

BAB 3

METODOLOGI PENELITIAN

3.1 Bahan dan Perangkat Penelitian

Perangkat lunak dan perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Laptop
2. Visual Studio Code
3. Database MySQL
4. Bahasa Pemrograman PHP (Laravel)

3.2 Subjek Penelitian

3.2.1 Kegiatan Penelitian

Kegiatan penelitian dilaksanakan dengan rincian sebagai berikut:

Judul Penelitian : Aplikasi Sistem Pakar Diagnosa Penyakit ISPA Berbasis Website Menggunakan Metode Certainty Factor

Lokasi Penelitian : Puskesmas Dukuh Kupang, Surabaya, Jawa Timur

Waktu Penelitian : Tahun Ajaran 2024

3.2.2 Profil Subjek Penelitian



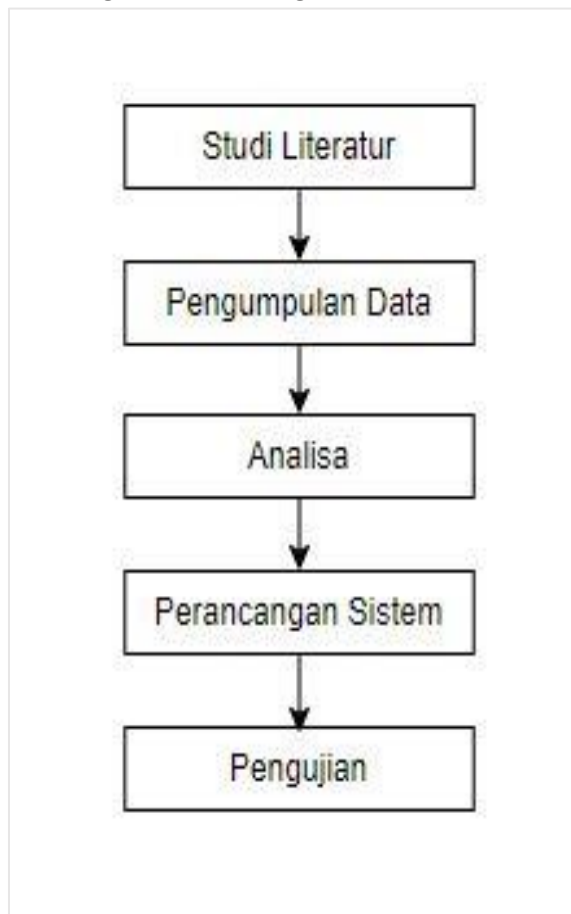
Gambar 3. 1 Puskesmas Dukuh Kupang

Puskesmas Dukuh Kupang Surabaya merupakan salah satu puskesmas yang terletak di Jalan Dukuh Kupang Timur XVII No. 1, Dukuh Kupang,

Surabaya. Puskesmas Dukuh Kupang merupakan pusat kesehatan masyarakat milik pemerintah yang beroperasi dibawah pengawasan Dinas Kesehatan Kota Surabaya yang memiliki peran cukup besar dalam menjalankan tugas dan wewenang pemerintahan daerah Kota Surabaya pada bidang kesehatan. Layanan yang disediakan meliputi perawatan darurat, rawat inap umum dan bersalin, perawatan lansia, keluarga berencana, kesehatan gigi dan mulut, layanan psikologis, konsultasi kesehatan lingkungan, konsultasi gizi, dan layanan farmasi. Puskesmas ini juga dilengkapi dengan fasilitas medis seperti ruang perawatan, ruang bersalin, dan ruang gawat darurat.

3.3 Tahapan Penelitian

Tahapan penelitian yang dilakukan penulis terdiri dari beberapa langkah yang harus dilakukan secara sistematis dan terstruktur. Tahapan penelitian dapat dilihat dari gambar 3.2 sebagai berikut.



Gambar 3. 2 Tahapan Penelitian

3.3.1 Studi Literatur

Studi literatur adalah jenis penelitian yang melibatkan, mencari dan menganalisis literatur yang sudah ada. Pemilihan sumber, penelusuran, pembacaan, dan pencatatan adalah proses studi literatur. Studi literatur membantu peneliti memanfaatkan informasi dan pemikiran yang relevan dengan topik penelitian ini.

3.3.2 Pengumpulan Data

Metode pengumpulan data yang dilakukan data yang dilakukan penulis adalah sebagai berikut :

1. Observasi : Observasi atau pengamatan dilakukan untuk mengetahui proses yang dilakukan untuk mendiagnosa penyakit di puskesmas Dukuh Kupang khususnya penyakit ISPA.
2. Wawancara : Melibatkan pertemuan dengan pihak terkait untuk bertukar informasi dan ide melalui tanya jawab, sehingga dapat dikonstruksikan makna dalam suatu topik tertentu
3. Data Sekunder : Pengumpulan data sekunder dilakukan untuk mendapatkan data pasien yaitu data laporan gejala pasien yang ada di Puskesmas.

3.3.3 Analisa

Pada tahap ini, penulis telah melakukan penelitian yang luas untuk mengumpulkan informasi terkait dengan topik penelitian. Dalam proses ini, penulis telah mencari jurnal, referensi lain, dan mengumpulkan data yang relevan. Dengan demikian, penulis telah memiliki gambaran yang jelas tentang kebutuhan yang akan digunakan dalam penelitian. Kebutuhan ini meliputi tools, bahasa pemrograman, metode, dan database yang akan digunakan dalam perancangan sistem.

Pertama, penulis telah menentukan tools yang akan digunakan dalam penelitian. Tools ini meliputi software, hardware, dan perangkat lunak yang diperlukan untuk mengembangkan sistem.

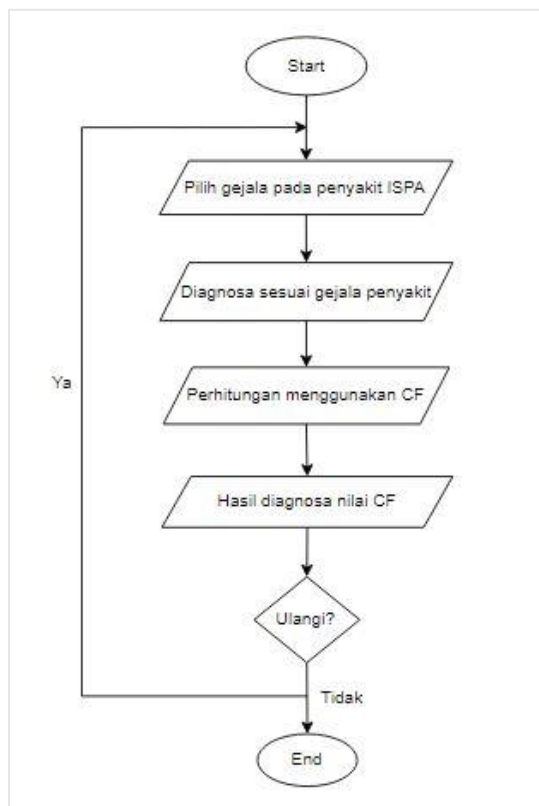
Selanjutnya, penulis telah menentukan bahasa pemrograman yang akan digunakan dalam penelitian. Bahasa pemrograman ini dipilih berdasarkan keterampilan penulis dan kebutuhan sistem yang akan dibangun.

Terakhir, penulis telah menentukan metode dan database yang akan digunakan dalam penelitian. Metode ini meliputi metode forward chaining, backward chaining, dan hybrid yang akan digunakan untuk mendiagnosa penyakit ISPA. Database yang akan digunakan adalah MySQL yang akan digunakan untuk menyimpan data dan aturan ke dalam basis pengetahuan sistem.

3.3.4 Perancangan Sistem

Setelah melakukan tahapan sebelumnya, yaitu analisis kebutuhan sistem, penulis telah membuat perancangan sistem untuk menggambarkan desain sistem yang akan dibuat. Perancangan sistem ini bertujuan untuk menggambarkan struktur dan fungsi sistem yang akan dibangun. Dalam penelitian ini, sistem yang dibuat menerapkan metode Certainty Factor (CF) yang digunakan untuk mendiagnosa penyakit ISPA.

Perancangan sistem ini meliputi beberapa komponen penting, seperti basis pengetahuan, engine, dan interface. Basis pengetahuan ini berisi data dan aturan yang digunakan untuk mendiagnosa penyakit ISPA. Engine ini berfungsi sebagai inti sistem yang melakukan proses mendiagnosa berdasarkan data dan aturan yang tersedia. Interface ini berfungsi sebagai antarmuka antara pengguna dengan sistem, memungkinkan pengguna untuk menginput data dan melihat hasil diagnosa. Dengan demikian, perancangan sistem ini dapat membantu dalam mengembangkan sistem yang efektif dan efisien dalam mendiagnosa penyakit ISPA



Gambar 3. 3 Perancangan Sistem

Adapun tabel gejala penyakit serta tabel penyakit pada ISPA bisa dilihat dari tabel 3.1 dan 3.2. Sedangkan untuk tabel matrik aturan gejala penyakit terhadap penyakit menunjukkan bahwa setiap penyakit memiliki gejala dan sebuah gejala terdapat pada beberapa penyakit, dapat dilihat pada tabel 3.3

Tabel 3. 1 Tabel Gejala

No.	Nama Gejala
1	Kesulitan bernafas
2	Suara yang serak
3	Suara batuk seperti menggonggong yang keras
4	Suara kasar bernada tinggi saat menghirup nafas
5	Mudah merasa Lelah
6	Tidak selera makan
7	Bernafas melalui mulut
8	Suara terdengar sengau
9	Munculnya lendir berwarna kuning/hijau dari hidung
10	Mengeluarkan bunyi saat bernafas
11	Muntah akibat rejan yang parah
12	Tubuh menjadi memerah dan membiru (sianosis)
13	Batuk berlangsung selama 5 hari atau lebih
14	Keluar dahak bening, kuning, hijau atau putih
15	Dada terasa sakit atau nyeri saat batuk
16	Demam
17	Hidung tersumbat
18	Terasa sakit pada bagian perut saat batuk
19	Tidak nafsu makan sehingga memicu dehidrasi
20	Keluar air liur (ngiler)
21	Nyeri tenggorokan
22	Gangguan menelan
23	Mengi, ditandai dengan suara seperti siulan ketika penderita

	bernapas
24	Batuk kering
25	Kulit yang membiru dan rasa sakit pada atau sekitar bagian tubuh yang kekurangan oksigen
26	Jeda antara nafas yang berlangsung 20 detik atau lebih
27	Nafas terengah-engah
28	Tersedak
29	Detak jantung lambat
30	Dyspnea (mungkin beraktivitas atau dengan istirahat, tergantung pada tingkat keparahan stenosis)
31	Pneumonitis berulang

Pada tabel gejala diatas menunjukkan apa saja gejala-gejala penyakit yang terdapat pada penyakit ISPA.

Tabel 3. 2 Tabel Penyakit

No.	ID Penyakit	Penyakit
1	P1	Croup
2	P2	Sinusitis Akut
3	P3	Pertusis (Batuk Rejan)
4	P4	Bronkitis
5	P5	Pneumonia
6	P6	Epiglottitis
7	P7	Bronchospasme
8	P8	Apnea Atiologies (Sleep Apnea/Apnea Tidur)
9	P9	Stenosis Subglotik

Tabel 3. 3 Tabel Relasi

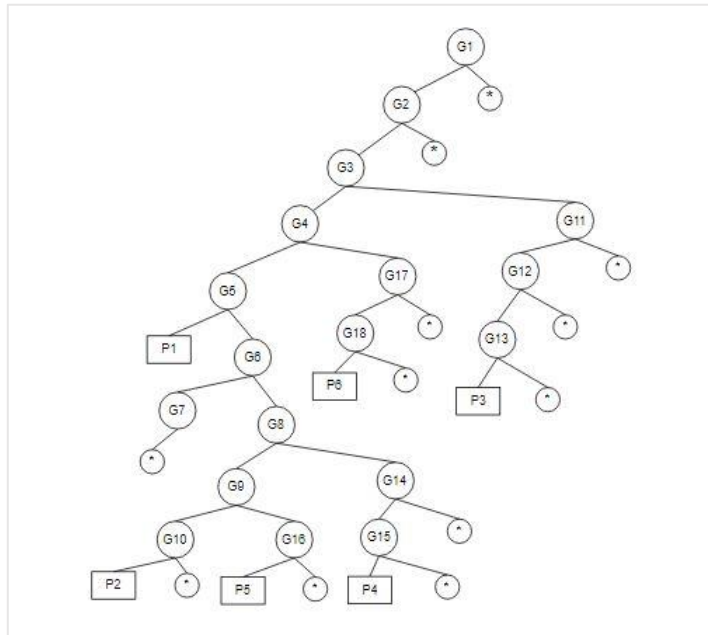
	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
P1	*	*	*	*																												
P2					x	x	x	x	x																							
P3	*										*	*	*			*																
P4													*	*	*	*																
P5	x											x				x	x	x	x													
P6	*		*								*				*				*	*	*											
P7	*													*								*	*	*								
P8												x														x	x	x	x	x		
P9	*		*									*	*																			*

Tabel 3. 4 Tabel Aturan Sistem Pakar

ID Relasi	ID Penyakit	ID Gejala	MB	MD
1	P1	G01	0.6	0.07
2	P1	G02	0.5	0.06
3	P1	G03	0.6	0.04
4	P1	G04	0.8	0.1
5	P2	G05	0.5	0.1
6	P2	G06	0.4	0.09
7	P2	G07	0.7	0.03
8	P2	G08	0.6	0.04
9	P2	G09	0.9	0.01
10	P3	G01	0.7	0.06
11	P3	G10	0.5	0.09
12	P3	G11	0.8	0.05
13	P3	G12	0.7	0.1
14	P3	G16	0.5	0.08
15	P4	G13	0.7	0.08

16	P4	G14	0.8	0.05
17	P4	G15	0.8	0.07
18	P4	G16	0.5	0.09
19	P5	G13	0.6	0.08
20	P5	G16	0.5	0.1
21	P5	G17	0.5	0.09
22	P5	G01	0.8	0.07
23	P5	G18	0.7	0.07
24	P5	G19	0.6	0.09
25	P6	G20	0.4	0.1
26	P6	G21	0.8	0.06
27	P6	G22	0.7	0.08
28	P6	G04	0.5	0.09
29	P6	G02	0.5	0.1
30	P6	G16	0.6	0.09
31	P6	G12	0.6	0.1
32	P7	G23	0.6	0.09
33	P7	G01	0.8	0.04

Berikut merupakan diagram pohon keputusan yang merupakan suatu rancangan yang digunakan untuk membangun suatu system pakar, di dalam diagram pohon keputusan tersebut akan dicari solusi hasil akhir dari setiap pemeriksaan. Diagram pohon keputusan dapat membantu dalam menyusun basis pengetahuan dan aturan serta menentukan tingkat kepastian dari setiap diagnosa gejala pada penyakit ISPA.



Gambar 3. 4 Pohon Keputusan

Keterangan :

G1-G18 = Gejala

P1 – P6 = Penyakit

(*) = Tidak terdiagnosa

Berdasarkan penilaian diagnosa sesuai gejala yang ada, maka dapat diperoleh data sebagai berikut:

Diketahui tingkat nilai keyakinan yang sudah ditentukan oleh seorang pakar, dimana pengguna diberi pilihan jawaban yang masing-masing bobotnya sebagai berikut.

Tabel 3. 5 Tabel Nilai Keyakinan

CF User	Keterangan
0	Tidak
0.2	Tidak Tahu
0.4	Sedikit Yakin
0.6	Cukup Yakin
0.8	Yakin
1	Sangat Yakin

Untuk mendapatkan nilai CF melakukan perhitungan sebagai berikut :

Tabel 3. 6 Tabel Perhitungan Algoritma CF

Gejala	Pilihan Anda	CF User	CF Role	CF(H,E)
Mudah merasa Lelah	Sedikit Yakin	0.4	0.4	0.16
Demam tinggi	Cukup Yakin	0.6	0.4	0.24
Batuk berdahak atau berdarah	Yakin	0.8	0.4	0.32
Bernafas dengan ritme cepat dan pendek	Yakin	0.8	0.6	0.48

Rule-rule yang baru tersebut kemudian dihitung nilai CF pakar/role dengan CF user menggunakan persamaan, dimana rumus untuk mendapat nilai CF(H,E) adalah sebagai berikut :

$$CF(H,E) = CF(E) * CF(Rule)$$

$$CF(H,E) = CF(User) * CF(Rule)$$

Langkah yang terakhir adalah mengkombinasikan nilai CF dari masing-masing rule. Kombinasikan CF1 sampai CF4 dengan persamaan, berikut adalah proses perhitungannya :

1. $CF(A) = CF(1) + CF(2) * [1 - CF(1)] = 0.16 + 0.24 * (1 - 0.16) = 0.3616$
2. $CF(B) = CF(3) + CF(A) * [1 - CF(3)] = 0.32 + 0.3616 * (1 - 0.32) = 0.565888$
3. $CF(C) = CF(4) + CF(B) * [1 - CF(4)] = 0.48 + 0.565888 * (1 - 0.48) = 0.77426176$

Dari perhitungan di atas, didapatkan nilai faktor kepastian (CF)yaitu 0.77426176 dan dari masukan gejala yang dialami, mengarah ke penyakit Pneumonia.

3.3.5 Pengujian

Pengujian sistem dilakukan dengan metode whitebox testing, istilah White Box merupakan salah satu cara untuk menguji suatu aplikasi atau software dengan cara melihat modul untuk dapat meneliti dan menganalisa kode dari

program yang dibuat. Singkatnya White Box ini menguji dengan cara melihat pure code dari suatu aplikasi atau software yang diuji tanpa memperdulikan tampilan atau UI dari aplikasi tersebut. Itulah sebabnya White Box ini berguna selama pengembangan aplikasi, bahkan jika itu dapat dilakukan selama beberapa fase kehidupan proyek. Metode ini dapat diterapkan untuk (terutama) tes unit, tes integrasi, dan tes sistem.

Pada pengujian ini, diagram alir flowgraph akan diterjemahkan berdasarkan alur source code yang telah dirancang sebelumnya untuk memudahkan identifikasi Region, Kompleksitas Siklomatis, dan Jalur Independen. Jika jumlah Region, Kompleksitas Siklomatis, dan Jalur Independen sama, maka sistem pakar dianggap benar. Namun, jika tidak, maka sistem masih memiliki kesalahan, mungkin terkait dengan logika atau aspek lainnya. Kompleksitas Siklomatis $V(G)$ untuk grafik alir dihitung menggunakan rumus berikut :

$$V(G) = E - N + 2$$

Dimana :

E = jumlah edge pada grafik alir

N = jumlah node pada grafik alir

```
resources > views > auth > login.blade.php
1  <x-guest-layout>
2  <x-slot name="title">
3  |   Login
4  </x-slot>
5
6  <x-auth-card>
7  |
8  |   {{-- show alert if there is errors --}}
9  |   <x-alert-error />
10 |
11 |   <x-slot name="title">
12 |   |   Login
13 |   </x-slot>
14 |   <form action="{{ route('login') }}" method="POST">
15 |       @csrf
16 |
17 |       <!-- Email field -->
18 |       <x-input type="text" text="Username" name="username" />
19 |
20 |       <!-- Password field -->
21 |       <x-input type="password" text="Password" name="password" />
22 |
23 |       <x-button type="primary btn-block" text="Login" for="submit" />
24 |   </form>
25 |   <div class="mt-4 text-center">
26 |       <hr>
27 |       <a href="{{ route('register') }}" class="text-primary">Create new account?</a>
28 |   </div>
29 |   </x-auth-card>
30 |
31 </x-guest-layout>
```

Gambar 3. 5 Kode Program Login

Pada source code di atas terdapat fungsi yang memungkinkan pengguna mengakses halaman utama sistem melalui browser, kemudian menuju ke menu form login. Pada form login, pengguna diharapkan mengisi username dan password yang telah terdaftar sebelumnya. Setelah pengguna memilih tombol

masuk, sistem akan memeriksa validitas username dan password yang diinputkan untuk menentukan proses selanjutnya, yaitu menuju ke menu dashboard.

Sedangkan pada source code di bawah ini terdapat fungsi yang memungkinkan pengguna melakukan diagnosa dengan sistem dari awal hingga pengguna meninggalkan sistem. Pada menu diagnosa, pengguna dapat berinteraksi dengan sistem atau melakukan pemilihan gejala sesuai dengan yang dirasakan hingga mencapai sebuah kesimpulan. Kesimpulan yang didapat adalah diagnosa gejala (penyakit ISPA). Setelah mendapatkan kesimpulan, pengguna dapat melakukan diagnosa ulang, mencetak hasil diagnosa, atau logout dari sistem.

```

{{-- chart section --}}


<x-alert-error></x-alert-error>
<div class="card">
<div class="card-body">
<form action="{{ route('admin.diagnosa') }}" method="post">
@csrf

@role('Admin')
<label for=""><b><i class="fas fa-user mr-1"></i> Nama</b></label>
<input type="text" class="form-control mb-3 w-50" name="nama">
@endrole

<p>Pilih gejala yang sedang dirasakan.</p>

<label for=""><b><i class="fas fa-th mr-1"></i> Gejala-gejala</b></label>
@foreach($gejala as $key => $value)
@php
$mod = ($key + 1) % 2;
@endphp

@if($mod == 1)
<div class="row">
@elseif
<div class="col-md-6">
<div class="d-flex align-items-center justify-content-between border mb-2 p-2">
<div>
<span class="ml-2">{{ $value->nama }}</span>
</div>
<div>
<select name="diagnosa[]" id="" class="form-control form-control-sm red-border">
<option value="{{ $value->id }}+1">Pasti tidak</option>
<option value="{{ $value->id }}+0.8">Hampir pasti tidak</option>
<option value="{{ $value->id }}+0.6">Kemungkinan besar tidak</option>
<option value="{{ $value->id }}+0.4">Mungkin tidak</option>
<option value="" selected>Tidak tahu</option>
<option value="{{ $value->id }}+0.4">Mungkin</option>
<option value="{{ $value->id }}+0.6">Sangat mungkin</option>
<option value="{{ $value->id }}+0.8">Hampir pasti</option>
<option value="{{ $value->id }}+1">Pasti</option>
</select>
</div>
</div>
</div>
</div>
@elseif($mod == 0)


```

Gambar 3. 6 Kode Program Dashboard/Diagnosa

Berdasarkan pengujian yang dilakukan untuk meneliti apakah sudah sesuai input dan output pada sistem yang ada maka dilakukan test case yang mana test casenya pada path 1 yaitu “Login user” hasil yang diharapkan user dapat login dan status validasinya “Valid”. Lalu pada path 2 yaitu “Login Admin” hasil yang diharapkan admin dapat login dan status validasinya “Valid”. Pada Test Case ke3 path 3 yaitu “Diagnosa” hasil yang diharapkan yaitu muncul hasil diagnosa dari gejala yang telah dimasukkan maka status validasinya “Valid”. Test case ke 4 yaitu “Cetak Hasil Diagnosa” hasil yang diharapkan memunculkan hasil rekap diagnosa penyakit ISPA.

Tabel 3. 7 Pengujian

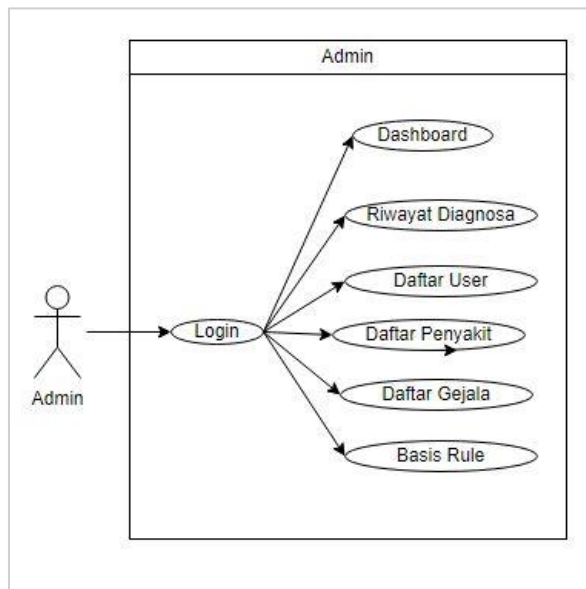
Akses	Fungsi	Output
Admin	Halaman login	v
	Fungsi data gejala	v
	Fungsi tambah gejala	v
	Fungsi edit gejala	v
	Fungsi hapus gejala	v
	Fungsi data penyakit	v
	Fungsi tambah penyakit	v
	Fungsi edit penyakit	v
	Fungsi hapus penyakit	v
	Fungsi data rule	v
	Fungsi tambah rule	v
	Fungsi edit rule	v
	Fungsi hapus rule	v
	Fungsi riwayat	v
	Halaman diagnosa	v
	Fungsi data user	v
	Fungsi tambah user	v
	Fungsi hapus user	v
	Halaman profil	v
	Fungsi logout	v
User	Halaman login	v
	Halaman register	v
	Halaman diagnose	v
	Halaman riwayat	v

3.4 Desain Sistem

3.3.5 Use Case Diagram

Pada use case diagram, admin memiliki hak akses keseluruhan dari sistem. Hak akses ini dimulai dari form login, di mana administrator memasukkan kredensialnya untuk memastikan identitasnya. Setelah login, administrator dapat mengakses dashboard sebagai halaman utama sistem. Dari dashboard, administrator dapat berinteraksi dengan sistem dalam berbagai cara. Mereka dapat melihat informasi sistem, mengelola data, serta melakukan perubahan konfigurasi.

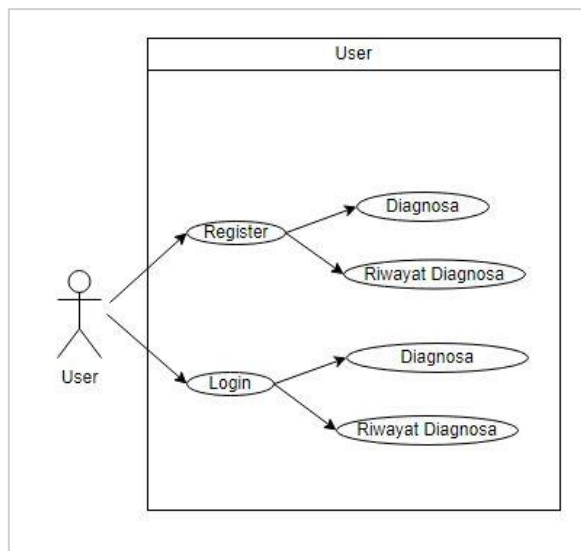
Interaksi ini dilakukan melalui berbagai fitur dan menu yang tersedia di dashboard, sehingga administrator dapat melakukan tugasnya dengan lebih efektif.



Gambar 3. 7 Use Case Diagram Admin

Pada use case diagram admin, administrator memiliki hak akses keseluruhan dari sistem. Dimulai dari form login kemudian ke dashboard. Administrator dapat berinteraksi dengan sistem seperti pada gambar 3.7 di atas. Administrator memiliki hak akses penuh dalam mengelola sistem pemesanan E-commerce seperti, kelola data pesanan, data user, data barang, dan data detail barang. Administrator dapat melakukan berbagai aktivitas seperti menambah, mengedit, dan menghapus data barang, serta mengelola seluruh form yang terdapat di halaman admin. Administrator juga dapat menghapus, mengupdate, dan menambah produk, serta mengkonfirmasi pesanan. Selain itu, administrator dapat mengelola seluruh form yang terdapat di halaman admin, termasuk form pemesanan, form pengguna, dan form produk. Administrator dapat mengakses seluruh fitur sistem dan melakukan perubahan yang diperlukan untuk memastikan sistem berjalan dengan baik.

Pada use case diagram user, pengguna atau user dapat berinteraksi dengan sistem dengan melakukan login. Login memungkinkan user untuk memasuki sistem dan memulai berbagai aktivitas. Selain itu, user juga dapat melakukan registrasi jika belum memiliki akun. Registrasi memungkinkan user untuk membuat akun dan memulai menggunakan sistem. Setelah login, user dapat melakukan diagnosa. Diagnosa memungkinkan user untuk memasukkan data dan informasi yang diperlukan untuk mendapatkan hasil diagnosa. Hasil diagnosa ini dapat membantu user dalam mengetahui penyebab masalah yang dialami dan cara mengatasi masalah tersebut.

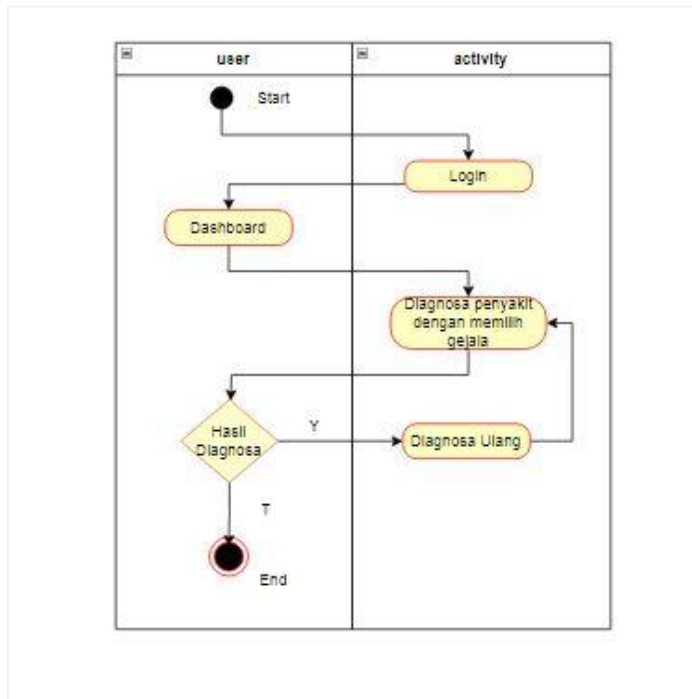


Gambar 3. 8 Use Case Diagram User

Selain melakukan diagnosa, user juga dapat melihat riwayat diagnosa yang telah dilakukan sebelumnya. Riwayat diagnosa ini dapat membantu user dalam melihat sejarah diagnosa yang telah dilakukan dan membandingkan hasil diagnosa yang telah diperoleh. Dengan demikian, user dapat lebih mudah dalam mengelola dan memantau riwayat diagnosa yang telah dilakukan.

3.3.5 Activity Diagram

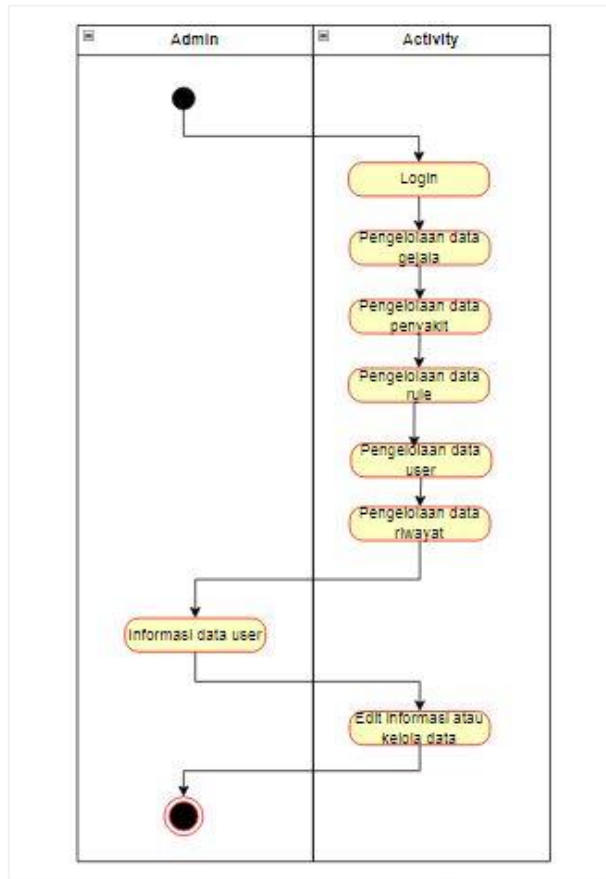
Activity diagram di bawah ini merupakan activity diagram user yang menunjukkan alur kegiatan pengguna dalam sistem. Proses ini dimulai dengan login, di mana pengguna memasukkan username dan password untuk memulai sesi. Setelah login, pengguna akan diarahkan ke tampilan dashboard yang menampilkan berbagai opsi untuk memulai proses diagnosa penyakit ISPA.



Gambar 3. 9 Activity Diagram User

Pada halaman dashboard, pengguna dapat memulai proses diagnose dengan mengakses fitur diagnose. Sistem akan menganalisis data yang diberikan dan menampilkan hasil diagnosa. Jika pengguna ingin mengulangi proses diagnose, mereka dapat kembali ke halaman diagnosa awal untuk memulai ulang. Jika pengguna telah selesai dengan proses diagnose, mereka dapat keluar dari website. Activity diagram ini menunjukkan bagaimana pengguna dapat menggunakan sistem secara efektif dan efisien, serta bagaimana sistem dapat memberikan hasil yang akurat dan bermanfaat untuk pengguna.

Pada gambar 3.10 di bawah, terlihat activity diagram admin yang dimulai dengan proses login. Setelah berhasil login, admin akan diarahkan ke halaman dashboard yang berfungsi sebagai pusat kontrol untuk mengelola berbagai data yang tersedia.



Gambar 3. 10 Activity Diagram Admin

Pada halaman dashboard admin, beberapa fungsi penting dapat dilakukan. Admin dapat mengelola data user, data penyakit, data gejala, data rule, serta riwayat diagnosa. Hal ini memungkinkan admin untuk memantau dan mengatur data yang relevan dengan aplikasi. Selain itu, pada halaman dashboard admin juga dapat melakukan proses pendagnosaan gejala penyakit ISPA. Proses ini mirip dengan yang dilakukan oleh user, namun dengan akses yang lebih luas dan lebih detail. Dengan demikian, admin dapat memantau dan mengelola data dengan lebih efektif dan efisien.

Halaman ini sengaja dikosongkan