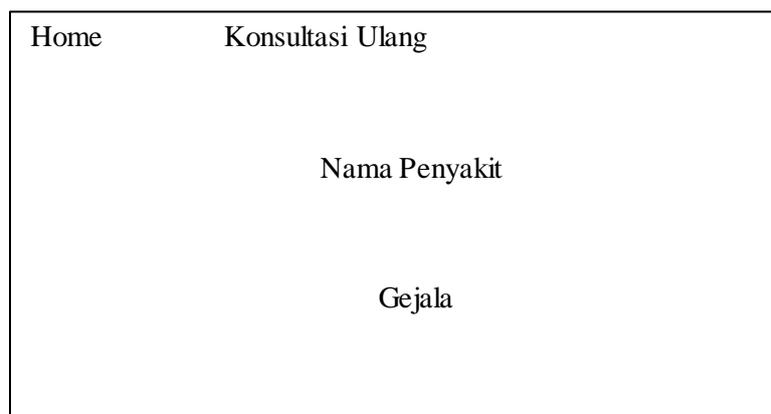


Home	Isi Data Penyakit Ulang
Konsultasi	
	Penyakit Kulit Wajah
	Gejala
Lanjut	

Gambar 3.26. Halaman Pasien
(Sumber: Pengolahan Data Penelitian 2019)

8. Halaman Form Konsultasi

Sebelum lanjut ke halaman form konsultasi pasien diwajibkan mengisi data gejala untuk mengetahui penyakit yang di alami.



Home	Konsultasi Ulang
	Nama Penyakit
	Gejala

Gambar 3.27. Halaman *Form Konsultasi*
(Sumber: Pengolahan Data Penelitian 2019)

3.6. Lokasi Dan Jadwal Penelitian

3.6.1. Lokasi Penelitian

Penelitian ini dilakukan di tempat Praktek dr. Amelia Fenowati di Klinik OASE BEUTY dan SKINCARE Kota Batam. Adapun jenis penelitiannya adalah deskriptif yaitu penelitian yang menggambarkan secara apa adanya atau fakta yang terdapat pada obyek penelitian.

3.6.2. Jadwal Penelitian

Adapun jadwal peneliti dalam melakukan penelitian tersebut adalah seperti tabel dibawah ini:

Tabel 3.5. Jadwal Penelitian

No	Kegiatan	2019													
		Maret			April			Mei			Juni			Juli	
1	Bimbingan dengan dosen pembimbing														
2	Input Judul														
3	Bab I														
4	Bab II														
5	Bab III														
6	Bab IV														
7	Bab V														

Sumber: Data Penelitian (2019)