

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **3.1 Analisis Sistem**

Analisis sistem merupakan suatu analisis yang terdiri dari suatu sistem informasi yang utuh kedalam suatu komponen dengan maksud untuk mengevaluasi permasalahan-permasalahan, kesempatan, hambatan dan kebutuhan yang diharapkan sehingga dapat diusulkan teori perbaikannya. Tahap ini bertujuan untuk mengetahui mekanisme aplikasi, proses-proses yang terlibat dalam aplikasi serta hubungan-hubungan proses. Ada tiga komponen atau perangkat yang dibutuhkan untuk membantu kinerja sistem agar tujuan dari sistem ini dapat tercapai. Perangkat tersebut adalah perangkat keras (*hardware*) berupa komputer, perangkat lunak (*software*) berupa program dan perangkat manusia (*brainware*) yakni manusia. Hasil dari sistem ini adalah sebuah aplikasi sistem pakar untuk mendiagnosa penyakit pada balita yang didalamnya terdapat informasi yang sangat berguna bagi *user* yang melakukan konsultasi.

##### **3.1.2 Analisis Kebutuhan Sistem**

Analisis kebutuhan sistem dilakukan untuk mengetahui fasilitas yang harus disediakan atau dimiliki oleh sistem agar dapat melayani kebutuhan pengguna sistem. Peran penting dari aplikasi ini adalah untuk membantu para masyarakat dan tenaga medis di Puskesmas Naibonat dalam proses penanganan penyakit pada balita agar lebih mudah dan cepat.

### 3.1.3 Analisis Peran Sistem

Peran dari sistem yang dibangun adalah sebagai berikut:

- a. Sistem yang dibangun dapat meng-*input* data-data *user*, data penyakit, data gejala, data relasi, data solusi, data informasi, saran dan data akun pakar.
- b. Sistem yang dibangun dapat merekam seluruh data-data yang dimasukkan. Semua data yang dimasukan direkam ke dalam *database* yang kemudian ditampilkan kembali ke tabel-tabel dalam *form*.
- c. Sistem yang dibangun dapat mencetak hasil diagnosa penyakit.

### 3.1.4 Analisis Peran Pengguna

Terdapat 3 pengguna dalam sistem ini antara lain :

#### 1. Admin

Pada sistem ini, *admin* memiliki fungsi untuk menginput data pakar, data informasi, menjaga keamanan *database* dari sistem, memperbarui *database* dan memberi batasan akses *user* sistem.

#### 2. Pakar

Pada sistem ini, pakar memiliki fungsi untuk menginput data gejala, data penyakit, informasi dan data relasi.

#### 3. User

Dalam sistem ini, sebelum berkonsultasi *user* harus melakukan registrasi data diri sehingga dapat melakukan diagnosa untuk mendapatkan hasil diagnosa penyakit.

### 3.2 Tabel Pengkodean Penyakit.

Tabel 3.1 Pengkodean Penyakit

No	Penyakit	Kode
1	Demam Berdarah Dengue (dbd)	P001
2	Malaria	P002
3	Cacar Air ( <i>varicella</i> )	P003
4	Campak ( <i>rubbela</i> )	P004
5	Diare	P005
6	Cacingan ( <i>helminthiasis</i> )	P006

### 3.3 Tabel Pengkodean Gejala.

Tabel 3.2 Pengkodean Gejala

No	Gejala	Kode
1	Sakit Kepala	G1
2	Demam	G2
3	Nyeri sendi	G3
4	Penurunan sel darah putih dan trombosit	G4
5	Mimisan atau pendarahan pada gusi	G5

6	Ruam atau bintik merah pada kulit	G6
7	Merasakan sensasi dingin dan menggigil	G7
8	Kelelahan dan terlihat pucat	G8
9	Bentol-bentol kecil kemerahan berisi air	G9
10	Gatal-gatal diseluruh tubuh	G10
11	Muncul bintik putih keabuan disekitar mulut dan tenggorokan	G11
12	Mata memerah dan berair	G12
13	Mata sensitif terhadap cahaya	G13
14	Kehilangan nafsu makan	G14
15	Feses lembek dan cair	G15
16	Terdapat darah pada feses	G16
17	Sakit perut	G17
18	Mual dan muntah	G18
19	Rasa buang air besar secara terus menerus	G19
20	Tampak cacing pada tinja	G20

### 3.4 Tabel Akuisisi Pengetahuan.

Akuisisi pengetahuan ialah mengimplementasikan pengetahuan yang sudah didapat dari dokter anak ke dalam sebuah tabel agar bisa menjadi basis pengetahuan dari sistem yang akan dibangun.

Tabel 3.3 Akuisisi Pengetahuan

No	Penyakit Gejala	P1	P2	P3	P4	P5	P6
1	Sakit Kepala	V	V	V	V		
2	Demam	V	V	V			
3	Nyeri sendi	V	V				
4	Penurunan sel darah putih dan trombosit	V					
5	Mimisan atau pendarahan pada gusi	V					
6	Ruam atau bintik merah pada kulit	V					
7	Merasakan sensasi dingin dan menggigil		V				
8	Kelelahan dan terlihat pucat		V				
9	Bentol-bentol kecil kemerahan berisi air			V			
10	Gatal-gatal diseluruh tubuh			V			
11	Muncul bintik putih keabuan disekitar mulut dan tenggorokan				V		

12	Mata memerah dan berair				V	
13	Mata sensitif terhadap cahaya				V	
14	Kehilangan nafsu makan				V	
15	Feses lembek dan cair				V	
16	Terdapat darah pada feses				V	V
17	Sakit perut				V	V
18	Mual dan muntah				V	V
19	Rasa buang air besar secara terus menerus				V	
20	Tampak cacing pada tinja					V

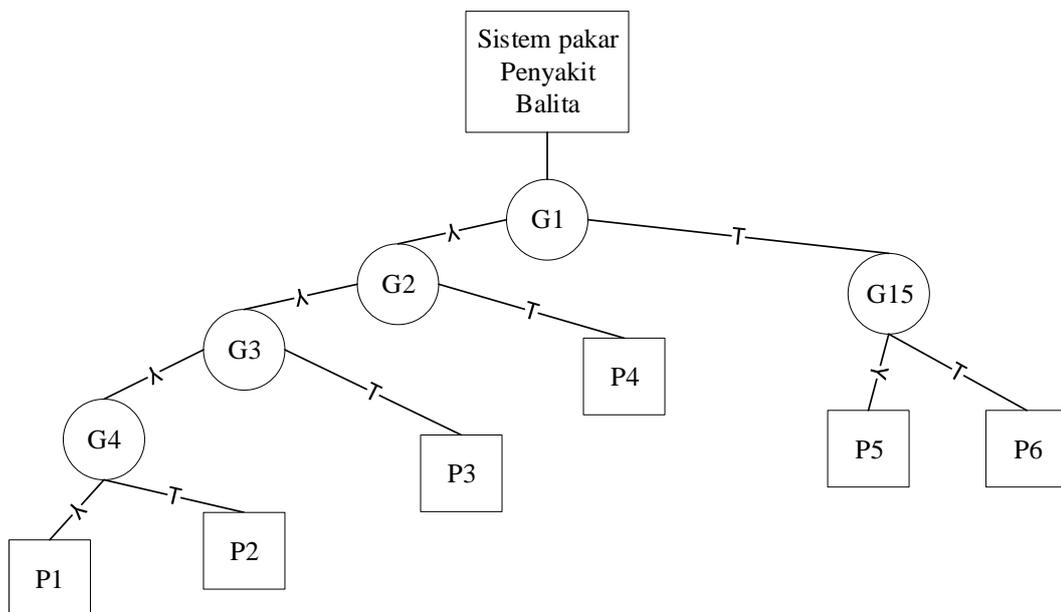
Berdasarkan tabel keputusan maka hasil inferensi pengetahuan yang dibuat dalam bentuk *IF-THEN* seperti berikut :

1. *IF* demam, sakit kepala, nyeri sendi, ruam atau bintik merah pada kulit, mimisan atau pendarahan ringan pada gusi, penurunan sel darah putih dan trombosit *THEN* Demam Berdarah Dengue (dbd).
2. *IF* demam, sakit kepala, nyeri sendi, merasakan sensasi dingin dan menggigil, kelelahan dan terlihat pucat *THEN* Malaria.
3. *IF* demam, sakit kepala, gatal-gatal diseluruh tubuh, terdapat bentol-bentol kecil kemerahan berisi air *THEN* Cacar air.

4. *IF* sakit kepala, mata memerah dan berair, kehilangan nafsu makan, mata sensitif terhadap cahaya, muncul bintik putih keabuan disekitar mulut dan tenggorokan *THEN* Campak (*rubella*).
5. *IF* sakit perut, mual dan muntah, terdapat darah pada feses, feses lembek dan cair, rasa buang air besar secara terus menerus *THEN* Diare.
6. *IF* sakit perut, mual dan muntah, terdapat darah dalam feses, tampak cacing pada tinja *THEN* cacingan (*helminthiasis*).

### 3.5 Pohon Keputusan

Pohon keputusan berfungsi memberikan keputusan pada basis pengetahuan yang sudah ada agar bisa diimplementasikan ke dalam sistem sesuai dengan keputusan yang sudah dibuat.



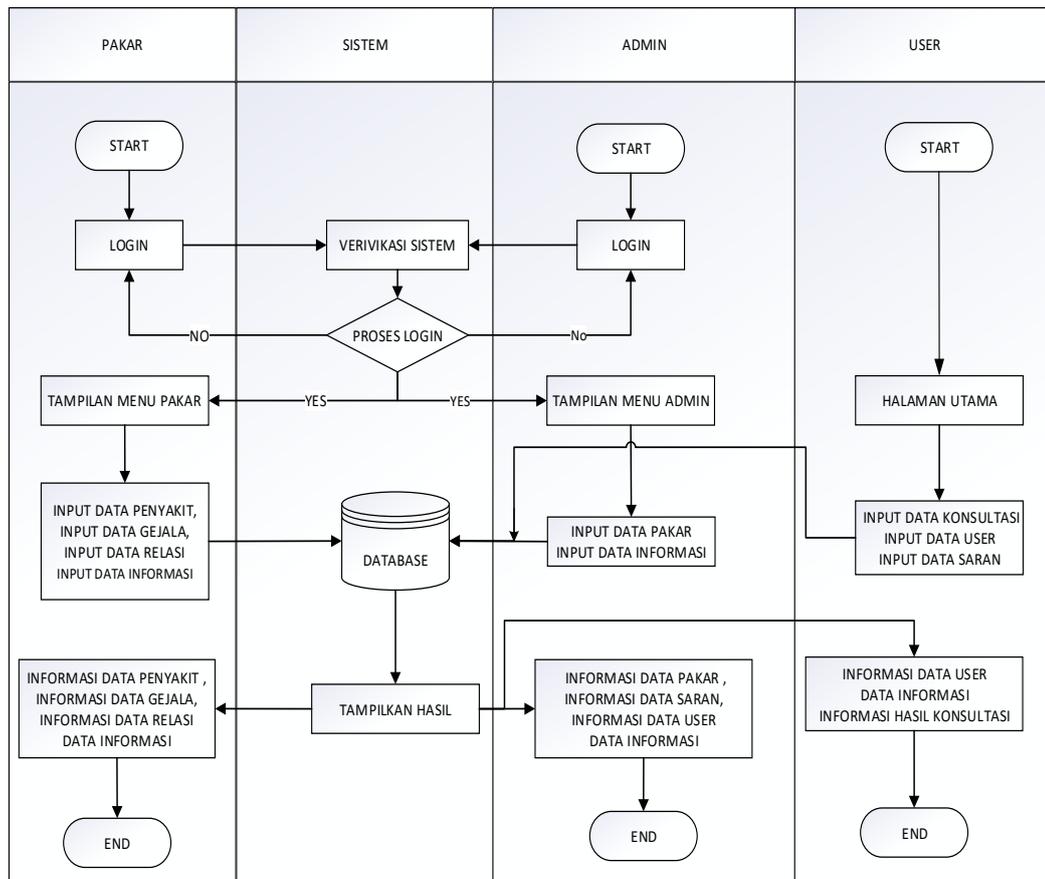
Gambar 3.1 Pohon Keputusan

### 3.6 Perancangan Sistem

Perancangan Sistem berfungsi untuk menentukan konfigurasi yang dibutuhkan oleh sistem dan metode yang digunakan dalam mengambil keputusan.

#### 3.6.2 *Flowchart* Sistem

Bagan alur dokumen (*Document Flowchart*) merupakan bagan yang menunjukkan arus penelitian secara keseluruhan dari sistem. Di bawah merupakan *flowchart* sistem pakar penyakit balita menggunakan metode *Forward Chaining* di mana sistem pakar penyakit balita harus menyediakan data-data mengenai penyakit, gejala, pengobatan dan aturan yang akan menjadi basis pengetahuan sistem pakar. Pengetahuan tersebut kemudian dimasukkan ke dalam sistem dan disimpan pada *database* sistem. Setelah fase ini selesai, *user* dapat menggunakan sistem dengan melakukan konsultasi. *user* tinggal memilih gejala-gejala yang dialami yang sudah di siapkan oleh sistem. Berdasarkan jawaban *user* tersebut sistem akan melakukan proses inferensi untuk menghasilkan konklusi yang merupakan diagnosis penyakit yang diderita, berdasarkan gejala pilihan *user* serta pencegahannya. Perancangan *flowchart* dari sistem pakar diagnosis penyakit balita terdapat pada gambar berikut:



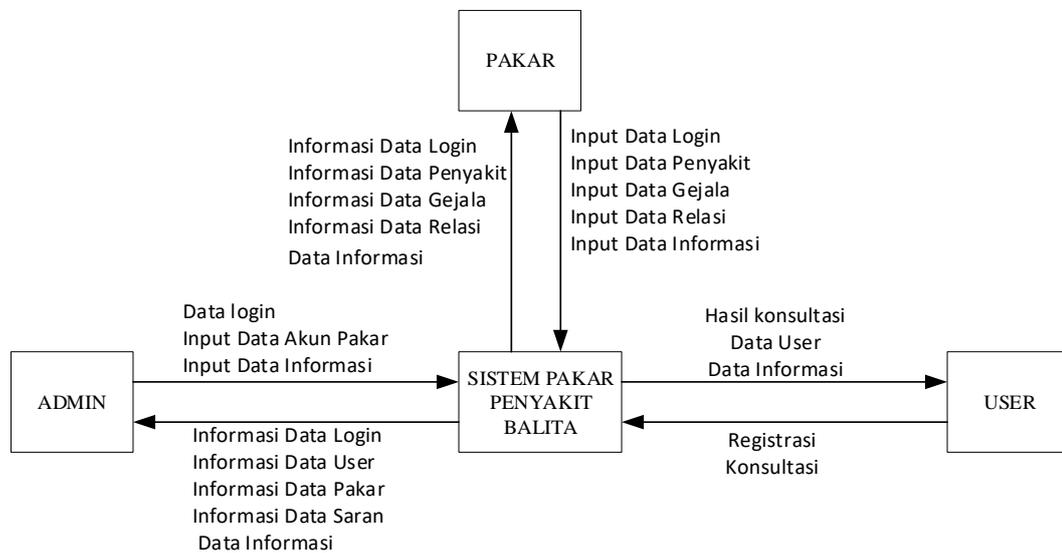
Gambar 3.2 *Flowchart* Sistem

Berdasarkan gambar *flowchart* pada Gambar 3.2 dapat dijelaskan sebagai berikut:

- a. *Admin* melakukan *login* dan *admin* bertugas menjaga keamanan dari sistem, meng-*input* data informasi, serta meng-*input* data pakar.
- b. Pakar melakukan *login* dan pakar akan meng-*input* data penyakit, data gejala, informasi dan membuat data relasi.
- c. Sebelum konsultasi *user* harus melakukan registrasi data diri. Setelah itu *user* akan menjawab pertanyaan berupa gejala yang dirasakan dan sistem akan mendiagnosa hasil dari gejala yang telah di pilih *user*. *User* juga dapat meng-*input* data saran.

### 3.6.3 Diagram Konteks (*Context Diagram*).

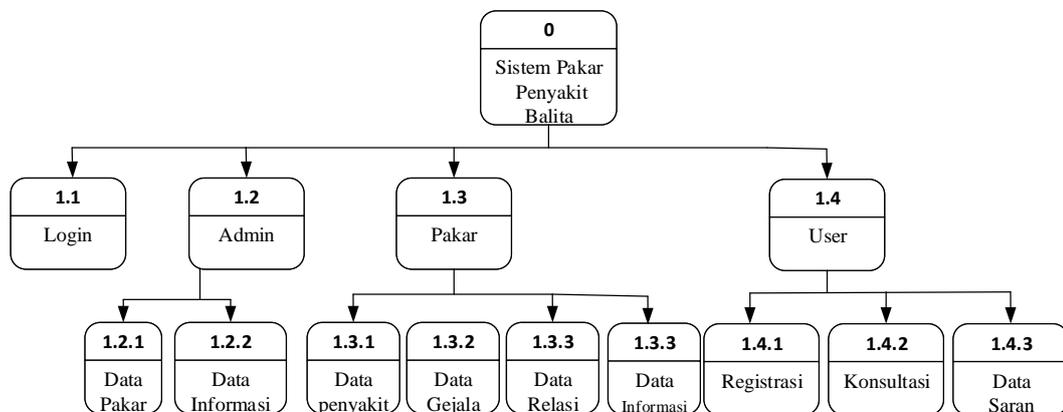
Diagram Konteks merupakan diagram yang menggambarkan hubungan sistem dengan lingkungan luarnya. Diagram konteks digunakan untuk menunjukkan secara garis besar hubungan dari *input*, proses dan *output*, dimana bagian *input* menunjukkan item-item data yang akan digunakan oleh bagian proses. Diagram konteks tersebut mewakili kegiatan seluruh sistem yang menggambarkan hubungan *input* atau *output* penggambaran diagram konteks yang digunakan pada sistem pakar untuk mendiagnosa penyakit balita. Diagram konteks dari sistem pakar penyakit balita dapat dilihat pada gambar berikut:



Gambar 3.3 Diagram Konteks.

### 3.6.4 Diagram Berjenjang (HIPO).

Diagram berjenjang (HIPO) merupakan penggambaran sistem secara hirarkis dengan menguraikan sistem ke dalam sub-proses pembentukkannya. Dalam sistem ini, pakar penyakit balita diuraikan ke dalam 2 level yaitu level 0 dan level 1. Proses diagram berjenjang dapat dilihat pada gambar berikut:



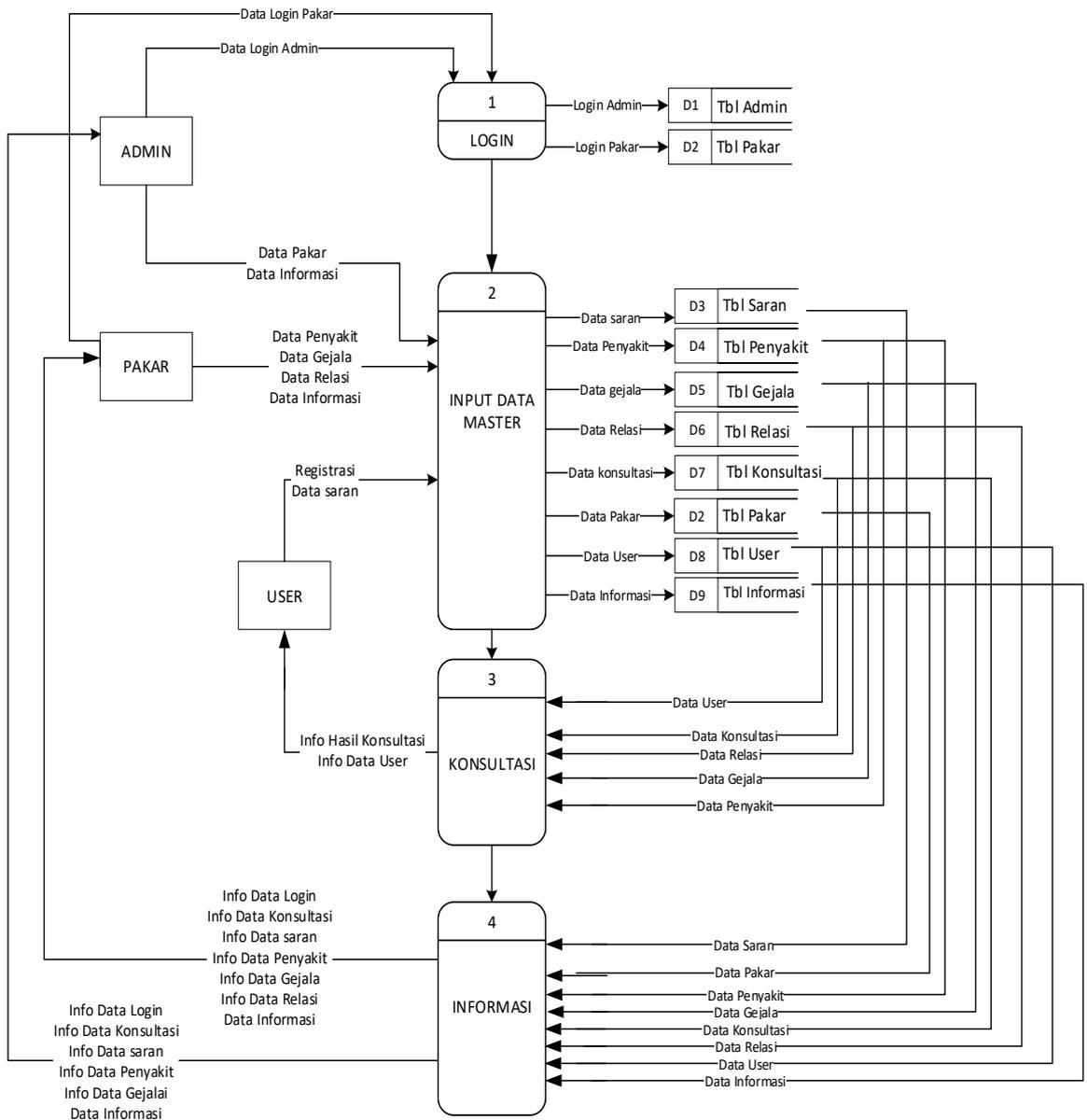
Gambar 3.4 Diagram Berjenjang (HIPO)

Berdasarkan gambar diagram berjenjang dapat dijelaskan sebagai berikut:

- a. *Admin* bertugas dalam meng-*input* data pakar dan data informasi.
- b. *Pakar* bertugas meng-*input* data penyakit, data gejala, informasi serta membuat relasi antara penyakit dan gejala.
- c. *User* melakukan registrasi kemudian langsung melakukan konsultasi dengan memilih gejala yang telah disiapkan oleh sistem dan juga meng-*input* data saran.

### 3.6.5 Data Flow Diagram (DFD) Level 1.

Diagram Arus Data (DAD) level 1 merupakan model logika atau proses yang dibuat untuk menggambarkan asal data dan tujuan data yang keluar dari sistem, tempat penyimpanan data, proses apa yang menghasilkan data tersebut serta interaksi antar data. Proses Diagram Arus Data (DAD) dapat dilihat pada gambar berikut:



Gambar 3.5 Data Flow Diagram (DFD) Level 1

Berdasarkan gambar diatas dapat dijelaskan sebagai berikut:

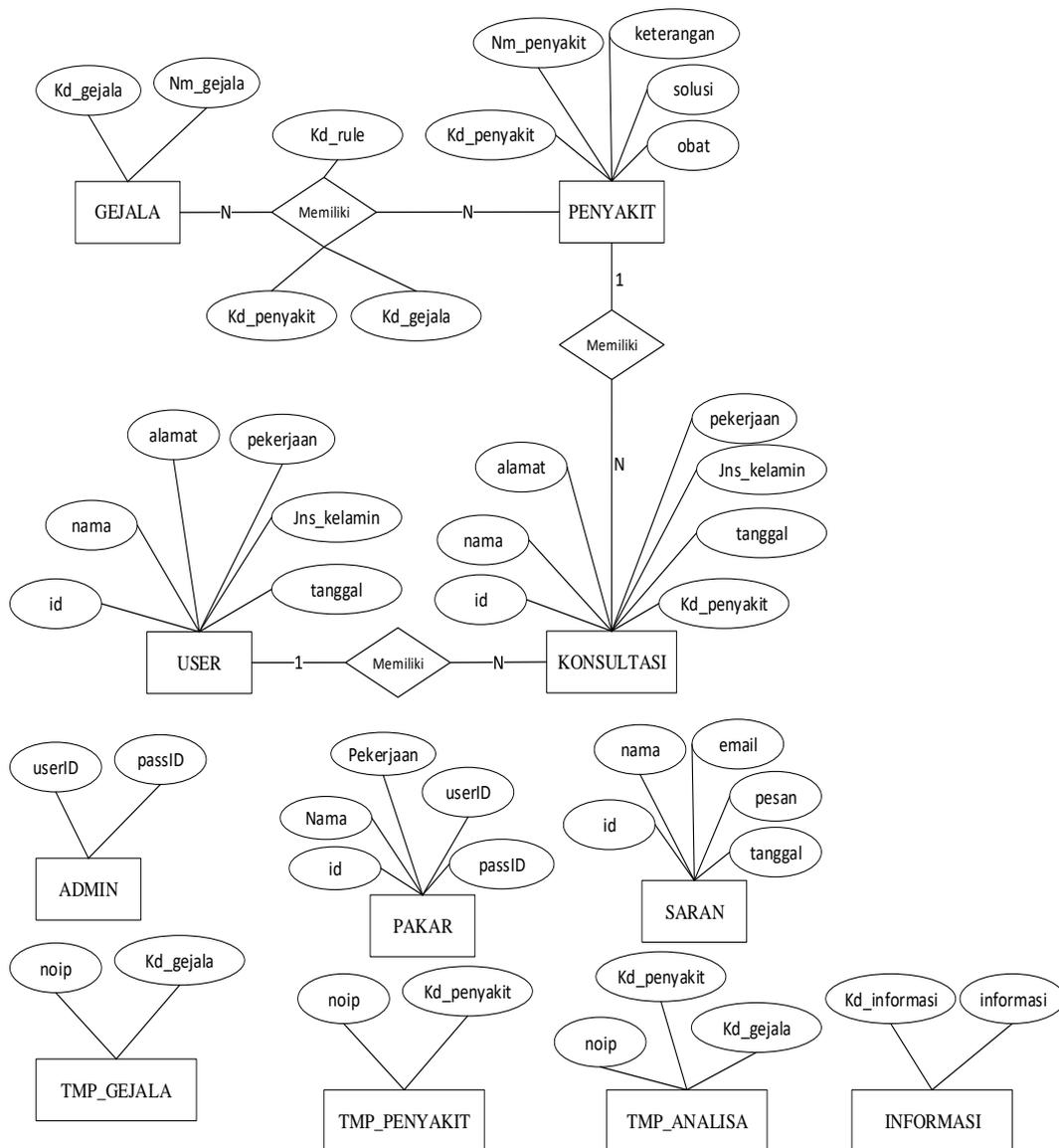
- a. *Admin* melakukan *login* kemudian meng-*input* data pakar, data informasi dan mencetak data hasil konsultasi yang dilakukan *user*.
- b. *Pakar*, melakukan *login* kemudian meng-*input* data gejala, data penyakit, dan membuat relasi dari penyakit dan gejala.
- c. *User*, melakukan registrasi sebelum melakukan konsultasi dan *user* akan menjawab pertanyaan berupa gejala dan sistem akan mendiagnosa hasil dari gejala yang telah di jawab oleh *user*.

### **3.7 Pemodelan Sistem.**

bagian ini akan menjelaskan *Entity Relationship Diagram* dan Relasinya.

#### **3.7.1 *Entity Relationship Diagram (ERD)***

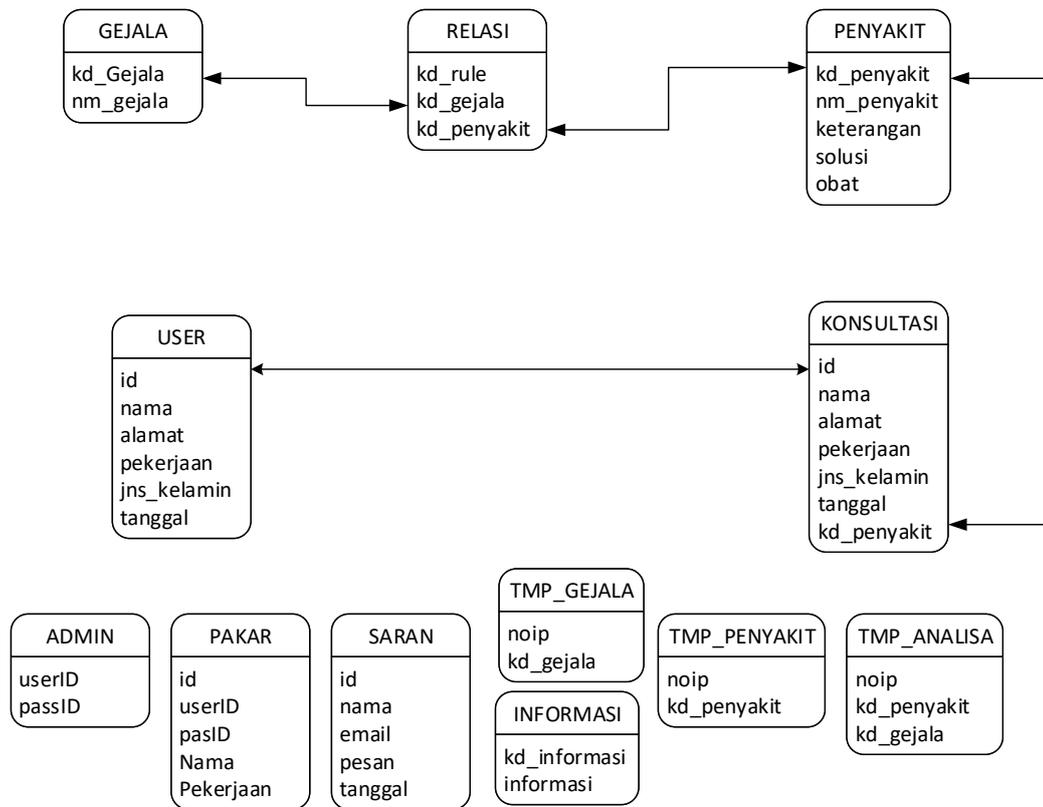
*Entity Relationship Diagram* berisi komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut yang mempresentasikan keseluruhan fakta. *ERD* menggambarkan hubungan data dalam basis data dengan menggunakan simbol-simbol dimana atribut dari suatu entitas mempunyai suatu hubungan atau relasi dengan atribut pada entitas yang lainnya.



Gambar 3.6 Entity Relationship Diagram (ERD)

### 3.7.2 Relasi Antar Tabel

Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu *database*.



Gambar 3.7 Relasi Antar Tabel.

### 3.8 Perancangan.

Pada tahap perancangan ini meliputi perancangan *database* dan perancangan antar muka.

#### 3.8.2 Perancangan *Database*.

Dalam merancang *database* pada suatu aplikasi perancangan sistem pakar penyakit balita menggunakan metode *forward chaining* dibutuhkan perancangan yang terdapat pada tabel-tabel berikut:

##### a. Tabel *Admin*.

Tabel *admin* digunakan untuk merekam data *admin*. Tabel *admin* memiliki 2 *field* dengan *Primary key*-nya adalah *UserID*.

Tabel 3.4 Admin.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>UserID</i>	<i>Varchar</i>	30	<i>Primary key</i>
<i>PassID</i>	<i>Varchar</i>	100	

b. Tabel Data Pakar.

Tabel data pakar digunakan untuk merekam data pakar. Tabel data pakar memiliki 5 *field* dengan *Primary key*-nya adalah *id*. Tabel data pakar dapat dilihat pada Tabel 3.5.

Tabel 3.5 Data Pakar.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Id</i>	<i>Int</i>	4	<i>Primary key</i>
<i>userID</i>	<i>Varchar</i>	30	
<i>PassID</i>	<i>Varchar</i>	100	
Nama	<i>Varchar</i>	50	
Pekerjaan	<i>Varchar</i>	50	

c. Tabel User.

Tabel *user* digunakan untuk merekam data *user*. Tabel *user* memiliki 6 *field* dengan *Primary key*-nya adalah *id*. Tabel *user* dapat dilihat pada Tabel 3.6.

Tabel 3.6 *User*.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Id</i>	<i>Int</i>	4	<i>Primary key</i>
Nama	<i>Varchar</i>	30	
Jns_kelamin	<i>Enum</i> ('laki-laki', 'perempuan')		
Alamat	<i>Varchar</i>	100	
Pekerjaan	<i>Varchar</i>	30	
Tanggal	<i>Datetime</i>		

d. Tabel Gejala.

Tabel gejala digunakan untuk merekam data gejala. Tabel gejala memiliki 2 *field* dengan *Primary key*-nya adalah *kd\_gejala*. Tabel gejala dapat dilihat pada Tabel 3.7

Tabel 3.7 *Gejala*.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Kd_gejala</i>	<i>Char</i>	4	<i>Primary key</i>
<i>Nm_gejala</i>	<i>Varchar</i>	100	

e. Tabel Konsultasi.

Tabel hasil konsultasi digunakan untuk merekam data hasil konsultasi. Tabel hasil konsultasi memiliki relasi dengan tabel penyakit. Tabel hasil konsultasi memiliki 7 *field* dengan *primary key*-nya *id*. Tabel konsultasi dapat dilihat pada Tabel 3.8.

Tabel 3.8 Konsultasi.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Id</i>	<i>Int</i>	4	<i>Primary key</i>
Nama	<i>Varchar</i>	30	
Alamat	<i>Varchar</i>	100	
Pekerjaan	<i>Varchar</i>	30	
Jns_kelamin	<i>Enum(laki-laki', 'perempuan')</i>		
Kd_penyakit	<i>Varchar</i>	4	<i>Foreign key</i>
Tanggal	<i>Datetime</i>		

f. Tabel Penyakit.

Tabel penyakit digunakan untuk merekam data penyakit. Tabel penyakit memiliki 5 *field* dengan *Primary key*-nya adalah *kd\_penyakit*.

Tabel penyakit dapat dilihat pada Tabel 3.9.

Tabel 3.9 Penyakit.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
Kd_penyakit	<i>Char</i>	4	<i>Primary key</i>
Nm_penyakit	<i>Varchar</i>	60	
Keterangan	<i>Text</i>		

Solusi	<i>Text</i>		
Obat	<i>Text</i>		

g. Tabel Relasi.

Tabel relasi digunakan untuk merekam data relasi antara penyakit dan gejala. Tabel relasi memiliki relasi dengan tabel penyakit dan tabel gejala. Memiliki 3 *field*. Tabel relasi\_penyakit\_gejala dapat dilihat pada Tabel 3.10.

Tabel 3.10 Relasi.

<i>Field</i>	<i>Type Data</i>	<i>Size</i>	<i>Index</i>
Kd_penyakit	<i>Varchar</i>	4	<i>Foreign key</i>
Kd_gejala	<i>Varchar</i>	4	<i>Foreign key</i>
Kd_rule	<i>Int</i>	4	<i>Primary key</i>

h. Tabel Saran.

Tabel saran digunakan untuk merekam data saran. Tabel saran memiliki 4 *field* dengan *primary key*-nya adalah *id*. Tabel saran dapat dilihat pada Tabel 3.11.

Tabel 3.11 Saran.

<i>Field</i>	<i>Type Data</i>	<i>Size</i>	<i>Index</i>
Id	<i>Int</i>	4	<i>primary key</i>
Nama	<i>Varchar</i>	50	
<i>Email</i>	<i>Text</i>		
Pesan	<i>Text</i>	-	
Tanggal	<i>Date</i>		

i. Tabel Informasi

Tabel informasi digunakan untuk merekam data informasi. Tabel Informasi memiliki 2 *field* dengan kd\_informasi sebagai *Primary key*.

Tabel 3.12 Informasi

<i>Field</i>	<i>Type Data</i>	<i>Size</i>	<i>Index</i>
Kd_informasi	<i>Int</i>	10	<i>primary key</i>
Informasi	<i>Text</i>		

j. Tabel Tmp\_analisa.

Tabel tmp\_analisa digunakan untuk merekam data tmp\_analisa. Tabel tmp\_analisa berelasi dengan tabel penyakit dan tabel gejala. Tabel tmp\_analisa memiliki 3 *field* dengan *primary key*-nya adalah *noip*.

Tabel 3.13 Tmp\_analisa.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Noip</i>	<i>Varchar</i>	30	<i>primary key</i>
<i>Kd_penyakit</i>	<i>Varchar</i>	4	<i>Foreign key</i>
<i>Kd_gejala</i>	<i>Varchar</i>	4	<i>Foreign key</i>

k. Tabel Tmp\_gejala.

Tabel tmp\_gejala digunakan untuk merekam data tmp\_gejala. Tabel tmp\_gejala berelasi dengan tabel gejala dan tabel tmp\_gejala memiliki 2 field dengan *primary key*-nya adalah *noip*.

Tabel 3.14 Tmp\_ gejala.

<i>Field</i>	<i>Tipe Data</i>	<i>Size</i>	<i>Index</i>
<i>Noip</i>	<i>Varchar</i>	20	<i>primary key</i>
<i>Kd_gejala</i>	<i>Varchar</i>	4	<i>Foreign key</i>

l. Tabel Tmp\_penyakit.

Tabel tmp\_penyakit digunakan untuk merekam data tmp\_penyakit. Tabel tmp\_penyakit berelasi dengan tabel penyakit dan tabel tmp\_penyakit memiliki 2 field dengan *primary key*-nya adalah *Noip*.  
Tabel tmp\_penyakit dapat dilihat pada Tabel 3.15

Tabel 3.15 Tmp\_ penyakit.

<i>Field</i>	<i>Type Data</i>	<i>Size</i>	<i>Index</i>
<i>Noip</i>	<i>Varchar</i>	20	<i>primary key</i>
<i>Kd_penyakit</i>	<i>Varchar</i>	4	<i>Foreign key</i>

### 3.8.3 Perancangan Antar Muka.

Perancangan antar muka bertujuan agar pengguna dan komputer dapat saling berinteraksi sehingga pengguna merasakan adanya kemudahan dari sistem komputer. Adapun desain *interface* dari sistem pakar mendiagnosa penyakit balita adalah sebagai berikut :

- a. Desain Halaman Utama.

HALAMAN UTAMA   KONSULTASI   DAFTAR PENYAKIT   SARAN   INFORMASI   LOGIN
SELAMAT DATANG SISTEM PAKAR PENYAKIT BALITA
Panduan XXX XXX XXX XXX
FOOTER

Gambar 3.8 Desain Halaman Utama

Halaman utama terdapat menu halaman utama, daftar penyakit, konsultasi, informasi, *login* dan saran. Menu ini berfungsi untuk menampilkan informasi sesuai dengan kegunaannya masing-masing dan *user* bisa memberikan saran kepada *admin* melalui menu saran yang sudah disediakan.

b. Desain Halaman Daftar Penyakit

BANNER ATAS	
DAFTAR PENYAKIT	
NAMA PENYAKIT	KETERANGAN
XXXXXXXXXXXX	<input type="button" value="Lihat"/>
FOOTER	

Gambar 3.9 Desain Halaman Daftar Penyakit

Halaman daftar penyakit berfungsi untuk memberi informasi bagi *user* tentang penyakit dalam sistem pakar penyakit balita, yang sudah disiapkan. Sehingga *user* bisa melihat daftar penyakit yang sudah disiapkan.

c. Desain Halaman Registrasi.

BANNER ATAS	
INPUTKAN DATA DIRI ANDA	
NAMA LENGKAP :	<input type="text"/>
ALAMAT :	<input type="text"/>
PEKERJAAN :	<input type="text"/>
JENIS KELAMIN :	<input type="radio"/> Laki-Laki <input type="radio"/> Perempuan
<input type="button" value="LANJUT"/> <input type="button" value="RESET"/>	
FOOTER	

Gambar 3.10 Desain Halaman Registrasi.

Halaman registrasi berfungsi untuk memberi informasi data diri *user* sehingga dapat berkonsultasi dan data *user* langsung tersimpan didalam *database*.

d. Desain Halaman Saran

<b>BANNER ATAS</b>	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Berikan Saran Anda</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Nama Anda</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Email Valid</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Pesan Anda</div> <div style="display: flex; justify-content: center; gap: 20px;"> <div style="border: 1px solid black; padding: 5px 15px;">KIRIM PESAN</div> <div style="border: 1px solid black; padding: 5px 15px;">BATAL</div> </div>	
<b>FOOTER</b>	

Gambar 3.11 Desain Halaman Saran

Halaman saran berfungsi agar bisa terjalin suatu komunikasi antara *user* dengan *admin* dan pakar. Dengan demikian *user* bisa langsung memberikan saran yang perlu disampaikan kepada *admin* dan pakar guna penyempurnaan aplikasi.

e. Desain Halaman Informasi

<b>BANNER ATAS</b>	
<div style="border: 1px solid black; padding: 5px; margin: 0 auto; width: 80%;">INFORMASI KESEHATAN BALITA</div>	
No	INFORMASI
1	XX XX XX XX
<b>FOOTER</b>	

Gambar 3.12 Desain Halaman Informasi

Halaman informasi berfungsi untuk informasi seputaran kesehatan balita kepada *user*.

f. Desain Halaman Konsultasi

BANNER ATAS
JAWABLAH PERTANYAAN BERIKUT Apakah Anak Anda Mengalami . . . ? <input type="button" value="Benar (Ya)"/> <input type="button" value="Salah (Tidak)"/> <input type="button" value="JAWAB"/>
GEJALA YANG TERPILIH YA (BENAR)
PENYAKIT YANG MUNGKIN MENYERANG ANAK ANDA
FOOTER

Gambar 3.13 Desain Halaman Konsultasi

Halaman menu konsultasi berfungsi untuk memulai proses konsultasi dengan cara menampilkan pertanyaan dan pilihan gejala pada *user*, dimana pilihan *user* nantinya akan mengarah kepada pertanyaan selanjutnya atau menghasilkan kesimpulan suatu penyakit. Setiap pilihan gejala *user* akan dibandingkan dengan data yang dimiliki oleh sistem.

g. Desain Halaman Hasil Konsultasi.

BANNER ATAS	
HASIL KEMUNGKINAN DIAGNOSA	
DATA PENGGUNA	
Nama :	
Alamat :	
Pekerjaan :	
Jenis kelamin :	
HASIL ANALISA	
Penyakit :	
Gejala :	
Keterangan :	
Solusi :	
Pengobatan :	<input type="button" value="CETAK"/>
FOOTER	

Gambar 3.14 Desain Halaman Hasil Konsultasi

Halaman menu hasil konsultasi berfungsi untuk menampilkan data hasil konsultasi berupa nama *user*, alamat *user*, pekerjaan, jenis kelamin, gejala yang di alami *user*, nama penyakit, solusi dan pengobatan.

h. Desain Halaman Pakar.

BANNER ATAS	
Beranda	AKUN PAKAR SELAMAT DATANG DIHALAMAN PAKAR
Daftar Penyakit	
Hasil Konsultasi	
Pesan	
Rule	
Penyakit	
Gejala	
Informasi	
Edit Akun	
Log Out	
FOOTER	

Gambar 3.15 Desain Halaman Pakar.

Halaman pakar berisi menu daftar penyakit, daftar hasil konsultasi, pesan, data rule, data gejala, data penyakit, informasi dan menu edit akun. Sebelum pakar *login* untuk masuk ke sistem, maka pakar harus terlebih dahulu mengisi *form login* bagi pakar. Menu ini berfungsi untuk pakar agar bisa mengontrol sistem sesuai dengan menu yang sudah disiapkan.

i. Desain Halaman *Admin*.

BANNER ATAS	
Beranda	AKUN ADMIN SELAMAT DATANG DIHALAMAN ADMIN
Daftar Penyakit	
Hasil Konsultasi	
Pesan	
Informasi	
Tamba Akun Pakar	
Edit Akun	
Log Out	
FOOTER	

Gambar 3.16 Desain Halaman *Admin*

Halaman *admin* berisi menu daftar penyakit, hasil konsultasi, pesan, informasi dan *admin* juga dapat menambah data akun pakar. Sebelum *admin login* untuk masuk ke sistem, maka *admin* harus terlebih dahulu mengisi *form login* bagi *admin*. Menu ini berfungsi untuk *admin* agar bisa mengontrol sistem sesuai dengan fungsinya masing-masing.