

LAPORAN TUGAS AKHIR

IMPLEMENTASI PERBAIKAN DETEKSI KELAINAN GINJAL PADA MANUSIA DENGAN METODE *FORWARD* *CHAINING* BERBASIS WEB

IMPLEMENTATION OF IMPROVED DETECTION OF KIDNEY DISORDERS IN HUMANS WITH WEB BASED *FORWARD* *CHAINING* METHOD

Diajukan untuk memenuhi salah satu syarat memperoleh gelar Sarjana Teknik
Informatika

Bidang: Teknologi *Web*



Disusun oleh:

Nama : Anisha Pratiwi

Nim : 1955201110004

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH BANJARMASIN

TAHUN 2022/2023

ABSTRAK

Penyakit ginjal merupakan masalah kesehatan yang serius dan seringkali sulit untuk dideteksi secara dini. Dalam penelitian ini, kami merancang dan membangun sebuah Sistem Pakar Diagnosa Penyakit Ginjal Berbasis Web menggunakan metode Forward Chaining. Sistem ini bertujuan untuk membantu masyarakat dalam mengenali gejala awal penyakit ginjal dan memberikan rekomendasi tindakan medis yang diperlukan. Dengan menggunakan sistem ini, pengguna dapat mendapatkan informasi yang akurat dan langsung dari dokter atau pakar dalam mendiagnosa penyakit ginjal. Sistem ini diharapkan dapat meningkatkan kesadaran masyarakat tentang pentingnya kesehatan ginjal dan memfasilitasi proses diagnosis yang lebih cepat dan efisien.

Kata kunci: Sistem Pakar, penyakit ginjal, deteksi awal, metode Forward Chaining, web.

ABSTRACT

Kidney disease is a serious health problem that is often difficult to detect early. In this study, we designed and built a Web-based Expert System for Kidney Disease Diagnosis using the Forward Chaining method. This system aims to assist the public in recognizing early symptoms of kidney disease and providing necessary medical recommendations. By using this system, users can obtain accurate and direct information from doctors or experts in diagnosing kidney disease. This system is expected to increase public awareness of the importance of kidney health and facilitate faster and more efficient diagnosis processes.

Keywords: *Expert System, kidney disease, early detection, Forward Chaining method, web.*

DAFTAR ISI

ABSTRAK	i
DAFTAR ISI	ii
DAFTAR TABEL	iv
DAFTAR GAMBAR	v
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat	4
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI	5
2.1 Tinjauan Studi	5
2.1.1 Sistem Pakar Penyakit Ginjal Berbasis Web	5
2.1.2 Sistem Pakar Diagnosis Penyakit Ginjal Dengan Metode Forward Chaining5	
2.1.3 Hubungan Antara Kadar Hemoglobin Dengan Kadar Kreatinin Serum Penderita Penyakit Ginjal Menahun (kronis)	6
2.1.4 Implementasi Metode <i>Forward Chaining</i> Dalam Sistem Pendeteksi Kerusakan <i>Hardware</i> Pada Komputer dan Laptop Berbasis Android.....	7
2.1.5 Rancang Bangun Aplikasi Identifikasi Gaya Belajar Siswa Dengan Metode <i>Forward Chaining</i> (<i>Studi Kasus: Sekolah Dasar Negeri Sumampir</i>)	7
2.2 Tinjauan Pustaka	7
2.2.1 Penyakit Ginjal.....	7
2.2.2 Sistem Pakar.....	11
2.2.3 Forward Chaining.....	12
2.2.4 PHP	13
2.2.5 MySQL.....	14
2.2.6 Basis Data.....	14
2.2.7 Laravel.....	15
2.2.8 XAMPP	15
2.2.9 Visual Studio Code	16
2.2.10 Flowchart.....	17

2.2.11	UML	18
2.2.12	Alur Sistem.....	23
2.3	Kerangka Pemikiran	24
BAB III METODOLOGI PENELITIAN.....		25
3.1	Metode Pengembangan Sistem.....	25
3.1.1	Perencanaan Kebutuhan	25
3.1.2	Usulan Pemecahan Masalah.....	27
3.1.3	Desain Sistem.....	28
3.2	Analisi Kebutuhan Sistem	60
3.2.1	Kebutuhan Fungsional	60
3.2.2	Kebutuhan Non Fungsional.....	60
BAB IV		61
ANALISIS HASIL PENELITIAN DAN PEMBAHASAN		61
4.1.1	Implementasi Basis Data.....	61
4.1.2	Implementasi <i>Interface</i>	67
BAB V.....		90
KESIMPULAN DAN PENELITIAN SELANJUTNYA.....		90
DAFTAR PUSTAKA		90

DAFTAR TABEL

Tabel 2. 1 Tabel Penyakit.....	8
Tabel 2. 2 Tabel Gejala	9
Tabel 2. 3 Tabel Saran	10
Tabel 2. 4 simbol Flowchart.....	17
Tabel 2. 5 simbol Use Case Diagram.....	19
Tabel 2. 6 simbol sequence diagram	20
Tabel 2. 7 Class Diagram.....	22
Tabel 3. 1 Tabel User	48
Tabel 3. 2 Tabel Aturan	48
Tabel 3. 3 Tabel Pasien	48
Tabel 3. 4 Tabel Gejala	49
Tabel 3. 5 Tabel Gejala Pasien.....	49
Tabel 3. 6 Tabel Penyakit.....	49
Tabel 3. 7 Tabel solusi	50
Tabel 3. 8 Tabel Riwayat Konsultasi	50
Tabel 4. 1 Tabel pengujian black box halaman login	68
Tabel 4. 2 Tabel pengujian black box halaman data pasien.....	70
Tabel 4. 3 Tabel pengujian black box halaman data gejala	72
Tabel 4. 4 Tabel pengujian black box halaman penyakit.....	75
Tabel 4. 5 Pengujian black box pada halaman solusi.....	77
Tabel 4. 6 Pengujian black box pada halaman aturan.....	79
Tabel 4. 7 Pengujian black box pada halaman pasien.....	82
Tabel 4. 8 Pengujian Black Box pada halaman konsultasi dan hasil konsulasi	85

DAFTAR GAMBAR

Gambar 2. 1 forward chaining	12
Gambar 3. 1 Metode Waterfall.....	25
Gambar 3. 2 Flowchart Login	29
Gambar 3. 3 Flowchart Data Pasien.....	30
Gambar 3. 4 Flowchart Input Data Gejala	31
Gambar 3. 5 Flowchart Edit Data Gejala	32
Gambar 3. 6 Flowchart Hapus Data Gejala	33
Gambar 3. 7 Flowchart Input Data Penyakit.....	34
Gambar 3. 8 Flowchart Edit Data Penyakit	35
Gambar 3. 9 Flowchart Hapus Data Penyakit.....	36
Gambar 3. 10 Flowchart Input Data Solusi.....	37
Gambar 3. 11 Flowchart Edit Data Solusi	38
Gambar 3. 12 Flowchart Hapus Data Solusi	39
Gambar 3. 13 Flowchart Data Aturan	40
Gambar 3. 14 Flowchart Konsultasi.....	41
Gambar 3. 15 Flowchart Riwayat Diagnosa	43
Gambar 3. 16 Use Case Diagram.....	44
Gambar 3. 17 Sequence Diagram Admin	45
Gambar 3. 18 Sequence Diagram Pengguna.....	46
Gambar 3. 19 Class Diagram	47
Gambar 3. 20 Rancangan Tabel Relasi	51
Gambar 3. 21 Tampilan Halaman Registrasi Akun	52
Gambar 3. 22 Tampilan Halaman Login.....	53
Gambar 3. 23 Tampilan halaman beranda admin	54
Gambar 3. 24 Halaman Data pasien.....	54
Gambar 3. 25 Halaman Data gejala	55
Gambar 3. 26 Halaman Penyakit	56
Gambar 3. 27 Halaman Aturan	57
Gambar 4. 1 Tabel Users.....	61

Gambar 4. 2 Tabel Aturan.....	62
Gambar 4. 3 Tabel Data Pasien.....	63
Gambar 4. 4 Tabel Gejala	63
Gambar 4. 5 Tabel Gejala Pasien.....	64
Gambar 4. 6 Tabel Penyakit.....	64
Gambar 4. 7 Tabel Solusi.....	65
Gambar 4. 8 Tabel Hasil Diagnosa	66
Gambar 4. 9 Tabel riwayat diagnosa.....	66
Gambar 4. 10 Halaman Login.....	68
Gambar 4. 11 Halaman Data Pasien	70
Gambar 4. 12 Halaman Gejala	72
Gambar 4. 13 Halaman penyakit.....	74
Gambar 4. 14 Halaman solusi	76
Gambar 4. 15 Halaman Aturan	79
Gambar 4. 16 Halaman pasien	81
Gambar 4. 17 Halaman masukan ID pasien.....	81
Gambar 4. 18 Halaman masukan datadiri pasien.....	82
Gambar 4. 19 Halaman konsultasi	84
Gambar 4. 20 Halaman Hasil konsultasi.....	85
Gambar 4. 21 Halaman Riwayat Diagnosa	88
Gambar 4. 22 Halaman Detail Riwayat Diagnosa	88

BAB I

PENDAHULUAN

1.1 Latar Belakang

Fungsi utama ginjal adalah untuk menyaring zat-zat yang tidak diperlukan dalam darah, seperti urea, dan membuangnya keluar dari tubuh melalui urine. Selain itu, ginjal juga berperan dalam menjaga keseimbangan asam dan basa dalam tubuh dan menghilangkan zat-zat sisa dari metabolisme tubuh yang tidak dibutuhkan[1]. Ginjal yang tidak berfungsi dengan baik akan memberikan tanda – tanda pada tubuh, meskipun terkadang tidak terlalu dirasakan oleh penderita. Ginjal merupakan organ yang berukuran 11x7x6 cm³[2]. Kurangnya kesadaran masyarakat akan kesehatan menjadi faktor banyaknya masyarakat yang menderita penyakit ginjal.

Penyakit ginjal merupakan suatu penyakit yang memerlukan seorang pakar dalam penanganan untuk membantu mencegah ataupun mendiagnosa, dan penyakit ini sering mengancam nyawa seseorang tanpa memandang usia[3]. Maka dari itu penyakit ginjal merupakan penyakit yang harus dihindari oleh semua orang. Penyakit ginjal sulit terdeteksi, dan sangat berdampak buruk terhadap kondisi penderita dari waktu ke waktu. Hal ini pula dapat dihindari kalau penyakit ginjal dapat dideteksi lebih awal. Pemilihan masalah jenis penyakit ginjal sebagai sampel dalam penelitian ini, adalah kenyataan bahwa ginjal merupakan organ penting dalam sistem metabolisme tubuh, karena padatnya aktivitas dari masyarakat sering lupa untuk memperhatikan kesehatan organ dalam tubuh terutama pada ginjal.

Seorang pakar atau ahli dalam bidang penyakit ginjal dapat mengetahui ciri-ciri seseorang terkena penyakit ginjal dari gejala-gejala ringan yang dialami oleh penderita. Tetapi, terbatasnya tenaga dan waktu seorang pakar yang mana tidak dapat setiap saat melayani pasien, kemudian mahalnya biaya untuk berobat kepada seorang pakar atau dokter menjadi alasan utama masyarakat untuk tidak menindaklanjuti pengobatan dari penyakit yang dideritanya, yang mana mungkin belum tentu terdiagnosa terkena penyakit ginjal. Oleh karena itu penulis berpikir bahwa perlu adanya sebuah sistem pakar untuk mengatasi masalah tersebut. Sistem

pakar atau *expert system* biasa disebut juga dengan *knowledge based system* yaitu suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik, yang menggunakan pengetahuan dan metode analisis dimana akan diidentifikasi terlebih dahulu oleh pakar yang sesuai bidang keahliannya[4]. Ada beberapa metode yang dapat digunakan dalam sistem pakar untuk mendeteksi penyakit. Pada sistem ini akan digunakannya sebuah metode yaitu, metode *Forward Chaining* untuk memprediksi atau mencari solusi dari suatu masalah yang dimulai dengan sekumpulan fakta yang didapatkan dari seorang pakar, dan juga data yang didapatkan dari data rekam medik pada suatu instansi kesehatan yang kemudian akan disimpulkan berdasarkan fakta yang didapatkan.

Metode *Forward Chaining* adalah mekanisme pencocokan fakta atau pernyataan yang dimulai dari kondisi (IF) terlebih dahulu dengan aturan (IF-THEN)[5]. Metode ini cocok untuk mengolah pengetahuan seperti gejala awal penyakit ginjal, yang kemudian akan dicocokkan alurnya sehingga menghasilkan suatu kesimpulan pada penyakit ginjal yang dialami oleh penderita. Sehingga pengguna dapat mengetahui lebih awal gejala penyakit ginjal yang dialaminya, yang mana kemudian dapat dengan cepat dilakukannya tindakan secara lanjut ke penanganan medis langsung dengan ahlinya.

Berdasarkan latar belakang yang telah dijelaskan, diperlukan suatu teknologi komputer untuk membantu masyarakat agar mengetahui bahwa gejala yang sedang dideritanya merupakan gejala awal dari penyakit ginjal yang kemudian dapat dilakukannya tindaklanjut oleh tenaga medis. Yang mana, sistem ini akan diterapkan dalam bentuk *web*, sehingga masyarakat dapat menggunakan aplikasi ini kapan saja dengan jaringan *internet*. Dengan harapan aplikasi ini dapat membantu masyarakat dalam mendeteksi gejala awal penyakit ginjal berdasarkan gejala yang dialami.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan maka dapat disusun rumusan masalah yaitu bagaimana membuat sistem pakar untuk deteksi awal penyakit ginjal pada manusia menggunakan metode *forward chaining*?

1.3 Batasan Masalah

Dari permasalahan di atas, berikut ini diberikan Batasan masalah untuk menghindari melebar nya masalah yang akan diselesaikan:

1. Sistem deteksi penyakit ginjal ini hanya untuk mendeteksi penyakit yang berkaitan dengan penyakit ginjal kronik yang telah diinput pada database sistem berdasarkan data yang telah didapatkan oleh penulis.
2. Data yang diinput merupakan data gejala yang telah ditentukan di dalam database sistem sesuai dengan data yang telah didapatkan oleh penulis.
3. Pembuatan sistem pakar hanya membantu mendiagnosa pasien sesuai gejala yang ada pada database sistem kepada *user*, jika *user* merasa kurang yakin maka *user* dapat langsung mendatangi dokter spesialis penyakit dalam ginjal.
4. Membangun dan merancang Sistem Pakar Diagnosa Penyakit Ginjal Pada Manusia dengan menggunakan metode *Forward Chaining* sebagai metode penalaran dan mesin inferensi.

1.4 Tujuan Penelitian

Tujuan yang dicapai dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Terciptanya sebuah sistem untuk mendeteksi penyakit ginjal kronik pada manusia yang dapat digunakan oleh masyarakat kapan dan di mana saja.
2. Merancang dan membangun Sistem Pakar Diagnosa Penyakit Ginjal Pada Manusia Berbasis Website untuk memfasilitasi dokter atau pakar dalam mendeteksi gejala penyakit Ginjal.

1.5 Manfaat

Adapun manfaat dari merancang dan membangun Sistem Pakar Diagnosa Penyakit Gagal Ginjal Kronik pada Manusia Dengan Berbasis Website adalah sebagai berikut:

1. Sebagai sarana alternatif dalam membantu masyarakat mengidentifikasi penyakit ginjal, sehingga dapat segera ditangani oleh seorang pakar atau dokter.
2. Sebagai sarana alternatif untuk membantu dokter mendeteksi gejala penyakit ginjal pada user dimana dan kapan saja.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Studi

2.1.1 Sistem Pakar Penyakit Ginjal Berbasis Web

Kurangnya pengetahuan mengenai gejala awal dan fasilitas kesehatan terkait penyakit ginjal menyebabkan semakin meningkatnya angka kematian yang disebabkan oleh penyakit ginjal di Indonesia. Oleh karena itu, diperlukan suatu teknologi komputer untuk memudahkan dalam mendiagnosa penyakit ginjal sehingga dapat diketahui dan bisa ditindaklanjuti segera. Dalam penelitian ini penulis menjelaskan bahwa dengan menggunakan aplikasi berbasis web, informasi dari suatu pakar akan mudah didapatkan oleh pengguna, tanpa harus dating pada seorang ahli/pakar yang ahli pada bidangnya.

Sistem pakar atau *expert system* ialah suatu perangkat lunak komputer yang memiliki basis pengetahuan untuk domain tertentu dan menggunakan penalaran inferensi menyerupai seorang pakar dalam menyelesaikan suatu masalah, dengan tujuan membantu banyak orang dalam mengimplementasikan ilmu para pakar tanpa menggantikan peran seorang ahli/pakar.

Pada jurnal ini disebutkan jumlah gejala yaitu 54 gejala dan jumlah penyakit yaitu 8 penyakit ginjal dalam tabel pakar. Kemudian penulis melakukan pencocokan *rule* dengan aturan-aturan yang digunakan pada sistem pakar dalam mengidentifikasi penyakit ginjal[6].

2.1.2 Sistem Pakar Diagnosis Penyakit Ginjal Dengan Metode Forward Chaining

Keterlambatan penanganan terhadap penyakit ginjal akan berakibat fatal bagi penderitanya. Penyakit ginjal sendiri masuk ke dalam penyakit kardiovaskular yaitu penyakit yang beresiko tinggi, yang memiliki angka kematian dan biaya perawatan yang tinggi pula. Penyebab utama banyak masyarakat terserang penyakit ginjal karena faktor ekonomi dan kurangnya pengetahuan sehingga tindakan yang diambil dalam penanganan sangat lambat. Sistem yang dibangun digunakan untuk membantu penderita mengetahui penyakit ginjal yang sedang diderita berdasarkan

gejala, juga mengetahui informasi terkait penyakit ginjal seperti pencegahan dan perawatan yang dapat dilakukan. Dari pengujian dan pelacakan (diagnosis), sistem yang dibuat dapat melakukan diagnose penyakit ginjal berdasarkan gejala, jenis penyakit dan *rule* yang telah dibangun. Kelebihan sistem yang dibuat dapat memberikan keterangan dan solusi terhadap penyakit yang terdiagnosa oleh sistem[7].

2.1.3 Hubungan Antara Kadar Hemoglobin Dengan Kadar Kreatinin Serum Penderita Penyakit Ginjal Menahun (kronis)

penelitian ini dilakukan untuk menentukan derajat anemia berdasarkan kadar hemoglobin dan melihat hubungan antara kadar hemoglobin dengan kadar kreatinin serum pada penderita PGK (Penyakit Ginjal Kronis). Penyakit ini disebabkan penurunan fungsi ginjal secara perlahan, terus-menerus dan meningkat dalam kurun waktu 3 bulan atau lebih yang ditandai dengan kelainan patologi ginjal. Pengujian penelitian ini dilakukan dengan menggunakan *autoanalyzer Lyasis* menggunakan metode kolorimetrik enzimatis dan pentra 80 menggunakan metode penghalangan (impedans). Kemudian data yang diperoleh akan diolah dengan metode SPSS 11,5. Pada penelitian ini sebagian besar penderita PGK mengalami anemia berat (38 orang atau 76%), sedangkan anemia ringan dan sedang masing-masing 3 orang (6%) dan 9 orang (18%). Anemia sendiri merupakan tampakan (manifestasi) yang banyak ditemukan di PGK sebab terjadinya penurunan eritropoitein (EPO) di 90% kasus, yang disebabkan kegagalan fungsi endokrin dan buangan (ekskretori) ginjal. Pada penelitian ini dijelaskan bahwa semua penderita anemia berat memiliki kadar kreatinin serum 3,5 mg/dl (76%), sedangkan anemia ringan hanya 4%, dan anemia sedang 12%. Penderita anemia ringan yang memiliki kadar kreatinin serum $\leq 3,5$ mg/dl sebanyak 2%, dan anemia sedang 6%. Gagal ginjal yang parah (progresif) dengan kadar kreatinin serum 3 mg/dl, dihubungkan dengan anemia berat[8].

2.1.4 Implementasi Metode *Forward Chaining* Dalam Sistem Pendeteksi Kerusakan *Hardware* Pada Komputer dan Laptop Berbasis Android

Pada penelitian ini masalah yang ada dibatasi hanya pada 21 (dua puluh satu) gejala/jenis kerusakan yang sering terjadi pada *PC* dengan 6 gejala/jenis kerusakan pada laptop, kesimpulan letak kerusakan yang dialami dan memberikan cara panganan atau solusi. Penelitian yang diharapkan dapat membantu pengguna mengetahui letak kerusakan pada komputer dan laptop serta membantu mengambil tindakan segera dalam penanganan kerusakan pada komputer atau laptop ini menggunakan metode *prototype* dalam proses pengembangannya seperti, sistem pendeteksi kerusakan *hardware* pada komputer dan laptop[9].

2.1.5 Rancang Bangun Aplikasi Identifikasi Gaya Belajar Siswa Dengan Metode *Forward Chaining* (*Studi Kasus: Sekolah Dasar Negeri Sumampir*)

Tujuan dari aplikasi ini membantu pengajar untuk mengetahui gaya belajar dari masing-masing siswanya agar proses pembelajaran dan pengerjaan tugas lebih efektif dan kondusif untuk mendapatkan hasil belajar yang baik serta menyenangkan bagi siswa. Siswa yang jenuh saat belajar disebabkan guru yang kurang mengetahui gaya belajar yang dimiliki tiap siswa menjadi permasalahan dalam penelitian ini. Yang mana pada hasil uji coba terhadap sistem yang sesuai akan diterapkan pada Sekolah Dasar Negeri Sumampir[10].

2.2 Tinjauan Pustaka

2.2.1 Penyakit Ginjal

Kelainan pada organ ginjal dapat terjadi karena berbagai faktor seperti infeksi, tumor, kelainan bawaan, penyakit metabolik, degeneratif, dan faktor lainnya yang memicu terjadinya penyakit ginjal. Penyakit ginjal dapat mempengaruhi fungsi ginjal dalam menyaring zat-zat berbahaya dan limbah metabolik dalam darah, yang pada gilirannya dapat memengaruhi kesehatan tubuh secara keseluruhan. Oleh karena itu, sangat penting untuk menerapkan gaya hidup yang sehat dan melakukan pemeriksaan kesehatan secara teratur untuk menjaga kesehatan ginjal. [11]. Penyakit ginjal juga merupakan gangguan yang terjadi pada organ ginjal, penyakit ginjal dapat disebabkan oleh kondisi seperti karena diabetes

dan tekanan darah tinggi, yang mana sangat beresiko bagi penderita. Beberapa jenis penyakit ginjal antara lain;

- a. Gagal ginjal, dapat terjadi ketika bakteri dari saluran kemih menyebar ke salah satu atau kedua ginjal. Hal ini merupakan hasil dari komplikasi infeksi saluran kemih.
- b. Batu ginjal, terjadi ketika garam dan mineral yang seharusnya disaring oleh ginjal mengeras dan menumpuk di dalam ginjal. Kondisi ini dapat disebabkan oleh urine yang terlalu pekat sehingga kristalisasi garam dan mineral terjadi.
- c. Gagal ginjal akut, kondisi dimana tiba-tiba ginjal tidak dapat berfungsi dengan normal yang menyebabkan menumpuknya garam dan zat kimia lainnya dalam tubuh yang mempengaruhi fungsi organ tubuh yang lain.
- d. Penyakit ginjal kronis atau gagal ginjal, penurunan fungsi ginjal hingga kurun waktu tiga bulan. Beberapa gejalanya seperti sesak napas, mual dan kelelahan. Kondisi ini tidak dirasakan oleh penderita yang masih stadium 1-3, itulah sebab banyak orang yang tidak menyadari bahwa dirinya sedang mengalami kondisi ini hingga mencapai stadium lanjutan[4].

Table 2.1 dan 2.2 di bawah merupakan tabel penyakit dan tabel gejala.

Tabel 2. 1 Tabel Penyakit

Kode	Penyakit
P01	Gagal Ginjal Kronik
P02	Hipertensi
P03	Diabetes Melitus
P04	Anemia Normositik Normokrom
P05	Kolestrol
P06	Diabetic Foot Ulcer
P07	Pneumonia
P08	Gagal Jantung Kronik
P09	Gagal Jantung
P10	Melena
P11	Asites
P12	HIV
P13	Hepatitis B
P14	Hidronefrosis
P15	Leukositosis
P16	Hiperkalemia

P17	Anemia
-----	--------

Tabel 2. 2 Tabel Gejala

Kode	Gejala
G01	Badan Lemas
G02	Pucat
G03	Rutin Hemodialisa
G04	Sesak Nafas
G05	Nyeri Pinggul
G06	Nyeri Perut
G07	Nyeri Kepala
G08	Nyeri Kemaluan
G09	Nyeri Dada
G10	Nyeri Ulu Hati
G11	Nyeri Pinggang
G12	BAB Berlendir
G13	BAB Cair
G14	Mual-Muntah
G15	Penurunan Kesadaran
G16	Tidak Selera Makan
G17	Susah Buang Air Kecil
G18	Perut Membesar
G19	Muntah Berdarah
G20	Batuk Berdarah
G21	Urine Berdarah
G22	Gusi Berdarah
G23	Pembengkakan Pada Kaki
G24	Pembengkakan Pada Paha
G25	Pembengkakan Pada Perut
G26	Pembengkakan Area Mata
G27	Demam
G28	Hemoglobin Rendah
G29	Fases Hitam
G30	Buang Air Kecil Sakit
G31	Kepala Pusing
G32	Kulit Kering Gatal
G33	Sering Buang Air Kecil
G34	Susah Tidur
G35	Keras Otot
G36	Kaki Luka
G37	Gangguan Pencernaan

Tabel 2. 3 Tabel Saran

Kode	Saran
S01	Diet Rendah Garam
S02	Olahraga Teratur
S03	Tidak Merokok
S04	Konsumsi Antibiotik
S05	Hemodialisis
S06	Transplantasi Darah
S07	Tidak Mengonsumsi Alkohol
S08	Diet Gizi Seimbang
S09	Menjaga Berat Badan Ideal
S10	Diet Gula
S11	Diet Kalori
S12	Mengonsumsi Serat Cukup
S13	Perawatan Luka
S14	Melakukan Pemeriksaan Berkala
S15	Perawatan Kaki
S16	Terapi Oksigen
S17	Konsumsi Obat Pereda Nyeri
S18	Istirahat yang Cukup
S19	Vaksin Pneumonia
S20	Fisioterapi
S21	Diet Rendah Lemak
S22	Transplantasi Jantung
S23	Pemberian Cairan Intravena
S24	Melakukan Tindak Endoskopi
S25	Pemeriksaan Saluran Pencernaan Atas
S26	Diet Rendah Protein
S27	Terapi Antiretroviral (ART)
S28	Terapi Gizi dan Fisik
S29	Penanganan Infeksi
S30	Terapi Antiviral
S31	Imunisasi Hepatitis B
S32	Menjaga Pola Makan Sehat
S33	Pencegahan Penularan
S34	Pemasangan Kateter
S35	Mengonsumsi Air Putih Cukup
S36	Transfusi Darah
S37	Terapi Leukofereisis

S38	Pemberian Kalsium
S39	Pemberian Insulin
S40	Pemberian Glukosa
S41	Diet Kalium

2.2.2 Sistem Pakar

Knowledge Based System atau Sistem Pakar adalah sebuah aplikasi komputer yang dirancang untuk membantu pengambilan keputusan atau memecahkan masalah dalam bidang yang spesifik. Sistem ini mengandalkan pengetahuan dan metode analisis yang diidentifikasi oleh para pakar di bidang keahlian tertentu[4]. Fungsi dari sebuah sistem adalah keterhubungan dari beberapa komponen yang berinteraksi secara kolektif terhadap sebuah kebutuhan dengan menggunakan perangkat keras, perangkat lunak, prosedur manual, dan model-model untuk analisis, perencanaan, pengendalian, pengambilan keputusan, dan basis data[12].

Beberapa karakteristik atau ciri dari sistem pakar antara lain:

- Terbatas pada bidang keahlian tertentu.
- Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
- Berdasarkan pada *rule* tertentu.
- Dapat mengemukakan rangkaian alasan yang diberikan dengan cara yang mudah untuk dipahami.
- Mudah dimodifikasi.

Berikut hal-hal yang menjadi keuntungan sistem pakar adalah sebagai berikut:

- Meningkatkan produktivitas, disebabkan sistem pakar dapat bekerja lebih cepat dari manusia.
- Meningkatkan kualitas, dengan memberi nasehat yang konsisten serta mengurangi kesalahan.
- Meningkatkan kemampuan dalam menyelesaikan masalah sebab sistem pakar mengambil sumber pengetahuan dari banyak pakar.

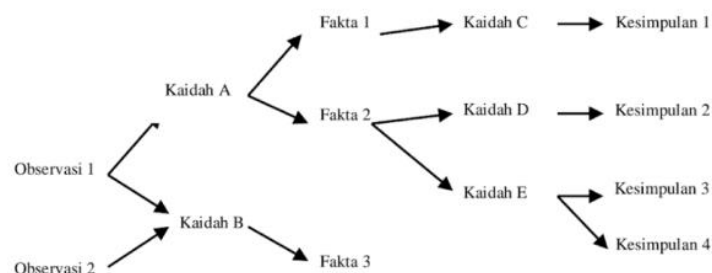
- d. Dapat menangkap pengetahuan serta keahlian seorang pakar.
- e. Meningkatkan kapabilitas sistem komputer. Yang mana integrasi sistem pakar dengan komputer yang lain dapat membuat sistem lebih efektif dan mencakup lebih banyak aplikasi
- f. Menghemat waktu dalam pengambilan keputusan

Disamping dari keuntungan sistem pakar juga memiliki beberapa kelemahan antara lain, sebagai berikut:

- a. Tingginya biaya pembuatan serta pemeliharaan yang diperlukan.
- b. Sulit dikembangkan. Hal ini erat kaitannya dengan ketersediannya pakar di bidangnya.
- c. Sistem pakar tidak 100% bernilai benar[3].

2.2.3 Forward Chaining

Pelacakan kedepan (*Forward Chaining*) yaitu memulai dari sekumpulan data menuju pada kesimpulan, seperti gambar di bawah ini:



Gambar 2. 1 forward chaining

Forward Chaining adalah Teknik pencarian yang dimulai dari fakta-fakta yang diketahui, kemudian dilakukan pencocokan fakta-fakta tersebut dengan bagian IF dari aturan IF-THEN. Hal ini dilakukan untuk mendapatkan kesimpulan yang diinginkan berdasarkan alasan yang berasal dari fakta-fakta yang ada[13]. Jika terdapat fakta yang sesuai dengan bagian IF, maka rule akan dijalankan dan menghasilkan fakta baru dari bagian THEN, yang kemudian akan ditambahkan ke dalam database. Pencocokan aturan dimulai dari rule teratas dan hanya dieksekusi satu kali. Proses pencocokan akan berhenti jika tidak ada lagi rule yang dapat dieksekusi[7].

Dalam mekanisme forward chaining, fakta-fakta yang diketahui dimasukkan ke dalam database, lalu dicocokkan dengan bagian IF dari aturan IF-THEN. Jika terdapat fakta yang sesuai dengan aturan IF, maka aturan tersebut akan dieksekusi. Setelah dieksekusi, akan muncul fakta baru (THEN) yang ditambahkan ke dalam database. Dalam konteks ini, IF dapat direpresentasikan sebagai gejala-gejala yang dirasakan oleh pasien, sedangkan THEN dapat berarti diagnosa awal dari penyakit pasien[14].

$$P(A) = \frac{n(A)}{n(s)} * 100 \quad [1]$$

Rumus 1 merupakan rumus probabilitas yang digunakan untuk menghitung Metode *Forward Chaining*. Adapun penjelasannya adalah sebagai berikut:

P = probabilitas

A = peristiwa

P(A) = probabilitas kejadian A

n(A) = jumlah A terjadi

n(s) = jumlah semua kejadian dalam ruang sampel terjadi

contoh perhitungan *Forward Chaining* terdapat pada lampiran ke-4.

2.2.4 PHP

PHP merupakan singkatan dari *Hypertext Preprocessor*. PHP adalah bahasa pemrograman web server-side yang open source. PHP digunakan untuk membuat skrip yang terintegrasi dengan HTML dan dapat dijalankan pada beberapa server, sehingga memungkinkan pembuatan halaman web yang dinamis[15]. Kelebihan dari PHP, yaitu;

- a. Bahasa pemrograman PHP ialah sebuah *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relative mudah.

- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis – milis dan developer yang siap membantu dalam pengembangan.
- d. Dalam sisi pemahaman, PHP ialah bahasa scripting yang paling mudah karena memiliki referensi yang banyak
- e. PHP merupakan bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui *console* serta juga dapat menjalankan perintah-perintah sistem[16].

2.2.5 MySQL

MySQL merupakan turunan dari database SQL (Structured Query Language). Konsep SQL memungkinkan pengoperasian database, terutama dalam pemilihan atau seleksi serta pemasukan data, sehingga pengoperasian data dapat dilakukan secara otomatis dengan mudah. MySQL sendiri merupakan produk Relational Database Management System (RDBMS) yang dapat dijalankan pada sistem operasi UNIX, Linux, dan Windows. Selain itu, MySQL juga didistribusikan secara gratis dengan lisensi GPL[16].

2.2.6 Basis Data

Pengertian basis data ialah kumpulan koleksi data-data yang saling berhubungan secara logika yang isi di dalamnya didesain untuk memenuhi kebutuhan informasi dari suatu perusahaan.

Ada beberapa istilah umum yang sering dipakai dalam *database*, yaitu sebagai berikut:

- a. *Field*, sekumpulan kecil dari kata atau sebuah deteran angka-angka.
- b. *Record*, kumpulan dari *field* yang berelasi secara logis.
- c. *File*, kumpulan dari *record* yang berelasi secara logis.
- d. *Entity*, orang, tempat, benda atau kejadian yang berkaitan dengan informasi yang disimpan.
- e. *Attribute*, setiap karakteristik yang menjelaskan suatu *entity*.
- f. *Primary key*, sebuah *field* yang nilainya unik yang tidak sama antara satu *record* dengan *record* yang lainnya.

- g. *Foreign key*, sebuah *field* yang nilainya berguna untuk menghubungkan *primary key* yang berbeda pada beberapa *table*[17].

2.2.7 Laravel

Laravel adalah sebuah framework PHP open-source yang gratis, yang dibuat oleh Taylor Otwell untuk pengembangan aplikasi web dengan pola MVC (Model-View-Controller).[18].

Ada 5 konsep arsitektur pada Laravel berserta dengan fungsinya, yaitu sebagai berikut:

- a. *Routes*, sebagai pemberi akses pada setiap *request* sesuai alur yang sudah ditentukan.
- b. *Controller*, sebagai penghubung antar model dan *view*. Perintah controller berfungsi untuk memproses bagaimana data ditampilkan dari model ke *view* begitu juga sebaliknya.
- c. Model, sekumpulan data yang memiliki fungsi dalam mengelola suatu table pada database. Struktur pemodelan data pada Laravel memiliki fungsi yang terdiri dari table, *primaryKey* dan *fillable*. Yang mana pada bagian table harus diisi dengan nama table yang sesuai database, pada bagian *primaryKey* harus diisi sesuai *primary key* pada table dan pada bagian *fillable* harus diisi dengan bagian-bagian yang mencakup dalam table tersebut juga.
- d. *View*, file yang berisi kode HTML (HyperText Markup Language) berfungsi menampilkan data ke dalam browser. Format *view* dalam Laravel harus menggunakan format blade, seperti *view.blade.php*.
- e. *Migration*, proses perancangan suatu table dengan fungsi sebagai blueprint database atau dapat diistilahkan sebagai penyedia sistem control untuk skema database.

2.2.8 XAMPP

XAMPP adalah software open-source yang berfungsi sebagai web server, dengan server MySQL dan bahasa pemrograman PHP terintegrasi di dalamnya untuk pembuatan website. XAMPP juga dikenal sebagai standalone server atau server yang dapat berdiri sendiri, yang bertujuan memudahkan pengguna dalam

menjalankan proses pengeditan, desain, dan pengembangan aplikasi. Terdapat tiga komponen utama dalam XAMPP, yaitu web server Apache, PHP, dan MySQL. Bagian XAMPP yang biasa digunakan pada umumnya htdoc, phpMyAdmin, dan Control Panel. XAMPP merupakan singkatan yang masing-masing hurufnya adalah sebagai berikut:

- X : Program dapat dijalankan pada banyak sistem operasi, seperti Windows, Linux, Mac Os, dan Solaris.
- A : Apache, ialah aplikasi webserver. Apache bertugas menampilkan halaman web kepada *user* berdasarkan kode PHP yang dituliskan oleh pembuat halaman website.
- M : MySQL, ialah aplikasi *database* server. Pengembangannya disebut SQL yang merupakan kepanjangan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah *database*. MySQL dapat digunakan untuk membuat serta mengelola *database* beserta isinya. MySQL dapat dimanfaatkan untuk menambahkan, mengubah, dan menghapus data yang berada dalam *database*.
- P : PHP, bahasa pemrograman web. PHP merupakan bahasa pemrograman untuk membuat web yang bersifat *server-side scripting*. PHP memungkinkan untuk membuat halaman web yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP adalah MySQL. PHP juga mendukung sistem manajemen *database* Oracle, Microsoft Access, Interbase, d-base, PostgreSQL, dan sebagainya[19].
- P : Perl, bahasa pemrograman.

2.2.9 Visual Studio Code

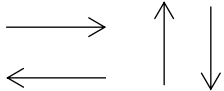
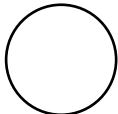
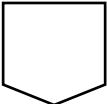
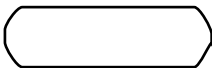

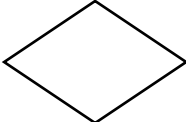
Visual Studio Code (VSCode) adalah sebuah editor kode sumber yang dikembangkan oleh Microsoft untuk sistem operasi Windows, Linux, dan macOS. VSCode dilengkapi dengan berbagai fitur seperti dukungan untuk debugging, integrasi dengan Git yang terhubung ke GitHub, fitur penyorotan sintaksis, penyelesaian kode cerdas, snippet, serta kemampuan untuk melakukan refactoring

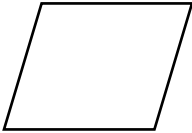



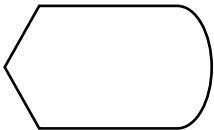

kode. Selain itu, pengguna dapat mengubah tema, preferensi, dan menginstal ekstensi yang menambahkan fungsionalitas tambahan pada editor.[20].

2.2.10 Flowchart

Flowchart adalah sebuah diagram yang menunjukkan aliran atau urutan langkah-langkah yang digunakan untuk merepresentasikan atau merancang sebuah program. *Flowchart* bertujuan untuk menggambarkan tahapan penyelesaian masalah secara sederhana dengan menggunakan simbol-simbol yang dapat dengan mudah dipahami. Alur dalam flowchart biasanya bergerak dari atas ke bawah atau dari kiri ke kanan, dan mengandung simbol-simbol yang mewakili keputusan, tindakan, pengulangan, input/output, dan proses lainnya. Flowchart digunakan sebagai alat bantu untuk memvisualisasikan alur kerja program dengan jelas dan terstruktur[21].

Tabel 2. 4 simbol *Flowchart*

	<p>Flow</p> <p>Simbol yang digunakan untuk menggabungkan antara symbol yang satu dengan symbol yang lain. Simbol ini disebut juga dengan Connecting Line.</p>
	<p>On-Page Reference</p> <p>Simbol untuk keluar masuk atau penyambungan proses dalam lembar kerja yang sama</p>
	<p>Off-Page Reference</p> <p>Simbol untuk keluar masuk atau penyambungan proses dalam lembar kerja yang berbeda</p>
	<p>Terminator</p> <p>Simbol yang menyatakan awal atau akhir suatu program</p>
	<p>Process</p> <p>Simbol yang menyatakan suatu proses yang dilakukan komputer</p>
	<p>Decision</p> <p>Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya atau tidak</p>

	Input/Output Simbol yang menyatakan proses input atau output tanpa tergantung peralatan
	Manual Operation Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer
	Document Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak
	Predefine Process Simbol yang digunakan untuk menyatakan pelaksanaan suatu bagian (sub-program) atau prosedur
	Display Simbol yang menyatakan peralatan output yang digunakan
	Preparation Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal

2.2.11 UML

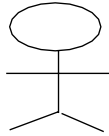
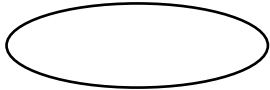

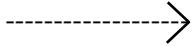

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan visual yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan sistem perangkat lunak dalam bentuk diagram dan notasi standar. UML terdiri dari beberapa jenis diagram seperti diagram use case, diagram class, diagram sequence, dan diagram aktivitas yang dapat membantu para developer dalam memahami dan berkomunikasi mengenai rancangan sistem perangkat lunak secara lebih efektif. UML telah menjadi standar industri dalam pengembangan perangkat lunak dan digunakan oleh banyak organisasi dan perusahaan di seluruh dunia. [22].

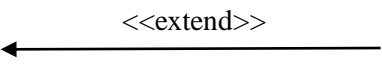
Beberapa jenis dari UML, antara lain:

a. *Use Case Diagram*

Use Case Diagram adalah diagram pemodelan yang digunakan untuk menggambarkan interaksi antara satu atau lebih aktor dan sistem yang akan dibuat. Diagram ini bersifat statis dan mendeskripsikan urutan langkah sistem yang akan dibuat, dengan menunjukkan fungsi-fungsi utama yang akan dilakukan oleh sistem tersebut dalam interaksi dengan aktor-aktor yang terlibat. Tujuan dari *Use Case Diagram* adalah untuk mempermudah pemahaman mengenai fitur-fitur dan fungsi-fungsi utama yang dimiliki oleh sistem, serta untuk membantu dalam perancangan dan pengembangan sistem perangkat lunak.[23].

Tabel 2. 5 simbol *Use Case Diagram*


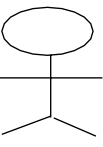
Simbol	Keterangan
	Aktor Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
	Use Case Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case.
	Association Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
	Generalisasi Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
	Include Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan


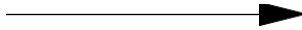

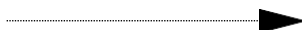
	<p>fungsinya atau sebagai syarat dijalankan <i>use case</i> ini Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ul style="list-style-type: none"> - <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan - <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan di jalankan
	<p>Extend</p> <p>Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan</p>

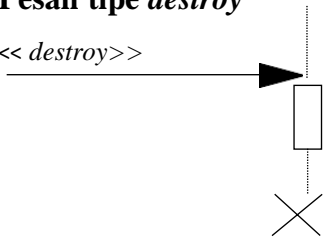
b. *Sequence Diagram*

Sequence diagram adalah sebuah diagram yang memperlihatkan urutan dari serangkaian aktivitas atau aksi yang dilakukan oleh objek-objek pada suatu use case. Diagram ini mendeskripsikan langkah-langkah objek serta pesan atau komunikasi yang dikirimkan dan diterima antar objek yang terlibat[23].

Tabel 2. 6 simbol *sequence diagram*

Simbol	Deskripsi
<p>Garis hidup/lifeline</p> 	Menyatakan kehidupan suatu objek
<p>Aktor</p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi

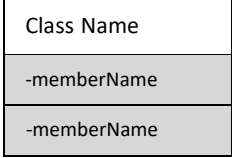


<p>Atau Nama Aktor</p> <p>Tanpa waktu aktif</p>	<p>aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.</p>
<p>Objek</p> <div style="border: 1px solid black; height: 30px; width: 200px; margin: 5px 0;"></div> <p><u>Nama Objek : nama kelas</u></p>	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> <div style="background-color: black; width: 30px; height: 60px; margin: 5px 0;"></div>	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semuanya yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukandi dalamnya</p>
<p>Pesan tipe <i>create</i></p> <div style="text-align: center; margin: 10px 0;">  </div> <p style="text-align: center;"><< create >></p>	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan Tipe <i>call</i></p> <div style="text-align: center; margin: 10px 0;">  </div> <p>1: nama_metode()</p>	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>
<p>Pesan tipe <i>send</i></p> <div style="text-align: center; margin: 10px 0;">  </div> <p>1: masukan</p>	<p>Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> <div style="text-align: center; margin: 10px 0;">  </div> <p>1: keluaran</p>	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>

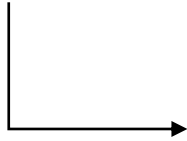
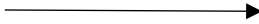

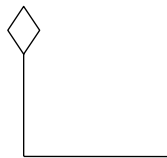
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i></p>
---	---

c. *Class Diagram*

Class diagram merupakan spesifikasi yang akan menghasilkan objek dan merupakan inti dari perkembangan dan desain yang berorientasi objek yang bersifat statis[24].

Tabel 2. 7 *Class Diagram*

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi/<i>association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>

Assosiasi berarah/directed association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan/dependensi 	Relasi antar kelas dengan makna kebergantungan antar kelas
Aggregation 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

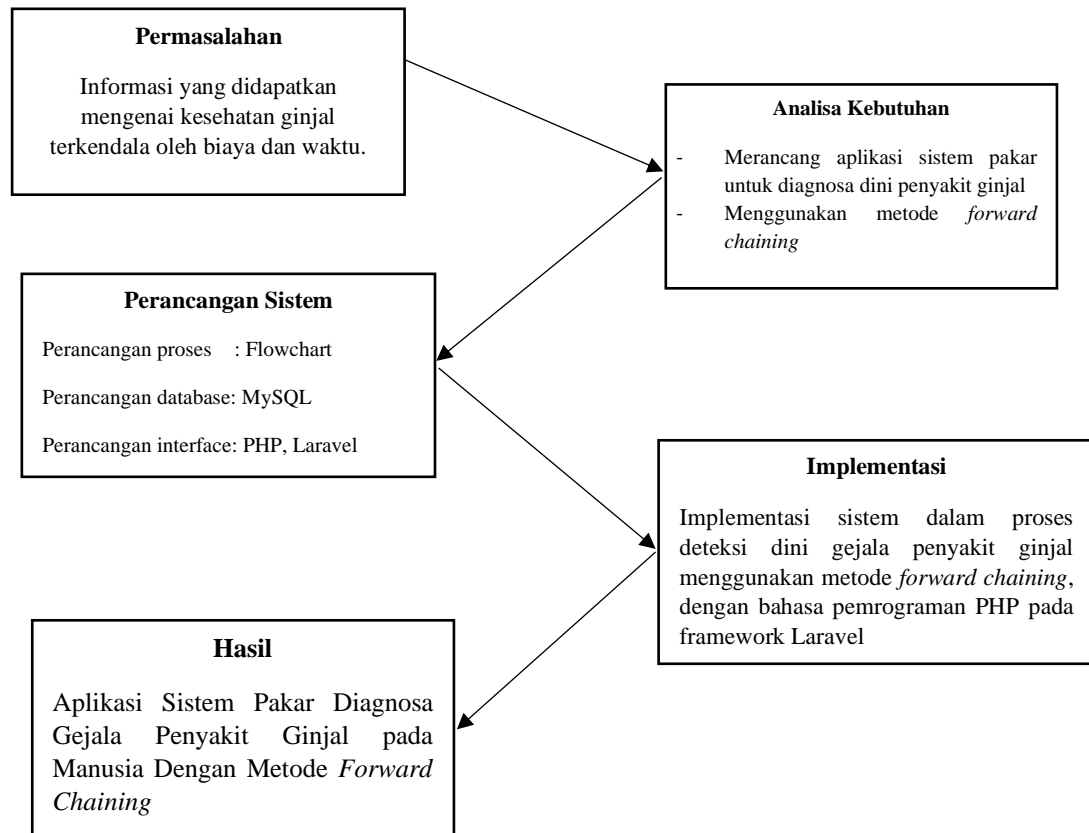
2.2.12 Alur Sistem

Sistem yang akan dibangun melibatkan 2 pihak, yaitu pakar/dokter sebagai admin, dan masyarakat umum (*user*). Admin (pakar) memiliki peran dalam melihat dan mengelola data-data mengenai gejala dan jenis penyakit seperti mengubah, menambah dan menghapus. Kemudian *user* merupakan orang yang dapat mengakses *website*, melihat informasi dan berkonsultasi terkait gejala yang sedang diderita.

Jadi, alur pada sistem ini dimulai dari Admin yang menginput data-data mengenai gejala dan jenis penyakit ginjal, dan solusi, setelah data selesai diinput, *user* dapat melakukan konsultasi dengan mengisi data diri pada saat registrasi kemudian melakukan login agar dapat mengakses halaman konsultasi untuk menjawab pertanyaan-pertanyaan terkait gejala penyakit ginjal yang diberikan oleh sistem.

2.3 Kerangka Pemikiran

Adapun kerangka pemikiran yang digunakan sebagai referensi dalam mengambil langkah-langkah yang akan digunakan pada penelitian ini adalah:

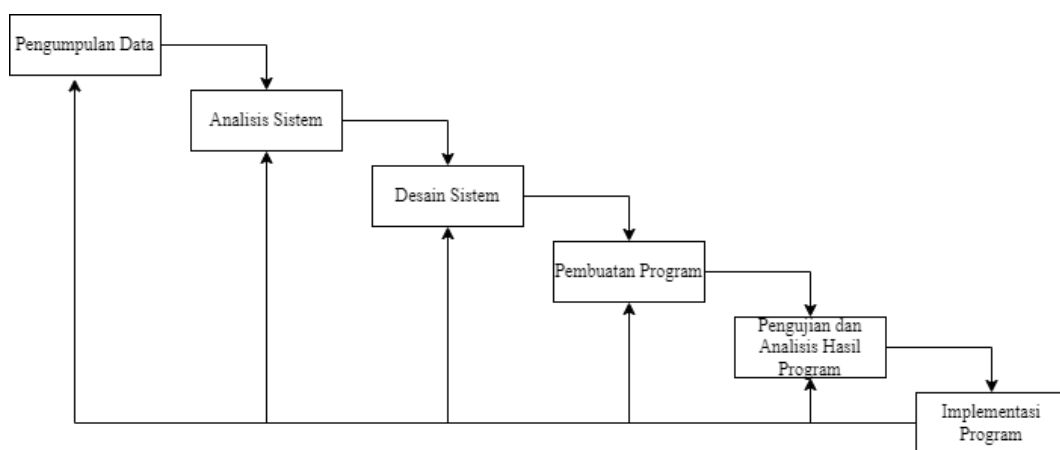


BAB III

METODOLOGI PENELITIAN

3.1 Metode Pengembangan Sistem

Penelitian yang akan diteliti menggunakan *System Development Life Cycle* (SDLC) dengan model *waterfall* atau disebut metode air terjun. Metode *waterfall* merupakan model pengembangan sistem yang dimulai dari tahap analisis, kemudian dilanjutkan ke tahap desain, implementasi, pengujian, dan pemeliharaan[25]. Pengembangan sistem model *waterfall* seperti gambar berikut:



Gambar 3. 1 Metode *Waterfall*

Pada gambar 3.1 diatas adalah tahapan-tahapan dalam metode *waterfall*. Tahap awal yang dilakukan ialah pengumpulan serta pengambilan data yang diperlukan di Rumah Sakit Umum Daerah Ulin Banjarmasin. Kemudian melakukan analisis sistem pada penelitian yang akan dilakukan. Selanjutnya membuat program berupa coding yang sesuai dengan rancangan desain sistem yang akan dibuat. Kemudian dilakukan pengujian dan analisis hasil program, lalu tahap terakhir yaitu melakukan implementasi program.

3.1.1 Perencanaan Kebutuhan

Adapun perencanaan kebutuhan yang digunakan dengan beberapa tahap dalam pengumpulan data. Tahap pengumpulan data dilakukan dengan mencari referensi pengetahuan dari beberapa buku dan jurnal yang berkaitan dengan sistem

pakar, metode *forward chaining*, serta data-data penyakit gagal ginjal kronik pada manusia. Pengumpulan data penyakit ginjal pada manusia ini dilakukan dalam kurun satu bulan untuk proses permohonan izin melakukan studi pendahuluan dan studi penelitian, dengan jumlah keseluruhan data yang diberikan adalah 154 pasien yang terdiagnosa penyakit gagal ginjal kronik dan dinyatakan sembuh dalam dengan beberapa tindakan yang diberikan, data yang didapatkan di RSUD Ulin Banjarmasin berbentuk file PDF. Data tersebut terlampir pada lampiran ke 1. Dalam penelitian ini tahap pengumpulan data dilakukan dengan beberapa cara antara lain, yaitu:

a. Observasi

Observasi dilakukan dengan langsung datang ke salah satu Rumah Sakit Umum Daerah yaitu RSUD Ulin Banjarmasin, untuk menanyakan proses permohonan izin permintaan data yang diperlukan dalam proses penelitian ini. Setelah mengikuti proses perizinan, tahap observasi yang dilakukan ialah dengan menemui kepala bagian diklit, kepala bagian riset, kepala bagian humas dan kepala bagian rekam medik RSUD Ulin Banjarmasin untuk proses akhir pengambilan data pasien yang menderita penyakit gagal ginjal kronik. Pada tahap pengambilan data ini penulis menghabiskan waktu kurang lebih selama 6 bulan terhitung dengan hari libur pada instansi kesehatan yang bersangkutan. Data yang didapatkan terlampir pada lampiran ke 2.

b. Studi Pustaka

Pada tahap ini dilakukan dengan mempelajari teori-teori atau literatur dari buku, jurnal, dan referensi yang berhubungan dengan penelitian ini untuk melengkapi data yang telah didapatkan berdasarkan hasil wawancara dan pengumpulan data mengenai gejala-gejala dan penyakit.

c. Wawancara

Wawancara dilakukan dengan tujuan melengkapi data yang telah didapatkan berdasarkan hasil observasi dan studi pustaka dari beberapa jurnal dan data yang didapatkan dari hasil rekam medik, wawancara dilakukan dengan menemui seorang dokter spesialis penyakit dalam (Ginjal Hipertensi) yaitu dr. Dewi Rizki Agustina,. Sp.PD yang melakukan praktek pada apotek Perintis

Kayutangi yang berlokasi di Jalan Hasan Basri No. 123, Alalak Utara, Kec. Banjarmasin Utara, Kota Banjarmasin, Kalimantan selatan. Hasil wawancara dilampirkan pada lampiran ke 3.

3.1.2 Usulan Pemecahan Masalah

Berdasarkan data dan hasil observasi yang dilakukan, maka perlunya dibuat sebuah Sistem Pakar Diagnosa Penyakit Ginjal Pada Manusia dengan Metode *Forward Chaining* agar dapat memudahkan proses pemeriksaan pasien yang kemungkinan menderita penyakit ginjal. Dengan dilakukannya pencocokan fakta-fakta gejala yang ada dengan bagian IF dari aturan IF-THEN, dan proses pencocokan akan berhenti jika tidak ada lagi rule yang dapat dieksekusi seperti pada pencocokan rule di bawah ini.

R1 = IF G01, G03, G04, G14, G15, G16, G07, G21 THEN P1

R2 = IF G01, G04, G09, G31 THEN P2

R3 = IF G01, G07, G08, G32, G26, G33 THEN P3

R4 = IF G05, G04, G06, G14, G09, G22, G31 THEN P4

- Gejala yang terdeteksi: G11, G14, G23, G05, G29, 32

- Hitung nilai persentasi peluang suatu kejadian!

Rumus: $P(A) = \frac{\text{Jumlah gejala}}{\text{Jumlah total gejala}} * 100$

- Rule yang memiliki gejala tereteksi:

R1 = IF G01, G03, G04, G14, G15, G16, G07, G21, G32, THEN P1

$$P(A) = \frac{2}{11} * 100 = 0,18$$

R3 = IF G01, G07, G08, G32, G26, G33 THEN P3

$$P(A) = \frac{1}{6} * 100 = 0,16$$

R4 = IF G05, G04, G06, G14, G09, G22, G31 THEN P4

$$P(A) = \frac{2}{7} * 100 = 0,28$$

R6 = IF G01, G23, G36 THEN P6

$$P(A) = \frac{1}{3} * 100 = 0,33$$

Detail perhitungan *forward chaining* terdapat pada lampiran 5.

3.1.3 Desain Sistem

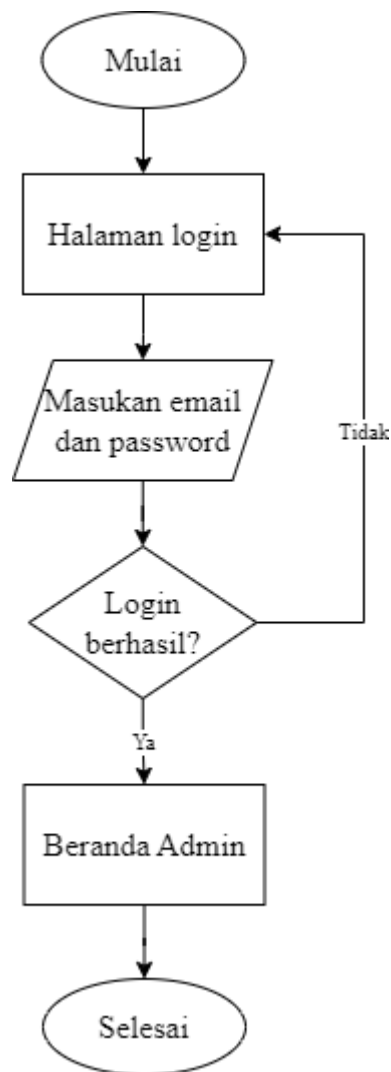
Berikut merupakan desain sistem yang akan digunakan dalam proses penggunaan sistem pakar diagnosa penyakit ginjal dengan menggunakan metode *forward chaining*. Pada sistem ini terdapat beberapa desain sistem antara lain yaitu *flowchart*, *usecase diagram*, dan *class diagram*.

a. Rancangan *Flowchart*

Berikut merupakan gambar-gambar rancangan *flowchart* yang digunakan oleh admin dan user pada sistem pakar diagnosa penyakit ginjal menggunakan metode *forward chaining*. *Flowchart* yang dibuat yaitu *flowchart* login admin, *flowchart* data pasien, *flowchart* data gejala, *flowchart* data penyakit, *flowchart* data solusi, *flowchart* aturan, *flowchart* konsultasi, *flowchart* riwayat diagnosa, serta *flowchart* tambah, ubah, dan hapus data.

1) *Flowchart* Login oleh admin

Flowchart login ini merupakan kegiatan pertama yang dilakukan admin sebelum dapat mengakses website, login ini hanya dapat dilakukan oleh admin.

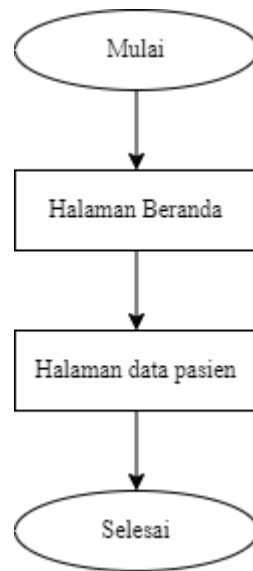


Gambar 3. 2 Flowchart Login

Proses login dilakukan dengan memasukkan email dan password terlebih dahulu pada *form* login yang telah tersedia. Jika login gagal, maka akan kembali ke halaman login. Dan jika login berhasil maka admin akan masuk ke halaman beranda admin.

2) *Flowchart* Data Pasien

Pada *flowchart* data pasien dapat dijabarkan bahwa kegiatan ini hanya dapat diakses oleh admin setelah berhasil melakukan login.

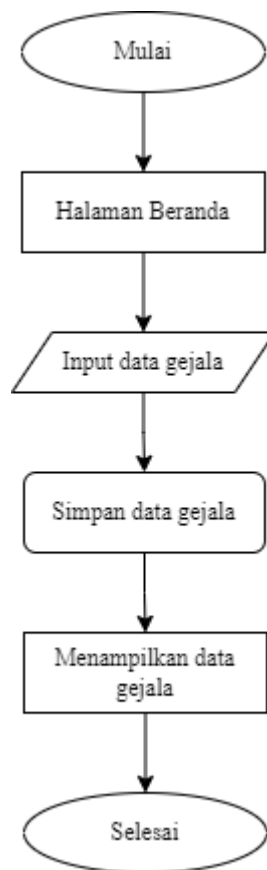


Gambar 3. 3 Flowchart Data Pasien

Pada *flowchart* data pasien ini admin hanya dapat melihat daftar pasien yang telah melakukan registrasi pada sistem yang kemudian akan disimpan ke dalam *database*. Data yang ditampilkan pada halaman data pasien berupa no.pasien, nama pasien, jenis kelamin, dan umur pasien.

3) *Flowchart* Input Data Gejala

Pada *flowchart* input data gejala ini merupakan kegiatan yang hanya dapat dilakukan oleh admin, kegiatan ini dilakukan setelah admin berhasil melakukan login pada sistem.

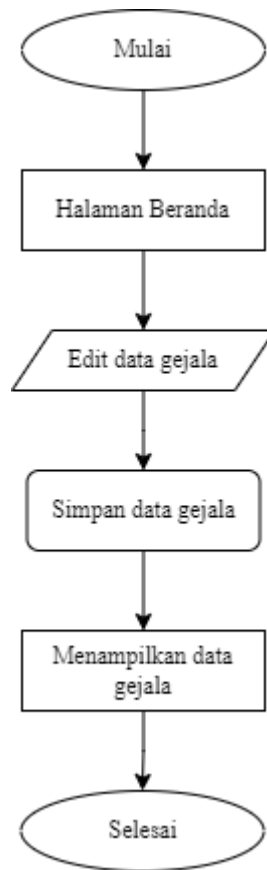


Gambar 3. 4 Flowchart Input Data Gejala

Pada *flowchart* input data gejala ini admin dapat menambahkan data gejala yang diperlukan pada sistem berupa kode gejala dan kondisi gejala akan disimpan ke dalam *database*, kemudian akan ditampilkan pada halaman data gejala berdasarkan data terbaru.

4) *Flowchart* Edit Data Gejala

Edit data gejala ini hanya dapat dilakukan oleh admin, pada *flowchart* edit data gejala ini admin dapat mengubah atau menambah data gejala pada sistem jika perlu dilakukannya pembaruan daftar gejala pada sistem.

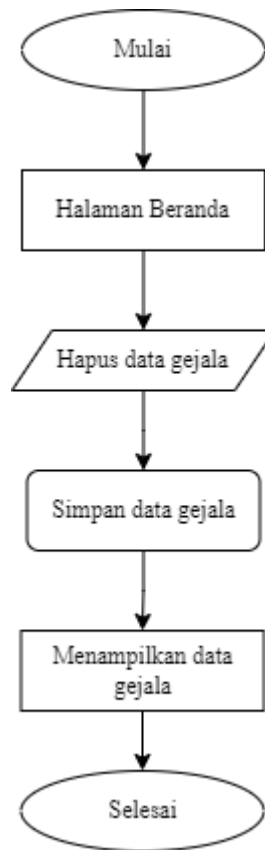


Gambar 3. 5 Flowchart Edit Data Gejala

Pada *flowchart* edit data gejala ini admin dapat mengubah data yang diperlukan pada sistem yang kemudian akan diperbarui dan tersimpan ke dalam *database*, selanjutnya data tersebut akan ditampilkan pada halaman data gejala berdasarkan data terbaru. Data yang dapat diubah pada *flowchart* edit data gejala adalah kode gejala, dan kondisi gejala.

5) *Flowchart* Hapus Data Gejala

Hapus data gejala ini hanya dapat dilakukan oleh admin, pada *flowchart* hapus data gejala ini admin dapat menghapus data gejala pada sistem jika perlu dihilangkan pada daftar gejala pada sistem.

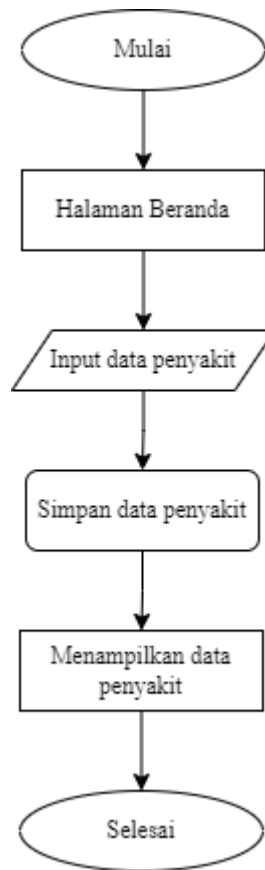


Gambar 3. 6 Flowchart Hapus Data Gejala

Pada *flowchart* hapus data gejala, admin dapat menghapus data gejala pada sistem yang mana data pada *database* pun akan ikut terhapus dan diperbarui yang kemudian akan ditampilkan sesuai dengan data yang ada pada sistem.

6) Flowchart Input Data Penyakit

Pada *flowchart* input data penyakit ini merupakan kegiatan yang hanya dapat dilakukan oleh admin, kegiatan ini dilakukan setelah admin berhasil melakukan login pada sistem.

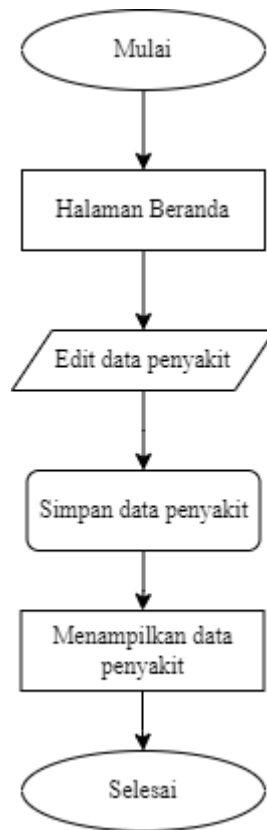


Gambar 3. 7 Flowchart Input Data Penyakit

Pada *flowchart* input data penyakit ini admin dapat menambahkan data penyakit yang diperlukan pada sistem berupa kode penyakit, nama penyakit, dan deskripsi penyakit akan disimpan ke dalam *database* dan kemudian akan ditampilkan pada halaman data penyakit berdasarkan data terbaru.

7) Flowchart Edit Data Penyakit

Edit data penyakit ini hanya dapat dilakukan oleh admin, pada *flowchart* edit data penyakit ini admin dapat mengubah atau menambah data penyakit pada sistem jika perlu dilakukannya pembaruan daftar penyakit pada sistem.

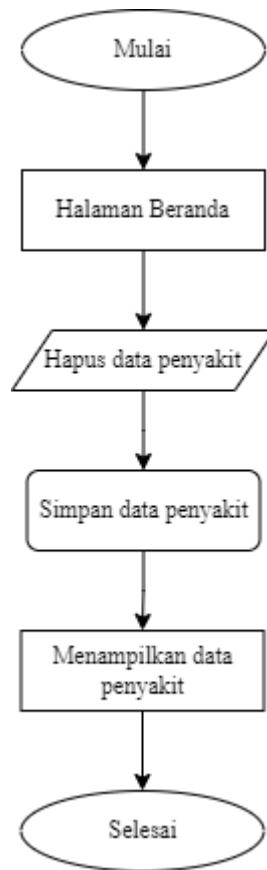


Gambar 3. 8 Flowchart Edit Data Penyakit

Pada *flowchart* edit data penyakit ini admin dapat mengubah data yang diperlukan pada sistem yang kemudian akan diperbarui dan tersimpan ke dalam *database*, selanjutnya data tersebut akan ditampilkan pada halaman data penyakit berdasarkan data terbaru. Data yang dapat diubah pada *flowchart* edit data penyakit adalah kode penyakit, nama penyakit, dan deskripsi dari penyakit tersebut.

8) Flowchart Hapus Data Penyakit

Hapus data penyakit ini hanya dapat dilakukan oleh admin, pada *flowchart* hapus data penyakit ini admin dapat menghapus data penyakit pada sistem jika perlu dihilangkan pada daftar penyakit pada sistem.

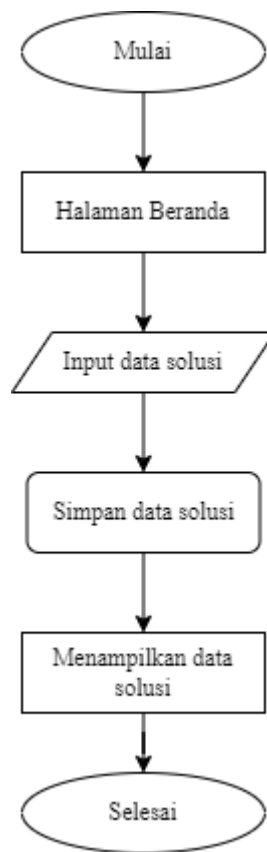


Gambar 3. 9 Flowchart Hapus Data Penyakit

Pada *flowchart* hapus data penyakit, admin dapat menghapus data penyakit pada sistem yang mana data pada *database* pun akan ikut terhapus dan diperbarui yang kemudian akan ditampilkan sesuai dengan data yang ada pada sistem.

9) Flowchart Input Data Solusi

Pada *flowchart* input data solusi ini merupakan kegiatan yang hanya dapat dilakukan oleh admin, kegiatan ini dilakukan setelah admin berhasil melakukan login pada sistem.

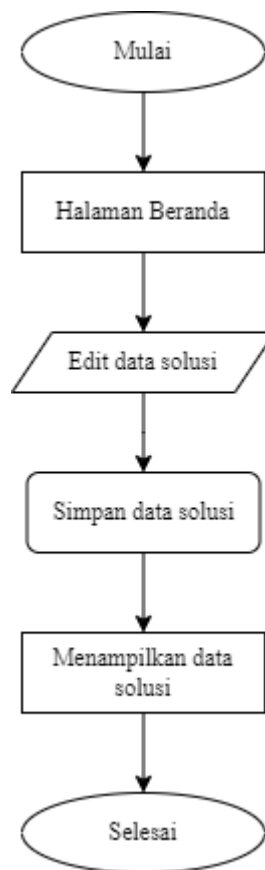


Gambar 3. 10 Flowchart Input Data Solusi

Pada *flowchart* input data solusi ini admin dapat menambahkan data penyakit yang diperlukan pada sistem berupa kode solusi, solusi sebagai saran, dan deskripsi dari solusi tersebut akan disimpan ke dalam *database*, kemudian akan ditampilkan pada halaman data solusi berdasarkan data terbaru.

10) Flowchart Edit Data Solusi

Edit data solusi ini hanya dapat dilakukan oleh admin, pada *flowchart* edit data solusi ini admin dapat mengubah atau menambah data solusi pada sistem jika perlu dilakukannya pembaruan daftar solusi pada sistem.

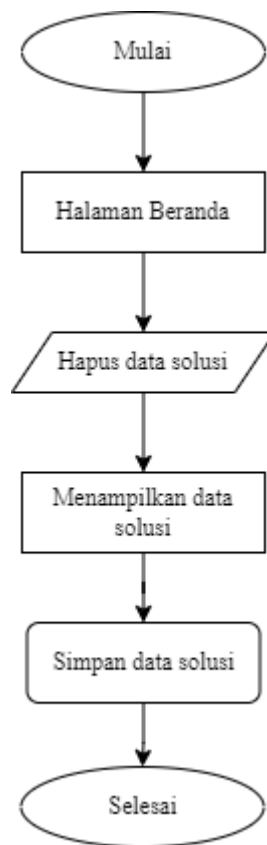


Gambar 3. 11 Flowchart Edit Data Solusi

Pada *flowchart* edit data solusi ini admin dapat mengubah data yang diperlukan pada sistem yang kemudian akan diperbarui dan tersimpan ke dalam *database*, selanjutnya data tersebut akan ditampilkan pada halaman data solusi berdasarkan data terbaru. Data yang dapat diubah pada *flowchart* edit data solusi adalah kode solusi, nama solusi, dan deskripsi sebagai penjabaran dari solusi tersebut.

11) Flowchart Hapus Data Solusi

Hapus data solusi ini hanya dapat dilakukan oleh admin, pada *flowchart* hapus data solusi ini admin dapat menghapus data solusi pada sistem jika perlu dihilangkan pada daftar solusi pada sistem.

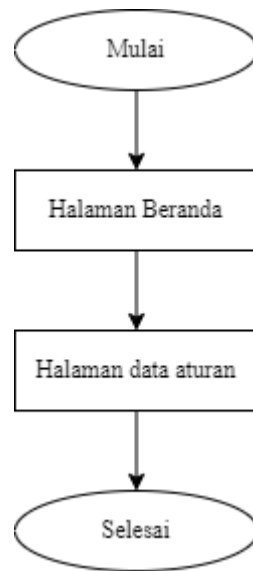


Gambar 3. 12 Flowchart Hapus Data Solusi

Pada *flowchart* hapus data solusi, admin dapat menghapus data solusi pada sistem yang mana data pada *database* pun akan ikut terhapus dan diperbarui yang kemudian akan ditampilkan sesuai dengan data yang ada pada sistem.

12) Flowchart Data Aturan

Flowchart data aturan di bawah ini merupakan alur pada sistem yang dapat dilakukan oleh admin untuk melihat data aturan dalam menghasilkan diagnosa penyakit pasien berdasarkan gejala yang ada pada sistem.

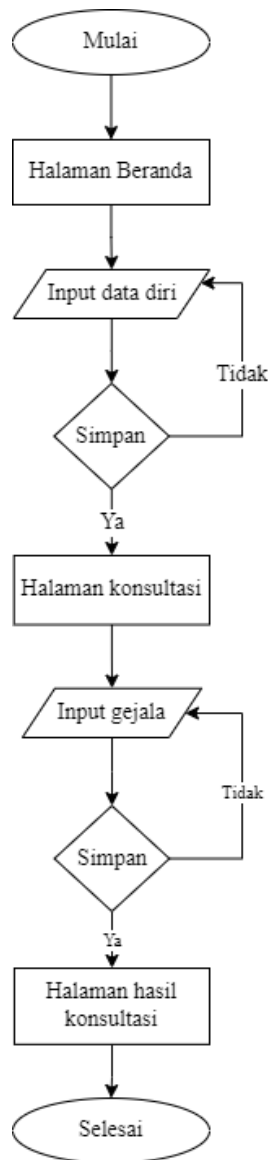


Gambar 3. 13 Flowchart Data Aturan

Pada *flowchart* data aturan di atas admin dapat melihat aturan berupa alur yang digunakan untuk menentukan diagnosa penyakit pada pasien berdasarkan gejala yang dirasakan oleh pasien.

13) Flowchart Konsultasi Pasien Belum Terdaftar

Gambar di bawah ini merupakan *flowchart* halaman konsultasi bagi pasien yang belum terdaftar pada sistem.

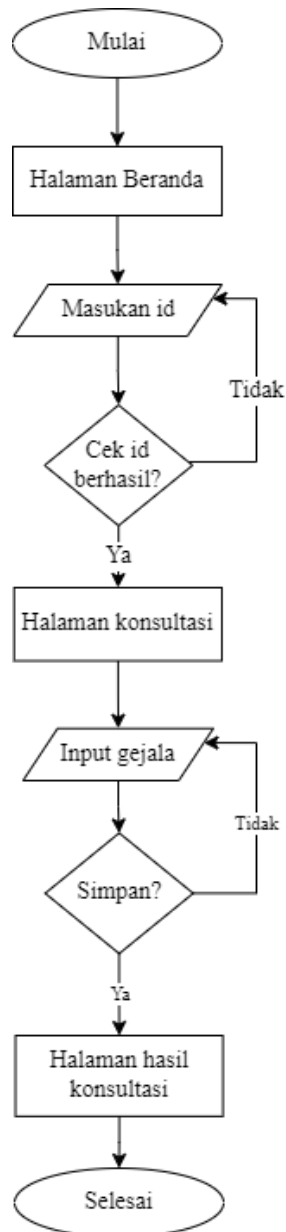


Gambar 3. 14 Flowchart Konsultasi Pasien Belum Terdaftar

Pada gambar *flowchart* di atas dapat dijabarkan bahwa pasien yang belum terdaftar pada sistem harus memasukkan data diri berupa nama lengkap, umur, dan jenis kelamin sebelum dapat melakukan konsultasi. kemudian setelah data diri pasien tersimpan maka sistem akan mengarahkan pasien menuju halaman konsultasi dan melakukan input gejala berdasarkan gejala yang sedang dialami oleh pasien, setelah selesai maka sistem akan membawa pasien ke halaman hasil konsultasi untuk melihat diagnosa penyakit berdasarkan gejala yang sedang dialami oleh pasien.

14) Flowchart Konsultasi Pasien Terdaftar

Gambar di bawah ini merupakan *flowchart* halaman konsultasi bagi pasien terdaftar pada sistem.



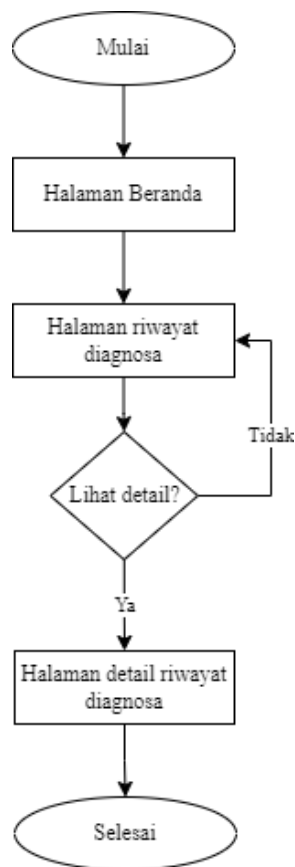
Gambar 3. 15 flowchart konsultasi pasien terdaftar

Gambar di atas merupakan rancangan *flowchart* halaman konsultasi pada pasien yang telah terdaftar atau telah memiliki akses untuk masuk ke sistem. Sebelum dapat mengakses halaman website maka pasien terlebih dahulu harus memasukan

id yang telah didapatkan saat mengisi data diri diawal mengakses website sebelum dapat melakukan konsultasi gejala penyakit pada sistem. Setelah berhasil mengisi id maka pasien akan diarahkan menuju halaman konsultasi untuk melakukan konsultasi berdasarkan gejala yang sedang dialami, setelah selesai maka sistem akan mengarahkan pasien ke halaman hasil konsultasi dan menampilkan kemungkinan diagnosa penyakit berdasarkan gejala yang dialami dan solusi yang dapat dilakukan guna penanganan pertama pada penyakit terdeteksi.

15) Flowchart Melihat Riwayat Diagnosa

Gambar di bawah ini merupakan halaman riwayat diagnosa pada sistem yang dapat diakses oleh pasien setelah selesai melakukan konsultasi.



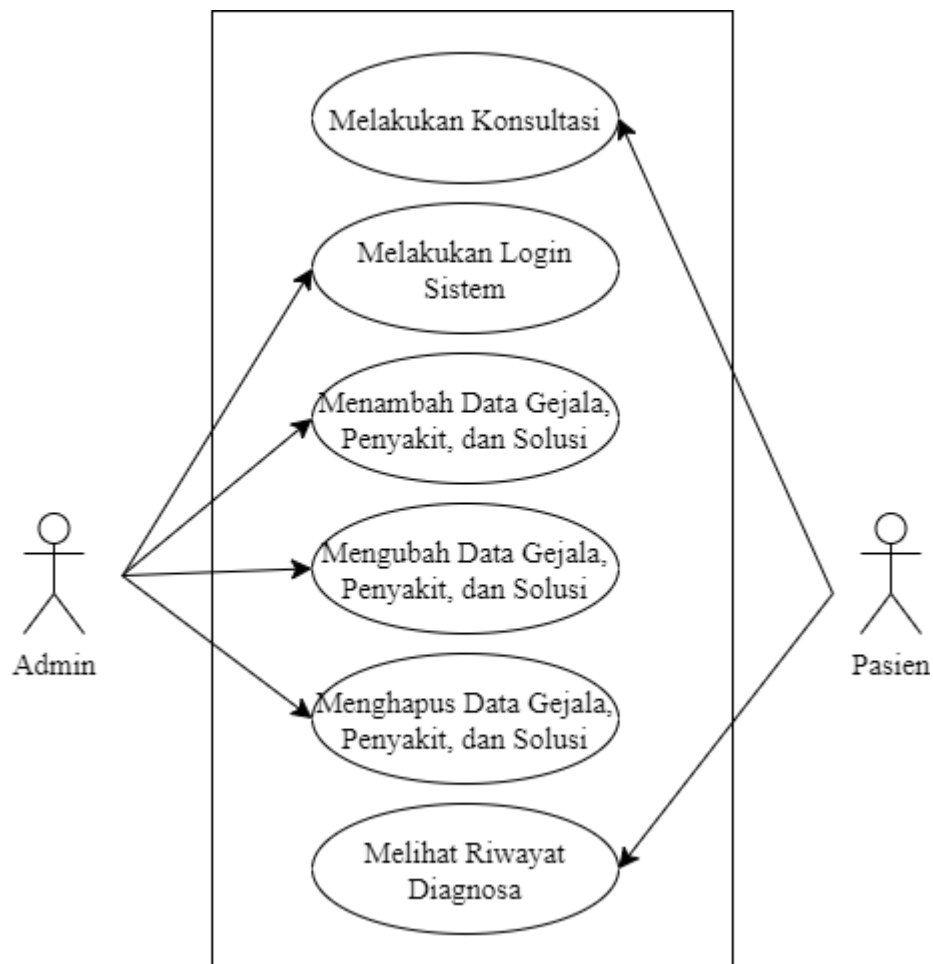
Gambar 3. 16 Flowchart Riwayat Diagnosa

Pada gambar rancangan *flowchart* di atas dapat dijabarkan bahwa pasien dapat melihat riwayat diagnosa berdasarkan hasil konsultasi yang telah dilakukan pasien pada sistem, yang mana pada halaman ini akan menampilkan hasil diagnosa

penyakit dan solusi berdasarkan gejala yang dialami oleh pasien. Selain itu, pasien juga dapat melihat detail riwayat diagnosa yang berisikan daftar dilakukannya konsultasi dan hasil diagnosa yang didapatkan antara lain gejala yang dialami pasien, penyakit yang terdeteksi berdasarkan gejala yang dialami, dan solusi berupa saran dan deskripsi penanganan pertama pada penyakit terdeteksi.

b. Rancangan *Use Case Diagram*

Use case diagram pada sistem terdiri dari 2 aktor, yaitu admin dan pengguna.



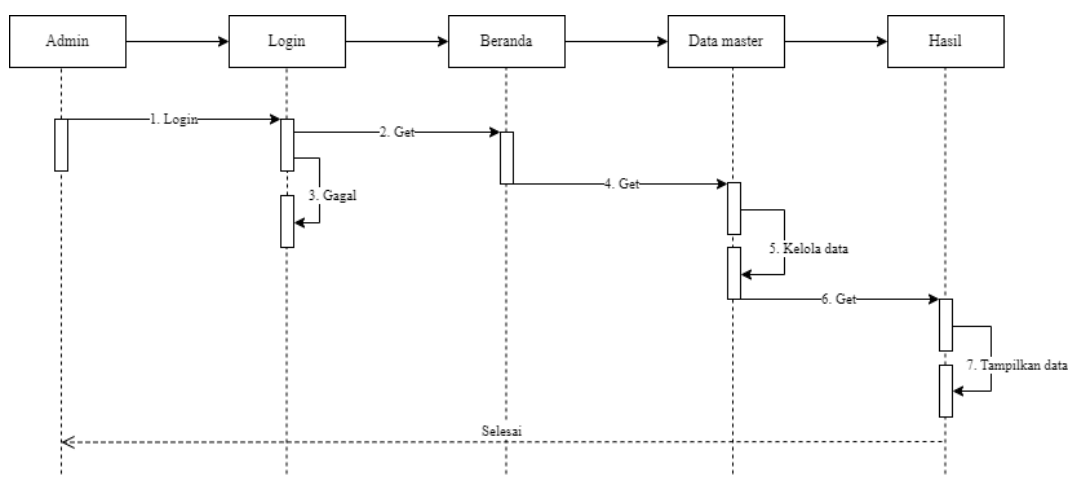
Gambar 3. 17 *Use Case Diagram*

Gambar 3.7 di atas merupakan *use case* pada sistem pakar diagnosa penyakit ginjal yang terdiri dari dua aktor, yaitu pakar admin dan pasien. Pada *use*

case, sebelum mengakses halaman website admin harus melakukan login terlebih dahulu. Setelah itu admin dapat melihat data pasien, dan mengelola ilmu kesehatan (tambah, edit, dan hapus) data gejala, data penyakit dan solusi. Berbeda dengan admin, pasien dapat langsung mengakses halaman website tanpa harus melakukan login terlebih dahulu, kemudian pasien yang telah terdaftar dapat melakukan kegiatan konsultasi dengan memasukkan id terlebih dahulu agar setelah itu dapat melakukan konsultasi berdasarkan gejala yang sedang dialami, sedangkan pasien yang belum terdaftar dapat mengisi data diri terlebih dahulu sebelum dapat melakukan konsultasi agar mendapatkan id dan dapat melihat daftar riwayat diagnosa pada sistem setelah melakukan konsultasi. Selanjutnya, setelah selesai pasien dapat melihat hasil konsultasi secara detail pada halaman riwayat diagnosa.

a. Rancangan Sequence Diagram

Adapun rancangan *sequence diagram* admin pada sistem ini sebagai berikut:

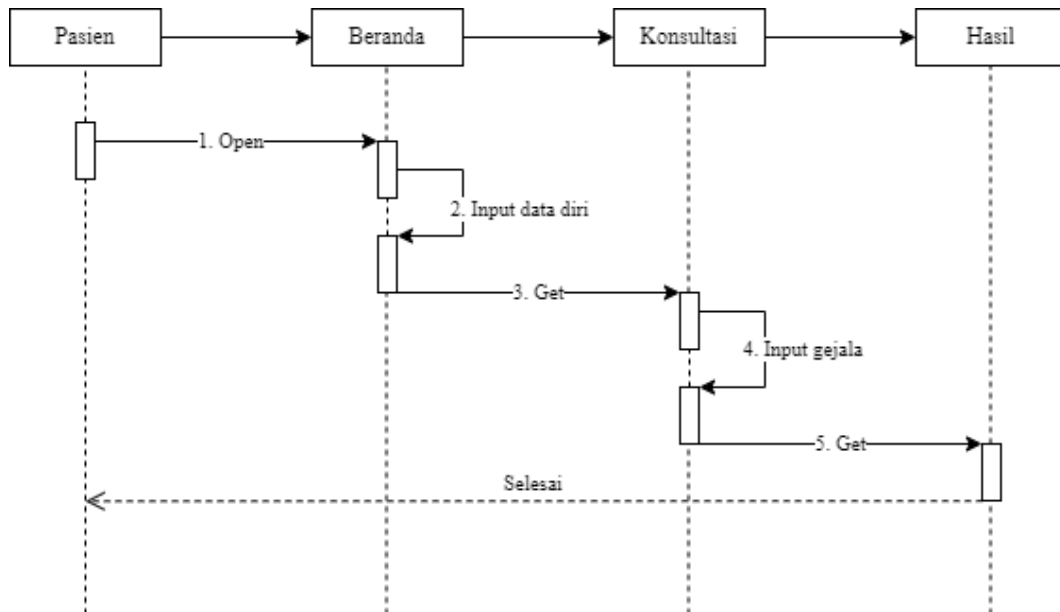


Gambar 3. 18 *Sequence Diagram* Admin

Gambar 3.18 di atas merupakan rancangan *sequence diagram* admin pada sistem pakar diagnosa penyakit ginjal dengan metode *forward chaining*. Pada *sequence diagram* admin dapat dijabar bahwa admin dapat mengakses website setelah melakukan login terlebih dahulu, jika login gagal maka sistem akan mengarahkan untuk tetap berada di halaman login, dan apabila login berhasil maka admin dapat mengakses halaman beranda yang mana pada halaman ini admin dapat mengakses

serta mengelola menu data master dan kemudian akan ditampilkan berupa hasil data yang telah dikelola oleh admin.

Adapun rancangan *sequence diagram* pengguna pada sistem ini sebagai berikut:

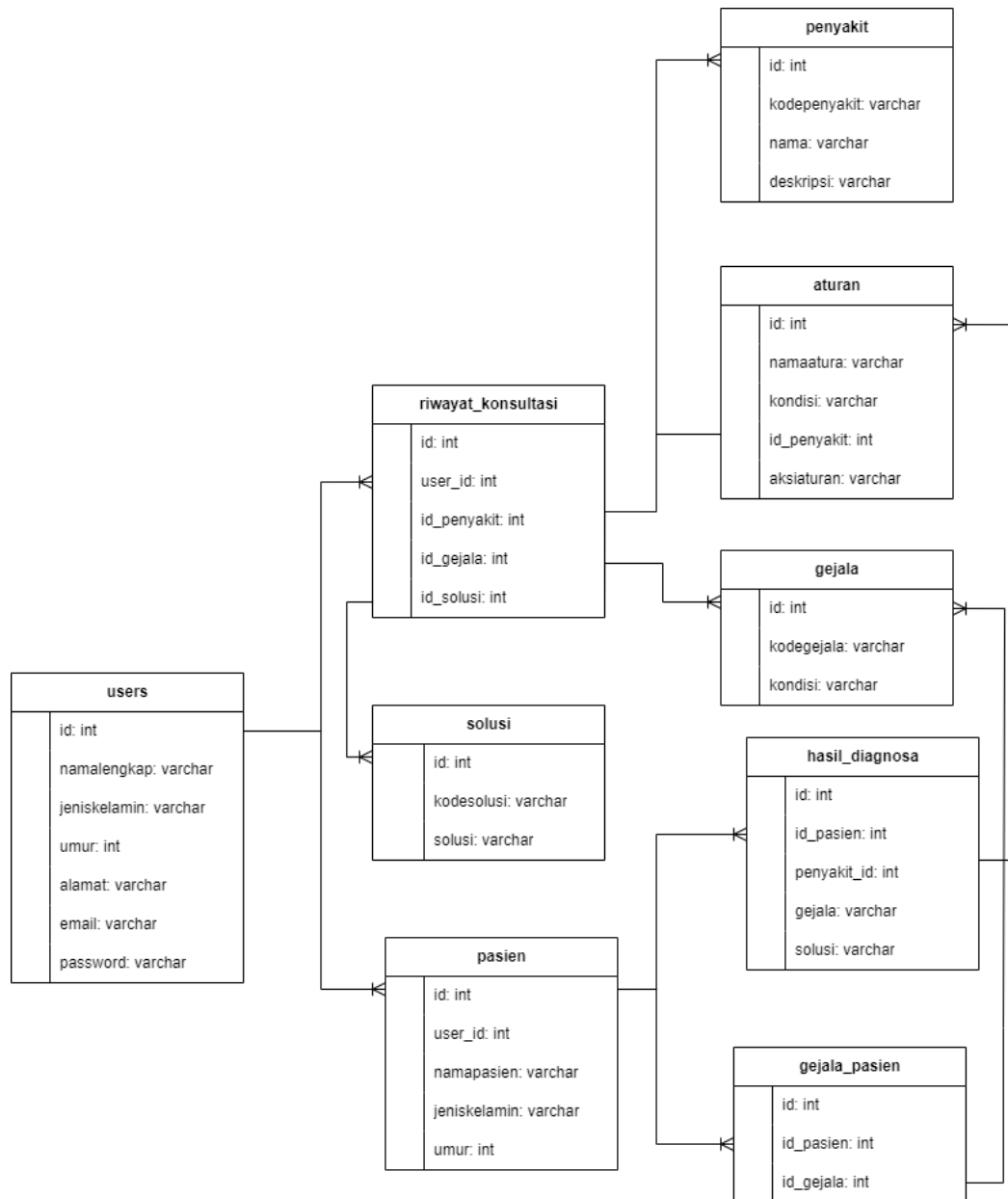


Gambar 3. 19 *Sequence Diagram* Pengguna

Gambar 3.19 di atas merupakan rancangan *sequence diagram* pasien pada sistem pakar diagnosa penyakit ginjal dengan metode *forward chaining*. Pada *sequence diagram* pasien dapat dijabar bahwa pasien dapat mengakses website tanpa harus melakukan login terlebih dahulu, jika login gagal maka sistem akan mengarahkan untuk tetap berada di halaman login, akan tetapi pasien terlebih dahulu menginputkan data diri agar kemudian dapat mengakses halaman konsultasi untuk menginputkan gejala berdasarkan keadaan pasien yang kemudian akan ditampilkan berupa hasil diagnosa berdasarkan gejala pasien.

d. Rancangan *Class Diagram*

Gambar di bawah ini merupakan *class diagram* yang digunakan pada sistem.



Gambar 3. 20 Class Diagram

e. Rancangan Tabel Basis Data

Adapun rancangan tabel basis data dalam sistem ini sebagai berikut:

1. Tabel Users

Tabel *users* merupakan tabel yang berfungsi dalam menyimpan data pengguna pada aplikasi.

Tabel 3. 1 Tabel User

No	Nama Field	Tipe Data	Keterangan
1	Id	int(10)	Primary key
2	namalengkap	varchar(255)	Nama lengkap pengguna
3	jeniskelamin	enum	Jenis kelamin pengguna
4	umur	int(11)	Umur pengguna
5	alamat	varchar(255)	Alamat pengguna
6	email	varchar(255)	Alamat email pengguna
7	password	varchar(255)	Password akses pengguna
8	level	varchar(255)	Level pengguna
9	Id_pasien	Varchar(100)	id pasien

2. Tabel Aturan

Tabel aturan digunakan untuk menyimpan data aturan pada aplikasi.

Tabel 3. 2 Tabel Aturan

No	Nama Field	Tipe Data	Keterangan
1	Id	int(10)	Primary key
2	namaaturan	varchar(255)	Nama aturan
3	kondisi	varchar(255)	Nama kondisi
4	aksiaturan	varchar(255)	Aksi aturan
5	Penyakit_id	Int(11)	Id penyakit yang berkaitan dengan aturan

3. Tabel Data Pasien

Tabel pasien digunakan untuk menyimpan data pasien pada aplikasi.

Tabel 3. 3 Tabel Pasien

No	Nama Field	Tipe Data	Keterangan
1	Id	Int(11)	Primary key
2	namapasien	varchar(100)	Nama pasien

3	jeniskelamin	varchar(15)	Jenis kelamin pasien
4	umur	int(5)	Umur pasien

4. Tabel Gejala

Tabel gejala berfungsi untuk menyimpan data gejala yang ada pada sistem.

Tabel 3. 4 Tabel Gejala

No	Nama Field	Tipe Data	Keterangan
1	Id	Int(10)	Primary key
2	kodegejala	Varchar(255)	kode gejala
3	kondisi	Varchar(255)	Kondisi gejala

5. Tabel Gejala Pasien

Tabel 3. 5 Tabel Gejala Pasien

No	Nama Field	Tipe Data	Keterangan
1	Id	Int(11)	Primary key
2	Id_pasien	Int(100)	Id pasien
3	Id_gejala	Int(11)	Id gejala

6. Tabel Penyakit

Tabel penyakit digunakan untuk menyimpan data penyakit yang dialami oleh penderita.

Tabel 3. 6 Tabel Penyakit

No	Nama Field	Tipe Data	Keterangan
1	Id	Int(11)	Primary key
2	kodepenyakit	Varchar(12)	kode penyakit
3	nama	Varchar(255)	nama penyakit
4	deskripsi	Varchar(1000)	deskripsi penyakit

7. Tabel Solusi

Tabel solusi merupakan tabel yang berfungsi menyimpan data solusi pada database sistem.

Tabel 3. 7 Tabel solusi

No	Nama Field	Tipe Data	Keterangan
1	id	Int(11)	Primary key
2	kodesolusi	Varchar(5)	Kode solusi
3	solusi	Varchar(255)	Solusi yang berkaitan dengan aturan
4	deskripsi	Varchar(1000)	Deskripsi solusi

8. Tabel Riwayat Diagnosa

Tabel ini digunakan untuk menyimpan data riwayat konsultasi yang telah dilakukan oleh pasien.

Tabel 3. 8 Tabel Riwayat Konsultasi

No	Nama Field	Tipe Data	Keterangan
1	Id	Int(11)	Primary key
2	pasien_id	Int(11)	Id dari pasien
3	Penyakit_id	Int(11)	Id konsultasi
4	gejala	Varchar(255)	Gejala yang berkaitan dengan sistem
5	solusi	Varchar(255)	Solusi yang berkaitan dengan sistem

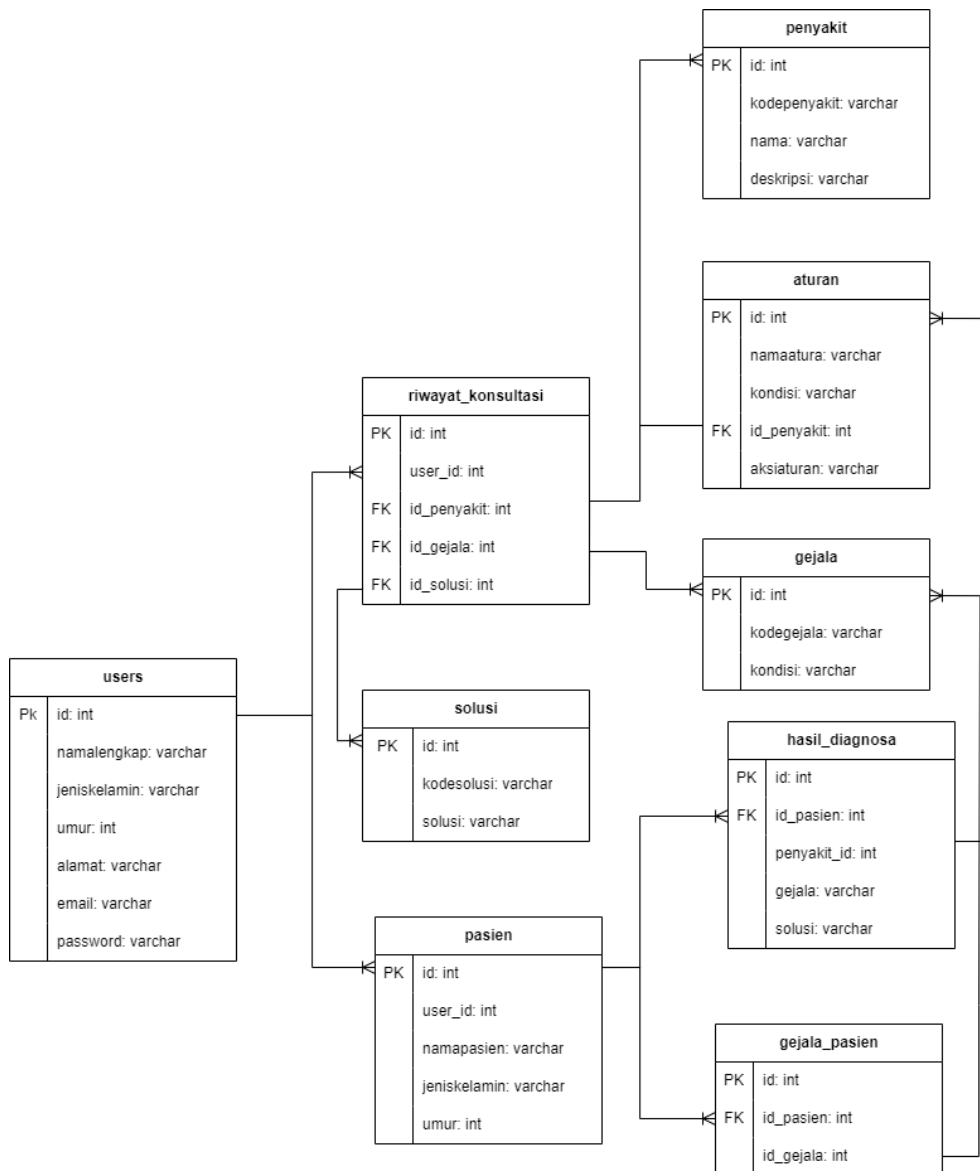
9. Tabel Hasil Diagnosa

Tabel ini digunakan untuk menyimpan data hasil diagnosa yang telah dilakukan oleh pasien.

No	Nama Field	Tipe Data	Keterangan
1	id	Int(11)	Primary key
2	Pasien_id	Int(11)	Id dari pasien
3	Penyakit_id	Int(11)	Id dari penyakit
4	gejala	Varchar(255)	Gejala yang berkaitan dengan sistem
5	solusi	Varchar(255)	Solusi yang berkaitan dengan sistem

g. Tabel Relasi

Gambar di bawah ini merupakan rancangan tabel relasi pada sistem pakar diagnosa penyakit ginjal.



Gambar 3. 21 Rancangan Tabel Relasi

h. Desain Interface

Adapun desain *interface* pada sistem pakar ini adalah sebagai berikut:

1. Tampilan Halaman Registrasi Akun

Gambar di bawah merupakan sketsa halaman registrasi akun untuk pengguna baru, halaman ini dapat digunakan oleh pasien sebelum mendapatkan akun untuk login ke aplikasi.

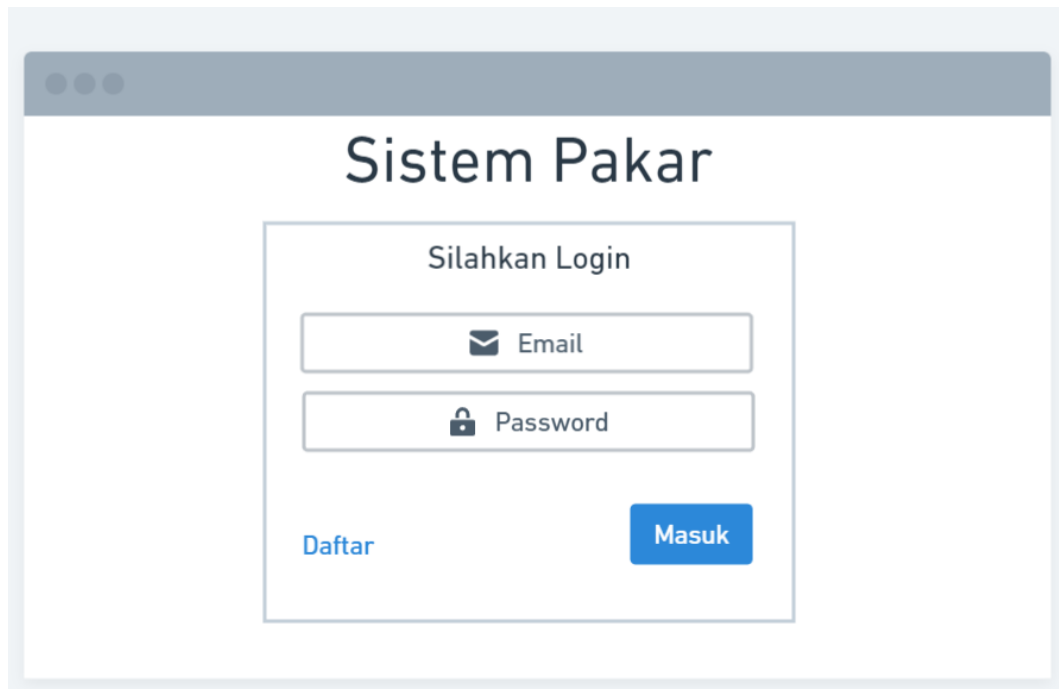
The image shows a web browser window with a registration form. The form is titled "Sistem Pakar" and "Daftar Akun". It contains six input fields: "Nama Lengkap", "Jenis Kelamin", "Umur", "Alamat", "Email", and "Password". Below the input fields, there is a link "Sudah memiliki akun" and a blue button labeled "Daftar".

Gambar 3. 22 Tampilan Halaman Registrasi Akun

Pada rancangan halaman registrasi akun di atas dapat digunakan oleh pasien sebelum mendapat email dan password untuk melakukan login pada sistem, dengan memasukan nama lengkap, jenis kelamin, umur, alamat, email, dan password.

2. Tampilan Halaman Log in

Gambar di bawah ini merupakan halaman login yang dapat digunakan oleh admin dan pengguna sebelum dapat mengakses fitur yang terdapat pada sistem.

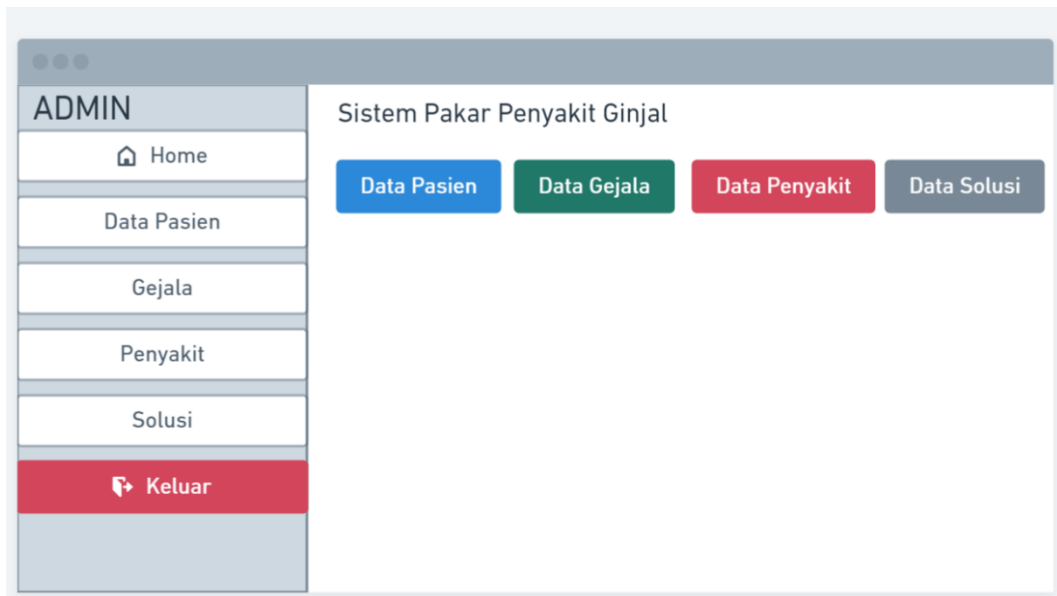


Gambar 3. 23 Tampilan Halaman Login

Pada rancangan halaman login di atas admin dan pasien dapat menggunakan sebelum mengakses website dengan memasukkan email dan password, kemudian mengklik tombol “masuk” dan sistem akan mengarahkan ke halaman beranda masing-masing pengguna.

3. Tampilan Halaman Beranda Admin

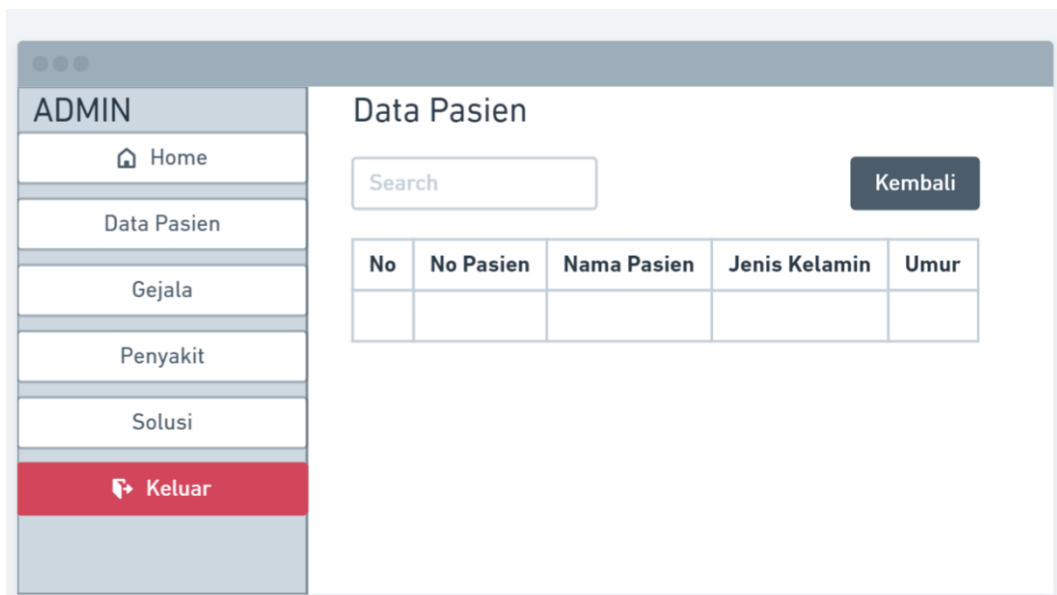
Gambar di bawah ini merupakan gambaran halaman beranda admin yang akan dibuat pada sistem, pada halaman beranda admin ini terdapat empat menu antara lain menu data user, menu data penyakit, menu data gejala, menu data solusi, dan terdapat satu button aktif keluar.



Gambar 3. 24 Tampilan halaman beranda admin

4. Tampilan Halaman Daftar Data Pasien Pada Admin

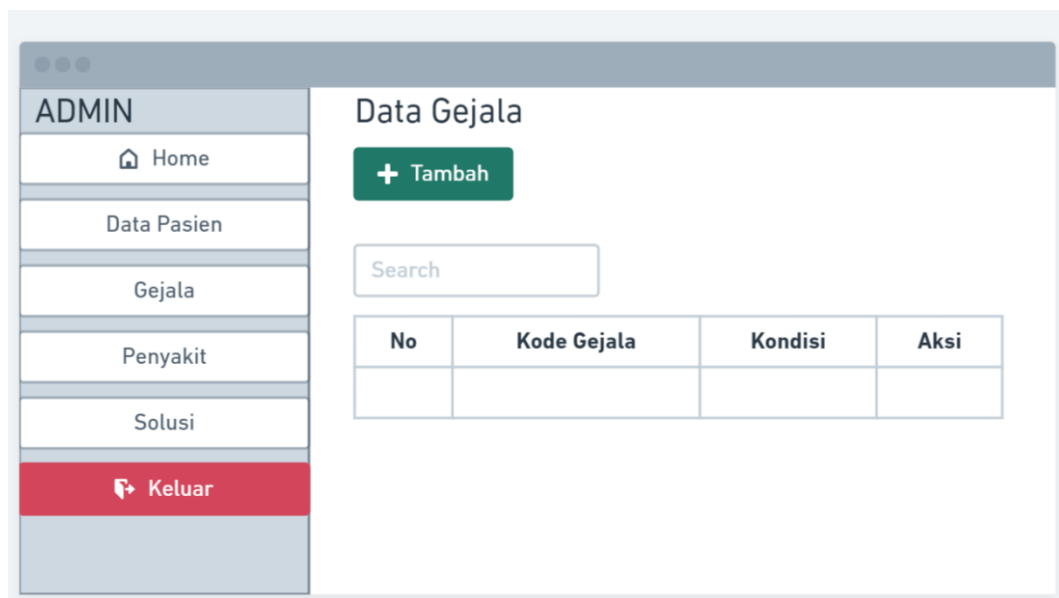
Gambar di bawah ini merupakan menu data pasien yang ada pada halaman admin, halaman ini berfungsi untuk menyimpan data pasien (user) yang ada pada database sistem yang kemudian akan ditampilkan pada halaman data pasien ini.



Gambar 3. 25 Halaman Data pasien

5. Tampilan Halaman Data Gejala Pada Admin

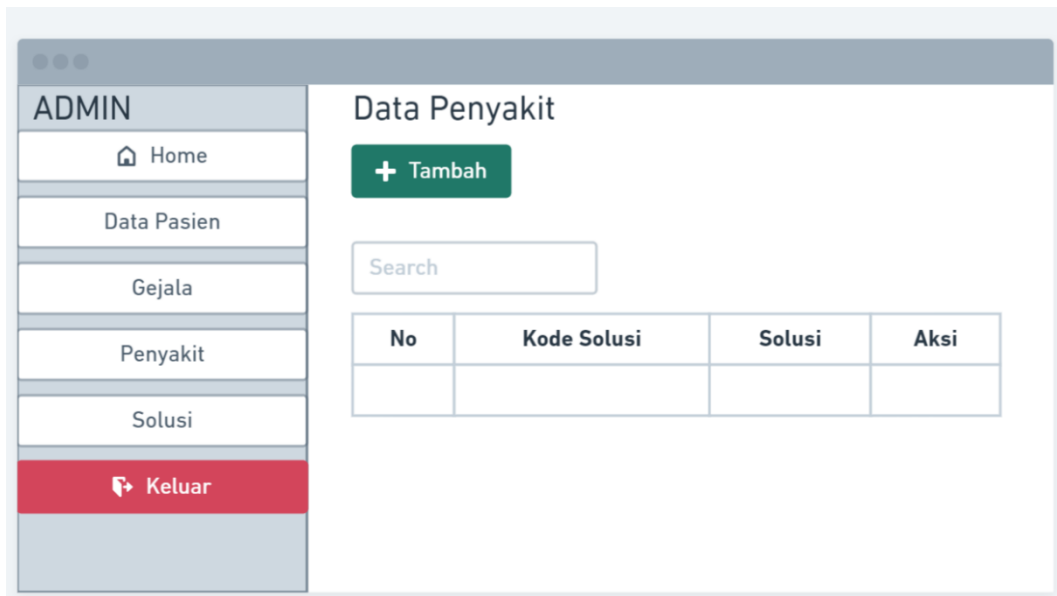
Gambar di bawah ini merupakan sketsa halaman data gejala pada admin, yang mana halaman ini berfungsi untuk admin menyimpan data gejala penyakit, selain itu admin juga dapat mengubah serta menghapus gejala yang ada pada sistem.



Gambar 3. 26 Halaman Data gejala

6. Tampilan Halaman Data Penyakit Pada Admin

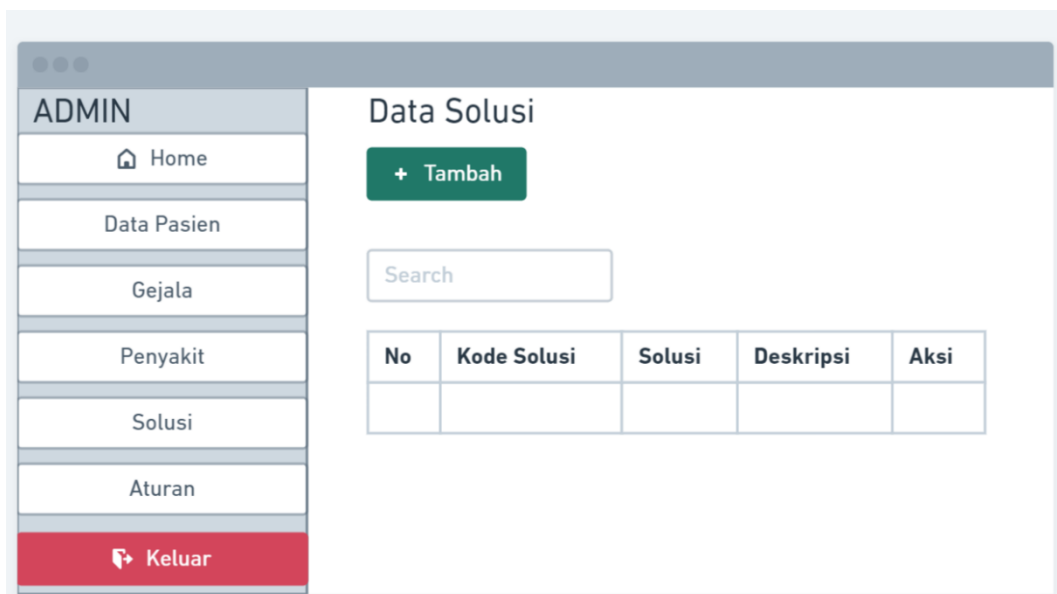
Gambar di bawah ini merupakan sketsa halaman data penyakit pada admin. Halaman data penyakit ini berfungsi untuk menyimpan data penyakit pada sisten, selain dapat menambah data penyakit admin juga dapat mengubah dan menghapus data penyakit yang ada pada sistem.



Gambar 3. 27 Halaman Penyakit

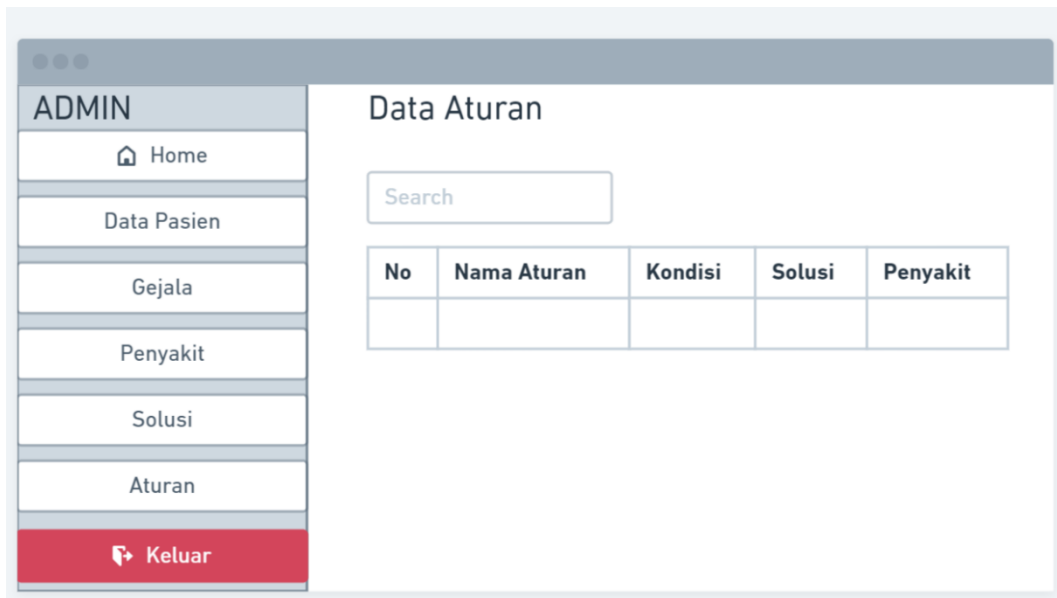
7. Tampilan Halaman Data Solusi Pada Admin

Gambar di bawah ini merupakan sketsa tampilan halaman data solusi pada admin. Halaman ini berfungsi menyimpan data solusi berupa saran atau tindakan penanganan dari penyakit yang diderita oleh pasien pada sistem, pada halaman ini admin dapat menambahkan data solusi, mengubah dan juga menghapus data pada sistem.



8. Tampilan Halaman Data Aturan Pada Admin

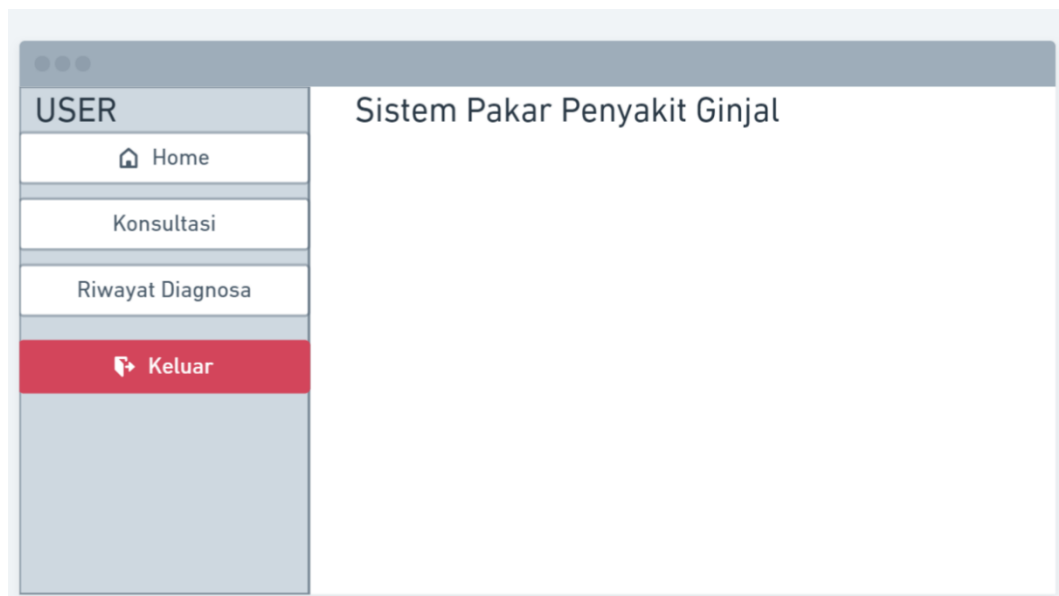
Gambar di bawah ini merupakan halaman aturan atau *rule* pada sistem. Halaman ini berfungsi memberikan aturan penentu penyakit dan solusi berdasarkan gejala yang diderita oleh pasien/pengguan.



Gambar 3. 28 Halaman Aturan

9. Tampilan Halaman Beranda User

Gambar di bawah ini merupakan sketsa awal tampilan halaman beranda pada user, pada halaman ini terdapat dua menu yaitu menu konsultasi dan menu riwayat diagnosa.



10. Tampilan Halaman Konsultasi User

Gambar di bawah ini merupakan sketsa halaman konsultasi pada user, halaman ini berfungsi membantu user mendeteksi penyakit berdasarkan gejala yang sedang dirasakan dengan menginputkan gejala pada kolom yang tersedia pada sistem.

No	Kondisi	Pilih
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>

11. Tampilan Halaman Hasil Konsultasi User

Gambar di bawah ini merupakan halaman hasil konsultasi user, halaman ini berfungsi menampilkan hasil yang telah didapatkan berdasarkan gejala yang

dirasakan oleh user, hasil yang didapatkan berupa diagnosa penyakit dan solusi penanganan.



12. Tampilan Halaman Riwayat Diagnosa User

Gambar di bawah ini merupakan halaman riwayat diagnosa pada user, halaman ini berfungsi menyimpan dan menampilkan riwayat diagnosa user berupa gejala, penyakit dan solusi yang didapatkan pada saat user melakukan konsultasi, selain itu pada halaman ini juga terdapat tanggal konsultasi dilakukan oleh user.



3.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk menghasilkan spesifikasi kebutuhan ketika sistem yang baru diimplementasikan, adapun analisis ini meliputi:

3.2.1 Kebutuhan Fungsional

- a. Pakar (Admin)
- b. Pasien (Pengunjung)

3.2.2 Kebutuhan Non Fungsional

- a. Perangkat Keras (Hardware)

Berikut merupakan perangkat keras yang digunakan dalam membangun sistem:

1. Processor – AMD 3020e dual-core
2. Menggunakan RAM 4GB
3. Menggunakan SSD 256GB PCI-e NVME
4. Menggunakan *mouse*, *keyboard* dan *monitor* sebagai perantara antar muka.
- b. Perangkat Lunak (Software)

Berikut adalah perangkat lunak yang digunakan dalam membangun sistem:

1. Sistem Operasi Windows 11
2. Google Chrome
3. Visual Studio Code
4. XAMPP

- c. Pengguna (*Brainware*)

Berikut adalah pengguna pada sistem yang dibangun:

1. Pihak Pakar atau Ahli dibidangnya (Admin)
2. Pihak Pengunjung (Pasien)

BAB IV

ANALISIS HASIL PENELITIAN DAN PEMBAHASAN

4.1 Implementasi Perangkat Lunak

Implementasi perangkat lunak dari sistem yang akan dibangun lebih lengkap dijelaskan di bawah ini:
























1. Sistem Operasi: Windows 11
2. Software: Google Chrom, Visual Studio Code, XAMPP

4.1.1 Implementasi Basis Data

Pembuatan *database* dilakukan dengan menggunakan DBMS MySQL, implementasi *database* dapat dilihat seperti di bawah ini:

a. Tabel Users

Tabel users memiliki 9 *field* yaitu id sebagai *primary key*, *namalengkap*, *jeniskelamin*, *umur*, *alamat*, *password*, *level*, *id_paisen* sebagai *foreign key*. Implementasi dari rancangan tabel users dapat dilihat pada gambar 4.1 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(10)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT	 Ubah  Hapus Lainnya
2	namalengkap	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
3	jeniskelamin	enum('Laki-laki', 'Perempuan')	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
4	umur	int(11)			Tidak	Tidak ada			 Ubah  Hapus Lainnya
5	alamat	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
6	email	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
7	email_verified_at	varchar(255)	utf8mb4_unicode_ci		Ya	NULL			 Ubah  Hapus Lainnya
8	password	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
9	level	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
10	remember_token	varchar(100)	utf8mb4_unicode_ci		Ya	NULL			 Ubah  Hapus Lainnya
11	id_paisen	varchar(100)	utf8mb4_unicode_ci		Ya	NULL			 Ubah  Hapus Lainnya

Gambar 4. 1 Tabel Users

Tabel users di atas digunakan untuk menyimpan data pengguna (users) yang telah mendaftar dalam sistem. Dengan memasukan informasi seperti nama, jenis

kelamin, umur, email, dan password yang kemudian disimpan dalam kolom-kolom (*field*) pada tabel ini. Tabel ini juga berfungsi untuk mevalidasi kredensial pengguna saat mencoba masuk ke dalam sistem.

b. Tabel Aturan

Tabel aturan memiliki 5 *field* yaitu id sebagai *primary key*, namaaturan, kondisi, aksiaturan, penyakit_id sebagai *foreign key*. Implementasi dari rancangan tabel aturan dapat dilihat pada gambar 4.2 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(10)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT	 Ubah  Hapus Lainnya
2	namaaturan	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
3	kondisi	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
4	aksiaturan	varchar(255)	utf8mb4_unicode_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
5	penyakit_id	int(11)			Tidak	Tidak ada			 Ubah  Hapus Lainnya
6	created_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya
7	updated_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya

Gambar 4. 2 Tabel Aturan

Tabel aturan di atas digunakan untuk menyimpan aturan-aturan yang digunakan dalam sistem. Setiap aturan memiliki atribut seperti ID aturan, nama aturan, kondisi, dan aksi aturan yang terkait. Kondisi merupakan fakta-fakta atau pernyataan yang harus dipenuhi, sedangkan aksiaturan adalah hasil atau kesimpulan yang akan diambil jika kondisi terpenuhi.

c. Tabel Data Pasien

Tabel pasien memiliki 4 *field* yaitu id sebagai *primary key*, namapasien, jeniskelamin, umur. Implementasi dari rancangan tabel data pasien dapat dilihat pada gambar 4.3 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(11)			Tidak	<i>Tidak ada</i>		AUTO_INCREMENT	 Ubah  Hapus  Lainnya
2	namapatient	varchar(100)	utf8_general_ci		Tidak	<i>Tidak ada</i>			 Ubah  Hapus  Lainnya
3	jeniskelamin	varchar(15)	utf8_general_ci		Tidak	<i>Tidak ada</i>			 Ubah  Hapus  Lainnya
4	umur	int(5)			Tidak	<i>Tidak ada</i>			 Ubah  Hapus  Lainnya
5	created_at	timestamp			Ya	<i>NULL</i>			 Ubah  Hapus  Lainnya
6	updated_at	timestamp			Ya	<i>NULL</i>			 Ubah  Hapus  Lainnya

Gambar 4. 3 Tabel Data Pasien

Tabel data pasien di atas digunakan untuk menyimpan data terkait pasien pada sistem. Pada kolom ID pasien, digunakan untuk menyimpan identifikasi unik setiap pasien, namapatient digunakan untuk menyimpan nama lengkap pasien, jenis kelamin digunakan untuk menyimpan informasi mengenai jenis kelamin pasien, dan umur digunakan untuk menyimpan umur pasien pada sistem.

d. Tabel Gejala

Tabel gejala memiliki 3 *field* yaitu id sebagai *primary key*, kodegejala, kondisi. Implementasi dari rancangan tabel gejala dapat dilihat pada gambar 4.4 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(10)		UNSIGNED	Tidak	<i>Tidak ada</i>		AUTO_INCREMENT	 Ubah  Hapus  Lainnya
2	kodegejala	varchar(255)	utf8mb4_unicode_ci		Tidak	<i>Tidak ada</i>			 Ubah  Hapus  Lainnya
3	kondisi	varchar(255)	utf8mb4_unicode_ci		Tidak	<i>Tidak ada</i>			 Ubah  Hapus  Lainnya
4	created_at	timestamp			Ya	<i>NULL</i>			 Ubah  Hapus  Lainnya
5	updated_at	timestamp			Ya	<i>NULL</i>			 Ubah  Hapus  Lainnya

Gambar 4. 4 Tabel Gejala

Tabel gejala di atas digunakan untuk menyimpan informasi tentang gejala-gejala yang terkait dengan penyakit dalam sistem. Tabel gejala berfungsi untuk mencatat dan mengelola data yang berkaitan dengan gejala.

e. Tabel Gejala Pasien

Tabel gejala pasien memiliki 3 *field* yaitu id sebagai *primary key*, id_pasien sebagai *foreign key*, id_gejala sebagai *foreign key*. Implementasi dari rancangan tabel gejala pasien dapat dilihat pada gambar 4.5 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	 Ubah  Hapus Lainnya
2	id_pasien	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
3	id_gejala	int(11)			Tidak	Tidak ada			 Ubah  Hapus Lainnya
4	created_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya
5	updated_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya

Gambar 4. 5 Tabel Gejala Pasien

Tabel gejala pasien di atas digunakan untuk mencatat gejala yang dialami oleh pasien pada sistem. Tabel ini berfungsi untuk menyimpan informasi tentang gejala-gejala yang dikonsultasikan oleh pasien.

f. Tabel Penyakit

Tabel penyakit memiliki 4 *field* yaitu id sebagai *primary key*, kodepenyakit, nama, deskripsi. Implementasi dari rancangan tabel penyakit dapat dilihat pada gambar 4.6 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	 Ubah  Hapus Lainnya
2	kodepenyakit	varchar(12)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
3	nama	varchar(255)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
4	deskripsi	varchar(1000)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus Lainnya
5	created_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya
6	updated_at	timestamp			Ya	NULL			 Ubah  Hapus Lainnya

Gambar 4. 6 Tabel Penyakit

Tabel penyakit di atas digunakan untuk menyimpan informasi mengenai penyakit-penyakit dalam sistem. Tabel ini berfungsi untuk mencatat dan mengelola data yang

berkaitan dengan penyakit dan kemudian akan dideskripsikan pada kolom deskripsi pada database.

g. Tabel Solusi

Tabel solusi memiliki 4 *field* yaitu id sebagai *primary key*, kodesolusi, solusi, dan deskripsi. Implementasi dari rancangan tabel solusi dapat dilihat pada gambar 4.7 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id 	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	 Ubah  Hapus  Lainnya
2	kodesolusi	varchar(5)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus  Lainnya
3	solusi	varchar(255)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus  Lainnya
4	deskripsi	varchar(1000)	utf8_general_ci		Tidak	Tidak ada			 Ubah  Hapus  Lainnya
5	created_at	timestamp			Ya	NULL			 Ubah  Hapus  Lainnya
6	updated_at	timestamp			Ya	NULL			 Ubah  Hapus  Lainnya

Gambar 4. 7 Tabel Solusi

Tabel solusi di atas digunakan untuk menyimpan informasi mengenai solusi atau tindakan yang direkomendasikan oleh sistem. Tabel ini berfungsi untuk mencatat dan mengelola data yang berkaitan dengan solusi atau langkah-langkah yang dapat diambil dalam berbagai kondisi.

h. Tabel Hasil Diagnosa

Tabel hasil diagnosa memiliki 5 *field* yaitu ID sebagai *primary key*, pasien_id sebagai *foreign key*, penyakit_id sebagai *foreign key*, gejala, dan solusi. Implementasi dari rancangan tabel hasil diagnosa dapat dilihat pada gambar 4.8 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	pasien_id	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
3	penyakit_id	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
4	gejala	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	solusi	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	created_at	timestamp			Ya	NULL			Ubah Hapus Lainnya
7	updated_at	timestamp			Ya	NULL			Ubah Hapus Lainnya

Gambar 4. 8 Tabel Hasil Diagnosa

Tabel hasil diagnosa di atas digunakan untuk menyimpan informasi tentang hasil diagnosa pasien dalam sistem. Tabel ini berfungsi untuk mencatat dan mengelola data yang berkaitan dengan hasil diagnosa yang diperoleh pasien dari proses konsultasi pada sistem.

i. Tabel Riwayat Diagnosa

Tabel riwayat diagnosa memiliki 5 *field* yaitu id sebagai *primary key*, pasien_id sebagai *foreign key*, penyakit_id sebagai *foreign key*, gejala, solusi. Implementasi dari rancangan tabel riwayat diagnosa dapat dilihat pada gambar 4.9 di bawah ini.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	pasien_id	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
3	penyakit_id	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
4	gejala	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	solusi	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	created_at	timestamp			Ya	NULL			Ubah Hapus Lainnya
7	updated_at	timestamp			Ya	NULL			Ubah Hapus Lainnya

Gambar 4. 9 Tabel riwayat diagnosa

Tabel riwayat diagnosa di atas digunakan untuk menyimpan informasi tentang riwayat diagnosa pasien dalam sistem. Tabel ini berfungsi untuk mencatat dan mengelola data yang berkaitan dengan riwayat diagnosa pasien, termasuk diagnosa

sebelumnya dan perubahan diagnosa sebelumnya dan perubahan diagnosa dari waktu ke waktu.

4.1.2 Implementasi *Interface*

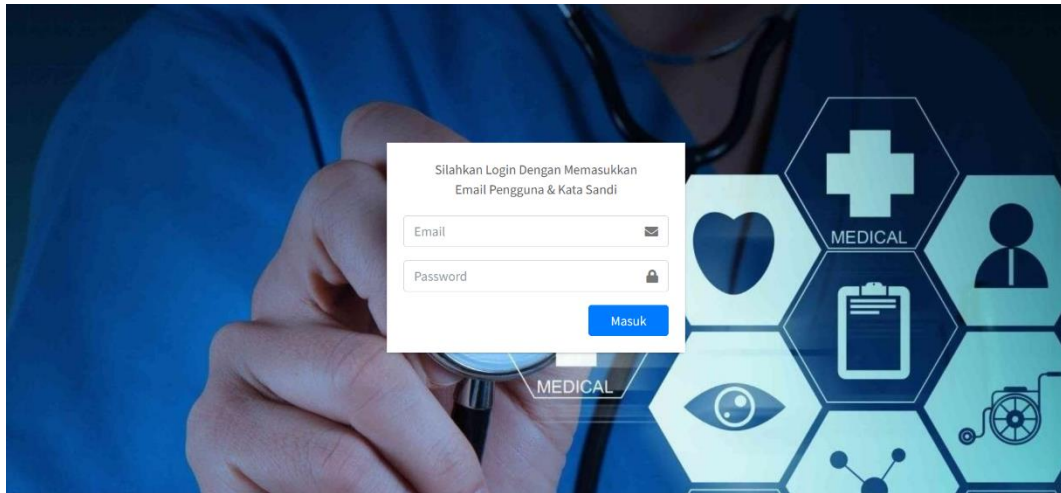
a. Pembuatan Halaman Login Admin

Halaman login merupakan halaman yang hanya dapat diakses oleh admin untuk dapat mengakses sistem dengan memasukkan email dan password masing-masing yang sudah terdaftar pada. Berikut merupakan potongan *source code* halaman login.

```
<form action="/postlogin" method="post">
  {{ csrf_field() }}
  <div class="input-group mb-3">
    <input type="email" class="form-control" name="email"
    placeholder="Email">
    <div class="input-group-append">
      <div class="input-group-text">
        <span class="fas fa-envelope"></span>
      </div>
    </div>
  </div>
  <div class="input-group mb-3">
    <input type="password" class="form-control" name="password"
    placeholder="Password">
    <div class="input-group-append">
      <div class="input-group-text">
        <span class="fas fa-lock"></span>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-8">
    </div>
    <div class="col-4">
      <button type="submit" class="btn btn-primary btn-
      block">Masuk</button>
    </div>
  </div>
</form>
```

1. Pengujian *white box* halaman login admin

Pengujian *white box* dilakukan dengan pengujian kode program (coding). Hasilnya berupa tampilan halaman login yang dapat dilihat pada gambar 4.10 di bawah ini.



Gambar 4. 10 Halaman Login

Pada halaman login di atas admin diharuskan memasukan email dan password kemudian menekan tombol “masuk” dan sistem akan mengarahkan admin agar dapat mengakses data pada sistem.

2. Pengujian *black box* halaman login

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman login. Pengujian dan hasil uji *black box* halaman login dapat dilihat pada tabel 4.2 di bawah ini.

Tabel 4. 1 Tabel pengujian *black box* halaman login

Aktivitas	Realisasi yang diharapkan	Hasil
Memasukan email dan password yang salah, kemudian mengklik tombol masuk.	Gagal login, dan tetap di halaman login	Sukses
Memasukan email dan password yang benar kemudian mengklik tombol masuk.	Masuk ke dalam sistem, dan dapat mengakses menu dan data yang ada pada sistem	Sukses

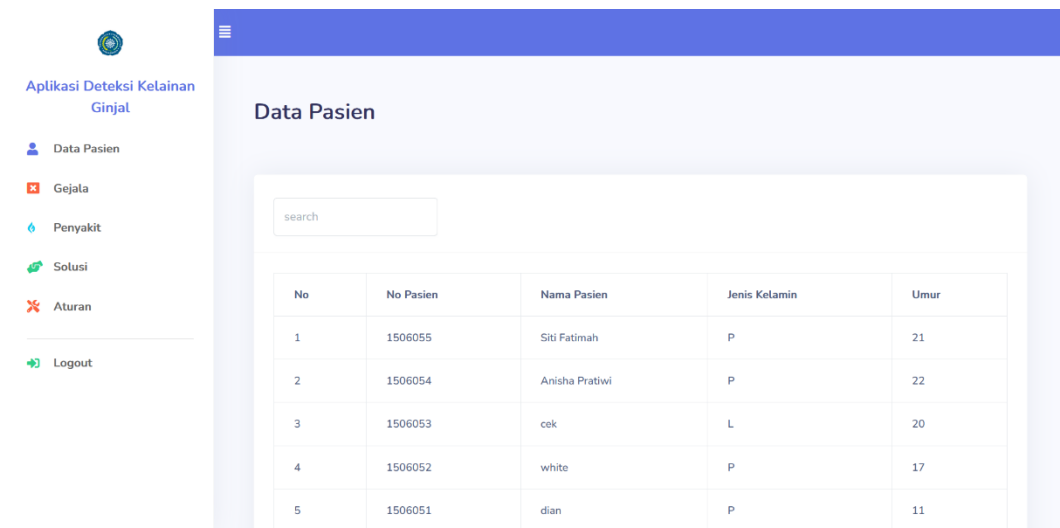
b. Pembuatan Halaman Data Pasien

Halaman data pasien adalah halaman yang difungsikan untuk menyimpan data pasien telah melakukan registrasi akun pada sistem. Berikut potongan *source code* halaman data pasien.

```
<div class="card-body">
<table class="table table-bordered">
<tr>
<th>No</th>
<th>No Pasien</th>
<th>Nama Pasien</th>
<th>Jenis Kelamin</th>
<th>Umur</th>
</tr>
@php
$no = 1;
@endphp
@foreach ($Datapasien as $index => $row )
<tr>
<td>{{ $index + $Datapasien->firstItem()}}</td>
<td>{{ $row->id }}</td>
<td>{{ $row->namapasien }}</td>
<td>{{ $row->jeniskelamin }}</td>
<td>{{ $row->umur }}</td>
</tr>
@endforeach
</table>
```

1. Pengujian *white box* pada halaman data pasien

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman data pasien yang dapat dilihat pada gambar 4.12 di bawah ini.



Gambar 4. 11 Halaman Data Pasien

Halaman data pasien ini digunakan untuk menyimpan informasi pasien secara elektronik setelah pasien memasukan data diri berupa nama lengkap, jenis kelamin, dan umur.

2. Pengujian *black box* pada halaman data pasien

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data pasien. Pengujian dan hasil uji *black box* dapat dilihat pada tabel di bawah ini.

Tabel 4. 2 Tabel pengujian black box halaman data pasien

Aktivitas	Realisasi yang diharapkan	Hasil
Mengklik menu data pasien, kemudian akan masuk ke halaman daftar data pasien.	Masuk ke halaman daftar data pasien.	Sukses
Mengklik search dengan mengisi nama atau nomer pasien.	Keluar nama atau nomer pasien yang dipanggil.	Sukses

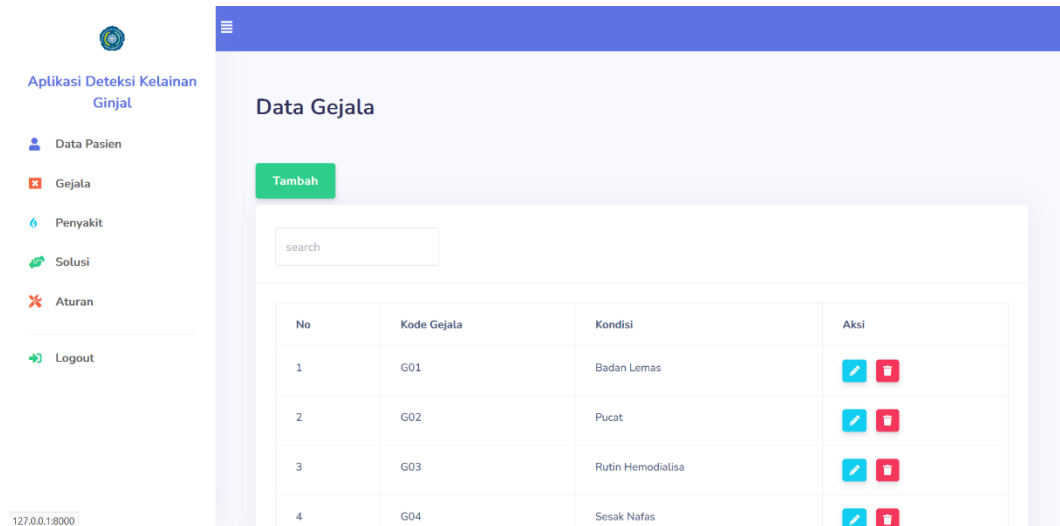
c. Pembuatan Halaman Gejala

Halaman gejala adalah halaman yang difungsikan untuk menambah, mengubah, dan menghapus data-data gejala yang ada pada sistem. Berikut potongan *source code* halaman data gejala.

```
<div class="card-body">
<table class="table table-bordered">
<tr>
<th>No</th>
<th>Kode Gejala</th>
<th>Kondisi</th>
<th>Aksi</th>
</tr>
@php
$no = 1;
@endphp
@foreach ($Datagejala as $index => $row )
<tr>
<td>{{ $index + $Datagejala->firstItem() }}</td>
<td>{{ $row->kodegejala }}</td>
<td>{{ $row->kondisi }}</td>
<td>
<a class="btn btn-info btn-sm" href="/editdatagejala/{{ $row->id
}}">
<i class="fas fa-pencil-alt">
</i>
</a>
<a class="btn btn-danger btn-sm delete" href="/deletedatagejala/{{
$row->id }}">
<i class="fas fa-trash">
</i>
</a>
</td>
</tr>
@endforeach
</table>
```

1. Pengujian *white box* halaman data gejala

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman data gejala yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 12 Halaman Gejala

Halaman data gejala digunakan untuk menyimpan informasi mengenai gejala yang ada pada sistem. Halaman data gejala ini berfungsi memberikan informasi kepada sistem dalam mendiagnosa penyakit berdasarkan gejala yang dirasakan oleh pasien.

2. Pengujian *black box* halaman data gejala

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman gejala. Pengujian dan hasil uji *black box* dapat dilihat pada tabel berikut.

Tabel 4. 3 Tabel pengujian black box halaman data gejala

Aktivitas	Realisasi yang diharapkan	Hasil
Mengklik menu gejala, akan masuk ke halaman daftar data gejala.	Masuk ke halaman daftar data gejala.	Sukses
Mengklik tombol tambah, kemudian	Muncul form untuk menambah data gejala lalu mengklik	Sukses

menginputkan data gejala “kode gejala, kondisi” kemudian klik tombol submit untuk menyimpan.	submit dan berhasil menyimpan data gejala ke dalam <i>database</i> , serta kembali ke halaman data gejala.	
Mengklik simbol “pena” maka masuk ke halaman edit gejala, kemudian dapat menekan tombol “submit” untuk menyimpan data gejala terbaru.	Masuk ke halaman edit data gejala, kemudian dapat mengubah data gejala dan menyimpan, kemudian akan masuk ke dalam <i>database</i> dan kembali ke halaman data gejala.	Sukses

d. Pembuatan Halaman Penyakit

Halaman penyakit adalah halaman yang difungsikan untuk menambah, mengubah, dan menghapus data-data penyakit yang ada pada sistem. Berikut potongan *source code* halaman data penyakit.

```
<div class="card-body">
<table class="table table-bordered">
<tr>
<th>No</th>
<th>Kode Penyakit</th>
<th>Nama Penyakit</th>
<th>Deskripsi</th>
<th>Aksi</th>
</tr>
<@php
$no = 1;
@endphp
@foreach ($Datapenyakit as $index => $row )
<tr>
<td>{{ $index + $Datapenyakit->firstItem() }}</td>
<td>{{ $row->kodepenyakit }}</td>
<td>{{ $row->nama }}</td>
<td>{{ $row->deskripsi }}</td>
```



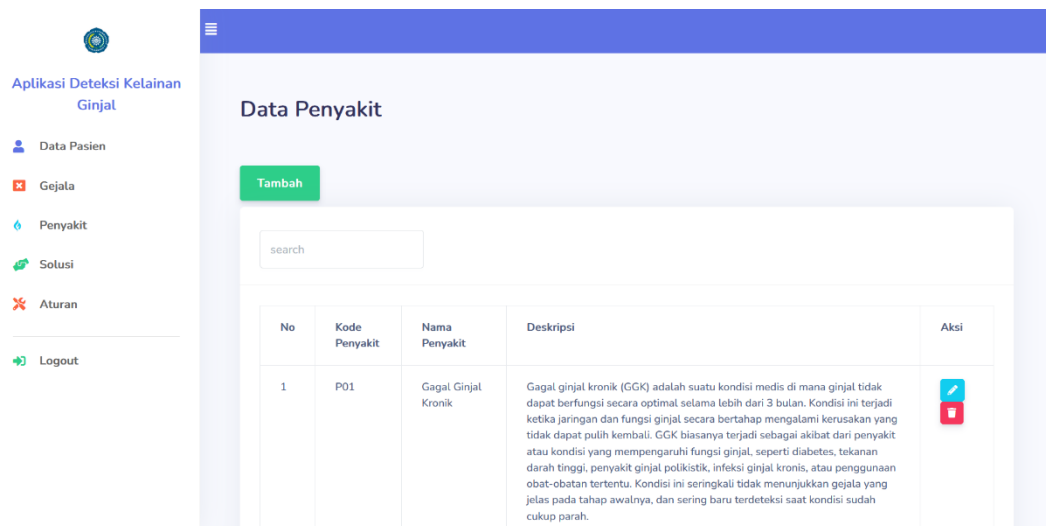
```

<td>
<a class="btn btn-info btn-sm" href="/editdatapenyakit/{{ $row->id
}}">
<i class="fas fa-pencil-alt">
</i>
</a>
<a class="btn btn-danger btn-sm delete" href="/deletedatapenyakit/{{
$row->id }}">
<i class="fas fa-trash">
</i>
</a>
</td>
</tr>
@endforeach
</table>
</div>

```

1. Pengujian *white box* pada halaman penyakit

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman data penyakit yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 13 Halaman penyakit

Halaman data penyakit ini digunakan untuk menyimpan data informasi mengenai penyakit pada sistem. Halaman ini berisikan kode penyakit, nama penyakit, dan deskripsi yang menjelaskan detail penyakit untuk memberikan pemahaman tentang kondisi kesehatan pasien.

2. Pengujian *black box* pada halaman penyakit

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data penyakit. Pengujian dan hasil uji *black box* dapat dilihat pada tabel di bawah ini.

Tabel 4. 4 Tabel pengujian black box halaman penyakit

Aktivitas	Realisasi yang diharapkan	Hasil
Mengklik tombol tambah, kemudian menginputkan data penyakit “kode penyakit, nama penyakit, deskripsi” kemudian klik tombol submit untuk menyimpan.	Muncul form untuk menambah data penyakit lalu mengklik submit dan berhasil menyimpan data penyakit ke dalam <i>database</i> , serta kembali ke halaman data penyakit.	Sukses

e. Pembuatan Halaman Solusi

Halaman solusi adalah halaman yang difungsikan untuk menambah, mengubah, dan menghapus data-data solusi yang ada pada sistem. Berikut potongan *source code* halaman data solusi.

```
<div class="card-body">  
    <table class="table table-bordered">  
        <tr>  
            <th>No</th>  
            <th>Kode Solusi</th>  
            <th>Solusi</th>  
            <th>Deskripsi</th>  
            <th>Aksi</th>  
        </tr>  
        @php  
        $no = 1;  
        @endphp  
        @foreach ($Datasolusi as $index => $row )  
            <tr>
```

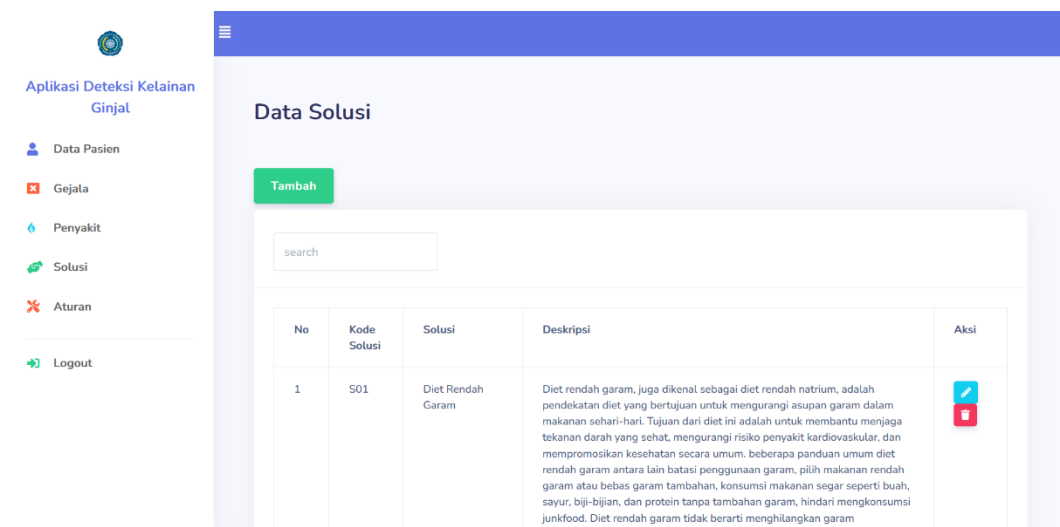
```

<td>{{ $index + $Datasolusi->firstItem() }}</td>
<td>{{ $row->kodesolusi }}</td>
<td>{{ $row->solusi }}</td>
<td>{{ $row->deskripsi }}</td>
<td>
<a class="btn btn-info btn-sm" href="/editdatasolusi/{{ $row->id
}}">
<i class="fas fa-pencil-alt">
</i>
</a>
<a class="btn btn-danger btn-sm delete" href="/deletedatasolusi/{{
$row->id }}">
<i class="fas fa-trash">
</i>
</a>
</td>
</tr>
@endforeach
</table>
</div>

```

1. Pengujian *white box* pada halaman solusi

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman data solusi yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 14 Halaman solusi

Halaman solusi digunakan untuk menyimpan informasi tentang solusi atau tindakan yang direkomendasikan untuk mengatasi penyakit yang terdiagnosa oleh sistem. Halaman ini memberikan informasi detail mengenai solusi atau tindakan yang diberikan sistem kepada pasien setelah melakukan konsultasi.

2. Pengujian *black box* pada halaman solusi

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data solusi. Pengujian dan hasil uji *black box* dapat dilihat pada tabel di bawah ini.

Tabel 4. 5 Pengujian black box pada halaman solusi

Aktivitas	Realisasi yang diharapkan	Hasil
Memilih menu solusi, kemudian akan masuk ke halaman daftar solusi.	Masuk ke halaman daftar solusi.	Sukses
Mengklik tombol tambah, kemudian menginputkan data solusi “kode solusi, solusi” kemudian klik tombol submit untuk menyimpan.	Muncul form untuk menambah data solusi lalu mengklik submit dan berhasil menyimpan data solusi ke dalam <i>database</i> , serta kembali ke halaman data solusi.	Sukses

f. Pembuatan Halaman Aturan

Halaman aturan adalah halaman yang difungsikan untuk menampilkan data aturan yang ada pada database ke sistem. Berikut potongan *source code* halaman data aturan.

```
<div class="card-body">
<table class="table table-bordered">
<tr>
```

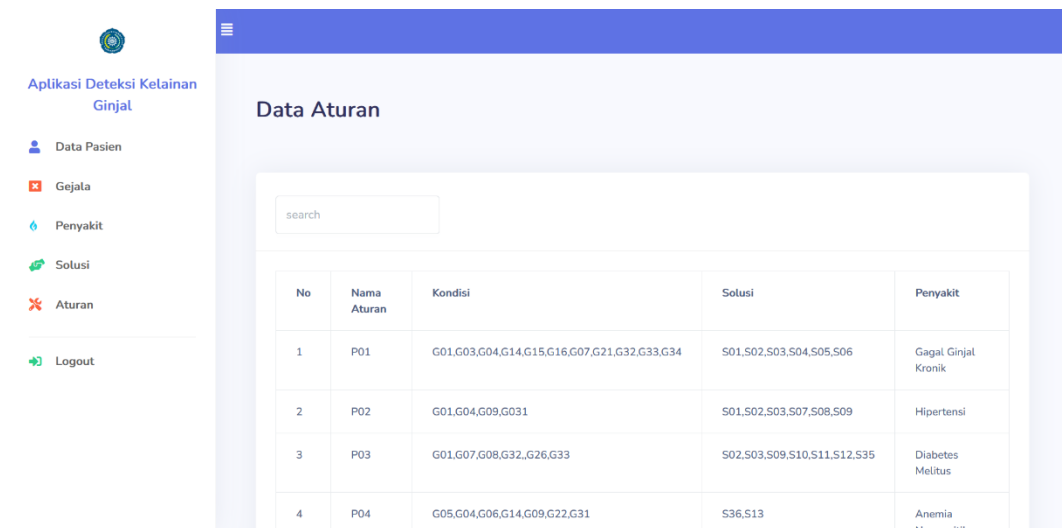
```

<th>No</th>
<th>Kode Solusi</th>
<th>Solusi</th>
<th>Deskripsi</th>
<th>Aksi</th>
</tr>
@php
$no = 1;
@endphp
@foreach ($Datasolusi as $index => $row )
<tr>
<td>{{ $index + $Datasolusi->firstItem() }}</td>
<td>{{ $row->kodesolusi }}</td>
<td>{{ $row->solusi }}</td>
<td>{{ $row->deskripsi }}</td>
<td>
<a class="btn btn-info btn-sm" href="/editdatasolusi/{{ $row->id
}}">
<i class="fas fa-pencil-alt">
</i>
</a>
<a class="btn btn-danger btn-sm delete" href="/deletedatasolusi/{{
$row->id }}">
<i class="fas fa-trash">
</i>
</a>
</td>
</tr>
@endforeach
</table>
</div>

```

1. Pengujian *white box* halaman aturan

Pengujian *white box* dilakukan dengan pengujian kode program (*coding*). Hasilnya berupa tampilan halaman aturan yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 15 Halaman Aturan

Halaman aturan ini berisikan daftar aturan atau *rule* penyakit dan solusi berdasarkan gejala yang ada pada *database* sistem.

2. Pengujian *black box* halaman aturan

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman aturan. Pengujian dan hasil uji *black box* dapat dilihat pada tabel di bawah ini.

Tabel 4. 6 Pengujian black box pada halaman aturan

Aktivitas	Realisasi yang diharapkan	Hasil
Mengklik search dengan mengisi nama aturan, kondisi, solusi	Keluar nama aturan, kondisi, solusi yang dipanggil.	Sukses

g. Pembuatan Halaman Pasien

Halaman pasien adalah halaman yang difungsikan untuk pasien agar dapat mengakses website dengan memasukan ID pada menu “pasien sudah terdaftar” apabila pasien telah terdaftar, jika belum terdaftar maka pasien harus memasukan data diri dengan memilih menu “pasien belum terdaftar”. Berikut potongan *source code* halaman pasien.

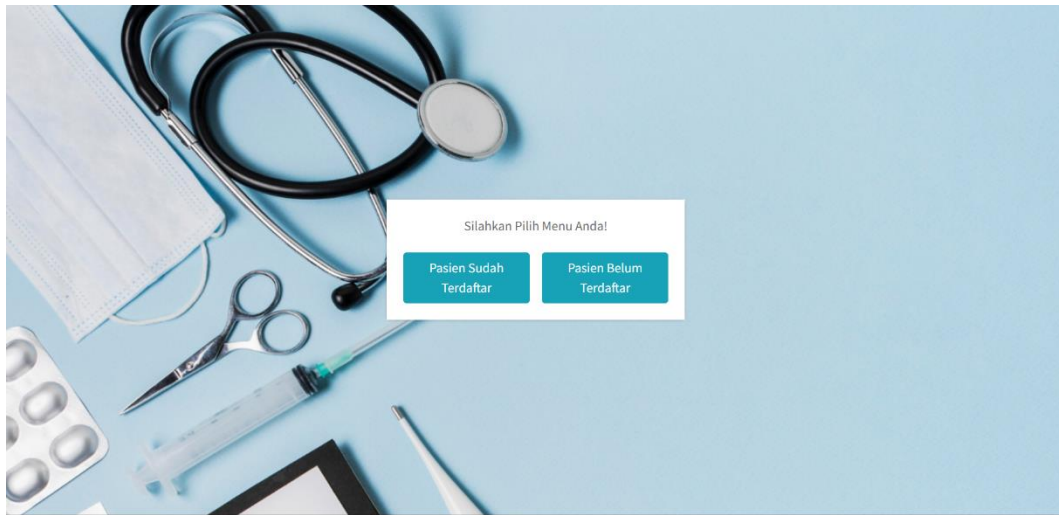
```

<div class="card">
<div class="card-body login-card-body">
<p class="login-box-msg">Silahkan Pilih Menu Anda!</p>
<div class="row">
<div class="col-6 text-center">
<a href="/loginId" class="btn btn-info">Pasien Sudah Terdaftar</a>
</div>
<div class="col-6 text-center">
<a href="/viewregister" class="btn btn-info">Pasien Belum
Terdaftar</a>
</div>
</div>
<form action="/postloginId" method="post">
{{ csrf_field() }}
<div class="input-group mb-3">
<input type="number" class="form-control" name="id"
placeholder="Input ID">
<div class="input-group-append">
<div class="input-group-text">
<span class="fas fa-"></span>
</div>
</div>
<form action="/register" method="post">
{{ csrf_field() }}
<div class="input-group mb-3">
<input type="text" class="form-control" name="namalengkap"
placeholder="Nama Lengkap">
</div>
<div class="input-group mb-3">
<select class="form-control" name="jeniskelamin" aria-label="Default
select example">
<option selected>Jenis Kelamin</option>
<option value="Laki-laki">Laki-laki</option>
<option value="Perempuan">Perempuan</option>
</select>
</div>
<div class="input-group mb-3">
<input type="number" class="form-control" name="umur"
placeholder="Umur">
</div>

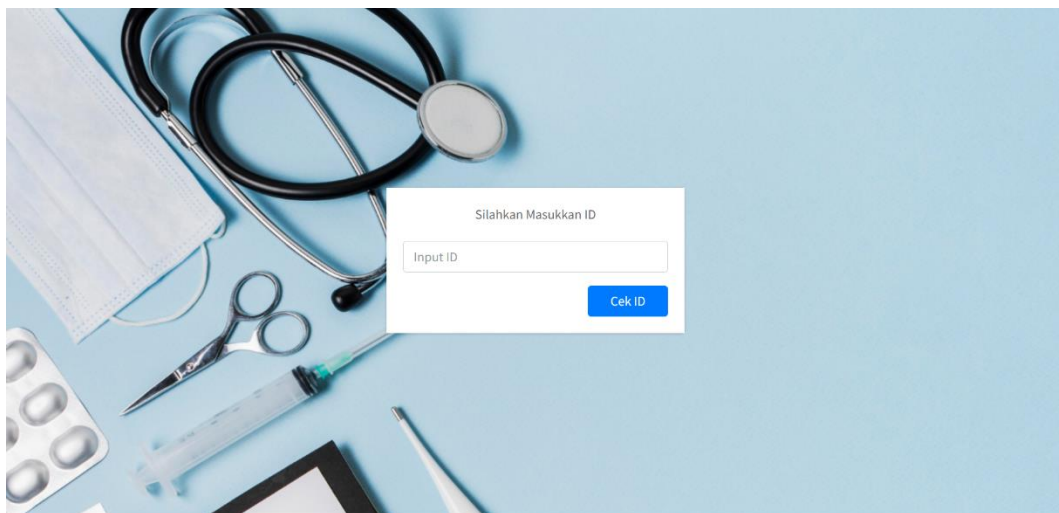
```

1. Pengujian white box halaman pasien

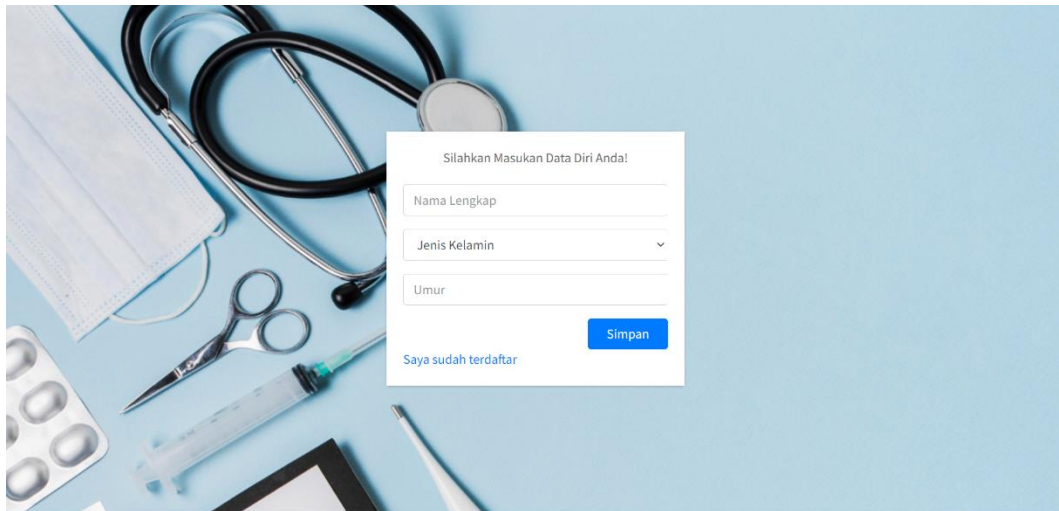
Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman pasien yang terdiri dari halaman masukan ID dan halaman masukan datadiri yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 16 Halaman pasien



Gambar 4. 17 Halaman masukan ID pasien



Gambar 4. 18 Halaman masukan data diri pasien

2. Pengujian black box halaman pasien

Pengujian *black box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman pasien. Pengujian dan hasil uji *black box* dapat dilihat pada tabel di bawah ini.

Tabel 4. 7 Pengujian black box pada halaman pasien

Aktivitas	Realisasi yang diharapkan	Hasil
Memilih menu “pasien sudah terdaftar” dengan mengklik tombol yang ada pada sistem	Masuk ke halaman <i>input</i> ID bagi pasien yang telah terdaftar dan mendapatkan ID pasien agar kemudian dapat melakukan konsultasi	Sukses
Memilih menu “pasien belum terdaftar” dengan mengklik tombol yang ada pada sistem	Masuk ke halaman <i>input</i> data diri bagi pasien yang belum terdaftar agar mendapatkan ID pasien agar kemudian dapat melakukan konsultasi	Sukses

h. Pembuatan Halaman Konsultasi

Halaman konsultasi adalah halaman yang difungsikan untuk menampilkan data gejala yang dapat dipilih pengguna berdasarkan keadaan yang dirasakan. Berikut potongan *source code* halaman konsultasi.

```
<div class="mb-4 d-flex justify-content-between">
<h1>Halaman Konsultasi</h1>
<h1>Nomor Pasien / ID : {{ auth()->user()->id }}</h1>
</div>
<section class="content">
<div class="content">
<div class="card card-info card-outline">
<div class="card-header">
<h5></h5>
<div class="row g-3 align-items-center">
<div class="col-auto">
<h5>Silahkan pilih kondisi di bawah sesuai dengan yang anda rasakan
saat ini.</h5>
<h6>Tekan Ctrl+Klik untuk memilih kondisi lebih dari satu</h6>
</div>
</div>
</div>
</div>
<form method="post" action="{{ route('diagnosa') }}" class="card-
body">
@csrf
<select name="gejala[]" multiple style="height: 300px;" class="form-
control">
@foreach ($data['gejala'] as $index => $row )
<option value="{{ $row->id }}">{{ $row->kondisi }}</option>
@endforeach
</select>
<button type="submit" class="btn btn-primary mt-2 float-
right">Simpan</button>
</form>
</div>
<div class="mb-5">
<h1>Halaman Hasil Konsultasi</h1>
</div>
<div class="card-body">
<table class="table table-bordered">
<tr>
<th>No</th>
<th>Kondisi</th>
```

```

<th>Tanggal Konsultasi</th>
</tr>
@php
$no = 1;
@endphp
@foreach ($Datagejala as $index => $row )
<tr>
<td>{{ $index + $Datagejala->firstItem() }}</td>
<td>{{ $row->kondisi }}</td>
<td><input type="checkbox"></td>
</tr>
@endforeach
</table>
</div>

```

1. Pengujian *white box* halaman konsultasi dan hasil konsultasi

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman konsultasi yang dapat dilihat pada gambar di bawah ini.

Gambar 4. 19 Halaman konsultasi



Gambar 4. 20 Halaman Hasil konsultasi

Halaman konsultasi diatas berfungsi memberikan akses ke informasi dan saran mengenai penyakit dan solusi berdasarkan gejala yang dialami oleh pasien dengan memilih pada box daftar gejala yang ada pada siste. Selanjutnya pasien dapat mengklik tombol “Simpan” yang kemudian pasien akan diarahkan oleh sistem ke halaman hasil konsultasi.

Sedangkan halaman hasil konsultasi di atas berfungsi menampilkan diagnosa penyakit, saran atau tindakan yang direkomendasikan oleh sistem berdasarkan gejala yang dirasakan oleh pasien.

2. Pengujian *black box* halaman konsultasi dan hasil konsultasi

Pengujian *black box* pada halaman konsultasi dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman. Pengujian dan hasil uji dapat dilihat pada tabel di bawah ini.

Tabel 4. 8 Pengujian Black Box pada halaman konsultasi dan hasil konsulasi

Aktivitas	Realisasi yang diharapkan	Hasil
Memilih gejala dengan mengklik <i>check box</i> ,	Masuk ke halaman hasil konsultasi dan muncul hasil diagnosa berupa gejala,	Sukses

kemudian mengklik tombol “simpan”	penyakit, solusi dari gejala yang dipilih	
-----------------------------------	---	--

i. Pembuatan Halaman Riwayat Diagnosa

Halaman riwayat diagnosa adalah halaman yang difungsikan untuk menampilkan hasil diagnosa berupa gejala, penyakit, solusi, dan tanggal dilakukannya konsultasi.

Berikut potongan *source code* halaman riwayat diagnosa.

```
<div class="mb-5">
<h1>Riwayat Diagnosa</h1>
</div>
<section class="content">
<div class="content">
<div class="card card-info card-outline">
<div class="card-body">
<table class="table mt-4 table-bordered">
<tr>
<th>No</th>
<th>Tanggal</th>
<th>Ops</th>
</tr>
<tr>
<td>{{ $id+1 }}</td>
<td>{{ $value->created_at }}</td>
<td>
<a href="{{ url('detail-riwayat-diagnosa/'.$value->id) }}"
class="btn btn-info">Detail</a>
</td>
</tr>
</table>
</div>
<div class="mb-5">
<h1>Detail Riwayat Diagnosa</h1>
</div>
<section class="content">
<div class="content">
<div class="card card-info card-outline">
<h5 class="mt-3 ml-3 mx-4">
```

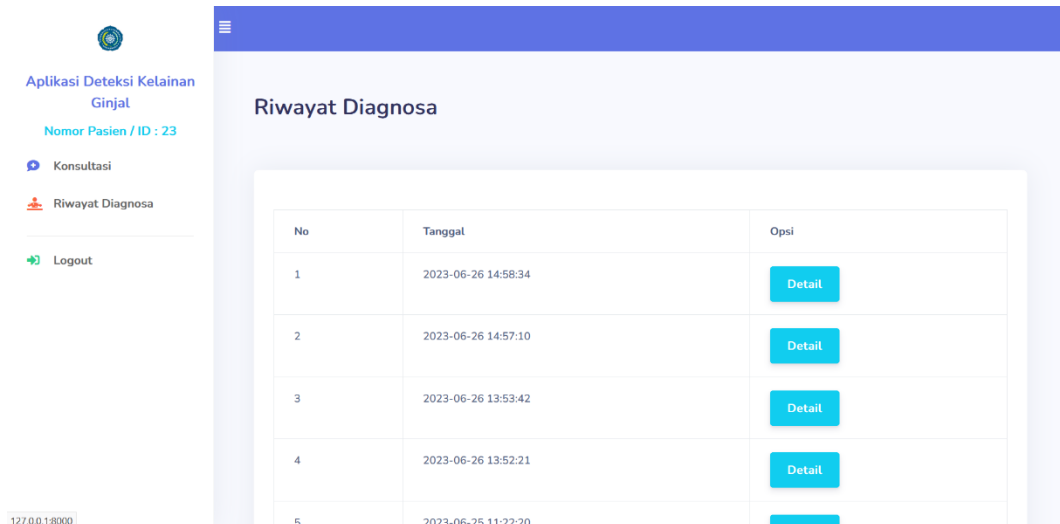
```

Tanggal Dibuat : {{ date('d M Y',
strtotime($data['riwayatdiagnosa']->updated_at)) }}
<a href="{{ url('riwayat-diagnosa') }}" class="btn btn-primary
float-right">Kembali</a>
</h5>
<div class="card-body">
<table class="table mt-4 table-bordered">
<tr>
<th>Gejala</th>
<th>Diagnosa penyakit</th>
<th>Solusi</th>
</tr>
<tr>
<td>
<ul>
@foreach($data['gejala'] as $val)
<li>{{ $val->kondisi }}</li>
@endforeach
</ul>
</td>
<td width="">
{{ $data['riwayatdiagnosa']->penyakit->nama }} <br><hr>
Deskripsi : {{ $data['riwayatdiagnosa']->penyakit->deskripsi }}
</td>
<td>
<ul>
@foreach($data['solusi'] as $val)
<li>
{{ $val->solusi }}
<hr style="margin:0">
{{ $val->deskripsi }}
<br>
<br>
</li>
@endforeach
</ul>
</td>
</tr>
</table>
</div>

```

1. Pengujian *white box* pada halaman riwayat diagnosa

Pengujian *white box* dilakukan dengan pengujian kode program. Hasilnya berupa tampilan halaman riwayat diagnosa yang dapat dilihat pada gambar di bawah ini.



Gambar 4. 21 Halaman Riwayat Diagnosa



Gambar 4. 22 Halaman Detail Riwayat Diagnosa

Halaman riwayat diagnosa di atas berfungsi sebagai tempat untuk merekam informasi mengenai diagnosa penyakit sebelumnya yang telah dilakukan oleh pasien.

Sedangkan halaman detail riwayat diagnosa berfungsi menyajikan informasi secara detail atau lengkap bagi setiap diagnosa medis yang telah diterima oleh pasien.

2. Pengujian *black box* pada halaman riwayat diagnosa

Pengujian *black box* pada halaman riwayat diagnosa dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman. Pengujian dan hasil uji dapat dilihat pada tabel di bawah ini.

Aktivitas	Realisasi yang diharapkan	Hasil
Memilih menu riwayat diagnosa, kemudian masuk ke halaman riwayat diagnosa	Masuk ke halaman riwayat diagnosa dengan gejala yang dipilih, penyakit, dan solusi	Sukses
Mengklik tombol “detail” dan kemudian masuk ke halaman detail riwayat diagnosa	Masuk ke halaman detail riwayat diagnosa yang berisikan informasi gejala, penyakit, solusi dan deskripsi penyakit serta solusi/	Sukses

4.2 Pembahasan

Aplikasi sistem pakar diagnosa penyakit ginjal menggunakan metode *Forward Chaining* memiliki fitur yang telah melalui proses pengujian perangkat lunak dengan metode *black box testing*. Hasil dari pengujian perangkat lunak dengan metode *black box* ini memberikan inputan pada setiap komponen yang ada pada setiap fitur aplikasi, dan memastikan bahwa setiap inputan yang dilakukan dapat berjalan dengan baik sesuai dengan fungsi tiap komponen yang ada pada fitur aplikasi yang dibuat.

Aplikasi ini diharapkan dapat membantu pihak pengguna dalam mendeteksi dini penyakit ginjal pada tubuh tanpa harus mengeluarkan dana yang cukup besar.

BAB V

KESIMPULAN DAN PENELITIAN SELANJUTNYA

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis maka sistem pakar berbasis web menggunakan *forward chaining* ini telah dibangun untuk mendeteksi penyakit ginjal kronik pada manusia. Aplikasi ini membantu masyarakat dalam mengidentifikasi gejala awal penyakit ginjal kapan dan di mana saja melalui web yang dapat diakses dengan mudah. Dalam sistem ini, digunakan database gejala yang telah ditentukan untuk mendukung proses diagnosa penyakit ginjal. Metode *forward chaining* dipilih sebagai metode penalaran dan mesin inferensi yang menghubungkan gejala yang diinputkan dengan aturan yang ada pada *database*, sehingga dapat menghasilkan kesimpulan terkait penyakit ginjal yang dialami oleh penderita.

Aplikasi ini telah melalui pengujian metode *black box* guna memastikan kinerja dan fungsionalitas yang optimal pada setiap komponen dan fitur aplikasi. Diharapkan aplikasi ini dapat menjadi alternatif yang lebih terjangkau dan mudah digunakan dalam mendeteksi gejala awal penyakit ginjal, sehingga dapat membantu mengurangi biaya dan mempercepat proses penanganan medis yang tepat bagi penderita.

5.2 Penelitian Selanjutnya

Untuk pengembangan penelitian selanjutnya disarankan untuk mengimplementasikan metode-metode pendukung sistem pakar lainnya selain metode *forward chaining* yang digunakan dalam sistem pakar. Pengembangan sistem juga dapat dilakukan dengan mengembangkan dan memperluas *database* gejala penyakit ginjal yang dapat dipadukan dengan informasi medis terbaru, memperluas integrasi aplikasi dengan data medis dan rekam medis elektronik dapat membantu sistem mengakses informasi lebih lengkap tentang pasien. Hal ini dapat membantu dokter atau pakar dalam mendiagnosa penyakit ginjal dengan lebih mudah. Menambahkan fitur edukasi dan kesadaran tentang pentingnya kesehatan

ginjal, faktor risiko, pencegahan, dan perawatan dapat membantu meningkatkan pemahaman masyarakat tentang penyakit ginjal serta mendorong upaya pencegahan yang lebih baik. Dan saran pengembangannya juga dapat dikembangkan dalam versi aplikasi yang kompatibel dengan perangkat mobile akan memberikan fleksibilitas kepada pengguna untuk mengaksesnya di mana saja dan kapan saja, bahkan saat mereka sedang berada di perjalanan.

JADWAL PENYUSUNAN TUGAS AKHIR

No	Kegiatan	Jadwal Pelaksanaan Penelitian Tugas Akhir 2022-2023																							
		Desember				Januari				Februari				Maret				April				Mei			
		I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV
1	Analisis																								
2	Desain Sistem																								
3	Pembangunan Sistem/Coding																								
4	Pengujian Sistem																								
5	Implementasi																								
6	Pembuatan Laporan																								

DAFTAR PUSTAKA

- [1] W. Ginting, “Sistem Pakar Mendiagnosa Penyakit Ginjal Menggunakan Metode Case Based Reasoning,” *Univ. Katolik St.*, vol. 3, no. 2, pp. 223–228, 2018.
- [2] A. Ariani and Samsuryadi, “Klasifikasi Penyakit Ginjal Kronis menggunakan K-Nearest Neighbor,” *Pros. Annu. Res. Semin. 2019*, vol. 5, no. 1, pp. 148–151, 2019.
- [3] W. A. Pulungan, “Sistem Pakar Menentukan Penyakit Ginjal Dengan Metode Forward Chaining,” *Ultim. InfoSys J. Ilmu Sist. Inf.*, vol. 11, no. 1, pp. 27–32, 2020, doi: 10.31937/si.v9i1.1369.
- [4] E. A. U. Malahina, “Sistem Pakar Deteksi Penyakit Ginjal Berbasis Mobile Android,” *High Educ. Organ. Arch. Qual. J. Teknol. Inf.*, vol. 10, no. 1, pp. 6–13, 2018, doi: 10.52972/hoaq.vol10no1.p6-13.
- [5] M. R. Ritonga, S. Solikhun, M. R. Lubis, and A. P. Windarto, “Sistem Pakar Diagnosa Gejala Awal Penyakit Akibat Virus Pada Anak Berbasis Mobile Dengan Forward Chaining,” *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 2, pp. 140–145, 2018, doi: 10.30743/infotekjar.v2i2.298.
- [6] M. A. Hendry, M. I. Gumanof, F. R. Mulya, and D. Meidelfi, “Sistem Pakar Penyakit Ginjal Berbasis Web,” *Inspir. J. Teknol. Inf. dan Komun.*, vol. 11, no. 1, p. 80, 2021, doi: 10.35585/inspir.v11i1.2609.
- [7] H. W. Putra, “Sistem Pakar Diagnosis Penyakit Ginjal Dengan Metoda Forward Chaining,” *J. Sains dan Inform.*, vol. 5, no. 1, p. 7, 2019, doi: 10.22216/jsi.v5i1.4081.
- [8] . R., M. Arif, and . H., “Hubungan Antara Kadar Hemoglobin Dengan Kadar Kreatinin Serum Penderita Penyakit Ginjal Menahun (Kronis),” *Indones. J. Clin. Pathol. Med. Lab.*, vol. 13, no. 3, p. 97, 2018, doi: 10.24293/ijcpml.v13i3.908.

- [9] P. Savitri and T. Hadi, "Implementasi Metode Forward Chaining Dalam Sistem Pendeteksi Kerusakan Hardware Pada Komputer Dan Laptop Berbasis Android," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 9, no. 1, pp. 623–632, 2018, doi: 10.24176/simet.v9i1.2004.
- [10] M. Ibrohim and N. Purwanty, "Rancang Bangun Aplikasi Identifikasi Gaya Belajar Siswa Dengan Metode Forward Chaining (Studi Kasus: Sekolah Dasar Negeri Sumampir)," *ProTekInfo(Pengembangan Ris. dan Obs. Tek. Inform.*, vol. 4, no. 2003, pp. 19–28, 2017, doi: 10.30656/protekinfo.v4i0.408.
- [11] I. W. Gamadarenda and I. Waspada, "Implementasi Data Mining untuk Deteksi Penyakit Ginjal Kronis (PGK) menggunakan K-Nearest Neighbor (KNN) dengan Backward Elimination," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, p. 417, 2020, doi: 10.25126/jtiik.2020721896.
- [12] M. T. Dr. Heny Pratiwi. S.Kom., M.Pd., *BUKU AJAR: SISTEM PAKAR*. Kuningan - Jawa Barat: Goresan Pena, 2016. [Online]. Available: https://books.google.co.id/books?hl=id&lr=&id=5tk9EAAAQBAJ&oi=fnd&pg=PA3&dq=sistem+pakar&ots=cqOzJmF7p_&sig=_dOyygb83G2RuvoIHokzVVcuogM&redir_esc=y#v=onepage&q&f=false
- [13] A. R. Sri Mujiyono, Ita Adhawiyah, "Implementasi Metode Forward Chaining Dalam Sistem Pakar Pendeteksi Gangguan Autism Pada Anak," in *Bukel*, M. P. Andriyanto, Ed. Lakeisha, p. 118. [Online]. Available: https://www.google.co.id/books/edition/IMPLEMENTASI_METODE_FORWARD_CHAINING_DAL/Bs6vEAAAQBAJ?hl=id&gbpv=0
- [14] N. Susilo, A. Riyadi, K. Kunci, S. Pakar, P. Gigi, and W. Mobile, "SISTEM PAKAR DIAGNOSA PENYAKIT GIGI DAN MULUT MENGGUNAKAN METODE FORWARD CHAINING BERBASIS WEB," vol. 2018, no. Senadi, pp. 119–122, 2018.
- [15] S. Anhar, *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta Selatan: mediakita, 2010. [Online]. Available:

https://www.google.co.id/books/edition/PHP_MySql_Secara_Otodidak/J711efbP9LYC?hl=id&gbpv=1&dq=php&pg=PA3&printsec=frontcover

- [16] D. Maharani, “Perancangan Sistem Informasi Akademik Berbasis Web Pada Sekolah Islam Modern Amanah,” *J. Manaj. Inform. dan Tek. Komput.*, vol. 2, no. akademik berbasis web, pp. 27–32, 2017, [Online]. Available: <http://jurnatik.amikroyal.ac.id/index.php/amikroyal/article/view/37>
- [17] A. Sahi, “Aplikasi Test Potensi Akademik Seleksi Saringan Masuk Lp3I Berbasis Web Online Menggunakan Framework Codeigniter,” *Tematik*, vol. 7, no. 1, pp. 120–129, 2020, doi: 10.38204/tematik.v7i1.386.
- [18] T. Bin Tahir, M. Rais, and M. A. Hs, “Aplikasi Point OF Sales Menggunakan Framework Laravel Point OF Sales Appilaction using Laravel Framework,” *J. Inform. dan Komputer) p-ISSN*, vol. 2, no. 2, pp. 2355–7699, 2019, [Online]. Available: <http://dx.doi.org/10.33387/jiko>
- [19] A. J. G. Suharyadi, Tintien Koerniawati, *IMPLEMENTASI TEKNOLOGI WORDPRESS UNTUK PEMBANGUNAN WEBSITE*. Uwais Inspirasi Indonesia, 2023. [Online]. Available: https://www.google.co.id/books/edition/IMPLEMENTASI_TEKNOLOGI_WORDPRESS_UNTUK_P/xXGwEAAAQBAJ?hl=id&gbpv=0
- [20] Agustini and W. J. Kurniawan, “Sistem E-Learning Do’a dan Iqro’ dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas,” *J. Mhs. Apl. Teknol. Komput. dan Inf.*, vol. 1, no. 3, pp. 154–159, 2019, [Online]. Available: <http://www.ejournal.pelitaindonesia.ac.id/JMApTeKsi/index.php/JOM/article/view/526>
- [21] S. Syamsiah, “Perancangan Flowchart dan Pseudocode Pembelajaran Mengenal Angka dengan Animasi untuk Anak PAUD Rambutan,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 4, no. 1, p. 86, 2019, doi: 10.30998/string.v4i1.3623.

- [22] M. Muslihudin and Oktafianto, *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: CV ANDI OFFSET, 2016. [Online]. Available: https://www.google.co.id/books/edition/Analisis_dan_Perancangan_Sistem_Informas/2SU3DgAAQBAJ?hl=id&gbpv=1&dq=UML&printsec=frontcover
- [23] H. Fuad, A. Budiman, and D. Kurniasari, "Perancangan Sistem Informasi Pemesanan Paket Pernikahan Berbasis Web Study Kasus Di Wedding Organizer PJ Management," *J. Sisfotek Glob.*, vol. 8, no. 2, 2018, doi: 10.38101/sisfotek.v8i2.202.
- [24] D. Yulawati, S. Saleh, and . I., "Prototype Pengadaan Dan Distribusi Barang Pada Waralaba Fried Chicken dan Burger lampung," *SIMADA (Jurnal Sist. Inf. Manaj. Basis Data)*, vol. 1, no. 1, p. 61, 2018, doi: 10.30873/simada.v1i1.1115.
- [25] R. Taufiq and A. P. Sandi, "Penerapan Metode Forward Chaining Diagnosa Penyakit Infeksi Saluran Kemih Berbasis Web," *Jika*, vol. ISSN, no. September, pp. 2722–2713, 2021.

Lampiran 1

Data Pasien Pengidap Penyakit Ginjal

No RM	Nama	Jenis Kelamin	Umur
140-76-96	MAKMUN	L	54
135-56-74	SITI HADIJAH	P	19
148-50-99	PATIMAH	P	32
148-53-42	SUMARNO	L	67
143-28-10	HENNY HADIANY	P	37
147-28-33	HARIYATI	P	28
147-07-30	MARSELINUS BHIA	L	43
137-02-05	VERY LIANSYAH	L	33
141-20-55	MAKHYUDINN S	L	62
143-90-49	SAPIAH	P	45
148-35-43	SYAHRANI	L	44
146-41-43	JUANDA	L	55
148-46-37	ALIDIN	L	58
136-96-25	ARYS DARMAWAN	L	39
78-89-27	LAUNURU GAUS	L	70
147-49-00	BUDIANTO	L	40
148-02-03	ASMADI	L	62
112-68-37	ABDUL HADI	L	47
148-51-10	FRANS IDI YANTO	L	59
148-80-47	NASRULLAH	L	40
138-16-73	AHMAD FIKRI HIDAYAT	L	77
106-32-63	DIDIK HARIYANTO	L	57
148-14-34	RIDUAN	L	75
123-60-28	GISLIYANOR	P	50
148-45-26	M ZAINI	L	47
148-91-05	SARIYEM	P	55

147-97-85	HARIANTO	L	60
114-40-05	SUGIONO	L	28
149-01-09	M RIDHA AL FADHIL	L	12
144-66-11	MINDARJO	L	57
148-59-98	ATRAN	L	71
142-74-16	DONY AMBARA	L	37
149-04-46	HELMIAH	P	46
146-47-92	SYARIFUDIN	L	54
145-56-83	BAGUS SATRIYO	L	40
118-69-04	HERLINA	P	49
148-98-67	SRI SARIYANI	P	55
146-77-22	MAWARDAH	P	56
137-61-08	SAMSUNI	L	62
122-59-81	SRI MISNA	P	51
147-02-28	SITI MAESYAROH	P	65
119-81-56	TAUPIK RAHMAN	L	59
148-78-51	SANIAH	P	61
149-14-86	IWAN ARDIANSYAH	L	54
149-03-15	EFLIN KOYANSOW	L	59
148-64-92	SRI NORMILA WATY	P	63
141-10-20	IRNANIAH	P	59
144-40-83	NUR ALDA YULIANI	P	19
148-95-12	ZAKARIA	L	49
109-97-78	DINA AULIANI	P	20
140-22-92	ROSITA EDDY	P	66
142-90-17	SYAMSUL HADI	L	45
142-34-27	SALASIAH	P	54
148-36-22	NAOMI RENGGO UTAMI	P	59
149-26-80	BAHRUDIN	L	60
135-44-40	RUSDIAN	L	57

149-10-11	M TAUFIQ	L	26
149-23-65	ABDUSSALAM	L	43
138-49-38	MUNTAHA	L	63
123-19-58	MUHAIMIN	L	54
147-59-04	ABDURRAHMAN	L	49
149-16-53	M DAFII	L	24
124-29-28	FATHUR RIDANI	L	47
150-04-16	SURIANI	P	42
148-17-23	NUR RIZKIA FADILLAH	P	14
137-18-35	SYAMSUDDIN RAMLI	L	60
100-86-26	NOOR ARRSAYIS	L	21
149-26-33	AGUNG	L	40
140-34-62	M HUSAINNI	L	65
73-34-28	ENY ROSDIANA	P	54
150-04-94	AINUN	P	58
150-01-96	HARTATIK	P	66
149-16-52	TJATUR ARIEFianto	L	45
149-27-58	AHMAD GAZAL	L	66
148-56-70	NORHAYATI	P	67
148-04-89	NI MADE WIRATI	P	33
150-04-92	IBRAMSYAH	L	55
106-43-34	ERNAWATI	P	48
144-22-90	DRS.H MANSYUR I	L	60
72-43-74	SITI SUKAINAH	P	39
150-11-04	AI SYAH	P	44
148-44-85	IRMA	P	33
150-09-91	SUKITAH	P	61
96-28-85	M IKHSAN LOFTI S	L	64
150-05-53	RUSMAWATI	P	66
150-09-32	SITI AISAH	P	61

149-09-31	CHRISTINA SUYATMI	P	62
150-19-16	KASIRAH	P	71
112-89-07	SARIWATI	P	42
150-06-51	M FADLY	L	35
146-21-95	MISLAN	L	50
101-54-69	MARDINEM	P	58
150-24-86	AGUS SUDIYONO	L	56
148-29-79	M ANDRI	L	31
110-83-61	NURLIANA	P	73
150-26-00	ISHAQ	L	40
135-76-63	ANANG SAPRUN	L	66
150-11-75	SULEMAN	L	43
150-31-53	AKHMAD ROSANJI	L	54
96-75-43	ARLIANSYAH	L	41
150-34-00	BRENKIN GOHONG	L	71
150-32-31	M KHAIRI	L	50
150-41-41	VERDINANDUS QUIDOLAU	L	41
150-42-97	PUGUH YULianto	L	35
135-32-08	PANAHATAN SIAHAAN	L	60
150-47-40	HASANUDDIN	L	61
146-22-73	HAIRIAH	P	51
137-05-41	AHMAD MULIADI	L	12
119-88-53	HJ.MNURSIAH	P	75
150-50-55	PAIMAN	L	56
150-28-39	ENY SUPRAPTI	P	42
150-41-52	ALMAIDA	P	25
136-64-59	FAHRUDDIN	L	61
150-40-21	NORSEHAN	P	67
150-60-43	NANGIMAH	P	59

Lampiran 2

Data Riwayat Rekam Medik

nama	keluhan	riwayat penyakit	diagnosa awal	diagnosa utama
siti hadijah	badan terasa lemas pucat	diagnosa CKD, operasi CAPD	CKD anemia	CKD/CAPD
marshelinus bhia	badan terasa lemas HD rutin 1xseminggu	CKD stage V, sesak	CKD stage V on HD anemia nelaked CKD	CKD
Patimah	sesak nafas nyeri pinggul lemas	hipertensi	CKD stage V on HD anemia nelaked CK HF	CKD
Makmun	bab cair 3x hari (fases lendir, darah, hitam) badan terasa lemas nyeri perut muntah-muntah	DM (Diabetes Melitus)	anemia NN moderate DM tipe II CKD stage V DFU wagner 2	CKD

			pulmonary infection treatment	
Hariyati	badan terasa lemas penurunan kesadaran mual muntah tidak selera makan tidak buang air kecil sejak \pm 2bulan		general weakness CKD on HD	gangguan fungsi jaringan cerebral encephalopathy uremic CKD stage V on HD
Very Liansyah 6/jan/2022	perut membesar sesak nafas muntah berdarah pedih ulu hati		Subdit ADHF (gagal jantung kronik) HFrEF (gagal jantung disertai penurunan fraksi ejeksi)	covid 19 CKD stage V

Henny Hadiani	kepala terasa nyeri/sakit muntah	kolestrol DM Hipertensi	SUSP Sah	dandy walker perdarahan SOL
Sumarno	badan terasa lemas susah buang air kecil kaki bengkak dan luka	diabeter DM	acute on CKD DM tipe II HT	CKD stage V on HD
Makhyudin S	tidak nafsu makan badan terasa lemas		CKD stage V on HD rutin anemia nelaked on CKD	CKD stage V on HD
Siti Maesyaroh 8/mar/2022	sesak nafas ada luka di kaki	hipertensi diabetes melitus ulkus	HF CKD stage V on HD anemia DM tipe 2	DFU wagner CKD on HD HF

Budianto	batuk sesak nafas sudah 2 bulan demam kaki bengkak urine yang keluar sedikit		HF - ADHF HT	CKD HT CAD
Siti Maesyaroh 29/apr/2022	nyeri pada kaki kiri sering sesak nafas cuci darah 2x seminggu	CKD cuci darah	CKD stage V on HD DFU	CKD stage V on HD HT anemia DM
Abdul Hadi 20/jan/2022	sesak nafas batuk berdahak kemerahan	hipertensi	CKD stage V on CMD pneumonia	CKD stage V no HD
Alidin 12/jan/2022	badan terasa lemas HB rendah	HD rutin	anemia NN - trombosit melena (feses hitam)	anemia NN dt blood loss HT on treatment CKD stage V on HD leukositosis trombositosis

Arys Darmawan 11/jan/2022	bengkak di kedua paha perut bengkak nyeri diarea penis dan skrotum	B20 (infeksi HIV) CDK rutin HD 2xseminggu HT	CKD stage V on HD	HIV CKD HT
Launuru Gaus 18/jan/2022	gangguan pencernaan perut membesar	perut membesar sejak 6 bulan	asites qr III HT BAB hitam	Asites qr III
Gisliyanor 1/feb/2022	sesak nafas tidak nafsu makan mual	CKD on HD	anemia afeksi	CDK stage V on HD rutin HT
Sugiono 9/feb/2022	sesak nafas nyeri perut sakit dada mual		SOB CKD st V on HD	CKD stage V on HD rutin HF HT on treatment
Sariyem 16/mar/2022	muntah-muntah nyeri perut	maag	ACKD dd AKI nausea commiting / mual dan muntah	ACKD dd AKI leukimia

M Zaini 27/jan/2022	sesak nafas		SOB anemia renal hipertiroid	CKD st V on HD anemia severe
Gisliyanor 4/feb/2022	sesak nafas ± 1minggu demam tidak nafsu makan mual muntah	HD rutin 2x seminggu HT	CKD on HD	CKD st V on HD rutin
Arys Darmawan 10/feb/2022	badan terasa lemas	CKD st V on HD rutin	CKD on HD anemia related CKD hipertensi stg II	CKD stage V on HD HIV
Noor Arrasyid 26/feb/2022	dada dan kepala terasa sakit demam	CKD on HD 1x seminggu	CKD on HD	komplikasi ACKD
Mawardah 5/feb/2022	badan terasa lemas pucat HD rutin 1x seminggu	CKD stg V on HD anemia DM	CKD stage V on HD axilla	CKD stage V on HD axilla

Arys Darmawan 26/feb/2022	sesak nafas perut membesar mual nyeri perut	CKD stg V on HD rutin 2xseminggu	SOB CImproved efusi pleura bilateral CKD stg V on HD HIV ascites permagna hipertensi	CKD stg V on HD B20 (infeksi HIV) efusi pleura bilateral
M Ridha Al Fadhil 8/feb/2022	tidak dapat menahan kencing badan terasa lemas pucat nyeri perut muntah-muntah	pernah op. usus buntu	CKD stage IV	CKD stage IV anemia mikrositik neurogenic bladder
very Liansyah 21/feb/2022	sering pusing sesak nafas perut membengkak muntah darah fases hitam	HD rutin 2xseminggu	melena anemia CKD stg V on HD	CKD stg V on HD rutin

Asmadi 20/feb/2022	sesak nafas badan terasa lemas batuk-batuk demam	CKD stage V on HD rutin (1x/minggu)	CKD stage V on HD rutin (1x/minggu) HT on treatment HF st C, FC II anemia NN	CKD stage V on HD HF stage C + C II CAP
Asmadi 20/feb/2022	sesak nafas badan terasa lemas batuk-batuk	CKD stage V on HD rutin (1x/minggu) HT DM	CKD stage V on HD anemia related on CKD	CKD stg V on HD rutin 2x/minggu anemia renal HT CAP
Asmadi 15/01/2022	sesak nafas dada terasa nyeri	CKD on HD 1xseminggu	SOB CKD stage V on HD	CKD stage V on hemodialisis
Samsuni	batuk-batuk badan terasa lemas fases berwarna hitam pusing	CKD HT	melena HF CKD stg V on HD	CKD stg V on HD melena HT 2 treatment HF

	demam mual dan muntah			
Herlina 15/feb/2022	perdarahan cairan CAPD lemas nyeri pada perut saat masuk cairan	CKD	HT stage II peritonitis CAPD	CKD stage IV
Frans Idi Yanto 26/des/2021	nyeri pada bagian perut perut membesar batuk berdahak	CKD stg V on HD HT op. batu ginjal	CKD asates anemia related	CKD stg V on HD
Helmiah 26/mei/2022	gusi berdarah badan terasa lemas pucat batuk	CKD stg V on HD rutin 2xseminggu	CKD stg V on HD rutin 2xseminggu anemia related to renal problem OBS perdarahan pada gusi	CKD stg V on HD rutin HF

Nasrullah 17/jan/2022	sesak nafas parah		SOB dt Susp. Pneumonia acute superimposed on CKD dg komplikasi HF stC Fc IV hepatitis B kronik severe anemia non severe CAP	Acute Superimpose on CKD anemia
Mindarjo 24/jan/2022	sesak nafas demam	CKD stg V on HD rutin 2xseminggu	SOB et. ADHF febris HI CKD stg V on HD CAP severe	CKD stg V on HD rutin anemia NN HT on treatment
Dony Ambara 3/feb/2022	badan terasa lemas kaki bengkak	CKD operasi CAPD 4x	CKD on CAPD	CKD stg V on CAPD

Syahrani 6/jan/2022	muntah-muntah mual kepala terasa pusing	Stroke DM HT HD rutin 1x/minggu	nausea vomitus CKD stg V on HD rutin HT on treatment Hepatitis B Anemia	AMS Nausea Vomitus CKD stg V on HD anemia NN HT on treatment Hep B DM tipe II
Juanda	kaki membengkak	CAPD pro PET hipertensi DM	CKD stg V on CAPD HT DM	ESRD on CAPD pro PET (gagal ginjal kronik stadium 5)
Eflin Koyansow	nyeri ulu hati badan terasa lemas pusing mual muntah	Hipertensi	HT Anemia NN CKD	CKD HT

Sri Sariyani	sesak nafas sakit saat buang air kecil nyeri pada bagian dada	HT DM	HN (Hidronefrosis)/ pembengkakan pada satu atau kedua ginjal akibat penumpukan urine	CKD on HD
Didik Haroyanto	nyeri pada tangan kiri	CKD HD rutin	acute thrombosis CKD st V on HD	Acute Thrombosis CKD on HD
Saniah	badan terasa lemas pucat	Hipertensi Tumor Kandung Kemih Diabetes Melitus Jantung	CKD stage V on HD Anemia	CKD stg V on HD rutin Anemia Renal
Riduan	sesak nafas badan terasa lemas pucat hilang nafsu makan	CKD on Hd rutin Hipertensi	anemia renal CKD stg V on HD	CKD on HD
Hariato	nyeri pada bekas operasi	DM HT CKD on HD	DM +	CKD on HD DM tipe II

Iwan Ardiansyah 5/Mar/2022	sesak nafas nyeri dibagian bekas operasi	Hipertensi stenosis draining vein	stenosis draining vein	stenosis draining vein
Nur Alda Yuliani	sesak nafas perut membesar	rutin cuci darah 2xseminggu	asi+J50:L50tes permagna related HF dd Nefrogenik CKD stg V on HD 2xseminggu HF HT	CKD stg V
Bagus Satriyo 22/feb/2022	badan terasa lemas sesak nafas batuk	HD rutin 3xseminggu HT DM	CKD stg V on HD rutin anemia related CKD HT on treatment	CKD stg V on HD Anemia related CKD HT on treatment HF Obesitas
Sri Misna	pusing berkunang- kunang mual muntah	HD rutin 1xseminggu	anemia DM tipe II HT on treatment	Nausea vomitus Anemia NN CKD stg V on HD rutin

Asmadi 20/mar/2022	sesak nafas batuk berdarah demam mual muntah	HD rutin 2xseminggu CKD on HD	anemia NN SUB dt	CKD stg V on HD
Abdurrahman	operasi CDL	CKD hepatitis B infection HT stg II		CKD stg V on HD
Sariyem 25/jan/2022	muntah-muntah sesak nafas batuk mual lemas	maag	shock condition (perbaikan) ACKD dd AKI	CKD stg V on HD
Sariyem	mual muntah badan lemas penurunan nafsu makan		acute superimposed CKD leukositosis nausea vomiting hiponathermia	CKD stg V on HD

Panahatan Siahaan	sesak nafas batuk mual muntah	HT on HD rutin 2xseminggu	SUB dt HT emergency CKD stg V on HD	CKD stg V on HD DM tp II
Suriani	nyeri ulu hati pusing badan terasa lemas mual muntah nyeri pinggang sempat tidak sadarkan diri	batu ginjal	hiperkalemia CKD stg V newly dx	CKD stg V
Suriani	badan terasa lemas	HD rutin 2xseminggu	CKD stg V on HD Anemia	CKD stg V on HD anemia HT
Syamsul Hadi	nyeri pinggang		CKD on HD	CKD on HD

Zakaria	badan terasa lemas mual muntah terdapat benjolan di paha kiri	Hipertensi DM	anemia NN CKD on HD	CKD stg V on HD
Zakaria	badan terasa lemas	CKD HTC +	CKD on HD anemia	CKD stg V on HD anemia
Muhammad Taufiq	badan terasa lemas	CKD on HD 2Xseminggu	anemia dt renal	anemia renal CKD stg V on HD
Aisyah	sesak nafas badan bengkak mual muntah	DM tipe II	SUB dt acute superimposed on CKD anemia nausea vomite dt gastropati uremikum DM tipe II HF ascites	CKD DM Anemia HF Hiperkalemia

Syarifudin	badan terasa lemas sesak nafas mual muntah demam	DM tipe II nausea vomite	Sub dt anemia NN DM tipe II HF CKD stg V on HD	CKD stg V on HD
Atran	badan lemas mual muntah	DM tipe II HT Stroke	CKD stg V newly diagnose anemia related to renal disease Sequele stroke	CKD stage V newly diagnosed Anemia NN
Atran	batuk demam batuk sesak nafas	pernah melakukan cuci darah 3x	anemia NN CKD stg V Stroke Sequele	CKD stg V on HD
Ishaq	penurunan kesadaran	anemia	Septic Syok Acute Kidney Injury (AKI)	CKD stg V on HD Syok condition
Brenkin Gohong	nyeri perut lemas	batu ginjal	anemia NN CKD stg V newly	akut superimposed on CKD stg V

Dina Auliani 12/jun/2022	sesak nafas nyeri pada dada demam	HD rutin 2xseminggu	Atypical Chest Pain HF fc II Anemia rt CKD HT on tr CKD stg V	CKD stg V on HD
Kasirah	sesak nafas nyeri ulu hati mual muntah	HT DM	SOB dt Acute Superimposed on CKD HF	CKD stg V
Sariwati	nyeri pundak sesak nafas lemas		typical chest pain HT CKD stg V on HD	CKD stg V on HD
Suriani	sesak nafas demam mual muntah	HD rutin 2xseminggu	SOB HF CKD st V on HD anemia	CKD stg V on HD

Sapiah	lemas demam pusing	Hipertensi	Hipertensi Anemia nelaked CKD	CKD
Ahmad Fikri Hidayat	sesak nafas pedih ulu hati		Gagal Jantung Kronik Covid 19	CDK
Asmadi	sesak nafas nyeri dada batuk	HD rutin	CKD on HD	CKD
Helmiah	lemas nyeri perut mual-muntah pusing	CKD stg V on HD	Hipertensi CKD	CKD
Hariyati	urine berdarah lemas HD rutin sesak nafas pusing nyeri dada		CKD Hipertensi	CKD on HD

Taupik Rahman	sesak nafas demam lemas		hepatitis B anemia NN	Anemia nelaked CKD
Sri Normilawati	sesak nafas demam	CKD on HD rutin	CKD stg V on HD	CKD
Irnaniah	sesak tidak selera makan mual-muntah pusing	Stroke DM II HD rutin	CKD stg V on HD	CKD
Siti Hadijah	lemas kaki luka demam sesak nafas	DM II Hipertensi Ulkus	DM Anemia NN CKD stg V	DFU CKD on HD
Dina Auliani	demam mual-muntah HD rutin 2x seminggu	CKD	CKD stg V	CKD
Ahmad Fikri Hidayat	mual-muntah pusing sesak nafas	CKD stgV	CKD gagal ginjal	CKD on HD

Rosita Eddy	bab cair 3x hari (fases lendir, darah, hitam) badan terasa lemas nyeri perut muntah-muntah	Hipertensi	Melena Anemia NN	melena CKD on HD
Salasiah	penurunan kesadaran sesak nafas lemas pucat	HD rutin 2x seminggu	Anemia nelaked CKD hipertensi anemia	CKD
Naomi Renggo Utami	sesak pusing mual		Pneumonia CKD	CKD stg V
Bahrudin	batuk sesak nafas susah buang air kecil	hipertensi diabetes	Hipertensi DM II CKD	CKD on HD
Arys Darmawan	susah tidur penurunan kesadaran lemas	HD rutin	CKD on HD	CKD on HD hipertensi

Rusdian	sesak BAB berlendir lemas demam kepala pusing	Hipertensi	Hipertensi hepatitis B	CKD on HD
Bagus Satriyo	lemas pusing pucat batuk	HD rutin DM	CKD stg V on HD rutin HT anemia	CKD on HD rutin Anemia nelaked CKD HT on treatment
Abdussalam	gusi berdarah lemas pucat pusing mual-muntah	HD rutin	Anemia CKD	CKD on HD Hipertensi
Muntaha	lemas muntah nyeri kepala	kolestrol DM tipe II Hipertensi	Kolestrol Hipertensi	CKD stg V on HD

MUHAIMIN	lemas susah BAK Kaki luka	DM II	DM acute on CKD HT	CKD on Hd
M DAFI'	mual-muntah nyeri perut lemas susah BAK	Diabetes melitus	DM anemia CKD	CKD stg V on HD rutin
IWAN ARDIANSYAH	sesak nyeri pinggul demam pusing lemas	hipertensi	anemia hipertensi CKD	CKD on HD
FATHUR RIDANI	lemas HD rutin sesak	CKD on HD rutin	anemia CKD	CKD
NUR RIZKIA FADILLAH	lemas demam sesak kepala nyeri		anemia renal acute thrombosis CKD on HD	Acute Thrombosis CKD on HD

SYAMSUDDIN RAMLI	sesak nafas nyeri dada pucat tidak selera makan	Hipertensi	Anemia related CKD hipertensi on ckd	CKD on HD
NOOR ARRASYID	nyeri pada kepala demam lemas pucat	HD rutin	CKD on HD rutin	Komplikasi CKD stg V on Hd
AGUNG	sakit saat BAK nyeri kepala hd rutin 1x seminggu	HD rutin 1x seminggu	CKD on HD	CKD stg V on HD

Lampiran 3

Wawancara Dengan dr. Dewi Rizki., Sp.PD.

1. Apakah penyakit ginjal merupakan salah satu kategori penyakit yang sering dialami oleh masyarakat?

Jawaban: ya, akan tetapi kebanyakan dari penderita baru menyadari bahwa dirinya mengidap penyakit ginjal setelah terdiagnosa penyakit ginjal pada stage V, hal ini disebabkan rata-rata dari penyakit ginjal pada stage I hingga III jarang sekali merasakan gejala-gejala mengenai penyakit ginjal.

2. Apakah dari gejala-gejala yang dialami oleh penderita penyakit ginjal tersebut kita dapat menganalisanya agar mendapatkan diagnose dugaan awal dari penyakit ginjal tersebut?

Jawaban: biasanya penyakit ginjal yang mudah dideteksi seperti gagal ginjal kronik (CKD), sebab penderita penyakit ini sudah pasti memiliki riwayat penyakit kronik seperti diabetes melitus, hipertensi atau bahkan sering mengkonsumsi obat-obatan herbal yang banyak diperjual belikan bebas tanpa anjuran dokter. Selain itu, penyakit ginjal kronik (CKD) juga merupakan kumpulan dari penyakit-penyakit kronik, yaitu penyakit yang termasuk sulit untuk ditangani dan merupakan penyakit yang memerlukan pengobatan lanjutan.

3. Gejala apa yang dialami oleh calon pasien penyakit ginjal?

Jawaban: gejala umum yang sering dialami seperti sesak nafas, kaki bengkak, demam, mual, juga muntah.

4. Apa saja jenis penyakit ginjal pada umumnya?

Jawaban: jenis dari penyakit ginjal sendiri sangat banyak dan luas, beberapa yang dapat saya sebutkan dan tidak jarang banyak masyarakat awam yang mengetahuinya seperti batu ginjal, gagal ginjal kronik atau dalam bahasa medis sering disebut CKD. Ckd ini merupakan penyakit yang sering terdeteksi dikarenakan pasien sebelumnya memiliki riwayat penyakit-penyakit kronik.

Lampiran 4

Tabel Rule Penyakit

NO	KODE PENYAKIT	GEJALA	PENYAKIT	SOLUSI
1	P01	badan lemas, rutin hemodialisa, sesak nafas mual-muntah, penurunan kesadaran, tidak selera makan, nyeri kepala, urine berdarah, kulit kering gatal, sering buang air kecil, susah tidur	Gagal Ginjal Kronik	diet rendah garam, olahraga teratur, tidak merokok, konsumsi antibiotik, hemodialisa, transplantasi darah, mengkonsumsi air putih cukup, istirahat yang cukup
2	P02	badan lemas, sesak nafas, nyeri dada, kepala pusing	Hipertensi	diet rendah garam, olahraga teratur, tidak merokok, tidak mengkonsumsi alkohol, diet gizi seimbang, menjaga berat badan ideal
3	P03	badan lemas, nyeri kepala, nyeri kemaluan, kulit kering gatal, pembengkakan area mata, sering buang air kecil	Diabetes Melitus	olahraga teratur, tidak merokok, menjaga berat badan ideal, diet gula, diet kalori, mengkonsumsi

				serat cukup, mengkonsumsi air putih cukup
4	P04	badan lemas, sesak nafas, nyeri perut, mual-muntah, nyeri dada, gusi berdarah, kepala pusing	Anemia NN	perawatan luka, transfusi darah
5	P05	sesak nafas, tidak selera makan, nyeri dada, pembengkakan area mata, kram otot	Kolestrol	menjaga berat badan ideal, olahraga teratur, tidak merokok, tidak mengkonsumsi alkohol, diet gula, mengkonsumsi serat cukup, melakukan pemeriksaan berkala
6	P06	badan lemas, pembengkakan pada kaki, kaki luka	Diabetic Foot Ulcer	perawatan luka, mengkonsumsi air putih cukup
7	P07	sesak nafas, demam, batuk berdarah, nyeri dada, kram otot	Pneumonia	terapi oksigen, konsumsi obat pereda nyeri, istirahat yang cukup, vaksin pneumonia, fisioterapi

8	P08	sesak nafas, muntah berdarah, pembengkakan pada kaki, demam, pembengkakan pada perut, nyeri ulu hati	Gagal Jantung Kronik	melakukan pemeriksaan berkala, transplantasi jantung, olahraga teratur, tidak merokok, diet rendah garam, diet rendah lemak
9	P09	badan lemas, sesak nafas, pembengkakan pada kaki, pembengkakan pada paha, susah tidur	Gagal Ginjal	melakukan pemeriksaan berkala, olahraga teratur, tidak merokok, diet rendah garam, diet rendah lemak
10	P10	badan lemas, nyeri perut, mual-muntah, demam, hemoglobin rendah, fases hitam, kepala pusing	Melena	pemberian cairan intravena, melakukan tidak endoskopi, pemeriksaan saluran pencernaan atas
11	P11	nyeri perut, mual-muntah, perut membesar, gangguan pencernaan	Asites	diet rendah garam, menjaga berat badan ideal, melakukan pemeriksaan berkala, diet rendah protein
12	P12	badan lemas, tidak selera makan, perut membesar, demam, nyeri kemaluan, kram otot	HIV (Human Immunodeficiency Virus)	olahraga teratur, tidak merokok, melakukan pemeriksaan berkala, istirahat yang cukup, terapi antiretroviral (ART), terapi gizi dan fisik, penanganan infeksi

13	P13	sesak nafas, BAB berlendir, BAB cair, mual-muntah, tidak selera makan, demam, kepala pusing	Hepatitis B	olahraga teratur, tidak merokok, tidak mengonsumsi alkohol, melakukan pemeriksaan berkala, istirahat yang cukup, terapi antiviral, imunisasi hepatitis B, menjaga pola makan sehat
14	P14	mual-muntah, sakit buang air kecil, nyeri pinggul	Hidronefrosis	melakukan pemeriksaan berkala, menjaga pola makan sehat, pemasangan kateter, mengonsumsi air putih cukup
15	P15	badan lemas, mual-muntah, tidak selera makan, demam, hemoglobin rendah, kepala pusing, kulit kering gatal	Leukositosis	olahraga teratur, konsumsi antibiotik, melakukan pemeriksaan berkala, menjaga pola makan sehat, transfusi darah, terapi leukofereisis
16	P16	sesak nafas, penurunan kesadaran, susah buang air kecil, nyeri dada, nyeri pinggang	Hiperkalemia	hemodialisis, pemberian kalsium, pemberian insulin, pemberian glukosa, diet kalium

17	P17	badan lemas, pucat, sesak nafas, penurunan kesadaran, susah tidur	Anemia	olahraga teratur, menjaga pola makan sehat, mengkonsumsi air putih cukup, transfusi darah
----	------------	--	---------------	---

Lampiran 5

Contoh Perhitungan *Forward Chaining*

R1	= IF G01, G03, G04, G14, G15, G16, G07, G21, G32, G33, G34 THEN P1	
R2	= IF G01, G04, G09, G31	THEN P2
R3	= IF G01, G07, G08, G32, G26, G33	THEN P3
R4	= IF G05, G04, G06, G14, G09, G22, G31	THEN P4
R5	= IF G04, G16, G09, G26, G35	THEN P5
R6	= IF G01, G23, G36	THEN P6
R7	= IF G04, G27, G20, G09, G35	THEN P7
R8	= IF G04, G19, G23, G27, G25, G10	THEN P8
R9	= IF G01, G04, G23, G24, G34	THEN P9
R10	= IF G01, G06, G14, G27, G28, G29, G31	THEN P10
R11	= IF G06, G14, G18, G37	THEN P11
R12	= IF G01, G16, G18, G27, G08, G35	THEN P12
R13	= IF G04, G12, G13, G14, G16, G27, G31	THEN P13
R14	= IF G05, G14, G30	THEN P14
R15	= IF G01, G14, G16, G27, G28, G31, G32	THEN P15
R16	= IF G04, G15, G17, G09, G11	THEN P16
R17	= IF G01, G02, G04, G15, G34	THEN P17

Langkah *Forward Chaining*

- Gejala yang terdeteksi: **G11, G14, G23, G05, G29, G32**
- Hitung nilai persentasi peluang suatu kejadian!

Rumus: $P(A) = \frac{\text{Jumlah gejala}}{\text{Jumlah total gejala}} * 100$

- Rule yang memiliki gejala tereteksi:

R1 = IF G01, G03, G04, **G14**, G15, G16, G07, G21, **G32**, G33, G34 THEN P1

$$P(A) = \frac{2}{11} * 100 = 0,18$$

R3 = IF G01, G07, G08, **G32**, G26, G33 THEN P3

$$P(A) = \frac{1}{6} * 100 = 0,16$$

R4 = IF **G05**, G04, G06, **G14**, G09, G22, G31 THEN P4

$$P(A) = \frac{2}{7} * 100 = 0,28$$

R6 = IF G01, G23, G36 THEN P6

$$P(A) = \frac{1}{3} * 100 = 0,33$$

R8 = IF G04, G19, G23, G27, G25, G10 THEN P8

$$P(A) = \frac{1}{6} * 100 = 0,16$$

R9 = IF G01, G04, G23, G24, G34 THEN P9

$$P(A) = \frac{1}{5} * 100 = 0,2$$

R10 = IF G01, G06, G14, G27, G28, G29, G31 THEN P10

$$P(A) = \frac{2}{7} * 100 = 0,28$$

R11 = IF G06, G14, G18, G37 THEN P11

$$P(A) = \frac{1}{4} * 100 = 0,25$$

R13 = IF G04, G12, G13, G14, G16, G27, G31 THEN P13

$$P(A) = \frac{1}{7} * 100 = 0,14$$

R14 = IF G05, G14, G30 THEN P14

$$P(A) = \frac{2}{3} * 100 = 0,66$$

R15 = IF G01, G14, G16, G27, G28, G31, G32 THEN P15

$$P(A) = \frac{2}{7} * 100 = 0,28$$

R16 = IF G04, G15, G17, G09, G11 THEN P16

$$P(A) = \frac{1}{5} * 100 = 0,2$$

Hasil terbesar terdapat pada R14 sebesar 0,66, berdasarkan rule R14 = IF G05, G14, G30 THEN P14 maka pasien di diagnosa penyakit Hidronefrosis/pembengkakan.