



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Harmayani dkk (2021:1), bahwa “Komputer adalah alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan. Kata *computer* semula dipergunakan untuk menggambarkan orang yang perkerjaannya melakukan perhitungan aritmatika, dengan atau tanpa alat bantu, tetapi arti kata ini kemudian dipindahkan kepada mesin itu sendiri. Asal mulanya, pengolahan informasi hampir eksklusif berhubungan dengan masalah aritmatika, tetapi komputer modern dipakai untuk banyak tugas yang tidak berhubungan dengan matematika”.

Hal yang sama juga diungkapkan oleh Ahmad (2019:2) yang menyatakan, “Komputer didefinisikan sebagai perangkat elektronik yang dapat dipakai untuk mengolah data dengan perantara sebuah program yang mampu memberikan informasi dan hasil dari pengolahan data tersebut. Komputer dapat pula diartikan sebagai suatu mesin yang menerima *input* untuk proses dan menghasilkan *output*”.

Dengan demikian dapat disimpulkan bahwa komputer merupakan alat elektronik yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan.

2.1.2 Komponen Komputer

Menurut Krisbiantoro (2018:3) Komputer terdiri dari 3 komponen utama yang tidak dapat dipisahkan, yaitu :

- a. Perangkat Keras (*Hardware*), merupakan peralatan fisik dari komputer yang dapat kita lihat dan rasakan. Perangkat keras ini terdiri dari : *Input/Output Device (L/O Device)*, *Storage Device* (perangkat penyimpanan), *Monitor /Screen*, *Casing Unit*, *Central Prosesing Unit (CPU)*.



- b. Perangkat Lunak (*Software*), merupakan program yang berisi perintah-perintah untuk melakukan proses tertentu, Program tersebut ditulis dengan bahasa khusus yang dimengerti oleh komputer. Perangkat Lunak terdiri dari beberapa jenis, yaitu: Sistem Operasi, Program *Utiliry*. Program Aplikasi, Program Paket, Bahasa Pemrograman.
- c. *User (Brainware)*, adalah personil-personil yang terlibat langsung dalam pemakaian komputer, seperti sistem analis, programmer, oprator, user, dll.

2.1.3 Pengertian Perangkat Lunak (*Software*)

Menurut Marina dkk (2017:40), bahwa “Yang dimaksud *software* atau piranti lunak adalah data yang diformat dan disimpan secara digital termasuk program komputer, dokumentasinya dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer “.

Sama halnya menurut Rachmadi (2020:5) menyatakan, “*Software* adalah perangkat lunak, istilah ini digunakan untuk data yang telah diformat, dan disimpan ke dalam media penyimpanan dalam bentuk digital”.

Dari beberapa definisi di atas penulis menyimpulkan bahwa perangkat lunak adalah suatu program komputer yang tersusun atas perintah-perintah atau fungsi-fungsi untuk menjalankan suatu tugas tertentu.

2.1.4 Pengertian Sistem

Menurut Nafiudin (2019:6), “Sistem dapat dikatakan sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk suatu tujuan bersama. Sistem dapat merupakan sesuatu yang abstrak dan maupun yang berwujud.”.

Sama halnya dengan pendapat Munawir (2018:3) yang menyatakan, “Sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian yang saling mempengaruhi (L Ackof.1997)”.

Dari pengertian diatas, disimpulkan bahwa sistem adalah suatu kumpulan atau komponen dari prosedur-prosedur yang saling berhubungan satu sama lain untuk mencapai tujuan tertentu.



2.1.5 Pengertian Basis Data (*Database*)

Menurut Enterprise (2017:2) menyatakan bahwa yang dimaksud dengan “*Database* sendiri adalah kumpulan data yang terorganisasi secara rapi.”.

Sedangkan menurut Rachmadi (2020:12) yang mendefinisikan bahwa Basis Data adalah “Himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah. Kumpulan data dalam bentuk file/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis, untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian data sesuai tujuan”.

Dari beberapa pernyataan di atas, dapat disimpulkan bahwa basis data adalah kumpulan informasi yang tersimpan di komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer.

2.1.6 Komponen Basis Data

Menurut Rachmadi (2020:27), Komponen Basis Data (*Database*):

1. *User*
2. *Database Application (Create, Process, Administer)*
3. *DBMS*
4. *Database*

2.1.7 Pengertian Website

Menurut Sa’ad (2020:3) bahwa, “*Website* atau *web* adalah sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet”.

Sedangkan menurut Arief dalam Nofyat dkk (2018:12) menyatakan bahwa *website* adalah, “Kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan internet dan memiliki *domain/URL (Uniform Resource Locator)* yang dapat diakses semua pengguna internet dengan cara mengetikkan alamatnya.”.



Berdasarkan pengertian diatas dapat disimpulkan bahwa *web* adalah suatu perangkat lunak yang berisi kumpulan-kumpulan halaman yang menampilkan berbagai macam informasi yang digunakan oleh banyak orang.

2.2 Teori Judul

2.2.1 Pengertian Sistem Pakar

Menurut Andirani (2016:9), “Sistem Pakar adalah sebuah sistem yang kinerjanya mengadopsi keahlian yang dimiliki seorang pakar dalam bidang tertentu ke dalam sistem atau program komputer yang disajikan dengan tampilan yang dapat digunakan oleh pengguna yang bukan seorang pakar sehingga dengan sistem tersebut pengguna dapat membuat sebuah keputusan atau menentukan kebijakan layaknya seorang pakar.”

Menurut Hayadi (2018:1), “Sistem pakar atau *Expert System* biasa disebut juga dengan *Knowledge Based System* yaitu suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik.”

Berdasarkan pengertian diatas, dapat disimpulkan bahwa Sistem Pakar merupakan teknologi yang diciptakan untuk mentransfer pemikiran seorang pakar ke dalam komputer sehingga dapat menghasilkan sebuah kebijakan seperti seorang pakar.

2.2.2 Keuntungan dan Kelemahan Sistem Pakar

Menurut Andirani (2016:12), Penggunaan sistem pakar secara umum memberikan keuntungan yang dapat dimanfaatkan langsung oleh pengguna. Adapun keuntungan dan penggunaan sistem pakar antara lain:

1. Memungkinkan pengguna yang bukan seorang pakar pada bidang tertentu dapat mengerjakan tugas dari seorang pakar.
2. Bisa melakukan proses yang sama secara berulang.
3. Sistem pakar dapat menyimpan pengetahuan dan keahlian dari pakar.
4. Dengan adanya sistem pakar produktivitas dan output sistem dapat ditingkatkan.



5. Meningkatkan kualitas.
6. Mampu mengambil dan melestarikan keahlian para pakar.
7. Mampu beroperasi dalam lingkungan yang berbahaya,
8. Memiliki kemampuan untuk mengakses pengetahuan.
9. Memiliki reabilitas.
10. Meningkatkan kapabilitas sistem komputer.
11. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
12. Sebagai media pelengkap dalam pelatihan.
13. Meningkatkan kapabilitas dalam penyelesaian masalah.
14. Menghemat waktu dalam pengambilan keputusan.

Sistem pakar juga mempunyai kelemahan selain banyaknya keuntungan yang diberikan, antara lain:

1. Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkan sistem pakar sangat mahal.
2. Sulit dikembangkan, karena ketersediaan pakar di bidangnya dan kepakaran sulit diekstrak dari manusia karena terkadang sulit bagi seorang pakar untuk menjelaskan langkah mereka dalam menangani masalah.
3. Sistem pakar tidak 100% benar karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar. Oleh karena itu setelah pembuatan sistem pakar harus dilakukan pengujian terlebih dahulu secara teliti sebelum digunakan.
4. Pendekatan oleh setiap pakar untuk suatu situasi atau problem bisa berbeda-beda, meskipun sama-sama benar.
5. Transfer pengetahuan dapat bersifat subjektif dan bias.
6. Kurangnya rasa percaya pengguna dapat menghalangi pemakaian sistem pakar.



2.2.3 Pengertian Diagnosa

Menurut Kamus Besar Bahasa Indonesia, “Penentuan jenis penyakit dengan cara meneliti (memeriksa) gejala-gejalanya”. Jadi dapat disimpulkan bahwa diagnosa adalah untuk mengetahui jenis penyakit berdasarkan kondisi yang terjadi.

2.2.4 Jenis-jenis Penyakit Ibu Hamil

Purwoastuti dan Walyani (2015:25), Berikut ini akan dijelaskan beberapa gangguan atau masalah kesehatan pada kehamilan yang sering ditangani pada fasilitas pelayanan kesehatan masyarakat.

1. Kehamilan di Luar Kandungan

Kehamilan di luar kandungan terjadi apabila sel telur yang sudah dibuahi tidak menempel pada rahim, namun pada *tuba falopi*. Hal ini memang jarang terjadi, namun apabila hal ini dibiarkan dan tidak diberi tindakan maka akan menimbulkan akibat yang serius. Apabila kondisi ini tidak di diagnosis sedini mungkin, *tuba falopi* bisa pecah dan memengaruhi kesuburan.

2. Keguguran dan Kematian Bayi saat Melahirkan

Keguguran adalah kondisi kematian bayi pada usia di bawah 20 minggu kehamilan. Kematian bayi adalah kondisi kematian bayi setelah berusia dua puluh minggu masa kehamilan. Angka statistik menunjukkan jika keguguran terjadi pada saat kehamilan menginjak usia dua belas minggu masa kehamilan dan biasanya disebabkan oleh ketidaknormalan yang terjadi pada janin. Keguguran sering disebabkan oleh ketidakmampuan leher rahim yang terbuka dalam waktu yang sama sebelum waktunya. Kekurangan plasenta juga dapat menyebabkan terjadinya keguguran, yakni ketika plasenta tidak bisa mencukupi kebutuhan oksigen dan nutrisi untuk membentuk bayi.

3. Pembekuan Darah

Pembekuan darah ditandai oleh rasa gatal berlebih yang di. alami oleh ibu hamil. Pada kondisi normal hati akan memproduksi empedu yang akan bergerak melalui usus untuk membantu proses pencernaan makanan. Pada ibu yang menderita pembekuan darah, kadar asam empedu akan meningkat dan



akan menghasilkan racun yang akan memasuki darah ibu dan mengakibatkan beberapa gejala. Kondisi seperti ini harus segera diidentifikasi karena bisa mendatangkan dampak yang serius untuk kesehatan bayi, terutama jika sudah memasuki masa kehamilan 36 minggu.

4. *Plasenta Previa*

Plasenta previa adalah kelainan posisi *plasenta* yang terletak di rahim bagian bawah. Biasanya pasien akan diinformasikan jika mereka mengalami kelainan ini dan pada umumnya 95% dari kasus ini *plasenta* biasanya akan kembali ke posisi yang semestinya setelah masa kehamilan memasuki usia enam bulan.

5. *Pemisahan Plasenta Prematur*

Keadaan ini terjadi pada saat *plasenta* terkelupas dari rahim. Ibu akan mengalami pendarahan hebat dan bayi akan kekurangan oksigen juga nutrisi. Namun, sangat jarang bisa terjadi.

6. *Pre-Eklamsia*

Pre-eklamsia dapat menyebabkan terjadinya kegagalan pada *plasenta* dan *eklamsia*. *Eklamsia* dapat menyebabkan kejang koma, dan terkadang dapat menyebabkan kematian. Hal ini dapat dideteksi oleh dokter dan bidan pada saat melakukan pemeriksaan rutin.

7. *Hidramnion*

Yaitu kehamilan dengan jumlah air ketuban lebih dari 2 liter. Keadaan ini mulai tampak pada trimester III, dapat terjadi secara perlahan-lahan atau sangat cepat. Pada kehamilan normal, jumlah air ketuban 3 sampai 1 liter. Karena rahim sangat besar akan menekan organ tubuh di sekitarnya.

8. *Anemia*

Anemia adalah kekurangan darah yang dapat mengganggu kesehatan ibu pada saat proses persalinan. Kondisi ibu hamil dengan kadar *hemoglobin* kurang dari 11 gr % pada trimester I dan III dan <105 gr % pada trimester II. *Anemia* dapat menimbulkan dampak buruk terhadap ibu maupun janin, seperti infeksi, *partus prematurus*, *abortus*, kematian janin, cacat bawaan.

Wanita tidak hamil mempunyai nilai normal *hemoglobin* 12 sampai 15 gr %. Angka tersebut juga berlaku untuk wanita hamil terutama wanita yang



mendapat pengawasan selama hamil. Oleh karena itu, pemeriksaan hemoglobin harus menjadi pemeriksaan darah rutin selama pengawasan antenatal, yaitu dilakukan setiap 3 bulan atau paling sedikit 1 kali pada pemeriksaan pertama atau pada trimester I dan sekali lagi pada trimester III.

2.2.5 Pengertian Konsultasi

Menurut Mujahidin (2018:133) “Aktivitas konsultasi atau perundingan seperti klien dengan penasehat hukumnya.”

Sedangkan menurut Kamus Besar Bahasa Indonesia *online*, “pertukaran pikiran untuk mendapatkan kesimpulan (nasihat, saran, dan sebagainya) yang sebaik-baiknya.”

Berdasarkan pengertian di atas dapat disimpulkan bahwa, konsultasi adalah proses memberikan bantuan kepada seseorang oleh orang yang ahli dan memenuhi standar kualifikasi pada area tertentu untuk mengetahui tentang diri mereka, mengembangkan potensi, menyelesaikan masalah, membuat keputusan, penyetulan diri, dan lain-lain.

2.2.6 Metode *Forward Chaining*

Menurut Andriani (2016:15), “*Forward Chaining* merupakan cara penalaran dengan memulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis atau mencocokkan fakta atau pernyataan dimulai dari bagian sebelah kiri dulu (*IF* dulu). *Forward Chaining* merupakan grup dari *multiple* inferensi yang melakukan pencarian dari suatu masalah kepada solusinya. Jika klausa premis sesuai dengan situasi (bernilai *TRUE*), maka proses akan meng-assert konklusi. *Forward chaining* cocok digunakan untuk suatu aplikasi yang menghasilkan *tree* yang lebar dan tidak dalam.”

Menurut Puji dan Usti dalam Pratama (2019 : 8) “*Forward chaining* adalah teknik pencarian yang dimulai dengan fakta yang diketahui dengan fakta yang diketahui kemudian mencocokkan fakta-fakta tersebut dengan bagian *IF* dari *rule IF-THEN*. Bila ada fakta yang cocok dengan bagian *IF*, maka tersebut dieksekusi. Bila sebuah *rule* dieksekusi, maka sebuah fakta baru (bagian *THEN*)

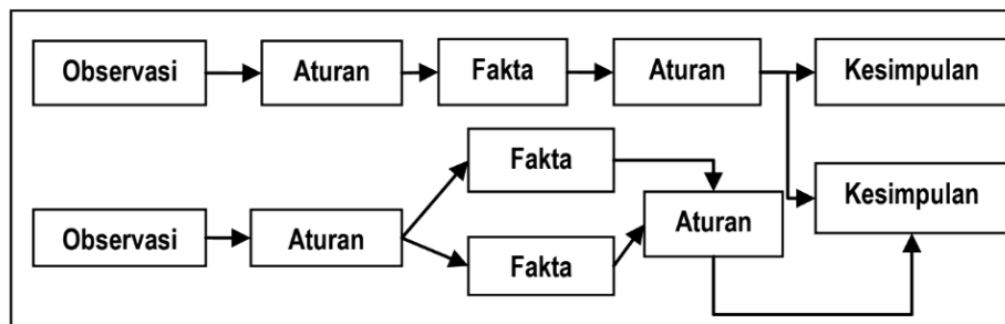


ditambahkan kedalam *database*. Setiap kali pencocokkan, dimulai dari *rule* teratas. Setiap *rule* hanya boleh dieksekusi sekali saja. Proses pencocokkan berhenti bila tidak ada lagi *rule* yang bisa dieksekusi”.

Sedangkan menurut Russel S. Norvig P (2018 : 9) “*Forward Chaining* adalah metode pencarian atau teknik pelacakan ke depan yang dimulai dengan informasi yang ada dan penggabungan rule untuk menghasilkan suatu kesimpulan atau tujuan.”

Pada metode *Forward Chaining*, pencarian dapat dilakukan dengan dua cara, yaitu:

1. Menginputkan semua data ke dalam sistem pakar dalam sesi konsultasi. Cara seperti ini tepat dan berguna pada sistem pakar dimana proses di dalamnya terotomatisasi dan langsung menerima data dari database atau dari satu set sensor.
2. Memberikan elemen spesifik dari data yang diperoleh selama sesi konsultasi dalam sistem pakar. Cara ini mengurangi jumlah data yang diminta, sehingga data yang diminta hanya data yang benar-benar dibutuhkan oleh sistem pakar tersebut yang nantinya akan digunakan untuk mengambil keputusan.



Gambar 2.1 Proses Inferensi *Forward Chaining*

2.2.7 Klinik Mitra Ananda

Klinik Mitra Ananda merupakan salah satu rumah bersalin di Kota Palembang. Rumah bersalin ini melayani pasien ibu yang akan melahirkan, cek kandungan, pemasangan KB, penanganan bayi prematur dan lainnya bagi warga Kota Palembang. Rumah bersalin mitra ananda di dukung oleh tenaga dokter ahli,



bidan dan perawat berkualitas. Pelayanan juga berkualitas dengan alat-alat medis yang modern dan lengkap.

2.2.8 Pengertian Sistem Pakar Diagnosa Penyakit dan Konsultasi Ibu Hamil pada Klinik Mitra Ananda

Sistem Pakar Diagnosa Penyakit dan Konsultasi Ibu Hamil pada Klinik Mitra Ananda adalah sistem yang dibuat dan dirancang peneliti untuk membantu mempermudah penanganan pasien khususnya Ibu hamil. Sehingga pasien dapat melakukan konsultasi secara online, dapat mengetahui diagnosa awal dari keluhan yang dialami, dan dapat menambah pengetahuan mengenai penyakit yang dialami.

2.3 Teori Khusus

2.3.1 Pengertian *Data Flow Diagram (DFD)*

Menurut Soufitri (2019:240), “*Data Flow Diagram (DFD)* adalah representasi grafis dari aliran data melalui sistem informasi. Hal ini memungkinkan untuk mewakili proses dalam sistem informasi dari sudut pandang data. DFD memungkinkan untuk memvisualisasikan bagaimana sistem beroperasi, apa sistem menyelesaikan dan bagaimana itu akan dilaksanakan, bila disempurnakan dengan spesifikasi lebih lanjut. *Data flow diagram* digunakan oleh analis sistem untuk merancang sistem pemrosesan informasi tetapi juga sebagai cara untuk model seluruh organisasi”.

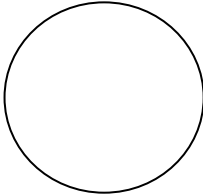
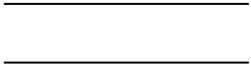
Sedangkan menurut Sutabri dikutip Veza dan Ropianto (2019:35), “*Data Flow Diagram* adalah suatu network yang menggambarkan suatu sistem automat atau komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sitem yang saling berhubungan sesuai aturan mainnya.”

DFD tidak sesuai untuk memodelkan sistem perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Jangan mencampuradukkan pemrograman terstruktur dan pemrograman


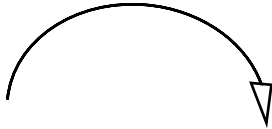


berorientasi objek. Notasi-notasi DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Data Flow Diagram* (DFD)

No	Notasi	Keterangan
1		<p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</p> <p>catatan: nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
2		<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi table-table basis data yang dibutuhkan, table-table ini juga harus sesuai dengan perancangan table-table pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))</p> <p>catatan: nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>

Lanjutan tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Notasi	Keterangan
3		Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan catatan: nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.
4		Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>) catatan: nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.

(Sumber : Sukamto dan Shalahuddin (2016:71-73))

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD :

1. Membuat DFD Level 0 atau sering disebut juga *context diagram*.

DFD Level 0 menggambarkan system yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun *system* lain. DFD level 0 digunakan untuk menggambarkan interaksi antara *system* yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1.



DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2.

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul ada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya.


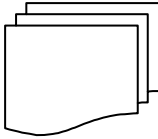
DFD Level 3,4,5 dan seterusnya merupakan *breakdown* dari modul pada DFD di atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.3.2 Pengertian *BlockChart*

Menurut Kristianto (2018:75), “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”.

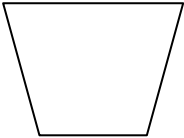

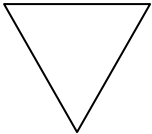
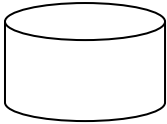

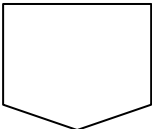
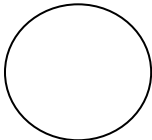

Adapun simbol-simbol yang sering digunakan dalam *BlockChart* dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol pada *BlockChart*

Simbol	Keterangan
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel, berkas atau cetakan
	Multi dokumen





Lanjutan Tabel 2.2 Simbol-simbol pada *BlockChart*

Simbol	Keterangan
	Proses Manual
	Proses yang dilakukan oleh komputer
	Menandakan dokumen yang diarsipkan (arsip manual)
	Data penyimpanan (data storage)
	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran



Lanjutan Tabel 2.2 Simbol-simbol pada *Blockchart*

Simbol	Keterangan
	Layar peraga (monitor)
	Pemasukan data secara manual

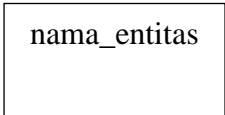
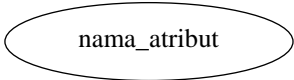
(Sumber : Kristanto (2018:75-77))

2.3.3 Pengertian *Entity Relationship Diagram (ERD)*

Menurut Sukamto dan Shalahuddin (2016:50), “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram (ERD)*. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relational”.

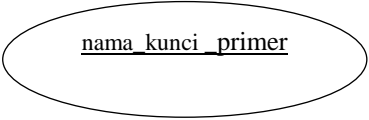
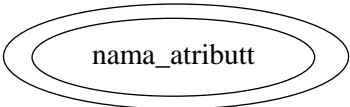
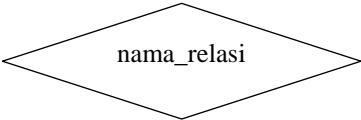
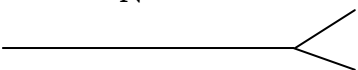
Rosa A.S dan Shalahuddin. (2016:50-51), menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu:

Tabel 2.3 Simbol-simbol pada ERD

No.	Simbol	Deskripsi
1.	Entitas / entity 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya dapat diakses oleh aplikasi computer. penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh di simpan dalam suatu entitas.



Lanjutan Tabel 2.3 Simbol-simbol pada ERD

No.	Simbol	Deskripsi
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4	Atribut multi nilai / <i>multi value</i> 	<i>Field</i> atau kolom data butuh disimpan dalam satu entitas yang dapat dimiliki nilai lebih dari satu.
5	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

(Sumber : Sukamto dan Shalahuddin (2016:50-51))






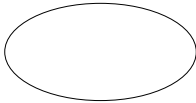
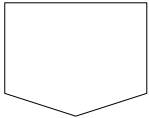
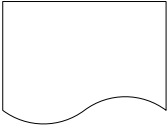

2.3.4 Pengertian *Flowchart*

Menurut Indrajani dalam Rusmawan (2019:48), “*Flowchart* merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program.”


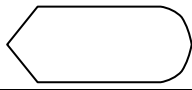


Sedangkan menurut Pahlevy dalam Rusmawan (2019:48), “*Flowchart* (bagan alir) merupakan sebuah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program yang menyatakan arah alur program tersebut”. Gambaran simbol-simbol *Flowchart* sebagai berikut:

Tabel 2.4 Simbol-simbol *Flowchart*

Simbol	Fungsi <i>Flowchart</i>
	Terminal menyatakan awal atau akhir dari suatu algoritma.
	Menyatakan proses
	Proses yang terdefinisi atau sub program
	Persiapan yang digunakan untuk member nilai awal suatu besaran
	Menyatakan masukan dan keluaran (input/output)
	Menyatakan penyambung kesimbol lain dalam satu halaman
	Menyatakan penyambung kehalaman lainnya
	Menyatakan pencetakan (dokumen) pada kertas.
	<i>Multidocument</i> (banyak dokumen).

**Lanjutan Tabel 2.4** Simbol-simbol *Flowchart*

Simbol	Fungsi <i>Flowchart</i>
	<i>Delay</i> (penundaan atau kelambatan).
	Layar peraga (monitor)

(Sumber : Rusmawan (2019:49))

2.3.5 Pengertian Kamus Data

Menurut Rusmawan (2019:36), Kamus data merupakan katalog fakta data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dibuat berdasarkan arus data yang ada pada diagram alir data. Kamus data harus juga dapat mencerminkan keterangan yang jelas tentang data yang dicatat.

Menurut Sukamto dan Shalahuddin (2016:73), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (output) dapat dipahami secara umum (memiliki standar cara penulisan).”

Sedangkan menurut Sukamto dan Shalahuddin (2016:73) menyatakan bahwa, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (output) dapat dipahami secara umum (memiliki standar cara penulisan)”. Berikut beberapa simbol-simbol yang terdapat pada kamus data:

Tabel 2.5 Simbol-simbol Kamus Data

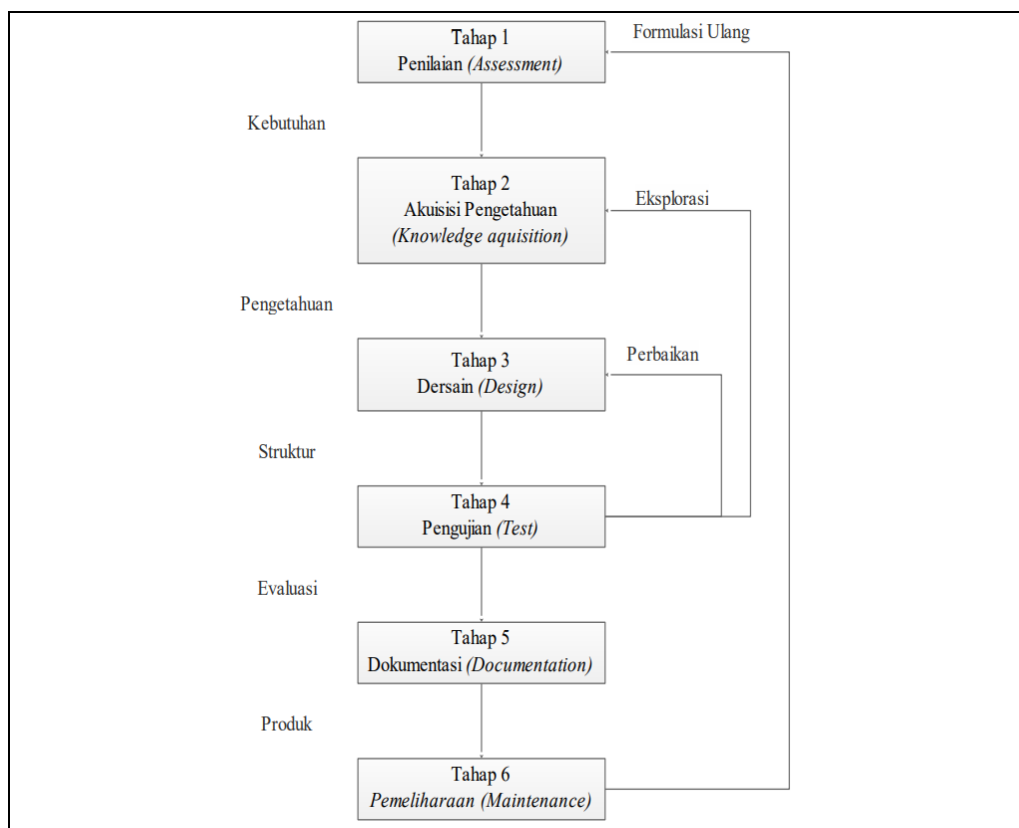
Simbol	Artinya
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau...
{ }n	N kali/ bernilai banyak
()	Data opsional
...	Batas komentar

(Sumber : Sukamto dan Shalahuddin (2016:74))



2.3.6 Metode *Expert System Development Life Cycle* (ESDLC)

Sunarya dan Destiani (2016:86-87), Metode pengembangan sistem yang digunakan dalam penelitian ini adalah model *Expert System Development Life Cycle* (ESDLC). Dalam pengembangan sistem pakar, akan digunakan pendekatan konvensional dengan metode *Expert System Development Life Cycle* (ESDLC) [4]. Tahap-tahap yang harus dilakukan pada metode ESDLC adalah sebagai berikut :



Gambar 2.2 Tahap Pengembangan Sistem Pakar

A. Tahap 1 Penilaian Keadaan (*Assessment*)

Penilaian merupakan tahap penentuan hal penting sebagai dasar dari permasalahan mengenai penyakit bawang merah dengan mengkaji dan membatasi masalah yang akan diimplementasikan. Adapun langkah – langkah yang dilakukan pada tahapan ini adalah :

1. Kelayakan dan Justifikasi Masalah Mendefinisikan masalah hama dan penyakit pada bawang merah yang ada di lapangan, selain itu penjelasan



mengenai kelayakan pemilihan topik yang akan dibahas untuk dijadikan suatu sistem pakar.

2. Tujuan Pengembangan Sistem Pakar Memaparkan tujuan dari pengembangan sistem pakar diagnosis penyakit bawang merah yang akan dibuat, dimana tujuan tersebut didapat dari hasil tahap sebelumnya.
3. Analisis Kebutuhan Memaparkan proses analisis menentukan hal – hal apa saja yang dibutuhkan pada saat pengembangan sistem pakar diagnosis hama dan penyakit bawang merah.
4. Sumber Pengetahuan Memaparkan bagaimana cara mendapatkan data mengenai penyakit bawang merah, serta apa saja gejala dan solusi pencegahan dan pengobatannya.

B. Tahap Akuisi Pengetahuan (*Knowledge Acquisition*)

Akuisi pengetahuan yang dilakukan adalah untuk mengetahui penyakit dan hama bawang merah yang akan digunakan untuk panduan dalam upaya pengembangan sistem pakar diagnosis penyakit bawang merah, dimana dilakukan pengumpulan data tentang macam – macam hama dan penyakit pada bawang merah, gejala, penyebab serta cara pengobatannya. Setelah data dikumpulkan maka akan digabung dan dibuat tabel basis pengetahuan lalu dibuat tabel keputusannya untuk mempermudah dalam pengembangan sistem pakar.

C. Tahap Perancangan (*Design*)

Tahap perancangan sistem merupakan suatu proses, desain dan cara supaya sistem dapat berjalan seperti yang diharapkan. Pengetahuan yang diperoleh dalam tahap akuisisi pengetahuan digunakan untuk melakukan pendekatan dengan merepresentasikan pengetahuan pakar serta sebagai strategi pemecahan masalah ke dalam sistem pakar. Tahapan desain ini berisi penetapan keseluruhan struktur dan organisasi dari sistem pengetahuan dan pembangunan *prototype sistem*.

1. Representasi Pengetahuan

Representasi pengetahuan merupakan metode yang digunakan untuk mengodekan pengetahuan dalam sebuah sistem pakar yang berbasis



pengetahuan yang dimaksud untuk mengambil sifat penting problem serta membuat informasi dapat diakses oleh prosedur pemecahan problema.

2. Basis Pengetahuan

Basis pengetahuan berisi pengetahuan – pengetahuan pakar berupa fakta – fakta, konsep, aturan, prosedur, dan hubungan diantaranya, yang telah direpresentasikan dalam bentuk yang dimengerti oleh sistem

3. Mesin Inferensi

Mesin inferensi merupakan bagian dari sistem pakar yang berfungsi sebagai otak yang melakukan penalaran untuk mengarahkan pengetahuan dari knowledge base hingga mencapai kesimpulan yang harus dieksekusi atau dilaksanakan.

4. Perancangan Basis Data

Basis data dalam pengembangan sistem pakar ini berperan sebagai basis pengetahuan yang menyimpan pengetahuan kepakaran yang diperlukan untuk mengolah data-data dan aturan yang berhubungan dengan penyakit tanaman bawang merah.

5. Struktur menu dan Antarmuka

Struktur menu menggambarkan menu-menu yang terdapat di dalam sistem pakar.

D. Tahap Pengujian (*Test*)

Sebelum pada tahap pengujian dilakukan perancangan struktur menu, dan perancangan antarmuka, selanjutnya dilakukan pengujian dengan menggunakan Black Box Test dan Alfa Test yang dilakukan oleh Pakar. Metode ini akan menentukan masukkan dan keluarannya apakah sudah benar atau salah. Pengujian Black Box ini merupakan suatu strategi testing yang hanya memperhatikan kepada faktor fungsionalitas dan spesifikasi dari perangkat lunak (software).

E. Dokumentasi (*Documentation*)

Tahap dokumentasi menangani kebutuhan untuk mengumpulkan seluruh informasi proyek ke dalam dokumen yang dapat memenuhi persyaratan baik pengguna dan pengembang sistem pakar, dokumentasi menampung kebutuhan pengguna yang sesuai dengan persyaratan yang ditemukan dalam kebanyakan



proyek perangkat lunak, dokumentasi menjelaskan bagaimana cara untuk mengoperasikan sistem dan mungkin menyediakan tutorial langkah-langkah melalui fitur operasional penting dari sebuah sistem. Dokumentasi harus berisi kamus pengetahuan yang menyediakan presentasi terorganisasi dari pengetahuan sistem dan prosedur pemecahan masalah, dokumentasi proyek berperan untuk menambah pengetahuan baru.

F. Pemeliharaan (*Maintenance*)

Setelah sistem ini diterapkan di lingkungan kerja, perlu dipertahankan secara berkala. Pengetahuan sistem perlu disempurnakan atau diperbarui untuk memenuhi kebutuhan saat ini. Perubahan kebutuhan sistem utama juga dapat terjadi yang akan memerlukan reformulasi spesifikasi sistem. Oleh karena itu, sangat penting perawatan yang efektif untuk keamanan proyek sistem pakar.

2.4 Teori Program

2.4.1 Pengertian *Laravel*

Menurut Utama dan Nandika dikutip Harianto dkk (2019:14), "*Laravel* merupakan sebuah *framework PHP* yang akan membantu para *web developer* untuk menciptakan aplikasi istimewa secara *simple*, kode yang bersih dan menyenangkan. Dalam penelitian ini *framework laravel* digunakan untuk mempermudah perancangan *PHP*".

Menurut Pakpahan (2020:1), "*Laravel* adalah *framework* aplikasi *web* yang dapat menggunakan *syntax* yang *expressive* dan *elegant*. Tim *Laravel* telah membuat fondasi yang dibutuhkan untuk membebaskan penggunaanya untuk bereaksi tanpa perlu terlalu memikirkan hal-hal kecil".

Menurut Yudhanto dan Prasetyo (2019:22), "*Laravel* adalah sebuah *framework PHP* yang dirilis di bawah lisensi MIT dan dibangun dengan konsep MVC (*model view controller*). *Laravel* adalah pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya meliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu.



2.4.2 Pengertian XAMPP



Gambar 2.3 Logo XAMPP

Menurut Aprilian dan Saputra (2020:113), “XAMPP adalah perangkat lunak sumber terbuka yang dikembangkan oleh teman-teman *Apache*. Paket perangkat lunak XAMPP berisi distribusi *Apache* untuk *server Apache*, *MariaDB*, *PHP*, dan *Perl*. Dan itu pada dasarnya adalah tuan rumah *local* atau *server* lokal. *Server* lokal ini berfungsi di komputer *desktop* atau laptop Anda sendiri”.

Menurut Roza dkk (2020:82), “XAMPP adalah perangkat lunak bebas, yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan meng-*install* XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server Apache*, *PHP* dan *MySQL* secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi”.

Dengan demikian dapat disimpulkan bahwa pengertian XAMPP adalah perangkat lunak yang berfungsi untuk pembuatan database.

2.4.3 Pengertian MySQL



Gambar 2.4 Logo MySQL

Menurut Roza dkk (2020: 88), “MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berlisensi yang penggunaannya terbatas). Jadi MySQL adalah *database server* yang gratis dengan lisensi *GNU General Public License (GPL)* sehingga dapat Anda pakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada”.



Menurut Lestanti dan Susana (2016: 72), “*MySQL* adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.”

Jadi dapat disimpulkan bahwa *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* yang *multithread*, *multi-user*.

2.4.4 Pengertian *Framework*

Definisi *Framework* menurut Habibi dan Sandi (2020: 60) “*Framework* dapat diartikan sekumpulan perintah/program dasar dimana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa harus membuat program dari awal”.

Sedangkan menurut Yudhanto dan Prasetyo (2019:8) “*Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan variabel, *file*, dan lain-lain sehingga pekerjaan *developer* lebih fokus dan lebih cepat dalam membangun aplikasi.”

Dapat di ambil kesimpulan bahwa *Framework* adalah berisi program yang memudahkan dalam pembuatan aplikasi tanpa harus membuat program dari awal.

2.4.5 Pengertian *PHP*

Menurut Yudhanto dan Prasetyo (2019:7), “*PHP* atau *Hypertext Preprocessor* adalah bahasa pemrograman *script server side* yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan *web*. Bahasa pemrograman ini memang dirancang untuk para pengembang *web* agar dapat menciptakan suatu halaman *web* yang bersifat dinamis.”



Sedangkan menurut Supono dan Putratama (2018:3), “*PHP (PHP: Hypertext Preprocessor)* adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat di mengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam *HTML*”.

Menurut Rusli dkk, (2016:63) “*PHP* adalah bahasa *scripting server* dan alat yang ampuh untuk membuat halaman *web* dinamis dan interaktif.”

2.4.6 Pengertian *Sublime Text*



Gambar 2.5 Logo *Sublime Text*

Menurut Pasaribu (2017:158) “*Sublime Text Editor* adalah *editor* teks untuk berbagai bahasa pemrograman termasuk pemrograman *PHP*. *Sublime Text Editor* merupakan *editor text* lintas *platform* dengan *Python Application Programming Interface (API)*”

Sedangkan menurut Haughee dikutip Habibi dan Suryansah (2020:60), “Bahwa *Sublime Text* adalah aplikasi *editor* untuk kode dan teks yang dapat berjalan di berbagai *platform operating system* dengan menggunakan teknologi *Python API*.”

Keunggulan *Sublime Text* :

1. Keunggulan pertama adalah adanya *multiple selection*. Fungsinya yaitu untuk menghasilkan perubahan pada sebuah kode dalam waktu yang sama dan dalam baris yang berbeda. *Multiple selection* ini juga adalah salah satu fitur unggulan dari *Sublime Text Editor*. Anda bisa saja meletakkan kursor pada kode yang akan di ubah / *edit*, lalu tekan *Ctrl + click* atau blok kode yang akan diubah kemudian *Ctrl + D*. Nah, setelahnya Anda bisa merubah kode secara bersamaan.



2. Keunggulan kedua adalah adanya *command pallette*. Fitur itu mempunyai fungsi yang berguna untuk mengakses *file shortcut* dengan gampang. Untuk mencari *file* tersebut Anda bisa tekan *Ctrl + Shift + P*, kemudian cari perintah yang memang Anda inginkan.
3. Keunggulan ketiga adalah adanya *distraction free mode*. Fitur satu ini punya fungsi untuk merubah tampilan layar menjadi penuh dengan menekan *SHIFT + F11*. Fitur ini sebenarnya memang sangat dibutuhkan ketika Anda ingin fokus pada bagian pekerjaan tertentu yang sedang dikerjakan saja.
4. Keunggulan keempat adalah adanya *find in project*. Dalam fitur ini Anda bisa mencari dan membuka *file* di dalam sebuah project dengan cepat dan mudah. Anda tinggal tekan *Ctrl + P* dan Anda sudah bisa mencari file yang di inginkan.
5. Keunggulan terakhir, *plugin* yang berbasis *Python Plugin API*. Teks editor ini punya plugin yang sangat beragam dan ini bisa memudahkan Anda dalam mengembangkan *software*-nya.



2.5 Referensi Jurnal

Dalam Tugas Akhir ini penulis menggunakan beberapa perbandingan jurnal. Dibawah ini akan diuraikan perbandingan dari referensi jurnal tersebut.

Tabel 2.6 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
1.	Sistem Pakar Diagnosa Keluhan Selama Masa Kehamilan Menggunakan Metode <i>Certainty Factor</i> Berbasis <i>Web</i> (Vol. 6, No. 1. ISSN : 2089-3787) Penulis : Arief Hasbiyanor dan Bahar Tahun : 2017	Bagaimana cara membantu petugas atau bidan muda dalam mendiagnosa penyakit ibu hamil secara cepat dan tepat saat bidan yang lebih berpengalaman tidak berada di tempat..	Sistem Pakar	Metode <i>Certainty Factor</i> , data uji, diagram pohon keputusan, perhitungan manual	Sistem pakar untuk mendiagnosa keluhan selama masa kehamilan berbasis Web menggunakan <i>Certainty Factor</i> yang nantinya dapat membantu petugas atau bidan muda agar tahu bagaimana solusi mengatasi suatu penyakit pada kehamilan



Lanjutan Tabel 2.6 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
2.	Sistem Pakar Berbasis Web Dengan Metode <i>Forward Chaining</i> Dalam Mendiagnosis Dini Penyakit <i>Tuberkulosis</i> di JawaTimur. (Vol.1, No.3. ISSN : 2503-2259) Penulis : Windah Supartini dan Hindarto Tahun : 2016	Bagaimana cara menekan jumlah penderita <i>Tuberkulosis</i> dengan adanya tindakan pendeteksian dini gejala penyakit <i>Tuberkulosis</i> .	Sistem Pakar, Tabel Relasi, DFD, <i>Flowchart</i>	Metode <i>Forward Chaining</i> , Memeriksa berkas pasien <i>Tuberkulosis</i> dan melakukan wawancara	Membantu mendiagnosis penyakit <i>Tuberkulosis</i> berdasarkan gejala-gejala yang dialami oleh pasien dan Hasil diagnosis dari sistem pakar yang dialami pasien sesuai dengan diagnosis dokter penyakit <i>Tuberkulosis</i> .



Lanjutan Tabel 2.6 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
3.	Penerapan <i>Teorema Bayes</i> Untuk Diagnosis Penyakit Pada Ibu Hamil Berbasis Android (Vol.6, No.1. ISSN : 2355-6579) Penulis : Nadya paramitha , Erfian Junianto , dan Sari Susanti Tahun : 2019	Bagaimana cara memberikan pengetahuan mengenai diagnosa penyakit gangguan kehamilan pada ibu hamil, memberikan sarana media konsultasi mengenai penyakit pada kehamilan serta mengurangi banyaknya biaya konsultasi ke dokter ahli.	Sistem Pakar	Metode <i>Teorema Bayes</i> , pengumpulan data dengan wawancara, observasi dan studi pustaka	Aplikasi sistem pakar yang dapat memberikan pengetahuan mengenai diagnosa penyakit gangguan kehamilan, menjadi media untuk berkonsultasi mengenai penyakit pada masa kehamilan.



Lanjutan Tabel 2.6 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
4.	Aplikasi Sistem Pakar Diagnosa Penyakit Pada Anak Bawah Lima Tahun Menggunakan Metode <i>Forward Chaining</i> (Vol.3, No.1. ISSN : 2443- 2555) Penulis : Bagus Fery Yanto, Indah Werdiningsih, dan Endah Purwanti Tahun : 2017	Bagaimana cara membuat sebuah aplikasi sistem pakar diagnosa penyakit pada Balita berbasis mobile, untuk membantu dalam menurunkan angka kematian Balita di Indonesia	Sistem pakar	Metode <i>Forward Chaining</i> , pengumpulan data dengan wawancara dan studi pustaka	Aplikasi sistem pakar berbasis <i>mobile</i> dengan fitur diagnosa penyakit, riwayat diagnosa dan kumpulan penyakit. Aplikasi sistem pakar yang dibuat dapat mendiagnosa penyakit dan memberikan saran penanganan



Lanjutan Tabel 2.6 Referensi Jurnal

No	Judul /Penulis/Tahun	Masalah	Teori	Metode Penelitian	Hasil
5.	Sistem Pakar Penyakit Lambung menggunakan Metode <i>Forward Chaining</i> (Vol.13, No.1. ISSN 0216-7395) Penulis : Andi Nurkholis, Agung Riyantomo dan Mohammad Tafrik Tahun : 2017	Bagaimana cara membantu mendefinisikan dan mendiagnosa suatu penyakit berdasarkan gejala-gejala yang dialami oleh manusia khususnya penyakit bagian lambung	Sistem pakar	Metode <i>Forward Chaining</i>	Sistem pakar ini memiliki keakuratan diagnosa sebesar 75% dan dapat dikategorikan baik, meskipun diperlukan lagi perbaikan-perbaikan dari segi keefektifan dan kelengkapan informasi.