

**RANCANG BANGUN SISTEM PAKAR DIAGNOSA PENYAKIT
GIGI DAN MULUT MENGGUNAKAN METODE *FORWARD
CHAINING* DAN *NAIVE BAYES* BERBASIS WEB**

SKRIPSI

Disusun untuk melengkapi syarat-syarat guna memperoleh gelar
Sarjana Komputer



**Disusun oleh:
HANA MAULINDA
3145136193**

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA**

2017

LEMBAR PERSETUJUAN HASIL SIDANG SKRIPSI

Rancang Bangun Sistem Pakar Diagnosa Penyakit Gigi dan Mulut Menggunakan Metode *Forward Chaining* dan *Naive Bayes* Berbasis Web

Nama : Hana Maulinda

No. Registrasi : 3145136193

	Nama	Tanda Tangan	Tanggal
Penanggung Jawab			
Dekan	: Prof. Dr. Suyono, M.Si. NIP. 19671218 199303 1 005
Wakil Penanggung Jawab			
Wakil Dekan I	: Dr. Muktiningsih, M.Si. NIP. 19640511 198903 2 001
Ketua	: Ir. Fariani Hermin I, M.T. NIP. 19600211 198703 2 001
Sekretaris	: Med Irzal, M.Kom. NIP. 19770615 200312 1 001
Penguji	: Ratna Widyati, S.Si, M.Kom. NIP. 19750925 200212 2 002
Pembimbing I	: Ria Arafyah, S.Si, M.Si NIP. 19751121 200501 2 004
Pembimbing II	: Drs. Mulyono, M.Kom. NIP. 19660517 199403 1 003

Dinyatakan lulus ujian skripsi tanggal: 18 Agustus 2017

HALAMAN PERSEMBAHAN

*Untuk Mama, Papa, Bestian,
dan sahabat-sahabatku tercinta.*

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Puji syukur kepada Allah SWT karena atas limpahan rahmat dan karunia-Nya lah penulis dapat menyelesaikan Tugas Akhir ini yang merupakan salah satu syarat dalam memperoleh gelar Sarjana Program Studi Ilmu Komputer Universitas Negeri Jakarta.

Tugas akhir ini dapat diselesaikan tidak terlepas dari adanya bantuan dari berbagai pihak. Oleh karena itu, dalam kesempatan ini dengan kerendahan hati penulis ingin menyampaikan terima kasih terutama kepada:

1. Ibu Ria Arafiyah, M.Si selaku Dosen Pembimbing I dan Bapak Drs. Mulyono, M.Kom., selaku Dosen Pembimbing II, yang telah meluangkan waktunya dalam memberikan bimbingan, saran, nasehat serta arahan sehingga Tugas Akhir ini dapat menjadi lebih baik dan terarah.
2. Bapak Drs. Mulyono, M.Kom., selaku Koordinator Prodi Ilmu Komputer FMIPA UNJ dan selaku Pembimbing Akademik atas segala bimbingan dan kerja sama Bapak selama perkuliahan, dan seluruh Bapak/Ibu dosen atas pengajarannya yang telah diberikan, serta karyawan/karyawati FMIPA UNJ yang telah memberikan informasi yang penulis butuhkan dalam menyelesaikan Tugas Akhir.
3. Papa dan Mama yang selalu mendukung, memberi motivasi, dan setia membantu penulis dengan penuh cinta dan kasih sayang yang tulus.
4. Bestian Fangaro yang telah membantu dan menginspirasi penulis dari awal penelitian hingga akhir dan selalu ada dan siap mendampingi dalam setiap keadaan apapun.

5. Sahabat tercinta yang tak henti-hentinya memberikan semangat serta bantuan doa yang besar kepada penulis.
6. Teman-teman Prodi Ilmu Komputer 2013 yang telah membantu dan mendukung, sehingga Tugas Akhir ini dapat selesai tepat waktu.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna, masih banyak kekurangan baik dari segi penyajian, pengkajian materi, bahasa maupun tata cara penulisan, karenanya penulis dengan lapang hati menerima kritik dan saran yang berifat membangun dari semua pihak sehingga Tugas Akhir ini dapat menjadi lebih baik lagi.

Akhir kata penulis berharap Tugas Akhir ini dapat berguna dan memberikan kontribusi dalam hal sumber bacaan sehingga dapat bermanfaat bagi lingkungan akademik di mana penulis selama ini menuntut ilmu, maupun pihak lain yang membutuhkan.

Wassalamu'alaikum Wr. Wb.

Jakarta, 18 Agustus 2017

Penulis

DAFTAR ISI

HALAMAN PERSEMBAHAN	iii
KATA PENGANTAR	iv
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xiii
ABSTRAK	xiv
<i>ABSTRACT</i>	xv
I PENDAHULUAN	1
A. Latar Belakang Masalah	1
B. Pembatasan Masalah	4
C. Perumusan Masalah	5
D. Tujuan Penelitian	5
E. Manfaat Penelitian	5
F. Jenis Penelitian	6
II LANDASAN TEORI	7
A. Sistem Pakar	7
1. Konsep Umum Sistem Pakar	8
2. Struktur dan Arsitektur Sistem Pakar	9
B. Probabilitas dan Teorema <i>Bayes</i>	14
1. Metode <i>Naive Bayes</i>	16

2.	Contoh Perhitungan <i>Naive Bayes</i>	17
C.	Gigi dan Mulut	18
1.	Penyakit Gigi dan Mulut	20
D.	Alat Perancangan Sistem	24
1.	<i>Unified Modelling Language (UML)</i>	24
2.	<i>Data Flow Diagram (DFD)</i>	26
3.	<i>Entity Relationship Diagram (ERD)</i>	27
4.	Struktur Navigasi	28
E.	Metode <i>Waterfall</i>	30
F.	<i>Framework CodeIgniter</i>	32
G.	<i>Web Server XAMPP</i>	34
H.	<i>Database MySQL</i>	35
III DESAIN MODEL		36
A.	Tahapan Penelitian	36
B.	Pengumpulan Data	37
C.	Analisa Sistem	37
1.	Akuisisi Pengetahuan	38
2.	Basis Pengetahuan	39
3.	Mesin Inferensi	52
4.	Representasi Pengetahuan	54
5.	Analisis <i>Naive Bayes</i>	59
6.	Analisis Fungsional	61
D.	Perancangan Sistem	66
1.	Perancangan <i>Database</i>	66
2.	Perancangan Struktur Navigasi	71
3.	Perancangan Antarmuka	72

IV PEMBAHASAN	77
A. Implementasi	77
1. Pembuatan Aplikasi	78
2. Hasil Implementasi	91
B. Pengujian	98
1. Pengujian Sistem	98
2. Pengujian Akurasi	100
V KESIMPULAN DAN SARAN	105
A. Kesimpulan	105
B. Saran	105
DAFTAR PUSTAKA	108
LAMPIRAN	109
A Nilai Probabilitas Penyakit	109
B Nilai Probabilitas Gejala Terhadap Penyakit	111

DAFTAR TABEL

Tabel 3.1	Daftar Penyakit	39
Tabel 3.2	Tabel Kode Penyakit dan Tindakan Penanganannya	40
Tabel 3.3	Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 1	41
Tabel 3.4	Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 2	42
Tabel 3.5	Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 3	43
Tabel 3.6	Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 4	44
Tabel 3.7	Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 5	45
Tabel 3.8	Tabel Kode Gejala-Gejala Penyakit	45
Tabel 3.9	Tabel Kode Gejala-Gejala Penyakit Lanjutan 1	46
Tabel 3.10	Tabel Kode Gejala-Gejala Penyakit Lanjutan 2	47
Tabel 3.11	Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya	48
Tabel 3.12	Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya Lanjutan 1	49
Tabel 3.13	Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya Lanjutan 2	50
Tabel 3.14	Tabel Probabilitas Kemunculan Suatu Penyakit	51
Tabel 3.15	Tabel Relasi Gejala Terhadap Penyakit	54
Tabel 3.16	Tabel Relasi Gejala Terhadap Penyakit Lanjutan	55
Tabel 3.17	<i>Database User</i>	67
Tabel 3.18	<i>Database Gejala</i>	67
Tabel 3.19	<i>Database Penyakit</i>	68
Tabel 3.20	<i>Database Aturan</i>	68
Tabel 3.21	<i>Database Temporary User</i>	69

Tabel 3.22	<i>Database Temporary Gejala</i>	69
Tabel 3.23	<i>Database Temporary Penyakit</i>	69
Tabel 3.24	<i>Database Temporary Analisa</i>	70
Tabel 3.25	<i>Database Hasil Gejala</i>	70
Tabel 3.26	<i>Database Hasil Analisa</i>	71
Tabel 4.1	Hasil Pengujian Sistem Lanjutan	99
Tabel 4.2	Hasil Pengujian Akurasi	100
Tabel 4.3	Hasil Pengujian Akurasi Lanjutan	101
Tabel 4.4	Hasil Pengujian Akurasi Lanjutan 2	102
Tabel 4.5	Hasil Pengujian Akurasi Lanjutan 3	103
Tabel 1.1	Tabel Nilai Probabilitas	110
Tabel 2.1	Nilai Probabilitas Gejala Terhadap Penyakit	111
Tabel 2.2	Nilai Probabilitas Gejala Terhadap Penyakit Lanjutan	112
Tabel 2.3	Nilai Probabilitas Gejala Terhadap Penyakit Lanjutan 2	113

DAFTAR GAMBAR

Gambar 2.1	Struktur Sistem Pakar [13]	10
Gambar 2.2	Proses <i>Forward Chaining</i> [13]	11
Gambar 2.3	Ilustrasi Bayes [14]	15
Gambar 2.4	Ilustrasi Naive Bayes [14]	16
Gambar 2.5	Anatomi Gigi	19
Gambar 2.6	<i>Use Case Diagram</i> [18]	25
Gambar 2.7	<i>Activity Diagram</i> [18]	26
Gambar 2.8	Simbol <i>Entity Relationship Diagram</i>	28
Gambar 2.9	Struktur Navigasi Linier [4]	28
Gambar 2.10	Struktur Navigasi Non-Linier [4]	29
Gambar 2.11	Struktur Navigasi Hirarki [4]	29
Gambar 2.12	Struktur Navigasi Campuran [4]	30
Gambar 2.13	Metode <i>Waterfall</i> [6]	30
Gambar 2.14	<i>Flowchart Framework CodeIgniter</i> [20]	33
Gambar 3.1	Tahapan Penelitian	36
Gambar 3.2	Arsitektur Sistem Pakar Diagnosa Penyakit Gigi dan Mulut	38
Gambar 3.3	Diagram Penalaran Sistem	56
Gambar 3.4	<i>Use Case Diagram User</i>	62
Gambar 3.5	<i>Activity Diagram User</i>	63
Gambar 3.6	Diagram Konteks	64
Gambar 3.7	DFD level 0	64
Gambar 3.8	DFD level 1 Proses Manajemen Basis Pengetahuan	65
Gambar 3.9	DFD level 1 Proses Diagnosa	65
Gambar 3.10	<i>Entity Relationship Diagram</i>	66

Gambar 3.11	Struktur Navigasi	71
Gambar 3.12	Tampilan <i>Home</i>	73
Gambar 3.13	Tampilan Informasi	73
Gambar 3.14	Tampilan Register	74
Gambar 3.15	Tampilan <i>Login</i>	74
Gambar 3.16	Tampilan Konsultasi	75
Gambar 3.17	Tampilan Hasil Konsultasi	75
Gambar 3.18	Tampilan Riwayat Konsultasi	76
Gambar 3.19	Tampilan <i>Help</i>	76
Gambar 4.1	Instalasi Framework Codeigniter	78
Gambar 4.2	Pengaktifan Modul Apache dan MySQL pada XAMPP	79
Gambar 4.3	Halaman Utama Codeigniter	79
Gambar 4.4	Pembuatan <i>Interface</i> Sistem menggunakan <i>Sublime Text</i>	80
Gambar 4.5	Membuat <i>Database</i> pada MySQL Menggunakan phpMyAdmin	80
Gambar 4.6	Membuat Tabel pada <i>Database</i>	81
Gambar 4.7	Mengisi Data pada Tabel	81
Gambar 4.8	static.php	82
Gambar 4.9	v_home.php	83
Gambar 4.10	v_informasi.php	84
Gambar 4.11	readmore.php	84
Gambar 4.12	login.php	85
Gambar 4.13	signup.php	86
Gambar 4.14	v_konsultasi.php	87
Gambar 4.15	v_hasil_konsultasi.php	88
Gambar 4.16	v_hasil_konsultasi.php	88

Gambar 4.17	v_riwayat_konsultasi.php	89
Gambar 4.18	v_detail_riwayat_konsultasi.php	90
Gambar 4.19	help.php	91
Gambar 4.20	Tampilan Menu <i>Home</i>	91
Gambar 4.21	Tampilan Menu Informasi	92
Gambar 4.22	Tampilan Detail Informasi	92
Gambar 4.23	Tampilan Form Login	93
Gambar 4.24	Tampilan Form Register	93
Gambar 4.25	Tampilan Halaman Konsultasi	94
Gambar 4.26	Tampilan Halaman Hasil Konsultasi	95
Gambar 4.27	Tampilan Halaman Penyakit Tidak Ditemukan	96
Gambar 4.28	Tampilan Halaman Riwayat Konsultasi	96
Gambar 4.29	Tampilan Halaman Detail Riwayat Konsultasi	97
Gambar 4.30	Tampilan Halaman Help	98

ABSTRAK

Penyakit gigi dan mulut merupakan salah satu masalah kesehatan yang banyak dikeluhkan oleh masyarakat Indonesia. Minimnya pengetahuan serta terbatasnya sumber informasi mengenai kesehatan gigi dan mulut menyebabkan kesadaran masyarakat untuk menjaga kesehatan gigi dan mulut masih rendah. Dalam penelitian skripsi ini, dipaparkan solusi untuk mengatasi permasalahan ini yaitu dengan mengembangkan salah satu cabang dari kecerdasan buatan yaitu sistem pakar yang dapat membantu masyarakat dalam mengetahui kesehatan gigi dan mulut. Sistem pakar ini dikembangkan dengan metode forward chaining dan metode naive bayes yang memiliki lima tahap penelitian yaitu identifikasi masalah, pengumpulan data, analisis sistem, implementasi sistem, dan pengujian sistem. Pengembangan sistem pakar berbasis web menggunakan framework codeigniter dan MySQL. Sistem pakar ini dapat mendiagnosa penyakit gigi dan mulut serta memberikan tindakan penanganan yang dapat dilakukan oleh user.

Kata kunci : Sistem Pakar, *Forward Chaining*, *Naive Bayes*, *Framework Codeigniter*.

ABSTRACT

Dental and oral diseases are one of the health problems that many people in Indonesia complain about. Lack of knowledge and limited sources of information about oral and dental health lead to people awareness to maintain their own is still low. In the study of this thesis, presented a solution to overcome this problem is by developing one branch of artificial intelligence which is an expert system that can help people in knowing the dental and oral health. This expert system is developed by forward chaining method and naive bayes method which has five steps of research that is problem identification, collecting data, system analysis, system implementation, and system testing. The development of web-based expert system using codeigniter and MySQL framework. This expert system can diagnose dental and oral diseases and provide handling actions that can be performed by the user.

Keywords : *Expert System, Forward Chaining, Naive Bayes, Framework Codeigniter.*

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Gigi dan mulut adalah bagian penting yang harus dipertahankan kesehatannya. Banyak masyarakat yang menganggap remeh kesehatan gigi dan mulut. Padahal masalah gigi yang dibiarkan akan menimbulkan penyakit lain yang berbahaya yang bahkan konsekuensinya bisa berbahaya. Gigi dapat menjadi sumber infeksi atau fokus infeksi yang berbahaya bagi organ tubuh lain [8].

Penyakit gigi dan mulut merupakan salah satu masalah kesehatan yang banyak dikeluhkan oleh masyarakat Indonesia. Menurut hasil riset kesehatan dasar Indonesia tahun 2013, sebesar 25,9% penduduk Indonesia mempunyai masalah gigi dan mulut. Minimnya pengetahuan serta terbatasnya sumber informasi mengenai kesehatan gigi dan mulut menyebabkan kesadaran masyarakat untuk menjaga kesehatan gigi dan mulut masih rendah. Diantara mereka, terdapat 31,1% yang menerima perawatan dan tenaga medis gigi yaitu perawat gigi, dokter gigi atau dokter gigi spesialis, sementara 68,9% lainnya tidak melakukan perawatan. Selain itu, secara keseluruhan tingkat kemampuan atau keterjangkauan masyarakat untuk mendapatkan pelayanan dari tenaga medis atau *Effective Medical Demand* (EMD) pun hanya sebesar 8,1% [10].

Menurut Dekan Fakultas Kedokteran Gigi Universitas Padjajaran pada pembukaan acara peringatan tahun emas FKG UNPAD tahun 2016, jumlah rasio ideal antara tenaga dokter gigi terhadap jumlah penduduk di Indonesia adalah 1 berbanding 9.000. Namun karena masih rendahnya tenaga dokter gigi di Indonesia, rasio itu membengkak hingga 1 berbanding 24.000. Jumlah rasio ideal ini sangat jauh dengan standar yang ditetapkan *World Health Organization* (WHO) yaitu 1 berbanding

2.000 penduduk. Kondisi memprihatinkan ini masih ditambah dengan belum meratanya persebaran dokter gigi, di mana 70% nya masih terpusat di Pulau Jawa [3].

Perkembangan salah satu bidang teknologi informasi yaitu kecerdasan buatan (*artificial intelligence*) telah banyak diaplikasikan dalam berbagai bidang kehidupan dapat dimanfaatkan sebagai solusi untuk mengatasi permasalahan ini. Salah satu cabang dari kecerdasan buatan yaitu sistem pakar (*expert system*) dapat diterapkan untuk membuat sistem yang dapat membantu masyarakat dalam mengetahui kesehatan gigi dan mulut serta dapat mendiagnosa prediksi awal penyakit gigi dan mulut yang dialami. Sistem pakar adalah suatu sistem komputer yang menyamai (*emulates*) kemampuan pengambilan keputusan dari seorang pakar [13]. Sistem pakar ini dapat menyimpan pengetahuan dari seseorang atau beberapa orang pakar dalam suatu basis pengetahuan (*knowledge base*) dan menggunakan sistem penalaran yang menyamai seorang pakar dalam memecahkan masalah.

Dalam perancangan suatu sistem pakar terdapat komponen mesin inferensi yang mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam memecahkan masalah. Terdapat dua metode untuk mengontrol inferensi dalam sistem pakar, salah satunya yaitu pelacakan ke depan (*forward chaining*). *Forward chaining* adalah metode yang dimulai dari sekumpulan data dan kemudian melakukan inferensi sesuai dengan aturan yang ditetapkan hingga menemukan kesimpulan. Metode ini sangat cocok digunakan untuk menangani masalah pengendalian dan peramalan [4]. Pelacakan pada kasus ini adalah untuk mengetahui apakah gejala-gejala penyakit gigi dan mulut yang dialami oleh *user* itu termasuk dalam satu jenis penyakit atau jenis-jenis penyakit yang lain atau bahkan bukan salah satu dari banyaknya penyakit tersebut, yang artinya sistem belum mampu mengambil kesimpulan karena keterbatasan aturan.

Dalam penelitian ini, metode *Forward Chaining* akan dikombinasikan dengan

metode *Naive Bayes*, yang digunakan untuk menentukan nilai maksimum dari setiap penyakit gigi dan mulut. Pemilihan penggunaan metode *Naive Bayes* dikarenakan metode ini menggunakan asumsi keindependenan atribut yang artinya menghilangkan kebutuhan banyaknya jumlah data latih dari perkalian kartesius seluruh atribut yang dibutuhkan untuk mengklasifikasikan suatu data atau dapat diartikan metode ini tidak membutuhkan biaya komputasi yang mahal seperti waktu, *processor*, dan ukuran memori yang besar untuk menghitung nilai probabilitas untuk tiap nilai dari perkalian kartesius untuk tiap nilai atribut dan tiap nilai kelas. Sehingga dampak negatif dari metode tersebut adalah keterkaitan yang ada antara nilai-nilai atribut diabaikan sepenuhnya [14].

Pengimplementasian metode *Forward Chaining* pada sebuah sistem pakar sudah pernah dilakukan oleh Denisha Trihapningsari dalam kasus untuk mendiagnosa gangguan sistem pencernaan pada lambung [4]. Pengimplementasian metode *Naive Bayes* pada sistem pakar juga sudah pernah dilakukan oleh Khairil Mustaqim dalam kasus untuk menentukan hama dan penyakit tanaman kelapa sawit [11]. Sistem yang akan peneliti buat nantinya akan memiliki kelebihan tertentu salah satunya seperti fitur untuk menyimpan riwayat konsultasi yang pernah dilakukan oleh *user*.

Dengan penjabaran di atas, penulis mengkombinasikan jurnal-jurnal tersebut untuk melakukan penelitian tentang “**Rancang Bangun Sistem Pakar Diagnosa Penyakit Gigi dan Mulut Menggunakan Metode *Forward Chaining* dan Metode *Naive Bayes* Berbasis Web**”. Sistem pakar ini dirancang berbasis web sehingga dapat diakses dengan mudah dan dimanfaatkan masyarakat secara luas dengan tidak terbatas oleh waktu dan tanpa biaya yang besar. Dengan adanya sistem pakar ini diharapkan dapat memberikan informasi kesehatan gigi dan mulut kepada masyarakat serta dapat melakukan prediksi awal, sehingga jenis penyakit gigi dan mulut yang dialami dapat diketahui dengan cepat dan memberikan gambaran tindakan penanganan

yang membuat masyarakat lebih peduli dengan kesehatan gigi dan mulut dan menyegerakan untuk berkonsultasi dengan dokter gigi sehingga mendapatkan penanganan yang tepat.

B. Pembatasan Masalah

Adapun batasan masalah pada sistem pakar diagnosa penyakit gigi dan mulut ini adalah sebagai berikut:

1. Sistem pakar dibuat hanya untuk mendiagnosa penyakit gigi dan mulut secara umum.
2. Sistem ini menggunakan 37 gejala dan 15 penyakit dengan nilai probabilitas masing-masing yang diperoleh dari pakar.
3. Sumber pakar terdiri dari 3 orang ahli dokter gigi pada Klinik A, Klinik B, dan Klinik C.
4. Interaksi *user* dengan sistem berupa pertanyaan mengenai gejala penyakit yang harus dijawab oleh *user* dengan pilihan jawaban “Ya” atau “Tidak”.
5. *Output* dari sistem pakar ini berupa kesimpulan yang berisikan prediksi awal penyakit gigi dan mulut serta tindakan penanganan yang berupa gambaran perawatan yang hanya dapat dilakukan di klinik gigi atau rumah sakit.
6. *User* yang dapat mengakses menu konsultasi adalah *user* yang telah melakukan login terlebih dahulu.
7. Pakar hanya sebagai sumber data basis pengetahuan yang terdiri dari data gejala dan penyakit dengan nilai probabilitas masing-masing, data penanganan dan data aturan diagnosis.

8. Perancangan dan pembangunan sistem pakar berbasis web menggunakan bahasa pemrograman *PHP* dan *database MySQL* dengan *framework Codeigniter*.
9. Sistem bersifat statis.
10. Pengujian akurasi hanya dilakukan dengan pakar yang merupakan narasumber.

C. Perumusan Masalah

Berdasarkan uraian di atas, maka rumusan masalah pada penelitian ini adalah bagaimana merancang dan membangun suatu sistem pakar yang dapat membantu *user* atau masyarakat untuk memberikan prediksi awal tentang penyakit gigi dan mulut yang dialami serta untuk mengetahui tindakan penanganan yang tepat dengan menggunakan metode *Forward Chaining* dan *Naive Bayes* berbasis web?

D. Tujuan Penelitian

Tujuan perancangan sistem pakar diagnosa penyakit gigi dan mulut ini adalah untuk merancang dan membangun suatu sistem pakar yang dapat membantu *user* atau masyarakat untuk memberikan prediksi awal tentang penyakit gigi dan mulut yang dialami serta untuk mengetahui tindakan penanganan yang tepat dengan menggunakan metode *Forward Chaining* dan *Naive Bayes* berbasis web.

E. Manfaat Penelitian

Manfaat perancangan aplikasi sistem pakar diagnosa penyakit gigi dan mulut ini adalah:

1. Memberikan informasi seputar kesehatan gigi dan mulut kepada *user* atau masyarakat.

2. Tersedianya aplikasi yang dapat memberikan informasi serta prediksi awal penyakit gigi dan mulut yang dialami *user* atau masyarakat melalui diagnosa gejala-gejala penyakitnya.
3. Memberikan informasi mengenai tindakan penanganan yang dapat dilakukan pada penyakit yang terkait.

F. Jenis Penelitian

Jenis penelitian yang dijalani oleh peneliti berjenis Kajian Teori. Jenis penelitian ini mengarahkan penulis kepada penerapan metode *Forward Chaining* dan *Naive bayes* dalam pengembangan sebuah sistem pakar diagnosa penyakit gigi dan mulut berbasis web.

BAB II

LANDASAN TEORI

A. Sistem Pakar

Sistem Pakar pertama kali dikembangkan oleh komunitas AI pada pertengahan tahun 1960. Sistem pakar yang muncul pertama kali adalah *General Problem Solver* (GPS) yang dikembangkan oleh Newel dan Simon. Nama sistem pakar berasal dari istilah *knowledge-based expert system*. Istilah ini muncul karena sistem pakar adalah sistem yang menggunakan pengetahuan seorang manusia yang ditangkap dalam komputer untuk memecahkan masalah yang memerlukan keahlian manusia [7].

Profesor Edward Feigenbaum dari Universitas Stanford yang merupakan seorang pelopor awal dari teknologi sistem pakar mendefinisikan sistem pakar adalah program komputer cerdas yang menggunakan pengetahuan dan prosedur inferensi untuk menyelesaikan masalah yang cukup sulit yang memerlukan keahlian manusia yang signifikan sebagai solusinya. Menurut Arhami (2005) suatu sistem pakar adalah suatu sistem komputer yang menyamai (*emulates*) kemampuan pengambilan keputusan dari seorang pakar. Istilah *emulates* berarti bahwa sistem pakar diharapkan dapat bekerja dalam semua hal seperti seorang pakar [13].

Perancangan sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (*inference rules*) dengan basis pengetahuan (*knowledge base*) 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 memecahkan masalah tertentu.

1. Konsep Umum Sistem Pakar

Konsep dasar dari suatu sistem pakar mengandung beberapa unsur atau elemen, yaitu [13]:

1. Kepakaran (*Expertise*)

Kepakaran merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dan lebih baik daripada seorang yang bukan pakar.

2. Pakar (*Expert*)

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus serta mampu menerapkannya untuk memecahkan masalah. Seorang pakar harus mampu menjelaskan dan mempelajari hal-hal baru yang berkaitan dengan topik permasalahan.

3. Pemindahan Kepakaran (*Transferring Expertise*)

Tujuan dari sistem pakar adalah memindahkan kepakaran dari seorang pakar kedalam komputer, kemudian ditransfer kepada orang lain yang bukan pakar. Proses ini melibatkan akuisisi pengetahuan, representasi pengetahuan, inferensi pengetahuan, dan pemindahan pengetahuan ke pengguna.

4. Inferensi (*Inference*)

Inferensi adalah sebuah prosedur (program) yang mempunyai kemampuan untuk penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Pengetahuan yang dimiliki oleh seorang pakar disimpan pada basis pengetahuan oleh sistem pakar. Tugas mesin inferensi adalah mengambil kesimpulan berdasarkan basis pengetahuan yang dimilikinya.

5. Aturan-Aturan (*Rules*)

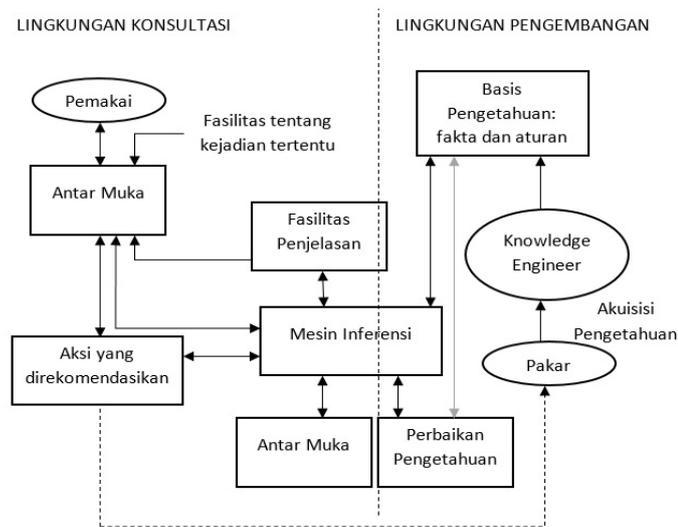
Sistem berbasis aturan (*rule based systems*) adalah salah satu representasi pengetahuan yang menghubungkan premis dengan konklusi yang diakibatkannya. Suatu aturan produksi dituliskan dalam bentuk jika maka (*IF-THEN*).

6. Kemampuan Menjelaskan (*Explanation Capability*)

Fasilitas lain dari sistem pakar adalah kemampuannya untuk menjelaskan saran atau rekomendasi yang diberikannya. Penjelasan dilakukan dalam subsistem yang disebut subsistem penjelas (*explanation*). Bagian dari sistem ini memungkinkan sistem untuk memeriksa penalaran yang dibuatnya sendiri dan menjelaskan operasi-operasinya.

2. Struktur dan Arsitektur Sistem Pakar

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) [7]. Lingkungan pengembangan digunakan oleh sistem pakar builder untuk membangun komponen dan memasukkan pengetahuan ke dalam basis pengetahuan. Lingkungan konsultasi digunakan oleh nonpakar untuk memperoleh pengetahuan dan nasihat pakar. Lingkungan ini dapat dipisahkan setelah sistem lengkap. Komponen-komponen sistem pakar dalam kedua bagian tersebut dapat dilihat dalam Gambar 2.1 berikut ini:



Gambar 2.1: Struktur Sistem Pakar [13]

Sistem pakar sebagai sebuah program yang difungsikan untuk menirukan pakar manusia harus bisa melakukan hal-hal yang dapat dikerjakan oleh seorang pakar[19]. Untuk membangun sistem yang seperti itu maka komponen-komponen yang harus dimiliki adalah sebagai berikut [13]:

1. Akuisisi Pengetahuan (*Knowledge Acquisition*)

Akuisisi pengetahuan adalah akumulasi, transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer [1]. Tahap *knowledge acquisition* ini dikerjakan oleh seorang *knowledge engineer*. *Knowledge engineer* adalah seseorang yang memiliki latar belakang pengetahuan tentang komputer dan mengerti cara pengembangan sistem pakar serta yang menerjemahkan pengetahuan seorang pakar dalam bentuk deklaratif sehingga dapat digunakan oleh sistem pakar.

2. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Komponen sistem pakar ini disusun atas dua ele-

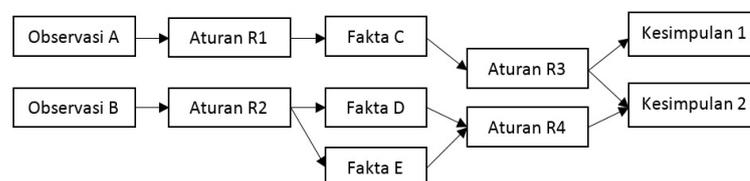
men dasar, yaitu fakta dan aturan. Fakta merupakan informasi tentang obyek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

3. Mesin Inferensi (*Inference Engine*)

Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran yang terdapat pada suatu kondisi berdasarkan pada basis pengetahuan yang ada, manipulasi dan mengarahkan kaidah, model dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian, yaitu strategi yang berfungsi sebagai panduan arah dalam melakukan proses penalaran. Ada dua metode inferensi yang dapat digunakan, yaitu *forward chaining* dan *backward chaining*.

a. *Forward Chaining*

Forward chaining atau pelacakan ke depan adalah pendekatan yang dimotori data (*data driven*). Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan.



Gambar 2.2: Proses *Forward Chaining* [13]

b. *Backward Chaining*

Backward chaining atau pelacakan ke belakang adalah pendekatan yang dimotori tujuan (*goal driven*). Dalam metode ini, penalaran dimulai dengan

tujuan merunut balik ke jalur yang akan mengarahkan ke tujuan tersebut. Pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya.

4. Daerah Kerja (*Workplace*)

Workplace merupakan area dari sekumpulan memori kerja (*working memory*). *Workplace* digunakan untuk merekam hasil-hasil antara dan kesimpulan yang dicapai.

5. Antar Muka Pengguna (*User Interface*)

User interface merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai. Menurut McLeod (1995), pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan sistem pakar menerima instruksi dan informasi (*input*) dari pemakai, juga memberikan informasi (*output*) kepada pemakai [17].

6. Fasilitas Penjelasan (*Explanation/Justifier*)

Fasilitas penjelasan adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pengguna bagaimana suatu kesimpulan dapat diambil.

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

Pakar memiliki kemampuan untuk menganalisis dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya. Kemampuan tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya.

8. Representasi Pengetahuan (*Knowledge Representation*)

Representasi pengetahuan merupakan kombinasi sistem berdasarkan dua elemen, yaitu struktur data dan penafsiran prosedur untuk digunakan pengetahuan dalam menyimpan struktur data. Dengan kata lain, representasi pengetahuan adalah cara untuk menyajikan pengetahuan yang diperoleh ke dalam suatu sistem skema atau diagram tertentu sehingga dapat diketahui relasi antara suatu pengetahuan dengan pengetahuan yang lain dan dapat dipakai untuk menguji kebenaran penalaran. Menurut Firebaugh (1989), terdapat 4 teknik yang telah dibuktikan efektif untuk representasi pengetahuan, yaitu [15]:

a. Jaringan Semantik

Jaringan semantik merupakan representasi pengetahuan yang digunakan untuk menggambarkan data dan informasi yang menunjukkan hubungan antara berbagai objek. Struktur dari jaringan semantik ditunjukkan secara grafik yang terdiri dari simpul (*node*) dan busur (*arc*) yang menghubungkannya. Simpul menyatakan objek dan busur sebagai *links* atau *edge*.

b. *Frame*

Frame digunakan untuk merepresentasikan pengetahuan *stereotype* atau pengetahuan yang didasarkan kepada karakteristik yang sudah dikenal yang merupakan pengalaman masa lalu dengan menggambarkan objek.

c. *Script*

Script merupakan skema representasi pengetahuan yang sama dengan *frame* namun *script* menggambarkan urutan peristiwa. Berbeda dengan *frame*, *script* biasanya direpresentasikan ke dalam konteks tertentu. Penggambaran urutan peristiwa pada *script* menggunakan serangkaian *slot* yang berisi informasi tentang orang, objek, dan tindakan-tindakan yang terjadi dalam suatu peristiwa

d. Kaidah Produksi

Kaidah produksi merupakan salah satu representasi pengetahuan yang menghubungkan premis dengan konklusi. Aturan produksi dituliskan dalam bentuk jika-maka (*IF* premis *THEN* konklusi). Konklusi pada bagian *THEN* bernilai benar jika premis pada bagian *IF* bernilai benar.

e. Logika

Representasi logika menggunakan ekspresi-ekspresi dalam logika formal untuk merepresentasikan basis pengetahuan. Pada dasarnya proses logika adalah proses membentuk kesimpulan dan menarik suatu inferensi berdasarkan fakta yang telah ada. *Input* dari proses logika berupa fakta-fakta yang diakui kebenarannya sehingga dengan melakukan penalaran pada proses logika dapat dibentuk suatu inferensi atau kesimpulan yang benar pula.

B. Probabilitas dan Teorema Bayes

Probabilitas Bayes merupakan salah satu cara untuk mengatasi ketidakpastian data dengan menggunakan formula Bayes yang dinyatakan [13]:

$$P(c|E) = \frac{P(E|c)P(c)}{P(E)}$$

Dimana

$P(c|E)$: probabilitas kelas c jika diberikan atribut E

$P(E|c)$: probabilitas munculnya atribut E jika diketahui kelas c

$P(c)$: probabilitas kelas c tanpa memandang atribut apapun

$P(E)$: probabilitas atribut E

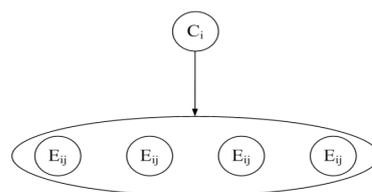
Umumnya, kelompok atribut E direpresentasikan dengan sekumpulan nilai atribut $(x_1, x_2, x_3, \dots, x_n)$ dimana x_i adalah nilai atribut X_i . C adalah variabel klasifikasi dan c adalah nilai dari C . Untuk menentukan pilihan kelas, digunakan peluang maksimal dari seluruh c dalam C , dengan fungsi[14]:

$$\underset{c \in C}{\operatorname{argmax}} \frac{P(E|c)P(c)}{P(E)}$$

Karena nilai $P(E)$ konstan untuk semua kelas, maka $P(E)$ dapat diabaikan sehingga menampilkan fungsi:

$$\underset{c \in C}{\operatorname{argmax}} P(E|c)P(c)$$

Rumus di atas menunjukkan adanya keterkaitan antara atribut E yang satu dengan yang lain, dengan kata lain atribut pada teorema *Bayes* bersifat dependen. Kehadiran atau ketiadaan dari suatu atribut tertentu dari suatu kelompok berhubungan dengan kehadiran atau ketiadaan dari atribut lainnya, seperti yang diilustrasikan dalam Gambar 2.3.



Gambar 2.3: Ilustrasi Bayes [14]

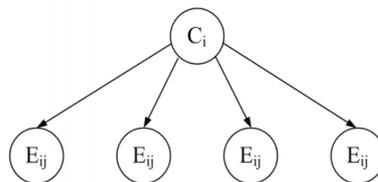
Pada Gambar di atas, C_i adalah variabel klasifikasi (hipotesis) dan E_{ij} adalah variabel atribut (*evidence*). Semua E_{ij} berada dalam satu lingkaran oval, menggambarkan keterkaitan antara atribut yang satu dengan atribut yang lain.

Pengklasifikasian menggunakan teorema *Bayes* membutuhkan waktu prosesor dan ukuran memori yang besar, karena kebutuhan untuk menghitung nilai proba-

bilitas untuk tiap nilai dari perkalian kartesius untuk tiap nilai atribut (*evidence*) dan tiap nilai kelas (hipotesis). Data latih untuk teorema *Bayes* membutuhkan paling tidak perkalian kartesius dari seluruh kelompok atribut yang mungkin, jika misalkan ada 16 atribut yang masing-masingnya berjenis *Boolean* tanpa *missing value*, maka data latih minimal yang dibutuhkan oleh teorema Bayes untuk digunakan dalam klasifikasi adalah $2^{16} = 65.536$ data [14].

1. Metode *Naive Bayes*

Untuk mengatasi permasalahan di atas, berbagai varian dari pengklasifikasian yang menggunakan teorema bayes diajukan, salah satunya adalah *Naive Bayes*, yaitu penggunaan Teorema *Bayes* dengan asumsi keindependenan atribut seperti pada Gambar 2.4 [14].



Gambar 2.4: Ilustrasi *Naive Bayes* [14]

Asumsi keindependenan atribut akan menghilangkan kebutuhan banyaknya jumlah data latih dari perkalian kartesius seluruh atribut yang dibutuhkan untuk mengklasifikasikan suatu data dengan persamaan:

$$P(c|E) = \underset{c \in C}{argmax} P(c) \prod_{j=1}^n P(E_j|c)$$

Dengan

$P(c|E)$: probabilitas kelas c jika diberikan atribut (*evidence*) E

$P(c)$: probabilitas awal kelas c (tanpa memandang atribut apapun)

$P(E_j|c)$: probabilitas munculnya atribut (*evidence*) E_j dalam kelas c

Langkah-langkah dalam menggunakan algoritma *Naive Bayes* adalah sebagai berikut:

1. Menentukan nilai probabilitas pada setiap jenis penyakit (*probabilitas prior*), $P(h)$ = peluang dari hipotesa h (jenis penyakit).
2. Menentukan nilai probabilitas sebuah fakta gejala pada penyakit yang mempengaruhi suatu hipotesa (*probabilitas likelihood*), $P(e|h)$ = peluang data fakta gejala e , bila diasumsikan bahwa hipotesa h benar.
3. Menghitung $P(h)\prod_{j=1}^n P(e_j|h)$ untuk setiap hipotesa (*probabilitas posterior*), $P(h|e)$ = peluang bahwa hipotesa benar untuk data fakta gejala e yang diamati

2. Contoh Perhitungan *Naive Bayes*

Dalam penelitian ini, probabilitas gejala dan probabilitas penyakit didapat dari hasil wawancara dengan pakar atau dokter gigi. Diketahui probabilitas mengalami penyakit *gingivitis* = 0,7. Contoh kasus *user* mengalami gejala-gejala sebagai berikut:

1. Gusi meradang
2. Gusi tampak merah dan bengkak
3. Gusi mudah berdarah
4. Bau mulut tak sedap
5. Gusi terasa sakit/nyeri
6. Bentuk gusi agak membulat atau tumpul

Untuk mengetahui jenis penyakit gigi dan mulut yang menyerang *user* tersebut, langkah penyelesaiannya yaitu:

1. Kalikan probabilitas semua gejala dalam suatu penyakit
2. Hasil 1 dikalikan dengan probabilitas penyakit
3. Perhitungan nilai probabilitas dengan nilai maksimum

Proses klasifikasi:

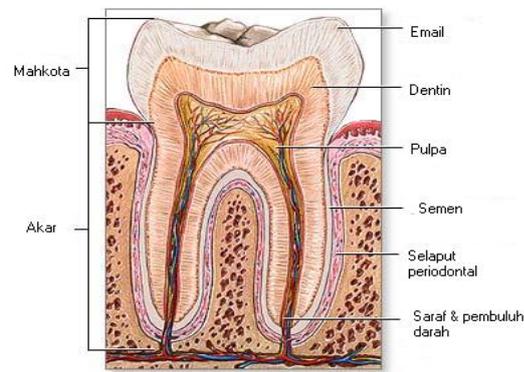
$$\begin{aligned}
& 1. P(\text{Gingivitis}|\text{Gejala}) \\
& = P(\text{Gingivitis}) * P(\text{Gusi meradang}|\text{Gingivitis}) * P(\text{Gusi tampak merah dan bengkak}|\text{Gingivitis}) * P(\text{Gusi mudah berdarah}|\text{Gingivitis}) * P(\text{Bau mulut tak sedap}|\text{Gingivitis}) \\
& \quad * H(\text{Gusi terasa sakit/nyeri}|\text{Gingivitis}) * P(\text{Bentuk gusi agak membulat atau tum-pull}|\text{Gingivitis}) \\
& = 0.7 * 0.77 * 0.63 * 0.8 * 0.7 * 0.5 * 0.8 \\
& = 0.07606368
\end{aligned}$$

Dari perhitungan di atas, diperoleh sebuah penyakit yaitu penyakit *gingivitis*. Kesimpulannya bahwa orang tersebut mengalami penyakit *gingivitis* dengan nilai *naive bayes* sebesar 0.07606368.

C. Gigi dan Mulut

Mulut adalah rongga terbuka tempat masuknya makanan dan air. Mulut merupakan bagian awal dari sistem pencernaan. Dalam bahasa Latin, mulut dikenal dengan nama *oral* atau *oris*. Komponen organ yang menyertai mulut antara lain gigi, bibir sebagai pembatas sebelah depan, *faring* sebagai pembatas sebelah belakang, pipi sebagai pembatas sebelah samping, lidah sebagai pembatas sebelah bawah, dan *planatum* (langit-langit) sebagai pembatas atas [12].

Gigi adalah tulang keras kecil berwarna putih yang tumbuh tersusun, berakar dalam gusi dan berfungsi untuk mengunyah dan menggigit [21]. Pada umumnya manusia memiliki struktur, bentuk dan jumlah gigi yang sama. Pada anak-anak terdapat 20 buah gigi sulung, dimana tiap-tiap rahang terdiri atas 4 gigi seri, 2 gigi taring, dan 5 gigi geraham. Untuk gigi dewasa, memiliki 32 buah gigi tetap, pada tiap rahangnya terdiri atas 4 gigi seri, 2 gigi taring, 4 gigi geraham kecil, dan 6 gigi geraham besar.



Gambar 2.5: Anatomi Gigi

Struktur gigi terdiri dari tiga bagian yaitu mahkota gigi merupakan bagian yang menonjol di atas gusi, akar gigi yang merupakan bagian terpendam dalam tulang rahang, dan leher gigi yang merupakan pembatas bertemunya mahkota dan akar gigi.

Gigi memiliki tiga bagian utama yang penting yaitu *email*, *dentin* dan *pulpa*. *Email* merupakan bagian terluar gigi yang paling tipis, berwarna putih mengkilap namun agak transparan (*translucent*) dan keras. Kekerasan *email* gigi ini dimungkinkan oleh bahan pembentuk *email* itu sendiri yang berupa struktur *kristalin* (*kristal hidroksiapatit*) dimana 96% terdiri dari bahan anorganik, 1% organik dan sisanya adalah air. Bagian kedua gigi setelah *email* adalah *dentin*. Berbeda dengan *email*, *dentin* berwarna kekuningan dan merupakan bagian terbesar dari gigi. Komposisi *dentin* juga berbanding terbalik dengan *email* dimana *dentin* terdiri dari 85% material organik, ini sebabnya *dentin* bersifat sedikit lebih lunak dari pada *email*. Bentuk *dentin* yang berpori karena tersusun dari *tubuli dentin*. *Tubuli dentin* menyebabkan *dentin* lebih sensitif terhadap rangsangan suhu maupun kimia. Sedangkan *pulpa* merupakan sebuah ruangan di tengah-tengah gigi. Pada ruang ini terdapat pembuluh darah kecil serta saraf-saraf gigi [12].

1. Penyakit Gigi dan Mulut

Menurut Rikesdas (2013), terdapat 9 masalah gigi dan mulut yang paling banyak dialami oleh masyarakat Indonesia dengan 7 diantaranya adalah *karies*, *pulpitis*, *gingivitis*, *stomatitis*, bau mulut (*halitosis*), hipersensitif, dan *periodontitis* [10]. Dengan dikombinasikan oleh *knowledge* yang diperoleh dari pakar, berikut ini adalah beberapa penyakit gigi dan mulut yang digunakan sebagai *knowledge base* pada sistem [21]:

1. Abrasi gigi

Abrasi gigi adalah hilangnya struktur atau terkikisnya bagian luar gigi dan terkadang struktur dalam gigi juga ikut terkikis. Hal ini terjadi disebabkan oleh seringnya menyikat gigi terlalu keras tetapi juga dapat disebabkan oleh terlalu banyak makan makanan yang bersifat asam. Penyebab abrasi gigi juga bisa terjadi akibat adanya gesekan antar gigi atas dan bawah atau mungkin terbiasa menggigit benda-benda keras seperti pulpen atau benda lainnya yang tanpa disadari sedikit demi sedikit dapat mengikis gigi.

2. Abses Periapikal

Abses periapikal adalah pengumpulan nanah yang telah menyebar dari sebuah gigi ke jaringan di sekitarnya. Penyebab *abses periapikal* adalah tubuh menyerang infeksi dengan mengeluarkan sejumlah besar sel darah putih. Sekumpulan sel darah putih dan jaringan yang mati ini disebut nanah. Biasanya nanah dari infeksi gigi pada awalnya dialirkan ke gusi, sehingga gusi yang berada di dekat akar gigi tersebut membengkak. Nanah juga bisa dialirkan ke kulit, mulut, tenggorokan, atau tengkorak, tergantung pada lokasi gigi yang terkena.

3. Abses Periodontal

Abses periodontal adalah inflamasi yang mengandung nanah pada jaringan pe-

riodontal, bersifat kronis atau akut, sering kali *abses* menjadi kronis dan *abses* kronis menjadi akut. *Abses periodontal* ditandai dengan adanya pengumpulan nanah atau pus sepanjang akar gigi yang disebabkan oleh infeksi jaringan *periodontal* dan gigi masih vital. *Abses periodontal* terjadi akibat adanya faktor iritasi seperti plak, kalkulus, infeksi bakteri, infeksi makanan atau trauma jaringan. Keadaan ini dapat menyebabkan kerusakan tulang *alveolar* sehingga terjadi gigi goyang.

4. *Caries* (Gigi Berlubang)

Caries atau gigi berlubang merupakan suatu daerah yang membusuk pada gigi akibat adanya proses yang secara bertahap melarutkan permukaan gigi yang keras (*email*) dan berlanjut ke dalam bagian dalam gigi. Karies diawali dengan timbulnya bercak cokelat atau putih yang kemudian berkembang menjadi lubang cokelat. Lubang ini terjadi karena luluhnya mineral gigi akibat reaksi fermentasi karbohidrat termasuk *sukrosa*, *fruktosa*, dan *glukosa* oleh beberapa tipe bakteri penghasil asam. Lubang pada gigi baru akan terasa sakit bila lubang gigi sudah dalam dan mencapai rongga pulpa yang berisi pembuluh darah dan saraf.

5. *Cheilitis*

Cheilitis adalah kondisi di mana terjadi peradangan pada bibir. Banyak kondisi dan faktor yang dapat menyebabkan *cheilitis*, kronis ataupun sementara, termasuk paparan matahari berlebih, kekurangan gizi, alergi, obat-obatan, dehidrasi, makanan, dan penyakit sistemik tertentu.

6. *Gingivitis*

Gingivitis adalah peradangan pada gusi atau *gingiva*. *Gingivitis* sering terjadi dan bisa timbul kapan saja setelah tumbuhnya gigi. Penyakit ini disebabkan

oleh timbunan *plak* berkepanjangan. *Plak* adalah material yang terdiri dari bakteri, lendir, dan sisa makanan yang menempel pada gigi bagian luar.

7. *Halitosis*

Halitosis adalah bau tidak sedap yang tercium ketika penderita menghembuskan nafasnya. Biasanya *halitosis* disebabkan oleh makanan atau zat tertentu yang ditelan, dihirup atau oleh fermentasi bagian-bagian makanan dalam mulut.

8. *Herpes*

Infeksi *Herpes* Mulut (*Gingivostomatitis Herpetik Primer, Herpes Labialis*) adalah suatu infeksi awal oleh virus *herpes simpleks* yang menyebabkan terbentuknya luka yang terasa nyeri di gusi dan bagian mulut lainnya. Sedangkan *Herpes* sekunder (*Herpes Labialis sekunder*) adalah suatu reaktivasi (pengaktifan kembali) virus *herpes simpleks*. Bila seseorang terinfeksi virus *herpes*, maka virus ini akan terus berada di saraf dalam keadaan laten yang artinya sewaktu-waktu penyakit ini dapat kambuh lagi.

9. *Hipersensitiva Dentin* (Gigi Sensitif)

Hipersensitiva dentin atau gigi sensitif adalah rasa tidak nyaman atau nyeri yang disebabkan oleh rangsangan termal, kimiawi, dan mekanik pada satu atau lebih gigi. Rasa sensitif ini terjadi apabila dentin terbuka yang disebabkan oleh *resesi gingiva*, abrasi, erosi, *periodontitis*, kerusakan restorasi, atau *karies*.

10. *Malocclusion*

Maloklusi adalah bentuk hubungan rahang atas dan bawah yang menyimpang dari bentuk standar yang diterima sebagai bentuk yang normal, *maloklusi* dapat disebabkan karena tidak ada keseimbangan *dentofasial*. Keseimbangan *dentofasial* ini tidak disebabkan oleh satu faktor saja, tetapi beberapa faktor saling

mempengaruhi. Faktor-faktor yang mempengaruhi adalah keturunan, lingkungan, pertumbuhan dan perkembangan, etnik, fungsional, dan patologi.

11. *Periodontitis*

Periodontitis adalah peradangan gusi yang menyebar sampai ke struktur penyangga gigi. *Periodontitis* merupakan salah satu penyebab utama lepasnya gigi pada orang dewasa dan lanjut usia.

12. Persistensi Gigi Sulung

Persistensi gigi sulung adalah suatu keadaan di mana gigi sulung belum tanggal walaupun waktu tanggalnya sudah tiba. Keadaan ini sering dijumpai pada anak usia 6-12 tahun. Persistensi gigi sulung tidak mempunyai penyebab tunggal tetapi merupakan gangguan yang disebabkan multifaktor, salah satu penyebabnya adalah gangguan nutrisi. Gangguan nutrisi dapat menghambat pertumbuhan dan pengembangan gigi. Gangguan akan mengkonsumsi vitamin A dapat menyebabkan terganggunya proses klasifikasi dari *dentin* dan *enamel*. Hal ini mengakibatkan proses erupsi menjadi terhambat sehingga terjadi persistensi.

13. *Pulpitis Reversible*

Pulpitis reversible adalah suatu kondisi inflamasi pulpa yang tidak parah. Jika penyebabnya dihilangkan, inflamasinya akan pulih kembali dan pulpa akan kembali normal. Penyebab *pulpitis reversible* adalah stimuli ringan seperti karies insipien dan fraktur *email* yang mengakibatkan terbukanya *dentin*.

14. *Pulpitis Irreversible*

Pulpitis irreversible adalah inflamasi pulpa parah yang tidak akan pulih kembali sekalipun penyebabnya dihilangkan. Pulpa lambat atau cepat akan menjadi *nekrosis*. Penyebab utama *pulpitis irreversible* adalah bakteri. Penyebab lainnya diantaranya adalah makanan manis, stimulus termis, mekanis kimiawi, dan

asam.

15. *Stomatitis* (Sariawan)

Sariawan atau *stomatitis* adalah radang yang terjadi pada jaringan lunak rongga mulut (*mukosa*), biasanya berupa bercak putih kekuningan. Bercak dapat berupa bercak tunggal maupun berkelompok dan bersifat hilang timbul. Hingga kini, penyebab dari sariawan belum dipastikan, tetapi ada faktor-faktor yang diduga kuat menjadi pemicunya antara lain: (1) trauma pada jaringan lunak mulut selain gigi, misal tergigit; (2) kekurangan nutrisi, terutama vitamin B12, asam *folat*, dan zat besi; (3) gangguan autoimun atau kekebalan tubuh; (4) gangguan hormonal, seperti saat wanita akan memasuki masa menstruasi; (5) penggunaan gigi tiruan atau *behel* yang mengiritasi jaringan lunak; (6) hipersensitivitas terhadap rangsangan antigenik tertentu terutama makanan.

D. Alat Perancangan Sistem

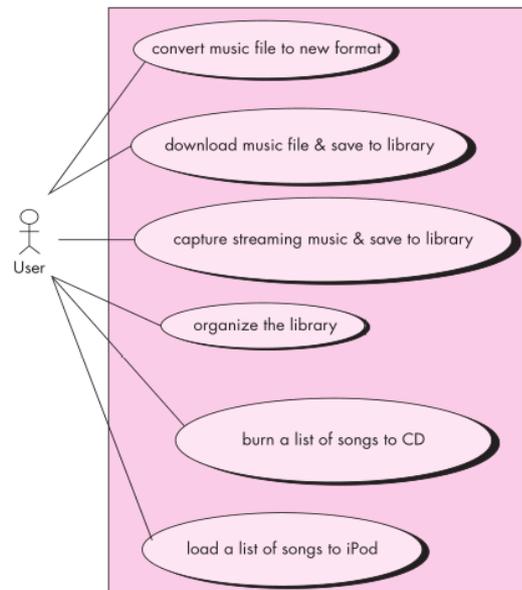
1. *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah bahasa standar untuk menulis rancangan *software*. UML dapat digunakan untuk membangun dan mendokumentasikan kerangka *software*. Di sisi lain, dengan membangun rancangan arsitektur untuk digunakan para konstruksi perusahaan, pembangun *software* membuat diagram UML untuk membantu pengembang *software* untuk membangun *software*. Sehingga lebih mudah untuk dipahami, mudah untuk menentukan sistem dan mudah untuk menjelaskan desain sistem UML terdiri dari *Class Diagram*, *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram* [18].

1. *Use Case Diagram*

Model *use case diagram* membantu untuk menentukan fungsi dan fitur *soft-*

ware dari perspektif pengguna. *Use case* menjelaskan bagaimana pengguna berinteraksi dengan sistem dengan cara mendefinisikan langkah langkah yang diperlukan untuk mencapai tujuan tertentu. *Use case* diagram merupakan gambaran dari semua *use case* dan hubungan antar setiap *use case*.

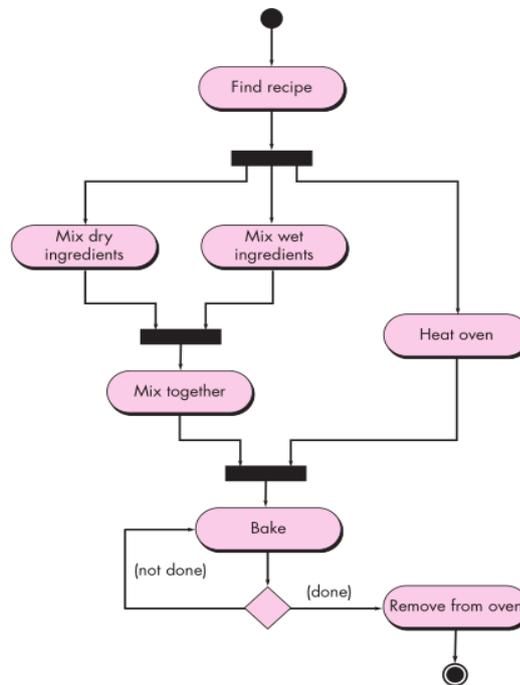


Gambar 2.6: *Use Case Diagram* [18]

Dalam *use case diagram*, *use case* ditampilkan dalam bentuk oval. Para aktor terhubung dengan setiap *use case* menggunakan garis. Karena *use case diagram* menampilkan semua *use case*, hal ini sangat membantu untuk memastikan bahwa semua fungsi dari sistem sudah tercakup.

2. *Activity Diagram*

Activity diagram menjelaskan perilaku dinamis dari sistem atau bagian bagian sistem melalui aliran proses yang dilakukan sistem. Hal ini sama dengan model *flowchart* tetapi sedikit berbeda dikarenakan *activity diagram* dapat menampilkan aliran proses sistem secara bersamaan.



Gambar 2.7: *Activity Diagram* [18]

Komponen utama dari *activity diagram* adalah *action node*, diwakili oleh bulat persegi panjang, yang sesuai dengan tugas yang dilakukan oleh software panah dari satu *node* ke *node* lain menjelaskan aliran kontrol. Hal ini berarti bahwa setelah *action* pertama selesai, *action* kedua baru dijalankan. Sebuah lingkaran hitam pekat menjelaskan awal proses *activity* dimulai. Sebuah titik hitam yang dikelilingi lingkaran hitam menjelaskan akhir dari *activity process*. Garis horizontal berwarna hitam merupakan pemisah dua *action* atau lebih secara bersamaan.

2. *Data Flow Diagram (DFD)*

Data Flow Diagram (DFD) merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan DFD adalah memudahkan pemakai atau *user* yang kurang menguasai bidang

komputer untuk mengerti sistem yang akan dikerjakan [2].

1. Diagram Konteks (*Context Diagram*)

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem. Sistem dibatasi oleh *boundary* dan hanya ada satu proses. Tidak ada *store* dalam diagram konteks.

2. Diagram Zero (*Overview Diagram*)

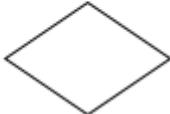
Diagram *zero* adalah diagram yang menggambarkan proses dari *data flow diagram*. Diagram *zero* memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan *external entity*. Pada level ini sudah dimungkinkan adanya *data store* yang digunakan.

3. Diagram Rinci (*Level Diagram*)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada di dalam diagram *zero* atau diagram level di atasnya.

3. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) merupakan sebuah diagram yang menggunakan beberapa notasi untuk menggambarkan data dalam bentuk entitas-entitas beserta hubungan yang terbentuk antar data tersebut[9]. Untuk menggambarkan ERD digunakan beberapa notasi dan simbol. ERD berisi komponen-komponen himpunan entitas dan relasi yang masing-masing dilengkapi dengan atribut yang merepresentasikan seluruh fakta dari dunia nyata yang kita tinjau dapat digambarkan dengan lebih sistematis [5].

No.	Simbol	Keterangan
1		<i>Entitas</i> , merupakan representasi dari sebuah objek tempat di mana sistem menyimpan data
2		<i>Atribut</i> , merupakan data yang dihubungkan dengan entitas. Digunakan untuk menjelaskan, mengidentifikasi, dan mengekspresikan hubungan antar entitas
3		<i>Relasi</i> , adalah hubungan yang terjadi antar suatu atau lebih entitas

Gambar 2.8: Simbol *Entity Relationship Diagram*

4. Struktur Navigasi

Struktur navigasi adalah struktur atau urutan alur dari suatu program yang mendefinisikan objek-objek dan merancang tampilan layar agar tidak menimbulkan kerancuan informasi. Menentukan struktur navigasi merupakan hal yang sebaiknya dilakukan sebelum membuat suatu aplikasi. Ada 4 macam bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan aplikasi, yaitu [4]:

1. Struktur Navigasi Linier

Pemakai menelusuri program secara berurutan atau satu arah. Biasanya digunakan untuk membuat multimedia presentasi. Bentuknya seperti pada Gambar 2.9.

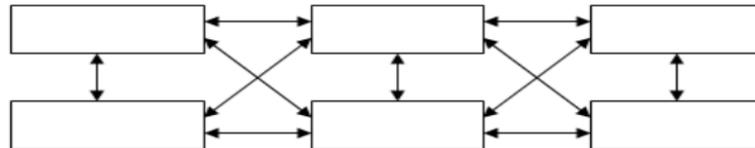


Gambar 2.9: Struktur Navigasi Linier [4]

2. Struktur Navigasi Non-Linier

Non linier merupakan pengembangan dari struktur navigasi linier dan pemakai

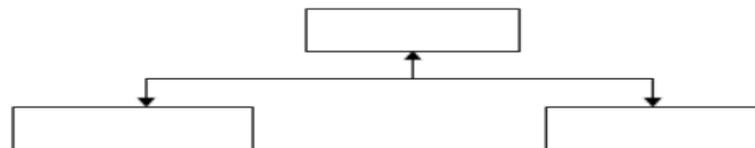
bebas menelusuri program tanpa dibatasi oleh suatu rute (tidak berurutan). Percabangan Non Linier berbeda dengan percabangan Hirarki, percabangan non linier tiap tampilannya mempunyai kedudukan yang sama tidak ada *master page* dan *slave page*, seperti pada Gambar 2.10.



Gambar 2.10: Struktur Navigasi Non-Linier [4]

3. Struktur Navigasi Hirarki

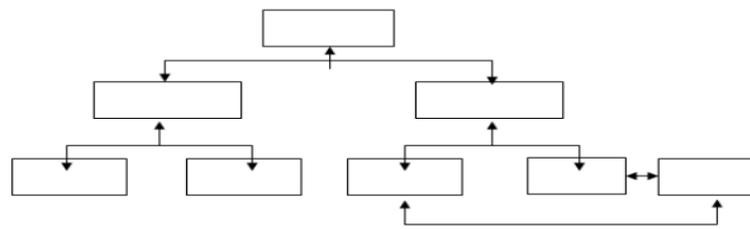
Hirarki menelusuri melalui titik-titik percabangan dari suatu struktur pohon. Tampilan pada menu pertama disebut sebagai *master page* (halaman utama ke-satu) yang mempunyai halaman percabangan yang dikaitkan dengan *slave page* (halaman utama kedua), seperti pada Gambar 2.11.



Gambar 2.11: Struktur Navigasi Hirarki [4]

4. Struktur Navigasi Campuran

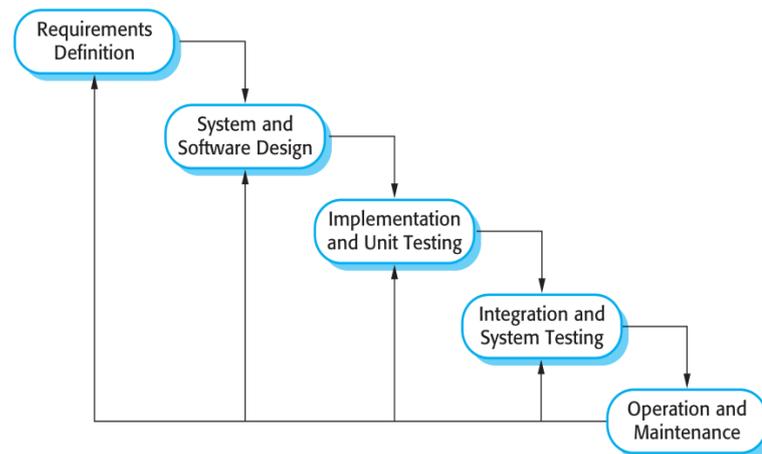
Campuran merupakan gabungan dari struktur navigasi linier, non linier dan hirarki. Pemakai dapat dengan bebas menelusuri program, tetapi pada bagian tertentu gerakan dibatasi secara hirarki ataupun linier, seperti pada Gambar 2.12.



Gambar 2.12: Struktur Navigasi Campuran [4]

E. Metode *Waterfall*

Metode *waterfall* merupakan metode proses pengembangan perangkat lunak yang pertama kali dipublikasikan diambil dari proses pengembangan sistem yang lebih umum (Royce, 1970). Metode *waterfall*, sering disebut dengan *classic life cycle*, memberikan pendekatan yang bersifat sistematis dan berurutan dalam pengembangan perangkat lunak [18]. Metode *waterfall* merupakan sebuah contoh proses yang menitikberatkan pada perencanaan, sehingga pada prinsipnya, aktivitas-aktivitas proses harus direncanakan dan dijadwalkan sebelum aktivitas tersebut dilakukan [6].



Gambar 2.13: Metode *Waterfall* [6]

Tahap-tahap dari metode *waterfall* merefleksikan aktivitas fundamental dari pengembangan perangkat lunak, yaitu sebagai berikut [6]:

1. *Requirement analysis and definition*

Layanan, batasan, dan tujuan dari sistem telah dibentuk dengan cara berkonsultasi dengan user. Elemen-elemen tersebut kemudian didefinisikan secara detail dan berfungsi sebagai spesifikasi sistem.

2. *System and Software Design*

Proses perancangan sistem mengalokasikan kebutuhan-kebutuhan ke dalam perangkat lunak dan perangkat keras sistem dengan cara membentuk arsitektur sistem secara keseluruhan. Perancangan sistem melibatkan proses identifikasi dan menjelaskan abstraksi fundamental sistem berikut hubungannya.

3. *Implementation and Unit Testing*

Pada tahap ini, perancangan sistem direalisasikan ke dalam kumpulan program atau unit program. Unit testing melibatkan proses verifikasi terhadap setiap unit agar sesuai dengan spesifikasinya.

4. *Integration and System Testing*

Unit program maupun program yang terpisah diintegrasikan dan diuji sebagai kesatuan sistem untuk memastikan bahwa kebutuhan dari perangkat lunak tersebut telah dipenuhi. Setelah pengujian, perangkat lunak tersebut dapat dikirimkan ke pelanggan.

5. *Operation and Maintenance*

Biasanya (walaupun tidak selalu), tahap ini merupakan tahap terpanjang dari SDLC. Sistem dipasang dan digunakan secara nyata. Tahap pemeliharaan melibatkan pembetulan kesalahan yang tidak ditemukan pada tahap pengembangan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

E. Framework CodeIgniter

CodeIgniter merupakan aplikasi *open source* yang merupakan salah satu *framework PHP* dengan menggunakan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. *CodeIgniter* pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc., sebuah perusahaan yang memproduksi CMS (*Content Management System*) yang bernama *Expression Engine* pada 28 Februari 2006. Saat ini, *CodeIgniter* dikembangkan oleh *British Columbia Institute of Technology*. Versi stabil terakhir dari *CodeIgniter* adalah versi 3.0.4 yang dirilis pada 13 Januari 2016 [20].

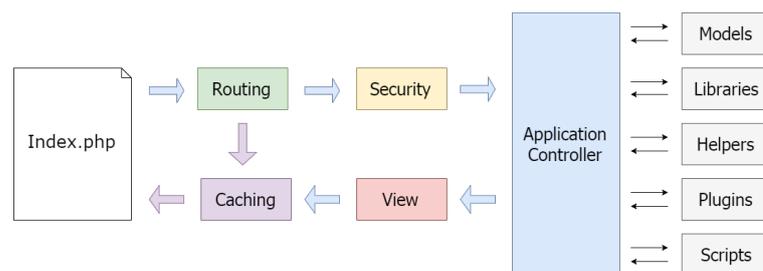
Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi atau prosedur-prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang *programmer*, tanpa harus membuat fungsi atau *class* dari awal. Beberapa alasan mengapa menggunakan *Framework* [20]:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi web.
2. Relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah framework (dengan syarat *programmer* mengikuti pola standar yang ada).
3. Umumnya *framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, *pagination*, *multiple database*, *scaffolding*, *pengaturan session*, *error handling*, dan lain-lain).
4. Lebih bebas dalam pengembangan jika dibandingkan CMS.

MVC merupakan suatu konsep yang cukup populer dalam pembangunan apli-

kasi web, dengan memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, *user interface*, dan bagian yang menjadi kontrol aplikasi. Komponen yang membangun suatu *MVC pattern* dalam suatu sistem yaitu [20]:

1. *View*, merupakan bagian yang menangani *presentation logic*. Pada suatu aplikasi web bagian ini biasanya berupa *file template HTML*, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.
2. *Model*, biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view*.
3. *Controller*, merupakan bagian yang mengatur hubungan antara bagian *model* dan bagian *view*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.



Gambar 2.14: Flowchart Framework CodeIgniter [20]

Ada beberapa kelebihan *CodeIgniter* (CI) dibandingkan dengan *Framework PHP* lain [20]:

1. Performa sangat cepat, salah satu alasan tidak menggunakan *framework* adalah karena eksekusinya yang lebih lambat daripada *PHP from the scratch*, tapi Co-

deIgniter sangat cepat bahkan mungkin bisa dibilang *codeigniter* merupakan *framework* yang paling cepat dibanding *framework* yang lain.

2. Konfigurasi yang sangat minim (*nearly zero configuration*), tentu saja untuk menyesuaikan dengan database dan keleluasaan *routing* tetap diizinkan melakukan konfigurasi dengan mengubah beberapa *file* konfigurasi seperti *database.php* atau *autoload.php*, namun untuk menggunakan *CodeIgniter* dengan *setting standard*, anda hanya perlu mengubah sedikit saja *file* pada folder *config*.
3. Banyak komunitas, dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.
4. Dokumentasi yang sangat lengkap, setiap paket instalasi *CodeIgniter* sudah disertai *user guide* yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

G. Web Server XAMPP

Web server adalah tempat penyimpanan aplikasi web kemudian mengaksesnya melalui internet. *Web server* dibutuhkan untuk *server side script* seperti PHP, yang artinya setiap perubahan yang dilakukan dapat ditampilkan dengan browser. Selain itu, untuk mengefisienkan waktu, dapat menggunakan *web server* lokal sehingga jika ada perubahan yang terjadi, hanya dengan *save* dan kemudian *refresh browser* [16].

Ada banyak *web server* yang bisa menjadi pilihan salah satunya adalah XAMPP. XAMPP berasal dari singkatan X yang berarti *cross platform* karena XAMPP bisa dijalankan di Windows, Linux, Mac dan sebagainya; A yang berarti *Apache* sebagai *web server* nya; M yang berarti MySQL sebagai *Database Management System*; dan PP yang berarti PHP dan Perl sebagai bahasa yang didukungnya.

H. Database MySQL

Basis data atau *database* adalah kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat. *Data Base Management System* (DBMS) adalah aplikasi yang dipakai untuk mengelola basis data. DBMS memiliki beberapa kemampuan yang terintegrasi seperti membuat, menghapus, menambah, dan memodifikasi basis data [16]. Salah satu aplikasi DBMS yang masih digunakan hingga saat ini adalah *MySQL Server*.

MySQL adalah suatu perangkat lunak sistem manajemen basis data yang menggunakan *Structured Query Language* (SQL) yakni bahasa yang berisi perintah-perintah untuk memanipulasi database, seperti menampilkan, menambahkan, menghapus, dan mengedit *database*.

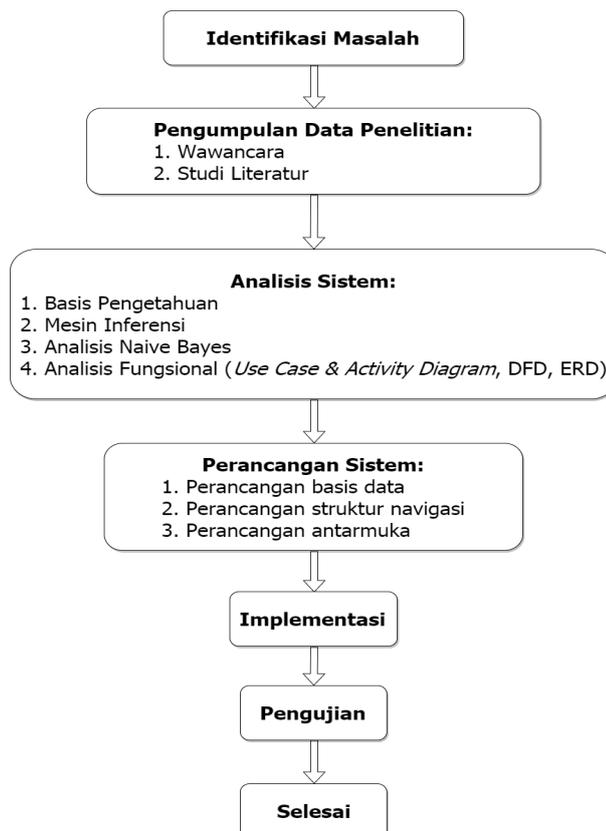
Keunggulan MySQL adalah kemampuannya dalam menyediakan berbagai fasilitas atau fitur-fitur yang dapat digunakan oleh bermacam-macam *user*. Sebagai sebuah *database management system*, MySQL mampu menangani beberapa instruksi sekaligus dari beberapa *user* dalam satu waktu serta merekam semua data *user* dalam tabel pada sistem [22].

BAB III

DESAIN MODEL

A. Tahapan Penelitian

Tahapan penelitian adalah langkah-langkah yang dilakukan dalam memperoleh berbagai data untuk diproses menjadi informasi yang lebih akurat sesuai dengan permasalahan yang akan diteliti. Berikut merupakan tahapan penelitian yang dilakukan dalam penelitian skripsi yang berjudul “Rancang Bangun Sistem Pakar Diagnosa Penyakit Gigi dan Mulut pada Manusia Menggunakan Metode *Forward Chaining* dan *Naive Bayes* Berbasis Web ”.



Gambar 3.1: Tahapan Penelitian

B. Pengumpulan Data

Metode pengumpulan data terdiri dari wawancara dan studi pustaka.

1. Wawancara

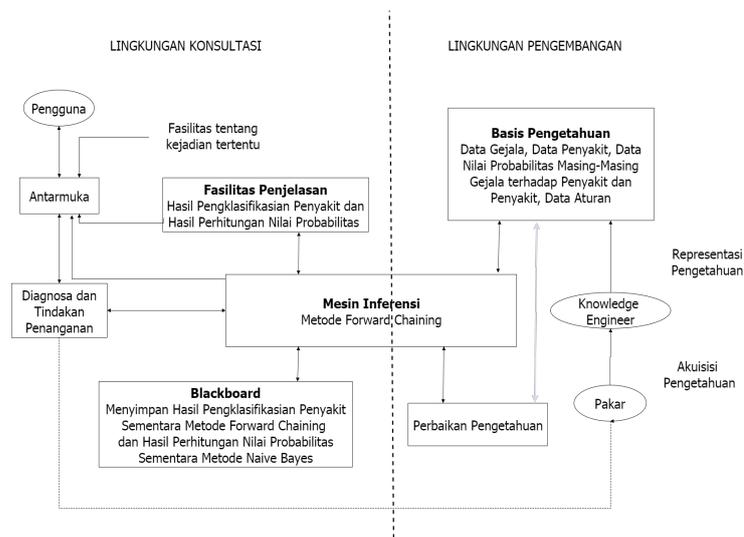
Proses wawancara dilakukan kepada 3 narasumber yang merupakan para ahli atau pakar dalam hal ini dokter gigi dan mulut untuk mendapatkan informasi yang dibutuhkan mengenai permasalahan dan hal-hal yang dibutuhkan selama proses pembuatan dan pengembangan aplikasi yaitu data yang akurat mengenai gejala dan penyakit gigi dan mulut beserta nilai probabilitasnya dan tindakan penanganan yang dapat dilakukan.

2. Studi Pustaka

Studi pustaka dilakukan dengan cara membaca dan mempelajari data informasi dari buku-buku, artikel-artikel, jurnal-jurnal dan bahan-bahan dari internet yang terkait dengan pokok bahasan penelitian.

C. Analisa Sistem

Analisa sistem berisi langkah kerja dari sistem secara menyeluruh, baik dari segi model maupun dari segi arsitektur sistem pakar yang akan dibangun. Tujuannya adalah untuk mempermudah implementasi dan pengujian sistem pakar diagnosa penyakit gigi dan mulut menggunakan metode *Forward Chaining* dan *Naive Bayes*. Langkah kerja dalam sistem pakar disesuaikan dengan arsitektur sistem pakar.



Gambar 3.2: Arsitektur Sistem Pakar Diagnosa Penyakit Gigi dan Mulut

Pengguna sistem adalah pengguna umum, pakar, dan *knowledge engineer*. Fasilitas penjelasan digunakan untuk memperjelas hasil diagnosa dan menjelaskan proses pengambilan keputusan. Basis pengetahuan menyimpan data-data yang digunakan dalam mesin inferensi untuk melakukan penalaran. Mesin inferensi akan memproses data tersebut menggunakan metode *Forward Chaining* dan memproses nilai peluang penyakit menggunakan metode *Naive Bayes* sehingga dapat menghasilkan keputusan diagnosa sistem.

1. Akuisisi Pengetahuan

Akuisisi pengetahuan merupakan proses transfer informasi dari pakar untuk mengambil pengetahuan. Pada tahap ini dilakukan penyeleksian data yang akan digunakan dalam membangun aplikasi sistem pakar. Banyaknya jenis penyakit gigi dan mulut yang ditemukan, membuat penulis membatasi fokus penyakit yang akan dijadikan basis pengetahuan dari penyakit gigi dan mulut yang banyak dialami oleh masyarakat Indonesia. Berdasarkan hasil wawancara dengan pakar, sistem ini dapat mendiagnosis 15 jenis penyakit pada gigi dan mulut dengan masing-masing gejala

yang menyertainya. 15 jenis penyakit yang dapat didiagnosis diantaranya adalah:

Tabel 3.1: Daftar Penyakit

No.	Penyakit
1	Abrasi Gigi
2	<i>Abses Periapikal</i>
3	<i>Abses Periodontal</i>
4	<i>Caries (Gigi Berlubang)</i>
5	<i>Cheilitis</i>
6	<i>Gingivitis</i>
7	<i>Halitosis (Bau Mulut)</i>
8	<i>Herpes</i>
9	<i>Hipersensitif Dentin (Gigi Sensitif)</i>
10	<i>Malocclusion</i>
11	<i>Periodontitis</i>
12	Persistensi Gigi Sulung
13	<i>Pulpitis Reversible</i>
14	<i>Pulpitis Irreversible</i>
15	<i>Stomatitis (Sariawan)</i>

2. Basis Pengetahuan

Dari hasil proses akuisisi pengetahuan yang telah diuraikan di atas, maka selanjutnya dapat disusun suatu tabel yang menampung basis pengetahuan yang diperoleh dari pakar (*human expert*) melalui tahap *knowledge transfer* kepada penulis selaku *knowledge engineer*. Basis pengetahuan tersebut terdiri dari basis pengetahuan penyakit beserta tindakan penanganannya dan basis pengetahuan gejala. Basis-basis

pengetahuan ini merupakan sumber pengetahuan (*knowledge*) dipakai oleh sistem. Oleh karena itu, pemberian kode pada masing-masing basis pengetahuan diperlukan untuk membedakan suatu sumber pengetahuan terhadap sumber pengetahuan yang lain.

Berikut ini Tabel 3.2-3.7 berisikan tentang kode penyakit, jenis-jenis penyakit, dan tindakan penanganannya.

Tabel 3.2: Tabel Kode Penyakit dan Tindakan Penanganannya

Kode	Penyakit	Penanganan
P1	Abrasi Gigi	Perawatan melakukan penambalan pada gigi yang abrasi menggunakan bahan tumpat GIC (<i>Glass Ionomer Cement</i>).
P2	<i>Abses Periapikal</i>	Perawatan menghilangkan infeksi dan membuang nanah melalui bedah mulut atau dengan pengobatan saluran akar (PSA). Untuk membantu menghilangkan infeksi biasanya diberikan antibiotik. Tindakan yang paling penting adalah mencabut pulpa yang terkena dan membuang nanahnya.
P3	<i>Abses Periodontal</i>	Perawatan pada <i>abses periodontal</i> akut dilakukan dengan mengurangi gejala, mengendalikan penyebaran infeksi dan melakukan <i>drainase</i> melalui saku periodontal dan insisi eksternal. Pada perawatan <i>abses periodontal</i> kronis dilakukan dengan <i>scalling</i> dan penyerutan akar atau terapi bedah, seperti gingivektomi atau

Tabel 3.3: Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 1

Kode	Penyakit	Penanganan
		prosedur bedah <i>flep</i> . Perawatan antibiotik sistemik diperlukan pada <i>abses periodontal</i> akut maupun kronis yang bertujuan untuk mencegah penyebaran bakteri dan komplikasi yang serius.
P4	<i>Caries</i> (Gigi Berlubang)	Perawatan yang diberikan adalah pembersihan jaringan gigi yang terkena karies dan penambalan (restorasi). Bahan tambal yang digunakan dapat bermacam-macam, misalnya <i>resin komposit</i> (penambalan dengan sinar dan bahannya sewarna gigi), <i>glass ionomer cement</i> , <i>komposer</i> , atau <i>amalgam</i> (sudah mulai jarang digunakan). Pada lubang gigi yang besar dibutuhkan restorasi yang lebih kuat, biasanya digunakan <i>inlay</i> atau <i>onlay</i> , bahkan mungkin mahkota tiruan. Pada karies yang sudah mengenai jaringan pulpa, perlu dilakukan perawatan saluran akar.
P5	<i>Cheilitis</i>	Perawatan dengan antijamur topikal seperti <i>nystatin</i> , <i>clotrimazole</i> , atau <i>econazole</i> juga dapat digunakan.
P6	<i>Gingivitis</i>	Perawatan dengan menjaga <i>oral hygiene</i> yaitu menyikat gigi dengan cara yang benar 2 kali sehari sesudah sarapan pagi dan malam sebelum tidur, bersihkan gigi dengan menggunakan benang gigi dengan cara yang benar, gunakan

Tabel 3.4: Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 2

Kode	Penyakit	Penanganan
		obat kumur antiseptik, kontrol ke dokter gigi setiap 6 bulan sekali untuk membersihkan karang gigi.
P7	<i>Halitosis</i> (Bau Mulut)	Perawatan dengan cara menjaga kebersihan mulut menyikat gigi dengan cara yang benar dan menggunakan pembersih lidah (pembersih berbentuk huruf U) untuk mengurangi <i>filamen</i> di lidah dan membersihkan partikel makanan untuk menangkal perkembangbiakan bakteri dan menggunakan obat kumur antiseptik non-alkohol.
P8	<i>Herpes</i>	Perawatan dengan pemberian salep <i>anestetik</i> untuk meredakan rasa nyeri di kulit seperti salep yang mengandung <i>lidocaine</i> . Obat herpes mulut yang diberikan secara <i>oral</i> atau suntikan <i>intra vena</i> dapat diberikan pada orang dengan sistem kekebalan tubuh rendah atau pada bayi umur < 6 bulan, atau pada orang yang menderita penyakit parah.
P9	<i>Hipersensitif Dentin</i> (Gigi Sensitif)	Perawatan <i>hipersensitif dentin</i> , yakni mencegah aliran cairan <i>tubulus dentin</i> dan mengurangi rangsangan terhadap syaraf. Berdasarkan berat ringannya dilakukannya, terapi <i>hipersensitif dentin</i> dapat bersifat invasif dan non invasif.

Tabel 3.5: Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 3

Kode	Penyakit	Penanganan
		Terapi <i>hipersensitif dentin</i> yang bersifat invasif antara lain bedah <i>mukogingival</i> , <i>resin</i> dan <i>pulpektomi</i> serta <i>laser</i> . Sedangkan terapi yang bersifat non invasif antara lain pasta desensitasi dan bahan topikal.
P10	<i>Malocclusion</i>	Perawatan khusus dengan menggunakan alat-alat ortodontik atau alat cekat (<i>braces</i>) yang harus diaplikasikan oleh dokter gigi spesialis ortodontik. Pemakaian alat cekat pada anak dan remaja umumnya untuk memperbaiki penampilan/estetis. Sebaliknya, orang dewasa memakai alat cekat lebih untuk memperbaiki fungsi pengunyahan.
P11	<i>Periodontitis</i>	Perawatan <i>periodontal</i> non bedah meliputi pemeliharaan kebersihan mulut, <i>Scaling</i> dan <i>Root planing</i> (SRP) dan pemberian antibiotik untuk mencegah, menghentikan serta eliminasi penyakit <i>periodontal</i> yang merupakan <i>initial phase therapy</i> . <i>Kuretase</i> yaitu mengeruk dinding <i>gingiva</i> dari kantung gusi untuk membuang jaringan lunak yang tidak sehat. Mengeluarkan jaringan lunak yang meradang dan merupakan dinding <i>lateral</i> dari <i>poket</i> .
P12	Persistensi Gigi Sulung	Perawatan melakukan pemeriksaan ke dokter gigi agar dilakukan pencabutan pada gigi sulung.

Tabel 3.6: Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 4

Kode	Penyakit	Penanganan
P13	<i>Pulpitis Reversible</i>	Perawatan terbaik untuk <i>pulpitis reversibel</i> adalah pencegahan. Perawatan periodik untuk mencegah perkembangan karies, penumpatan awal bila <i>kavitas</i> meluas dan lain sebagainya. Apabila dijumpai <i>pulpitis reversibel</i> , penghilangan stimulasi (jejas) biasanya sudah cukup, begitu gejala telah reda, gigi harus di tes vitalitasnya untuk memastikan bahwa tidak terjadi <i>nekrosis</i> .
P14	<i>Pulpitis Irreversible</i>	Perawatannya adalah <i>eksterpasi</i> , untuk kemudian dilakukan <i>pulpektomi</i> . <i>Prognosa</i> untuk pulpa adalah baik, bila iritasi diambil cukup dini, kalau tidak kondisinya dapat berkembang menjadi <i>pulpitis irreversible</i> .
P15	<i>Stomatitis (Sariawan)</i>	Perawatan merupakan tindakan simptomatik dengan tujuan untuk mengurangi gejala, mengurangi jumlah dan ukuran <i>ulkus</i> , dan meningkatkan periode bebas penyakit. Perawatan terbaik adalah yang akan mengendalikan ulkus untuk waktu selama mungkin dengan efek samping yang minimal. Tujuan perawatan harus ditentukan dengan keparahan penyakit (rasa sakit), riwayat medis pasien, frekuensi rasa sakit diantara kunjungan dan kemampuan pasien untuk mentoleransi pengobatan. Pada pasien dengan SAR,

Tabel 3.7: Tabel Kode Penyakit dan Tindakan Penanganan Lanjutan 5

Kode	Penyakit	Penanganan
		penting untuk menentukan faktor-faktor predisposisi dan merawat faktor-faktor tersebut, kapanpun mungkin dilakukan, sebelum memulai terapi yang lebih spesifik.

Tabel di atas merupakan basis pengetahuan penyakit dengan masing-masing tindakan penanganannya. Selanjutnya, basis pengetahuan kode gejala beserta gejala-gejala penyakit pada gigi dan mulut terdapat pada Tabel 3.8-3.10.

Tabel 3.8: Tabel Kode Gejala-Gejala Penyakit

Kode	Gejala Penyakit
G1	Bau mulut tak sedap
G2	Bentuk gigi tampak terkikis
G3	Bentuk gusi agak membulat/tumpul
G4	Bibir kering
G5	Bibir pecah-pecah
G6	Bibir terasa perih
G7	Bibir mudah berdarah
G8	Gigi berjejal
G9	Gigi goyang
G10	Gigi renggang
G11	Gigi sulung yang tak kunjung tanggal
G12	Gigi terasa sakit atau berdenyut

Tabel 3.9: Tabel Kode Gejala-Gejala Penyakit Lanjutan 1

Kode	Gejala Penyakit
G13	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin
G14	Gigi tonggos
G15	Gigi yang berlubang terasa sakit bila masuk makanan
G16	Gigi/gusi bernanah
G17	Gingsul
G18	Gusi yang turun membuat gigi terlihat lebih panjang
G19	Gusi meradang
G20	Gusi mudah berdarah
G21	Gusi tampak merah dan bengkak
G22	Gusi terasa sakit/nyeri
G23	Nyeri saat luka tersentuh
G24	Nyeri saat membuka mulut
G25	Nyeri saat menggigit
G26	Nyeri saat mengunyah
G27	Nyeri sampai ke daerah sinus, pelipis, mata atau telinga
G28	Pembengkakan kelenjar getah bening
G29	Rasa sakit tajam hanya sebentar
G30	Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum
G31	Terdapat gelembung (vesikel) di dalam rongga mulut
G32	Terdapat lubang pada gigi
G33	Terdapat luka berbentuk oval atau bulat yang berwarna putih/kuning dan tepi yang merah di dalam rongga mulut

Tabel 3.10: Tabel Kode Gejala-Gejala Penyakit Lanjutan 2

Kode	Gejala Penyakit
G34	Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah
G35	Terjadi demam
G36	Timbulnya benih gigi dengan posisi yang abnormal
G37	Timbulnya bercak coklat, hitam atau putih pada gigi

Dari data 15 jenis penyakit serta 37 gejala-gejala yang ada, terdapat aturan-aturan yang diterapkan sebelum sistem dapat mencapai kesimpulan. Aturan-aturan ini meliputi hubungan suatu gejala terhadap suatu penyakit serta nilai probabilitas kemunculan gejala pada suatu penyakit dan nilai probabilitas kemunculan suatu penyakit. Aturan-aturan tersebut tidak didapatkan dari metode *Forward Chaining* maupun metode *Naive Bayes* melainkan hak mutlak dari pakar berdasarkan pengalaman (*experience*) dan teori ilmu yang ada pada pakar.

Dengan demikian, berikut adalah tabel aturan relasi atau hubungan gejala terhadap suatu penyakit beserta nilai probabilitasnya:

Tabel 3.11: Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya

Penyakit	Gejala Penyakit	Probabilitas
Abrasi Gigi	Bentuk gigi tampak terkikis	0.90
	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau,dingin	0.73
<i>Abses Periapikal</i>	Gusi tampak merah dan bengkak	0.83
	Gigi terasa sakit atau berdenyut	0.73
	Terjadi demam	0.60
	Nyeri saat mengunyah	0.60
	Pembengkakan kelenjar getah bening	0.53
<i>Abses Periodontal</i>	Gusi tampak merah dan bengkak	0.73
	Nyeri saat mengunyah	0.73
	Gusi atau gigi bernanah	0.50
	Nyeri saat membuka mulut	0.43
<i>Caries (Gigi Berlubang)</i>	Terdapat lubang pada gigi	1.00
	Timbulnya bercak coklat, hitam atau putih pada gigi	0.97
	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0.90
	Gigi terasa sakit atau berdenyut	0.80
	Nyeri saat menggigit	0.70
<i>Cheilitis</i>	Bibir kering	0.70
	Bibir terasa perih	0.67
	Bibir pecah-pecah	0.67
	Bibir mudah berdarah	0.60

Tabel 3.12: Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya Lanjutan 1

Penyakit	Gejala Penyakit	Probabilitas
<i>Gingivitis</i>	Bentuk gusi agak membulat/tumpul	0.80
	Gusi mudah berdarah	0.80
	Gusi meradang	0.77
	Bau mulut tak sedap	0.70
	Gusi tampak merah dan bengkak	0.63
	Gusi terasa sakit/nyeri	0.50
<i>Halitosis</i> (Bau Mulut)	Bau mulut tak sedap	1.00
<i>Herpes</i>	Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum	0.77
	Terjadi demam	0.70
	Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah	0.70
	Gusi mudah berdarah	0.67
	Terdapat gelembung (<i>vesikel</i>) di dalam rongga mulut	0.60
	Pembengkakan kelenjar getah bening	0.60
	Gusi tampak merah dan bengkak	0.53
<i>Hipersensitive Dentin</i>	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0.93
<i>Malocclusion</i>	Gigi berjejal	0.87
	Gigi tonggos	0.70
	Gigi Renggang	0.60
	Gingsul	0.57

Tabel 3.13: Tabel Aturan Kemunculan Suatu Gejala Terhadap Penyakit dan Probabilitasnya Lanjutan 2

Penyakit	Gejala Penyakit	Probabilitas
<i>Periodontitis</i>	Gigi goyang	0.77
	Bau mulut tak sedap	0.73
	Gusi tampak merah dan bengkak	0.70
	Gusi yang turun membuat gigi terlihat lebih panjang	0,67
	Gusi mudah berdarah	0.63
Persistensi Gigi Sulung	Gigi sulung yang tak kunjung tanggal	0.93
	Timbulnya benih gigi dengan posisi yang abnormal	0.77
<i>Pulpitis Reversible</i>	Terdapat lubang pada gigi	0.77
	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0.73
	Rasa sakit tajam hanya sebentar	0.73
<i>Pulpitis Irreversible</i>	Terdapat lubang pada gigi	0.73
	Gigi terasa sakit atau berdenyut	0,70
	Nyeri sampai ke daerah sinus, pelipis, mata atau telinga	0.60
	Gigi yang berlubang terasa sakit bila masuk makanan	0.60
<i>Stomatitis (Sariawan)</i>	Nyeri saat luka tersentuh	0.90
	Terdapat luka berbentuk oval atau bulat yang berwarna putih/kuning dan tepi yang merah di dalam rongga mulut	0.83

Selain aturan hubungan antara kemunculan gejala yang ada dengan suatu penyakit beserta nilai probabilitasnya, maka dibutuhkan juga aturan nilai probabilitas dari kemunculan suatu penyakit yang juga diperoleh dari pakar. Berikut adalah tabelnya:

Tabel 3.14: Tabel Probabilitas Kemunculan Suatu Penyakit

Penyakit	Probabilitas
Abrasi Gigi	0.57
<i>Abses Periapikal</i>	0.57
<i>Abses Periodontal</i>	0.30
<i>Caries (Gigi Berlubang)</i>	0.83
<i>Cheilitis</i>	0.17
<i>Gingivitis</i>	0.70
<i>Halitosis (Bau Mulut)</i>	0.50
<i>Herpes</i>	0.17
<i>Hipersensitif Dentin (Gigi Sensitif)</i>	0.60
<i>Malocclusion</i>	0.60
<i>Periodontitis</i>	0.30
Persistensi Gigi Sulung	0.70
<i>Pulpitis Reversible</i>	0.50
<i>Pulpitis Irreversible</i>	0.70
<i>Stomatitis (Sariawan)</i>	0.40

3. Mesin Inferensi

Mesin inferensi adalah program komputer yang dibuat untuk memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace* untuk memformulasikan kesimpulan. Dengan kata lain, mesin inferensi mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini akan menganalisa suatu masalah tertentu dan kemudian mencari jawaban atau kesimpulan yang terbaik. Dari fakta-fakta yang diperoleh selama proses tanya jawab dengan *user*, serta aturan-aturan yang tersimpan di basis pengetahuan, mesin inferensi dapat menarik suatu kesimpulan dan memberikan rekomendasi atau saran yang diharapkan oleh *user*.

Mesin inferensi dalam sistem ini menggunakan metode *Forward Chaining* yang merupakan suatu strategi pengambilan keputusan yang dimulai dari bagian sebelah kiri (IF terlebih dahulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis. Sehingga aturan-aturan dalam basis pengetahuan dapat diuraikan seperti di bawah ini:

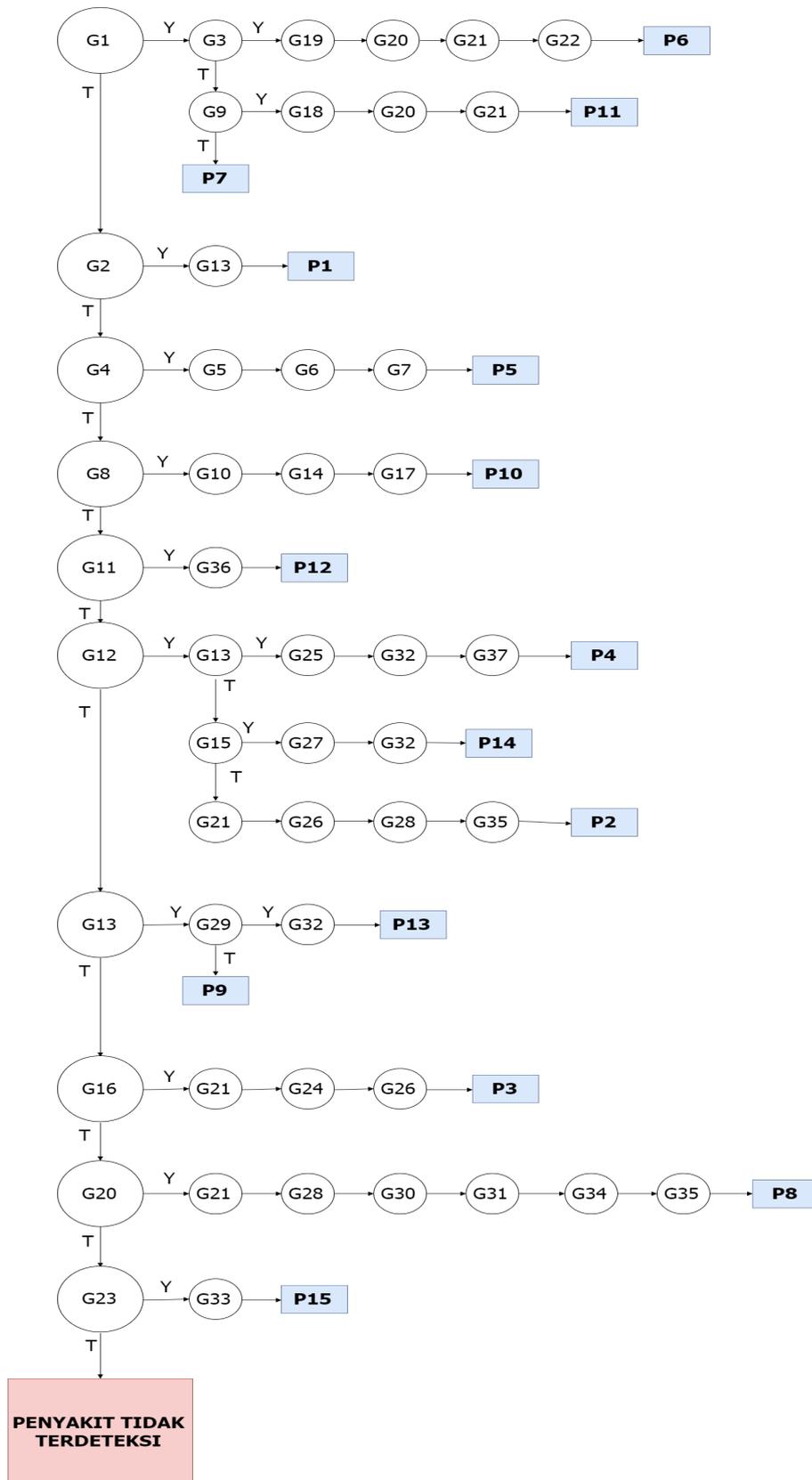
1. R1: IF Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin AND Bentuk gigi tampak terkikis THEN Abrasi Gigi
2. R2: IF Gusi tampak merah dan bengkak AND Gigi terasa sakit atau berdenyut AND Nyeri saat mengunyah AND Terjadi demam AND Pembengkakan kelenjar getah bening THEN Abses Periapikal
3. R3: IF Gusi tampak merah dan bengkak AND Gusi atau gigi bernanah AND Nyeri saat mengunyah AND Nyeri saat membuka mulut THEN Abses Periodontal
4. R4: IF Timbulnya bercak coklat, hitam atau putih pada gigi AND Terdapat lubang pada gigi AND Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin AND Gigi terasa sakit atau berdenyut AND Nyeri saat menggigit THEN Karies

5. R5: IF Bibir kering AND Bibir pecah-pecah AND Bibir terasa perih AND Bibir mudah berdarah THEN Cheilitis
6. R6: IF Gusi meradang AND Gusi tampak merah dan bengkak AND Gusi mudah berdarah AND Bau mulut tak sedap AND Gusi terasa sakit/nyeri AND Bentuk gusi agak membulat/tumpul THEN Gingivitis
7. R7: IF Bau mulut tak sedap THEN Halitosis
8. R8: IF Terjadi demam AND Gusi tampak merah dan bengkak AND Gusi mudah berdarah AND Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum AND Terdapat gelembung (vesikel) di dalam rongga mulut AND Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah AND Pembengkakan kelenjar getah bening THEN Herpes
9. R9: IF Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin THEN Hipersensitif Dentin
10. R10: IF Gigi berjejal AND Gingsul AND Gigi tonggos AND Gigi renggang THEN Malocclusion
11. R11: IF Gusi tampak merah dan bengkak AND Gusi mudah berdarah AND Bau mulut tak sedap AND Gusi yang turun membuat gigi terlihat lebih panjang AND Gigi goyang THEN Periodontitis
12. R12: IF Gigi sulung yang tak kunjung tanggal AND Timbulnya benih gigi dengan posisi yang abnormal THEN Persistensi Gigi Sulung
13. R13: IF Terdapat lubang pada gigi AND Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin AND Rasa sakit tajam hanya sebentar THEN Pulpitis Reversible
14. R14: IF Terdapat lubang pada gigi AND Gigi terasa sakit atau berdenyut AND Gigi yang berlubang terasa sakit bila masuk makanan AND Nyeri sampai ke daerah sinus, pelipis, mata atau telinga THEN Pulpitis Irreversible
15. R15: IF Terdapat luka berbentuk oval atau bulat yang berwarna putih/kuning dan tepi yang merah di dalam rongga mulut AND Nyeri saat luka tersentuh THEN Stomatitis

Tabel 3.16: Tabel Relasi Gejala Terhadap Penyakit Lanjutan

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
G16			*												
G17										*					
G18											*				
G19						*									
G20						*		*			*				
G21		*	*			*		*			*				
G22						*									
G23															*
G24			*												
G25				*											
G26		*	*												
G27														*	
G28		*						*							
G29													*		
G30								*							
G31								*							
G32				*									*	*	
G33															*
G34								*							
G35		*						*							
G36												*			
G37				*											

Dalam sistem pakar, dibutuhkan alur penalaran untuk menerapkan aturan-aturan yang ada dengan relasi yang telah ditentukan seperti pada diagram berikut:



Gambar 3.3: Diagram Penalaran Sistem

Keterangan tentang penomoran kode:

1. Nama Gejala

- G1 Bau mulut tak sedap
- G2 Bentuk gigi tampak terkikis
- G3 Bentuk gusi agak membulat/tumpul
- G4 Bibir kering
- G5 Bibir pecah-pecah
- G6 Bibir terasa perih
- G7 Bibir mudah berdarah
- G8 Gigi berjejal
- G9 Gigi goyang
- G10 Gigi renggang
- G11 Gigi sulung yang tak kunjung tanggal
- G12 Gigi terasa sakit atau berdenyut
- G13 Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin
- G14 Gigi tonggos
- G15 Gigi yang berlubang terasa sakit bila masuk makanan
- G16 Gigi/gusi bernanah
- G17 Gingsul
- G18 Gusi yang turun membuat gigi terlihat lebih panjang
- G19 Gusi meradang
- G20 Gusi mudah berdarah
- G21 Gusi tampak merah dan bengkak
- G22 Gusi terasa sakit/nyeri
- G23 Nyeri saat luka tersentuh
- G24 Nyeri saat membuka mulut
- G25 Nyeri saat menggigit
- G26 Nyeri saat mengunyah
- G27 Nyeri sampai ke daerah sinus, pelipis, mata atau telinga
- G28 Pembengkakan kelenjar getah bening
- G29 Rasa sakit tajam hanya sebentar
- G30 Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum
- G31 Terdapat gelembung (vesikel) di dalam rongga mulut
- G32 Terdapat lubang pada gigi
- G33 Terdapat luka berbentuk oval atau bulat yang berwarna putih/kuning dan tepi yang merah di dalam rongga mulut
- G34 Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah
- G35 Terjadi demam
- G36 Timbulnya benih gigi dengan posisi yang abnormal
- G37 Timbulnya bercak coklat, hitam atau putih pada gigi

2. Nama Penyakit

- P1 Abrasi Gigi
- P2 Abses Periapikal
- P3 Abses Periodontal
- P4 Caries (Gigi Berlubang)
- P5 Cheilitis
- P6 Gingivitis
- P7 Halitosis (Bau Mulut)
- P8 Herpes
- P9 Hipersensitif Dentin (Gigi Sensitif)
- P10 Malocclusion
- P11 Periodontitis
- P12 Persistensi Gigi Sulung
- P13 Pulpitis Reversible
- P14 Pulpitis Irreversible
- P15 Stomatitis (Sariawan)

Sistem akan memulai pertanyaan dengan menampilkan gejala pertama yang ada pada basis pengetahuan yaitu G1. Kemudian jawaban *user* akan menentukan pertanyaan selanjutnya, jika *user* menjawab “Ya” maka pertanyaan selanjutnya adalah menampilkan gejala-gejala dari seluruh kemungkinan penyakit yang mengandung gejala G1, yaitu gejala-gejala pada penyakit P6 dan P11. Jika *user* memenuhi semua aturan gejala pada penyakit P6 maka dapat disimpulkan bahwa *user* menderita penyakit P6. Sebaliknya jika *user* memenuhi semua aturan gejala pada penyakit P11 maka dapat disimpulkan bahwa *user* menderita penyakit P11. Jika *user* tidak memiliki gejala yang dimiliki oleh penyakit P6 dan P11, maka dapat disimpulkan *user* hanya mengalami gejala G1 dengan kesimpulan *user* menderita penyakit P7. Akan tetapi jika pada awal *user* memilih “Tidak” pada gejala G1, maka sistem akan menampilkan gejala selanjutnya yaitu gejala G2 dan seterusnya hingga aturan pada sistem habis.

5. Analisis *Naive Bayes*

Metode yang digunakan untuk melakukan diagnosa penyakit gigi dan mulut pada manusia ini menggunakan metode *Naive Bayes* yang dilakukan dengan cara mengklasifikasikan penyakit. Metode ini digunakan untuk mengatasi ketidakpastian dengan nilai probabilitas berkisar antara 0 hingga 1. Ketidakpastian di sini adalah munculnya gejala yang sama dari penyakit yang berbeda. *Naive Bayes* akan menghitung nilai probabilitas setiap gejala yang sama dari penyakit yang berbeda. Nilai probabilitas munculnya gejala jika diketahui suatu penyakit dan nilai probabilitas penyakit tanpa memandang gejala apapun didapat langsung melalui wawancara dengan pakar atau dokter gigi.

Contoh kasus adalah seorang *user A* ingin melakukan konsultasi tentang penyakit gigi dan mulut dengan menjawab pertanyaan seputar gejala yang dialaminya yang diberikan oleh sistem dengan jawaban “Ya” atau “Tidak”. Pertanyaan pertama pada awal diagnosa adalah “Apakah bau mulut tidak sedap?”. Metode inferensi yang digunakan untuk melakukan penelusuran adalah forward chaining.

Contoh pengklasifikasian penyakit:

1. Apakah bau mulut tidak sedap? Jika user A menjawab “Ya”, maka pertanyaan selanjutnya,
2. Apakah bentuk gusi agak membulat atau tumpul? Jika user A menjawab “Tidak”, maka pertanyaan selanjutnya,
3. Apakah gigi goyang? Jika user A menjawab “Ya”, maka pertanyaan selanjutnya,
4. Apakah gusi yang turun membuat gigi terlihat lebih panjang? Jika user A menjawab “Ya”, maka pertanyaan selanjutnya,
5. Apakah gusi mudah berdarah? Jika user A menjawab “Ya”, maka pertanyaan

selanjutnya,

6. Apakah gusi tampak merah dan bengkak? Jika user A menjawab “Ya”, maka sistem akan menampilkan gejala-gejala yang dialami oleh *user A*, yaitu :
- a. Bau mulut tak sedap (G1)
 - b. Gigi goyang (G9)
 - c. Gusi yang turun membuat gigi terlihat lebih panjang (G18)
 - d. Gusi tampak merah dan bengkak (G21)
 - e. Gusi mudah berdarah (G20)

Selanjutnya sistem melakukan proses perhitungan nilai maksimum penyakit dengan metode *Naive Bayes* berikut ini.

$$P(c|E) = \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{j=1}^n P(E_j|c)$$

Dengan

$P(c|E)$: probabilitas kelas c jika diberikan atribut (*evidence*) E

$P(c)$: probabilitas awal kelas c (tanpa memandang atribut apapun)

$P(E_j|c)$: probabilitas munculnya atribut (*evidence*) E_j dalam kelas c

Proses Perhitungan *Naive Bayes*

1. Periodontitis (P11)
 - = $P(11)*P(G1|P11)*(G9|P11)*P(G18|P11)*P(G20|P11)*P(G21|P11)$
 - = $0.30*0.73*0.77*0.67*0.63*0.70$
 - =0.12376

Dari proses klasifikasi melalui metode *Forward Chaining* didapat penyakit yaitu “*Periodontitis*” dengan nilai maksimum yang diperoleh dari metode *Naive Bayes* sebesar 0.12376 atau 12.4%.

Hasil diagnosa:

Anda menderita penyakit *Periodontitis* dengan nilai naive bayes sebesar 0.12376, selanjutnya sistem akan menampilkan hasil diagnosa penyakit gigi dan mulut *user A* sebagai berikut.

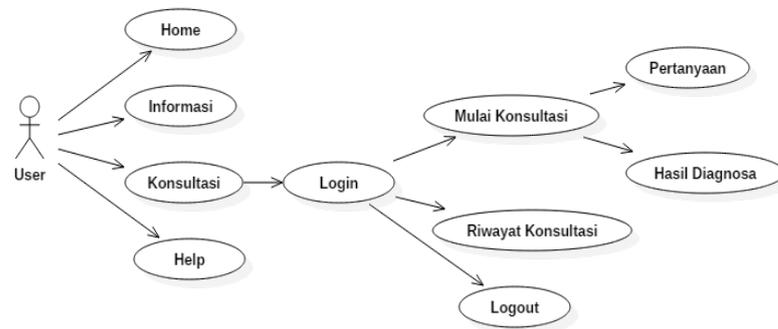
Solusi penyakit:

Perawatan periodontal non bedah meliputi pemeliharaan kebersihan mulut, *Scaling* dan *Root Planing* (SRP) dan pemberian antibiotik untuk mencegah, menghentikan serta mengeliminasi penyakit periodontal yang merupakan *intial phase therapy* dan kuretase yaitu mengeruk dinding gingiva dari kantung gusi untuk membuang jaringan lunak yang tidak sehat. Mengeluarkan jaringan lunak yang meradang dan merupakan dinding lateral dari poket.

6. Analisis Fungsional

1. Use Case Diagram

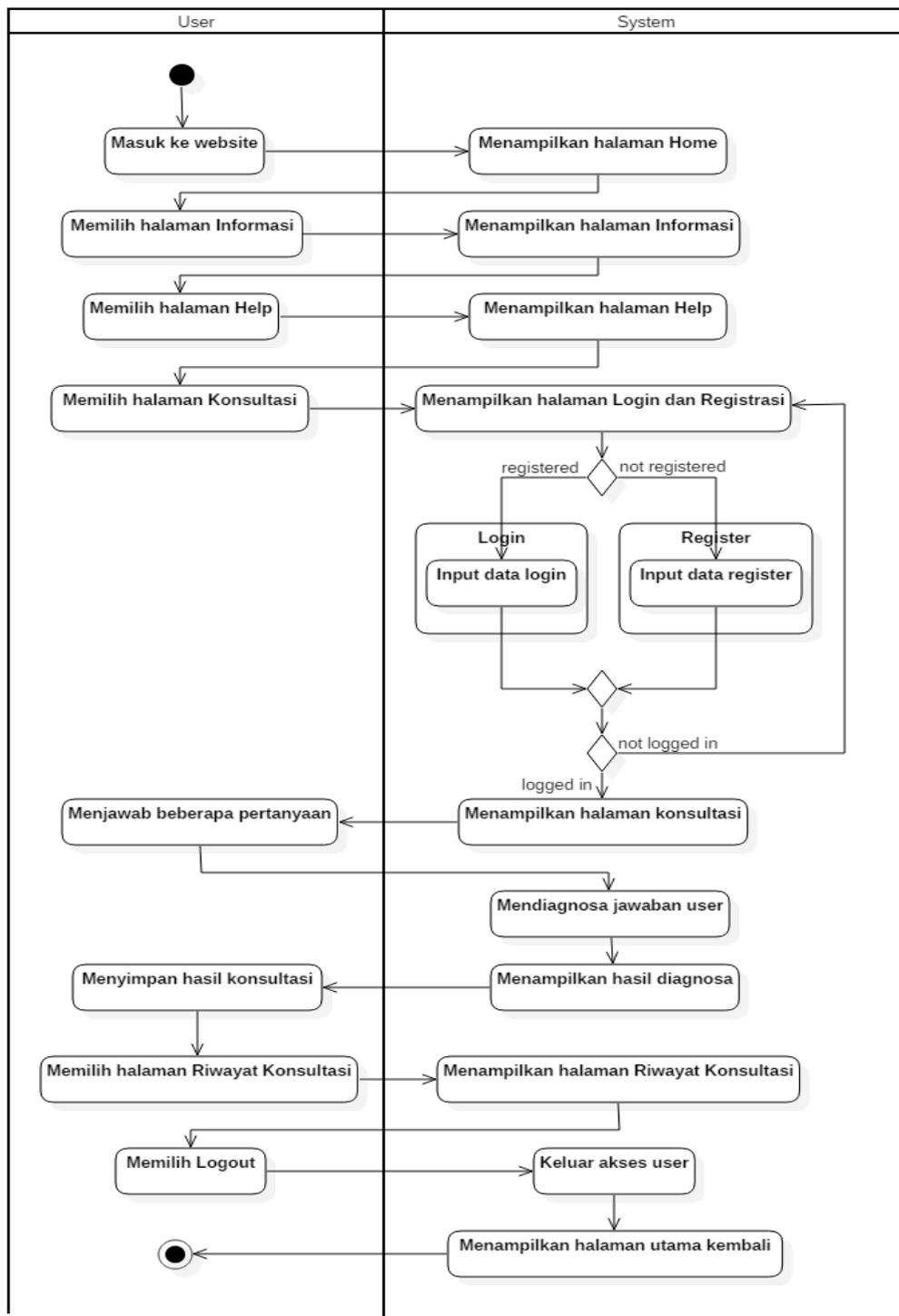
Pada rancangan *use case diagram* terdapat actor, sebuah karakter yang dipresentasikan sebagai *user*. *Use case* digunakan sebagai gambaran fungsi pada suatu sistem, dan garis penghubung antara *actor* dan *use case*. *User* pada aplikasi ini adalah pengguna aplikasi yang akan melakukan *main action* yaitu konsultasi. Untuk melakukan konsultasi, diperlukan proses *Login* atau Registrasi yang bertujuan untuk menyimpan data *user*. *User* dapat melakukan konsultasi, melihat riwayat konsultasi dan *Logout*. Gambaran *use case diagram* terdapat pada Gambar 3.4.



Gambar 3.4: *Use Case Diagram User*

2. Activity Diagram

Activity diagram adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini digunakan untuk menjelaskan proses bisnis dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem. Diagram ini menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Pada *activity diagram* proses awal perancangan alur digambarkan dengan *InitialState* dan diakhiri dengan *FinalState*. Sedangkan objek yang ada, yaitu sebuah *activities* untuk menggambarkan tampilan yang ada pada aplikasi ini. Hubungan setiap *activities* dihubungkan dengan *transition* menggunakan fungsi *control flow*. Gambar 3.5 merupakan rancangan *activity diagram* pada aplikasi sistem pakar ini.



Gambar 3.5: Activity Diagram User

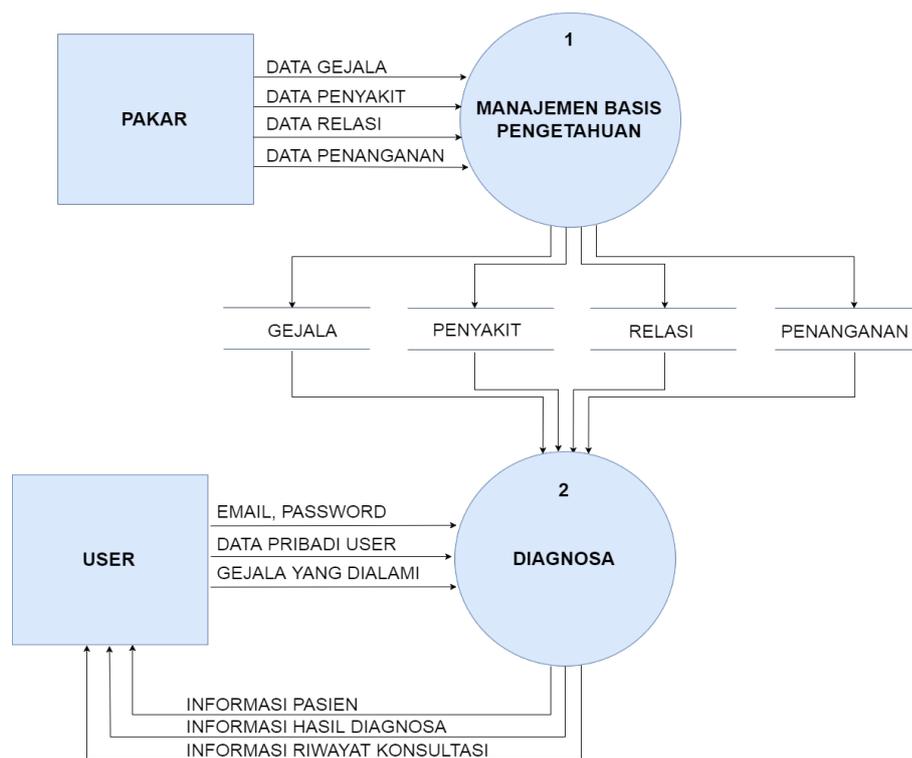
3. Data Flow Diagram (DFD)

a. Diagram Konteks



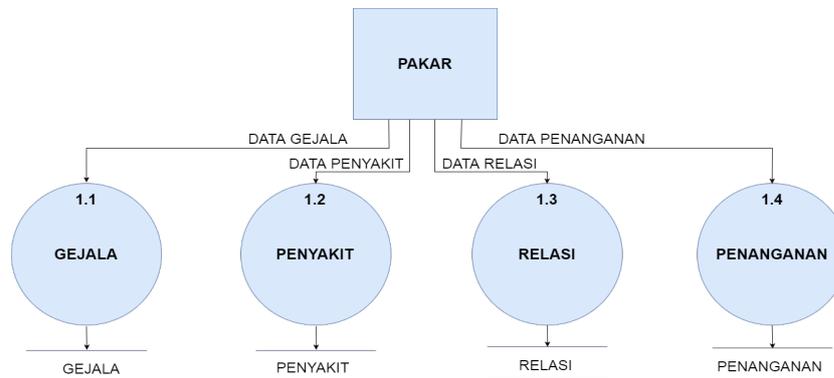
Gambar 3.6: Diagram Konteks

b. DFD level 0



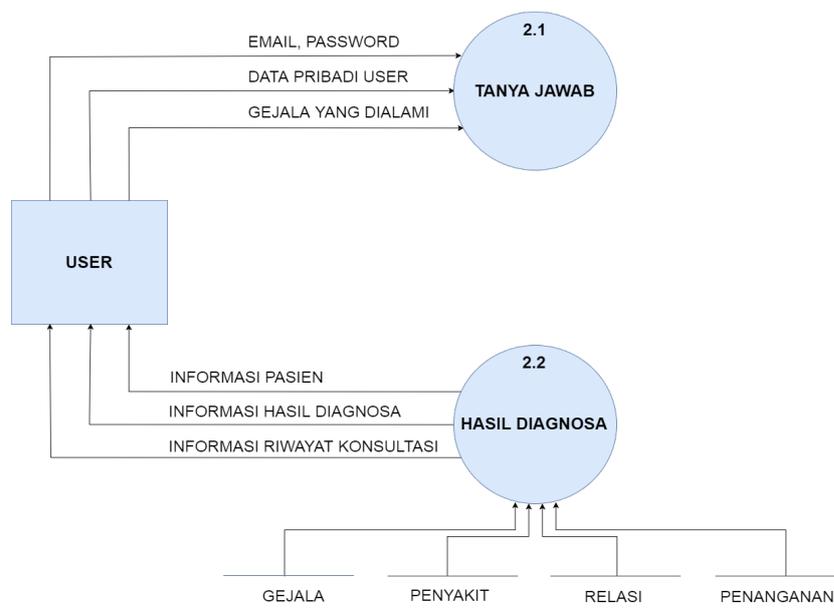
Gambar 3.7: DFD level 0

c. DFD level 1 Proses Manajemen Basis Pengetahuan



Gambar 3.8: DFD level 1 Proses Manajemen Basis Pengetahuan

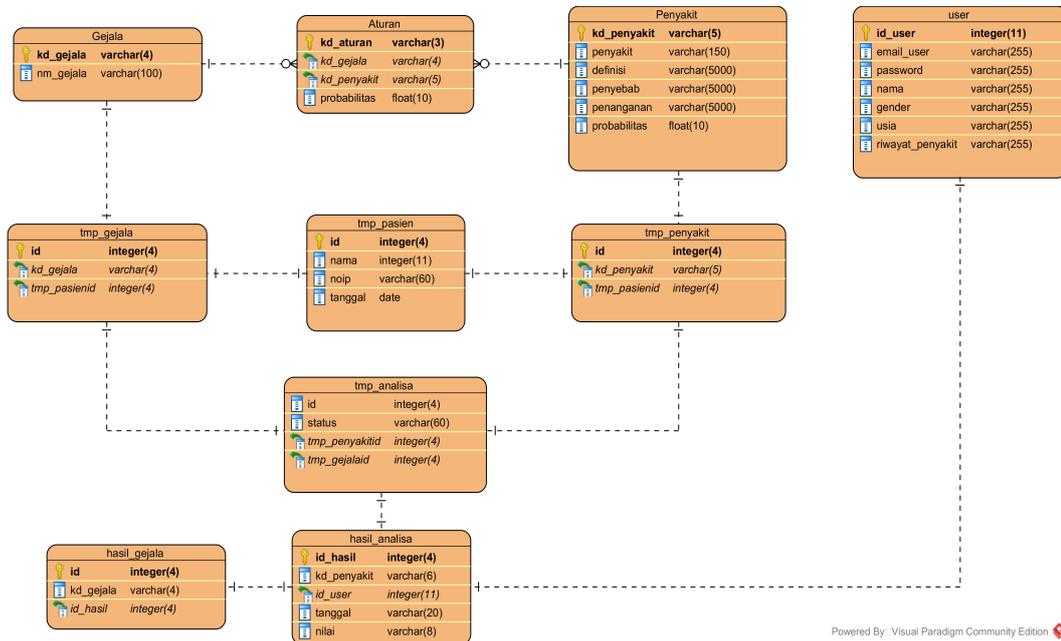
d. DFD level 1 Proses Diagnosa



Gambar 3.9: DFD level 1 Proses Diagnosa

4. *Entity Relationship Diagram (ERD)*

Entity relationship diagram merupakan suatu desain sistem yang digunakan untuk merepresentasikan, menentukan dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan *database*. Dalam perancangan sistem ini telah dibuat ERD yang merupakan diagram relasi antar tabel pada *database* yang sesuai dengan rancangan dari sistem pakar penyakit gigi dan mulut.



Gambar 3.10: Entity Relationship Diagram

D. Perancangan Sistem

1. Perancangan Database

Pada tahap perancangan aplikasi sistem pakar ini database disimpan dalam MySQL yang berfungsi sebagai tempat penyimpanan data user dan hasil diagnosa serta data gejala, penyakit, penanganan, dan nilai probabilitas sebagai basis pengetahuan dalam sistem pakar ini. Tabel-tabel yang dibutuhkan dalam pembuatan aplikasi sistem pakar ini sebagai berikut.

1. Tabel *User*

Tabel *user* merupakan tabel yang digunakan untuk menyimpan data *user* dari sistem penyakit gigi dan mulut dengan *primary key* *id_user*. Struktur tabel *user* ditunjukkan pada Tabel 3.17.

Tabel 3.17: Database User

No.	Name	Type	Length/Value	Action
1	id_user	int	11	<i>Primary key</i>
2	email	varchar	255	
3	password	varchar	255	
4	nama	varchar	255	
5	gender	varchar	255	
6	riwayat_penyakit	text		

2. Tabel Gejala

Tabel gejala merupakan tabel yang digunakan untuk menyimpan data gejala penyakit gigi dan mulut dengan *primary key* kd_gejala. Struktur tabel gejala ditunjukkan pada Tabel 3.18.

Tabel 3.18: Database Gejala

No.	Name	Type	Length/Value	Action
1	kd_gejala	varchar	4	<i>Primary key</i>
2	nm_gejala	varchar	100	

3. Tabel Penyakit

Tabel penyakit merupakan tabel yang digunakan untuk menyimpan data penyakit gigi dan mulut dengan *primary key* kd_penyakit. Struktur tabel penyakit ditunjukkan pada Tabel 3.19.

Tabel 3.19: Database Penyakit

No.	Name	Type	Length/Value	Action
1	kd_penyakit	varchar	5	<i>Primary key</i>
2	penyakit	text		
3	deskripsi	text		
4	penyebab	text		
5	penanganan	text		
6	probabilitas	float		

4. Tabel Aturan

Tabel aturan diagnosa merupakan tabel yang digunakan untuk menyimpan data aturan gejala dengan penyakit gigi dan mulut yang terkait serta nilai probabilitasnya. Struktur tabel penanganan ditunjukkan pada Tabel 3.20.

Tabel 3.20: Database Aturan

No.	Name	Type	Length/Value	Action
1	kd_aturan	int	11	<i>Primary Key</i>
2	kd_penyakit	varchar	5	
3	kd_gejala	char	5	
4	probabilitas	float		

5. Tabel *Temporary*

Tabel *temporary* merupakan tabel yang digunakan untuk menyimpan data yang bersifat sementara. Tabel *temporary* digunakan saat *user* sedang melakukan konsultasi. Tabel *temporary* dalam sistem ini terbagi menjadi tabel tmp_user,

tmp_gejala, tmp_penyakit, dan tmp_analisa.

Tabel 3.21: *Database Temporary User*

No.	Name	Type	Length/Value	Action
1	id	int	4	<i>Primary key</i>
2	nama	varchar	60	
3	kelamin	enum	'P ', 'W '	
4	alamat	varchar	100	
5	pekerjaan	varchar	60	
6	noip	varchar	60	
7	tanggal	datetime		

Tabel 3.22: *Database Temporary Gejala*

No.	Name	Type	Length/Value	Action
1	id	int	4	<i>Primary key</i>
2	kd_gejala	varchar	4	
3	noip	varchar	60	

Tabel 3.23: *Database Temporary Penyakit*

No.	Name	Type	Length/Value	Action
1	id	int	4	<i>Primary key</i>
2	kd_penyakit	varchar	4	
3	noip	varchar	60	

Tabel 3.24: *Database Temporary Analisa*

No.	Name	Type	Length/Value	Action
1	noip	varchar	60	<i>Primary key</i>
2	kd_gejala	varchar	4	
3	kd_penyakit	varchar	4	
4	status	enum	'Y', 'N'	

6. Tabel Hasil

Tabel Hasil merupakan tabel yang digunakan untuk menampilkan data hasil konsultasi yang dilakukan oleh *user*. Tabel *temporary* ini juga digunakan saat *user* melihat riwayat konsultasi. Tabel hasil dalam sistem ini terbagi menjadi tabel hasil_gejala dan hasil_analisa.

Tabel 3.25: *Database Hasil Gejala*

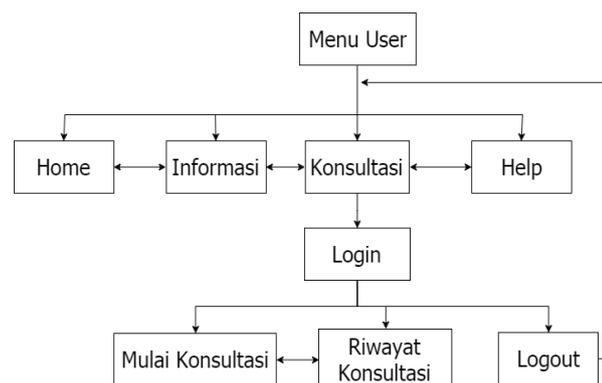
No.	Name	Type	Length/Value	Action
1	id	int	4	<i>Primary key</i>
2	kd_gejala	varchar	4	
	id_hasil	int	4	

Tabel 3.26: Database Hasil Analisa

No.	Name	Type	Length/Value	Action
1	id_hasil	int	4	<i>Primary key</i>
2	kd_penyakit	varchar	6	
3	id_user	int	5	
4	tanggal	datetime		
5	nilai	varchar	8	

2. Perancangan Struktur Navigasi

Pembuatan program aplikasi sistem pakar penyakit pada gigi dan mulut berbasis web ini membutuhkan struktur navigasi untuk mengetahui alur atau jalannya program aplikasi yang dibuat. Struktur navigasi yang dipakai oleh penulis pada pembuatan program aplikasi ini adalah struktur navigasi campuran, yaitu gabungan antara struktur navigasi non linier dan hirarki.

**Gambar 3.11:** Struktur Navigasi

Pada Gambar 3.11 di atas adalah ketika *user* menjalankan aplikasi maka akan muncul tampilan menu utama yang memiliki 4 menu yaitu *Home*, *Informasi*, *Konsultasi*, dan *Help*. Setiap menu memiliki fungsi tersendiri. Jika *user* memilih menu

Home maka *user* akan menuju ke halaman yang berisi tentang web sistem pakar. Jika *user* memilih menu Informasi maka *user* akan menuju ke halaman definisi-definisi dari anatomi gigi dan macam-macam penyakit gigi dan mulut pada manusia. Jika *user* memilih menu *Help* maka akan menuju halaman bantuan mengenai informasi penggunaan aplikasi sistem pakar ini. Jika *user* memilih menu Konsultasi maka *user* akan menuju halaman *Login* atau Registrasi sebelum memulai konsultasi. Jika proses registrasi dan/atau *login* sudah selesai maka *user* akan menuju halaman utama Konsultasi. Pada halaman tersebut *user* akan memilih setiap gejala dengan menjawab “Ya” atau “Tidak” pada pertanyaan-pertanyaan yang terdapat pada sistem. Setelah proses memilih selesai maka *user* akan mendapatkan halaman tampilan hasil konsultasi. Jika *user* memilih halaman Riwayat Kosultasi maka akan menuju halaman yang menampilkan data *user* dan riwayat konsultasi penyakit yang sudah pernah dilakukan oleh *user*. Jika *user* memilih *Logout*, maka sistem akan menampilkan kembali halaman utama *website*.

3. Perancangan Antarmuka

Bagian ini akan menjelaskan tentang rancangan antarmuka yang akan digunakan pada aplikasi sistem pakar ini. Adapun yang akan dijelaskan pada bagian ini adalah tampilan *home*, tampilan informasi, tampilan register/login, tampilan konsultasi, tampilan hasil konsultasi, tampilan riwayat konsultasi, dan tampilan *help*.

1. Perancangan Halaman *Home*

HEADER
Home Informasi Konsultasi Help
Selamat Datang
FOOTER

Gambar 3.12: Tampilan *Home*

2. Perancangan Halaman Informasi

HEADER
Home Informasi Konsultasi Help
Konten Halaman Informasi
FOOTER

Gambar 3.13: Tampilan Informasi

3. Perancangan Halaman Registrasi/*Login*

HEADER
Home Informasi Konsultasi Help
Sign Up
Email : <input type="text"/>
Password : <input type="text"/>
Nama : <input type="text"/>
Gender : <input type="radio"/> Laki-Laki <input type="radio"/> Perempuan
Usia : <input type="text"/>
Riwayat Penyakit : <input type="text"/>
<input type="button" value="Sign Up"/>
FOOTER

Gambar 3.14: Tampilan Register

HEADER
Home Informasi Konsultasi Help
Sebelum memulai konsultasi, silahkan Login .
LOGIN
Username : <input type="text"/>
Password : <input type="text"/>
<input type="button" value="Login"/>
Belum punya akun? Daftar di sini.
FOOTER

Gambar 3.15: Tampilan *Login*

4. Perancangan Halaman Konsultasi

HEADER
Home Informasi Konsultasi Riwayat Konsultasi Logout Help
Halaman Konsultasi Apakah anda mengalami gejala Nama Gejala <input type="radio"/> Ya <input type="radio"/> Tidak <input type="button" value="Jawab"/>
FOOTER

Gambar 3.16: Tampilan Konsultasi

5. Perancangan Halaman Hasil Konsultasi

HEADER
Home Informasi Konsultasi Riwayat Konsultasi Logout Help
Hasil Analisa Data User : <input type="text"/> Hasil Analisa Penyakit : <input type="text"/> <input type="button" value="Simpan Hasil Konsultasi"/> <input type="button" value="Coba Konsultasi Lagi"/>
FOOTER

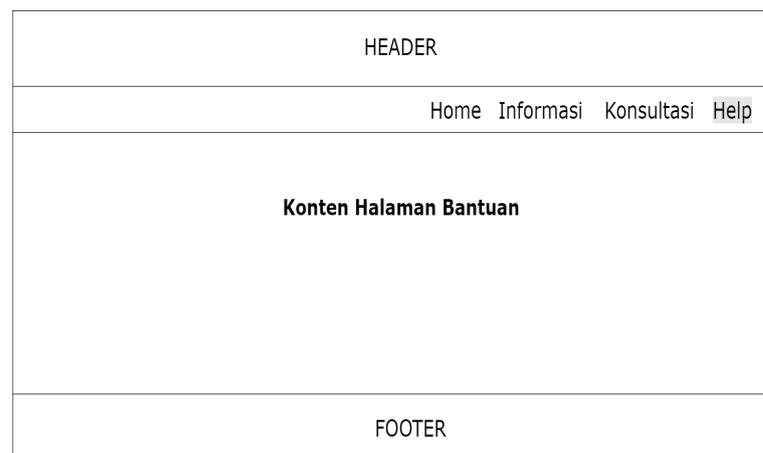
Gambar 3.17: Tampilan Hasil Konsultasi

6. Perancangan Halaman Riwayat Konsultasi



Gambar 3.18: Tampilan Riwayat Konsultasi

7. Perancangan Halaman *Help*



Gambar 3.19: Tampilan *Help*

BAB IV

PEMBAHASAN

A. Implementasi

Implementasi merupakan tahap penyusunan perangkat lunak sistem (*coding*) untuk mengetahui apakah sistem yang dikembangkan telah menghasilkan tujuan yang diinginkan dengan melakukan pengkodean dari hasil analisa dan perancangan kedalam sistem. Adapun modul pakar yang akan dibuat *user interfacenya* telah dijelaskan pada bab III membutuhkan perangkat pendukung, perangkat tersebut berupa perangkat lunak dan perangkat keras.

1. Perangkat Keras (*Hardware*)

Dalam pembuatan aplikasi ini dibutuhkan perangkat keras komputer. Berikut ini perangkat keras yang digunakan untuk pembuatan dan menjalankan program:

- a. *Processor* : Intel Core i3 @ 1.8 GHz
- b. RAM : 10 GB
- c. *Harddisk* : 500 GB

2. Analisis Kebutuhan Perangkat Lunak (*Software*)

Selain perangkat keras yang dibutuhkan pada pembuatan aplikasi ini, dibutuhkan juga perangkat lunak yang digunakan untuk membuat sistem pakar ini, dengan spesifikasi sebagai berikut:

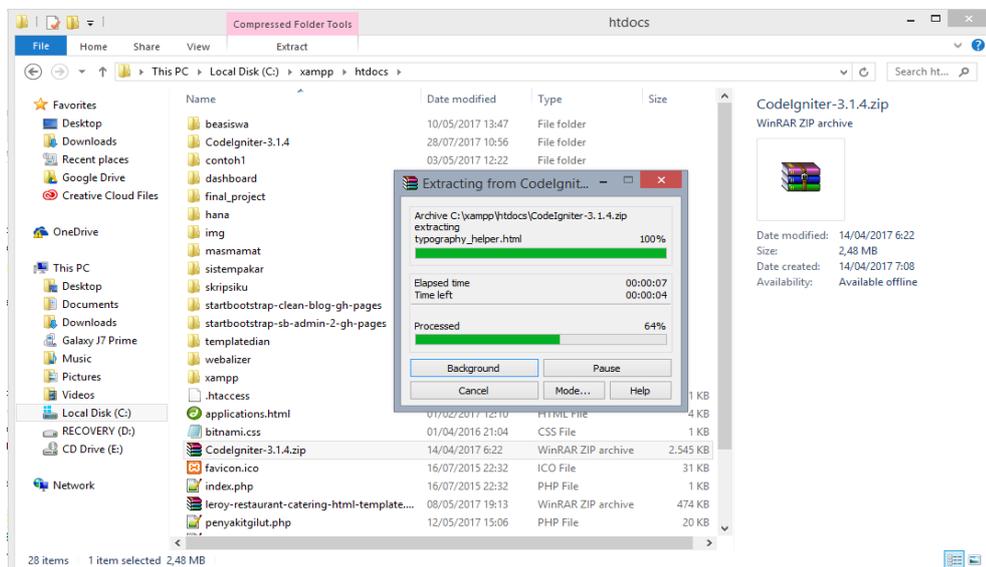
- a. Sistem Operasi Windows 8.1 64-bit
- b. *Web browser*, seperti Mozilla Firefox atau Google Chrome
- c. *Framework CodeIgniter*
- d. *Web Server* XAMPP
- e. *Database* MySQL

1. Pembuatan Aplikasi

Berisi tentang langkah-langkah implementasi dari analisa dan perancangan ke dalam aplikasi ini. Berikut pembuatan Sistem Pakar Diagnosa Penyakit Gigi dan Mulut. Pengkodean program dilakukan dengan menggunakan *Framework CodeIgniter*.

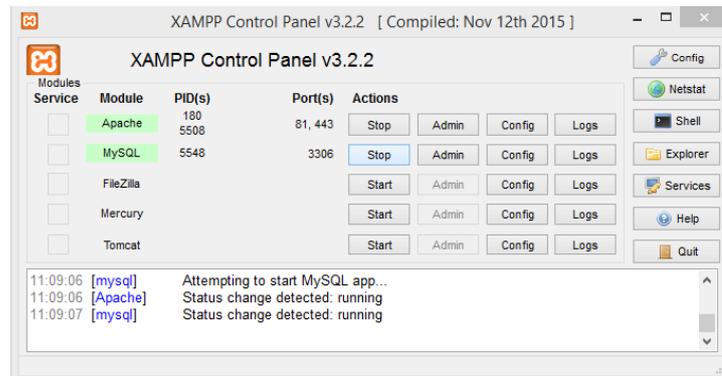
1. Instalasi Framework Codeigniter

Download package Codeigniter di <http://codeigniter.com> lalu *extract package* tersebut dalam *file* htdocs pada XAMPP.



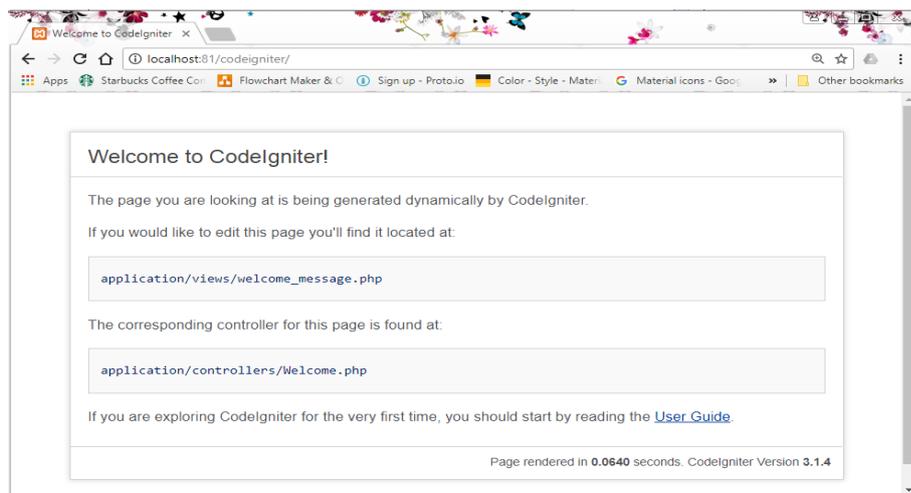
Gambar 4.1: Instalasi Framework Codeigniter

Rename folder codeigniter yang sudah *diextract* untuk memudahkan pemanggilan nama folder di browser. Lalu nyalakan modul Apache dan MySQL pada XAMPP.



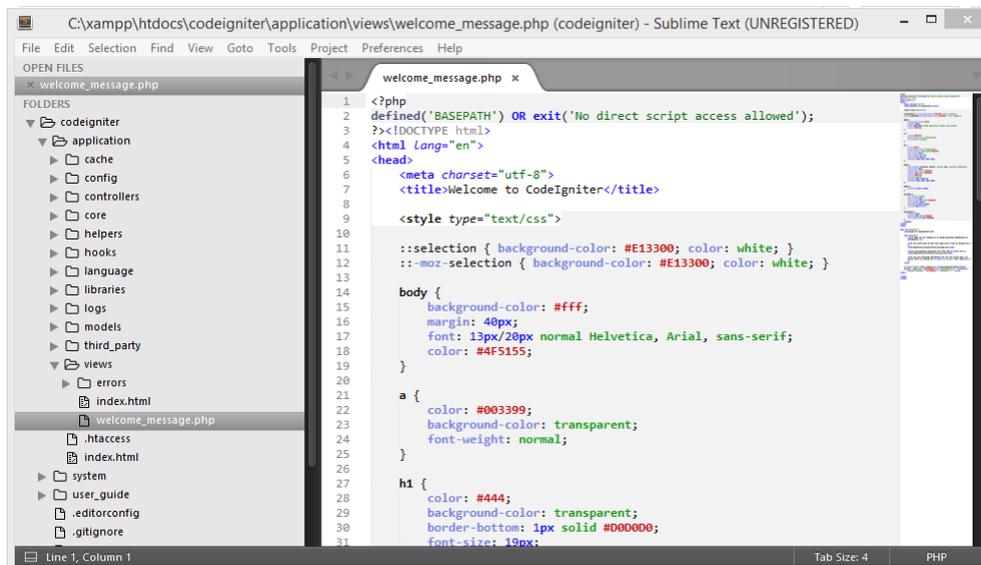
Gambar 4.2: Pengaktifan Modul Apache dan MySQL pada XAMPP

Buka web browser lalu panggil `http://localhost/namafolder`. Jika tampilan yang keluar seperti gambar di bawah ini, maka instalasi codeigniter berhasil.



Gambar 4.3: Halaman Utama Codeigniter

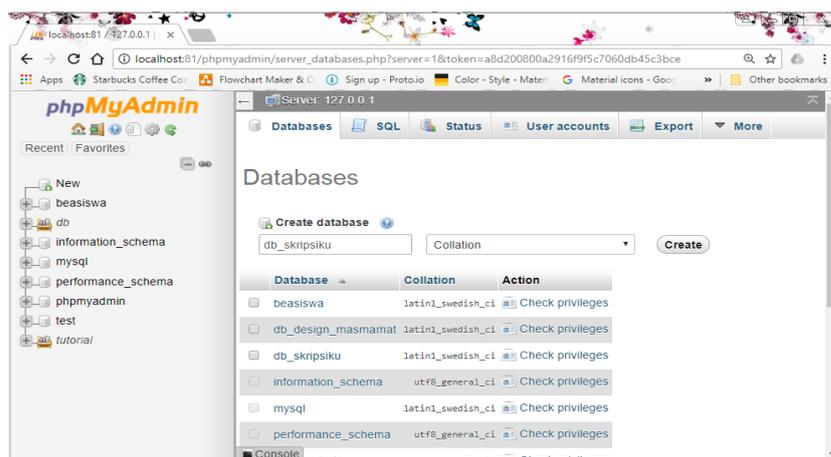
Untuk membuat interface pada framework codeigniter menggunakan fitur *model-view-controller* yang ada dalam folder codeigniter tersebut. Pembuatannya dilakukan dengan menggunakan aplikasi Sublime Text.



Gambar 4.4: Pembuatan *Interface* Sistem menggunakan *Sublime Text*

2. Pembuatan *Database*

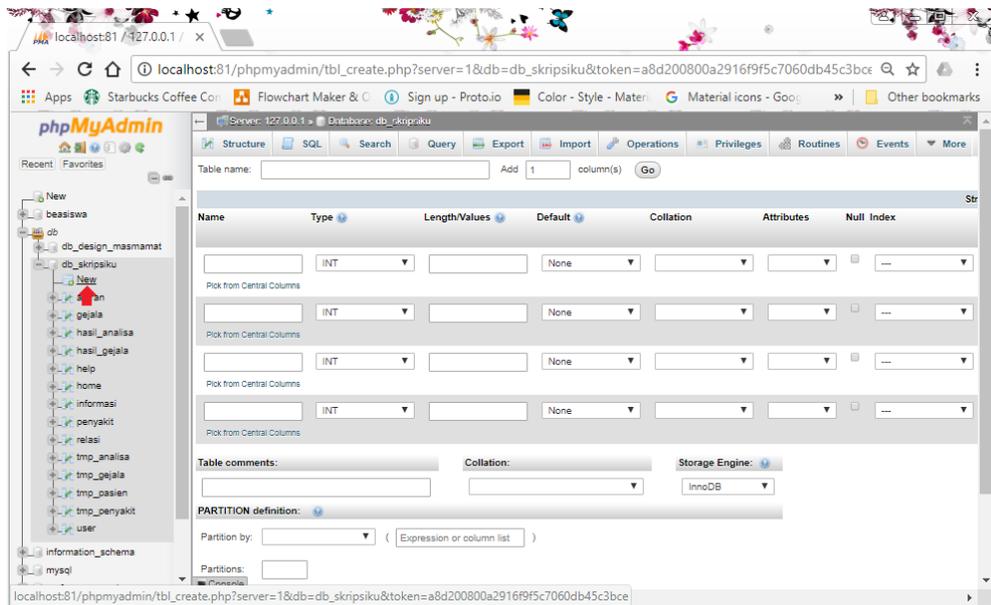
Pembuatan *database* dalam MySQL menggunakan phpMyAdmin dilakukan dengan mengetik `http://localhost/phpmyadmin` pada browser, lalu klik *new* dan *create database*.



Gambar 4.5: Membuat *Database* pada MySQL Menggunakan phpMyAdmin

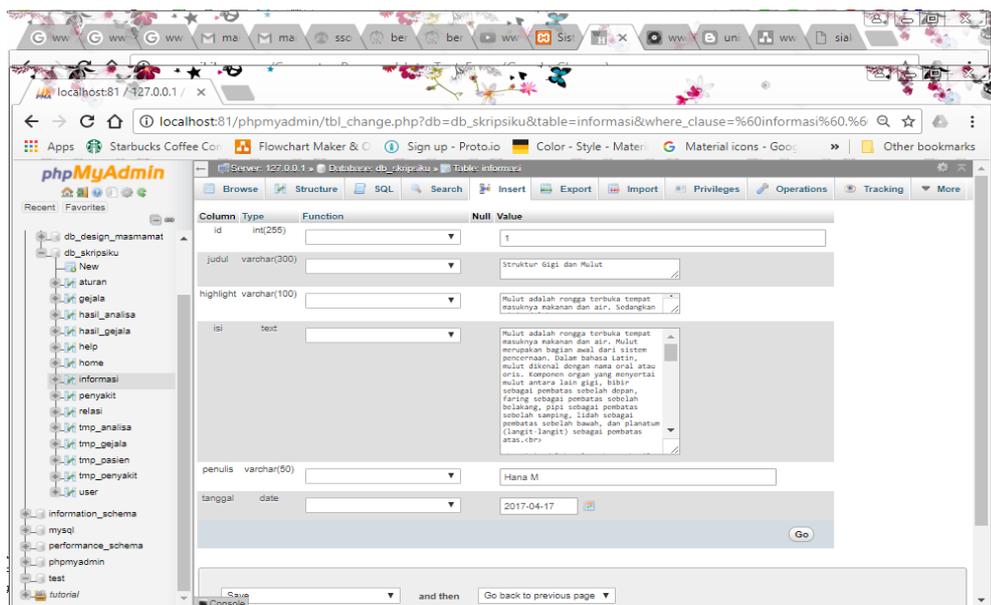
Untuk membuat tabel pada *database* yang sudah dibuat, klik *new* dan isi jumlah kolom yang akan dibuat serta mengisi nama-nama, jenis data (*type*), *length* dan

attribute masing-masing kolom dan *save*.



Gambar 4.6: Membuat Tabel pada *Database*

Setelah menentukan kolom-kolom tersebut, selanjutnya mengisi tabel dengan data yang sesuai dengan kolom yang ada dengan klik *insert* untuk mengisi data di dalam tabel tersebut dan *save*.



Gambar 4.7: Mengisi Data pada Tabel

3. Pembuatan Halaman Home

Halaman ini adalah halaman menu utama yang dilihat *user* saat pertama kali mengakses website. Pada tampilan halaman ini terdapat pilihan menu yang terdiri dari menu home, menu informasi, menu konsultasi, dan menu help. Pembuatan menu dalam sistem ini menggunakan php yang disimpan dalam *libraries* dengan nama `Template.php` dan `static.php` yang berisi menu-menu utama yang disimpan dalam *view*. Tujuan penggunaan template ini mempermudah proses pengkodean sehingga tidak perlu menulis secara berulang-ulang dalam file *view* yang lainnya.

```

56 <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
57 <ul class="nav navbar-nav navbar-right">
58 <li>
59 <a href="<?php echo base_url();?>">Home</a>
60 </li>
61 <li>
62 <a href="<?php echo base_url();?>index.php/informasi">Informasi</a>
63 </li>
64 <li>
65 <a href="<?php echo base_url();?>index.php/konsultasi">Konsultasi</a>
66 </li>
67
68 <?php if ($this->session->userdata("logged_in")){?>
69 <li>
70 <a href="<?php echo base_url();?>index.php/riwayat_konsultasi">Riwayat Konsultasi</a>
71 </li>
72 <li>
73 <a href="<?php echo base_url();?>index.php/konsultasi/logout">Logout</a>
74 </li>
75 <?php } ?>
76
77 <li>
78 <a href="<?php echo base_url();?>index.php/help">Help</a>
79 </li>
80 </ul>
81 </div>

```

Gambar 4.8: `static.php`

Konten dalam halaman Home ini disimpan dalam *database* `db_skripsiku` pada Tabel Home. Pengaturan tampilan konten pada halaman ini terdapat pada `v_home.php` dalam *views*. Proses pengambilan konten dalam database untuk dimunculkan pada web diatur oleh `m_home.php` dalam *models*. Kedua file tersebut di *extends* dalam `home.php` yang disimpan di *controller*. *Source code* tampilan Menu *Home* terdapat pada gambar 4.9.

```

1 <div class="container">
2   <div class="row">
3     <div class="col-lg-8 col-lg-offset-2 col-md-10 col-md-offset-1">
4
5       <h1>
6         <?php foreach ($home->result() as $key): ?>
7           <?php echo $key->judul; ?></h1>
8         <?php endforeach ?>
9
10      <p>
11        <?php foreach ($home->result() as $key): ?>
12          <?php echo $key->isi; ?>
13        <?php endforeach ?>
14      </p>
15    </div>
16  </div>
17 </div>

```

Gambar 4.9: v_home.php

4. Pembuatan Halaman Informasi

Halaman ini menampilkan informasi mengenai struktur gigi dan mulut dan jenis-jenis penyakit gigi dan mulut. Konten pada halaman informasi disimpan dalam *database* db_skripsiku dalam Tabel Informasi. Pengaturan tampilan konten pada halaman ini terdapat pada v_informasi.php dan tampilan detail informasi dalam readmore.php dalam *views*. Proses pengambilan konten dalam *database* untuk dimunculkan pada web diatur oleh m_informasi.php dalam *models*. Kemudian kedua file tersebut di *extends* dalam informasi.php yang disimpan di *controller*. tampilan Menu Informasi terdapat pada gambar 4.10.

```

1 <div class="container">
2   <div class="row">
3     <div class="col-lg-8 col-lg-offset-2 col-md-10 col-md-offset-1">
4
5     <?php foreach ($informasi->result() as $key) { ?>
6
7       <div class="post-preview">
8         <a href="<?php echo base_url('index.php/informasi/readmore/'.$key->id);?>">
9           <h2 class="post-title">
10            <?php echo $key->judul; ?>
11          </h2>
12          <h3 class="post-subtitle">
13
14            <?php
15              $artikel = $key->isi;
16              $potong = substr($artikel, 0, 200);
17              echo $potong;
18            ?>
19
20          </h3>
21        </a>
22        <p class="post-meta">Posted by <a href="#"><?php echo $key->penulis; ?></a> on <?php echo $key->tanggal;
23        ?></p>
24      </div>
25      <hr>
26    <?php } ?>
27
28  </div>
29 </div>
30 </div>

```

Gambar 4.10: v_informasi.php

```

1 <div class="container">
2   <div class="row">
3     <div class="col-lg-8 col-lg-offset-2 col-md-10 col-md-offset-1">
4
5     <?php foreach ($informasi->result() as $key) { ?>
6
7       <div class="post-preview">
8         <!--<a href="post.html">-->
9         <h2>
10          <?php
11            echo $key->judul;
12          ?>
13        </h2>
14        <p>
15          <?php
16            echo $key->isi;
17          ?>
18        </p>
19        </a>
20        <p class="post-meta">Posted by <a href="#"><?php echo $key->penulis; ?></a> on <?php echo $key->tanggal;
21        ?></p>
22      </div>
23      <hr>
24    <?php } ?>
25
26  </div>
27 </div>
28 </div>

```

Gambar 4.11: readmore.php

5. Pembuatan Halaman Login dan Register

Halaman ini menampilkan form login dan register sebelum *user* memulai konsultasi. Form ini terletak pada halaman konsultasi. Tampilan halaman konsultasi yang sesungguhnya akan muncul ketika *user* berhasil login. Untuk *user* yang belum memiliki akun, setelah berhasil melakukan registrasi akan langsung *direct* ke halaman konsultasi tanpa harus login kembali. Pengaturan tampilan form login terdapat pada `v_konsultasi.php` dan tampilan form register dalam `v_signup.php` dalam *views*. Proses penyimpanan data *user* ke dalam *database* saat register diatur oleh `signup.php` dalam *controller*. Sedangkan proses pengecekan data *user* dalam *database* saat login dilakukan oleh `m_loginuser.php` dalam *models* dan `konsultasi.php` dalam *controller*.

```

337 <?php } else { ?>
338     <section class="content-section">
339         <div class="container">
340             <div class="row">
341                 <div class="col-lg-8 col-lg-offset-2 col-md-10 col-md-offset-1">
342
343
344                     <p>Sebelum memulai konsultasi, silahkan <b>Login</b>.</p>
345
346
347                     <center><h2>Login</h2></center>
348
349                     <form method="post" role="form" id="form_login" action="<?php echo site_url('index.php/konsultasi/login')>">
350                         <div class="imgcontainer">
351
352                         <table>
353                             <div class="container">
354                                 <label><b>Email</b></label>
355                                 <input type="text" placeholder="Enter Email Address" name="email" class="form-control" id="email" required>
356                             <br>
357                                 <label><b>Password</b></label>
358                                 <input type="password" placeholder="Enter Password" name="password" class="form-control" id="password" required>
359                             <br>
360                                 <p style="color:red"><?php echo $this->session->flashdata('notification')></p>
361                                 <center><button type="submit"><b>Login</b></button></center>
362
363                                 <!-- <div class="container" style="background-color:#f1f1f1">
364                                     <button type="button" class="cancelbtn">Cancel</button>
365                                 -->
366                             </div></table>
367
368
369                         </div>
370                         <center><a href="<?php echo base_url();>index.php/signup">Belum punya akun? Daftar di sini.</a></center>
371                     </div>
372                 </div>
373             </section>
374 <?php } ?>

```

Gambar 4.12: login.php

```

75 <form method="post" role="form" id="form_login" action="<?php echo site_url('index.php/signup/proses')>">
76 <div class="imgcontainer">
77 <table>
78 <div class="container">
79 <div class="form-group">
80 <label><b>Email:</b></label>
81 <div class="input-group input-group-lg">
82 <input type="text" placeholder="Email Address" name="email" class="form-control" id="email" required>
83 </div>
84 </div>
85
86 <div class="form-group">
87 <label><b>Password:</b></label>
88 <div class="input-group input-group-lg">
89 <input type="password" placeholder="Password" name="password" class="form-control" id="password" required>
90 </div>
91 </div>
92
93 <div class="form-group">
94 <label><b>Nama:</b></label>
95 <div class="input-group input-group-lg">
96 <input type="text" placeholder="Nama" name="nama" class="form-control" id="nama" required>
97 </div>
98 </div>
99
100 <div class="form-group">
101 <label><b>Gender:</b></label>
102 <div class="input-group input-group-lg">
103 <input type="radio" name="gender" id="gender" value="Laki-Laki" style="text-indent: 0.5in" />
    Laki-Laki
    <input type="radio" name="gender" id="gender" value="Perempuan" style="text-indent: 0.5in" /> Perempuan
104 </div>
105 </div>
106
107 <div class="form-group">
108 <label><b>Usia:</b></label>
109 <div class="input-group input-group-lg">
110 <input type="text" placeholder="Usia" name="usia" class="form-control" id="usia" required>
111 </div>
112 </div>
113
114 <div class="form-group">
115 <label><b>Riwayat Penyakit:</b></label>
116 <div class="input-group input-group-lg">
117 <textarea name="riwayat_penyakit" required="required" class="form-control" id="riwayat_penyakit"
118 placeholder="Riwayat Penyakit"></textarea>
119 </div>
120 </div>
121 <center><button type="submit"><b>Sign Up</b></button></center>

```

Gambar 4.13: signup.php

6. Pembuatan Halaman Konsultasi

Halaman ini menampilkan interaksi utama *user* dengan sistem yaitu tanya jawab seputar gejala-gejala penyakit yang mungkin *user* alami. Proses dalam konsultasi ini menggunakan metode *Forward Chaining*. Semua data yang diperlukan dalam sistem ini disimpan dalam *database* *db_skripsiku*. Tampilan dan proses konsultasi di atur oleh *v_konsultasi.php* dalam *views*, *konsultasi.php* dalam *controller*, dan dibantu oleh beberapa fungsi yang disimpan dengan na-

ma fungsi_helper.php dalam *helpers*.

```

149 <form action="<?php echo site_url('index.php/Konsultasi/konsulcek');?>" method="post" name="form1" target="_self">
150 <div class="content-box">
151     <div class="content-box-wrapper">
152         <center><p>Apakah anda mengalami gejala<p>
153             <input name="TxtKdGejala" type="hidden" value="<?php echo $kdgejala; ?>">
154             <b><p> <?= $gejala; ?> ?</p></b>
155
156             <br>
157 <input type="radio" name="RbPilih" value="YA" checked>
158 Benar (YA)
159 <input type="radio" name="RbPilih" value="TIDAK">
160 Salah (TIDAK)
161 <br>
162 <br>
163 <br>
164
165 <button type="submit"><b>Jawab</b></button>
166 </center>
167
168 </form>

```

Gambar 4.14: v_konsultasi.php

Setelah selesai melakukan menjawab seluruh pertanyaan yang diajukan oleh sistem, *user* akan mendapatkan hasil kesimpulan penyakit yang mungkin dideritanya. Dalam kasus sistem ini, *user* diperbolehkan untuk menyimpan hasil konsultasi tersebut dalam riwayat konsultasi. *User* juga diperbolehkan untuk melakukan konsultasi kembali apabila konsultasi sebelumnya kurang meyakinkan.

```

198 <table class="table table-bordered table-striped table-condensed flip-content">
199 <tr align="center">
200 <td colspan="2" bgcolor="#2285DD"><b>HASIL ANALISA</b></td>
201 </tr>
202 <tr>
203 <td colspan="2"><b>DATA PASIEN :</b></td>
204 </tr>
205 <tr bgcolor="#FFFFFF">
206 <td width="86">Nama</td>
207 <td width="689"><?php echo $du->nama; ?></td>
208 </tr>
209 <tr bgcolor="#FFFFFF">
210 <td>Kelamin</td>
211 <td><?php echo $du->gender; ?></td>
212 </tr>
213 <tr bgcolor="#FFFFFF">
214 <td>Usia</td>
215 <td><?php echo $du->usia; ?></td>
216 </tr>
217 <tr bgcolor="#FFFFFF">
218 <td>Tanggal Konsultasi</td>
219 <td><?php echo date("Y-m-d"); ?></td>
220 </tr>
221 <tr bgcolor="#FFFFFF">
222 <td>&nbsp;</td>
223 <td>&nbsp;</td>
224 </tr>
225 <tr>
226 <td colspan="2"><b>HASIL ANALISA TERAKHIR :</b></td>
227 </tr>
228 <tr bgcolor="#FFFFFF">
229 <td>Penyakit</td>
230 <td><?php echo $d->penyakit; ?></td>
231 </tr>
232 <tr>
233 <tr bgcolor="#FFFFFF">
234 <td valign="top">Gejala</td>
235 <td>
236 <?php
237 $sql_gejala = "SELECT * FROM relasi t join gejala g ON t.kd_gejala=g.kd_gejala
238 WHERE kd_penyakit='$kd_penyakit'";
239 $qry_gejala = $this->db->query($sql_gejala)->result();
240 $i=0;
241 foreach ($qry_gejala as $data_gejala)
242 {
243 $i++;
244 echo "$i . $data_gejala->nama_gejala <br>";
245 }

```

Gambar 4.15: v_hasil_konsultasi.php

```

247 >?
248 </td>
249 </tr>
250 <tr bgcolor="#FFFFFF">
251 <td valign="top">Definisi</td>
252 <td><?php echo $d->deskripsi; ?></td>
253 </tr>
254 <tr bgcolor="#FFFFFF">
255 <td valign="top">Penyebab</td>
256 <td><?php echo $d->penyebab; ?></td>
257 </tr>
258 <tr bgcolor="#FFFFFF">
259 <td valign="top">Penanganan</td>
260 <td><?php echo $d->penanganan; ?></td>
261 </tr>
262 <tr bgcolor="#FFFFFF">
263 <td valign="top">Nilai</td>
264 <td><?php echo $sub_nilai; ?></td>
265 </tr>
266 </table>
267 </table>
268 <table class="table table-bordered table-striped table-condensed flip-content">
269 <tr>
270 <td><a href="<?php echo site_url('index.php/konsultasi/simpan_hasil'); ?>" class="btn btn-success"><i class="fa fa-save"></i> Simpan
271 Hasil Konsultasi</a> <a href="<?php echo site_url('index.php/konsultasi/reset_hasil'); ?>" class="btn btn-danger"><i class="fa fa-repeat">
272 </i> Coba Konsultasi Lagi!</a></td></tr>

```

Gambar 4.16: v_hasil_konsultasi.php

7. Pembuatan Halaman Riwayat Konsultasi

Halaman ini menampilkan daftar riwayat konsultasi yang pernah dilakukan *user*. Hasil konsultasi yang terdaftar dalam halaman ini adalah hasil konsultasi yang disimpan oleh *user*. Hasil konsultasi yang tidak disimpan tidak akan muncul dalam halaman ini. Tampilan dan proses riwayat konsultasi di atur oleh `v_riwayat_konsultasi.php` dalam *views*, `konsultasi.php` dalam *controller*.

```

70 <!-- Konsultasi Section -->
71 <section class="container text-center">
72   <div class="container">
73     <center><h2 class="font-black">
74       Riwayat Konsultasi
75     </h2></center>
76     <hr>
77
78
79     <?php
80     $NOIP = $this->session->userdata("uid");
81     ?>
82
83     <table class="table table-bordered table-striped table-condensed flip-content" id="mytable">
84       <thead class="flip-content">
85         <tr>
86           <th width="8px">No</th>
87           <th>tanggal</th>
88           <th>Penyakit</th>
89           <th>Action</th>
90         </tr>
91       </thead>
92       <tbody>
93         <?php
94         $start = 0;
95         $query="SELECT * FROM hasil_analisa h join penyakit p ON h.kd_penyakit=p.kd_penyakit where id_user='$NOIP' order by `tanggal` desc";
96         $result= $this->db->query($query)->result();
97         foreach ($result as $d)
98         {
99           ?>
100           <tr>
101             <td><?php echo ++$start ?></td>
102             <td><?php echo $d->tanggal; ?></td>
103             <td><?php echo $d->penyakit; ?></td>
104
105             <td style="text-align:center" width="200px">
106               <a href="<?php echo site_url('index.php/Riwayat_konsultasi/detail/'. $d->id_hasil);?>" class="btn btn-primary btn-xs purple"><i
107                 class="fa fa-search"></i> Detail</a>
108             </td>
109           </tr>
110         }
111       </tbody>
112     </table>
113

```

Gambar 4.17: `v_riwayat_konsultasi.php`

Rincian dari riwayat konsultasi dapat dilihat dengan mengklik Detail. Tampilan dan proses detail riwayat konsultasi di atur oleh `v_detail_riwayat_konsultasi.php` dalam *views*, `konsultasi.php` dalam *controller*.

```

70     <!-- Konsultasi Section -->
71     <section class="container text-center">
72         <div class="container">
73             <center><h2 class="font-black">
74                 Detail Konsultasi
75             </h2></center>
76             <hr>
77
78
79             <?php
80
81             $NOIP = $this->session->userdata("uid");
82             $su = "SELECT * FROM user
83                 WHERE id_user='$_SESSION[uid]' ";
84             $du = $this->db->query($su)->row();
85
86             $s = "SELECT * FROM penyakit
87                 WHERE kd_penyakit='$kd_penyakit' ";
88             $d = $this->db->query($s)->row();
89
90
91             >
92
93             <table class="table table-bordered table-striped table-condensed flip-content">
94                 <tr align="center">
95                     <td colspan="2" bgcolor="#00AC11"><b>HASIL ANALISA</b></td>
96                 </tr>
97                 <tr>
98                     <td colspan="2"><b>DATA PASIEN :</b></td>
99                 </tr>
100                <tr bgcolor="#FFFFFF">
101                    <td width="86">Nama</td>
102                    <td width="689"><?php echo $du->nama; ?></td>
103                </tr>
104                <tr bgcolor="#FFFFFF">
105                    <td>Kelamin</td>
106                    <td><?php echo $du->xgender; ?></td>
107                </tr>
108                <tr bgcolor="#FFFFFF">
109                    <td>Usia</td>
110                    <td><?php echo $du->xusia; ?></td>
111                </tr>
112                <tr bgcolor="#FFFFFF">
113                    <td>Tanggal Konsultasi</td>
114                    <td><?php echo $tanggal; ?></td>
115                </tr>

```

Gambar 4.18: v_detail_riwayat_konsultasi.php

8. Pembuatan Halaman Help

Halaman ini menampilkan petunjuk atau langkah-langkah penggunaan website sistem pakar diagnosa penyakit gigi dan mulut. Tampilan dan proses help di atur oleh v_help.php dalam *views*, m_help dalam *models*, dan help.php dalam *controller*.

```

1 <div class="container">
2   <div class="row">
3     <div class="col-lg-8 col-lg-offset-2 col-md-10 col-md-offset-1">
4
5     <?php foreach ($help->result() as $key) { ?>
6       <h2>
7         <?php echo $key->judul; ?>
8       </h2>
9       <p>
10        <?php echo $key->isi; ?>
11      </p>
12    <?php } ?>
13
14    </div>
15  </div>
16 </div>

```

Gambar 4.19: help.php

2. Hasil Implementasi

Berisi tentang langkah-langkah untuk menggunakan Web Sistem Pakar Diagnosa Penyakit Gigi dan Mulut melalui browser.

1. Start Modul Apache dan MySQL pada XAMPP Control Panel. Buka browser dan ketik localhost/hana/sistempakar. Setelah itu akan muncul tampilan utama pada web yang terdiri dari menu Home, menu Informasi, menu Konsultasi, menu Riwayat Konsultasi dan menu Help. Tampilan utama web terdapat pada gambar 4.20.



Selamat Datang!

Welcome to My Expert System Site

Website sistem pakar ini bertujuan untuk memberikan informasi kesehatan seputar gigi dan mulut kepada masyarakat serta memberikan kemudahan bagi masyarakat dalam mendiagnosa (prediksi awal) penyakit gigi dan mulut yang dideritanya.

Untuk melakukan proses pendiagnosaan penyakit pasien cukup menjawab pertanyaan-pertanyaan dari sistem berupa gejala yang mungkin dirasakannya. Setelah itu sistem akan menampilkan hasil diagnosa penyakit user.

Gambar 4.20: Tampilan Menu *Home*

2. Kemudian halaman Informasi yang berisi informasi mengenai struktur gigi dan mulut dan macam-macam penyakit gigi dan mulut.



Gambar 4.21: Tampilan Menu Informasi



Gambar 4.22: Tampilan Detail Informasi

3. Kemudian halaman Konsultasi akan memunculkan form login dan form register sebelum *user* dapat memulai konsultasi. Tampilan halaman konsultasi yang sesungguhnya akan muncul ketika *user* berhasil login. Untuk *user* yang belum

memiliki akun, setelah berhasil melakukan registrasi akan langsung di *direct* ke halaman konsultasi tanpa harus login kembali.



Sebelum memulai konsultasi, silahkan Login.

Login

Email

Password

Belum punya akun? Daftar di sini.

Gambar 4.23: Tampilan Form Login



Sign Up

Email:

Password:

Nama:

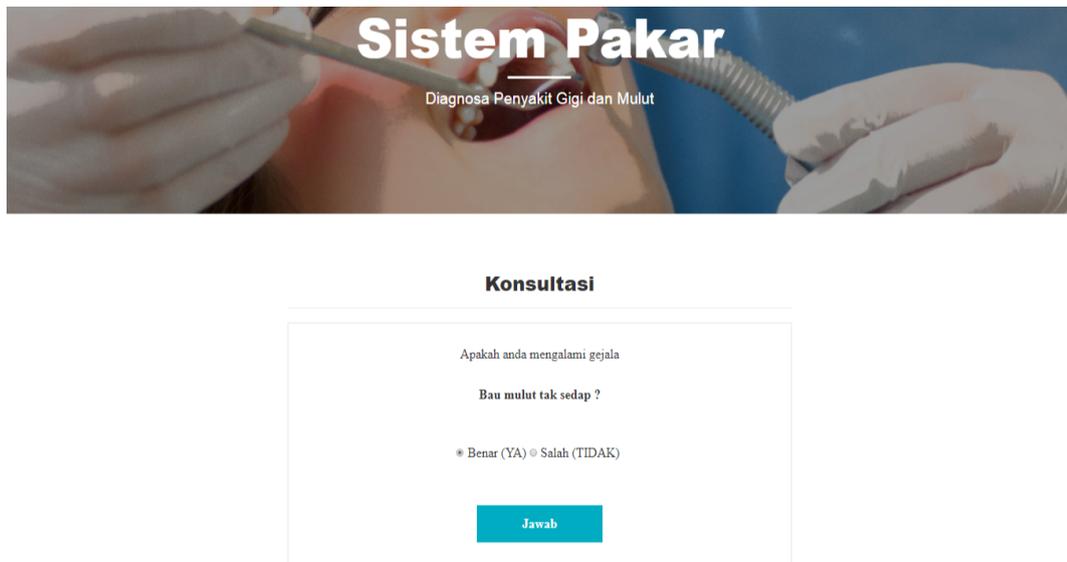
Gender:
 Laki-Laki Perempuan

Usia:

Riwayat Penyakit:

Gambar 4.24: Tampilan Form Register

Setelah *user* berhasil login, halaman ini akan menampilkan halaman konsultasi dimana interaksi utama *user* dengan sistem terjadi yaitu tanya jawab seputar gejala-gejala penyakit yang mungkin *user* alami.



Gambar 4.25: Tampilan Halaman Konsultasi

Setelah selesai melakukan menjawab seluruh pertanyaan yang diajukan oleh sistem, *user* akan mendapatkan hasil kesimpulan penyakit yang mungkin dideritanya. Dalam kasus sistem ini, *user* diperbolehkan untuk menyimpan hasil konsultasi tersebut dalam riwayat konsultasi. *User* juga diperbolehkan untuk melakukan konsultasi kembali apabila konsultasi sebelumnya kurang meyakinkan.



Konsultasi

HASIL ANALISA	
DATA USER :	
Nama	Hana Maulinda
Kelamin	Perempuan
Usia	21
Tanggal Konsultasi	2017-08-20
HASIL ANALISA PENYAKIT :	
Penyakit	Periodontitis
Gejala	1 . Bau mulut tak sedap 2 . Gigi goyang 3 . Gusi yang turun membuat gigi terlihat lebih panjang 4 . Gusi tampak merah dan bengkak 5 . Gusi mudah berdarah
Definisi	Periodontitis adalah peradangan gusi yang menyebar sampai ke struktur penyangga gigi.
Penyebab	Periodontitis merupakan salah satu penyebab utama lepasnya gigi pada orang dewasa dan lanjut usia.
Penanganan	Perawatan periodontal non bedah meliputi pemeliharaan kebersihan mulut, Scaling dan Root Planing (SRP) dan pemberian antibiotik untuk mencegah, menghentikan serta mengeliminasi penyakit periodontal yang merupakan intial phase therapy. Kuretase yaitu mengeruk dinding gingiva dari kantung gusi untuk membuang jaringan lunak yang tidak sehat. Mengeluarkan jaringan lunak yang meradang dan merupakan dinding lateral dari poket.
Nilai Peluang	0.049825106183106
<div style="display: flex; justify-content: space-between;"> SIMPAN HASIL KONSULTASI COBA KONSULTASI LAGI! </div>	

Gambar 4.26: Tampilan Halaman Hasil Konsultasi

Sementara itu, apabila dalam proses konsultasi *user* menjawab pertanyaan seputar gejala-gejala yang tidak ada aturannya dalam mesin inferensi, maka sistem akan *handle* kasus tersebut dengan memberikan pernyataan “Penyakit tidak ditemukan” namun *user* dapat melakukan konsultasi kembali.



Gambar 4.27: Tampilan Halaman Penyakit Tidak Ditemukan

4. Kemudian halaman Riwayat Konsultasi yang berisi daftar riwayat konsultasi yang pernah dilakukan *user*. Hasil konsultasi yang terdaftar dalam halaman ini adalah hasil konsultasi yang disimpan oleh *user*. Hasil konsultasi yang tidak disimpan tidak akan muncul dalam halaman ini.



Gambar 4.28: Tampilan Halaman Riwayat Konsultasi

Rincian dari riwayat konsultasi dapat dilihat dengan mengklik Detail. Tampilan

detail riwayat konsultasi terdapat pada Gambar 4.29.



Detail Konsultasi

HASIL ANALISA	
DATA USER :	
Nama	Hana Maulinda
Kelamin	Perempuan
Usia	22
Tanggal Konsultasi	2017-08-16 11:18:17
HASIL ANALISA PENYAKIT :	
Penyakit	Herpes
Gejala	<ol style="list-style-type: none"> 1 . Pembengkakan kelenjar getah bening 2 . Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum 3 . Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah 4 . Terjadi demam 5 . Gusi mudah berdarah 6 . Gusi tampak merah dan bengkak 7 . Terdapat gelembung (vesikel) di dalam rongga mulut
Definisi	Infeksi Herpes Mulut (Gingivostomatitis Herpetik Primer, Herpes Labialis) adalah suatu infeksi awal oleh virus herpes simpleks yang menyebabkan terbentuknya luka yang terasa nyeri di gusi dan bagian mulut lainnya. Sedangkan Herpes sekunder (Herpes Labialis sekunder) adalah suatu reaktivasi (pengaktifan kembali) virus herpes simpleks.
Penyebab	Bila seseorang terinfeksi virus herpes, maka virus ini akan terus berada di saraf dalam keadaan laten yang artinya sewaktu-waktu penyakit ini dapat kambuh lagi.
Penanganan	Perawatan dengan pemberian salep anestetik untuk meredakan rasa nyeri di kulit seperti salep yang mengandung lidocaine. Obat herpes mulut yang diberikan secara oral atau suntikan intra vena dapat diberikan pada orang dengan sistem kekebalan tubuh rendah atau pada bayi umur < 6 bulan, atau pada orang yang menderita penyakit parah.
Nilai	0.008199
C KEMBALI	

Gambar 4.29: Tampilan Halaman Detail Riwayat Konsultasi

5. Kemudian halaman Help yang berisi tentang petunjuk atau langkah-langkah penggunaan website sistem pakar diagnosa penyakit gigi dan mulut.



PETUNJUK A : MENU-MENU PADA WEBSITE

Pada website sistem pakar untuk mendiagnosa penyakit gigi dan mulut ini terdapat 4 menu utama, yaitu:

Menu Home : Menu ini berfungsi untuk melihat halaman utama website.

Menu Informasi : Menu ini berfungsi untuk melihat informasi seputar gigi dan mulut, seperti 15 jenis penyakit gigi dan mulut.

Menu Konsultasi : Menu ini merupakan menu yang paling penting dalam website sistem pakar penyakit gigi dan mulut. Menu ini berfungsi untuk melakukan proses pendiagnosaan penyakit. Petunjuk proses pendiagnosaan penyakit dapat anda lihat di petunjuk bagian B. Menu ini juga berfungsi untuk melihat riwayat konsultasi pada masing-masing user.

Menu Help : Menu ini berfungsi untuk melihat informasi menu-menu pada website ini.

PETUNJUK B : LANGKAH-LANGKAH PROSES DIAGNOSA

Gambar 4.30: Tampilan Halaman Help

Apabila *user* mengklik Logout maka *user* akan keluar dari sistem pakar dan kembali ke menu utama Web Sistem Pakar Diagnosa Penyakit Gigi dan Mulut.

B. Pengujian

Pada tahap ini dilakukan pengujian terhadap sistem yang telah selesai dibuat. Uji coba yang dilakukan pada sistem ini memberikan gambaran dari setiap tampilan yang ada pada sistem. Tahap ini dilakukan untuk mengetahui fungsi-fungsi yang diharapkan berjalan dengan benar dan untuk mengetahui kekurangan pada sistem ini.

1. Pengujian Sistem

Metode pengujian sistem ini dilakukan menggunakan metode *Black Box Testing*. Pengujian ini dilakukan untuk memperlihatkan bahwa fungsi-fungsi dalam sistem bekerja dengan baik dalam arti masukkan yang diterima sistem dan keluaran yang dihasilkan sistem benar-benar sesuai atau tepat.

Tabel 4.1: Hasil Pengujian Sistem Lanjutan

No.	Fungsi	Input	Output	Hasil
1.	Menu Home	Memilih Menu Home	Sistem menampilkan konten halaman home	Sesuai
2.	Menu Informasi	Memilih Menu Informasi	Sistem menampilkan list/daftar informasi	Sesuai
3.	Detail Informasi	Memilih salah satu list informasi	Sistem menampilkan detail informasi	Sesuai
4.	Login User	Memasukkan username dan password user	Sistem menampilkan halaman utama konsultasi	Sesuai
5.	Registrasi User	Memasukkan data registrasi user	Sistem menampilkan halaman utama konsultasi	Sesuai
6.	Menu Konsultasi	Memilih Menu Konsultasi	Sistem menampilkan pertanyaan konsultasi	Sesuai
7.	Proses Diagnosa	Menjawab pertanyaan "ya" atau "tidak"	Sistem menampilkan hasil diagnosa penyakit gigi dan mulut	Sesuai
8.	Simpan Hasil Konsultasi	Memilih simpan hasil konsultasi	Sistem menyimpan hasil konsultasi ke dalam menu riwayat konsultasi	Sesuai
9.	Menu Riwayat Konsultasi	Memilih Menu Riwayat Konsultasi	Sistem menampilkan list/daftar riwayat konsultasi	Sesuai
10.	Detail Riwayat Konsultasi	Memilih detail pada salah satu list riwayat konsultasi	Sistem menampilkan detail hasil riwayat konsultasi	Sesuai
11.	Menu Help	Memilih Menu Help	Sistem menampilkan halaman petunjuk pemakaian sistem	Sesuai
12.	Logout	Memilih Logout	Sistem mampu me-logout user	Sesuai

Dari hasil Pengujian *Black Box* di atas, dapat disimpulkan bahwa sistem pakar gigi dan mulut ini sudah mampu berjalan dengan baik sesuai dengan masing-masing fungsi yang ada.

2. Pengujian Akurasi

Pengujian akurasi adalah pengujian yang dilakukan dengan membandingkan hasil kesimpulan penyakit antara hasil pakar dengan hasil sistem pakar. Hasil dari pengujian pakar adalah sebagai berikut:

Tabel 4.2: Hasil Pengujian Akurasi

No.	Gejala yang diuji atau dipilih berdasarkan jawaban "ya"	Hasil Diagnosa Sistem	Hasil Diagnosa Pakar	Kesesuaian Hasil
1.	1. Bau mulut tak sedap 2. Bentuk gusi agak membulat/tumpul 3. Gusi meradang 4. Gusi mudah berdarah 5. Gusi tampak merah dan bengkak 6. Gusi terasa sakit/nyeri,	Gingivitis	Gingivitis	1
2.	1. Bau mulut tak sedap 2. Gigi goyang 3. Gusi mudah berdarah 3. Gusi tampak merah dan bengkak 4. Gusi yang turun membuat gigi terlihat lebih panjang,	Periodontitis	Periodontitis	1
3.	1. Bau mulut tak sedap	Halitosis	Halitosis	1

No.	Gejala yang diuji atau dipilih berdasarkan jawaban "ya"	Hasil Diagnosa Sistem	Hasil Diagnosa Pakar	Kesesuaian Hasil
4.	1. Bentuk gigi tampak terkikis 2. Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	Abrasi Gigi	Abrasi Gigi	1
5.	1. Bibir kering 2. Bibir mudah berdarah 3. Bibir pecah-pecah 4. Bibir terasa perih	Cheilitis	Cheilitis	1
6.	1. Gigi,berjejal 2. Gigi renggang 3. Gigi tonggos 4. Gingsul	Malocclusion	Malocclusion	1
7.	1. Gigi sulung yang tak kunjung tanggal 2. Timbulnya benih gigi dengan posisi yang abnormal	Persistensi Gigi Sulung	Persistensi Gigi Sulung	1
8.	1. Gusi tampak merah dan bengkak 2. Gigi terasa sakit atau berdenyut 3. Nyeri saat mengunyah 4. Pembengkakan kelenjar getah bening 5. Terjadi demam,	Abses Periapikal	Abses Periapikal	1

Tabel 4.3: Hasil Pengujian Akurasi Lanjutan

Tabel 4.4: Hasil Pengujian Akurasi Lanjutan 2

No.	Gejala yang diuji atau dipilih berdasarkan jawaban "ya"	Hasil Diagnosa Sistem	Hasil Diagnosa Pakar	Kesesuaian Hasil
9.	1. Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin 2. Gigi terasa sakit atau berdenyut 3. Nyeri saat menggigit 4. Terdapat lubang pada gigi 5. Timbulnya bercak coklat, hitam atau putih pada gigi	Caries	Caries	1
10.	1. Ada lubang pada gigi 2. Gigi terasa sakit atau berdenyut 3. Gigi yang berlubang terasa sakit bila masuk makanan 4. Nyeri sampai ke daerah sinus, pelipis, mata atau telinga	Pulpitis Irreversible	Pulpitis Irreversible	1
11.	1. Ada lubang pada gigi 2. Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin 3. Rasa sakit tajam hanya sebentar	Pulpitis Reversible	Pulpitis Reversible	1
12.	1. Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	Hipersensitif Dentin	Hipersensitif Dentin	1
13.	1. Gusi atau gigi bernanah 2. Gusi,tampak merah dan bengkak 3. Nyeri saat membuka mulut 4. Nyeri saat mengunyah,	Abses Periodontal	Abses Periodontal	1

Tabel 4.5: Hasil Pengujian Akurasi Lanjutan 3

No.	Gejala yang diuji atau dipilih berdasarkan jawaban "ya"	Hasil Diagnosa Sistem	Hasil Diagnosa Pakar	Kesesuaian Hasil
14.	1. Gusi mudah berdarah 2. Gusi tampak merah dan bengkak 3. Pembengkakan kelenjar getah bening 4. Terasa sakit, panas, perih, atau gatal terutama saat makan dan, minum 5. Terdapat gelembung (vesikel) di dalam rongga mulut 6. Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah 7. Terjadi demam	Herpes	Herpes	1
15.	1. Nyeri saat luka tersentuh 2. Terdapat luka berbentuk oval atau bulat yang berwarna putih/kuning dan tepi yang merah di dalam rongga mulut	Stomatitis	Stomatitis	1

Hasil akurasi yang bernilai 1 mempunyai arti bahwa diagnosa sistem sesuai dengan diagnosa yang diberikan pakar. Sebaliknya, hasil akurasi yang bernilai 0 berarti diagnosa sistem tidak sesuai dengan diagnosa pakar. Berdasarkan Tabel Pengujian Akurasi di atas dengan 15 sampel data penyakit gigi dan mulut menghasilkan nilai akurasi sebesar:

$$\text{Nilai akurasi} = \frac{15}{15} \times 100\% = 100\%$$

Sehingga dapat disimpulkan bahwa akurasi sistem pakar diagnosa penyakit gi-

gi dan mulut menggunakan metode forward chaining dan metode naive bayes mempunyai tingkat keberhasilan yang baik sesuai dengan diagnosa pakar yaitu sebesar 100%.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil analisa, perancangan, implementasi dan pengujian yang dilakukan pada sistem pakar diagnosa penyakit gigi dan mulut berbasis web, maka dapat diambil kesimpulan sebagai berikut:

1. Sistem pakar diagnosa penyakit gigi dan mulut berhasil diimplementasikan menggunakan metode *Forward Chaining* dalam proses inferensi dan metode *Naive Bayes* dalam perhitungan probabilitas penyakit berdasarkan aturan-aturan yang diperoleh dari pakar.
2. Antarmuka (UI) dalam sistem meliputi menu home, menu informasi, menu konsultasi, menu riwayat konsultasi dan menu help.
3. Hasil pengujian sistem menggunakan *Black Box Testing* menunjukkan bahwa sistem memiliki fungsionalitas yang sesuai untuk keseluruhan fungsi yang ada pada program.
4. Hasil pengujian akurasi diagnosa sistem dengan diagnosa pakar menunjukkan bahwa nilai akurasi hasil keluaran sistem adalah sebesar 100%.

B. Saran

Sistem pakar diagnosa penyakit gigi dan mulut menggunakan metode *Forward Chaining* dan *Naive Bayes* berbasis web ini masih memiliki beberapa kekurangan. Saran yang dapat diberikan untuk pengembangan penelitian selanjutnya antara lain:

1. Untuk pengembangan penelitian lebih lanjut, sistem pakar ini dapat dikembangkan dengan menggunakan metode yang berbeda atau dengan mengkombinasikan metode *Forward Chaining* atau metode *Naive Bayes* dengan metode lain.
2. Sistem dapat dikembangkan untuk kasus pada domain lain, dengan kasus penyakit tertentu dan gejala yang menyertainya serta aturan-aturan yang lebih bervariasi.
3. Interaksi user dengan sistem memiliki pilihan jawaban yang lebih bervariasi.
4. Sistem dikembangkan kembali menjadi sistem yang bersifat dinamis.
5. Pengujian akurasi hasil diagnosa sistem dengan hasil diagnosa pakar menggunakan pakar yang lebih bervariasi.

DAFTAR PUSTAKA

- [1] A. Desiani, *Konsep Kecerdasan Buatan*. ANDI Yogyakarta, 2006.
- [2] A. Ladhjamudin, *Analisis dan Desain Sistem Informasi*. Graha Ilmu, 2005.
- [3] A. Ratih. Jumlah dokter gigi di indonesia jauh dari ideal. [Online]. Available: <http://news.unpad.ac.id/?p=15729>
- [4] D. Trihapningsari, “Aplikasi sistem pakar diagnosa gangguan sistem pencernaan pada lambung menggunakan metode forward chaining berbasis android,” 2016.
- [5] Fathansyah, *Basis Data*. CV. Informatika, 1999.
- [6] I. Sommerville, *Software Engineering 9th Edition*. Addison-Wesley, 2011.
- [7] J. A. d. T. L. E. Turban, *Decision Support System and Intelligent System 7th Edition*. Prentice Hall International, 2005.
- [8] J. Pos. Perawatan gigi mahal hanya mitos. [Online]. Available: www.pressreader.com/indonesia/jawa-pos/20161125/282754881302865
- [9] J. W. dan L.D. Bentley, *Systems Analysis and Design Methods*. McGraw-Hill, 2007.
- [10] K. K. RI, “Laporan hasil riset kesehatan dasar indonesia,” 2013.
- [11] K. Mustaqim, “Aplikasi sistem pakar untuk diagnosa hama dan penyakit tanaman kelapa sawit menggunakan naive bayes,” 2013.
- [12] Martawiansyah, *Gigiku Kuat Mulutku Sehat*. Salamadani Publishing, 2008.
- [13] M. Arhami, *Konsep Dasar Sistem Pakar*. ANDI Yogyakarta, 2005.

- [14] M. Shadiq, "Keoptimalan naive bayes dalam klasifikasi," 2009.
- [15] M. W. Firebaugh, *Artificial Intelligence: A Knowledge-Based Approach*. PWS-Kent Publishing Company, 1989.
- [16] P. H. dan J.K. Kawistara, *Pemrograman Web*. Informatika Bandung, 2014.
- [17] R. McLeod, *Management Information System 6th Edition*. New Jersey. Prentice Hall Inc, 1995.
- [18] R. Pressman, *Software Engineering: A Practitioners Approach 7th Edition*. McGraw-Hill, 2010.
- [19] S. H. dan S. Iswati, *Sistem Pakar dan Pengembangannya*. Graha Ilmu, 2008.
- [20] T. Wikipedia. Codeigniter. [Online]. Available: <https://en.wikipedia.org/wiki/CodeIgniter>
- [21] Y. M. dan E. Pratiwi, *45 Masalah dan Solusi Penyakit Gigi dan Mulut*. Rapha Publishing, 2013.
- [22] W. Komputer, *Panduan Belajar MySQL Database Server*. MediaKita, 2010.

LAMPIRAN A

Nilai Probabilitas Penyakit

Tabel 1.1: Tabel Nilai Probabilitas

No.	Penyakit	Nilai Probabilitas Menurut Ahli 1	Nilai Probabilitas Menurut Ahli 2	Nilai Probabilitas Menurut Ahli 3	Rata-Rata (Mean)
1.	Abrasi Gigi	0,7	0,4	0,6	0,57
2.	Abses Periapikal	0,4	0,8	0,5	0,57
3.	Abses Periodontal	0,2	0,3	0,4	0,30
4.	Caries (Gigi Berlubang)	0,7	1	0,8	0,83
5.	Cheilitis	0,2	0,1	0,2	0,17
6.	Gingivitis	0,8	0,7	0,6	0,70
7.	Halitosis (Bau Mulut)	0,5	0,5	0,5	0,50
8.	Herpes	0,2	0,2	0,1	0,17
9.	Hipersensitif Dentin (Gigi Sensitif)	0,5	0,7	0,6	0,60
10.	Malocclusion	0,7	0,5	0,6	0,60
11.	Periodontitis	0,4	0,3	0,2	0,30
12.	Persistensi Gigi Sulung	0,7	0,7	0,7	0,70
13.	Pulpitis Reversible	0,8	0,4	0,3	0,50
14.	Pulpitis Irreversible	0,6	0,8	0,7	0,70
15.	Stomatitis (Sariawan)	0,5	0,3	0,4	0,40

LAMPIRAN B

Nilai Probabilitas Gejala Terhadap Penyakit

Tabel 2.1: Nilai Probabilitas Gejala Terhadap Penyakit

No.	Penyakit	Gejala Penyakit	Nilai Probabilitas Menurut Ahli 1	Nilai Probabilitas Menurut Ahli 2	Nilai Probabilitas Menurut Ahli 3	Rata-Rata (Mean)
1.	Abrasi Gigi	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0,7	0,7	0,8	0,73
		Bentuk gigi tampak terkikis	1	0,8	0,9	0,90
2.	Abses Periapikal	Gusi tampak merah dan bengkak	1	0,8	0,7	0,83
		Gigi terasa sakit atau berdenyut	0,7	0,8	0,7	0,73
		Nyeri saat mengunyah	0,6	0,6	0,6	0,60
		Terjadi demam	0,7	0,5	0,6	0,60
		Pembengkakan kelenjar getah bening (di sekitar leher)	0,6	0,5	0,5	0,53
3.	Abses Periodontal	Gusi tampak merah dan bengkak	1	0,5	0,7	0,73
		Gusi atau gigi bermanah	0,7	0,3	0,5	0,50
		Nyeri saat mengunyah	1	0,5	0,7	0,73
		Nyeri saat membuka mulut	0,5	0,3	0,5	0,43
4.	Caries (Gigi Berlubang)	Terdapat lubang pada gigi	1	1	1	1,00
		Timbulnya bercak coklat, hitam atau putih pada gigi	1	0,9	1	0,97
		Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0,8	0,9	1	0,90
		Gigi terasa sakit atau berdenyut	0,7	0,8	0,9	0,80
		Nyeri saat menggigit	0,7	0,7	0,7	0,70
5.	Cheilitis	Bibir kering	0,7	0,7	0,7	0,70
		Bibir pecah-pecah	0,6	0,7	0,7	0,67
		Bibir terasa perih	0,7	0,6	0,7	0,67
		Bibir mudah berdarah	0,6	0,7	0,5	0,60

Tabel 2.2: Nilai Probabilitas Gejala Terhadap Penyakit Lanjutan

No.	Penyakit	Gejala Penyakit	Nilai Probabilitas Menurut Ahli 1	Nilai Probabilitas Menurut Ahli 2	Nilai Probabilitas Menurut Ahli 3	Rata-Rata (Mean)
6.	Gingivitis	Gusi meradang	0,6	0,9	0,8	0,77
		Gusi tampak merah dan bengkak	0,3	0,9	0,7	0,63
		Gusi mudah berdarah	0,7	0,9	0,8	0,80
		Bau mulut tak sedap	0,6	0,8	0,7	0,70
		Gusi terasa sakit/nyeri	0,3	0,7	0,5	0,50
		Bentuk gusi agak membulat/ tumpul	0,8	0,7	0,9	0,80
7.	Halitosis (Bau Mulut)	Bau mulut tak sedap	1	1	1	1,00
8.	Herpes	Terjadi demam	0,7	0,7	0,7	0,70
		Gusi tampak merah dan bengkak	0,4	0,7	0,5	0,53
		Gusi mudah berdarah	0,4	0,7	0,9	0,67
		Terasa sakit, panas, perih, atau gatal terutama saat makan dan minum	0,7	0,9	0,7	0,77
		Terdapat gelembung (vesikel) di dalam rongga mulut	0,2	0,9	0,7	0,60
		Terdapat luka (ulkus) yang berwarna kekuningan pada gelembung yang pecah	0,7	0,9	0,5	0,70
		Pembengkakan kelenjar getah bening (di sekitar leher)	0,7	0,6	0,5	0,60
9.	Hipersensitif Dentin (Gigi Sensitif)	Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0,8	1	1	0,93
10.	Malocclusion	Gigi berjejal	0,8	0,9	0,9	0,87
		Gingsul	0,5	0,7	0,5	0,57
		Gigi tonggos	0,6	0,8	0,7	0,70
		Gigi renggang	0,4	0,8	0,6	0,60

Tabel 2.3: Nilai Probabilitas Gejala Terhadap Penyakit Lanjutan 2

No.	Penyakit	Gejala Penyakit	Nilai Probabilitas Menurut Ahli 1	Nilai Probabilitas Menurut Ahli 2	Nilai Probabilitas Menurut Ahli 3	Rata-Rata (Mean)
11.	Periodontitis	Gusi tampak merah dan bengkak	0,4	0,9	0,8	0,70
		Gusi mudah berdarah	0,3	0,9	0,7	0,63
		Bau mulut tak sedap	0,7	0,8	0,7	0,73
		Gusi yang turun membuat gigi terlihat lebih panjang	0,6	0,8	0,6	0,67
		Gigi goyang	0,7	0,8	0,8	0,77
12.	Persistensi Gigi Sulung	Gigi sulung yang tak kunjung tanggal	0,8	1	1	0,93
		Timbulnya benih gigi dengan posisi yang abnormal	0,8	0,7	0,8	0,77
13.	Pulpitis Reversible	Ada lubang pada gigi	0,7	0,8	0,8	0,77
		Gigi terasa ngilu dan lebih sensitif terhadap rasa manis, panas, atau dingin	0,7	0,8	0,7	0,73
		Rasa sakit tajam hanya sebentar	0,7	0,8	0,7	0,73
14.	Pulpitis Irreversible	Ada lubang pada gigi	0,7	0,8	0,7	0,73
		Gigi terasa sakit atau berdenyut	0,7	0,8	0,6	0,70
		Gigi yang berlubang terasa sakit bila masuk makanan	0,3	0,7	0,8	0,60
		Nyeri sampai ke daerah sinus, pelipis, mata atau telinga	0,6	0,7	0,5	0,60
15.	Stomatitis (Sariawan)	Terdapat luka berbentuk oval atau bulat yang berwarna putih /kuning dan tepi yang merah di dalam rongga mulut	0,6	0,9	1	0,83
		Nyeri saat luka tersentuh	0,7	1	1	0,90

DAFTAR RIWAYAT HIDUP



HANA MAULINDA. Lahir di Jakarta, 07 Agustus 1995. Anak kedua dari pasangan Bapak Hidayat dan Ibu Neneng Yana Marlina. Saat ini beralamatkan di Jalan Linggarsari No. 37A Kalisari, Jakarta Timur.

No. Ponsel : 081318400299

Email : hanamaulinda@gmail.com

Riwayat Pendidikan : Penulis mengawali pendidikan di TK Islam Azzahra Bekasi pada tahun 2000 - 2002, dan kemudian melanjutkan pendidikan di SD Jatirasa III Bekasi dari kelas I sampai kelas III pada tahun 2002 - 2004 dan kemudian kembali melanjutkan ke SDN Pekayon 16 Pagi Jakarta dari kelas IV sampai kelas VI pada tahun 2004 - 2007. Setelah itu, penulis melanjutkan ke SMP Negeri 179 Jakarta hingga tahun 2010. Kemudian penulis melanjutkan ke SMA Negeri 98 Jakarta pada tahun 2010 - 2013. Di Tahun yang sama penulis melanjutkan ke Universitas Negeri Jakarta (UNJ), Program Studi Ilmu Komputer, melalui jalur UMBPTN. Di pertengahan tahun 2017 (Jumat, 18 Agustus 2017) penulis telah memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta.

Riwayat Organisasi : Selama di bangku perkuliahan, penulis aktif di organisasi keilmiahan Program Studi Ilmu Komputer sebagai anggota. Penulis juga berpartisipasi dalam kegiatan Biner (Be Innovative and Educated Researcher) yaitu kegiatan workshop dan seminar yang diadakan oleh DEFAULT, dimana penulis tergabung sebagai anggota panitia.

Riwayat Penelitian : Selain mengikuti organisasi, peneliti juga aktif dalam kegiatan

Penelitian Dosen FMIPA UNJ bersama Muhammad Eka Suryana, M.Kom dan Dra.
Sri Sudaryati, M.Pd tahun 2015 dan 2016.