

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

Menurut landasan teori ini perlu di tegakkan agar penelitian itu mempunyai dasar yang kokoh, dan bukan sekedar perbuatan coba-coba (*trial and error*). adanya landasan teoritis ini merupakan ciri bahwa penelitian itu merupakan cara ilmiah untuk mendapatkan data.

Suatu teori juga dapat menuju pada suatu cara menerangkan yang menggeneralisasi. Di sini biasanya terdapat hubungan yang fungsional antara data dan pendapat yang teoritis. teori adalah suatu konseptualisasi yang umum, Konseptualisasi atau sistem pengertian ini di diperoleh melalui jalan yang sistematis(Sugiyono, 2014: 52-53).

2.1.1. Kecerdasan buatan atau *Artificial Intelligence* (AI)

Kecerdasan berasal dari Bahasa inggris “*artificial Intelligence*” atau disingkat AI, yaitu *intelligence* adalah kata sifat yang berarti cerdas, sedangkan artificial artinya buatan. Kecerdasan buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan di ambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. Cerdas berarti memiliki pengetahuan, pengalaman, dan penalaran untuk membuat keputusan dan mengambil tindakan. Untuk membuat sebuah mesin menjadi cerdas (dapat

bertindak seperti manusia) maka harus diberi bekal pengetahuan dan diberi kemampuan untuk menalar. Kecerdasan buatan memungkinkan komputer untuk berpikir atau menalar dan menirukan proses belajar manusia sehingga informasi baru dapat diserap sebagai pengetahuan, pengalaman, dan proses pembelajaran serta dapat digunakan sebagai acuan di masa-masa yang akan datang(Sutojo, Mulyanto and Suhartono, 2011: 3).

Istilah AI pertama kali di kemungkakan pada tahun 1956 di konferensi darthmouth, sejak saat itu , AI terus di kembangkan sebab sebagai penelitian mengenai teori-teori dan prinsi-prinsipnya juga terus berkembang, meskipun istilah AI baru muncul pada tahun 1956, tetapi teori teori yang mengarah ke AI sudah muncul sejak tahun 1941(Prapti, Kanthi Pangestuning, Ridwan Iskandar, 2015).

3.1.1.1 Fuzzy logic

Logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, *multi-channel* atau *workstation* berbasis akusisi data dan sistem kontrol. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya. Dalam logika klasik dinyatakan bahwa segala sesuatu bersifat biner, yang artinya adalah hanya mempunyai dua kemungkinan, “ya atau tidak”, “benar atau salah”, “baik atau buruk”, dan lain lain (Sutojo, Mulyanto and Suhartono, 2011:211).

Fuzzy logic memiliki beberapa metode inferensinya menurut yaitu:

1. Metode Tsukamoto
2. Metode Mamdani
3. Metode Sugeno

3.1.1.2 Jaringan Saraf Tiruan (JST)

Jaringan saraf tiruan paradigma pengolahan informasi yang terinspirasi dari sistem saraf secara biologis, seperti sistem informasi pada otak manusia. Elemen kunci dari paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang saling berhubungan (*neuron*), bekerja serentak untuk menyelesaikan masalah tertentu. Cara kerja JST seperti cara kerja manusia, yaitu belajar melalui contoh. Sebuah JST dikonfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau klasifikasi data, melalui proses pembelajaran. Kelebihan kelebihan yang diberikan JST antara lain (Sutojo, Mulyanto and Suhartono, 2011: 283-284):

- a) *Belajar Adaptive*: kemampuan untuk mempelajari bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan atau pengalaman awal
- b) *Self-Organisation*: sebuah JST dapat membuat organisasi sendiri atau representasi dan informasi yang diterimanya selama waktu belajar.
- c) *Real Time Operation*: perhitungan JST dapat dilakukan secara paralel sehingga perangkat keras yang dirancang dan diproduksi secara khusus dapat mengambil keuntungan dari kemampuan ini.

Selain mempunyai kelebihan-kelebihan tersebut, JST juga mempunyai kelemahan-kelemahan yaitu tidak efektif jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi, tidak efisien jika digunakan untuk melakukan operasi algoritma aritmatik, operasi logika, dan simbolis, serta untuk beroperasi JST butuh pelatihan sehingga bila jumlah datanya besar, waktu yang digunakan untuk proses pelatihan sangat lama (Sutojo, Mulyanto and Suhartono, 2011: 284-285) .

Salah satu elemen yang menentukan baik tidaknya suatu mode jaringan saraf tiruan adalah hubungan antar-neuron atau arsitektur jaringan. Neuron-neuron tersebut terkumpul dalam lapisan-lapisan yang disebut neuron layers. Terdapat 3 bagian lapisan penyusun jaringan saraf tiruan (Sutojo, Mulyanto and Suhartono, 2011:292):

- a) Lapisan *input (input layer)* adalah unit-unit dalam lapisan input disebut unit-unit input yang bertugas menerima pola inputan dari luar yang menggambarkan suatu permasalahan
- b) Lapisan tersembunyi (*Hidden layer*) adalah unit unit dalam lapisan tersembunyi di sebut unit-unit tersembunyi, yang mana nilai outputnya tidak dapat diamati secara langsung
- c) Lapisan *output (output layer)* adalah unit-unit dalam lapisan output disebut unit-unit output, yang merupakan solusi JST terhadap suatu permasalahan.

3.1.1.3 Sistem Pakar

Sistem pakar dapat membantu manusia dalam menyelesaikan masalah tertentu, suatu sistem yang di rancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan sehingga menemukan jawaban dari permasalahannya. Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna, Sehingga dengan bantuan sistem pakar seseorang yang bukan pakar/ahli dapat menjawab pertanyaan, menyelesaikan masalah dan mengambil keputusan yang biasanya dilakukan oleh seorang pakar.

Didalam sistem pakar ada dua metode teknik inferensi yang digunakan yaitu metode forward chaining dan backward chaining (Sutojo, Mulyanto and Suhartono, 2011:13).

2.1.2 Sistem Pakar (*Expert System*)

Suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah, pada zaman sekarang ini sistem pakar di buat atau diciptakan untuk dapat menirukan keahlian seorang pakar sehingga seseorang yang bukan pakar dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar.

Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna. Dengan bantuan sistem pakar seseorang yang bukan pakar /ahli dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil

keputusan yang biasanya dilakukan oleh seorang pakar (Merlina and Hidayat, 2012:4).

Sistem pakar mulai dikembangkan pada pertengahan 1960, ditandai dengan lahirnya sistem pakar pertama bernama General-purpose Problem Solver (GPS) yang dikembangkan oleh Newel dan Simon. Kemudian bermunculan sistem pakar lain di berbagai bidang seperti MYCIN untuk diagnosis penyakit, DENDRAL untuk mengidentifikasi struktur molekul campuran yang tak dikenal, XCON & XSEL untuk membantu konfigurasi sistem komputer besar, SOPHIE untuk analisis sirkuit elektronik, Prospector digunakan di bidang geologi untuk membantu mencari dan menemukan deposit, FOLIO digunakan untuk membantu memberikan keputusan bagi seorang manajer dalam masalah stok dan investasi, DELTA dipakai untuk pemeliharaan lokomotif listrik diesel, dan sebagainya (Sutojo, Mulyanto and Suhartono, 2011:159-160).

Suatu sistem dikatakan sebagai sistem pakar jika memiliki ciri-ciri sebagai berikut (Sutojo, Mulyanto and Suhartono, 2011:162):

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami.
4. Bekerja berdasarkan kaidah tertentu.
5. Mudah dimodifikasi.
6. Basis pengetahuan dan mekanisme inferensi diletakkan terpisah.
7. Keluarannya (output) bersifat anjuran.

8. Sistem dapat mengaktifkan kaidah secara terpisah secara searah, sesuai dengan dialog dengan pengguna.

Didalam sistem pakar ada dua metode teknik inferensi yang digunakan yaitu metode forward chaining dan backward chaining (Sutojo, Mulyanto and Suhartono, 2011:13):

1. Forward chaining

Teknik pencarian yang di mulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian IF dan rules IF-THEN. Bila ada fakta yang cocok dengan bagian IF, maka rule tersebut di eksekusi. Bila sebuah rule di eksekusi, maka sebuah fakta baru (bagian THEN) ditambahkan kedalam database. Setiap kali pencocokan, dimulai rule teratas, setiap rule hanya boleh dieksekusi sekali saja. Proses pencocokan berhenti bila tidak ada lagi rule yang bisa di eksekusi(Sutojo, Mulyanto and Suhartono, 2011:171).

Konsep ini dapat juga disebut sebagai pencarian yang dimotori data (*data driven search*). Runut maju melakukan proses peruntutan (penalaran) dimulai dari premis-premis atau informasi masukan (*IF*) terlebih dahulu kemudian menuju konklusi atau *derived information* (*THEN*). Konsep ini dapat dimodelkan sebagai berikut:

IF (informasi masukan)

THEN (konklusi)

Informasi masukan dapat berupa suatu pengamatan sedangkan konklusi dapat berupa diagnosa sehingga dapat dikatakan jalannya

penalaran runut maju dimulai dari pengamatan menuju diagnosa. Pada metode ini, sistem tidak melakukan praduga apapun terhadap konklusi, namun sistem akan menerima semua gejala yang diberikan pengguna lalu sistem akan memeriksa gejala-gejala tersebut dan selanjutnya mencocokkan dengan konklusi yang sesuai (Hartati dan Iswanti, 2008: 45-46).

2. Backward chaining

metode inferensi yang bekerja mundur ke arah kondisi awal. Proses diawali oleh goal (yang berada dibagian THEN dari rule IF-THEN), kemudian pencarian mulai dijalankan untuk mencocokkan apakah fakta-fakta yang ada cocok dengan premis-premis dibagian IF. Jika cocok, rule dieksekusi, kemudian hipotesis dibagian THEN ditempatkan di basis data sebagai fakta baru. Jika tidak cocok, simpan premis dibagian IF ke dalam stack sebagai sub goal. Proses berakhir jika goal di temukan atau tidak ada rule yang bisa membuktikan kebenaran dari subgoal atau goal (Sutojo, Mulyanto and Suhartono, 2011:178).

2.1.2.1 Manfaat sistem pakar

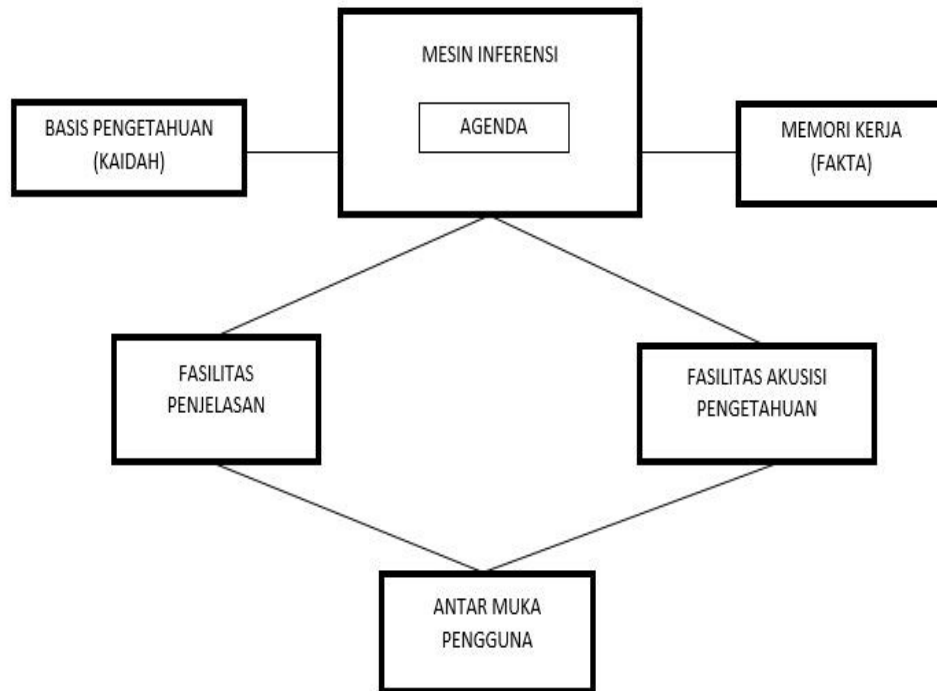
Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang bisa di implementasikan ke berbagai bidang untuk membantu memudahkan pekerjaan manusia seperti bidang kesehatan, pertanian, bidang geologi, dll. Menurut (Sutojo, Mulyanto and Suhartono, 2011: 160) ada beberapa manfaat yang dapat diberikan sistem pakar, diantaranya yaitu:

1. Dapat meningkatkan produktifitas, karena sistem pakar dapat bekerja lebih cepat dari pada manusia
2. Membuat orang yang awam bekerja seperti layaknya sistem pakar
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi tingkat kesalahan
4. Mampu menangkap pengetahuan dan kepakaran seseorang
5. Dapat beroperasi di lingkungan yang berbahaya
6. Memudahkan akses pengetahuan seorang pakar
7. Andal, sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit
8. Meningkatkan kapabilitas sistem komputer
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti
10. Bisa digunakan sebagai media pelengkap dalam pelatihan
11. Meningkatkan kemampuan untuk menyelesaikan masalah.

2.1.2.2 Struktur sistem pakar

Struktur sistem pakar adalah bagian-bagian terpenting yang harus ada dalam pembentukan sebuah sistem pakar, yang meliputi beberapa jenis struktur sistem pakar yang berbeda-beda, tergantung dilihat dari sudut pandang dari struktur pakar tersebut yaitu yang pertama struktur sistem pakar secara umum atau secara menyeluruh, yang kedua struktur sistem pakar yang dilihat dari sudut pandang lingkungan sistem, dan yang ke tiga struktur sistem pakar berbasis kaedah produksi (IF...THEN...), yang akan di jelaskan sebagai berikut yaitu:

1) Struktur sistem pakar secara menyeluruh (umum)



Gambar 2.1 Struktur Sistem Pakar

(Sumber: Hartati and Iswanti, 2008: 4)

Berdasarkan struktur sistem pakar pada gambar 2.1 dapat dijelaskan bahwa:

a) Antar Muka Pengguna

Sistem pakar menggantikan seorang pakar dalam suatu situasi tertentu, maka sistem harus menyediakan pendukung yang diperlukan oleh pemakai yang tidak memahami masalah teknis.

b) Basis Pengetahuan

Basis pengetahuan merupakan kumpulan pengetahuan bidang tertentu pada tingkatan pakar dalam format tertentu.

c) Mesin Inferensi

Mesin inferensi merupakan otak dari sistem pakar, berupa perangkat lunak yang melakukan tugas inferensi penalaran sistem pakar, biasanya dikatakan sebagai mesin pemikir (*thinking machine*).

d) Memori Kerja

Merupakan bagian dari sistem pakar yang menyimpan fakta-fakta yang diperoleh saat dilakukan proses konsultasi.

e) Fasilitas Penjelasan

Fasilitas penjelasan inilah yang dapat memberikan informasi kepada pemakai mengenai jalanya penalaran sehingga dihasilkan suatu keputusan.

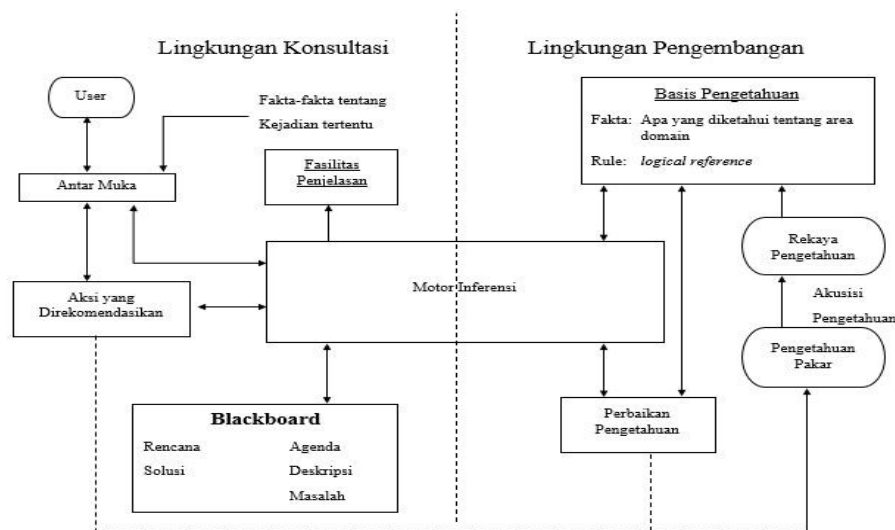
f) Fasilitas Akuisisi Pengetahuan

Proses pengumpulan, perpindahan, dan transformasi dari keahlian/kepakaran pemecahan masalah yang berasal dari beberapa sumber pengetahuan kedalam bentuk yang di mengerti oleh komputer.

2) struktur sistem pakar sudut pandang lingkungan sistem

ada dua bagian dari struktur sistem pakar yang dilihat dari sudut pandang lingkungan sistem yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Menurut (Sutojo, Mulyanto and Suhartono, 2011: 166) lingkungan pengembangan

digunakan oleh pembuat sistem pakar untuk membangun komponen-komponen dalam pengembangan sistem ke dalam basis pengetahuan, sedangkan lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasehat dari sistem pakar layaknya berkonsultasi dengan seorang pakar.



Gambar 2.2 Struktur Sistem Pakar lingkungan sistem

(sumber: Sutojo, Mulyanto and Suhartono, 2011: 167)

Berdasarkan struktur sistem pakar sudut pandang lingkungan sistem gambar 2.2 dapat dijelaskan bahwa:

a) Akuisisi pengetahuan

Subsistem ini digunakan untuk memasukan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan dengan format tertentu (dalam bentuk representasi pengetahuan).

b) Basis pengetahuan

Basis pengetahuan mengandung pengetahuan yang diperlukan untuk memahami, memformulasikan, dan menyelesaikan masalah.

c) Mesin inferensi

Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada memanipulasi dan mengarahkan kaidah, model, dan fakta yang di simpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan.

d) Daerah kerja (*blackboard*)

Untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang terjadi, sistem pakar membutuhkan blackboard, yaitu area pada memori yang berfungsi sebagai basis data.

e) Antarmuka pengguna (*user interface*)

Digunakan sebagai media komunikasi antara pengguna dan sistem pakar.

f) Subsistem penjelasan (*Explanation subsystem/ justifier*)

Berfungsi memberi penjelasan kepada pengguna, bagaimana suatu kesimpulan dapat diambil.

g) Sistem perbaikan pengetahuan (*knowledge refining system*)

Kemampuan untuk memperbaiki pengetahuan (*knowledge refining system*) dari seorang pakar diperlukan untuk menganalisis pengetahuan,

belajar dari kesalahan masalalu, kemudian memperbaiki pengetahuannya sehingga dapat dipakai pada masa mendatang.

h) Pengguna (*user*)

Pada umumnya pengguna sistem pakar bukanlah seorang pakar (non-expert) yang membutuhkan solusi, saran, atau pelatihan (training) dari berbagai permasalahan yang ada.

3) struktur sistem pakar berbasis kaidah produksi (IF...THEN...)

Struktur sistem pakar kaidah produksi merupakan struktur sistem pakar yang berbasis (if...then...) yang sama dengan struktur sistem pakar yang digunakan dalam penelitian ini. Menurut Firebaugh (1988) dalam Hartati dan Iswanti (2008: 10) struktur sistem pakar yang berbasis kaidah produksi terdiri dari 4 komponen, yaitu:

a) Antarmuka pemakai

Sistem pakar menggantikan seorang pakar dalam suatu situasi tertentu, maka system harus menyediakan pendukung yang diperlukan oleh pemakai yang tidak memahami masalah teknik (Hartati dan Iswanti, 2008: 4).

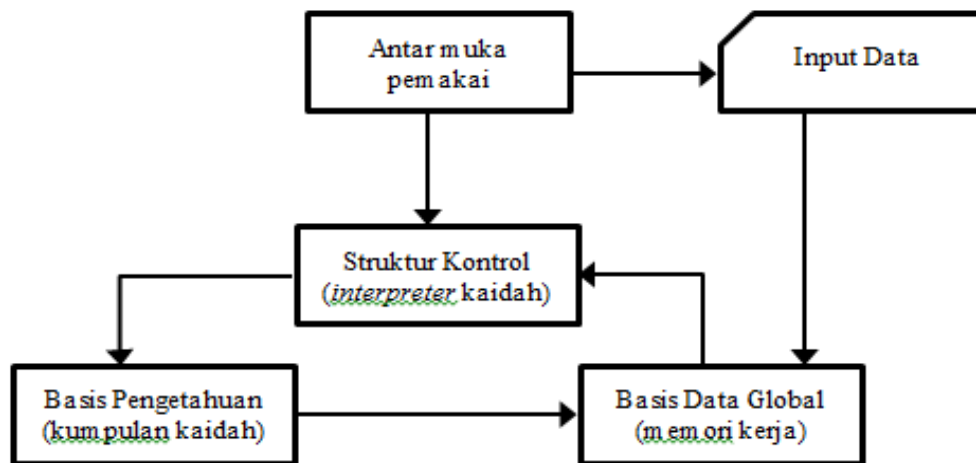
b) Basis pengetahuan

Basis pengetahuan merupakan kumpulan pengetahuan bidang tertentu pada tingkatan pakar dalam format tertentu. Pengetahuan ini diperoleh dari akumulasi pengetahuan pakar dan sumber-sumber pengetahuan lainnya seperti yang telah disebutkan sebelumnya. Basis pengetahuan bersifat dinamis, bisa berkembang dari waktu ke waktu.

Perkembangan ini disebabkan karena pengetahuan selalu bertambah, terupdate (Hartati dan Iswanti, 2008: 5).

c) Struktur kontrol (Mesin Inferensi)

Struktur kontrol merupakan interpreter kaidah atau mesin inferensi yang menggunakan pengetahuan-pengetahuan yang tersimpan dalam basis pengetahuan untuk memecahkan atau menyelesaikan permasalahan yang ada.



Gambar 2.3 Struktur Sistem Pakar Kaidah Produksi
(Sumber:Hartati and Iswanti, 2008:10)

Sebelum sampai pada bentuk kaidah produksi, pengetahuan yang berhasil didapatkan dari domain tertentu disajikan dalam bentuk tabel keputusan kemudian dibuat pohon keputusannya.

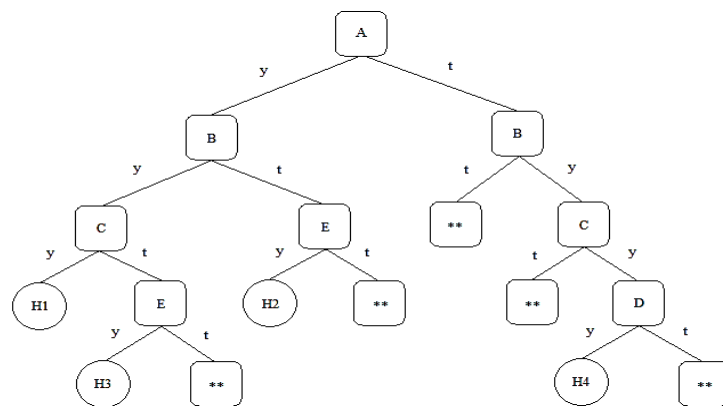
Berikut ini adalah contoh penyajian dalam bentuk tabel keputusan dan pohon keputusan (Hartati dan Iswanti, 2008: 26-39).

Tabel 2.1 Keputusan

Hipotesa <i>Evidence</i>	Hipotesa 1	Hipotesa 2	Hipotesa 3	Hipotesa 4
<i>Evidence A</i>	Ya	ya	ya	Tidak
<i>Evidence B</i>	Ya	tidak	ya	Ya
<i>Evidence C</i>	Ya	tidak	tidak	Ya
<i>Evidence D</i>	Tidak	tidak	tidak	Ya
<i>Evidence E</i>	Tidak	Ya	ya	Tidak

(Sumber:Hartati and Iswanti, 2008:32)

Mengacu pada tabel keputusan pada tabel 2.1 dapat dihasilkan pohon keputusan sebagai berikut:



Gambar 2.4 Pohon Keputusan
(Sumber:Hartati and Iswanti, 2008:33)

Berdasarkan pohon keputusan Gambar 2.4 dapat dijelaskan bahwa:

A = *evidence A*, H1 = hipotesa 1, y = ya

B = *evidence B*, H2 = hipotesa 2, t = tidak

C = *evidence C*, H3 = hipotesa 3, ** = tidak menghasilkan hipotesa tertentu

D = *evidence* D, H4 = hipotesa 4

Dari gambar diatas diketahui bahwa hipotesa H1 terpenuhi jika memenuhi evidence A, B, dan C. Hipotesa H2 terpenuhi jika memiliki evidence A dan evidence E. Hipotesa H3 akan terpenuhi jika memiliki evidence A, B, dan E. Hipotesa H4 akan dihasilkan jika memenuhi evidence B, C, dan D. Notasi “y” mengandung arti memenuhi node (evidence) di atasnya, notasi “t” artinya tidak memenuhi.

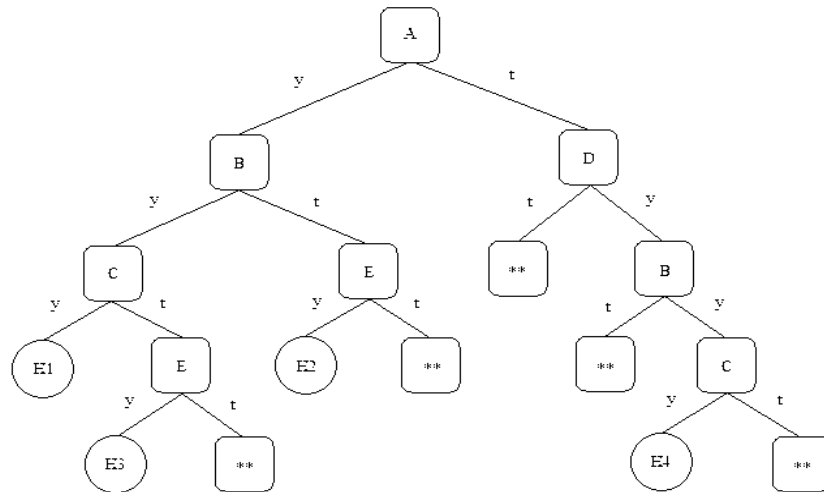
Dalam sesi konsultasi pada sistem pakar, node-node yang mewakili evidence biasanya akan menjadi pertanyaan yang diajukan oleh sistem. Dengan melihat pohon keputusan pada **gambar 2.2** permasalahan dapat saja terjadi pada awal sesi konsultasi yaitu pada saat sistem pakar menanyakan “apakah memiliki evidence A?”. Permasalahannya adalah apapun jawaban pengguna baik “ya” atau “tidak” maka sistem akan menanyakan evidence B. Ini berarti jawaban pengguna tidak akan mempengaruhi sistem. Salah satu cara untuk mengatasi hal ini adalah dengan mengubah urutan pada tabel keputusan seperti terlihat pada **tabel 2.2**.

Tabel 2.2 Alternatif Tabel Keputusan

Hipotesa <i>Evidence</i>	Hipotesa 1	Hipotesa 2	Hipotesa 3	Hipotesa 4
<i>Evidence A</i>	Ya	Ya	ya	Tidak
<i>Evidence D</i>	Tidak	Tidak	tidak	Ya
<i>Evidence B</i>	Ya	Tidak	ya	Ya
<i>Evidence C</i>	Ya	Tidak	Tidak	Ya
<i>Evidence E</i>	Tidak	Ya	Ya	Tidak

(Sumber:Hartati and Iswanti, 2008:34)

Berdasarkan tabel 2.2 dapat dihasilkan pohon keputusan sebagai berikut:



Gambar 2.5 Alternatif Pohon Keputusan
(Sumber:Hartati and Iswanti, 2008:35)

Berdasarkan alternatif pohon keputusan gambar 2.5 dapat dijelaskan bahwa:

A = *evidence* A, H1 = hipotesa 1, y = ya

B = *evidence* B, H2 = hipotesa 2, t = tidak

C = *evidence* C, H3 = hipotesa 3, ** = tidak menghasilkan hipotesa tertentu

D = *evidence* D, H4 = hipotesa 4

Dilihat dari gambar 2.5, masing-masing *node* yang mewakili *evidence* tertentu untuk kondisi “y” dan “t” sudah tidak mengarah pada *evidence* yang sama. Hal ini berarti jawaban pengguna yang berbeda akan mengarah pada pertanyaan yang berbeda pula.

Kaidah yang dapat dihasilkan berdasarkan pohon keputusan pada **gambar 2.3** adalah sebagai berikut:

1. Kaidah 1: *IF A AND B AND C THEN H1*
2. Kaidah 2: *IF A AND B AND E THEN H3*

3. Kaidah 3: *IF A AND E THEN H2*

4. Kaidah 4: *IF D AND B AND C THEN H4*

Model representasi pengetahuan kaidah produksi banyak digunakan pada aplikasi sistem pakar karena model representasi ini mudah dipahami dan bersifat deklaratif sesuai dengan jalan pikiran manusia dalam menyelesaikan suatu masalah, dan mudah diinterpretasikan. Menurut penelitian (Engin *et al.*, 2014) ada banyak keuntungan yang didapat dari sistem pakar yaitu:

- a) Mereka menurunkan biaya karena mengurangi kebutuhan akan tenaga ahli manusia
 - b) mereka bersifat permanen
 - c) mereka dapat digunakan untuk sistem pengetahuan yang berbeda, yang meningkatkan fungsionalitas
 - d) Mereka meningkatkan kehandalan karena meminimalkan kesalahan yang cenderung ditanggung manusia
 - e) dan jika dirancang oleh banyak ahli, bisa meningkatkan kepercayaan diri.
- Akhirnya, mereka kekurangan emosi manusia, yang merupakan sumber kesalahan dalam sistem berbasis manusia.

2.1.3 Basis Data (*Data Base*)

Database adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat yang sering digabungkan dengan algoritma program.

Sistem *database* adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data atau informasi yang sudah diolah dan membuat informasi

tersedia saat dibutuhkan. Kebutuhan basis data meliputi memasukkan, menyimpan, dan mengambil data serta membuat laporan berdasarkan data yang telah disimpan. Dan Salah satu bentuk basis data yang dibutuhkan dalam sebuah sistem yaitu *Database Management System (DBMS)* merupakan suatu sistem aplikasi yang digunakan untuk menyimpan mengelola, dan menampilkan data (A.S and M. Shalahuddin, 2011; 43-45).

Syarat suatu sistem aplikasi di sebut DBMS jika memenuhi persyaratan minimal sebagai berikut:

- a) Menyediakan fasilitas untuk mengelola akses data
- b) Mampu menangani integritas data
- c) Mampu menangani akses data yang dilakukan
- d) Mampu menangani backup data

2.1.4 Validasi Sistem

Validasi mengacu pada sekumpulan aktifitas yang berbeda yang menjamin bahwa sistem atau perangkat lunak yang dibangun telah sesuai dengan yang diharapkan(A.S and M. Shalahuddin, 2016; 273).

Beberapa pendekatan dalam melakukan pengujian untuk validasi sistem antara lain(A.S and M. Shalahuddin, 2016; 275-276).

1) Black-Box Testing (pengujian kotak hitam)

Pendekatan ini dilakukan dengan menguji sistem atau perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Tujuannya

untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari sistem atau perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan menggunakan sistem atau perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan *black-box testing* harus dibuat dengan kasus benar dan kasus salah.

2) *White-Box Testing* (pengujian kotak putih)

Pendekatan ini dilakukan dengan menguji sistem atau perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi yang dibutuhkan. *White-box testing* dilakukan dengan memeriksa logika dari kode program. Pembuatan kasus uji dapat mengikuti standar pengujian dari standar pemrograman yang ada.

2.2 Variabel Penelitian

Variabel penelitian adalah suatu atribut atau sifat atau nilai dari orang, objek atau kegiatan yang mempunyai variasi tertentu yang ditetapkan oleh peneliti untuk dipelajari dan kemudian ditarik kesimpulannya (Sugiyono, 2014; 38).

Oleh karena itu didalam penelitian ini, Objek yang digunakan adalah tanaman buah naga. Untuk variabel yang ditetapkan dalam penelitian ini yaitu penyakit tanaman buah naga.

2.2.1 Tanaman Buah Naga

buah naga merupakan salah satu jenis tanaman yang memiliki rasa berbeda (khas) yang merupakan kombinasi antara rasa manis, asam, dan rasa gurih menyegarkan, karena itulah buah ini mempunyai daya tarik tersendiri. selain itu, buahnya pun mengandung vitamin C, vitamin B, protein, serta kaya serat dan mineral bahkan buahnya pun mengandung zat-zat berkhasiat sebagai obat. sehingga Buahnya sangat tepat disajikan dalam setiap acara sarapan maupun di sela-sela waktu (Kristanto, 2014: 3).

Berikut ini gambaran unsur pembentuk (*morfologi*) dari tanaman buah naga yaitu (Kristanto, 2014: 17-18):

1) Akar

Perakaran bersifat epirit, yaitu merambat dan menempel pada batang tanaman lain, sangat tahan dengan kekeringan, dan tidak tahan akan genangan yang cukup lama.

2) Batang dan cabang

Batang tanaman buah naga mengandung air dalam bentuk lendir dan berlapis lilin jika sudah dewasa, warnanya hijau kebiru-biruan atau ungu, batang tersebut berukuran panjang dan bentuknya siku atau segi tiga.

3) Bunga

Kuncup bunga yang sudah berukuran panjang sekitar 30 cm akan mulai mekar oleh sinar matahari dan perubahan oleh suhu yang agak tajam antara siang dan malam hari.

4) Buah

Buah berbentuk bulat panjang serta berkulit warna merah dan sangat tebal, letak buah pada umumnya mendekati ujung cabang atau batang.

5) Biji

Biji berbentuk bulat berukuran kecil dan berwarna hitam. Kulit biji sangat tipis, tetapi keras. Biji dapat digunakan untuk memperbanyak tanaman secara generatif.

Buah naga merupakan buah yang mempunyai nilai ekonomi tinggi dan prospek yang baik untuk dibudidayakan di Indonesia. Iklim tropis dengan intensitas cahaya matahari yang cukup baik memungkinkan buah naga dapat berbuah sepanjang tahun. Berdasarkan hasil penelitian, ternyata buah ini banyak memiliki khasiat seperti menguatkan fungsi ginjal, meningkatkan ketajaman mata, menstabilkan kadar gula darah, menguraikan kolesterol, keputihan dan sebagai anti oksida (Prapti, Kanthi Pangestuning, Ridwan Iskandar, 2015).

Klasifikasi buah naga ini sendiri yaitu buah naga termasuk dalam kelompok tanaman kaktus atau famili *Cactaceae* dan subfamili *Hylocereanea*. Dalam subfamili ini terdapat beberapa genus, sedangkan buah naga termasuk dalam Genus *Hylocereus*. Genus ini pun terdiri atas sekitar 16 spesies, dua di antaranya memiliki buah yang komersial, *Hylocereus undatus* (berdaging putih) dan *Hylocereus Costaricensis* (daging merah) (Kristanto, 2014: 13).

Tanaman buah naga berasal dari Meksiko, Amerika Tengah, dan Amerika Selatan bagian utara. Di habitat aslinya, tanaman ini hidup di lingkungan hutan belantara, konon dinamakan buah naga karena batangnya yang tumbuh memanjang seperti naga. Orang Vietnam yang menganut budaya Cina lebih mengenalnya dengan nama *thang loy*, sedangkan di Amerika Selatan, buah naga disebut *pitaya roja* (pitaya merah). Dalam perkembangannya, tanaman buah naga ini juga menyebar ke Israel, Thailand, dan Australia. Sedangkan di Indonesia tanaman buah naga masuk sekitar tahun 2000-an, dan Joko Raino Sigit merupakan orang yang pertama kali membudidayakan tanaman buah naga ini, ia mendatangkan 250 bibit dari Thailand (Samadi, 2013:1).

Dalam budidaya buah naga banyak kendala yang sering terjadi seperti gangguan penyakit, gangguan ini dapat menyerang tanaman sejak mulai pembibitan sampai tanaman berproduksi (Samadi, 2013:53).

Ada beberapa macam penyakit tanaman buah naga yang diteliti yang diambil dari data penelitian 2017 antara lain:

1) Busuk Pangkal Batang



Gambar 2.6 Penyakit Busuk Pangkal Batang
(Sumber: data penelitian 2018)

Pembusukan tersebut umumnya disebabkan oleh kelembaban tanah yang terlalu tinggi sehingga merangsang pertumbuhan jamur *sclerotium rolfsii* sacc. Tanaman buah naga yang terserang penyakit ini menunjukkan pembusukan pada pangkal batang, terdapat bulu-bulu putih, batang tampak berair, dan batang berwarna kecoklatan. Pengendalian dari penyakit busuk pangkal batang ini adalah dengan menggunakan fungisida (semacam pestisida buat penyakit tanaman), menggunakan benlate 2 gr/liter air atau ridomil 2 gr/liter per dua minggu sekali selama satu bulan dengan cara disemprotkan atau bisa juga disiramkan ke pangkal batangnya.

2) Busuk Bakteri



Gambar 2.7 Penyakit Busuk Bakteri
(Sumber: data penelitian 2018)

Penyakit busuk bakteri disebabkan Terdapat genangan air disekitar tanaman buah naga yang dapat menimbulkan kelembaban. Tanaman buah naga yang terserang penyakit ini menunjukan gejala layu pada batang,

batang kusam berlendir, cabang berwarna putih kekuningan. Pengendalian dari penyakit busuk bakteri ini adalah dengan melakukan pemangkasan bagian batang atau cabang yang terserang busuk bakteri setelah itu lakukan sanitasi kebun secara teratur dan hindari genangan air pada saat curah hujan tinggi yang dapat menimbulkan kelembapan.

3) Fusarium



Gambar 2.7 Penyakit Fusarium
(Sumber: data penelitian 2018)

Fusarium adalah penyakit tanaman buah naga yang disebabkan Jamur *fusarium oxysporium schl*, penyakit ini sering ditemukan pada tanaman buah naga yang telah berproduksi maupun pada bibit buah naga. Terdapat beberapa gejala penyakit fusarium yaitu Layu pada batang, Cabang berkerut, Bercak bertitik hitam, dan Berwarna busuk coklat. Pengendalian dari penyakit fusarium ini adalah dengan menyemprot tanaman buah naga seminggu sekali dengan fungisida (semacam pestisida

buat penyakit tanaman), seperti benlate atau derosal 60 WP pada bagian tanaman yang diserang.

4) Antraknosa



Gambar 2.7 Penyakit Antraknosa
(Sumber: data penelitian 2018)

Penyakit antraknosa buah pada buah naga disebabkan Disebabkan oleh jamur *colltotrichum sp* yang berkembang pada saat pergantian musim. Pada tanaman buah naga yang menunjukkan gejala penyakit antraknosa buah ditandai dengan kulit buah berwarna coklat kehitaman, dan Bercak-bercak berbentuk bulat dan cekung. Pengendalian dari penyakit antraknosa buah ini adalah dengan mengikis bercak-bercak yang ada pada kulit buah dan menyemprotkan dengan cairan obat gusadrin dengan dosis 2cc/liter air.

5) Mosaik



Gambar 2.7 Penyakit Mosaik
(Sumber: data penelitian 2018)

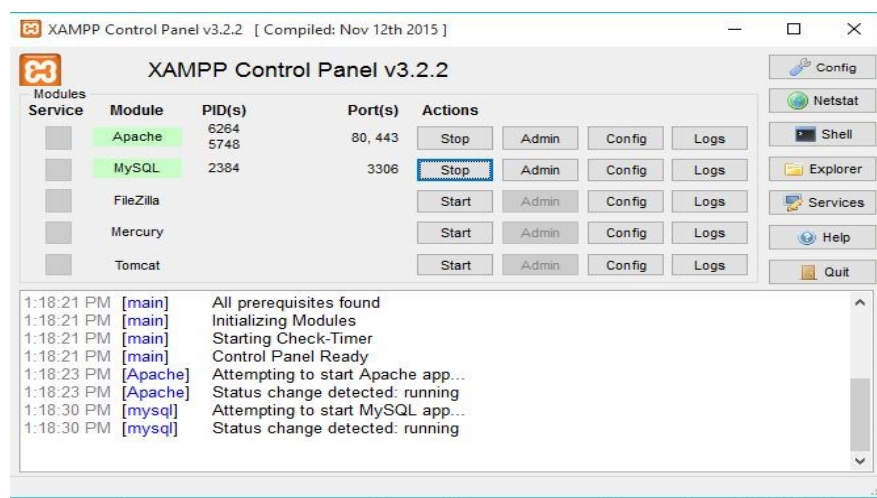
penyebab penyakit ini, disebabkan oleh virus karena keasaman tanah kurang dari pH 5 yang berarti kurangnya zat hara pada media tanah. Di batang tanaman buah naga dapat ditemukan gejala mosaik berwarna kuning, timbul bintik-bintik berwarna merah pada bagian tengah mosaik. Pada batang yang menunjukkan gejala, umumnya memiliki ukuran yang lebih pendek dari pada ukuran batang normal. Pengendalian dari penyakit mosaik ini adalah dengan mengatur keasaman tanah pada pH 7, bila pH tanah dibawah 5 maka pertumbuhan tanaman menjadi lambat bahkan kerdil.

2.3 Software Pendukung

Dalam pembuatan aplikasi sistem pakar, tentu adanya beberapa software pendukung yang merupakan beberapa perangkat lunak yang nantinya bisa digunakan untuk mendukung pembuatan aplikasi sistem pakar penelitian ini. Perangkat lunak tersebut antara lain yaitu.

2.3.1 XAMPP

Menurut (Winarno, Zaki and Community, 2014:1) XAMPP adalah software web server yang bisa digunakan untuk mengakomodasi sistem operasi, Apache (A), MySQL (M), PHP (P), Perl (P).



Gambar 2.6 Menu Konsep Penelitian
(Sumber: Winarno, Zaki and Community, 2014)

XAMPP memiliki banyak paket untuk berbagai sistem operasi yang ada di dunia, seperti Windows, Mac OS X atau Linux. XAMPP ditujukan untuk pekerjaan pengembangan program lokal saja, dan tidak disarankan untuk tahap produksi. Karena walaupun strukturnya sama seperti di computer server biasa, tapi kurang *secure*, sehingga anda mudah di *-hack* ketika menjalankan program di tahap produksi dengan menggunakan XAMPP.

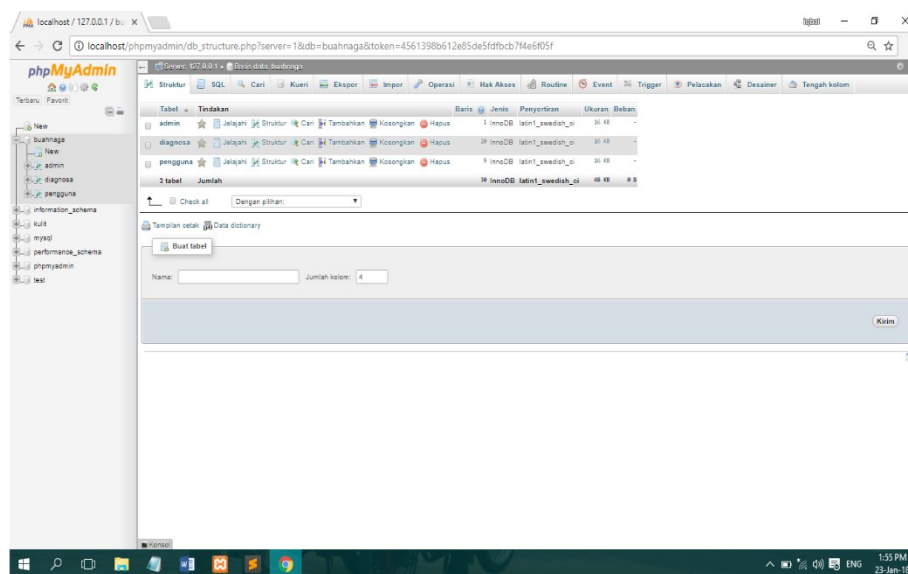
2.3.2 Web

Didalam penelitian ini, peneliti menerapkan sistem berbasis web karena web pada saat sekarang ini telah menjadi bagian dalam sebuah internet yang sangat populer dikalangan masyarakat dunia karena popularitasnya sebagai penyedia informasi dan interface yang di butuhkan oleh pengguna internet dari masalah informasi dan komunikasi. Web memudahkan pengguna komputer untuk berinteraksi dengan pelaku internet lainnya dalam menelusuri (informasi) di internet.

Menurut (Sidik and Pohan, 2014: 1) WWW (World Wide Web) atau yang lebih dikenal dengan web merupakan salah satu layanan yang didapat oleh pengguna komputer yang terhubung dengan internet. Web pada awalnya adalah ruang informasi dalam internet, dengan menggunakan teknologi hypertext, pemakai dituntun untuk menemukan informasi dengan mengikuti link yang disediakan dalam dokumen web yang ditampilkan dalam browser web. Kini internet identik dengan web, karena kepopuleran web sebagai standar interface pada layanan- layanan yang ada di internet, dari awalnya sebagai media informasi kini digunakan juga untuk komunikasi dari email sampai dengan chatting, sampai dengan melakukan transaksi bisnis (Commerce). Kini, web seakan lebih populer dari pada email, walaupun secara statistik email masih merupakan aplikasi terbanyak yang di gunakan oleh pengguna internet. Web lebih populer bagi khalayak umum dan pemula, terutama untuk tujuan pencarian informasi dan melakukan komunikasi e-mail yang menggunakan web sebagai interfacenya.

2.3.2 Php MyAdmin

Php My Admin adalah perangkat lunak gratis yang ditulis di PHP, yang ditujukan untuk menangani administrasi MySQL melalui Web. phpMy admin mendukung berbagai operasi yang terdapat di MySQL. Operasi yang sering digunakan (mengelola database, tabel, kolom, relasi, indeks, pengguna, perizinan, dll)



Gambar 2.7 *Php My Admin*
(Sumber: Software Penelitian 2017)

dapat dilakukan melalui antarmuka pengguna, sementara Anda masih memiliki kemampuan untuk mengeksekusi pernyataan SQL secara langsung.

phpMyAdmin adalah proyek yang matang dengan basis kode yang stabil dan fleksibel, Proyek phpMyAdmin adalah anggota *Software Freedom Conservancy*. SFC adalah organisasi nirlaba yang membantu mempromosikan, memperbaiki, mengembangkan, dan membela proyek Perangkat Lunak Bebas, Libre, dan Open Source (www.phpmyadmin.net/).

2.3.3 HTML

HTML (*Hyper Text Markup Language*) bahasa pemrograman yang digunakan untuk membuat dokumen HTML atau yang dikenal sebagai web page, dan dokumen HTML ini telah didukung oleh banyak browser-browser ternama pada saat ini seperti google chrome, mozilla, internet explorer dll, sebagai *interface* dari sebuah website.

Menurut (Sidik and Pohan, 2014: 9) Dokumen HTML merupakan file teks murni yang dapat dibuat menggunakan editor teks apapun yang disajikan dalam browser web surfer seperti Mozilla, Google Chrome, Internet Explorer, dan sebagainya. Dokumen ini biasanya berisi informasi atau interface aplikasi di dalam internet. Elemen merupakan istilah bagi komponen-komponen dasar pembentuk dokumen HTML. Tag HTML digunakan untuk menandai berbagai elemen dalam suatu dokumen HTML. Elemen dasar HTML yang dibutuhkan untuk membuat suatu dokumen HTML ditandai dengan tag `<html>`, `<head>`, dan `<body>` berikut tag-tag pasangannya yaitu `</html>`, `</head>`, dan `</body>`. Setiap dokumen HTML harus mempunyai pola dasar sebagai berikut (Sidik dan Pohan, 2009: 10-12):

```
<html>
```

```
<head>
```

```
Berisi informasi tentang head dokumen HTML
```

```
</head>
```

```
<body>
```

Berisi informasi yang ditampilkan dalam browser web

`</body>`

`</html>`

Berdasarkan keterangan diatas dapat dijelaskan bahwa:

1. Setiap dokumen HTML harus diawali dengan menuliskan tag `<html>` dan tag `</html>` di akhir dokumen. Tag ini menandai elemen html yang berarti dokumen ini adalah dokumen HTML.
2. Elemen head ditandai dengan tag `<head>` dan tag `</head>`. Elemen ini berisi informasi tentang dokumen HTML. Minimal informasi yang dituliskan dalam elemen ini adalah judul dokumen yang ditandai dengan menggunakan tag `<title>` dan diakhiri dengan tag `</title>`.
3. Elemen body ditandai dengan tag `<body>` dan diakhiri dengan tag `</body>`. Elemen ini merupakan elemen terbesar di dalam dokumen HTML. Elemen ini mengandung isi dokumen yang akan ditampilkan pada browser yang meliputi paragraf, grafik, link, tabel, dan sebagainya.

2.3.4 PHP: Hypertext Preprocessor

PHP merupakan secara umum dikenal sebagai bahasa pemrograman script yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server web*. Karna itu PHP dikenal juga sebagai bahasa pemrograman *server side* (Sidik, 2014: 4-5).

Pada awalnya nama PHP sendiri adalah PHP/FI yaitu singkatan dari *Personal Home Page Form Interface* yang dibuat pertama kali oleh Rasmus

Lerdoff. PHP, awalnya merupakan program CGI yang di khususkan untuk menerima input melalui form yang ditampilkan dalam browser web. Software ini disebarakan dan dilisensikan sebagai perangkat lunak *open source*. Seiring dengan berjalanya waktu, kini PHP adalah kependekan dari PHP: *Hypertext Preprocessor* yang merupakan bahasa utama *scrip server-side* yang disisipkan pada HTML, yang dijalankan di server, dan juga bisa digunakan untuk membuat aplikasi desktop.

2.3.5 CSS (*Cascading Style Sheet*)

CSS Merupakan singkatan *Cascade Style Sheet*, CSS feature baru dari HTML yang berfungsi untuk pengaturan font, ukuran halaman webpage dll yang bersinergi dengan html. menurut (Sidik and Pohan, 2014: 132) Hal ini di perlukan setelah melihat perkembangan HTML menjadi kurang praktis karena web page terlalu banyak dibebani hal-hal yang berkaitan dengan faktor tampilan seperti font dan lain-lain.

Untuk itu jika dikumpulkan style tersebut dikelola secara terpisah maka manajemen page menjadi lebih mudah dan efisien. Pada prakteknya penggunaan CSS ini di dukung oleh browser- browser terpopuler pada internet. Terdapat tiga cara pendefinisian dalam menggunakan CSS, yaitu (Sidik dan Pohan, 2009: 134-137):

- 1) *Style sheet external*

Pada teknik ini, style sheet didefinisikan di luar dokumen HTML dan disimpan dalam file berekstensi css (*.css). Dalam pendefinisian external tidak perlu lagi menggunakan tag html diawal dan akhir dokumen.

2) *Style sheet internal*

Style sheet didefinisikan secara internal biasanya karena web page tertentu bersifat sangat unik sehingga membutuhkan definisi terpisah dibandingkan dengan web page lainnya.

3) *Inline style sheet*

Style sheet inline hanya bisa digunakan pada lokasi yang sangat spesifik dimana style sheet ditempatkan. Kekurangan dari teknik ini adalah dokumen menjadi lebih besar karena style didefinisikan satu per satu.

2.3.6 MySQL

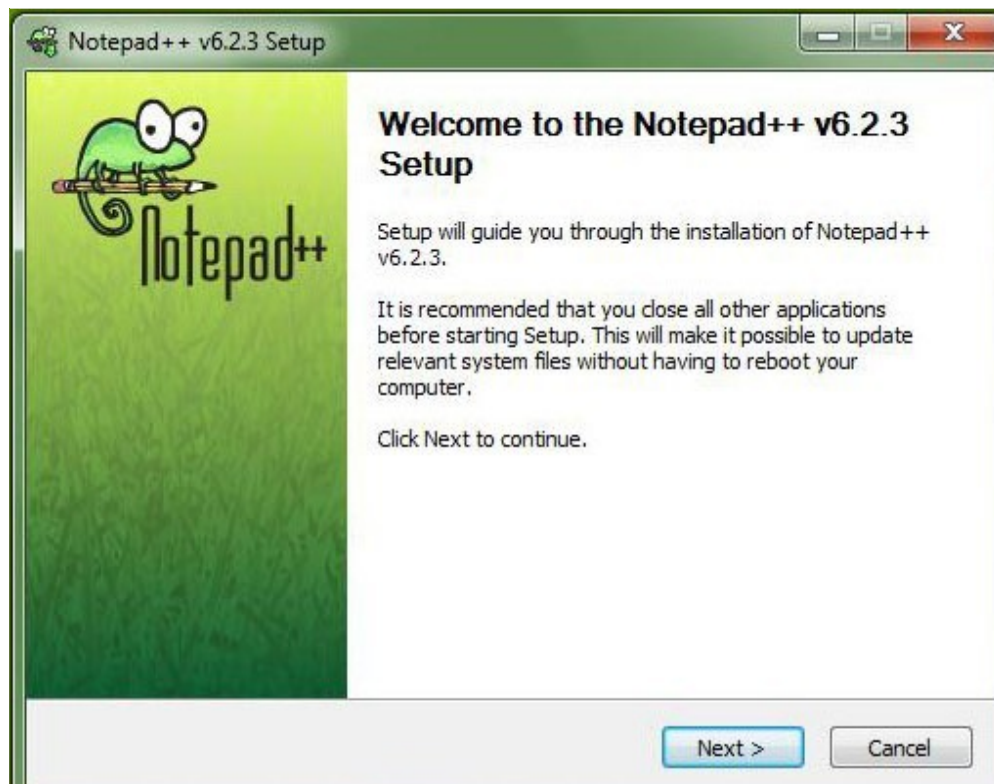
MySQL merupakan database yang sedang populer pada saat ini banyak pengguna memanfaatkan database ini karena gratis (*open souce*) karena itulah database cocok untuk pemakaian perorangan, apalagi *MySQL* memiliki sistem dengan database jaringan, sehingga setiap klien dapat berkomunikasi dengan server yang dijalankan secara lokal pada mesin pengguna atau dengan server yang dijalankan ditempat lain.

Sistem database *MySQL* menggunakan arsitektur klien-server yang memiliki kendali pusat di server, Server tersebut merupakan sebuah program yang dapat memanipulasi database, Program klien tidak melakukannya secara langsung, tetapi ia mengkomunikasikan tujuan pengguna kepada server dengan cara menuliskan

query dengan bahasa *SQL (Structured Query Language)*. Ketika program *MySQL* di gunakan secara interaktif, *MySQL* akan menampilkan sebuah prompt untuk menuliskan *query*, mengirim *query* itu ke server *MySQL* untuk di eksekusi, dan menampilkan hasilnya (Sianipar, 2015:1).

2.3.7 Notepad++

Saat ini begitu banyak editor- editor yang memudahkan seorang untuk melakukan *scripting kode (coding)* untuk pemrograman php maupun bahasa pemrograman lainnya, yaitu salah satunya notepad++, yang merupakan editor pemrograman yang banyak dipakai oleh programmer- programmer pada saat ini.

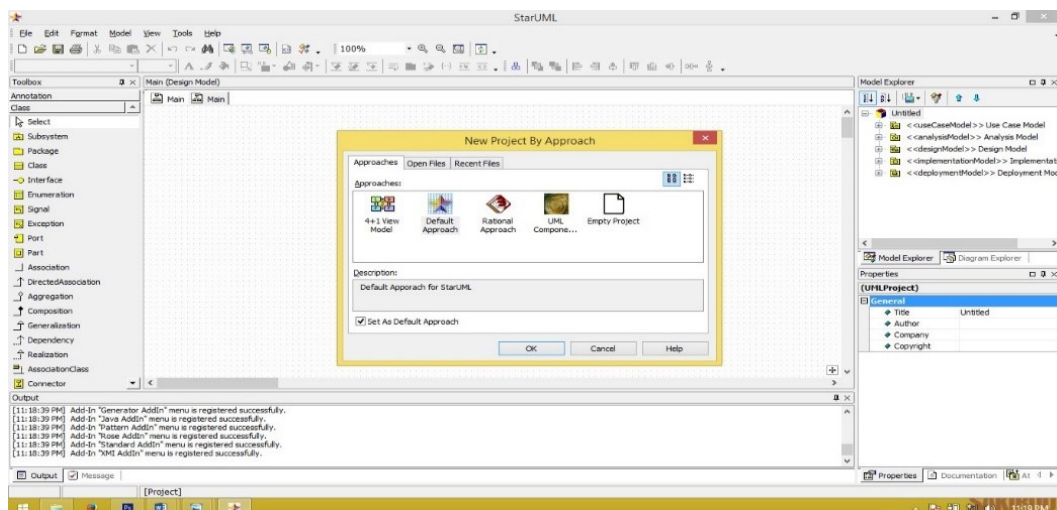


Gambar 2.8 Notepad++
(Sumber: EMS, 2014)

Menurut (EMS, 2014:54) salah satu editor text yang fiturnya sangat banyak dan cocok untuk pemrograman, selain itu editor text notepad++ tersedia dalam berbagai versi, yaitu versi installer dan versi portabel.

2.3.8 Star UML

Salah satu permodelan yang saat ini paling banyak digunakan adalah *UML*, dan perangkat lunak atau software yang digunakan untuk membuat permodelan yaitu *Star UML*, *Star UML* secara garis besar adalah perangkat lunak atau software untuk membuat diagram kelas *UML* secara otomatis untuk menghasilkan *UML* diagram yang sesuai dengan kebutuhan penggunaanya.



Gambar 2.8 Halaman Utama Software *Star UML*
(Sumber: A.S and M. Shalahuddin, 2016)

Menurut (A.S and M. Shalahuddin, 2016; 133-137) *UML (Unified Modeling Language)* adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefenisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa permodelan untuk pembangunan perngkat lunka yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language (UML)*. *UML* muncul karena adanya kebutuhan permodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

Menurut (A.S and M. Shalahuddin, 2016140-141) *UML* terbaru adalah *UML 2.3* yang terdiri dari 13 macam diagram yang dikelompokan dalam tiga kategori yaitu:

1) *Structure Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang di modelkan. Diagram *UML* yang termasuk dalam kategori ini antara lain *class diagram*, *object diagram*, *component diagram*, *composite structure diagram*, *package diagram*, dan *deployment diagram*.

2) *Behavior Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

Diagram *UML* yang termasuk dalam kategori ini antara lain *use case diagram*, *activity diagram*, dan *state machine diagram*.

3) *Interaction Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. Diagram *UML* yang termasuk dalam kategori ini antara lain *sequence diagram*, *communication diagram*, *timing diagram*, dan *interaction overview diagram*.

Menurut A.S. dan Shalahudin (2013: 18) *use case* dan *sequence diagram* merupakan bagian dari desain sistem. Dalam penelitian ini, diagram yang akan digunakan untuk desain sistem yaitu:

1) *Use case diagram*

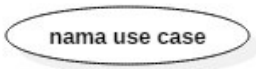
Use case diagram merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu sistem atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Ada 2 hal utama yang terdapat pada *use case* yaitu (A.S. dan Shalahuddin, 2013: 155):

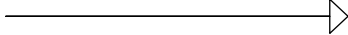
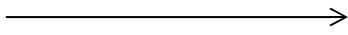
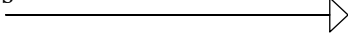
- a) Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.

- b) Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor.

Berikut ini adalah simbol-simbol yang digunakan dalam *use case diagram* (A.S. dan Shalahuddin, 2013: 156-160):

Tabel 2.3 Simbol *Use Case Diagram*

Simbol	Deskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor/<i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
asosiasi/<i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi/<i>extend</i> <<extend>> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu. Arah panah mengarah pada <i>use case</i> yang ditambahkan.

generalisasi/<i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara 2 buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari fungsi lainnya. Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)
Menggunakan/<i>include/uses</i> <<include>>  <<uses>> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankannya <i>use case</i> ini. Arah panah mengarah pada <i>use case</i> yang ditambahkan

(Sumber: A.S. dan Shalahuddin (2013: 156-160))

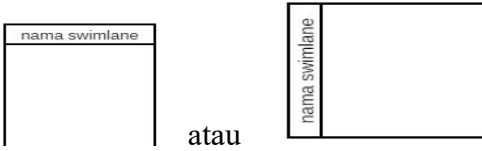
1) *Activity diagram*

Activity diagram merupakan diagram yang menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Jadi dapat dikatakan bahwa *activity diagram* menggambarkan aktifitas sistem, bukan apa yang dilakukan oleh aktor (A.S and M. Shalahuddin, 2013).

Simbol-simbol yang digunakan dalam *activity diagram* ditampilkan dalam tabel berikut:

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktifitas 	Aktifitas yang dilakukan sistem, aktifitas biasanya diawali dengan kata kerja

Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktifitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktifitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi

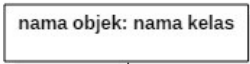
Sumber: A.S. dan Shalahuddin (2013: 162-163)

2) *Sequence diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki. Banyaknya *sequence diagram* yang harus digambar minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri, semakin banyak *use case* yang didefinisikan maka *sequence diagram* yang harus dibuat juga semakin banyak (A.S. dan Shalahuddin, 2013: 165).

Simbol-simbol yang digunakan dalam *sequence diagram* ditampilkan dalam tabel berikut:

Tabel 2.5 Simbol *Sequence Diagram*

Simbol	Deskripsi
Aktor/<i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
Garis hidup/<i>lifeline</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Objek 	Menyatakan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. Aktor tidak memiliki waktu aktif
Pesan tipe <i>create</i> <<create>> 	Menyatakan suatu objek membuat objek yang lain. Arah panah mengarah pada objek yang dibuat
pesan tipe <i>call</i> 1 : <u>nama_metode()</u> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi/metode.
Pesan tipe <i>send</i> 1 : <u>masukan</u> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya. Arah panah mengarah pada objek yang dituju
pesan tipe <i>return</i> 1 : <u>keluaran</u> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu. Arah panah mengarah pada objek penerima
Pesan tipe <i>destroy</i>	Menyatakan suatu objek mengakhiri hidup objek lain. Arah panah mengarah pada objek yang diakhiri



Sumber: A.S. dan Shalahuddin (2013: 165-167)

2.4 Penelitian Terdahulu

Untuk mendukung teori yang berkaitan dengan penelitian, peneliti mencantumkan beberapa penelitian terdahulu di bidang sistem pakar dalam kategori diagnosa.

1. Menurut (Prapti, Kanthi Pangestuning, Ridwan Iskandar, 2015), ISSN: 1411-5549, Vol.15 No.3 Hal. 94-98, **Strategi Peningkatan Kinerja Supply Chain Buah Naga Di Kecamatan Bangorejo Kabupaten Banyuwangi Berdasarkan Proses Inti *Scor***. Pembahasan: Buah naga merupakan buah yang mempunyai nilai ekonomi tinggi dan prospek yang baik untuk dibudidayakan di Indonesia. Iklim tropis dengan intensitas cahaya matahari yang cukup baik memungkinkan buah naga dapat berbuah sepanjang tahun. Berdasarkan hasil penelitian, ternyata buah ini banyak memiliki khasiat seperti menguatkan fungsi ginjal, meningkatkan ketajaman mata, menstabilkan kadar gula darah, menguraikan kolesterol, keputihan dan sebagai anti oksida.
2. Menurut (Sholikin and Eniyati, 2012), ISSN: 2085-3343, Vol. 04 No. 01, **sistem pakar diagnosa penyakit tanaman jambu merah delima berbasis**

web. Pembahasan: suatu sistem yang mempunyai kemampuan seorang pakar, mampu mendiagnosa jenis penyakit/hama dilihat dari gejala yang ada. yang menggunakan basis aturan (*rulebased reasoning*) dengan metode inferensi forward chaining. Sistem pakar berbasis web yang telah dikembangkan, dirancang dengan bahasa pemrograman dan database yang open source yaitu Php dan Mysql dan mempunyai keunggulan kemudahan akses dan kemudahan pemakaian.

3. Menurut (Wibowo, Widiastuti and Agustina, 2011), ISSN: Vol. 17, No. 2, 2011: 66–72, **Penyakit-Penyakit Penting buah naga di tiga Sentra Pertanian Di Jawa Tengah.** Pembahasan: penyakit yang umum terdapat pada buah naga tersebut adalah busuk batang yang disebabkan oleh *Erwinia* sp. dan kudis yang disebabkan oleh *Pestalotiopsis* sp. Adapun penyakit-penyakit lain yang dijumpai antara lain bercak coklat (*Fusarium* sp.), antraknosa (*Colletotrichum* sp.), mosaik yang kemungkinan disebabkan oleh *CactusVirus X*, puru akar (*Meloidogyne* sp.), serta busuk hitam dan bercak merah yang belum teridentifikasi penyebabnya.
4. Menurut (Sakdiah and Herwina, 2015), ISSN: 2303-2162, **Intensitas Serangan Semut pada Tanaman Buah Naga (*Hylocereus* sp.) di Kota Pariaman, Sumatera Barat.** Pembahasan: buah naga tidak luput dari serangan hama dan penyakit. Terdapat dua penyakit yang paling sering dijumpai yaitu busuk lunak batang dan antraknosa. Penyakit pada buah naga disebabkan oleh infeksi dimulai dari area luka (khususnya jaringan batang)

yang disebabkan oleh gigitan serangga, namun sampai saat ini belum teridentifikasi jenis serangganya.

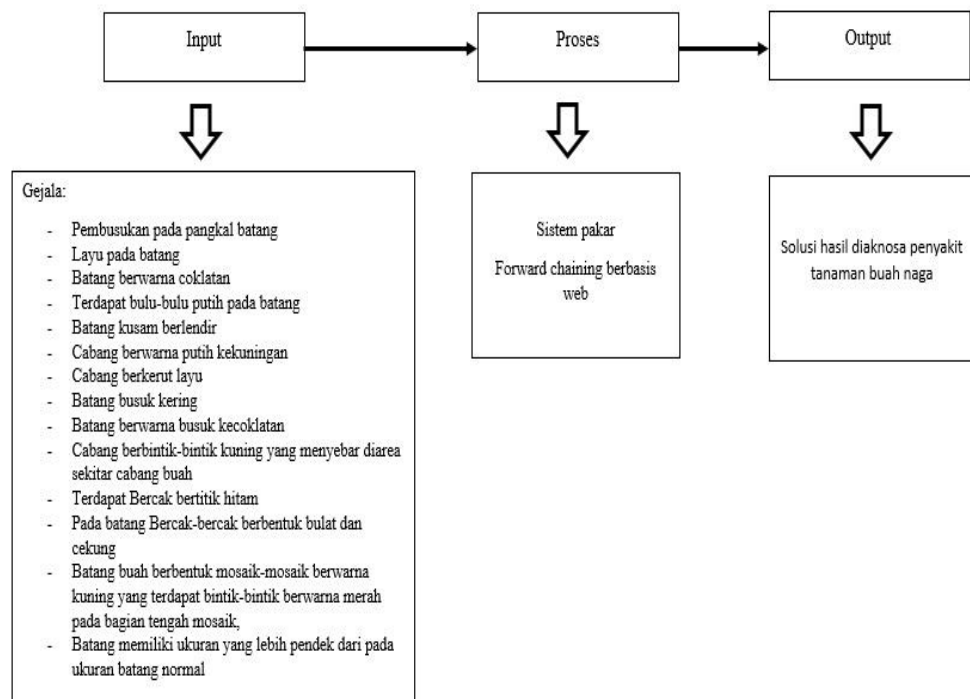
5. Menurut (Phongtongpasuk, Poadang and Yongvanich, 2016), ISSN: SCI 89 (2016) 239 – 247, ***Environmental-friendly method for synthesis of silver nanoparticles from dragon fruit peel extract and their antibacterial activities.*** Pembahasan: *Dragon fruit (Hylocereus undatus) or red pitaya is a tropical fruit which has a red skin with green fins on the fruit. It belongs to the cactus family, Cactaceae. The pulp of dragon fruit contains high quantity of vitamin C and watersoluble fiber. Usually, dragon fruit is consumed directly or being processed into juice for functional drink.*
6. Menurut (Engin et al., 2014), ISSN: SCI 31 2014) 22 – 31, ***Rule-based expert systems for supporting university students.*** Pembahasan: *there are many advantages expert systems: 1.) They decrease costs since they reduce the need for human experts, 2.) they are permanen, 3.) they can be used for different knowledge systems, which increases functionality 4.) they increase reliability since they minimize errors that humans are prone to, 5.) and if designed by multiple experts, can increase confidence Finally, they lack human emotions, which are sources of mistakes in human based systems.*

2.5 Kerangka Pemikiran

Kerangka pemikiran memuat pemikiran terhadap alur-alur yang dipahami sebagai acuan dalam pemecahan masalah yang diteliti secara logis dan sistematis antara hubungan variable yang diteliti. Kerangka berfikir yang baik akan

menjelaskan secara teoritis pertautan antar variabel yang diteliti (Sugiyono, 2014: 60).

Berikut ini adalah kerangka pemikiran yang mendasari penelitian ini.



Gambar 2.9 Kerangka Pemikiran
(Sumber: Data Penelitian: 2017)

Berikut ini adalah penjelasan dari kerangka penelitian yang ada pada gambar 2.9 di atas:

1. Input

Pada tahapan ini adalah tahapan pada lingkungan sistem pakar yang digunakan oleh *user* untuk menginput permasalahan yang akan dikonsultasikan.

2. Proses

Berikutnya dilanjutkan ke tahap proses, ditahap ini sistem pakar *forward chaining* berbasis web akan mengolah data masukan dari *user* dan menyesuaikan dengan fakta-fakta yang ada (*diagnosis*) di dalam sistem pakar.

3. *Output*

Dan selanjutnya adalah tahap *output* yang merupakan tahapan terakhir di dalam sistem pakar yang akan memberikan solusi dan hasil *diagnosis* sistem pakar