

**SISTEM PAKAR DIAGNOSA PENYAKIT KEJIWAAN
DENGAN METODE FORWARD CHAINING
BERBASIS WEB**

SKRIPSI

Diajukan sebagai salah satu syarat
untuk Menempuh Ujian Akhir Progam Sarjana (S1)
Program Studi Teknik Informatik STMIK – Indonesia Mandiri

Disusun Oleh :

Alvin Surya Ramadhan

361601013



**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
INDONESIA MANDIRI
BANDUNG
2021**

LEMBAR PENGESAHAN

**SISTEM PAKAR DIAGNOSA PENYAKIT KEJIWAAN DENGAN
METODE FORWARD CHAINING BERBASIS WEB**

**PSYCHIATRIC DISEASE DIAGNOSIS EXPERT SYSTEM
WITH FORWARD CHAINING METHOD
WEB-BASED**

Oleh :

ALVIN SURYA RAMADHAN

361601013

Tugas Akhir Ini Telah Diterima Dan Disahkan Untuk
Memenuhi Persyaratan Mencapai Gelar
Sarjana Teknik Informatika

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
INDONESIA MANDIRI**

Bandung, 2 Januari 2021
Disetujui Oleh

Ketua Program Studi

Dosen Pembimbing

Chalifa Chazar, S.T., M.T.
NIDN : 0421098704

Dr. Pahlawan Sagala
NIDN : 042206580

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa :

- 1) Tugas akhir ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik. Baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan tinggi lainnya.
- 2) Tugas akhir ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya pihak lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitas dan jenis bantuannya di dalam lembar ucapan terimakasih.
- 3) Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya seni saya ini, maka hal ini adalah diluar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Bandung, 2 Januari 2021
Yang Membuat Pernyataan

Alvin Surya Ramadhan
361601013

ABSTRAK

Gangguan kejiwaan merupakan suatu gangguan terhadap mood, pola pikir, hingga tingkah laku secara umum yang dapat menghambat pada kehidupan sosial. Tenaga profesional kejiwaan tidaklah sebanyak tenaga profesional kesehatan lain dapat dilihat dari minimnya rumah sakit kejiwaan. Pada penelitian ini, peneliti menganalisis apakah seseorang mengidap gangguan kejiwaan dari beberapa gejala umum gangguan kejiwaan. Tujuannya adalah membantu orang-orang dalam mengetahui informasi tentang gangguan kejiwaan beserta gejala, penanganan, dan pencegahannya. Penelitian ini menggunakan metode inversi *Forward Chaining* dan di aplikasikan berbasis web dengan menggunakan bahasa pemograman PHP (*Hypertext Preprocessor*), MySQL sebagai database. Hasil akhir dari penelitian ini adalah sebuah sistem pakar diagnosa gangguan kejiwaan.

Kata Kunci : Sistem Pakar, Gangguan Kejiwaan, *Forward Chaining*

ABSTRACT

Psychiatric disorders are a disorder of mood, mindset, and general behavior that can hinder social life. Psychiatric professionals are not as many as other health professionals can be seen from the lack of psychiatric hospitals. In this study, researchers analyzed whether a person has a psychiatric disorder from some common symptoms of psychiatric disorders. The goal is to help people with information about psychiatric disorders and their symptoms, treatment and prevention. This research uses forward chaining inversion method and in web-based application using programming language PHP (Hypertext Preprocessor), MySQL as a database. The end result of this study is a system of experts diagnosing psychiatric disorders.

Keywords : *Expert System, Psychiatric Disorders, Forward Chaining*

UCAPAN TERIMA KASIH

Dengan mengucapkan syukur Alhamdulillah, penelitian ini dapat diselesaikan untuk memenuhi syarat tugas akhir. Laporan penelitian dalam tugas akhir ini diajukan untuk memenuhi dan melengkapi salah satu syarat akademik dalam kelulusan jenjang Strata Satu (S1) jurusan Teknik Informatika pada Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Penyusunan tugas akhir ini tidak lepas dari dukungan dan bimbingan dari berbagai pihak, maka pada kesempatan ini penulis ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada :

1. Bapak Pahlawan Sagala, Dr, . selaku Dosen pembimbing yang selalu meluangkan waktu, pikiran dan tenaga dalam memberikan bimbingan, masukan dan saran-sarannya.
2. Bapak Dr. Chairudin, M.T., M.M. selaku Ketua Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri (STMIK-IM).
3. Ibu Chalifa Chazar, S.T., M.T. selaku Ketua program studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri (STMIK-IM).
4. Seluruh Dosen, Staff dan Karyawan Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri (STMIK-IM) yang telah mendidik dan membantu dalam memberika informasi serat motivasi dalam proses studi maupun tugas akhir berlangsung.

5. Teruntuk Kedua Orang Tua Tercinta Bapak Aryanto dan Ibu Sulissyana yang sangat penulis sayangi dan cintai. Terimakasih selalu memberikan nasehat, dukungan, didikan, kasih sayang, serta Do'a yang penuh dan tulus.
6. Teruntuk kakak-kakak dan adik penulis, kakek, nenek, paman, bibi, dan serta keluarga besar penulis, yang sangat penulis sayangi dan cintai, terimakasih selalu telah mendukung, dan mendo'akan penulis selalu.
7. Teruntuk sepupu yang selalu memberikan Do'a dan saran serta semangat dalam penyusunan laporan penelitian tugas akhir ini.
8. Sahabat-sahabat penulis yaitu Rezki Dwi, Adnan Hafizh, Irsan Hamdani, Muhammad Ramdha Desapta, Aswin Winangga, Bobby Rahman, Ananda Giri Sukmawijaya yang sama-sama berjuang untuk menyelesaikan laporan penelitian tugas akhir ini.
9. Sahabat-sahabat penulis yaitu, Aldo Surya Ramadhan, Regista Handarina, Angga, Egi Tianto, Leo Apriansyah, Astri Widayanti, Belsi, Adi Kurniawan, dan Iqban Aditia yang telah memberikan Do'a serta semangat dalam penyusunan laporan penelitian tugas akhir ini.
10. Seluruh rekan STMIK-IM angkatan 2016 khususnya pada program studi Teknik Informatika yang sama-sama berjuang untuk terus meraih impian, yang saling membantu juga memberikan semangat, dan seringkali menjadi tempat sharing, baik yang berhubungan dengan materi perkuliahan maupun lainnya.
11. Seluruh rekan sahabat, dan pihak-pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu Penulis baik secara langsung maupun tidak

langsung memberikan semangat kepada Penulis dalam menyelesaikan laporan penelitian tugas akhir.

Penulis menyadari bahwa masih banyak kekurangan yang mendasar pada laporan penelitian tugas akhir ini. Oleh karena itu penulis mengundang pembaca untuk memberikan saran serta kritik yang dapat membangun penulis. Penulis berharap adanya kritik konstruktif dan saran yang membangun dari semua pihak.

Akhir kata saya, berharap semoga dengan selesainya laporan penelitian Tugas Akhir ini dapat memberikan manfaat bagi semua pihak serta menambah wawasan bagi pemikiran kita semua. Terimakasih.

KATA PENGANTAR

Dengan mengucapkan Alhamdulillah sebagai wujud syukur ke Hadirat Allah SWT, yang senantiasa memberikan rahmat dan karunia-NYA sehingga penulis dapat menyelesaikan laporan penelitian tugas akhir ini dengan baik dan tepat pada waktunya.

Tugas akhir ini, berjudul *SISTEM PAKAR DIAGNOSA PENYAKIT KEJIWAAN DENGAN METODE FORWARD CHAINING BERBASIS WEB* , disusun untuk melengkapi tahapan akhir studi yang dijalani di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Tugas akhir ini berisi mengenai perancangan sebuah aplikasi sistem pakar yang dapat menghasilkan diagnosa penyakit kejiwaan dengan memilih gejala penyakit, yang meliputi analisis dari sistem yang diusulkan dengan harapan dapat mengatasi masalah yang diidentifikasi.

Dengan segala keterbatasan tentunya diharapkan aplikasi ini dapat bermanfaat bagi berbagai pihak, khususnya bagi penulis sendiri.

Bandung , 2 Januari 2021

Penulis

Alvin Surya Ramadhan
361601013

DAFTAR ISI

LEMBAR PENGESAHAN	i
SURAT PERNYATAAN.....	ii
ABSTRAK	iii
<i>ABSTRACT</i>	iv
UCAPAN TERIMA KASIH.....	v
KATA PENGANTAR	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xv
DAFTAR GAMBAR	xviii
BAB I.....	1
1.1. Latar Belakang	1
1.2. Identifikasi Masalah	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah.....	3
1.5. Metode Penelitian.....	4
1.5.1. Metode Pengumpulan Data	4
1.5.2. Metode Pengembangan Sistem	5

1.6. Penelitian Sebelumnya	7
1.6.1. Diagnosis penyakit kejiwaan menggunakan metode certainty factor	7
1.6.2. Aplikasi probabilitas bayes dalam sistem pakar untuk mendiagnosis gangguan kejiwaan bipolar	7
1.7. Sistematika Penulisan	8
BAB II.....	10
LANDASAN TEORI.....	10
2.1. Sistem.....	10
2.1.1. Pengertian Sistem.....	10
2.1.2. Karakteristik Sistem.....	10
2.1.3. Klasifikasi Sistem	13
2.2. Basis Data (Database)	15
2.2.1. Pengertian Basis Data	15
2.2.2. Database Management System (DBMS)	15
2.2.3. Fasilitas Database Management System (DBMS)	15
2.2.4. Komponen Utama <i>Database Management System</i> (DBMS).....	16
2.2.5. Keuntungan & Kerugian <i>Database Management System</i> (DBMS)....	17
2.3. Pakar.....	19
2.3.1. Pengertian Pakar.....	19
2.3.2. Ciri-Ciri Pakar.....	20

2.4 Sistem Pakar.....	20
2.4.1. Pengertian Sistem Pakar.....	20
2.4.2. Konsep Dasar Sistem Pakar	21
2.4.3. Bentuk Sistem Pakar	23
2.4.4. Struktur Sistem Pakar.....	23
2.4.5. Antarmuka Pengguna (<i>User Interface</i>)	24
2.4.6. Basis Pengetahuan (<i>Knowledge Base</i>)	25
2.4.7. Akuisisi Pengetahuan (<i>Knowledge Acquisition</i>)	25
2.4.8. Mesin Inferensi (<i>Inference Engine</i>)	26
2.4.9. Workplace	26
2.4.10. Fasilitas Penjelasan	27
2.4.11. Perbaikan Pengetahuan	27
2.4.12. Klasifikasi Sistem Pakar	27
2.4.13. Representasi Pengetahuan.....	29
2.4.14. Metode Pengembangan Sistem Pakar	29
2.4.15. Kelebihan dan Kekurangan Sistem Pakar	31
2.5. Gangguan jiwa	33
2.5.1. Kriteria Gangguan.....	33
2.6. Klasifikasi gangguan jiwa	33
2.7. Jenis Gangguan Jiwa.....	35

2.8. <i>Forward Chaining</i>	38
2.9. <i>WEB</i>	39
2.10. Tinjauan Perangkat Lunak	39
2.10.1. PHP (Hypertext Processing)	39
2.10.2. XAMPP	40
2.10.3. MySQL.....	40
2.10.4. CSS (Cascading Style Sheets).....	41
2.10.5. <i>Jquery</i>	42
2.10.6. <i>Bootstrap</i>	42
2.10.7. <i>Apache Web Server</i>	43
2.11. Entity Relationship Diagram (ERD)	44
2.12. <i>Flowchart</i>	47
2.13. DFD (<i>Data Flow Diagram</i>)	52
2.14. <i>Testing</i>	54
2.14.1. Pengertian <i>Testing</i>	54
2.14.2. <i>Behavioral Test (Black-Box Test)</i>	55
BAB III	56
ANALISIS MASALAH DAN PERANCANGAN PROGRAM	56
3.1. Communication	56
3.1.1. Pengumpulan Data	56

3.1.2. Analisis Sistem Pakar.....	58
3.1.3. Analisis Identifikasi Masalah.....	59
3.1.4. Analisis Sistem Yang Sedang Berjalan.....	63
3.1.5. Analisis Sistem Diusulkan	64
3.1.6. Analisis Pengguna.....	66
3.2.5. Sumber Informasi.....	68
3.2. Planning	68
3.3. Modeling	70
3.3.1. Definisi Aktor	75
3.3.2. Perancangan Sistem	76
3.3.3. <i>Design Interface</i>	94
BAB IV	104
4.1. <i>Construction (Code & Test)</i>	104
4.1.1. Implementasi <i>Hardware & Software</i>	104
4.1.2. Implementasi Antarmuka.....	105
4.1.3. <i>Testing</i>	109
4.1.4. Pengujian Keakuratan	126
4.1.5. Kesimpulan Testing	127
BAB V.....	128
5.1. Kesimpulan	128

5.2. Saran.....	129
DAFTAR PUSTAKA	130
LAMPIRAN.....	133

DAFTAR TABEL

TABEL : 2.1. Perbandingan Kemampuan Pakar Dengan Sistem Pakar.....	19
Lanjutan TABEL : 2.1. Perbandingan Kemampuan Pakar Dengan Sistem Pakar	20
TABEL : 2.3. Simbol ERD (Fathansyah dalam Joni, 2016).....	45
TABEL : 2.4. Tabel Simbol Penghubung/Alur <i>Flowchart</i>	50
TABEL : 2.5. Tabel Proses <i>Flowchart</i>	50
Lanjutan TABEL : 2.5. Tabel Proses <i>Flowchart</i>	51
TABEL : 2.6. Tabel Input/Output Flowchart.....	51
Lanjutan TABEL : 2.6. Tabel <i>Input/Output Flowchart</i>	52
TABEL : 2.7. Tabel Notasi DFD	53
TABEL : 3.1 Referensi Penelitian	57
Lanjutan TABEL : 3.1 Referensi Penelitian	58
TABEL : 3.2. Tabel Penjadwalan Penelitian	69
TABEL : 3.3. Tabel Penyakit Kejiwaan	70
TABEL : 3.4. Tabel Gejala Penyakit Kejiwaan.....	71
Lanjutan TABEL : 3.4. Tabel Gejala Penyakit Kejiwaan.....	72
Lanjutan TABEL : 3.4. Tabel Gejala Penyakit Kejiwaan.....	73
TABEL : 3.5. Tabel Rule	73
Lanjutan TABEL : 3.5. Tabel Rule.....	74
Lanjutan TABEL : 3.5. Tabel Rule.....	75

TABEL : 3.6. Tabel Penyakit.....	92
TABEL : 3.7. Tabel Gejala	93
TABEL : 3.8. Tabel Rule	93
TABEL : 3.9. Tabel Solusi.....	94
TABEL : 3.10. Tabel Pencegahan	94
TABEL : 4.1. Tabel Rencana Pengujian	109
Lanjutan TABEL : 4.1. Tabel Rencana Pengujian.....	110
TABEL : 4.2. Tabel Hasil Pengujian Login Pakar.....	110
TABEL : 4.3. Tabel Hasil Pengujian Tambah Data Penyakit.....	111
TABEL : 4.4. Tabel Hasil Pengujian Ubah Data Penyakit	112
TABEL : 4.5. Tabel Hasil Pengujian Hapus Data Penyakit	113
TABEL : 4.6. Tabel Hasil Pengujian Tambah Data Gejala	114
TABEL : 4.7. Tabel Hasil Pengujian Ubah Data Gejala.....	114
Lanjutan TABEL : 4.7. Tabel Hasil Pengujian Ubah Data Gejala	115
TABEL : 4.8. Tabel Hasil Pengujian Hapus Data Gejala	115
TABEL : 4.9. Tabel Hasil Pengujian Tambah Data Rule	116
TABEL : 4.10. Tabel Hasil Pengujian Tambah Data Rule	116
Lanjutan TABEL : 4.10. Tabel Hasil Pengujian Tambah Data Rule.....	117
TABEL : 4.11. Tabel Hasil Pengujian Tambah Data Rule	117
TABEL : 4.12. Tabel Hasil Pengujian Tambah Data Pencegahan	118

TABEL : 4.13. Tabel Hasil Pengujian Ubah Data Pencegahan.....	119
TABEL : 4.14. Tabel Hasil Pengujian Hapus Data Pencegahan	120
TABEL : 4.15. Tabel Hasil Pengujian Tambah Data Solusi.....	121
TABEL : 4.16. Tabel Hasil Pengujian Ubah Data Solusi	122
TABEL : 4.17. Tabel Hasil Pengujian Hapus Data Solusi	123
TABEL : 4.18. Tabel Hasil Pengujian Diagnosa Gejala.....	124
TABEL : 4.3. Rencana <i>Testing</i> Klasifikasi Penyakit Kejiwaan	125

DAFTAR GAMBAR

GAMBAR : 2.1. Konsep Dasar Fungsi Sistem Pakar(Muhammad Arhami, “Konsep Dasar Sistem Pakar”)	21
GAMBAR : 2.2. Arsitektur Sistem Pakar (Turban, 1995)	24
GAMBAR : 2.3. Model Waterfall (Pressman, 2015:42)	30
GAMBAR : 2.4. One to one (Fathansyah dalam Joni, 2016)	46
GAMBAR : 2.5. One to Many (Fathansyah dalam Joni, 2016)	46
GAMBAR : 2.6. Many to Many (Fathansyah dalam Joni, 2016)	47
GAMBAR : 3.1. <i>Flowmap</i> Sistem Yang Sedang Berjalan	64
GAMBAR : 3.2. <i>Flowmap</i> Sistem Pakar Yang Diusulkan	65
GAMBAR : 3.3. <i>Context</i> Diagram	76
GAMBAR : 3.4. DFD Level 1 Sistem Pakar Diagnosa Penyakit Kejiwaan	77
GAMBAR : 3.5. DFD Level 2 Sistem Pakar Diagnosa Penyakit kejiwaan	78
GAMBAR : 3.6. DFD Level 3 (Data Pencegahan)	79
GAMBAR : 3.7. DFD Level 3 (Data Solusi)	80
GAMBAR : 3.8. DFD Level 3 (Data Gejala)	81
GAMBAR : 3.9. DFD Level 3 (Data Penyakit)	81
GAMBAR : 3.10. DFD Level 3 (Data Rule)	82
GAMBAR : 3.11. DFD Level 3 (Konsultasi)	83

GAMBAR : 3.12. DFD Level 3 (Informasi Kepakaran)	83
GAMBAR : 3.13. <i>Flowchart Login</i>	84
GAMBAR : 3.14. <i>Flowchart</i> Kelola Data Pencegahan	85
GAMBAR : 3.15. <i>Flowchart</i> Kelola Data Solusi	86
GAMBAR : 3.16. <i>Flowchart</i> Kelola Data Gejala.....	87
GAMBAR : 3.17. <i>Flowchart</i> Data Penyakit.....	88
GAMBAR : 3.18. <i>Flowchart</i> Kelola Data Rule.....	89
GAMBAR : 3.18. <i>Flowchart</i> Konsultasi	90
GAMBAR : 3.20. <i>Flowchart</i> Informasi Kepakaran	91
GAMBAR : 3.21. ERD Sistem Pakar Diagnosa Penyakit Kejiwaan.....	92
GAMBAR : 3.22. Rancangan Menu Awal Sistem Pakar Diagnosa Penyakit Kejiwaan	95
GAMBAR : 3.23. Rancangan Menu Diagnosa Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	96
GAMBAR : 3.24. Rancangan Hasil dari Konsultasi Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	97
GAMBAR : 3.25. Rancangan Memasukan Data Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	98
GAMBAR : 3.26. Rancangan Dari Memasukan Data Gejala Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Kejiwaan.....	99

GAMBAR : 3.27. Rancangan Hasil dari Memasukan Data Rule Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	100
GAMBAR : 3.28. Rancangan Hasil dari Memasukan Data Pencegahan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	101
GAMBAR : 3.29. Rancangan Hasil dari Memasukan Data Solusi Pada Sistem Pakar Diagnosa Penyakit Kejiwaa	102
GAMBAR : 3.30. Rancangan Informasi Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan	103
GAMBAR : 4.1. Hasil Tampilan Memilih Gejala Penyakit Kejiwaan.....	105
GAMBAR : 4.2. Hasil Tampilan Dari Hasil Proses Diagnosa Metode <i>Forward Chaining</i>	106
GAMBAR : 4.3. Hasil Tampilan Dari Data Penyakit Kejiwaan	106
GAMBAR : 4.4. Hasil Tampilan Dari Data Gejala Kejiwaan.....	107
GAMBAR : 4.5. Hasil Tampilan Dari Data Basis Pengetahuan.....	107
GAMBAR : 4.6. Hasil Tampilan Dari Data Pencegahan Penyakit Kejiwaan	108
GAMBAR : 4.7. Hasil Tampilan Dari Data Solusi Penyakit Kejiwaan	108

BAB I

PENDAHULUAN

1.1. Latar Belakang

Sakit jiwa adalah gangguan mental yang berdampak kepada mood, pola pikir, hingga tingkah laku secara umum. Seseorang disebut mengalami gangguan kejiwaan, jika gejala dan tanda gangguan jiwa yang dialami membuatnya tertekan dan tidak mampu melakukan aktivitas sehari-hari secara normal. Ciri-ciri orang yang mengalami sakit jiwa dapat berbeda-beda tergantung dari jenisnya. Pada umumnya, orang yang mengalami gangguan jiwa dapat dikenali dari beberapa gejala tertentu, seperti perubahan *mood* yang sangat drastis, merasa ketakutan secara berlebihan, menarik diri dari kehidupan sosial, mengalami delusional, dan terkadang gejala diiringi oleh gangguan fisik, seperti sakit kepala, nyeri punggung, sakit perut, atau nyeri lain yang tidak dapat dijelaskan.

Kemajuan ilmu pengetahuan dan teknologi informasi memberikan banyak kemudahan bagi kehidupan manusia bahkan tidak hanya dibidang-bidang ilmu komputer saja. Sistem pakar adalah merupakan sebagian dari cabang dari kecerdasan buatan yang mempelajari bagaimana cara meniru cara berfikir para pakar-pakar pada bidangnya dalam menyelesaikan suatu permasalahan dan membuat keputusan serta dapat mengambil kesimpulan sejumlah fakta. Intinya dalam sistem pakar adalah bagaimana membuat pikiran atau pengetahuan yang dimiliki oleh seorang pakar ke dalam komputer menjadi sebuah program komputer, dan bagaimana membuat keputusan serta mengambil kesimpulan berdasarkan

pengetahuan itu. Dengan menyimpan informasi dan digabungkan dengan himpunan aturan yang memadai memungkinkan komputer memberikan kesimpulan atau keputusan seperti seorang pakar.

Perancangan *software* sistem pakar di dalam bidang kesehatan diharapkan dapat memudahkan bagi masyarakat awam dan membantu orang yang berkecimpung dalam bidang kesehatan untuk mengetahui dan memudahkan mendiagnosa gangguan kejiwaan. Hasil keluaran *software* ini akan berupa gangguan kejiwaan yang diidentifikasi yang berhubungan dengan gejala-gejala gangguan kejiwaan menggunakan metode Forward Chaining.

Alasan menggunakan metode ini dapat memberikan hasil yang akurat yang didapat dari sekumpulan fakta-fakta dengan mencari kaidah yang cocok dengan dugaan/hipotesa yang akan menuju kesimpulan akhir. Penelitian ini berjudul **“PERANCANGAN APLIKASI SISTEM PAKAR DIAGNOSIS PENYAKIT KEJIWAAN DENGAN METODE FORWARD CHAINING BERBASIS WEB”**.

1.2. Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan diatas, sehingga dapat dirumuskan beberapa permasalahan yang ada, yaitu :

1. Bagaimana cara membuat suatu sistem pakar dapat mendiagnosa penyakit kejiwaan dengan akurat?
2. Bagaimana cara menerapkan metode forward chaining untuk mendiagnosis penyakit kejiwaan?

3. Bagaiman mengetahui suatu gejala termasuk kedalam gangguan kejiwaan?

1.3. Tujuan Penelitian

Adapun tujuan dari penelitian ini :

1. Cara menentukan penyakit kejiwaannya dengan penelusuran fakta-fakta dari sebuah hipotesa.
2. Sebagai media yang membantu masyarakat mempelajari apa itu penyakit kejiwaan dengan jelas dan apa saja yang termasuk ke dalam gejala-gejala penyakit kejiwaan.
3. Untuk membantu masyarakat dalam mencegah maupun cara mengobati penyakit penyakit kejiwaan tersebut.

1.4. Batasan Masalah

Berdasarkan rumusan masalah diatas penulis membatasi permasalahan yang akan dibahas, yaitu :

1. Aplikasi ini dibuat menggunakan bahasa pemograman *PHP* dan *database MySql*.
2. Permasalahan kesehatan yang dibahas merupakan penyakit kejiwaan yang dicocokkan sumber sebenarnya.
3. Penelitian ini dilakukan hanya sampai proses *construction* (tidak melakukan proses *deployment*).
4. Sistem yang dibuat tidak bisa memproses penyakit kejiwaan lebih dari 1 penyakit kejiwaan.
5. Gejala dan penyakit yang didapat hanya berdasarkan pengetahuan pakar.

6. Metode yang digunakan pada sistem pakar ini dengan menggunakan metode *Forward Chaining*.
7. Interaksi antara sistem dan user menggunakan pertanyaan berupa gejala yang dialami oleh user.
8. Output yang dihasilkan sistem berupa informasi hasil diagnosa penyakit beserta solusi dan juga pencegahan dari penyakit kejiwaan.

1.5. Metode Penelitian

1.5.1. Metode Pengumpulan Data

Adapun metode pengumpulan data yang dilakukan untuk mendapatkan data-data dan referensi yang dibutuhkan dalam menyusun penelitian ini, meliputi :

1. Studi Literatur

Dilakukan dengan cara membaca dan mempelajari buku-buku, referensi yang mendukung dengan topik yang akan dibahas dalam penyusunan penelitian ini, Selain itu penulis mengumpulkan data dari situs *internet* yang berhubungan dengan penelitian ini.

2. Studi Lapangan

a. Observasi

Melakukan observasi tidak langsung melalui media untuk mendapatkan data-data yang diperlukan.

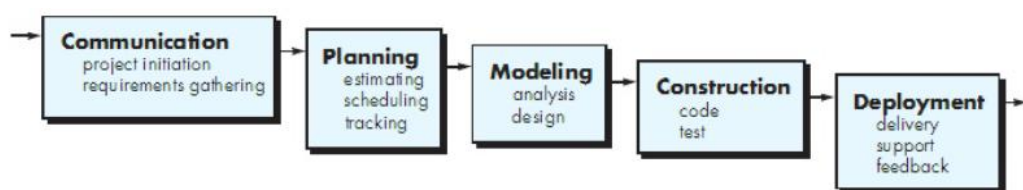
b. Wawancara

Melakukan wawancara kepada pakar kejiwaan untuk mendapatkan data-data yang diperlukan dalam penelitian.

1.5.2. Metode Pengembangan Sistem

Dalam penelitian ini peneliti menggunakan metode penelitian kuantitatif. Penelitian kuantitatif digunakan dalam meneliti status kelompok manusia. Suatu sistem pemikiran atau kelas peristiwa pada waktu tertentu. Sehingga melalui metode ini akan diperoleh data dan informasi tentang gambaran suatu fenomena, fakta, sifat serta hubungan fenomena tertentu secara komperhensif dan integral (Sugiyono, 2013: 13). Dengan demikian pengulangan dalam penelitian kuantitatif dilakukan dalam rangka mendapatkan konsistensi atau reliabilitas data penelitian yang ada.

Kerangka berpikir dalam penelitian ini menggunakan metode pengembangan (SDLC) dengan model *waterfall*. Model ini melakukan pendekatan secara sistematis dan berurutan. Metode *waterfall* mempunyai tahapan-tahapan yang digambarkan pada Gambar 1.1.



GAMBAR : 1.1. Model Waterfall (Pressman, 2015:42)

Berikut ini adalah penjelasan dari tahapan-tahapan yang dilakukan di dalam Model *Waterfall* menurut pressman.

1. *Communication (Project Initiation & Requirements Gathering)*

Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan *customer* demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek, seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi *software*. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, dan *internet*.

2. *Planning (Estimating, Scheduling, Tracking)*

Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan *tracking* proses pengerjaan sistem.

3. *Modeling (Analysis & Design)*

Tahapan ini adalah tahap perancangan dan permodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur *software*, tampilan *interface*, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

4. *Construction (Code & Test)*

Tahapan *Construction* ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang

sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

5. *Deployment (Delivery, Support, Feedback)*

Tahapan *Deployment* merupakan tahapan implementasi *software* ke *customer*, pemeliharaan *software* secara berkala, perbaikan *software*, evaluasi *software*, dan pengembangan *software* berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya.

(Pressman, 2015:17).

Dalam penelitian ini hanya akan dilakukan hingga tahap empat sehingga tahap penelitian ini tidak dilakukan dan menjadi batasan masalah.

1.6. Penelitian Sebelumnya

1.6.1. Diagnosis penyakit kejiwaan menggunakan metode certainty factor

Pada penelitian ini sistem yang dibangun hanya mendiagnosa 4 penyakit gangguan kejiwaan yaitu Skizofrenia, Bipolar, Depresi, Gangguan kecemasan (Anxiety Disorder). Menggunakan metode certainty factor user dapat memilih gejala-gejala yang dialami pasien.

1.6.2. Aplikasi probabilitas bayes dalam sistem pakar untuk mendiagnosis gangguan kejiwaan bipolar

Pada penelitian ini sistem yang dibangun menggunakan metode Probabilitas bayes pada penelitian ini digunakan 2 macam basis pengetahuan yaitu

menggunakan data pembelajaran bipolar saja dan dengan menggunakan data pembelajaran bipolar dan skizofrenia. Tingkat keakuratan mencapai 44,83%.

1.7. Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini peneliti menjelaskan latar belakang masalah mengapa peneliti memilih tema ini. Di samping itu, bab ini juga memuat rumusan masalah yang bertujuan agar pembahasan dalam skripsi ini tidak meluas dari garis yang telah ditetapkan. Disini juga terdapat tujuan penelitian yang menjelaskan tentang hal-hal yang disampaikan untuk menjawab permasalahan yang telah ditentukan, dan yang terakhir sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisikan tentang penjabaran mengenai literatur yang digunakan dan mendukung terhadap permasalahan yang dikaji, yaitu mengemukakan penjelasan berbagai sumber kepustakaan yang menjadi rujukan serta relevan dengan permasalahan yang akan dibahas yaitu “Sistem pakar diagnosis gangguan kejiwaan berbasis *web*”. Penulis menggunakan sumber *jurnal, skripsi*, dan buku-buku yang membahas tentang gangguan kejiwaan

BAB III ANALISIS MASALAH DAN PERANCANGAN PROGRAM

Bab ini berisikan tentang proses perancangan program sistem pakar penentu gangguan kejiwaan berdasarkan analisa permasalahan terlebih dahulu dan disesuaikan dengan metode penelitian yang telah dipilih.

BAB IV IMPLEMENTASI DAN UJI COBA

Bab ini berisikan tentang penelitian yang dilakukan sehingga dapat dibuat perancangan pada program yang dibuat. Disini juga ada tahapan implementasi program, termasuk juga kelebihan dan kekurangan pada program. Terdapat juga evaluasi pada program dengan menggunakan metode kualitatif, kuantitatif, atau metode lainnya yang bisa dan sesuai digunakan.

BAB V PENUTUP

Bab ini menjelaskan mengenai kesimpulan dan saran yang dibutuhkan terhadap hasil akhir dari penelitian yang telah dibuat dan untuk peningkatan topik skripsi yang dapat digunakan di masa mendatang.

BAB II

LANDASAN TEORI

2.1. Sistem

2.1.1. Pengertian Sistem

Sistem adalah suatu kumpulan elemen yang di dalamnya terdapat prosedur yang digunakan dalam rangka mencari suatu tujuan bersama melalui cara pengoperasian barang dan data pada saat tertentu. Cara ini ditujukan untuk mendapatkan data atau informasi yang diinginkan. (Murdick, R.G, 1991: 27).

2.1.2. Karakteristik Sistem

Mengemukakan sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu :

1. Komponen

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

2. Batasan Sistem

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut

3. Lingkungan Luar Sistem

Lingkungan luar (*evinronment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. sedang lingkunagn luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kalangsungan hidup dari sistem.

4. Penghubung Sistem

Penghubung (*interfance*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

Sedangkan menurut Burch dan Grundnitski desain sistem dapat didefinisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi menentukan bagaimana suatu sistem akan menyelesaikan tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sesistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem (Jogiyanto ; 2005:196).

Berdasarkan beberapa defenisi diatas maka desain sistem dapat diartikan sebagai berikut:

- Tahap setelah analisis dari siklus pengembangan sistem
- Pendefinisian dari kebutuhan-kebutuhan fungsional
- Persiapan untuk rancang bangun untuk implementasi
- Menggambarkan bagaimana suatu sistem dibentuk

Yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi

Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

5. Masukan Sistem

Masukan (*input*) sistem adalah energi yang masukan kedalam sistem. masukan dapat berupa masukan perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukan supaya tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. sebagai contoh didalam komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem

Keluaran (*output*) sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluran yang berguna dan sisa pembuangan, misalnya untuk sistem komputer. Panas yang dihasilkan adalah keluaran yang tidak berguna

dan merupakan hasil sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

8. Sasaran Sistem

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem tersebut dapat dikatakan berhasil apabila mencapai/mengenuhi sasaran atau pun tujuan. (Jogiyanto, 2005: 3).

2.1.3. Klasifikasi Sistem

Bahwa sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem Abstrak (*Abstract system*)

Sistem ini merupakan sistem yang tidak tampak secara fisik, karena hanya berupa pemikiran atau ide-ide.

2. Sistem Fisik (*Physical system*)

Sistem fisik merupakan sistem yang tampak secara fisik sehingga setiap makhluk dapat melihatnya.

3. Sistem Alamiah (*Natural system*)

Sistem alamiah ini adalah sistem yang terjadi dari proses - proses alam dalam arti tidak dibuat oleh manusia.

4. Sistem Buatan Manusia (*Human made system*)

Sistem ini merupakan sistem yg dirancang & didisain oleh manusia.

5. Sistem Deterministik (*Deterministic System*)

Sistem yg beroperasi dengan tingkah laku yg dapat diramalkan disebut sistem deterministik . Interaksi antar tiap-tiap bagian dapat dideteksi, sehingga out- putnya juga dapat diramalkan.

6. Sistem Tak Tentu (*Probabilistic System*)

Sistem ini adalah sistem dimana kondisi masa depannya tak dapat diramalkan karena mengandung probabilitas.

7. Sistem Tertutup (*closed system*)

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak dipengaruhi oleh lingkungan luarnya. Sebenarnya didunia ini tidak ada sistem yg benar-benar tertutup. Yang ada hanyalah sistem yang secara relatif tertutup (*relatively closed system*)

8. Sistem Terbuka (*open system*)

Sistem ini kebalikan dari sistem tertutup, karena sistem terbuka adalah sistem yg berhubungan dan dipengaruhi oleh lingkungannya. Oleh sebab itu sistem ini harus mempunyai suatu sistem pengendalian (*control system*) yang baik, agar yg masuk hanya pengaruh – pengaruh yang baik saja. (Jogiyanto, 2001:9).

2.2. Basis Data (Database)

2.2.1. Pengertian Basis Data

Basis data adalah kumpulan data yang saling berhubungan secara logis dan didesain untuk mendapatkan data yang dibutuhkan oleh suatu organisasi. (Indrajani, 2015:70)

2.2.2. Database Management System (DBMS)

DBMS adalah sebuah sistem perangkat lunak yang mengizinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengontrol akses ke dalam basis data. (Connolly dan Begg, 2010 :66)

2.2.3. Fasilitas Database Management System (DBMS)

1. Mengizinkan pengguna untuk mendefinisikan basis data, dengan melalui *Data Definition Language (DDL)*. *DDL* mengizinkan pengguna untuk menentukan tipe, struktur, serta kendala data yang nantinya akan disimpan ke dalam basis data.
2. Mengizinkan pengguna untuk melakukan menambah, mengubah, menghapus, dan mengambil data dari basis data tersebut, dengan menggunakan *Data Manipulation Language (DML)*. *Standard* bahasa dari *DBMS* ialah *Structured Query Language (SQL)*.
3. Menyediakan akses kontrol ke dalam basis data, seperti :
 - a. Sistem keamanan, yang dapat mencegah pengguna yang tidak diberi kuasa untuk mengakses basis data.
 - b. Sistem integritas, yang dapat menjaga konsistensi dari data yang tersimpan.

- c. Sistem kontrol konkurensi, yang mengizinkan berbagi akses dengan basis data.
- d. Sistem kontrol pemulihan, jika terjadi kegagalan perangkat keras atau perangkat lunak maka sistem kontrol pemulihan ini dapat mengembalikan basis data ke keadaan yang konsisten dari yang sebelumnya.

2.2.4. Komponen Utama *Database Management System* (DBMS)

1. Hardware

Hardware yang digunakan dapat berupa *Personal Computer* (PC) yang akan disesuaikan dengan kebutuhan perusahaan dan *DBMS* yang akan digunakan.

2. Software

Komponen *software* terdiri dari *software DBMS* itu sendiri dan program aplikasi, bersamaan dengan sistem operasinya, serta termasuk *software* jaringan, apabila *DBMS* yang akan digunakan melalui sebuah jaringan.

3. Data

Data adalah komponen yang terpenting pada *DBMS*, karena data merupakan sebuah jembatan penghubung antara komponen mesin dengan manusia.

4. Procedures

Prosedur berisikan instruksi serta aturan yang digunakan untuk merancang dan menggunakan sebuah basis data.

5. *People*

Komponen terakhir adalah manusia yang dapat terlibat langsung dengan sistem tersebut.

2.2.5. Keuntungan & Kerugian *Database Management System* (DBMS)

A. Keuntungan *Database Management System* (DBMS)

1. Mengendalikan redudansi data

Menghilangkan redudansi dengan cara mengintegrasikan *file-file* tersebut agar salinan dari data yang sama tidak disimpan. Dikarenakan apabila data yang sama dengan data tersebut ditemukan lebih dari satu tabel di dalam basis data maka akan terjadi redudansi.

2. Meningkatkan Integritas data

Integritas basis data megacu pada validitas dan konsistensi data yang tersimpan. Integritas dinyatakan dalam *constraints*.

3. Meningkatkan keamanan

Keamanan sebuah basis data melindungi basis data dari pengguna yang tidak berwenang. dengan cara membuat *username* dan *password* untuk mengidentifikasi pengguna yang memiliki hak akses dalam menggunakan basis data. Akses yang diberikan pada pengguna yang telah memiliki hak akses dapat melakukan operasi seperti, *retrieval*, *insert*, *update*, *delete*.

4. Meningkatkan pelayanan *backup* dan *recovery*

Menyediakan fasilitas untuk pemulihan data bila terjadi kegagalan pada *software* atau *hardware* dapat di pulihkan sehingga dapat meminimalkan jumlah pemrosesan yang hilang.

5. Berbagi Data

DBMS memungkinkan pengguna untuk menggunakan data yang sama secara bersamaan tentunya dengan pengguna yang berwenang.

B. Kerugian *Database Management System (DBMS)*

1. Kompleksitas

DBMS merupakan sebuah perangkat lunak yang sangat kompleks. *Database Designers, Database Developer, Database Administrator* dan *End-user* harus mengerti fungsionalitasnya.

2. Ukuran

Karena besarnya kompleksitas pada *DBMS* membuat *DBMS* membutuhkan sebuah kapasitas penyimpanan yang besar agar dapat menjalankan aplikasinya.

3. Biaya dalam *DBMS*

Biaya dalam *DBMS* sangat bervariasi, dan itu tergantung dari lingkungan dan fungsionalitas yang diinginkan.

4. Biaya tambahan dari *hardware*

Membutuhkan biaya tambahan untuk kapasitas penyimpanan dan agar dapat mencapai kinerja yang diinginkan sehingga membutuhkan mesin yang lebih besar.

5. Biaya Konversi

Biaya konversi relatif lebih kecil. Biaya konversi meliputi pelatihan karyawan untuk menggunakan sistem yang baru dan memungkinkan karyawan dengan keahlian yang khusus untuk membantu konversi dan menjalankan sistem.

2.3. Pakar

2.3.1. Pengertian Pakar

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus, serta mampu menerapkannya untuk memecahkan masalah atau memberi nasehat. Seorang pakar harus mampu menjelaskan untuk mempelajari hal-hal baru yang berkaitan dengan topik permasalahan, jika perlu harus mampu menyusun kembali pengetahuan-pengetahuan yang didapatkan, dan dapat memecahkan aturan-aturan serta menentukan relevansi kepakarannya. (T. Sutojo, 2011 : 163)

Seorang pakar dengan sistem pakar mempunyai banyak perbedaan. (Durkin, 1995) mengemukakan perbandingan kemampuan antara seorang pakar dengan sebuah sistem seperti pada tabel 2.1 berikut ini :

TABEL : 2.1. Perbandingan Kemampuan Pakar Dengan Sistem Pakar(Durkin, 1994)

<i>Factor</i>	<i>Human Expert</i>	<i>Expert System</i>
<i>Time Availability</i>	Hari Kerja	Setiap Saat
Geografis	Lokal/Tertentu	Dimana Saja

Lanjutan TABEL : 2.1. Perbandingan Kemampuan Pakar Dengan Sistem Pakar(Durkin, 1994)

<i>Factor</i>	<i>Human Expert</i>	<i>Expert System</i>
Keamanan	Tidak Tergantikan	Dapat Diganti
Perishable/Dapat Habis	Ya	Tidak
Performansi	Variabel	Konsisten
Kecepatan	Variabel	Konsisten
Biaya	Tinggi	Terjangkau

2.3.2. Ciri-Ciri Pakar

Seorang pakar harus mampu melakukan kegiatan–kegiatan sebagai berikut:

1. Mengenali dan memformulasikan permasalahan
2. Memecahkan permasalahan secara cepat dan tepat
3. Menerangkan pemecahannya
4. Belajar dari pengalaman
5. Merekstrukturisasi pengetahuan
6. Memecahkan aturan – aturan menentukan relevansi (T. Sutojo, 2011 : 163)

2.4 Sistem Pakar

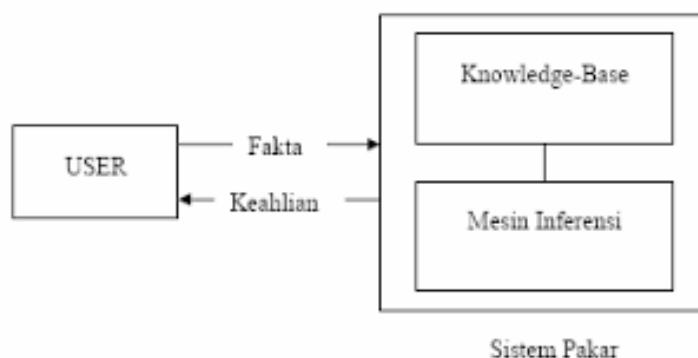
2.4.1. Pengertian Sistem Pakar

Sistem pakar adalah suatu program komputer cerdas yang menggunakan pengetahuan (*knowledge*) dan prosedur inferensi untuk menyelesaikan masalah

yang cukup sulit sehingga membutuhkan seorang yang ahli untuk menyelesaikannya (Weny Widiastuti, 2012).

2.4.2. Konsep Dasar Sistem Pakar

Tiap-tiap orang mempunyai keahlian masing-masing yang mungkin satu orang dengan yang lainnya mempunyai keahlian berbeda, tergantung pengetahuannya masing-masing. Komputer dapat diprogram untuk berbuat seperti ahli dalam bidang tertentu. Komputer yang demikian dapat dijadikan konsultan atau tenaga ahli di bidang tertentu yang dapat menjawab pertanyaan dan nasihat-nasihat yang dibutuhkan. Sistem yang demikian disebut dengan sistem pakar.



GAMBAR : 2.1. Konsep Dasar Fungsi Sistem Pakar (Muhammad Arhami, “Konsep Dasar Sistem Pakar”)

Gambar 2.1 menggambarkan konsep dasar suatu sistem pakar. Pengguna (user) menyampaikan fakta atau informasi untuk sistem pakar dan kemudian menerima saran dari pakar atau jawaban ahlinya. Bagian dalam sistem pakar terdiri dari dua komponen utama, yaitu knowledge base yang berisi pengetahuan dasar dari pakar atau ahlinya dan mesin inferensi yang menggambarkan kesimpulan. Kesimpulan tersebut merupakan respon dari sistem pakar atas permintaan pengguna.

Suatu pengetahuan (*knowledge*) dari sistem pakar bersifat khusus untuk satu *domain* masalah saja. Sistem pakar menyerupai kepakaran manusia yang secara umum dirancang untuk menjadi pakar dalam satu *domain* masalah saja. Sebagai contoh, sistem pakar kedokteran yang dirancang untuk mendiagnosis infeksi penyakit akan mempunyai suatu uraian pengetahuan (*knowledge*) tentang gejala-gejala penyakit yang disebabkan oleh infeksi penyakit. Dalam kasus ini domain pengetahuan (*knowledge*-nya) adalah bidang kedokteran yang terdiri dari *knowledge* tentang penyakit, gejala, dan cara pengobatan.

Menurut Turban terdapat tiga orang yang terlibat dalam lingkungan sistem pakar, yaitu:

1. Pakar

Orang yang memiliki pengetahuan khusus, pendapat, pengalaman, dan metode, serta kemampuan untuk mengaplikasikan keahliannya tersebut untuk menyelesaikan masalah.

2. Perakayasa sistem

Orang yang membantu pakar dalam menyusun area permasalahan dengan menginterpretasikan dan mengintegrasikan jawaban pakar atas pertanyaan yang diajukan, menggambarkan analogi, dan menerangkan kesulitan-kesulitan konseptual.

3. Pengguna

Sistem pakar memiliki beberapa pengguna yaitu pengguna bukan pakar, pelajar, pembangun sistem pakar yang ingin meningkatkan dan menambah basis pengetahuan, dan pakar.

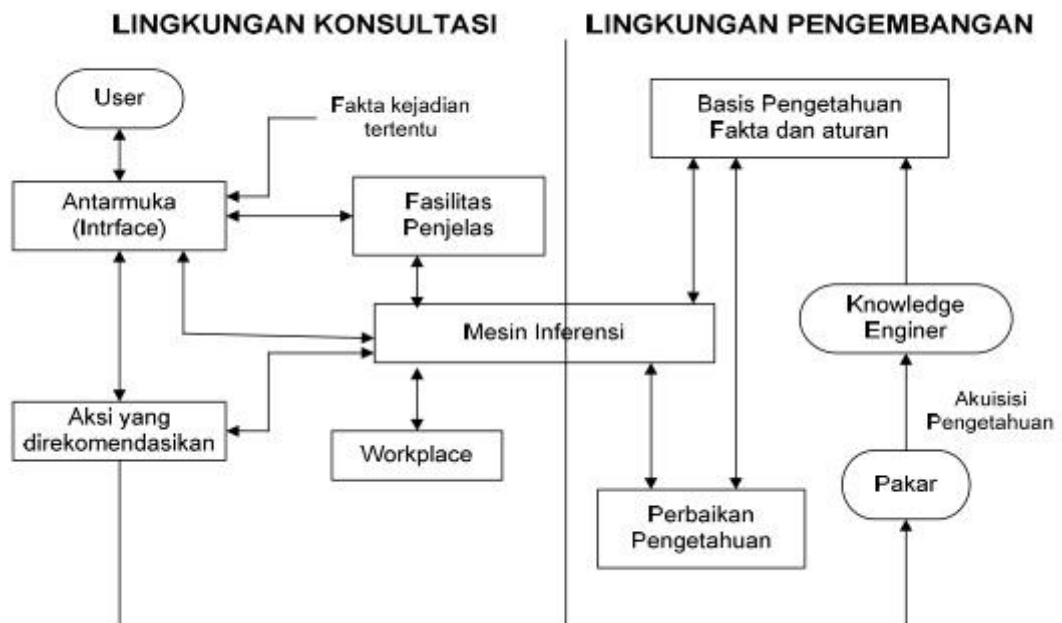
2.4.3. Bentuk Sistem Pakar

Ada empat bentuk sistem pakar, yaitu :

1. Berdiri sendiri. Sistem pakar merupakan software yang berdiri sendiri, tidak tergabung dengan software yang lainnya.
2. Tergabung. Sistem pakar jenis ini merupakan bagian program yang terkandung di dalam suatu algoritma, atau program dimana fungsinya memanggil algoritma sub-rutin lain.
3. Menghubungkan ke *software* lain. Bentuk ini biasanya merupakan sistem pakar yang menghubungkan ke suatu paket program tertentu, misalnya DBMS (*Database Management System*)
4. Sistem mengabdikan. Sistem pakar merupakan bagian dari komputer khusus yang dihubungkan dengan suatu fungsi tertentu. Misalnya sistem pakar yang digunakan untuk membantu menganalisa data radar.

2.4.4. Struktur Sistem Pakar

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) (Turban, 1995.). Lingkungan pengembangan sistem pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar. Komponen-komponen sistem pakar dalam kedua bagian tersebut dapat dilihat dalam Gambar 2.2 berikut ini :



GAMBAR : 2.2. Arsitektur Sistem Pakar (Turban, 1995)

1. Lingkungan pengembangan digunakan sebagai pembangunan sistem pakar, baik dari segi pembangunan komponen maupun basis pengetahuan.
2. Lingkungan konsultasi digunakan oleh seorang yang bukan ahli untuk berkonsultasi.

Komponen-komponen yang terdapat dalam sistem pakar adalah seperti yang terdapat pada Gambar 2.3, yaitu *User Interface* (antarmuka pengguna), basis pengetahuan, akuisisi pengetahuan, mesin inferensi, *workplace*, fasilitas penejelasan, perbaikan pengetahuan.

2.4.5. Antarmuka Pengguna (*User Interface*)

User interface merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu

antarmuka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.(Muhammad Arhami, 2005:14).

2.4.6. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah, ada dua bentuk pendekatan basis pengetahuan yang sangat umum digunakan, yaitu diantaranya sebagai berikut :

1. Penalaran Berbasis Aturan (*Rule-Based Reasoning*) Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk IF-THEN. Bentuk ini digunakan apabila memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu.
2. Penalaran Berbasis Kasus (*Case-Based Reasoning*) Pada penalaran berbasis kasus, basis pengetahuan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip).

2.4.7. Akuisisi Pengetahuan (*Knowledge Acquisition*)

Akuisisi pengetahuan adalah akumulasi, transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer. Dalam tahap ini *knowledge engineer* berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke basis pengetahuan.(Muhammad Arhami, 2005:16).

2.4.8. Mesin Inferensi (*Inference Engine*)

Mesin inferensi berperan sebagai otak dari sistem pakar. Mesin inferensi berfungsi untuk memandu proses penalaran terhadap suatu kondisi, berdasarkan pada basis pengetahuan yang tersedia. Di dalam mesin inferensi terjadi proses untuk memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan dalam rangka mencapai solusi atau kesimpulan.

Dalam prosesnya, mesin inferensi menggunakan strategi penalaran dan strategi pengendalian. Strategi penalaran dibagi menjadi dua bagian yaitu :

1. Strategi Penalaran Pasti (*Exact Reasoning*)

Exact reasoning akan dilakukan jika semua data yang dibutuhkan untuk menarik suatu kesimpulan tersedia.

2. Strategi Penalaran Tidak Pasti (*Inexact Reasoning*)

Inexact reasoning akan dilakukan jika keadaan sebaliknya dari strategi penalaran pasti.

2.4.9. Workplace

Workplace merupakan area dari sekumpulan memori kerja (*working memory*). *Workplace* digunakan untuk merekam hasil-hasil antara dan kesimpulan yang dicapai. Ada 3 tipe keputusan yang dapat direkam, yaitu :

1. Rencana

Bagaimana menghadapi masalah.

2. Agenda

Aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi.

3. Solusi

Calon aksi yang akan dibangkitkan

2.4.10. Fasilitas Penjelasan

Fasilitas penjelasan adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pemakai.

2.4.11. Perbaikan Pengetahuan

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

2.4.12. Klasifikasi Sistem Pakar

Berdasarkan kegunaannya, sistem pakar dapat diklasifikasikan menjadi enam jenis, yaitu :

1. Diagnosis

Sebuah tindakan dalam menangani ketidakberesan akan suatu hal yang menggunakan alat bantu atau sistem sehingga menghasilkan informasi dan membuat inferensi kemungkinan terjadinya ketidakberesan itu sendiri. Diagnosis sistem pakar biasanya digunakan untuk merekomendasikan suatu tindakan seperti diagnosis penyakit, kerusakan mesin, dan sebagainya.

2. Pengajaran

Sistem pakar ini digunakan untuk mengajar siswa. Prinsipnya adalah mendiagnosa apa yang menjadi masalah siswa yang mengalami kesulitan dalam belajar, kemudian memberikan solusi untuk memperbaikinya.

3. Interpretasi

Sistem pakar ini digunakan untuk menganalisa data yang tidak lengkap, tidak teratur, dan data yang kontradiktif. Misalnya menginterpretasi citra.

4. Prediksi

Keunggulan seorang pakar adalah dapat memprediksi ke depan. Sebagai contohnya, bagaimana seorang ahli meteorologi dapat meramal cuaca besok dengan menggunakan data-data cuaca sebelumnya. Sistem pakar ini biasanya digunakan peramalan cuaca, penentuan masa tanam, dan sebagainya.

5. Perencanaan

Untuk sistem pakar ini, perencanaan sistem pakar ini memiliki cakupan luas, mulai dari perencanaan bangunan sampai manajemen bisnis. Keunggulan penggunaan sistem ini dapat menghemat biaya, waktu, dan material. Contohnya penggunaan sistem ini antara lain sistem konfigurasi, komputer, tata letak sirkuit, dan sebagainya.

6. Kontrol

Sistem kontrol ini digunakan untuk melakukan pengontrolan terhadap kegiatan yang membutuhkan waktu dengan presisi tinggi, misalnya industri-industri berteknologi tinggi.

2.4.13. Representasi Pengetahuan

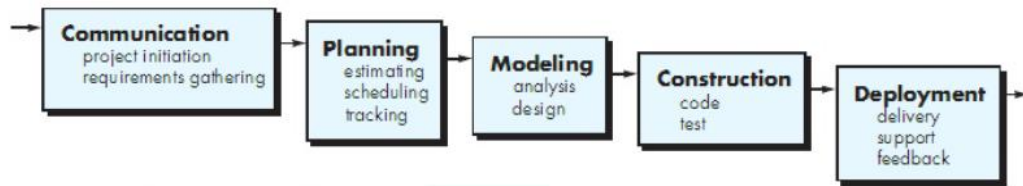
Pengetahuan merupakan kemampuan untuk membentuk model mental yang menggambarkan objek dengan tepat dan merepresentasikannya dalam aksi yang dilakukan terhadap suatu objek.

Representasi pengetahuan merupakan metode yang digunakan untuk mengodekan pengetahuan dalam sebuah sistem pakar yang berbasis pengetahuan. Perepresentasian dimaksudkan untuk menangkap sifat-sifat penting problem dan membuat informasi itu dapat diakses oleh prosedur pemecah problema.

Salah satu representasi pengetahuan yang terdapat dalam sistem pakar yaitu adalah Kaidah Produksi (*Production Rule*). Pada pengetahuan ini disajikan dalam aturan-aturan yang berbentuk pasangan keadaan-aksi (*condition-action*) : “IF keadaan terpenuhi atau terjadi THEN suatu aksi akan terjadi”. Sistem pakar yang basis pengetahuannya disajikan dalam bentuk aturan produk disebut dengan sistem berbasis-aturan (*rule-based system*). Kondisi dapat terdiri atas banyak bagian, demikian pula dengan aksi. Urutan keduanya juga dapat dipertukarkan letaknya. (Nita Merlina & Rahmat Hidayat, 2012:19).

2.4.14. Metode Pengembangan Sistem Pakar

Metode perancangan yang digunakan dalam tugas akhir ini adalah menggunakan salah satu dari metode pengembangan perangkat lunak *software development life cycle* (SDLC) yaitu metode *waterfall* yang memiliki beberapa tahapan yaitu seperti pada Gambar 2.3.



GAMBAR : 2.3. Model Waterfall (Pressman, 2015:42)

Berikut ini adalah penjelasan dari tahapan-tahapan yang dilakukan di dalam Model *Waterfall* menurut pressman pada Gambar 2.3 :

1. *Communication (Project Initiation & Requirements Gathering)*

Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan *customer* demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek, seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi *software*. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, dan internet.

2. *Planning (Estimating, Scheduling, Tracking)*

Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan *tracking* proses pengerjaan sistem.

3. Modeling (*Analysis & Design*)

Tahapan ini adalah tahap perancangan dan permodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur *software*, tampilan *interface*, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

4. Construction (*Code & Test*)

Tahapan *Construction* ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

5. Deployment (*Delivery, Support, Feedback*)

Tahapan *Deployment* merupakan tahapan implementasi *software* ke *customer*, pemeliharaan *software* secara berkala, perbaikan *software*, evaluasi *software*, dan pengembangan *software* berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya. (Pressman, 2015:17).

2.4.15. Kelebihan dan Kekurangan Sistem Pakar

A. Kelebihan Sistem Pakar

Kelebihan yang didapatkan dengan adanya sistem pakar, antara lain :

1. Menjadikan pengetahuan dan nasihat lebih mudah didapat.
2. Meningkatkan output dan produktivitas.
3. Menyimpan kemampuan dan keahlian pakar.

4. Meningkatkan penyelesaian masalah – menerusi panduan pakar, penerangan, sistem pakar khas.
5. Meningkatkan realibilitas.
6. Memberikan respon (jawaban) yang cepat.
7. Merupakan panduan yang *intelligence* (cerdas).
8. Dapat bekerja dengan informasi yang kurang lengkap dan mengandung ketidakpastian.
9. *Intelligence database* (basis data cerdas), bahwa sistem pakar dapat digunakan untuk mengakses basis data dengan cara cerdas.
(Muhammad Arhami, 2005:9).

B. Kekurangan Sistem Pakar

Kekurangan dari sistem pakar diantaranya, yaitu :

1. Masalah dalam mendapatkan pengetahuan dimana pengetahuan tidak selalu bisa didapatkan dengan mudah. Karena kadangkala pakar dari masalah yang kita buat tidak ada, dan walaupun ada kadang-kadang pendekatan yang dimiliki pakar berbeda-beda.
2. Untuk membuat suatu sistem pakar yang benar-benar berkualitas tinggi sangatlah sulit dan memerlukan biaya yang sangat besar untuk pengembangan dan pemeliharaannya.
3. Boleh jadi sistem tak dapat membuat keputusan.
4. Sistem pakar tidaklah 100% menguntungkan, walaupun seorang tidak sempurna atau tidak terlalu benar. Oleh Karena itu perlu diuji ulang secara teliti sebelum digunakan.

Kelemahan-kelemahan atau kekurangan dari sistem pakar tersebut bukanlah sama sekali tidak bisa diatasi, tetapi dengan terus melakukan perbaikan dan pengolahan berdasarkan pengalaman yang telah ada maka hal itu diyakini akan dapat diatasi, walaupun dalam waktu yang panjang dan terus menerus.

2.5. Gangguan jiwa

Gangguan jiwa adalah bentuk dari manifestasi penyimpangan perilaku akibat distorsi emosi sehingga ditemukan tingkah laku dalam ketidak wajaran. Hal tersebut dapat terjadi karena semua fungsi kejiwaan menurun (Nasir, Abdul, & Muhith, 2011).

2.5.1. Kriteria Gangguan

Menurut Videbeck dalam Nasir, (2011) mengatakan bahwa kriteria umum gangguan adalah sebagai berikut :

1. Tidak puas hidup di dunia.
2. Ketidak puasan dengan karakteristik, kemampuan dan prestasi diri.
3. Koping yang tidak afektif dengan peristiwa kehidupan.
4. Tidak terjadi pertumbuhan personal.

2.6. Klasifikasi gangguan jiwa

Gangguan jiwa merupakan kumpulan dari keadaan-keadaan yang tidak normal. Keabnormalan tersebut dapat dibedakan menjadi :

1. Neurosis atau gangguan jiwa

Neurosis atau gangguan jiwa merupakan gangguan jiwa ditandai dengan kecemasan, biasanya gejala tidak tenang dan menekan lainnya. Sementara

pemeriksaan realitasnya tetap utuh (O'Brien, 2013). Orang yang terkena neurosis masih merasakan kesukaran, mengetahui, serta kepribadiannya tidak jauh dari realitas dan masih hidup dalam kenyataan pada umumnya (Yosep, H. Iyus & Sutini, 2014).

Neurosis memiliki karakteristik sebagai berikut :

- a. Uji realitas lengkap.
- b. Gejala kelompok yang mengganggu dan dikenal sebagai sesuatu yang asing dan tidak dapat diterima oleh individu.
- c. Gangguan cukup lama atau kambuh kembali jika tanpa pengobatan, bukan merupakan reaksi terhadap stressor, perilaku tidak mengganggu norma sosial dan tidak terlihat adanya penyebab dan faktor organik (Stuart, 2013).

2. Psikosis atau sakit jiwa

Psikosis atau sakit jiwa merupakan gangguan jiwa yang dapat menyebabkan individu mengalami gangguan nyata pada disintegrasi kepribadian berat, pemeriksaan realitas dan hambatan untuk memenuhi kebutuhan hidup sehari-hari (O'Brien, 2013). Orang yang terkena psikosis tidak memahami kejadiannya dan perasaan, segi tanggapan, dorongan, motivasi terganggu, kesukaran-kesukarannya dan tidak ada integritas mereka hidup jauh dari alam kenyataan (Yosep, H. Iyus & Sutini, 2014). Psikosis memiliki karakteristik sebagai berikut :

- a. Disintegrasi kepribadian.
- b. Penurunan bermakna pada tingkat kesadaran.
- c. Perilaku agresif.
- d. Kesulitan yang besar dalam berfungsi secara adekuat, kerusakan yang nyata atau berat pada realitas (Stuart, 2013).

2.7. Jenis Gangguan Jiwa

Berikut ini ialah jenis gangguan jiwa yang sering ditemukan di masyarakat menurut Nasir, (2011) adalah sebagai berikut :

1. *Skizofrenia*

Kelainan jiwa ini menunjukkan gangguan dalam fungsi kognitif atau pikiran berupa disorganisasi, jadi gangguannya adalah mengenai pembentukan psikis serta arus pikiran

2. Gangguan Bipolar

Gangguan Bipolar adalah gangguan mental yang ditandai dengan perubahan emosi yang drastis. Seseorang yang menderita bipolar dapat merasakan gejala mania (sangat senang) dan depresif (sangat terpuruk).

3. Psikopat

Psikopat atau secara medis dikenal dengan kepribadian anti-sosial, merupakan gangguan kepribadian yang ditandai dengan kurangnya empati dan kebiasaan

melanggar peraturan. Seorang psikopat biasanya tidak dapat berada dalam relasi yang harmonis dengan orang lain dan mudah melakukan kekerasan.

4. Gangguan Obsesif Kompulsif

Obsessive compulsive disorder (OCD) adalah gangguan mental yang menyebabkan penderitanya merasa harus melakukan suatu tindakan secara berulang-ulang. Bila tidak dilakukan, penderita OCD akan diliputi kecemasan atau ketakutan.

5. Skizoafektif

Gangguan skizoafektif merupakan gangguan jiwa ketika seseorang mengalami sekaligus kombinasi antara tanda dan gejala dari skizofrenia yang disertai tanda dan gejala dari gangguan suasana hati atau *mood*. Tanda dan gejala dari skizofrenia antara lain adalah halusinasi atau delusi. Sedangkan tanda dan gejala gangguan suasana hati misalnya depresi atau mania.

6. Anorexia Nervosa

Gangguan makan yang ditandai dengan rasa takut yang berlebihan bila berat badan bertambah, dan gangguan persepsi pada bentuk tubuh. Penderita anoreksia terobsesi untuk memiliki tubuh kurus, dan melakukan berbagai upaya untuk mendapatkan bentuk tubuh yang ideal menurut mereka.

7. Gangguan Kepribadian Ganda

Kepribadian ganda adalah kondisi di mana seseorang memiliki dua atau lebih kepribadian yang berbeda. Kepribadian ganda disebut juga gangguan identitas

disosiatif atau *dissociative identity disorder* (DID), dan umumnya disebabkan oleh pengalaman traumatis yang terjadi secara berulang di masa kanak-kanak.

8. Menyakiti Diri Sendiri

Self-injury adalah perilaku menyakiti dan melukai diri sendiri yang dilakukan secara sengaja. Ini merupakan salah satu bentuk dari gangguan perilaku yang terkait dengan sejumlah penyakit kejiwaan.

9. Gangguan Kecemasan

Generalized anxiety disorder adalah kecemasan kronis yang ditandai dengan rasa khawatir dan tegang yang berlebihan. Jenis gangguan kecemasan ini berlangsung secara terus-menerus dan cenderung tidak terkendali.

10. Gangguan Pemusatan Perhatian/Hiperaktif

ADHD atau attention deficit hyperactivity disorder adalah gangguan mental yang menyebabkan seorang anak sulit memusatkan perhatian, serta memiliki perilaku impulsif dan hiperaktif, sehingga dapat berdampak pada prestasi anak di sekolah.

11. Gangguan Stres Pascatrauma

PTSD (*post-traumatic stress disorder*) atau gangguan stres pascatrauma adalah gangguan mental yang muncul setelah seseorang mengalami atau menyaksikan peristiwa yang tidak menyenangkan.

12. Gangguan Perkembangan Pada Anak

Autism spectrum disorder (ASD) atau yang lebih sering disebut autisme merupakan gangguan perkembangan saraf yang memengaruhi perkembangan bahasa dan kemampuan seorang anak untuk berkomunikasi, berinteraksi, serta berperilaku. Bukan hanya autisme, ASD juga mencakup sindrom Asperger, sindrom Heller, dan gangguan perkembangan pervasif (PPD-NOS).

2.8. *Forward Chaining*

Algoritma *forward-chaining* adalah satu dari dua metode utama *reasoning* (pemikiran) ketika menggunakan *inference engine* (mesin pengambil keputusan) dan bisa secara logis dideskripsikan sebagai aplikasi pengulangan dari *modus ponens* (satu set aturan inferensi dan argumen yang valid).

Forward-chaining mulai bekerja dengan data yang tersedia dan menggunakan aturan-aturan inferensi untuk mendapatkan data yang lain sampai sasaran atau kesimpulan didapatkan. Mesin inferensi yang menggunakan *forward-chaining* mencari aturan-aturan inferensi sampai menemukan satu dari *antecedent* (dalil hipotesa atau klausa IF – THEN) yang benar. Ketika aturan tersebut ditemukan maka mesin pengambil keputusan dapat membuat kesimpulan, atau konsekuensi (klausa THEN), yang menghasilkan informasi tambahan yang baru dari data yang disediakan. Mesin akan mengulang melalui proses ini sampai sasaran ditemukan.

Forward-chaining adalah contoh konsep umum dari pemikiran yang dikendalikan oleh data (*data-driven*) yaitu, pemikiran yang mana fokus perhatiannya dimulai dari

data yang diketahui. *Forward-chaining* bisa digunakan dalam agen untuk menghasilkan kesimpulan dari persepsi-persepsi yang datang, seringkali tanpa query yang spesifik (Sharma, Tiwari, & Kelkar, 2012).

2.9. WEB

Website adalah kumpulan halaman *web* yang saling terhubung dan *file-file*-nya saling terkait. *Web* sendiri terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas, dengan halaman-halaman terkait berada dibawahnya. Biasanya setiap halaman dibawah *homepage* disebut *child page*, yang berisi *hyperlink* ke halaman lain dalam *web*(Gregorius, 2000:30).

2.10. Tinjauan Perangkat Lunak

Kebutuhan perangkat lunak adalah suatu kondisi atau keadaan yang harus dipenuhi atau dimiliki oleh suatu sistem, atau suatu kondisi keadaan yang diperlukan oleh pemakai untuk mencapai tujuan. Perangkat lunak yang digunakan dalam pembuatan aplikasi ini adalah perangkat lunak yang mendukung *web* beserta bahasa pemrogramannya sebagai berikut :

2.10.1. PHP (Hypertext Processing)

PHP (*Hypertext Processing*) merupakan skrip yang berjalan di *server* dan sangat populer di lingkungan *Linux*. PHP pun dapat diartikan dengan bahasa *server-side scripting* yang menyatu dengan HTML. PHP termasuk dalam *Open Source Product*, sehingga *source code* PHP dapat diubah dan didistribusikan secara bebas. PHP dapat berjalan pada berbagai *web server* seperti IIS (*Internet Information*

Server), *PWS (Personal Web Server)*, *Apache*, *Xitami*. PHP juga mampu lintas *platform*. Artinya PHP dapat berjalan di banyak sistem operasi yang beredar saat ini, diantaranya *Windows*, *Linux*, *Mac OS*, *Solaris*. (Abdul Kadir, 2003).

Salah satu keunggulan yang dimiliki oleh PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data/*Database Management System (DBMS)*. PHP mempunyai koneksitas yang baik dengan beberapa DBMS anatar lain *Oracle*, *Sybase*, *mSQL*, *MySQL*, *Microsoft SQL Server*, *Solid*, *Postgre SQL*, *Adabas*, *FilePro*, *Velocis*, *dBase*, *Unix*, *dbm*, dan tak terkecuali semua *database* ber-*interface* ODBC. PHP juga memiliki integrasi dengan beberapa *library eksternal* hingga dapat mem-*parse* XML. PHP mendukung komunikasi dengan layanan lain melalui protokol IMAP, SNMP, NNTP, POP3, atau bahkan HTTP.

2.10.2. XAMPP

XAMPP adalah sesuatu bundel *web server* yang populer digunakan untuk coba-coba di Windows karena memudahkan instalasinya. Bundel program *open source* tersebut berisi antara lain *server web*, *Apache*, *interpreter* PHP, dan basis data MySQL. Setelah menginstall XAMPP, kita bisa memulai pemograman PHP di komputer sendiri maupun mencoba menginstall aplikasi-aplikasi *web*. (Bunafit Nugroho, 2008:2).

2.10.3. MySQL

MySQL merupakan *database* yang berbasis *server*. Anda bisa menggunakan *database* MySQL apabila memiliki izin hak akses didalamnya. Hal

ini seperti halnya pada saat anda hendak menggunakan klien MySQL untuk masuk pada *server* MySQL. (Bunafit Nugrogo, 2008:29).

Keunggulan dari MySQL sebagai berikut :

1. Bersifat *open source*
2. Sistem *software*-nya tidak memberatkan kerja *server* atau komputer karena dapat bekerja di *background*.

2.10.4. CSS (Cascading Style Sheets)

CSS (*Cascading Style Sheets*) adalah sebuah cara untuk memisahkan isi dengan *layout* dalam halaman-halaman *web* yang dibuat. CSS memperkenalkan *template* yang berupa *style* untuk membuat dan mempermudah penulisan dari halaman-halaman yang dirancang. Hal ini sangat penting karena halaman yang menggunakan CSS dapat dibaca secara bolak balik dan isinya dapat dilihat oleh pengunjung dari manapun. CSS mampu menciptakan halaman yang tampak sama pada resolusi layar dari pengunjung yang berbeda tanpa memerlukan suatu tabel. Dengan CSS, anda akan lebih mudah melakukan *setting* tampilan keseluruhan *web* hanya dengan menggantikan atribut-atribut atau perintah dalam *style* CSS dengan atribut yang diinginkan tanpa harus mengubah satu per satu atribut tiap elemen yang ada dalam situs yang dibuat. Saat ini CSS merupakan *style* yang banyak digunakan karena berbagai kemudahan dan kelengkapan atribut yang dimilikinya. Pengguna CSS dalam *web* akan lebih efisien karena CSS dapat digunakan untuk penggunaan secara berulang pada tag-tag tertentu sehingga tidak usah mengetikkan ulang seluruh perintah pemformatan seperti halnya HTML klasik. CSS akan lebih mudah

dipelajari jika telah mengetahui struktur pembuatan dokumen *web* dengan bahasa HTML.(Sugiri dan Budi Kurniawan, 2007:21).

2.10.5. JQuery

Jquery merupakan *library* dari *JavaScript* yang fokus dalam meng-*query* obyek *JavaScript*. *Selector* yang digunakan dalam *Jquery* biasanya mengakses kelas CSS untuk menangkap DOM dan mengolahnya dengan *method* tertentu. Beberapa fitur yang menjadi inti dari *Jquery* adalah :

1. *Syntax* Ekspresif (*CSS selectors*) untuk menunjukkan *element* pada dokumen.
2. *Query* yang efektif dan efisien dalam menemukan *element* pada dokumen.
3. Kumpulan *method* yang berguna untuk memanipulasi *element* yang terpilih.
4. *Succinct Idom* (*method* berantai) untuk membuat *sequence* dari operasi yang dibuat (Flanagan, 2011, p1).

2.10.6. Bootstrap

Bootstrap merupakan sebuah *framework* CSS yang memudahkan pengembang untuk membangun *website* yang menarik dan *reponsif*. Tidak konsistensinya terhadap aplikasi individual membuat sulitnya untuk mengembangkan dan pemeliharaan. *Bootstrap* adalah CSS tetapi bentuk dengan *LESS*, sebuah *pre-processor* yang memberi fleksibilitas dari CSS biasa. *Bootstrap* memberikan solusi rapi dan seragam terhadap solusi yang umum, tugas *interface* yang setiap pengembang hadapi. *Bootstrap* dapat dikembangkan dengan tambahan

lainnya karena ini cukup fleksibel terhadap pekerjaan *design* butuhkan (M. Otto dan J. Thornton, 2011).

Keunggulan dalam menggunakan *Bootstrap* adalah semua bagian untuk antarmuka pengguna menggunakan *style CSS*, *bootstrap* dapat menggunakan LESS *pre-processor* sebuah teknologi yang mengurangi dan mengoptimalkan penulisan kode CSS. *Bootstrap* dapat diintegrasikan dengan *JavaScript* untuk menjadikan lebih menarik dengan efek-efek yang dapat diberikan dengan *JavaScript* (Tectale, 2012).

Kelemahan dalam menggunakan *Bootstrap* adalah dengan adanya penggunaan *bootstrap* menjadi tidak adanya keunikan didalam *website* karena akan samanya tampilan yang diberikan. Terdapat juga laporan bahwa sistem *grid* pada *bootstrap* tidak responsif (Tectale, 2012).

2.10.7. Apache Web Server

Web server adalah suatu *internet* yang menggunakan *protocol* HTTP (*Hypertext Transfer Protocol*) untuk melayani semua proses penransferan data. *Web server* hingga saat ini merupakan *server* yang dapat dikatakan sebagai tulang punggung bagi semua pengguna *internet* hal ini dikarenakan, *web server* bukan hanya dapat melayani jenis data dalam bentuk *text*, tetapi juga mampu menampilkan data berbentuk gambar dengan jenis 2D atau 3D, suara dan juga dapat berinteraksi dalam dunia *wireless internet* yang menjadikannya sebagai *Wireless Access Protocol* (WAP) *gateway* dan sebagainya. Salah satu aplikasi yang digunakan di *web server* adalah *apache*. *Apache* sendiri merupakan turunan dari

web server yang dikembangkan oleh NCSA (*National Center for Supercomputing Application*). *Apache* sendiri juga memiliki kelebihan dibanding *web server* lain diantaranya :

1. Kecepatan yang dimiliki lebih baik dibandingkan dengan aplikasi-aplikasi lain yang digunakan untuk *web server*.
2. *Performance* yang sangat baik
3. Dapat diperoleh dengan gratis. Selain itu *Apache* juga memiliki program pendukung yang dapat memberikan layanan yang cukup bagi penggunanya diantaranya kontrol akses, *common gateway interface*, PHP dan SSI (*Server Side Include*).

2.11. Entity Relationship Diagram (ERD)

ERD adalah diagram yang memperlihatkan *entitas-entitas* yang terlibat dalam suatu sistem serta hubungan-hubungan atau relasi antar *entitas* tersebut. Model *Entity-Relationship* yang berisi komponen-komponen himpunan *entitas* dan relasi yang masing-masing dilengkapi dengan *atribut-atribut* yang merepresentasikan seluruh fakta dari “*Dunia Nyata*” yang ditinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan diagram *Entity Relationship*. (Fathansyah dalam Joni, 2016).

ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data, pada dasarnya ada tiga macam komponen yang digunakan yaitu:


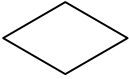

1. ***Entitas*** adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang

akan dibuat. Sebagai contoh pelanggan, pekerja dan lain-lain.

2. **Atribut** berfungsi mendeskripsikan karakter entiti. Misalnya atribut nama pekerja dari entiti pekerja. Setiap entiti bisa terdapat lebih dari satu atribut.
3. **Hubungan** atau (**Relationship**) sebagaimana halnya entiti maka dalam hubunganpun harus dibedakan antara hubungan atau bentuk hubungan antar entiti dengan isi dari hubungan itu sendiri. Misalnya dalam kasus hubungan antara entiti siswa dan entiti mata_kuliah adalah mengikuti, sedangkan isi hubungannya dapat berupa nilai_ujian. *Relationship* disimbolkan dalam bentuk intan / *diamonds*.

Simbol-simbol yang digunakan dalam *Entity Relationship Diagram* dapat dilihat pada Tabel 2.3.

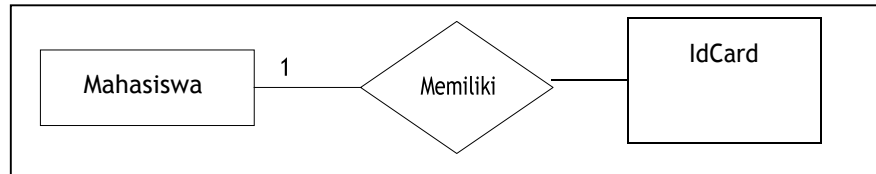
TABEL : 2.3. Simbol ERD (Fathansyah dalam Joni, 2016)

Simbol	Keterangan
	<p>Entitas Melambangkan himpunan entitas.</p>
	<p>Relasi Melambangkan himpunan relasi.</p>
	<p>Penghubung Melambangkan penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.</p>

Jenis relationship (Fathansyah dalam Joni, 2016):

A. *One – to – One*

Contoh:

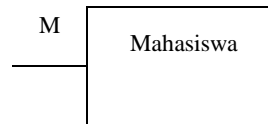


GAMBAR : 2.4. One to one (Fathansyah dalam Joni, 2016)

- 1) Setiap mahasiswa memiliki satu *IdCard*.
- 2) Satu *Id Card* hanya dimiliki oleh satu mahasiswa.

B. *One – to – Many (1:M atau M:1)*

Contoh:

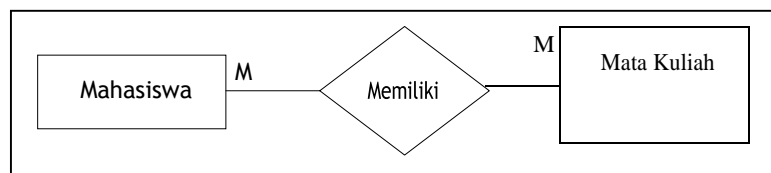


GAMBAR : 2.5. One to Many (Fathansyah dalam Joni, 2016)

- 1) Pembimbing akademik memiliki banyak mahasiswa.
- 2) Satu mahasiswa hanya memiliki satu pembimbing akademik.

C. *Many – to – Many* (M:M)

Contoh:



GAMBAR : 2.6. Many to Many (Fathansyah dalam Joni, 2016)

- 1) Mahasiswa memiliki banyak mata kuliah.
- 2) Mata kuliah dimiliki oleh banyak Mahasiswa.

Menentukan *primary key* merupakan salah satu hal yang penting, pertama tentukan suatu *attribute (field)* yang menyebabkan setiap *entry (record)* menjadi unik, kedua bila *attribute* tersebut tidak ada, tentukan gabungan beberapa *attribute* yang menyebabkan setiap *entry* menjadi unik, dan yang ketiga, pilih *primary key* dengan jumlah *attribute* yang paling sedikit. (Fathansyah dalam Joni, 2016).

2.12. *Flowchart*

Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program,. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut.(Indrajani, 2011:22).

Flowchart di bedakan menjadi 5 jenis *flowchart*, antara lain *system flowchart*, *document flowchart*, *schematic flowchart*, *program flowchart*, *process flowchart*. Masing-masing jenis *flowchart* akan dijelaskan berikut ini:

A. *System Flowchart*

System Flowchart dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alur sistem menunjukkan apa yang dikerjakan di sistem.

B. *Document Flowchart*

Bagan alur dokumen (*document flowchart*) atau disebut juga bagan alur formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alur yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

C. *Schematic Flowchart*

Bagan alur skematik (*schematic flowchart*) merupakan bagan alur yang mirip dengan bagan alur sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah, bagan alur skematik selain menggunakan simbol-simbol bagan alur sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaa gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagian alur. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarnya.

D. Program Flowchart

Bagan alur program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alur program dibuat dari derivikasi bagan alur sistem.

Bagan alur program dapat terdiri dari dua macam, yaitu bagan alur logika program (*program logic flowhart*) dan bagan alur program computer terinci (*detailed computer program flowchart*). Bagan alur logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program computer secara logika. Bagan alat logika program ini dipersiapkan oleh analurs sistem. Gambar berikut menunjukkan bagan alur logika program. Bagan alur program computer terinci (*detailed computer program flowchart*) digunakan untuk menggambarkan instruksi-instruksi program computer secara terinci. Bagan alur ini dipersiapkan oleh pemogram.

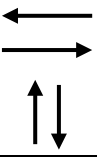


E. Process Flowchart

Bagan alur proses (*process flowchart*) merupakan bagan alur yang banyak digunakan di teknik industry. Bagan alur ini juga berguna bagi analurs sistem untuk menggambarkan proses dalam suatu prosedur. Berikut ini merupakan notasi atau symbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok yaitu:

A. Flow Direction Symbols (Simbol Penghubung/alur)

Simbol yang digunakan untuk menghubungkan antara symbol yang satu dengan yang lainnya. Simbol ini juga disebut *connecting line*. Simbol-simbol penhubung/alur terdapat pada tabel dibawah ini :


TABEL : 2.4. Tabel Simbol Penghubung/Alur *Flowchart*

No	<i>Symbol</i>	Nama	Keterangan
1		Arus / <i>Flow</i>	Untuk menyatakan jalannya arus suatu proses
2		<i>Connector</i>	Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman / lembaran sama
3		<i>Offline Connector</i>	Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembaran yang berbeda


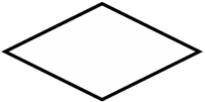
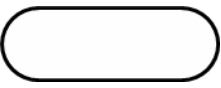
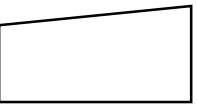
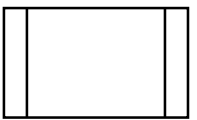
B. *Processing Symbols* (Simbol Proses)

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur. Simbol – simbol proses pada *flowchart* terdapat pada tabel dibawah ini :

TABEL : 2.5. Tabel Proses *Flowchart*

No	Symbol	Nama	Keterangan
1		Proses	Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi

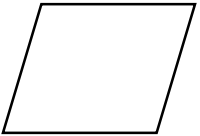
Lanjutan TABEL : 2.5. Tabel Proses *Flowchart*

No	Symbol	Nama	Keterangan
2		Symbol manual	Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer (manual)
3		<i>Decision / Logika</i>	Untuk menunjukkan suatu kondisi tertentu, dgn dua kemungkinan, YA / TIDAK
4		Terminal	Untuk menyatakan permulaan atau akhir suatu program
5		<i>Manual Input</i>	Untuk memasukkan data secara manual dengan menggunakan online keyword
6		<i>Predefined Process</i>	Untuk menunjukkan suatu bagian prosedur


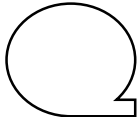


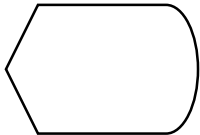
C. *Input/Output Symbol*

Simbol ini sering digunakan untuk menunjukan operasi pengolahan dalam suatu proses *input/output* di *flowchart*, akan dijelaskan pada tabel dibawah ini :

TABEL : 2.6. Tabel *Input/Output Flowchart*

No	<i>Symbol</i>	Nama	Keterangan
1		<i>Input/Output</i>	Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung jenis peralatannya.

Lanjutan TABEL : 2.6. Tabel *Input/Output Flowchart*

No	Symbol	Nama	Keterangan
2		<i>Punched Card</i>	Simbol menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu
3		<i>Magnetic Tape</i>	Simbol menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
4		<i>Disk Storage</i>	Simbol menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
5		<i>Document</i>	Simbol mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)
6		<i>Display</i>	Simbol mencetak keluaran dalam layar monitor

2.13. DFD (*Data Flow Diagram*)

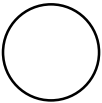

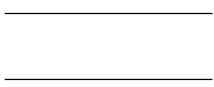
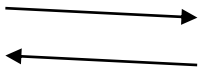
Menurut Kristanto (2008 : 61), “*Data Flow Diagram* merupakan suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan , proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.

Menurut Sukanto dan Shalahuddin (2014 : 288), “*Data Flow Diagram* atau dalam bahasa Indonesia menjadi *Diagram Alir Data* (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (*input*) dan keluaran

(*output*). DFD tidak sesuai untuk memodelkan sistem yang menggunakan pemrograman berorientasi obyek”.

Notasi-notasi pada DFD dapat dilihat pada Tabel 2.7 sebagai berikut :

TABEL : 2.7. Tabel Notasi DFD

No	Gambar	Nama	Keterangan
1		Proses	Simbol ini menyatakan transformasi dari masukan menjadi keluaran
2		Entitas Eksternal	Simbol ini menyatakan dimana entitas tersebut berkomunikasi dengan sistem
3		Penyimpanan	Simbol ini menyatakan penyimpanan dalam sebuah <i>database</i>
4		Aliran	Simbol ini menyatakan aliran data yang masuk ke proses atau keluar dari suatu proses

Menurut Sukamto dan Shalahuddin (2014 : 72), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD.

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram* DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0

digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1 untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya dibuat.
3. Membuat DFD Level 2 untuk modul-modul pada DFD Level 1 dapat di *berakdown* menjadi DFD Level 2. Modul mana saja yang baru di *breakdown* lebih *detail* tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup *detail* dan rinci maka modul tersebut sudah tidak perlu untuk di *berakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di *breakdown*.
4. Membuat DFD Level 3 dan seterusnya. DFD Level 3, 4, 5 dan seterusnya merupakan *breakdown* dari modul pada DFD Level di atasnya. *Breakdown* pada Level 3, 4, 5 dan seterusnya antaranya sama persis dengan DFD Level 1 atau Level 2.

2.14. Testing

2.14.1. Pengertian Testing

Menurut Quadri dan Farooq (2010 : 1), pengujian software adalah proses *verifikasi* dan *validasi* apakah sebuah aplikasi *software* atau program memenuhi persyaratan bisnis dan persyaratan teknis yang mengarahkan desain dan pengembangan dan cara kerjanya seperti yang diharapkan dan juga

mengidentifikasi kesalahan yang penting yang digolongkan berdasarkan tingkat *severity* pada aplikasi yang harus diperbaiki.

Menurut Nidhra dan Dondeti (2012:1), pengujian software adalah teknik yang sering digunakan untuk *verifikasi* dan *validasi* kualitas suatu *software*. Pengujian *software* adalah prosedur untuk eksekusi sebuah program atau sistem dengan tujuan untuk menemukan kesalahan.

2.14.2. Behavioral Test (Black-Box Test)

Menurut Black (2009 :3), *Tester* menggunakan *behavioral test* (disebut juga *Black-Box Tests*), sering digunakan untuk menemukan bug dalam *high level operations*, pada tingkatan fitur, profil operasional dan skenario *customer*. *Tester* dapat membuat pengujian fungsional black box berdasarkan pada apa yang harus sistem lakukan. *Behavioral testing* melibatkan pemahaman rinci mengenai domain aplikasi, masalah bisnis yang dipecahkan oleh sistem dan misi yang dilakukan sistem. Behavioral test paling baik dilakukan oleh penguji yang memahami desain sistem, setidaknya pada tingkat yang tinggi sehingga mereka dapat secara efektif menemukan bug umum untuk jenis desain.

Menurut Nidhra dan Dondeti (2012:1), *black box testing* juga disebut *functional testing*, sebuah teknik pengujian fungsional yang merancang *test case* berdasarkan informasi dari spesifikasi.

BAB III

ANALISIS MASALAH DAN PERANCANGAN PROGRAM

3.1. Communication

Pada tahap ini dilakukan proses pengumpulan informasi berupa data yang berkaitan dengan penelitian.

3.1.1. Pengumpulan Data

Pengumpulan data merupakan langkah yang dilakukan peneliti untuk mendapatkan data-data yang berkaitan dengan penyakit gangguan kejiwaan untuk mendukung penelitian ini yang sedang dilakukan peneliti. Teknik pengumpulan data yang dilakukan dalam penelitian ini adalah :

3.1.1.1. Metode Wawancara

Wawancara untuk mendapatkan data-data yang berkaitan dengan penelitian tentang penyakit gangguan kejiwaan

3.1.1.2. Studi Literatur

Peneliti melakukan studi literatur dengan mengumpulkan data-data dengan membaca, dan memahami referensi teoritis yang berasal dari buku-buku, jurnal-jurnal penelitian, dan sumber pustaka lainnya yang berkaitan dengan penelitian. Metode yang digunakan *Forward Chaining* dan informasi lainnya berkaitan dengan penelitian, berikut adalah tabel literatur yang menjadi referensi penelitian :

TABEL : 3.1 Referensi Penelitian

No	<i>Literature</i>	Pembahasan
1.	Tika Christy “ Implementasi Sistem Pakar Diagnosa Penyakit Cabe Menggunakan Metode <i>Forward Chaining</i> “ Seminar Nasional Royal (SENAR), 2018	Penelitian ini dibuat untuk mendeteksi penyakit tanaman cabe menggunakan metode <i>Forward Chaining</i> .
2.	Fersalina Indah Mevung, Addy Suyatno, Septya Maharani “ Diagnosis Penyakit Kejiwaan Menggunakan Metode <i>Certainty Factor</i> “ Prosiding Seminar Ilmu Komputer dan Teknologi Informasi Volume 2, No. 1 2017	Penelitian ini dibuat untuk mendiagnosa penyakit kejiwaan menggunakan metode <i>Certainty Factor</i> . Dalam penelitian ini sistem mendiagnosis dan memberi saran penanganan penyakit kejiwaan
3.	Rima Diah Wardhani, Rekyan Regasari Mardi Putri, Budi Darma Setiawan “ Sistem Pakar Diagnoss Penyakit Schizophrenia Menggunakan Metode <i>Bayesian Network</i> “ Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Volume 1, No. 11 2017	Penelitian ini dibuat untuk mendiagnosa penyakit kejiwaan menggunakan metode <i>Bayesian Network</i> . Sistem yang dibangun dalam penelitian ini memiliki akurasi pengujian 92,86%.
4.	Esti Rahmawati, Hari Wibawanto “ Sistem Pakar Diagnosis Penyakit Paru-Paru Menggunakan Metode <i>Forward Chaining</i> ” Jurnal Teknik Elektro Volume 8, No. 2 2016	Penelitian ini dibuat untuk mengetahui efektivitas metode <i>Forward Chaining</i> pada sistem pakar dan keakuratan sistem sebesar 84,21%.

Lanjutan TABEL : 3.1 Referensi Penelitian

No.	<i>Literature</i>	Pembahasan
5.	Landung Sudarmana, Febty Lestari “ Aplikasi Sistem Pakar untuk Mendiagnosis Gangguan Jiwa Schizophrenia “ Jurnal Pengembangan IT Volume 3 No. 1 2018	Penelitian ini dibuat untuk mengetahui penyakit gangguan kejiwaan Schizophrenia secara dini.

3.1.2. Analisis Sistem Pakar

Dalam membangun sebuah perangkat lunak sistem pakar untuk mendiagnosa penyakit kejiwaan berbasis web dilakukan beberapa tahap yaitu :

1. Menentukan masalah yang akan dibangun untuk sebuah perangkat lunak sistem pakar. Sistem pakar yang akan dibangun merupakan sebuah perangkat lunak sistem pakar untuk mendiagnosa penyakit kejiwaan berbasis web.
2. Mengumpulkan data yang diperlukan untuk membangun sistem, yaitu berupa informasi tentang pengertian penyakit, gejala, jenis penyakit dan cara pengobatannya melalui studi literatur dan observasi yang digunakan sebagai *base knowledge*.
3. Mempresentasikan pengetahuan ke dalam tabel gejala yang telah dianalisis, aturan produksi serta pohon pelacakan dan penelusuran gejala dan jenis penyakit.

4. Usulan sistem yang akan dibangun.

3.1.3. Analisis Identifikasi Masalah

Analisis masalah merupakan langkah dimana mengidentifikasi masalah yang akan dikaji, dalam hal ini adalah dengan mengidentifikasi permasalahan yang akan dibuat terlebih dahulu, adapun masalah-masalah yang akan diambil dalam pembangunan sistem pakar untuk mendiagnosa penyakit kejiwaan beserta solusi penanganannya.

Jenis penyakit dan ciri-ciri penyakit kejiwaan sebagai berikut :

1. *Skizofrenia*

Skizofrenia adalah gangguan mental yang terjadi dalam jangka panjang. Gangguan ini menyebabkan penderitanya mengalami halusinasi, delusi atau waham, kekacauan berpikir, dan perubahan perilaku. Gejala tersebut merupakan gejala dari psikosis, yaitu kondisi di mana penderitanya kesulitan membedakan kenyataan dengan pikirannya sendiri.

2. *Gangguan Bipolar*

Gangguan bipolar adalah gangguan otak yang menyebabkan perubahan yang tidak biasa dalam suasana hati, energi, tingkat aktivitas, dan memengaruhi kemampuan untuk melaksanakan aktifitas sehari-hari. Juga dikenal sebagai penyakit manik-depresif.

3. *Psikopat*

Psikopat, atau secara medis dikenal dengan kepribadian anti-sosial, merupakan gangguan kepribadian yang ditandai dengan kurangnya empati

dan kebiasaan melanggar peraturan. Seorang psikopat biasanya tidak dapat berada dalam relasi yang harmonis dengan orang lain dan mudah melakukan kekerasan.

4. *Gangguan Obsesif kompulsif*

Obsessive Compulsive Disorder (OCD) adalah gangguan perilaku kronis yang menyebabkan penyandanginya tidak memiliki kontrol atas pikiran-pikiran obsesifnya dan perilakunya yang kompulsif atau berulang-ulang. Dahulu, gangguan ini dimasukkan ke kategori gangguan kecemasan. Tapi saat ini telah menjadi kategori tersendiri.

5. *Skizoafektif*

Gangguan skizoafektif merupakan gangguan jiwa ketika seseorang mengalami sekaligus kombinasi antara tanda dan gejala dari skizofrenia yang disertai tanda dan gejala dari gangguan suasana hati atau *mood*. Tanda dan gejala dari skizofrenia antara lain adalah halusinasi atau delusi. Sedangkan tanda dan gejala gangguan suasana hati misalnya depresi atau mania.

6. *Anorexia nervosa*

Anoreksia nervosa adalah suatu gangguan pola makan yang tidak sehat yang ditandai dengan memiliki berat badan yang terlampau rendah untuk usia dan tinggi badan orang tersebut. Orang-orang yang menderita gangguan emosi ini mengalami ketakutan yang teramat sangat

terhadap kenaikan berat badan, bahkan saat mereka sebenarnya sudah sangat kurus.

7. Kepribadian Ganda

Gangguan identitas disosiatif adalah gangguan jiwa yang mengakibatkan terbentuknya dua atau lebih kepribadian yang berbeda. Masing-masing individu dengan ingatan sendiri, kepercayaan, perilaku, pola pikir, serta cara melihat lingkungan dan diri mereka sendiri. Setidaknya dua kepribadian ini secara berulang memegang kendali penuh atas tubuh si individu.

8. Menyakiti Diri Sendiri

Self-harm adalah ketika seseorang menyakiti diri sendiri sebagai cara untuk mengatasi, mengungkapkan, atau bertahan dari keadaan yang sangat sulit. Menyakiti diri dapat dilakukan secara fisik seperti, menyayat, mencakar, memukul, menggigit, membenturkan kepala ke dinding, menarik rambut, menelan sesuatu yang berbahaya, atau overdosis zat tertentu. Menyakiti diri juga dapat dilakukan secara halus seperti, tidak memerhatikan kondisi fisik, tidak memedulikan kebutuhan emosional, atau menempatkan diri pada situasi yang berbahaya.

9. Gangguan Kecemasan

Kecemasan kronis yang ditandai dengan rasa khawatir dan tegang yang berlebihan. Jenis gangguan kecemasan ini berlangsung secara terus-menerus dan cenderung tidak terkendali.

Orang dengan GAD bisa sangat gelisah meski sedang tidak berada dalam situasi yang menegangkan sekalipun, alias baik-baik saja. Rasa khawatir dan tegang yang berlebihan terkadang juga disertai dengan gejala fisik, seperti gelisah, sulit berkonsentrasi, dan bahkan kesulitan tidur (insomnia).

Meski sulit untuk menenangkan diri, orang yang sedang mengalami GAD umumnya hanya bisa merasakan kekhawatirannya sendiri tanpa mampu mengungkapkannya pada orang lain. Jika dibiarkan terus-terusan, kondisi ini bisa mengganggu aktivitas sehari-hari penderitanya

10. Gangguan Pemusatan Perhatian/Hiperaktif

ADHD adalah gangguan perkembangan saraf masa kanak-kanak yang paling umum. Kondisi ini biasanya didiagnosis pertama kali saat anak-anak dan bisa bertahan hingga dewasa.

11. Gangguan Stres pascatrauma

Post-traumatic stress disorder atau PTSD (gangguan stres pascatrauma) adalah kondisi mental di mana Anda mengalami serangan panik yang dipicu oleh trauma pengalaman masa lalu. Mengalami kejadian traumatis adalah hal yang berat bagi siapapun.

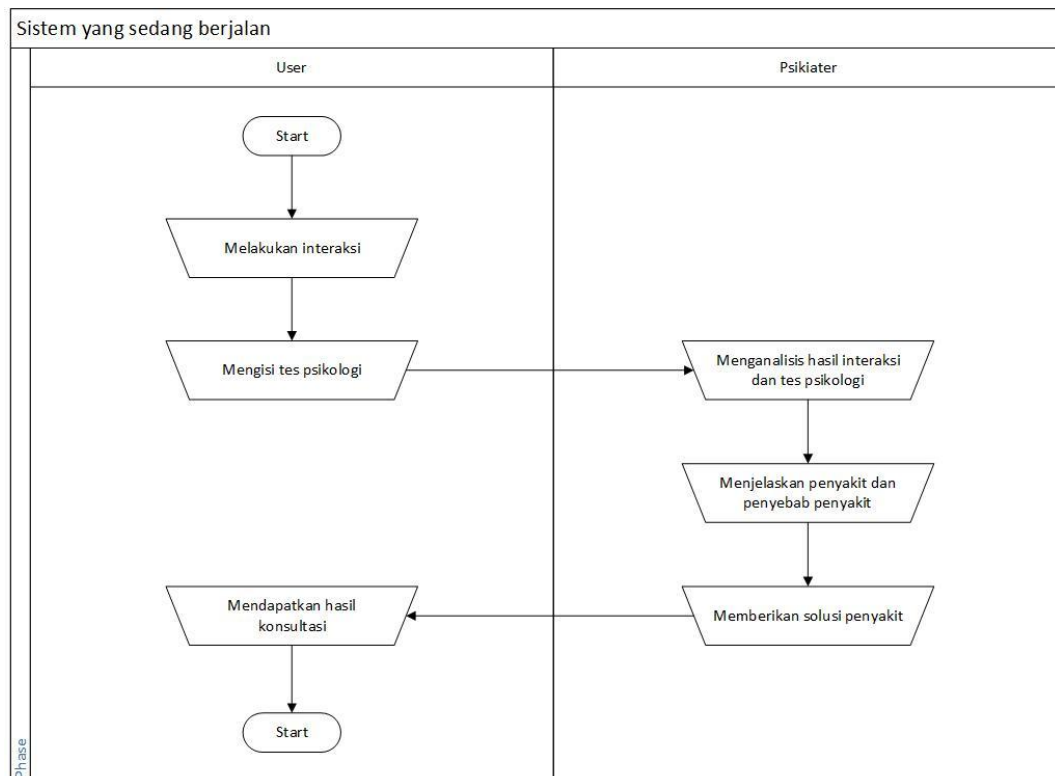
12. Gangguan Perkembangan pada Anak

Gangguan perkembangan pada otak dan saraf (neurologis) dapat memengaruhi kemampuan anak dalam berinteraksi, bersoliasiasi, berperilaku, dan berkomunikasi secara verbal maupun nonverbal, seiring ia bertumbuh besar.

3.1.4. Analisis Sistem Yang Sedang Berjalan

Sistem yang sekarang yang sedang berjalan pada Diagnosis penyakit kejiwaan yaitu sebagai berikut :

1. Orang yang akan melakukan konsultasi harus datang langsung ke psikiater.
2. Orang tersebut melakukan interkasi dengan psikiater.
3. Setelah melakukan interaksi, orang tersebut harus melakukan tes psikologi (psikotes).
4. Hasil interkasi dan tes psikologi (psikotes) tersebut di diagnosa oleh psikiater dan psikiater tersebut dapat menyimpulkan apakah orang yang konsultasi menderita penyakit kejiwaan atau tidak.
5. Orang yang berkonsultasi akan mendapatkan hasil dari kegiatan konsultasi yang dilakukan sebelumnya dengan psikiater.



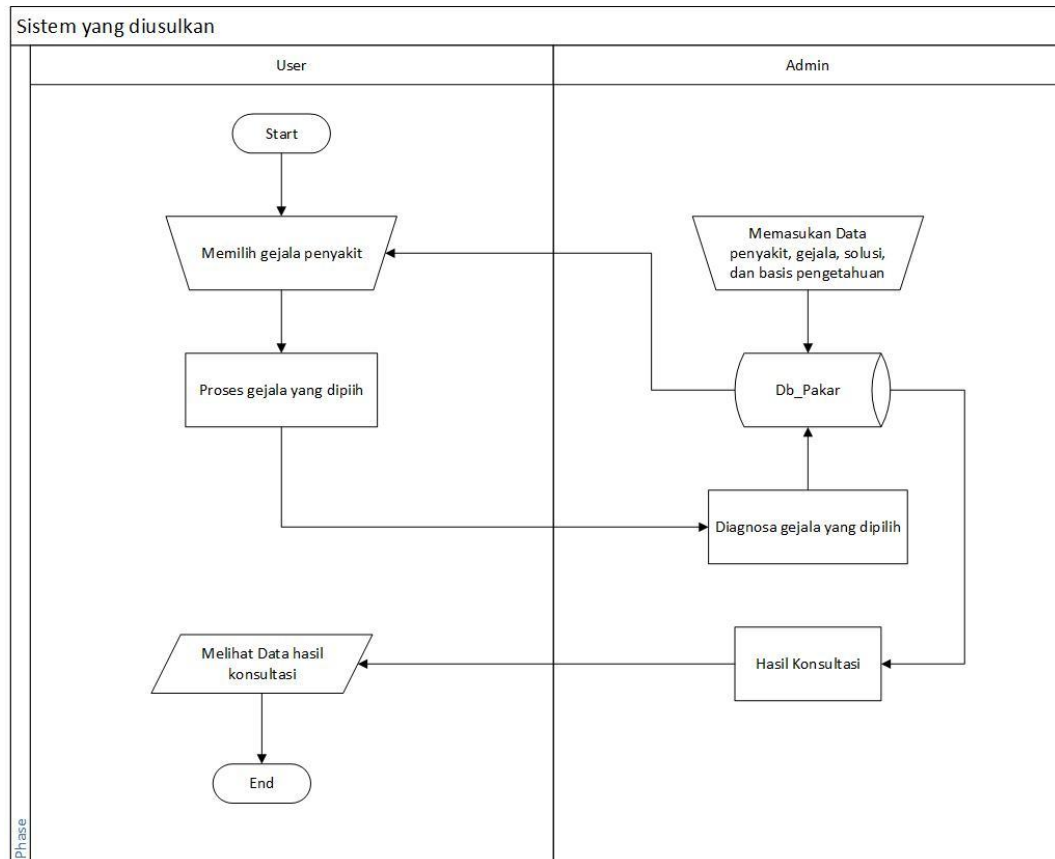
GAMBAR : 3.1. Flowmap Sistem Yang Sedang Berjalan

3.1.5. Analisis Sistem Diusulkan

Sistem pakar diagnosis penyakit kejiwaan menggunakan metode *forward chaining*.

1. Peserta konsultasi masuk ke halaman *web* diagnosa penyakit kejiwaan dan klik menu diagnosa.
2. Peserta menjawab pertanyaan yang ditampilkan.
3. Peserta konsultasi akan di berikan informasi dari hasil jawaban yang telah dilakukan sebelumnya. Jawaban dari konsultasi tadi berupa gejala dari penyakit kejiwaan. Peserta akan mendapatkan hasil berupa penyakit yang didiagnosis yang didapat dari gejala penyakit kejiwaan.

4. Peserta konsultasi akan diberikan informasi penyebab dan solusi penyakit yang didiagnosis oleh sistem.



GAMBAR : 3.2. Flowmap Sistem Pakar Yang Diusulkan

Analisis terhadap sistem yang berjalan bertujuan untuk mengetahui bagaimana cara kerja suatu sistem dan mengetahui masalah yang dihadapi sistem untuk dapat dijadikan sebagai landasan usulan perancangan. Tahap analisis diperlukan untuk mengetahui bagaimana proses yang bisa dijalankan pada sistem pakar mendiagnosa penyakit kejiwaan. Penulis menggambarkan sistem yang berjalan dalam bentuk *flowchart*. observasi, proses mendiagnosa penyakit kejiwaan pada sistem pakar ini dilakukan sebagai berikut :

1. Peserta konsultasi melakukan konsultasi dengan cara menjawab pertanyaan yang terdapat pada sistem pakar.
2. Setelah melakukan konsultasi peserta konsultasi akan mendapatkan hasil dari penyakit kejiwaan yang didiagnosis berdasarkan gejala penyakit kejiwaan.

3.1.6. Analisis Pengguna

Agar penggunaan aplikasi dapat berjalan secara optimal dan sesuai dengan yang diharapkan, maka dibutuhkan kemampuan untuk bisa menjalankan aplikasi sistem pakar yang di rancang oleh peneliti. Dalam hal ini, kemampuan tersebut terutama adalah dapat mengoperasikan sistem pakar dengan struktur pada aplikasi mudah untuk di operasikan oleh peserta konsultasi seperti orang/masyarakat yang mengalami gejala dari penyakit kejiwaan ataupun anggota keluarga dari pasien yang mengidap penyakit kejiwaan.

3.1.6.1. Peserta Konsultasi

Peserta konsultasi disini bertujuan untuk melakukan konsultasi terhadap perilaku keseharian peserta. Sistem pakar ini akan menampilkan hasil konsultasi yang telah dilakukan oleh peserta konsultasi.

3.1.6.2. Pakar

Pakar disini bertugas mengelola sistem, menambahkan, mengedit, dan menghapus gejala penyakit dan penyakit untuk sistem pakar yang dirancang ini.

3.1.6.3. Analisis Perangkat Keras

Analisis kebutuhan perangkat keras dimaksudkan untuk mengetahui spesifikasi perangkat keras yang sedang digunakan pada sistem pakar ini adalah sebagai berikut :

1. *Processor* dengan kecepatan minimal 1GHz + *Motherboard*
2. RAM minimal 256 MB
3. *Hard disk* minimal 5 GB
4. *VGA Card* minimal 64 MB
5. *Monitor*
6. *Mouse*
7. *Keyboard*

Secara keseluruhan spesifikasi perangkat keras semua yang ada sudah memenuhi syarat untuk kebutuhan yang akan di aplikasikan.

3.1.6.4. Analisis Perangkat Lunak

Perangkat lunak yang digunakan peserta konsultasi dan psikiater adalah sebagai berikut :

1. Perangkat Penelusuran seperti *google chrome*, *Mozilla Firefox*, dan *Internet Explorer*.

Perangkat lunak yang digunakan dalam perancangan aplikasi ini adalah sebagai berikut :

1. Sistem Operasi *Windows 7*
2. *MySQL* sebagai DBMS

3. *PHP*
4. *Jquery*
5. *Apache*
6. *Bootstrap*

Berdasarkan analisis perangkat lunak yang telah dijelaskan diatas sebagian besar sudah mencukupi untuk dapat menjalankan aplikasi yang akan dirancang.

3.2.5. Sumber Informasi

Data mengenai penyakit kejiwaan aplikasi sistem pakar diagnosis penyakit kejiwaan, terkait ciri-ciri penyakit, jenis penyakit, dan solusi penyakit tersebut didapatkan dari jurnal, website resmi serta data rumah sakit.

3.2. Planning

Dalam tahap ini penellitian memfokuskan pada penjadwalan pengerjaan penelitian. Pada penelitian ini terdapat beberapa proses yang harus dilakukan dari tahap *communication* hinggann *implementation* dan *testing* maka dari itu diperlukan penjadwalan yang tepat agar penelitian ini dapat selesai pada waktunya. Berikut penjadwalan penelitian berdasarkan aktifitas yang dilakukan dengan skala waktu.

Tabel 3.2 dibawah ini akan menjelaska penjadwalan penelitian :

TABEL : 3.2. Tabel Penjadwalan Penelitian

Tahap	No.	Aktivitas	Juni				Juli				Agustus			
			1	2	3	4	1	2	3	4	1	2	3	
<i>Scheduling</i>	1	Penjadwalan												
<i>Communication</i>	1	Pegumpulan Data												
	2	Analisis Masalah dan Kebutuhan												
	3	Pendefinisian Fungsi												
<i>Scheduling Estimating, and Tracking</i>	2	Analisis kebutuhan Perangkat Keras dan Perangkat Lunak												
<i>Analysis and Desgin</i>	1	Perancangan DFD dan ERD												
	2	Perancangan Interface												
<i>Construction</i>	1	Pengkodean												
		Testing												

3.3. Modeling

Peneliti telah melakukan proses akuisisi pengetahuan dengan mengumpulkan pengetahuan dan fakta dari sumber-sumber yang tersedia. Sumber pengetahuan dan fakta diperoleh melalui wawancara dengan pakar atau konselor dan studi literatur tentang materi yang berkaitan dengan penyakit kejiwaan. Sumber pengetahuan dan fakta yang didapat berupa data-data yang berhubungan dengan penyakit kejiwaan, gejala penyakit kejiwaan didapat berdasarkan penelitian sebelumnya, website resmi dan juga data rumah sakit.

TABEL : 3.3. Tabel Penyakit Kejiwaan

Kode Penyakit	Penyakit
P01	Penyakit Skizofrenia
P02	Penyakit Gangguan Bipolar
P03	Penyakit Psikopat
P04	Penyakit Gangguan Obsesif Komplusif
P05	Penyakit Skizoafektif
P06	Penyakit Anorexia Nervosa
P07	Penyakit Gangguan Kepribadian Ganda
P08	Penyakit Menyakiti Diri sendiri
P09	Penyakit Gangguan Kecemasan
P10	Penyakit Gangguan Pemusatan Perhatian/Hiperaktif
P11	Penyakit Gangguan Stress Pascatrauma
P12	Penyakit Gangguan Perkembangan pada Anak

TABEL : 3.4. Tabel Gejala Penyakit Kejiwaan

Kode Gejala	Gejala
G01	Cenderung mengasingkan diri
G02	Mudah marah dan depresi
G03	Kurang konsentrasi dan motivasi
G04	Suasana hati cepat berubah
G05	Kehilangan minat dalam aktivitas sehari-hari
G06	Sering berkhayal
G07	Mudah berbohong
G08	Melakukan tindakan tanpa memikirkan resikonya dengan matang
G09	Tidak memedulikan orang lain
G10	Pikiran ragu-ragu yang berlebihan dan keperluan untuk memastikan berulang-ulang
G11	Pikiran agresif tentang diri sendiri atau orang lain
G12	Mengulangi kata-kata atau kalimat
G13	Mendengar suara-suara tau melihat hal yang sebetulnya tidak ada
G14	Mengalami Depresi, seperti merasa hampa, sedih, atau tidak berharga

Lanjutan TABEL : 3.4. Tabel Gejala Penyakit Kejiwaan

Kode Gejala	Gejala
G15	Kesulitan dalam merawat diri sendiri, termasuk kebersihan dan penampilan fisik
G16	Ketakutan teramat sangat terhadap penambahan berat badan atau menjadi gemuk, bahkan saat sudah kurus
G17	Memuntahkan makanan secara sengaja
G18	Meminum obat-obatan yang menstimulasi buang air kecil dan buang air besar
G19	Merasa seperti adanya orang lain didalam tubuh
G20	Tidak mampu mengatasi emosi dengan baik
G21	Perubahan suasana hati yang tidak menentu
G22	Merasa tidak pantas untuk dicintai
G23	Kesulitan bersosialisasi
G24	Selalu menghindari masalah
G25	Mudah merasa lelah
G26	Gelisah terus-terusan
G27	Mudah tersinggung
G28	Sulit memusatkan perhatian
G29	Berprilaku implusif
G30	Hiperaktif

Lanjutan TABEL : 3.5. Tabel Rule

Kd	Penyakit											
	P 01	P 02	P 03	P 04	P 05	P 06	P 07	P 08	P 09	P 10	P 11	P 12
G09			√									
G10				√								
G11				√								
G12				√								
G13					√							
G14					√							
G15					√							
G16						√						
G17						√						
G18						√						
G19							√					
G20							√					
G21							√					
G22								√				
G23								√				
G24								√				
G25									√			
G26									√			

Lanjutan TABEL : 3.5. Tabel Rule

Kd	Penyakit											
	P 01	P 02	P 03	P 04	P 05	P 06	P 07	P 08	P 09	P 10	P 11	P 12
G27									√			
G28										√		
G29										√		
G30										√		
G31											√	
G32											√	
G33											√	
G34												√
G35												√
G36												√

3.3.1. Definisi Aktor

Sistem pakar ini diunakan oleh dua pengguna, yaitu pakar dan *user*, dimana kedua bagian tersebut memiliki wewenang dan hak akses yang berbeda.

a. Pakar

Pakar disini adalah seorang psikiater yang memiliki keahlian dalam ilmu psikiatri.

b. Pengguna

Peserta konsultasi disini adalah pengguna seorang pengguna yang akan melakukan konsultasi tentang penyakit kejiwaan. Peserta konsultasi memiliki hak akses hanya untuk melakukan konsultasi, dan melihat hasil konsultasi.

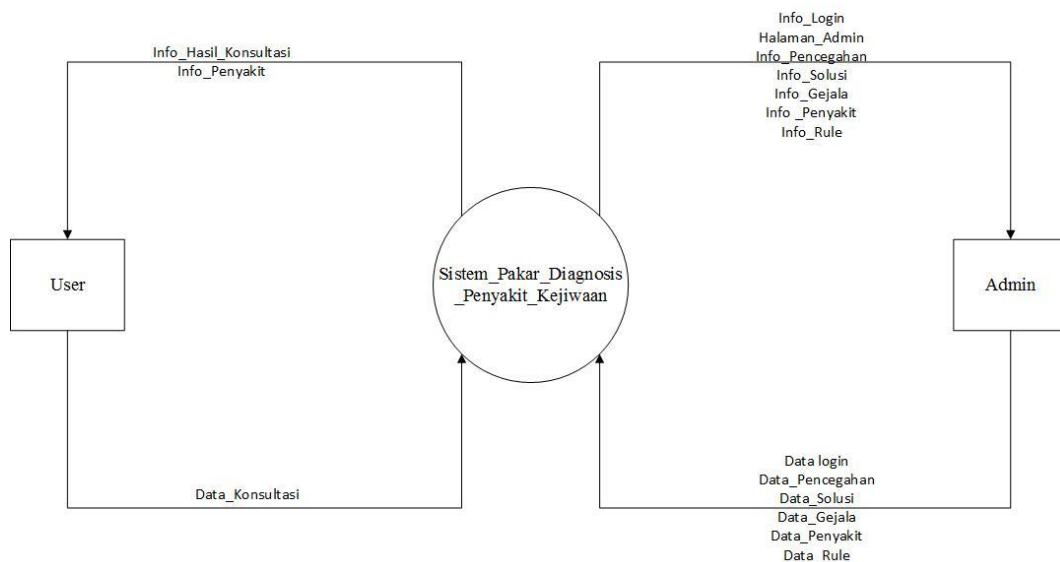
3.3.2. Perancangan Sistem

Perancangan ini bertujuan untuk memenuhi kebutuhan user mengenai gambaran yang jelas tentang perancangan sistem yang akan dirancang serta diimplementasikan. Dalam perancangan sistem yang digunakan penulis yaitu menggunakan *Data Flow Diagram* (DFD).

3.3.2.1. Context Diagram

Context Diagram sistem pakar ini yang terdiri dari beberapa *entity*, yang pertama *entity* admin, yang kedua *entity* user, dan yang ketiga adalah *entity* pakar.

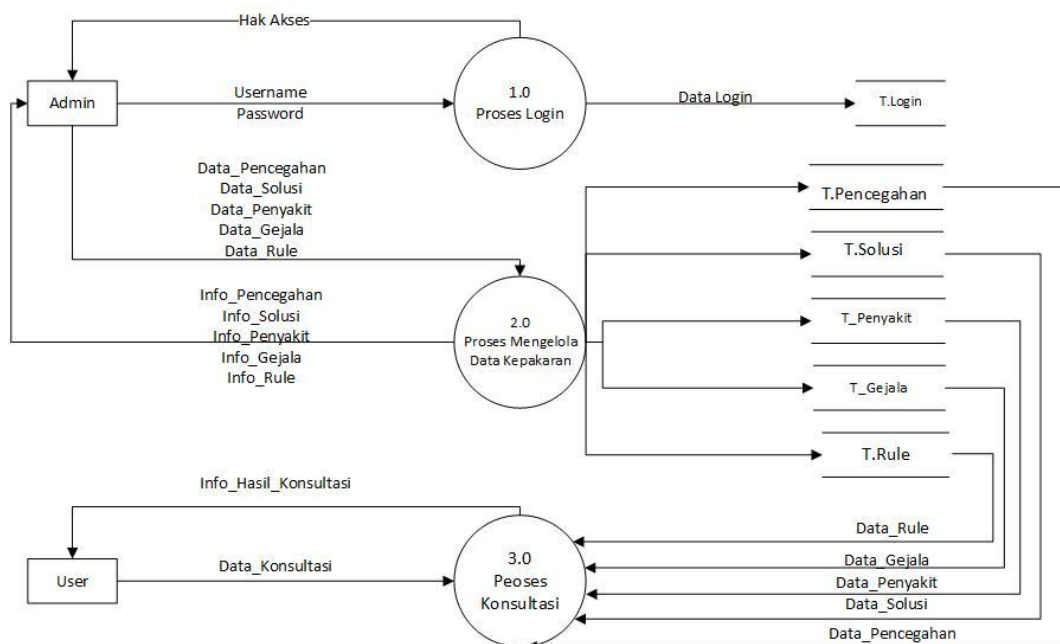
Context diagram yang dirancang oleh penulis terdapat pada Gambar 3.3. :



GAMBAR : 3.3. Context Diagram

3.3.2.2. DFD Level 1

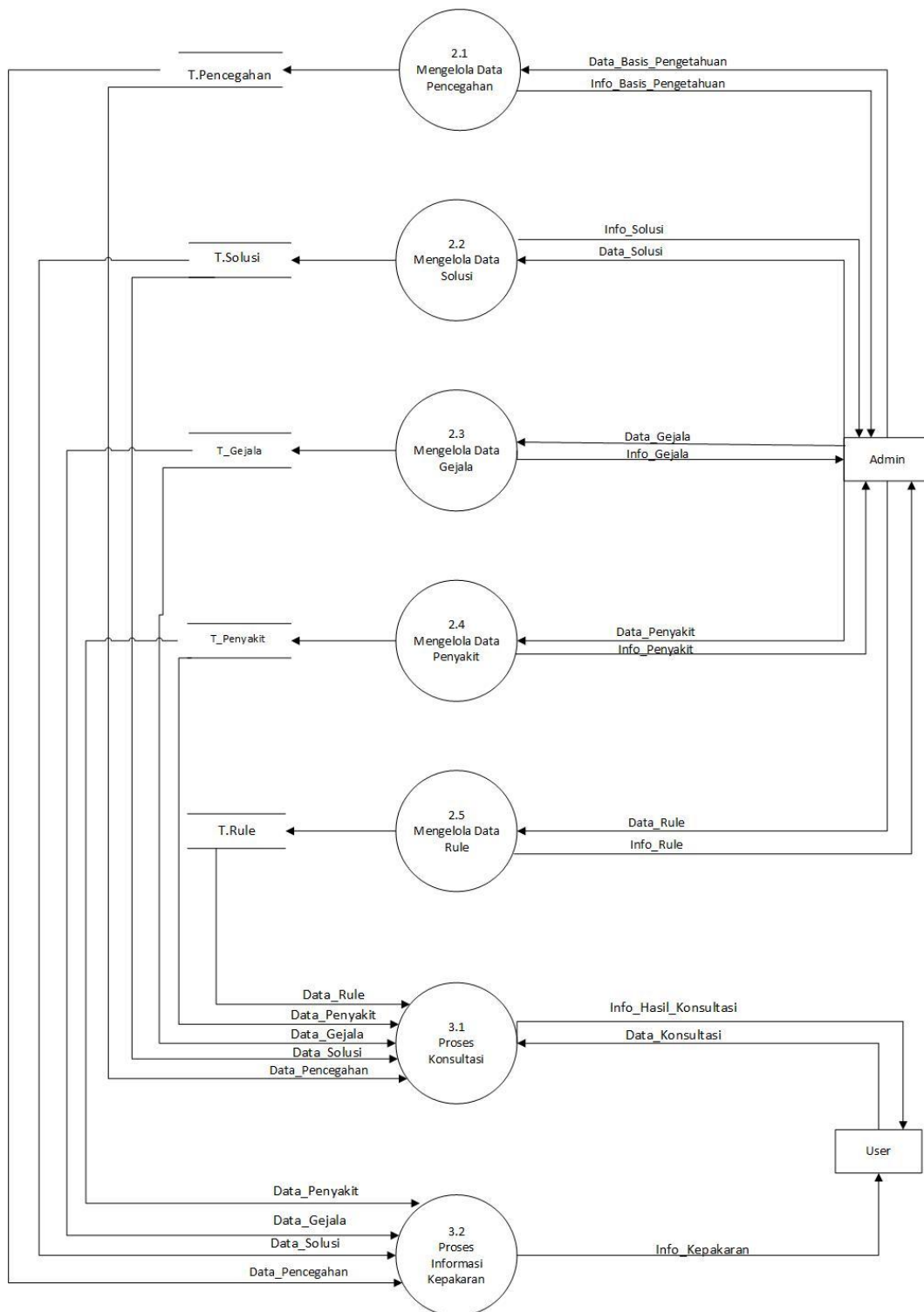
Gambar ini merupakan diagram sub-proses level 1. Pada diagram ini terdapat 3 (tiga) macam proses. Proses pertama adalah *login* untuk admin, kedua mengelola data kepakaran yang dilakukan oleh admin, ketiga proses konsultasi dilakukan oleh *user*. Berikut ini merupakan gambar perancangan DFD Level 1 :



GAMBAR : 3.4. DFD Level 1 Sistem Pakar Diagnosa Penyakit Kejiwaan

3.3.2.3. DFD Level 2

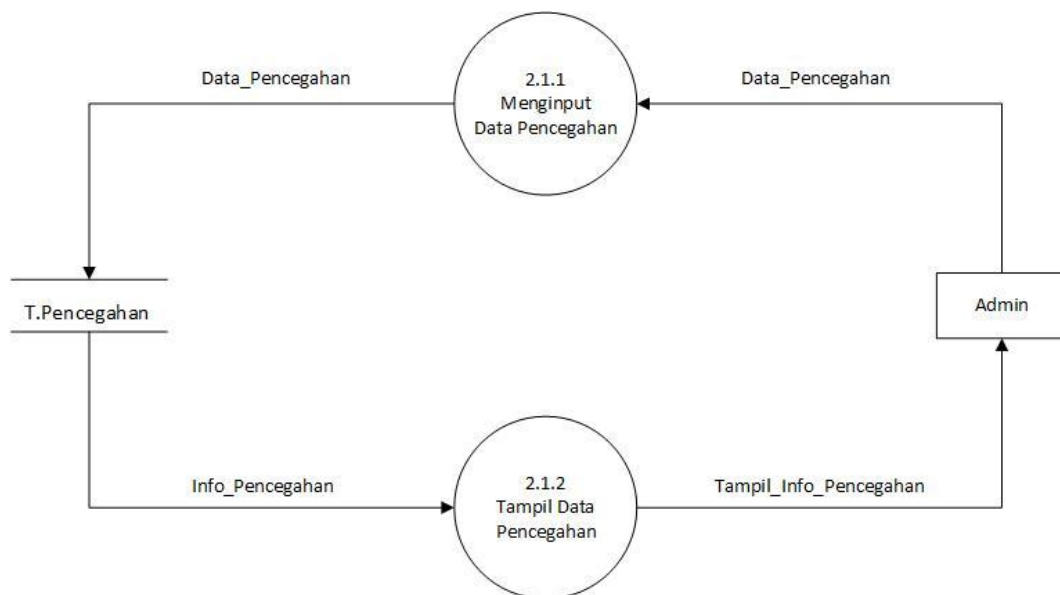
Gambar ini merupakan diagram sub-proses level 2. Pada diagram ini terdapat 5 (lima) macam proses. Proses pertama adalah proses mengelola data basis pengetahuan untuk admin, kedua proses mengelola data penyakit, ketiga proses mengelola data gejala, keempat proses konsultasi , dan kelima proses informasi kepakaran. Berikut ini merupakan gambar perancangan pada DFD Level 2:



GAMBAR : 3.5. DFD Level 2 Sistem Pakar Diagnosa Penyakit kejiwaan

3.3.2.4. DFD Level 3 (Data Pencegahan)

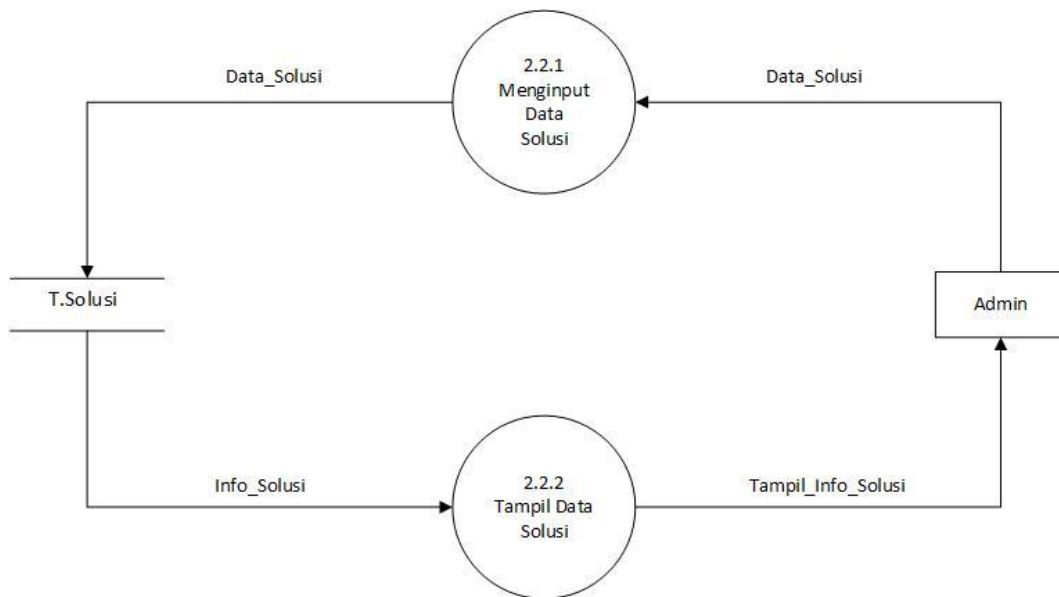
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah memasukan data pencegahan, dan kedua tampil data pencegahan. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.6. DFD Level 3 (Data Pencegahan)

3.3.2.5. DFD Level 3 (Data Solusi)

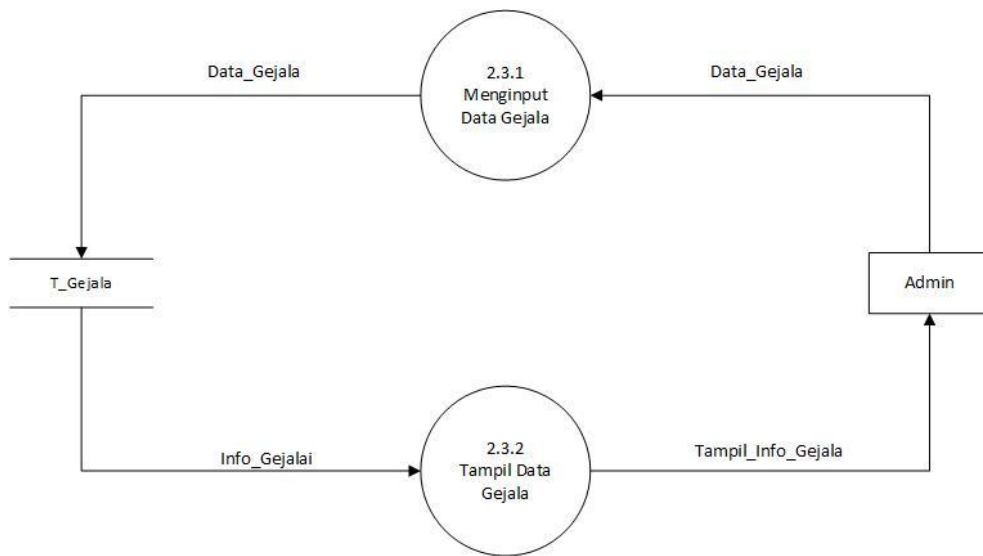
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah memasukan data solusi ,dan kedua tampil data solusi. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.7. DFD Level 3 (Data Solusi)

3.3.2.6. DFD Level 3 (Data Gejala)

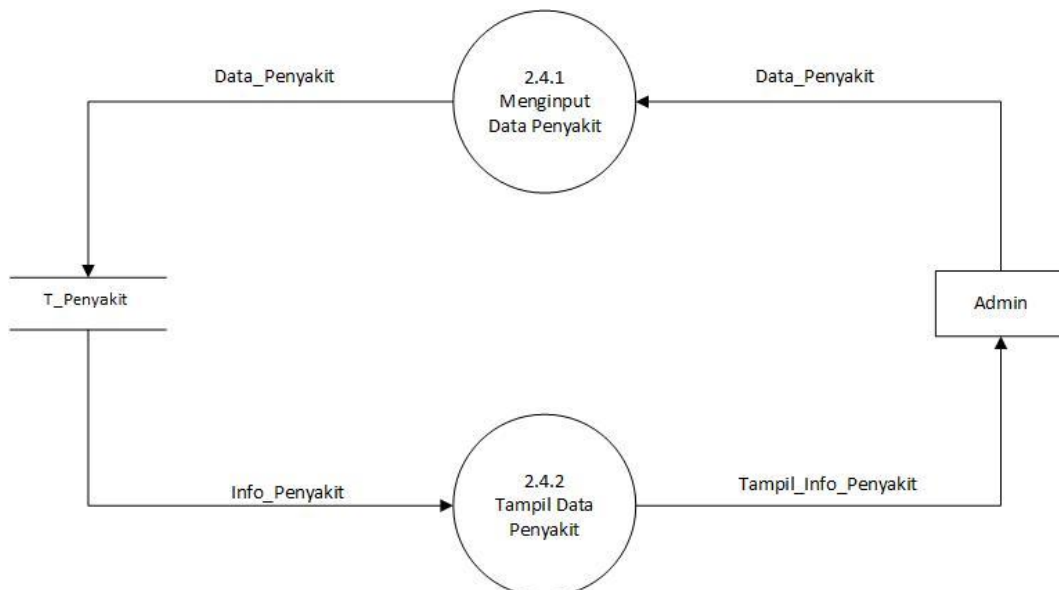
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah memasukan data gejala, dan kedua tampil data gejala. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.8. DFD Level 3 (Data Gejala)

3.3.2.7. DFD Level 3 (Data Penyakit)

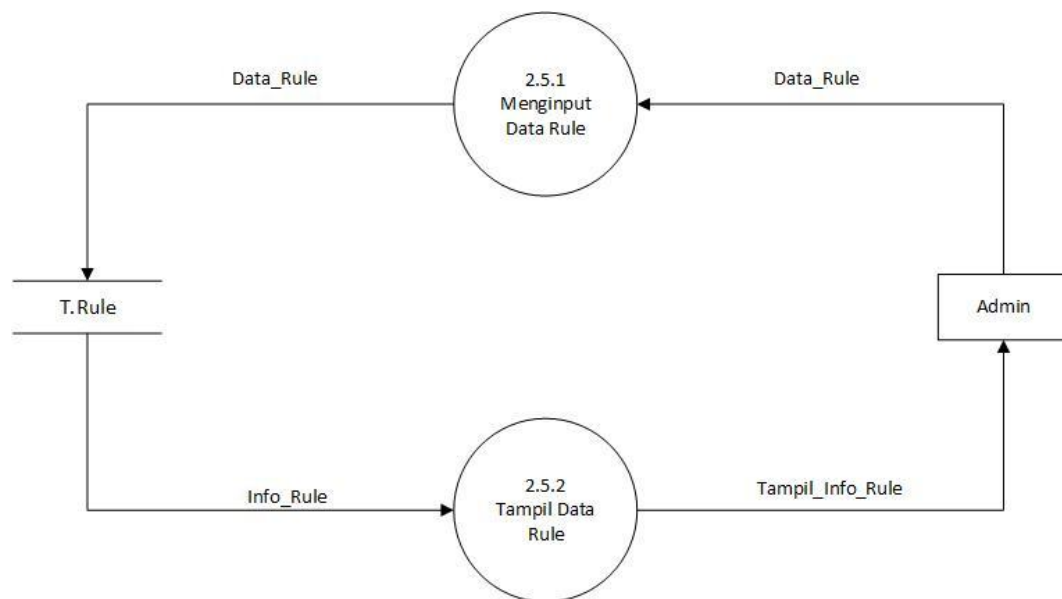
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah memasukan data penyakit, dan kedua tampil data penyakit. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.9. DFD Level 3 (Data Penyakit)

3.3.2.8. DFD Level 3 (Data Rule)

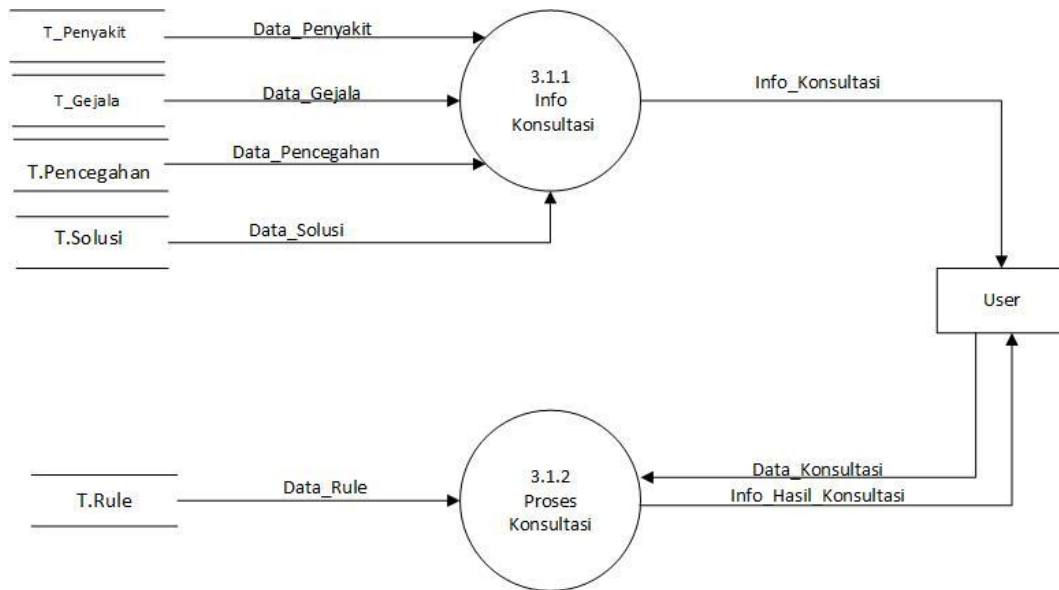
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah memasukan data rule, dan kedua tampil data rule. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.10. DFD Level 3 (Data Rule)

3.3.2.9. DFD Level 3 Konsultasi

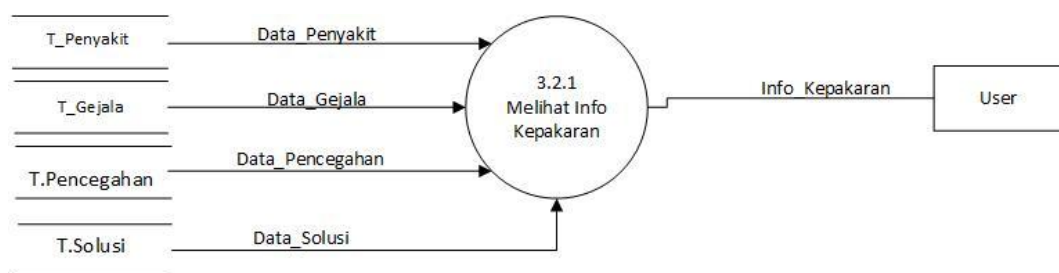
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 2 (dua) macam proses. Proses pertama adalah melihat pertanyaan konsultasi, dan kedua konsultasi. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.11. DFD Level 3 (Konsultasi)

3.3.2.10. DFD Level 3 Informasi Kepakaran

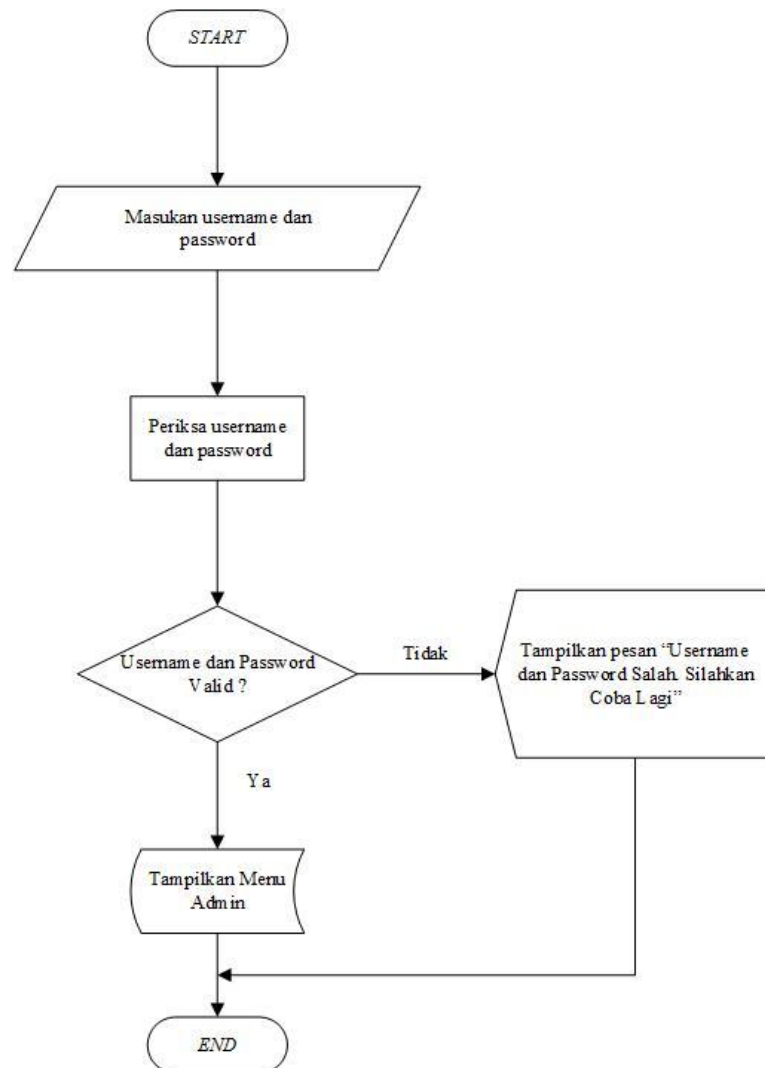
Gambar ini merupakan diagram sub-proses level 3. Pada diagram ini terdapat 1 (satu) macam proses. Yaitu proses melihat informasi kepakaran. Berikut ini merupakan gambar perancangan pada DFD Level 3:



GAMBAR : 3.12. DFD Level 3 (Informasi Kepakaran)

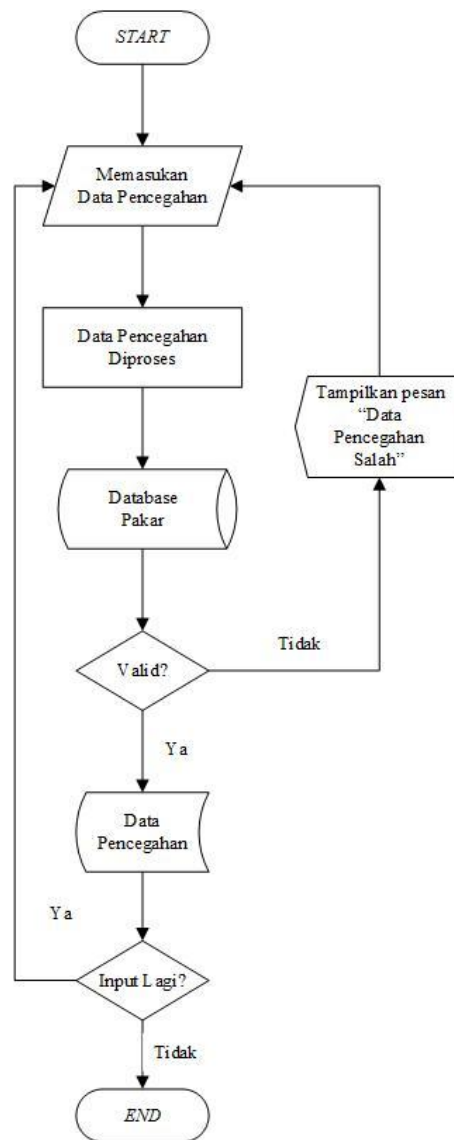
3.3.2.11. Flowchart Diagram DFD

Pada gambar diagram sub-proses level 1 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 1 (Login) ini, yaitu sebagai berikut :



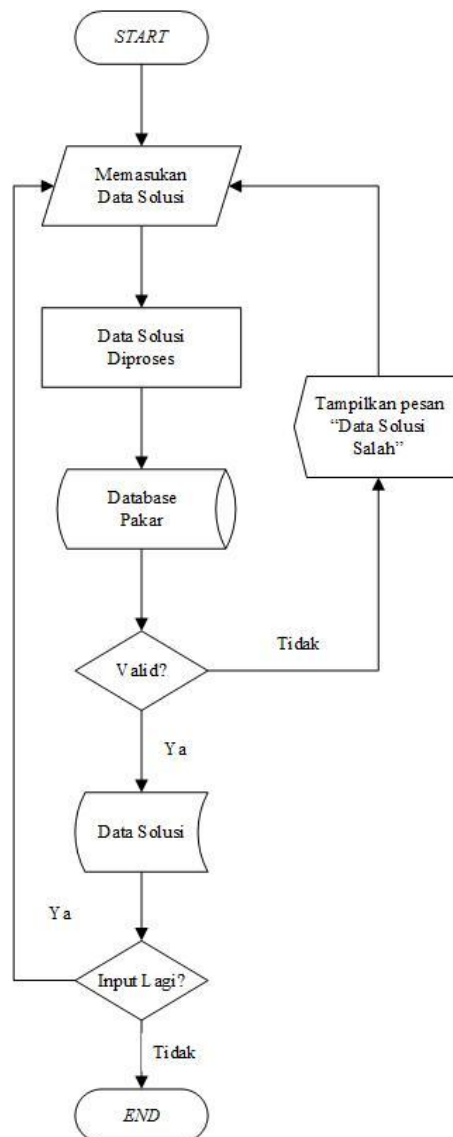
GAMBAR : 3.13. Flowchart Login

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Data Pencegahan) ini, yaitu sebagai berikut:



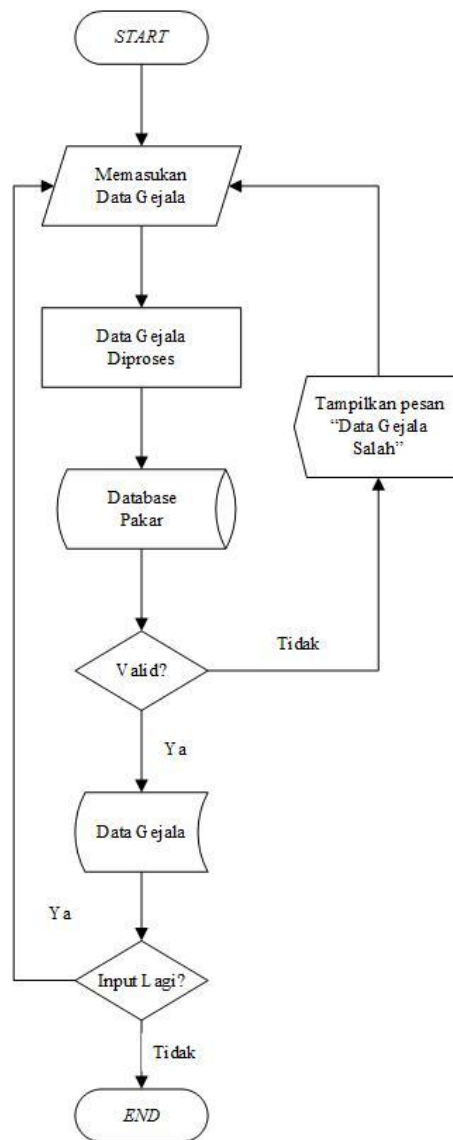
GAMBAR : 3.14. Flowchart Kelola Data Pencegahan

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Data Gejala Solusi) ini, yaitu sebagai berikut :



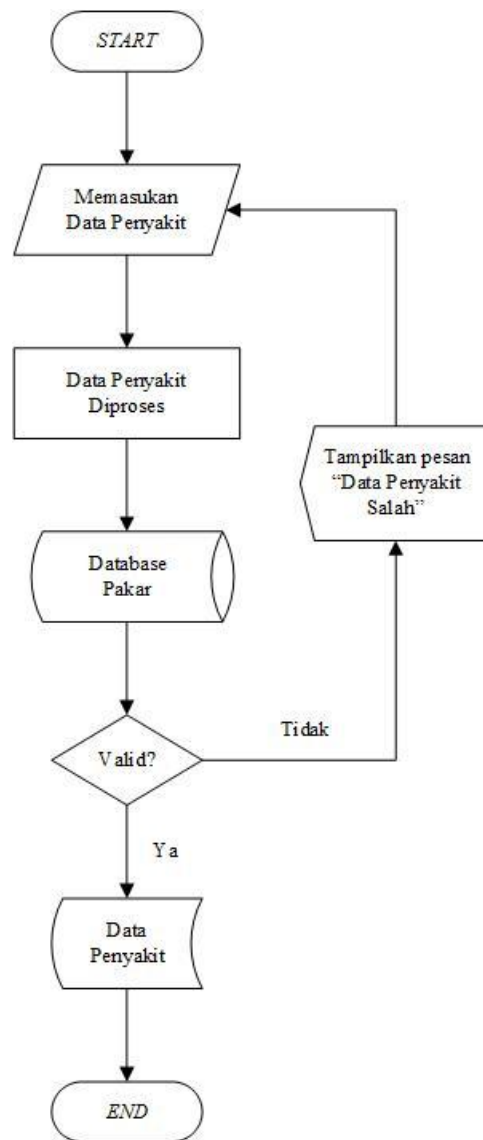
GAMBAR : 3.15. Flowchart Kelola Data Solusi

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Data Gejala) ini, yaitu sebagai berikut :



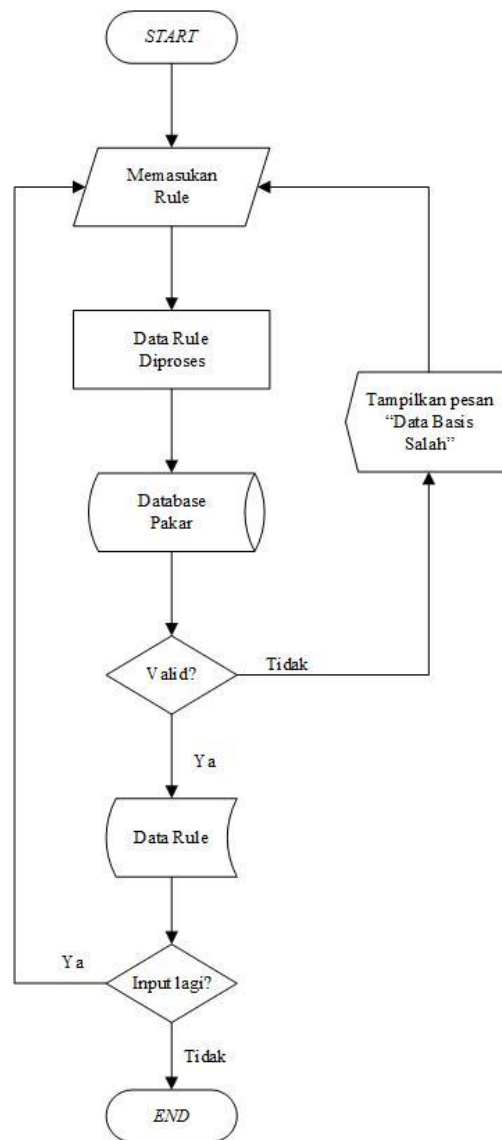
GAMBAR : 3.16. Flowchart Kelola Data Gejala

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Data Penyakit) ini, yaitu sebagai berikut :



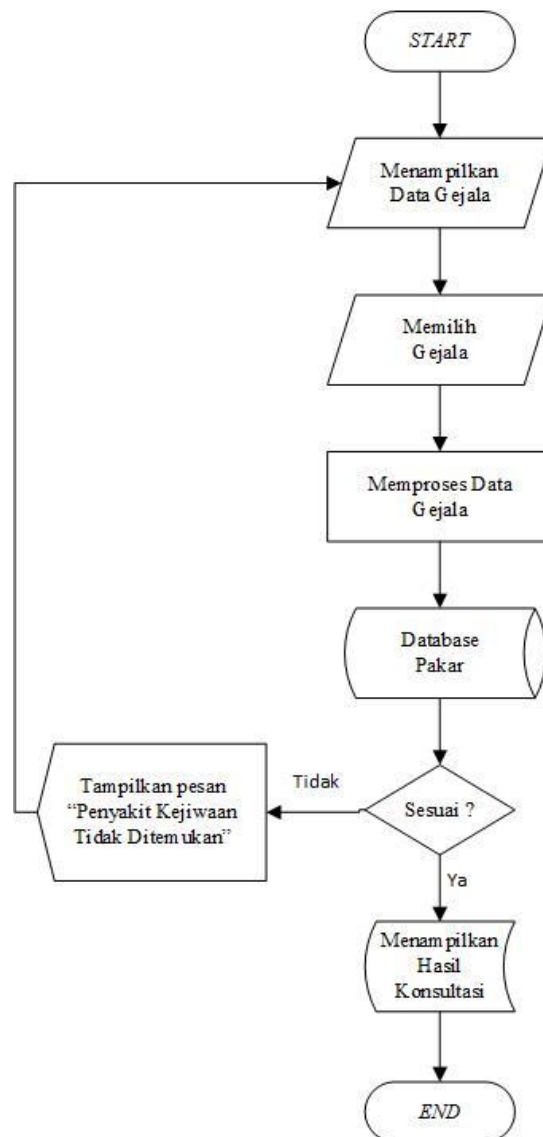
GAMBAR : 3.17. Flowchart Data Penyakit

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Data Rule) ini, yaitu sebagai berikut :



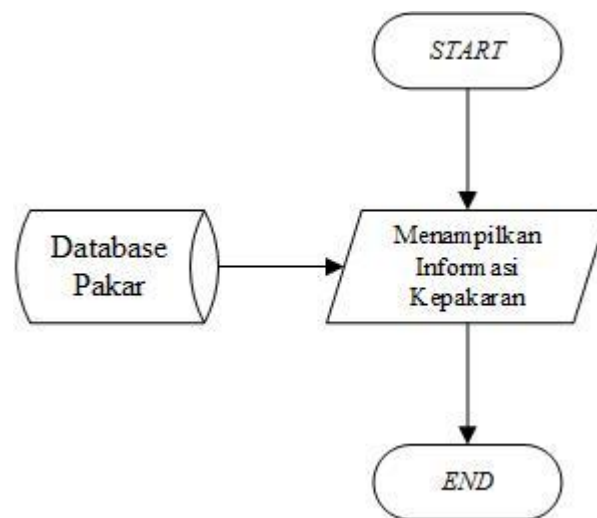
GAMBAR : 3.18. Flowchart Kelola Data Rule

Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Konsultasi) ini, yaitu sebagai berikut :



GAMBAR : 3.18. Flowchart Konsultasi

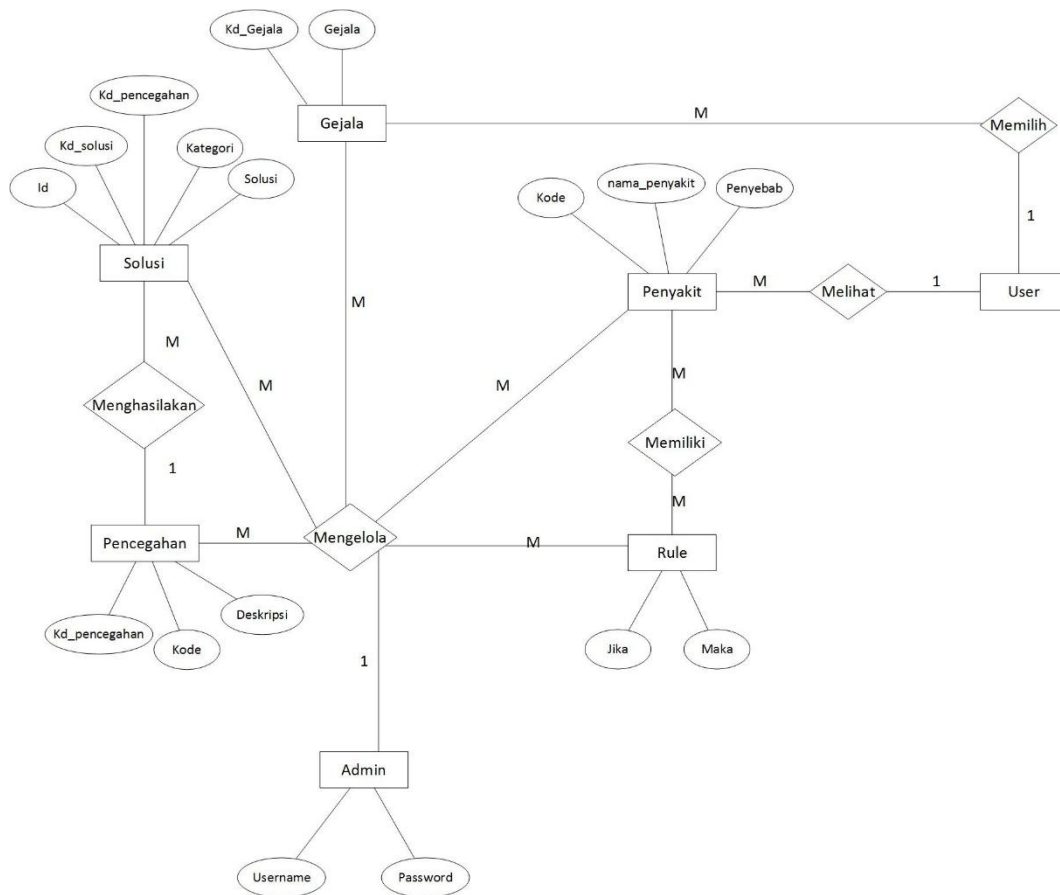
Pada gambar diagram sub-proses level 3 ini terdapat *flowchart* untuk menjelaskan proses dari DFD Level 3 (Informasi Kepakaran) ini, yaitu sebagai berikut :



GAMBAR : 3.20. *Flowchart Informasi Keahlian*

3.3.2.10. ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah salah satu model yang digunakan untuk merancang database dengan tujuan menggambarkan data yang berelasi pada sebuah database. Pada gambar dibawah ini adalah ERD yang dirancang :



GAMBAR : 3.21. ERD Sistem Pakar Diagnosa Penyakit Kejiwaan

Terdapat lima tabel pada aplikasi ini dimana ada tabel penyakit, tabel gejala, tabel rule, tabel solusi, dan tabel pencegahan. Tabel berikut ini akan menjelaskan isi dari tabel penyakit pada sistem yang dirancang.

TABEL : 3.6. Tabel Penyakit

Nama	Tipe Data	Ukuran	Key
Id	Int	11	<i>primary key</i>
Kode	Varchar	10	-
Nama_penyakit	Varchar	50	-
penyebab	Varchar	200	-

Tabel berikut ini akan menjelaskan isi dari tabel gejala pada sistem yang dirancang.

TABEL : 3.7. Tabel Gejala

Nama	Tipe Data	Ukuran	Key
Id	Int	11	<i>primary key</i>
Kd_gejala	Varchar	10	-
Gejala	Text		-

Tabel berikut ini akan menjelaskan isi dari tabel rule pada sistem yang dirancang.

TABEL : 3.8. Tabel Rule

Nama	Tipe Data	Ukuran	Key
Id	Int	11	<i>primary key</i>
Jika	Varchar	50	-
Maka	Varchar	10	-

Tabel berikut ini akan menjelaskan isi dari tabel solusi pada sistem yang dirancang.

TABEL : 3.9. Tabel Solusi

Nama	Tipe Data	Ukuran	Key
Id	Int	11	<i>primary key</i>
Kd_solusi	Varchar	10	-
Kd_pencegahan	Varchar	10	-
Kategori	Text	-	-
Solusi	Text	-	-

Tabel berikut ini akan menjelaskan isi dari tabel pencegahan pada sistem yang dirancang.

TABEL : 3.10. Tabel Pencegahan

Nama	Tipe Data	Ukuran	Key
Id	Int	11	<i>primary key</i>
Kd_pencegahan	Varchar	10	-
Kode	Varchar	10	-
Deskripsi	Longtext	-	-

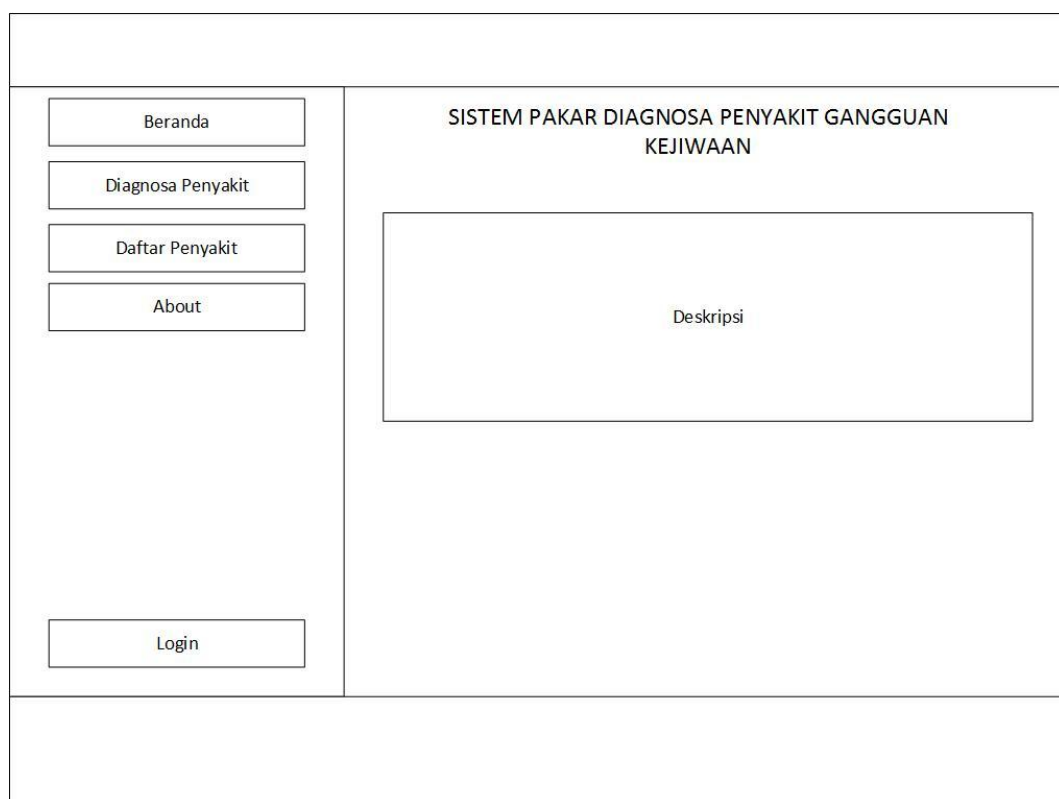
3.3.3. Design Interface

Design Interface merupakan rancangan antarmuka yang akan digunakan sebagai perantara *user* dengan perangkat lunak yang dikembangkan. *Design*

interface dari Sistem Pakar Diagnosa Penyakit Kejiwaan Berbasis Web ini dapat dilihat pada gambar dibawah ini :

1. Perancangan antarmuka sistem pakar diagnosa penyakit kejiwaan

Pada gambar ini merupakan rancangan antarmuka yang akan dibuat di sistem pakar diagnosa penyakit kejiwaan tampilan menu awal/home.



GAMBAR : 3.22. Rancangan Menu Awal Sistem Pakar Diagnosa Penyakit Kejiwaan

2. Perancangan antarmuka Konsultasi Penyakit kejiwaan

Pada gambar ini merupakan rancangan antarmuka untuk memulai proses diagnosa penyakit kejiwaan.

The image shows a web application interface for disease diagnosis. It is divided into two main sections: a left sidebar and a main content area.

Left Sidebar:

- Beranda
- Diagnosa Penyakit
- Daftar Penyakit
- About
- Login

Main Content Area:

DIAGNOSA PENYAKIT

- Gejala 1
- Gejala 2
- Gejala 3
- Gejala 4
- Gejala 5

Cek Diagnosa

GAMBAR : 3.23. Rancangan Menu Diagnosa Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

3. Perancangan antarmuka hasil diagnosa

Pada gambar ini merupakan tampil antarmuka untuk melihat hasil diagnosa sistem pakar diagnosa penyakit kejiwaan.

HASIL DIAGNOSA PENYAKIT	
Beranda	Gejala : <input type="text"/>
Diagnosa Penyakit	Penyakit : <input type="text"/>
Daftar Penyakit	Penyebab : <input type="text"/>
About	Pencegahan : <input type="text"/>
	Kategori : <input type="text"/>
	Solusi : <input type="text"/>
	<input type="button" value="Kembali"/>
Login	

GAMBAR : 3.24. Rancangan Hasil dari Konsultasi Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

4. Perancangan form penyakit kejiwaan

Pada gambar ini merupakan tampilan antarmuka untuk memasukan data penyakit kejiwaan.

INPUT PENYAKIT	
KODE :	<input type="text"/>
PENYAKIT :	<input type="text"/>
PENYEBAB :	<input type="text"/>
<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>	

Navigation sidebar (left):

- Beranda
- Penyakit
- Gejala
- Basis Pengetahuan
- Pencegahan
- Solusi
- Log Out

GAMBAR : 3.25. Rancangan Memasukan Data Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

5. Perancangan form gejala penyakit kejiwaan

Pada gambar ini merupakan tampilan antarmuka untuk memasukan data gejala penyakit kejiwaan.

INPUT GEJALA	
Beranda	KODE GEJALA : <input type="text"/>
Penyakit	GEJALA : <input type="text"/>
Gejala	
Basis Pengetahuan	
Pencegahan	<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>
Solusi	
Log Out	

GAMBAR : 3.26. Rancangan Dari Memasukan Data Gejala Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Kejiwaan

6. Perancangan anatrmuka form rule/basis pengetahuan

Pada perancangan ini merupakan form memasukan data rule.

The image shows a web application interface for entering knowledge base rules. It consists of a left sidebar with navigation buttons and a main content area for rule input.

Left Sidebar (Navigation):

- Beranda
- Penyakit
- Gejala
- Basis Pengetahuan
- Pencegahan
- Solusi
- Log Out

Main Content Area (INPUT BASIS PENGETAHUAN):

JIKA :

MAKA :

Buttons:

GAMBAR : 3.27. Rancangan Hasil dari Memasukan Data Rule Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

7. Perancangan antarmuka form pencegahan penyakit kejiwaan

Pada perancangan ini merupakan form memasukan data pencegahan penyakit kejiwaan.

INPUT PENCEGAHAN	
Beranda	KODE PENCEGAHAN : <input type="text"/>
Penyakit	KODE PENYAKIT : <input type="text"/>
Gejala	PENCEGAHAN : <input type="text"/>
Basis Pengetahuan	
Pencegahan	
Solusi	
	<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>
Log Out	

GAMBAR : 3.28. Rancangan Hasil dari Memasukan Data Pencegahan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

8. Perancangan antarmuka form solusi penyakit kejiwaan

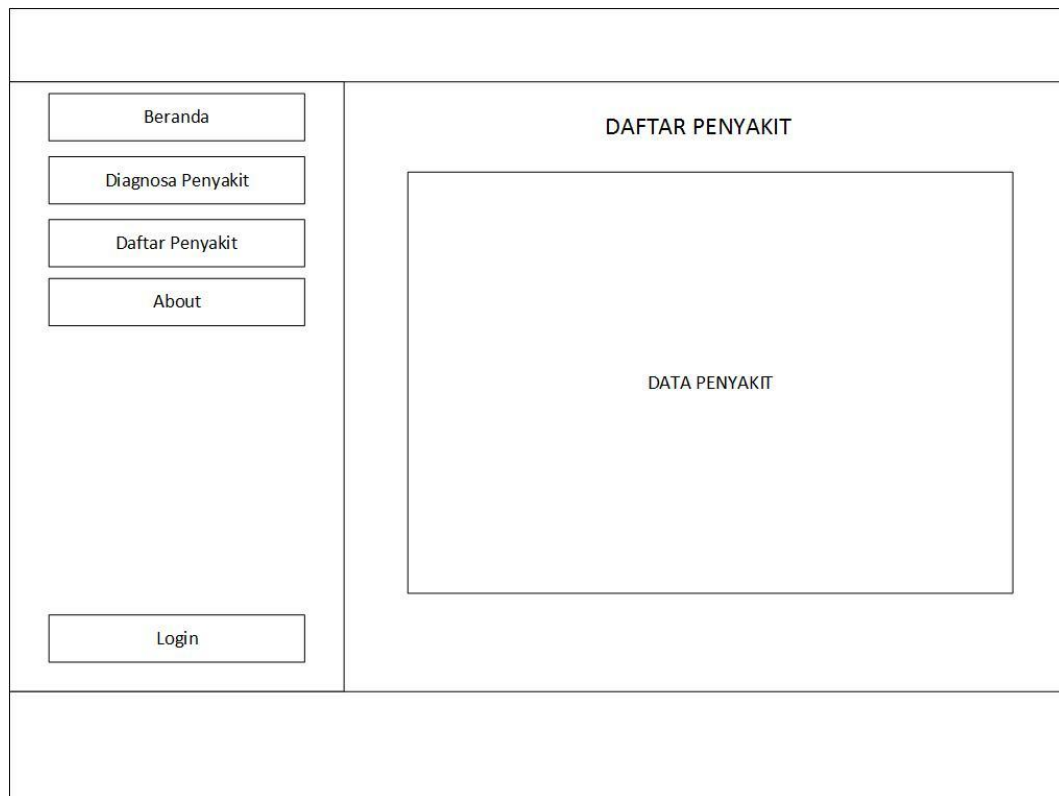
Pada perancangan ini merupakan form memasukan data solusi penyakit kejiwaan.

INPUT SOLUSI	
Beranda	KODE SOLUSI : <input type="text"/>
Penyakit	KODE PENCEGAHAN : <input type="text"/>
Gejala	KATEGORI : <input type="text"/>
Basis Pengetahuan	SOLUSI : <input type="text"/>
Pencegahan	
Solusi	<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>
Log Out	

GAMBAR : 3.29. Rancangan Hasil dari Memasukan Data Solusi Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

9. Perancangan antarmuka daftar penyakit

Pada perancangan ini merupakan daftar penyakit kejiwaan.



GAMBAR : 3.30. Rancangan Informasi Penyakit Kejiwaan Pada Sistem Pakar Diagnosa Penyakit Kejiwaan

BAB IV

IMPLEMENTASI DAN UJI COBA

4.1. Construction (Code & Test)

Dalam tahap ini penelitian berfokus pada pengkodean menggunakan bahasa *PHP*, setelah itu dilakukan pengujian hasil menggunakan metode *blackbox testing*. *Blackbox testing* ini menguji spesifikasi suatu fungsi atau modul, apakah berjalan sesuai dengan semestinya dan sesuai yang diharapkan atau tidak.

4.1.1. Implementasi Hardware & Software

Dalam tahap ini peneliti memerlukan spesifikasi *hardware* dan *software* untuk mengoperasikan sistem pakar. Untuk lebih jelasnya akan dijelaskan dibawah ini :

4.1.1.1. Hardware

Hardware yang digunakan sebagai berikut :

1. *Processor* AMD E1-1200 APU 1.40GHz
2. *Memory* 2,00 GB
3. *Hard disk* Minimal 1 GB

4.1.1.2. Software

1. Sistem Operasi *Windows 7*
2. *Browser* (*Google Chrome, Mozilla Firefox*)
3. *PHP*
4. *MySql*

5. *Bootstrap*

6. *Apache*

4.1.2. Implementasi Antarmuka

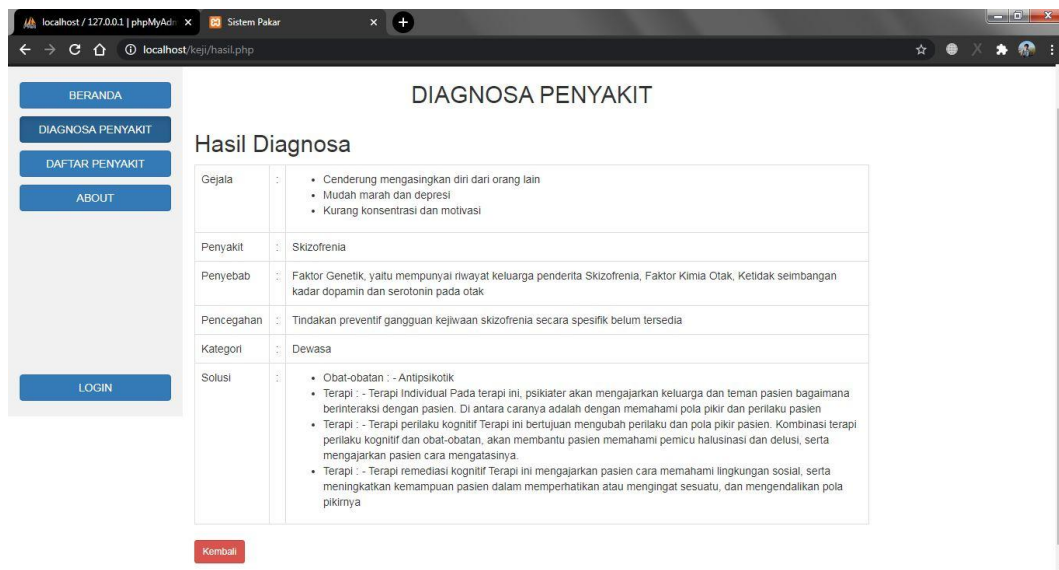
Pada gambar berikut ini menjelaskan tampilan menu konsultasi, jadi pengguna memilih gejala penyakit kejiwaan dengan melakukan *checkboxlist* pada *checkbox*.



The screenshot shows a web application interface for mental health diagnosis. The page is titled "DIAGNOSA PENYAKIT". On the left, there is a sidebar menu with buttons for "BERANDA", "DIAGNOSA PENYAKIT", "DAFTAR PENYAKIT", "ABOUT", and "LOGIN". The main content area contains a checklist with three items: "Apakah Pasien Cenderung Mengasingkan Diri Dari Orang Lain?", "Apakah Pasien Mudah Marah dan Depresi?", and "Apakah Pasien Kurang Konsentrasi dan Motivasi?". Below the checklist are two buttons: "Cek Diagnosa" and "Tidak ada".

GAMBAR : 4.1. Hasil Tampilan Memilih Gejala Penyakit Kejiwaan

Pada gambar berikut ini menjelaskan tampilan hasil metode *Forward Chaining* pada saat sudah memilih gejala penyakit kejiwaan.



DIAGNOSA PENYAKIT

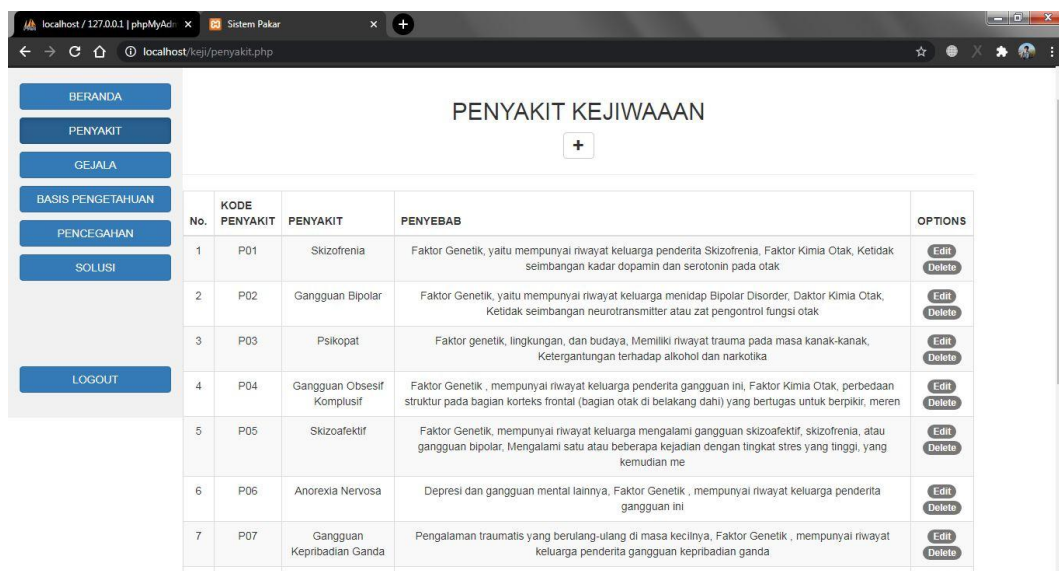
Hasil Diagnosa

Gejala	<ul style="list-style-type: none"> Cenderung mengasingkan diri dari orang lain Mudah marah dan depresi Kurang konsentrasi dan motivasi
Penyakit	Skizofrenia
Penyebab	Faktor Genetik, yaitu mempunyai riwayat keluarga penderita Skizofrenia, Faktor Kimia Otak, Ketidak seimbangan kadar dopamin dan serotonin pada otak
Pencegahan	Tindakan preventif gangguan kejiwaan skizofrenia secara spesifik belum tersedia
Kategori	Dewasa
Solusi	<ul style="list-style-type: none"> Obat-obatan :- Antipsikotik Terapi :- Terapi Individual Pada terapi ini, psikiater akan mengajarkan keluarga dan teman pasien bagaimana berinteraksi dengan pasien. Di antara caranya adalah dengan memahami pola pikir dan perilaku pasien Terapi :- Terapi perilaku kognitif Terapi ini bertujuan mengubah perilaku dan pola pikir pasien. Kombinasi terapi perilaku kognitif dan obat-obatan, akan membantu pasien memahami pemicu halusinasi dan delusi, serta mengajarkan pasien cara mengatasinya. Terapi :- Terapi remediasi kognitif Terapi ini mengajarkan pasien cara memahami lingkungan sosial, serta meningkatkan kemampuan pasien dalam memperhatikan atau mengingat sesuatu, dan mengendalikan pola pikirnya

[Kembali](#)

GAMBAR : 4.2. Hasil Tampilan Dari Hasil Proses Diagnosa Metode *Forward Chaining*

Pada gambar berikut ini menjelaskan tampilan data dari penyakit kejiwaan telah dimasukan sebelumnya berdasarkan pengetahuan pakar..



PENYAKIT KEJIWAAAN

No.	KODE PENYAKIT	PENYAKIT	PENYEBAB	OPTIONS
1	P01	Skizofrenia	Faktor Genetik, yaitu mempunyai riwayat keluarga penderita Skizofrenia, Faktor Kimia Otak, Ketidak seimbangan kadar dopamin dan serotonin pada otak	Edit Delete
2	P02	Gangguan Bipolar	Faktor Genetik, yaitu mempunyai riwayat keluarga menidap Bipolar Disorder, Doktor Kimia Otak, Ketidak seimbangan neurotransmitter atau zat pengontrol fungsi otak	Edit Delete
3	P03	Psikopat	Faktor genetik, lingkungan, dan budaya. Memiliki riwayat trauma pada masa kanak-kanak, Ketergantungan terhadap alkohol dan narkotika	Edit Delete
4	P04	Gangguan Obsesif Kompulsif	Faktor Genetik, mempunyai riwayat keluarga penderita gangguan ini, Faktor Kimia Otak, perbedaan struktur pada bagian korteks frontal (bagian otak di belakang dahi) yang bertugas untuk berpikir, meren	Edit Delete
5	P05	Skizoafektif	Faktor Genetik, mempunyai riwayat keluarga mengalami gangguan skizoafektif, skizofrenia, atau gangguan bipolar, Mengalami satu atau beberapa kejadian dengan tingkat stres yang tinggi, yang kemudian me	Edit Delete
6	P06	Anorexia Nervosa	Depresi dan gangguan mental lainnya, Faktor Genetik, mempunyai riwayat keluarga penderita gangguan ini	Edit Delete
7	P07	Gangguan Kepribadian Ganda	Pengalaman traumatis yang berulang-ulang di masa kecilnya, Faktor Genetik, mempunyai riwayat keluarga penderita gangguan kepribadian ganda	Edit Delete

GAMBAR : 4.3. Hasil Tampilan Dari Data Penyakit Kejiwaan

Pada gambar berikut ini menjelaskan tampilan data dari gejala penyakit telah dimasukan sebelumnya berdasarkan pengetahuan pakar.

No.	KODE GEJALA	GEJALA	OPTIONS
1	G01	Cenderung mengasingkan diri dari orang lain	Edit Delete
2	G02	Mudah marah dan depresi	Edit Delete
3	G03	Kurang konsentrasi dan motivasi	Edit Delete
4	G04	Suasana hati cepat berubah	Edit Delete
5	G05	Kehilangan minat dalam aktivitas sehari-hari	Edit Delete
6	G06	Seing berkhayal	Edit Delete
7	G07	Mudah berbobong	Edit Delete
8	G08	Melakukan tindakan berbahaya tanpa memikirkan risikonya dengan matang	Edit Delete
9	G09	Tidak memedulikan perasaan orang lain	Edit Delete
10	G10	Pikiran ragu-ragu yang berlebihan dan keperluan untuk memastikan berulang-ulang	Edit Delete
11	G11	Pikiran agresif tentang diri sendiri atau orang lain	Edit Delete
12	G12	Mengulangi kata-kata atau kalimat	Edit Delete

GAMBAR : 4.4. Hasil Tampilan Dari Data Gejala Kejiwaan

Pada gambar berikut ini menjelaskan tampilan data dari basis pengetahuan telah dimasukan sebelumnya berdasarkan pengetahuan pakar.

No.	JIKA	MAKA	OPTIONS
1	G01ANDG02ANDG03	P01	Edit Delete
2	G01ANDG02	P01	Edit Delete
3	G01ANDG03	P01	Edit Delete
4	G02ANDG03	P01	Edit Delete
5	G04ANDG05ANDG06	P02	Edit Delete
6	G04ANDG05	P02	Edit Delete
7	G04ANDG06	P02	Edit Delete
8	G05ANDG06	P02	Edit Delete
9	G07ANDG08ANDG09	P03	Edit Delete
10	G07ANDG08	P03	Edit Delete
11	G07ANDG09	P03	Edit Delete

GAMBAR : 4.5. Hasil Tampilan Dari Data Basis Pengetahuan

Pada gambar berikut ini menjelaskan tampilan data dari pencegahan penyakit kejiwaan telah dimasukan sebelumnya berdasarkan pengetahuan pakar.

No.	KODE PENCEGAHAN	KODE PENYAKIT	PENCEGAHAN	OPTIONS
1	PP01	P01	Tindakan preventif gangguan kejiwaan skizofrenia secara spesifik belum tersedia	Edit Delete
2	PP02	P02	Langkah pencegahan yang bisa dilakukan adalah dengan mengurangi efek dari bipolar	Edit Delete
3	PP03	P03	keluarga idealnya bisa menciptakan suasana yang penuh kasih sayang, jauh dari kekerasan, dan menumbuhkan kondisi yang baik untuk kesehatan mental saat membesarkan anak	Edit Delete
4	PP04	P04	Tidak ada cara yang pasti untuk mencegah gangguan obsesif-kompulsif	Edit Delete
5	PP05	P05	Belum ada metode yang terbukti efektif secara sepenuhnya mencegah timbulnya gangguan ini	Edit Delete
6	PP06	P06	Tidak ada pencegahan spesifik yang bisa dilakukan terhadap anoreksia	Edit Delete
7	PP07	P07	Pencegahan kepribadian ganda adalah dengan menghindari faktor pencetusnya, yaitu stres psikis	Edit Delete
8	PP08	P08	Belum ada cara pasti pencegahan menyakiti diri sendiri	Edit

GAMBAR : 4.6. Hasil Tampilan Dari Data Pencegahan Penyakit Kejiwaan

Pada gambar berikut ini menjelaskan tampilan data dari solusi penyakit kejiwaan telah dimasukan sebelumnya berdasarkan pengetahuan pakar.

No.	KODE SOLUSI	KODE PENCEGAHAN	KATEGORI	SOLUSI	OPTIONS
1	S01	PP01	Dewasa	Obat-obatan : - Antipsikotik	Edit Delete
2	S02	PP01	Dewasa	Terapi : - Terapi Individual Pada terapi ini, psikiater akan mengajarkan keluarga dan teman pasien bagaimana berinteraksi dengan pasien. Di antara caranya adalah dengan memahami pola pikir dan perilaku pasien	Edit Delete
3	S03	PP01	Dewasa	Terapi : - Terapi perilaku kognitif Terapi ini bertujuan mengubah perilaku dan pola pikir pasien. Kombinasi terapi perilaku kognitif dan obat-obatan, akan membantu pasien memahami pemicu halusinasi dan delusi, serta mengajarkan pasien cara mengatasinya.	Edit Delete
4	S04	PP01	Dewasa	Terapi : - Terapi remediasi kognitif Terapi ini mengajarkan pasien cara memahami lingkungan sosial, serta meningkatkan kemampuan pasien dalam memperhatikan atau mengingat sesuatu, dan mengendalikan pola pikirnya	Edit Delete
5	S05	PP02	Dewasa	Obat-obatan : - Moodstabilize, Antikonvulsan, Antipsikotik, dan Antidepresan	Edit Delete
6	S06	PP02	Dewasa	Terapi : - Interpersonal and social rhythm therapy (IPSRT) IPSRT merupakan metode yang terfokus pada kestabilan ritme aktivitas sehari-hari, seperti waktu untuk tidur, bangun, hingga makan. Terapurnya ritme dalam beraktivitas mampu membantu pasien untuk mengendalikan gejala gangguan bipolar	Edit Delete

GAMBAR : 4.7. Hasil Tampilan Dari Data Solusi Penyakit Kejiwaan

4.1.3. Testing

Testing atau pengujian merupakan bagian penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian perangkat lunak adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas dan dapat diandalkan.

4.1.3.1. Testing Blackbox

Pengujian *black box* digunakan untuk mengetahui fungsi-fungsi khusus dari aplikasi perangkat lunak yang telah dibuat

TABEL : 4.1. Tabel Rencana Pengujian


No.	Kelas Uji	Butir Uji	Jenis Pengujian
1.	Login Pakar	Verifikasi username	<i>Blackbox</i>
		Verifikasi password	
2.	Pengolahan Data Penyakit	Tambah data penyakit	<i>Blackbox</i>
		Ubah data penyakit	
		Hapus data penyakit	
3.	Pengolahan Data Gejala	Tambah data gejala	<i>Blackbox</i>
		Ubah data gejala	
		Hapus data gejala	
4.	Pengolahan Data Rule	Tambah data rule	<i>Blackbox</i>
		Ubah data rule	
		Hapus data rule	

Lanjutan TABEL : 4.1. Tabel Rencana Pengujian



No.	Kelas Uji	Butir Uji	Jenis Pengujian
5.	Pengolahan Data Pencegahan	Tambah data pencegahan	<i>Blackbox</i>
		Ubah data pencegahan	
		Hapus data pencegahan	
6.	Pengolahan Data Solusi	Tambah data solusi	<i>Blackbox</i>
		Ubah data solusi	
		Hapus data solusi	
7.	Diagnosa Gejala	Diagnosa	<i>Blackbox</i>

Setelah dilakukan pengujian dengan menggunakan metode *blackbox* maka didapatkan hasil pengujian pada tabel berikut ini :



TABEL : 4.2. Tabel Hasil Pengujian Login Pakar

No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
1.	Login Pakar	<i>Username</i> dan <i>Password</i>	Data <i>username</i> dan <i>password</i> dicari dari <i>database</i> Masuk ke menu pakar		Diterima


TABEL : 4.3. Tabel Hasil Pengujian Tambah Data Penyakit

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
2	Pengujian Tambah Data Penyakit	Kode penyakit, nama penyakit, dan penyebab	Data tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
2	Pengujian Tambah Data Penyakit	Kode penyakit, nama penyakit, penyebab tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima



TABEL : 4.4. Tabel Hasil Pengujian Ubah Data Penyakit

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
2	Pengujian Ubah Data Penyakit	Kode penyakit, nama penyakit, dan penyebab yang baru	Data yang diubah tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
2	Pengujian Ubah Data Penyakit	Kode penyakit, nama penyakit, penyebab, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima

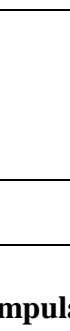
TABEL : 4.5. Tabel Hasil Pengujian Hapus Data Penyakit

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
2	Pengujian Hapus Data Penyakit	Penyakit yang akan dihapus	Menampilkan pesan 'yakin ingin menghapus data ini?' apabila memilih 'cancel' maka data batal dihapus. Apabila memilih 'ok' maka data akan dihapus		Diterima


TABEL : 4.6. Tabel Hasil Pengujian Tambah Data Gejala

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
3	Pengujian Tambah Data Gejala	Kode gejala dan gejala	Data tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
3	Pengujian Tambah Data Gejala	Kode gejala dan gejala, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		


TABEL : 4.7. Tabel Hasil Pengujian Ubah Data Gejala

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
3	Pengujian Ubah Data Gejala	Kode gejala dan gejala yang baru	Data yang diubah tersimpan kedalam database		Diterima



Lanjutan TABEL : 4.7. Tabel Hasil Pengujian Ubah Data Gejala

Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
3	Pengujian Ubah Data Gejala	Kode gejala dan gejala tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima

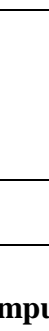
TABEL : 4.8. Tabel Hasil Pengujian Hapus Data Gejala

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
3	Pengujian Hapus Data Gejala	Gejala yang akan dihapus	Menampilkan pesan 'yakin ingin menghapus data ini?' apabila memilih 'cancel' maka data batal dihapus. Apabila memilih 'ok' maka data akan dihapus		Diterima


TABEL : 4.9. Tabel Hasil Pengujian Tambah Data Rule

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
4	Pengujian Tambah Data Rule	Jika dan maka	Data tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
4	Pengujian Tambah Data Rule	Jika dan maka, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima


TABEL : 4.10. Tabel Hasil Pengujian Tambah Data Rule

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
4	Pengujian Ubah Data Rule	Jika dan maka yang baru	Data yang diubah tersimpan kedalam database		Diterima



Lanjutan TABEL : 4.10. Tabel Hasil Pengujian Tambah Data Rule

Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
4	Pengujian Ubah Data Rule	Jika dan maka tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima



TABEL : 4.11. Tabel Hasil Pengujian Tambah Data Rule

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
4	Pengujian Hapus Data Rule	Rule yang akan dihapus	Menampilkan pesan 'yakin ingin menghapus data ini?' apabila memilih 'cancel' maka data batal dihapus. Apabila memilih 'ok' maka data akan dihapus		Diterima


TABEL : 4.12. Tabel Hasil Pengujian Tambah Data Pencegahan

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
5	Pengujian Tambah Data Pencegahan	Kode pencegahan, kode penyakit, dan pencegahan	Data tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
5	Pengujian Tambah Data Pencegahan	Kode pencegahan, kode penyakit, dan pencegahan, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima



TABEL : 4.13. Tabel Hasil Pengujian Ubah Data Pencegahan

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
5	Pengujian Ubah Data Pencegahan	Kode pencegahan, kode penyakit, dan pencegahan yang baru	Data yang diubah tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
5	Pengujian Ubah Data Pencegahan	Kode pencegahan, kode penyakit, dan pencegahan yang baru, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima



TABEL : 4.14. Tabel Hasil Pengujian Hapus Data Pencegahan

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
5	Pengujian Hapus Data Pencegahan	Pencegahan yang akan dihapus	Menampilkan pesan ‘yakin ingin menghapus data ini?’ apabila memilih ‘cancel’ maka data batal dihapus. Apabila memilih ‘ok’ maka data akan dihapus		Diterima


TABEL : 4.15. Tabel Hasil Pengujian Tambah Data Solusi

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
6	Pengujian Tambah Data Solusi	Kode sousi, kode pencegahan, kategori, dan solusi	Data tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
6	Pengujian Tambah Data Solusi	Kode sousi, kode pencegahan, kategori, dan solusi, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima



TABEL : 4.16. Tabel Hasil Pengujian Ubah Data Solusi

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
6	Pengujian Ubah Data Solusi	Kode sousi, kode pencegahan, kategori, dan solusi yang baru, dan pencegahan yang baru	Data yang diubah tersimpan kedalam database		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
6	Pengujian Ubah Data Solusi	Kode sousi, kode pencegahan, kategori, dan solusi yang baru, tetapi salah satu dari form dikosongkan	Tidak terjadi proses simpan, menampilkan pesan 'Isi kolom dengan benar'		Diterima

TABEL : 4.17. Tabel Hasil Pengujian Hapus Data Solusi

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
6	Pengujian Hapus Data Solusi	Solusi yang akan dihapus	Menampilkan pesan ‘yakin ingin menghapus data ini?’ apabila memilih ‘cancel’ maka data batal dihapus. Apabila memilih ‘ok’ maka data akan dihapus		Diterima

TABEL : 4.18. Tabel Hasil Pengujian Diagnosa Gejala

Kasus dan Hasil Uji (Data Normal)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
7.	Diagnosa Gejala	Pilih gejala yang akan didiagnosa	Data gejala akan diperiksa berdasarkan tabel rule dan akan menampilkan nama penyakit, gejala, kategori, penyebab, pencegahan, dan solusi		Diterima
Kasus dan Hasil Uji (Data Salah)					
No.	Kelas Uji	Data Inputan	Hasil yang Diharapkan	Hasil Yang Ditampilkan Sistem	Kesimpulan
7.	Diagnosa Gejala	Gejala yang dipilih tidak sesuai dengan rule	Tidak terjadi proses diagnosa, menampilkan pesan 'Silahkan pilih gejala'		Diterima

4.1.3.2. *Testing* Klasifikasi Penyakit Kejiwaan

Pengujian klasifikasi penyakit kejiwaan ini digunakan untuk menguji seberapa besar kemampuan sistem dalam mengklasifikasi penyakit kejiwaan, dimana pada penelitian ini penyakit kejiwaan yang telah diklasifikasikan akan muncul hasil diagnosa dan besarnya akurasinya terhadap kesamaan penyakit kejiwaan yang diuji dengan penyakit kejiwaan yang ada di *database*.

Pada penelitian ini penulis melakukan *testing* klasifikasi penyakit kejiwaan dengan menggunakan kelas uji. Kelas uji berisi penyakit kejiwaan yang akan diklasifikasikan.

TABEL : 4.3. Rencana *Testing* Klasifikasi Penyakit Kejiwaan

No.	Kelas Uji
1.	Skizofrenia
2.	Gangguan Bipolar
3.	Psikopat
4.	Gangguan Obsesif Komplusif
5.	Skizoafektif
6.	Anorexia Nervosa
7.	Gangguan Kepribadian Ganda
8.	Menyakiti Diri Sendiri
9.	Gangguan Kecemasan
10.	Gangguan Pemusatan Perhatian/Hiperaktif
11.	Gangguan Stres Pascatrauma
12.	Gangguan Perkembangan pada Anak

4.1.4. Pengujian Keakuratan

Pengujian terhadap 14 data user/pasien dilakukan, maka hasil ini dibandingkan dengan hasil dari pakar. Tujuannya adalah mengukur tingkat akurasi hasil yang di dapatkan. Keakuratan sistem terdiri dari level 0 dan level 1. Level 0 diberikan jika hasil diagnosa sistem tidak sama dengan hasil diagnosa pakar. Level 1 diberikan jika hasil diagnosa sistem dan pakar memberikan hasil yang sama. Perhitungan nilai probabilitas (P) pada pengujian data menggunakan *google form* adalah sebagai berikut:

$$\text{Rumus Probabilitas : } P_{(E)} = \frac{X}{N} \times 100\%$$

Keterangan :

P : Probabilitas;

E : Event;

X : Jumlah kejadian yang terjadi;

N : Jumlah seluruh kejadian

Hasil pengujian sistem pakar ini terhadap 14 data user/pasien adalah sebagai berikut:

$$\begin{aligned} P_{\text{jumlah (Akurat)}} &= \frac{\text{Total Level 1}}{\text{Total Pengujian}} \times 100\% \\ &= \frac{11}{14} \times 100\% \\ &= 79\% \end{aligned}$$

$$\begin{aligned} P_{\text{jumlah (Tidak Akurat)}} &= \frac{\text{Total Level 1}}{\text{Total Pengujian}} \times 100\% \\ &= \frac{3}{14} \times 100\% \\ &= 21\% \end{aligned}$$

Setelah dilakukan pengujian dan perhitungan tingkat akurasi sistem, maka di dapatkan tingkat akurasi sebesar 78 % dan tidak akurat sebesar 21 % dari 14 data pengujian. Berdasarkan tingkat akurasi tersebut, maka sistem pakar dengan metode forward chaining cukup akurat dalam mengidentifikasi penyakit kejiwaan.

4.1.5. Kesimpulan Testing

Berdasarkan hasil testing yang telah dilakukan dapat ditarik kesimpulan bahwa aplikasi sudah berjalan sesuai yang diharapkan, tetapi tidak menutup kemungkinan dapat terjadi kesalahan suatu saat pada saat aplikasi digunakan, sehingga membutuhkan proses maintenance untuk lebih mengetahui kekurangan dari aplikasi.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pembahasan beserta penelitian yang telah dilakukan, maka dapat diambil beberapa kesimpulan, diantaranya sebagai berikut :

1. Dalam sistem pakar yang dibangun untuk menentukan aturan premis dalam gangguan kejiwaan berdasarkan gejala dari penyakit kejiwaan itu sendiri sehingga menghasilkan rule dengan metode *forward chaining*.
2. Penerapan metode *forward chaining* dalam sistem pakar yang dibangun dengan menganalisis data berdasarkan penelitian sebelumnya sehingga bisa disimpulkan premis dan juga konklusi yang tepat untuk sistem pakar ini.
3. Pada sistem pakar yang dibangun penyakit kejiwaan memiliki beberapa gejala penyakit umum dan juga khusus, jadi satu penyakit mungkin saja memiliki gejala penyakit yang umum dan juga khusus. Pengelompokan gejala penyakit serta penyakit kejiwaan dari seorang pakar.

5.2. Saran

Dengan adanya kesimpulan diatas, ada beberapa saran yang dapat dikemukakan sebagai bahan pertimbangan lebih lanjut guna meningkatkan produktifitas kerja dari aplikasi sistem pakar ini.

1. Aplikasi ini dapat digunakan oleh masyarakat luas, dengan syarat harus memiliki koneksi internet untuk membuka *browser*.
2. Diagnosa penyakit kejiwaan ini dapat dijadikan bahan pertimbangan bagi pengguna/pasien untuk mengetahui gejala penyakit, pencegahan penyakit, dan juga solusi/ penanganan lebih lanjut akan penyakit kejiwaan.
3. Diharapkan penyakit yang diangkat pada penelitian selanjutnya lebih luas lagi, karena penyakit kejiwaan tidak hanya penyakit kejiwaan umum saja. Agar dapat membantu pemahaman masyarakat akan penyakit kejiwaan.

DAFTAR PUSTAKA

- Abdul Kadir (2003:54)*, Pengenalan Sistem Informasi, Andi. Yogyakarta. *Abdul Kadir (2003:10)*, Pengenalan Sistem Informasi, Andi.
- Agung, *Gregorius*. 2000. Microsoft Frontpage 2000 Webbot. Jakarta: PT. Elex. Media Komputindo. Alexander, F. K. Sibero. 2011.
- Andrian, Kevin. 2018. “Psikopat”, <https://www.alodokter.com/jangan-tertipu-watak-seorang-psikopat>, diakses pada 20 agustus 2020 pukul 20.25.
- Andrian, Kevin. 2019. “Self-Injury, Gangguan Psikologis Menyakiti Diri Sendiri”, <https://www.alodokter.com/self-injury-gangguan-psikologis-menyakiti-diri-sendiri#:~:text=Self%2Dinjury%20adalah%20perilaku%20menyakiti,terkait%20dengan%20sejumlah%20penyakit%20kejiwaan.>, diakses pada 20 agustus 2020 pukul 20.52.
- Andrian, Kevin. 2020. “Mengenal Axinety yang mengganggu dan berbagai jenisnya”, <https://www.alodokter.com/mengenal-anxiety-yang-mengganggu-dan-berbagai-jenisnya.>, diakses pada 20 agustus 2020 pukul 21.06.
- Arhami, Muhammad*. 2005. Konsep Dasar Sistem Pakar. Penerbit Andi. Yogyakarta. Ayuningtiyas, Ika Kurnianti. 2007.
- Bunafit, Nugroho*. (2008), Latihan Membuat Aplikasi Web PHP dan Mysql dengan Dreamwaver, Gava Media, Yogyakarta.
- Christy, Tika*. (2018), Implementasi Sistem Pakar Diagnosa Cabe Menggunakan Metode *Forward Chaining*. Seminar Nasional Royal.
- Connolly, Thomas and Begg, Carolyn. (2010). Database Systems A Practical Approach to Design, Implementation, and Management Fifth Edition. Boston: Pearson Education.
- Durkin, & Jhon*. (1994). Expert System Design And Development. Hall: Pretice.
- Fitriyanto, G., Azhar, N. A., & Kurniawan, M. (2016).
- Flanagan, David, 2011, JavaScript: The Definitive Guide, Sobastopol: O’Relly.
- Indrajani, 2001, *Analisis dan Perancangan Sistem Berorientasi Object*. Bandung, Informatika.

- Jogiyanto HM. 2005. *Analisis & Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta. Andi.
- M. Otto and J. Thornton, “Bootstrap”, 19 Agustus 2011. [Online]. Availabel : www.getbootstrap.com. [Accessed 5 52014].
- Merlina, Nita, M.Kom., & Rahmat Hidayat, S.Kom. 2012. Perancangan Sistem. Pakar. Ghalia Indonesia. Yogyakarta. Kusriani,. 2008. Aplikasi Sistem Pakar. Andi.
- Mevung, Fersalina Indah, Addy Suyatno, Septya Maharani. 2017. Diagnosa Penyakit Kejiwaan Menggunakan Metode *Certainty Factor*. Prosiding Seminar Komputer dan Teknologi Informatika. Vol 2, No. 1
- Nasir, Abdul dan, Abdul, mhith. 2011. *Dasar-dasar Keperawatan jiwa, Pengantar dan Teori*. Jakarta: Salemba Medika.
- Pane, Merry Dame Cristy. 2020. “Kepribadian Ganda”, <https://www.alodokter.com/kepribadian-ganda>, diakses pada 20 agustus 2020 pukul 20.45.
- Pane, Merry Dame Cristy. 2020. “PTSD (*post-traumatic stress disorder*) atau gangguan stres pascatrauma adalah gangguan mental yang muncul setelah seseorang mengalami atau menyaksikan peristiwa yang tidak menyenangkan”, [https://www.alodokter.com/ptsd#:~:text=PTSD%20\(post%2Dtraumatic%20stress%20disorder,penderitanya%20teringat%20pada%20kejadian%20traumatis.](https://www.alodokter.com/ptsd#:~:text=PTSD%20(post%2Dtraumatic%20stress%20disorder,penderitanya%20teringat%20pada%20kejadian%20traumatis.), diakses pada 20 agustus 2020 pukul 21.21.
- Pressman, R.S.2015. *Rekayasa Perangkat Lunak : Pendekatan Praktisi Buku I*. Yogyakarta : Andi
- Rahmawati, Esti, Hari Hibawanto. 2016. Sistem Pakar Diagnosa Penyakit Paru-Paru Menggunakan Metode *Forward Chaining*. Jurnal Teknik Elektro Vol. 8 No. 2.
- Robert G Murdick, dkk, *Sistem Informasi Untuk Manajemen Modern*, Jakarta : Erlangga, 1991.
- Sudarmana, Landung, Febty Lestari. 2018. Aplikasi Sistem Pakar Mendiagnosa Gangguan Jiwa *Schizofrenia*. Jurnal Pengembangan IT Vol. 3 No. 1
- Sugiri, Budi Kurniawan.2007. *Desain Web Menggunakan HTML + CSS*. Penerbit Andi Yogyakarta.

- Tectale. (2012). *An Introduction to Twitters Bootstrap*. Tectale Website
- Teguh Prasetyo, 2010. *Hukum Pidana*, PT. Raja Grafindo Persada. Jakarta
- Turban, E. "Decision Support System and Expert System", Penerbit Prentice. Hall Internasional, United State, 1995.
- Utari, Reni. 2019. "Gangguan Pertumbuhan", <https://www.sehatq.com/penyakit/gangguan-pertumbuhan#:~:text=Gangguan%20pertumbuhan%20adalah%20suatu%20kondisi,gangguan%20kelenjar%20atau%20penyakit%20tertentu.>, diakses pada 20 agustus 2020 pukul 21.29.
- Wardhani, Rima Diah, Rekyan Regasari Mardi Putri, Budi Darma Setiawan, 2017. *Sistem Pakar Diagnosa Penyakit Schizofrenia Menggunakan Metode Bayesian Network*, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Vol.1 No. 11.
- Willy, Tjin. 2018. "Skizofrenia", <https://www.alodokter.com/skizofrenia>, diakses pada 20 agustus 2020 pukul 20.13.
- Willy, Tjin. 2018. "Gangguan Bipolar", <https://www.alodokter.com/gangguan-bipolar>, diakses pada 20 agustus 2020 pukul 20.17.
- Willy, Tjin. 2018. "Anoreksia Nervosa", <https://www.alodokter.com/anoreksia-nervosa>, diakses pada 20 agustus 2020 pukul 20.38.
- Willy, Tjin. 2019. "Pengertian ADHD", <https://www.alodokter.com/adhd#:~:text=Pengertian%20ADHD,pada%20prestasi%20anak%20di%20sekolah.>, diakses pada 20 agustus 2020 pukul 21.13.
- Yosep, Iyus, Sutini, Titin. (2014). *Buku Ajar Keperawatan Jiwa (dan Advance mental Health nursing)*. Bandung Refika Aditama.

LAMPIRAN

Diagnosa.php

```
<?php

include('koneksi.php');

if(isset($_SESSION['login_user'])){

header("location: about.php");

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Sistem Pakar</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link                                rel="stylesheet"

href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
```

```
<link rel="stylesheet" href="css/style.css">
```

```
<script
```

```
src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
```

```
<script
```

```
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></scri
```

```
pt>
```

```
</head>
```

```
<body>
```

```
<nav class="navbar navbar-inverse">
```

```
<div class="container-fluid">
```

```
<div class="navbar-header">
```

```
<button type="button" class="navbar-toggle" data-toggle="collapse" data-  
target="#myNavbar">
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
</button>
```

```
</div>
```

```
<div class="collapse navbar-collapse" id="myNavbar">

  <ul class="nav navbar-nav">

  </ul>

  <ul class="nav navbar-nav navbar-right">

  </ul>

</div>

</div>

</nav>

<div class="container-fluid text-center">

  <div class="row content">

    <div class="col-sm-2 sidenav">

      <p><a href="index.php"><button type="button" class="btn btn-primary btn-
block">BERANDA</button></a></p>

      <p><a href="diagnosa.php"><button type="button" class="btn btn-primary
btn-block active">DIAGNOSA PENYAKIT</button></a></p>

      <p><a href="daftarpenyakit.php"><button type="button" class="btn btn-
primary btn-block">DAFTAR PENYAKIT</button></a></p>
```

```
<p><a href="about.php"><button type="button" class="btn btn-primary btn-block">ABOUT</button></a></p>
```

```
<br><br><br><br><br><br><br><br><br><br>
```

```
<p><button type="button" class="btn btn-primary btn-block" id="myBtn">LOGIN</button></p>
```

```
</div>
```

```
<div class="col-sm-8 text-left">
```

```
<center><h2>DIAGNOSA PENYAKIT</h2></center>
```

```
<div class="row">
```

```
<div class="col-lg-14">
```

```
<?php if (isset($_GET['error'])) {echo "<div class='alert alert-warning alert-dismissible fade show' role='alert'>
```

```
<button type='button' class='close' data-dismiss='alert' aria-label='Close'>
```

```
<span aria-hidden='true'>&times;</span>
```

```
</button>
```

```
<strong>Ups! </strong> $_GET[error]
```

```
</div>";} else { echo "";} ?>
```

```
<form method="POST" action="hasil.php" name="diagnosa" enctype="form-data/multipart">
```

```
<div class="panel panel-success">

<div class="panel-heading"></div>

<?php

$stampil="select * from gejala where nama_penyakit='Skizofrenia';

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

<?php }?>

<div class="panel-body">

</div>

</div>

<div class="panel panel-success">

<div class="panel-heading"></div>

<?php
```

```

$stampil="select * from gejala where nama_penyakit='Gangguan Bipolar'";

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

    <?php }?>

    <div class="panel-body">

    </div>

    </div>

    <div class="panel panel-success">

    <div class="panel-heading"></div>

    <?php

$stampil="select * from gejala where nama_penyakit='Psikopat'";

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

```



```
$a = $data['gejala'];

?>

<input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

<?php }?>

<div class="panel-body">

</div>

</div>

<div class="panel panel-success">

<div class="panel-heading"></div>

<?php

$stampil="select * from gejala where nama_penyakit='Gangguan Obsesif
Komplusif'";

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

$a = $data['gejala'];

?>
```

```

        <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

        <?php }?>

        <div class="panel-body">

        </div>

        </div>

        <div class="panel panel-success">

        <div class="panel-heading"></div>

        <?php

Stampil="select * from gejala where nama_penyakit='Skizoafektif';

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

        <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

        <?php }?>

        <div class="panel-body">

```

```
</div>
```

```
</div>
```

```
<div class="panel panel-success">
```

```
<div class="panel-heading"></div>
```

```
<?php
```

```
$stampil="select * from gejala where nama_penyakit='Anorexia Nervosa'";
```

```
$query = mysqli_query($koneksi,$stampil);
```

```
while ($data=mysqli_fetch_array($query)) {
```

```
    $a = $data['gejala'];
```

```
    ?>
```

```
    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
```

```
name="cek[]" /> <?php echo $data['gejala']; ?><br />
```

```
<?php }?>
```

```
<div class="panel-body">
```

```
</div>
```

```
</div>
```

```

<div class="panel panel-success">

<div class="panel-heading"></div>

<?php

$stampil="select * from gejala where nama_penyakit='Gangguan Kepribadian
Ganda";

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

<?php }?>

<div class="panel-body">

</div>

</div>

<div class="panel panel-success">

<div class="panel-heading"></div>

<?php

```

```

$stampil="select * from gejala where nama_penyakit='Menyakiti Diri Sendiri';

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

    <?php }?>

    <div class="panel-body">

    </div>

    </div>

    <div class="panel panel-success">

    <div class="panel-heading"></div>

    <?php

$stampil="select * from gejala where nama_penyakit='Gangguan Kecemasan';

$query = mysqli_query($koneksi,$stampil);

while ($data=mysqli_fetch_array($query)) {

```

```

    $a = $data['gejala'];

    ?>

    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

    <?php }?>

    <div class="panel-body">

    </div>

    </div>

    <div class="panel panel-success">

    <div class="panel-heading"></div>

    <?php

    $stampil="select * from gejala where nama_penyakit='Gangguan Pemusatan
Perhatian/Hiperaktif'";

    $query = mysqli_query($koneksi,$stampil);

    while ($data=mysqli_fetch_array($query)) {

    $a = $data['gejala'];

    ?>

```

```

        <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

        <?php }?>

        <div class="panel-body">

        </div>

        </div>

        <div class="panel panel-success">

        <div class="panel-heading"></div>

        <?php

        $stampil="select * from gejala where nama_penyakit='Gangguan Stres
Pascatrauma'";

        $query = mysqli_query($koneksi,$stampil);

        while ($data=mysqli_fetch_array($query)) {

                $a = $data['gejala'];

                ?>

                <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"
name="cek[]" /> <?php echo $data['gejala']; ?><br />

                <?php }?>

```

```
<div class="panel-body">
```

```
</div>
```

```
</div>
```

```
<div class="panel panel-success">
```

```
<div class="panel-heading"></div>
```

```
<?php
```

```
$stampil="select * from gejala where nama_penyakit='Gangguan  
Perkembangan Pada Anak';
```

```
$query = mysqli_query($koneksi,$stampil);
```

```
while ($data=mysqli_fetch_array($query)) {
```

```
    $a = $data['gejala'];
```

```
    ?>
```

```
    <input type="checkbox" value="<?php echo $data['kd_gejala'];?>"  
name="cek[]" /> <?php echo $data['gejala']; ?><br />
```

```
<?php }?>
```

```
<div class="panel-body">
```

```
</div>
```

```
</div>
```



```
        <input type="submit" class="btn btn-medium btn-primary" value="Cek
Diagnosa" name="proses" />
```

```
    </form>
```

```
</div>
```

```
</div>
```

```
<br /><br />
```

```
</main>
```

```
</body>
```

```
</html>
```

Hasil.php

```
<?php
```

```
include('koneksi.php');
```

```
if(isset($_SESSION['login_user'])){
```

```
header("location: about.php");

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Sistem Pakar</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet"

href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<link rel="stylesheet" href="css/style.css">

<script

src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<script

src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></scri

pt>

</head>

<body>
```

```
<nav class="navbar navbar-inverse">

  <div class="container-fluid">

    <div class="navbar-header">

      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target="#myNavbar">

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

        <span class="icon-bar"></span>

      </button>

    </div>

    <div class="collapse navbar-collapse" id="myNavbar">

      <ul class="nav navbar-nav">

      </ul>

      <ul class="nav navbar-nav navbar-right">

      </ul>

    </div>

  </div>
```

```
</nav>
```

```
<div class="container-fluid text-center">
```

```
<div class="row content">
```

```
<div class="col-sm-2 sidenav">
```

```
<p><a href="index.php"><button type="button" class="btn btn-primary btn-  
block">BERANDA</button></a></p>
```

```
<p><a href="diagnosa1.php"><button type="button" class="btn btn-primary  
btn-block active">DIAGNOSA PENYAKIT</button></a></p>
```

```
<p><a href="daftarpenyakit.php"><button type="button" class="btn btn-  
primary btn-block">DAFTAR PENYAKIT</button></a></p>
```

```
<p><a href="about.php"><button type="button" class="btn btn-primary btn-  
block">ABOUT</button></a></p>
```

```
<br><br><br><br><br><br><br><br><br><br>
```

```
<p><button type="button" class="btn btn-primary btn-block"  
id="myBtn">LOGIN</button></p>
```

```
</div>
```

```
<div class="col-sm-8 text-left">
```

```
<center><h2>HASIL DIAGNOSA PENYAKIT</h2></center>
```

```
<main role="main">
```

```
<!-- Example row of columns -->
```

```
<div class="row">
```

```
<div class="col-lg-12">
```

```
<div class="panel panel-success">
```

```
<div class="panel-heading"></div>
```

```
<?php
```

```
if(isset($_POST['proses']))
```

```
{
```

```
    $a = $_POST['cek'];
```

```
    $aa = implode('AND',$a);
```

```
    //echo $aa;
```

```
    $sql = mysqli_query($koneksi, "SELECT pencegahan.*, penyakit.*, solusi.*,  
rule.*
```

```
        FROM pencegahan, penyakit, solusi, rule
```

```
        WHERE rule.maka=penyakit.kode AND
```

```
        pencegahan.kode=penyakit.kode AND
```

```

        solusi.kd_pencegahan=pencegahan.kd_pencegahan AND

        rule.jika='$aa');

    if(mysqli_num_rows($sql) == 0){

        header('location:diagnosa.php?error=Tidak ditemukan penyakit kejiwaan
dengan gejala tersebut, silahkan cek <a href="daftar.php">daftar penyakit
kejiwaan</a>');

        }else{

            $row = mysqli_fetch_array($sql);

        }

    }

?>

<table id="example1" class="table table-bordered table-striped">

<tr>

<td>Gejala</td>

<td>:</td>

<td>

<?php

if(isset($a[0])){

```

```

        $sql1 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[0]'");

        $row1 = mysqli_fetch_array($sql1);

        echo "<ul><li>$row1[gejala]</li>";

        } else {echo "";}

    ?>

<?php

    if(isset($a[1])){

        $sql2 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[1]'");

        $row2 = mysqli_fetch_array($sql2);

        echo "<li>$row2[gejala]</li>";

        } else {echo "";}

    ?>

<?php

    if(isset($a[2])){

        $sql3 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[2]'");

        $row3 = mysqli_fetch_array($sql3);

```

```
        echo "<li>$row3[gejala]</li>";

    } else {echo "";}

?>

<?php

    if(isset($a[3])){

        $sql4 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[3]'");

        $row4 = mysqli_fetch_array($sql4);

        echo "<li>$row4[gejala]</li>";

    } else {echo "";}

?>

<?php

    if(isset($a[4])){

        $sql4 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[4]'");

        $row4 = mysqli_fetch_array($sql4);

        echo "<li>$row4[gejala]</li>";

    } else {echo "";}

?>
```



```

        <?php
            if(isset($a[5])){
                $sql4 = mysqli_query($koneksi, "SELECT * FROM gejala WHERE
kd_gejala='$a[5]'");
                $row4 = mysqli_fetch_array($sql4);
                echo "<li>$row4[gejala]</li></ul>";
            } else {echo "";}
        ?>
    </td>

</tr>

<tr>

<td>Penyakit</td>

<td>:</td>

<td colspan="2"><?php echo $row['nama_penyakit']; ?></td>

</tr>

<tr>

<td>Penyebab</td>

```

```
<td>:</td>
```

```
<td colspan="2"><?php echo $row['penyebab']; ?></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Pencegahan</td>
```

```
<td>:</td>
```

```
<td colspan="2"><?php echo $row['deskripsi']; ?></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Kategori</td>
```

```
<td>:</td>
```

```
<td colspan="2"><?php echo $row['kategori']; ?></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Solusi</td>
```

```
<td>:</td>
```

```
<td colspan="2">
```

```
<ul>
```

```
<?php

$nomor = $row['kd_pencegahan'];

$query = mysqli_query($koneksi,"SELECT * FROM solusi WHERE
kd_pencegahan='$nomor'");

$no=0;

while ($data=mysqli_fetch_array($query)) {

    ?>

    <li><?php echo $data['solusi']; ?></li>

    <?php }

    ?>

</ul>

</td>

</tr>

</table>

<div class="panel-body">
```

</div>

</div>

Kembali

</div>

</div>

</main>

</body>

</html>