

## BAB II

### TINJAUAN PUSTAKA

#### A. Penelitian Terdahulu

Menurut Jasmir (2011) dalam jurnal yang ditulisnya dengan judul Rancangan Sistem Pakar dengan Metode Forward Chaining dan Heteroassocoative Memory Untuk Mendeteksi Tingkat Depresi Seseorang, Sistem pakar diagnosa ini nantinya dapat menghasilkan diagnosa tingkat depresi pada seseorang dan solusi yang perlu dilakukan. Dengan adanya aplikasi ini tentunya sangat membantu seseorang untuk mengenali gejala awal tingkat depresi yang di dideritanya sehingga dapat mengetahui tingkat depresi yang dialaminya dan dapat tersadar untuk merujuknya ke dokter jiwa. Untuk perbedaan dengan Aplikasi yang dibuat penulis adalah dari tambahan gejala dan juga metode yang digunakan untuk memicu rulle forward chaining yaitu penulis menggunakan pohon keputusan sedangkan penelitian terdahulu menggunakan Heteroassocoative Memory.

Sedangkan Menurut Giarratano dan Filey (2005) Sistem Pakar adalah salah satu cabang kecerdasan buatan yang menggunakan pengetahuan-pengetahuan khusus yang dimiliki oleh seorang ahli untuk menyelesaikan suatu masalah tertentu.

Sistem Pakar adalah hasil tranformasi pengetahuan dari seorang pakar kedalam sebuah perangkat atau system yang dapat digunakan untuk menyelesaikan masalah seperti yang biasa dilakukan para ahli (Pakar).

#### B. Aplikasi

Menurut Dhanta (2009 : 32) aplikasi (*Application*) adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, Microsoft Excel. Aplikasi berasal dari kata *application* yang artinya penerapan lamaran penggunaan.

Sutabri (2012 : 147) Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya.

Dari pengertian di atas, dapat disimpulkan bahwa aplikasi adalah alat yang dibuat untuk melaksanakan tugas tertentu sesuai dengan fungsi dan kemampuannya.

## **C. Sistem**

Menurut Ladjamudin (2005 : 6) Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya. Karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam system tersebut.

### **1. Karakteristik Sistem**

Menurut Ladjamuddin (2005 : 3) suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem penghubung, masukan, keluaran, pengolahan dan saran atau tujuan. Adapun Karakteristik yang dimaksud adalah :

#### **1. Komponen Sistem :**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan.

#### **2. Batasan Sistem :**

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya.

#### **3. Lingkungan luar Sistem :**

Lingkungan luar Sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

#### **4. Penghubung Sistem :**

Merupakan media yang menghubungkan antara satu subsistem yang lainnya.

#### **5. Masukan Sistem :**

Masukan sistem adalah energi yang dimasukkan ke dalam sistem, masukan dapat berupa masukan perawatan dan masukan sinyal maintenance.

6. Keluaran Sistem :

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan subsistem yang lain.

7. Pengelolaan Sistem :

Suatu bagian pengolah atau sistem itu sendiri sebagai pengelolannya. Pengelolaan yang akan berubah menjadi keluaran.

8. Sasaran Sistem :

Suatu sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada.

#### **D. Sekilas Tentang Sistem Pakar**

Sistem pakar merupakan program aplikasi dimana program tersebut menirukan proses penalaran dari seorang ahli dalam memecahkan masalah, dengan kata lain sistem pakar merupakan pengetahuan dari seorang ahli/pakar yang di transformasi kedalam sebuah aplikasi dan pengetahuan ahli tersebut disimpan didalam basis pengetahuan yang di gunakan saat proses pemecahan masalah

Menurut Arhami (2005) seorang pakar adalah orang yang mempunyai keahlian dalam bidang tertentu, yaitu pakar yang mempunyai Pengetahuan atau kemampuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dimilikinya. Ketika sistem pakar dikembangkan pertama kali sekitar tahun 70- an sistem pakar hanya berisi Pengetahuan yang eksklusif. Namun demikian sekarang ini istilah sistem pakar sudah digunakan untuk berbagai macam sistem yang menggunakan teknologi sistem pakar itu. Teknologi sistem pakar ini meliputi bahasa sistem pakar, program dan perangkat keras yang dirancang untuk membantu pengembangan dan pembuatan sistem pakar.

Pengetahuan dalam sistem pakar mungkin saja seorang ahli, atau pengetahuan yang umumnya terdapat dalam buku, majalah dan orang yang mempunyai pengetahuan tentang suatu bidang. Istilah sistem pakar, sistem berbasis pengetahuan, atau sering digunakan istilah sistem pakar *knowledge-base*, dengan

arti yang sama. Kebanyakan orang menggunakan istilah sistem pakar karena lebih singkat, bahkan walau belum benar- benar pakar, hanya menggunakan pengetahuan secara umum. (Arhami, 2005 : 3)

### **1. Manfaat Sistem Pakar**

Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang diberikannya, diantaranya adalah :

1. Meningkatkan produktivitas.
2. Membuat orang awam bekerja seperti layaknya seorang pakar.
3. Mampu menangkap pengetahuan dan kepakaran seseorang.
4. Memudahkan akses pengetahuan seorang pakar.
5. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
6. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
7. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
8. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar

### **2. Ciri- ciri sistem pakar**

Menurut Kusrini (2006), ciri- ciri sistem pakar adalah :

1. Terbatas pada pada bidang yang spesifik.
2. Dapat memberikan penalaran untuk data- data yang tidak lengkap atau tidak pasti.
3. Dapat mengemukakan rangkaian alasan- alasan yang diberikannya dengan cara yang dapat di pahami.
4. Berdasarkan pada *rulle* atau kaidah tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap.
6. *Output* bersifat nasehat atau anjuran.

7. *Output* tergantung dari dialog dengan user.
8. *Knowledge base* dan *interface engine* terpisah.

### **3. Keuntungan sistem pakar**

1. Menjadikan pengetahuan dan nasihat lebih mudah didapat.
2. Memberikan jawaban yang cepat.
3. Menyimpan kemampuan dan keahlian pakar.

### **4. Kelemahan sistem pakar**

1. Masalah dalam mendapatkan pengetahuan di mana pengetahuan tidak selalu bisa didapatkan dengan mudah, karena kadangkala pakar dari masalah yang kita buat tidak ada, dan walaupun ada kadang pendekatan yang dimiliki oleh pakar berbeda- beda.
2. Boleh jadi sistem tak dapat memuat keputusan.
3. Sistem pakar tidaklah 100% menguntungkan, walaupun seorang tetap tidak sempurna atau tidak selalu benar (Arhami, 2005 : 10).

### **5. Konsep Dasar Sistem Pakar**

Konsep dasar dari sistem pakar yaitu meliputi keahlian (*expertise*), ahli (*experts*), pemindahan kepakaran (*transferring expertise*), inferensi (*inferencing*), aturan (*rules*) dan kemampuan memberikan penjelasan (*explanation capability*).

Keahlian (*expertise*) adalah pengetahuan yang mendalam tentang suatu masalah tertentu, dimana keahlian bisa diperoleh dari pelatihan/ pendidikan, membaca dan pengalaman dunia nyata. Ada dua macam pengetahuan yaitu pengetahuan dari sumber yang ahli dan pengetahuan dari sumber yang tidak ahli. Pengetahuan dari sumber yang ahli dapat digunakan untuk mengambil keputusan dengan cepat dan tepat.

Ahli atau Pakar (*experts*) adalah seorang yang memiliki keahlian tentang suatu hal dalam tingkatan tertentu, ahli dapat menggunakan suatu permasalahan yang ditetapkan dengan beberapa cara yang berubah-ubah dan merubahnya

kedalam bentuk yang dapat dipergunakan oleh dirinya sendiri dengan cepat dan cara pemecahan yang mengesankan.

Pemindahan Kepakaran (*transferring expertise*) adalah tujuan dari sistem pakar yaitu memindahkan kepakaran dari seorang pakar ke dalam computer, kemudian ditransfer kepada orang lain yang bukan pakar. Proses ini melibatkan empat kegiatan, yaitu :

1. Akuisisi pengetahuan ( dari pakar atau sumber lain )
2. Representasi pengetahuan ( pada komputer )
3. Inferensi pengetahuan
4. Pemindahan pengetahuan ke pengguna

Inferensi (*inferencing*) adalah sebuah *procedure* ( program ) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi. Tugas mesin inferensi adalah mengambil kesimpulan berdasarkan basis pengetahuan yang dimilikinya.

Software system pakar komersial adalah system yang berbasis *rule* ( *rule-based systems*), yaitu pengetahuan disimpan terutama dalam bentuk *rule*, sebagai *procedure* – *procedure* pemecahan masalah.

Fasilitas yang lain dari system pakar adalah kemampuannya untuk menjelaskan saran atau rekomendasi yang diberikannya. Penjelasan dilakukan dalam subsistem yang disebut subsistem penjelasan (*explanation*). Bagian dari system ini memungkinkan system untuk memeriksa penalaran yang dibuat sendiri dengan menjelaskan operasi – operasinya.

Karakteristik dan kemampuan yang dimiliki oleh system pakar berbeda dengan system konvensional.

Tabel 2.1 : Perbandingan Sistem Konvensional dengan Sistem Pakar

Sistem Konvensional	Sistem Pakar
Informasi dan pemrosesannya biasanya menjadi satu dengan program	Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi

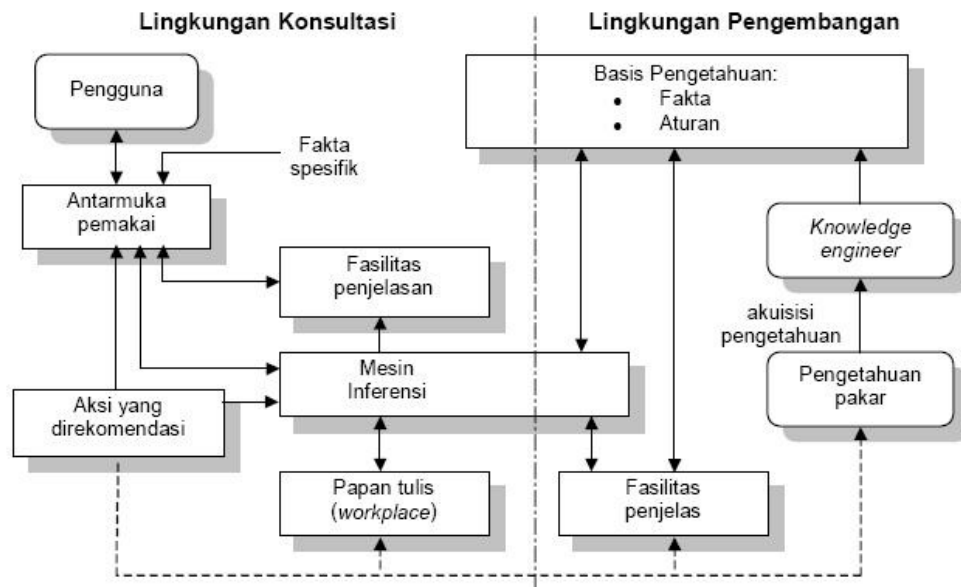
Biasanya tidak dapat menjelaskan mengapa suatu input data dibutuhkan atau bagaimana output itu diperoleh	Penjelasan adalah bagian terpenting dari sistem pakar
Pengubahan program cukup sulit dan merepotkan	Pengubahan pada aturan/kaidah dapat dilakukan dengan mudah
Eksekusi dilakukan langkah demi langkah secara algoritmik	Eksekusi dilakukan pada keseluruhan basis pengetahuan secara heuristic atau logic
Menggunakan data	Menggunakan pengetahuan
Program tidak pernah salah (kecuali pemrogramnya yang salah)	Program bisa saja melakukan kesalahan

---

Sumber : Konsep kecerdasan buatan (Desiani, A. dan Muhammad Arhami, 2006).

## 6. Struktur Sistem Pakar

Sistem pakar terdiri dari dua bagian pokok, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan sebagai pembangun sistem pakar baik dari segi pembangun komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seorang yang bukan ahli untuk berkonsultasi.



Sumber : <http://elib.unikom.c.id>

Gambar 2.1 : Struktur Sistem Pakar

Keterangan :

1. Akuisisi Pengetahuan

Akuisi pengetahuan adalah akumulasi, transfer dan transformasi keahlian dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh computer.

2. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Basis pengetahuan terdiri dari dua elemen dasar, yaitu :

- Fakta, misalnya situasi atau permasalahan yang ada
- Rule* ( Aturan ), untuk mengarahkan penggunaan pengetahuan dalam memecahkan masalah

3. Mesin Inferensi (*Inference Engine*)

Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace*, dan untuk memformulasikan kesimpulan.



4. Daerah Kerja (*Workplace*)

*Workplace* merupakan area dari sekumpulan memori kerja (*working memory*). *Workplace* digunakan untuk merekam hasil-hasil antara kesimpulan yang dicapai.

Terdapat 3 tipe keputusan yang dapat direkam pada *workplace* yaitu :

- a. Rencana : bagaimana menghadapi masalah
- b. Agenda : aksi-aksi potensial yang sedang menunggu untuk dieksekusi
- c. Solusi : calon aksi yang akan dibangkitkan

5. Antarmuka Pengguna (*User Interface*)

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

6. Fasilitas Penjelasan

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

7. Perbaikan Pengetahuan

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

8. Pengguna (*User*)

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

## 7. Rule Sebagai Teknik Representasi Pengetahuan

Setiap rule terdiri dari dua bagian, yaitu bagian IF disebut sebagai *evidence* (Fakta-fakta) dan bagian THEN disebut *Hipotesis* atau Kesimpulan.

Syntax *Rule* adalah : **IF E THEN H**

Secara umum *rule* yang mempunyai *evidence* lebih dari satu yang dihubungkan oleh kata penghubung **AND** atau **OR** atau kombinasi keduanya. Tetapi sebaiknya biasakan menghindari penggunaan **AND** dan **OR** secara sekaligus dalam satu *rule*.

**IF ( E1 AND E2 AND E3 ..... AND En) THEN H**

**IF ( E1 OR E2 AND E3 ..... OR En) THEN H**

Selain itu satu *evidence* bisa juga mempunyai *hipotesis* lebih dari satu. Seperti di bawah ini :

**IF E THEN (H1 AND H2 AND H3 ..... AND Hn)**

### E. Depresi

Menurut Nugroho (2012) Depresi adalah perasaan sedih, ketidakberdayaan dan pesimis yang berhubungan dengan suatu penderitaan. Dapat berupa serangan yang ditujukan kepada diri sendiri atau perasaan marah yang dalam.

Menurut Davison dkk (2006) Depresi merupakan kondisi emosional yang biasanya ditandai dengan kesedihan yang amat sangat mendalam, perasaan tidak berarti dan bersalah, menarik diri dari orang lain dan tidak dapat tidur, kehilangan selera makan, hasrat seksual dan minat serta kesenangan dalam aktivitas yang biasa dilakukan.

Menurut Kaplan & Sadock (2010) Depresi merupakan gangguan suasana hati atau mood yang dalam edisi DMS (Dignostic and Statistical Manual of Mental Disorders) yang dikenal sebagai gangguan afektif).

Menurut Hawari (2010) Depresi adalah salah satu bentuk gangguan kejiwaan pada alam perasaan (affective/mood disorder), yang ditandai dengan kemurungan, kelesuan, ketidak gairahan hidup, perasaan tidak berguna, dan putus asa.

<http://repository.usu.ac.id>

**Terdapat 3 tingkatan depresi yaitu :**

**1. Depresi Ringan**

Depresi ringan dialami oleh banyak orang pada umumnya, gejalanya pun tidak terlalu banyak dan tidak termasuk gejala berat seperti halusinasi dan bunuh diri. Biasanya hanya berlangsung 2 minggu. Orang yang mengalami depresi ringan masih bisa melakukan berbagai aktivitas seperti orang biasanya.

**2. Depresi Sedang**

Depresi sedang hampir mirip dengan depresi ringan, namun untuk gejalanya lebih banyak gejala sedang. Orang yang mengalami depresi sedang masih mampu untuk melakukan aktivitas namun sangat terbatas.

**3. Depresi Berat**

Orang yang mengalami depresi berat memiliki gejala lebih banyak dari gejala ringan dan gejala sedang, bahkan terkadang pasien sudah tidak dapat melakukan aktivitas sama sekali, tidak mau berkomunikasi dan juga tidak dapat merawat diri.

<http://repository.usu.ac.id>

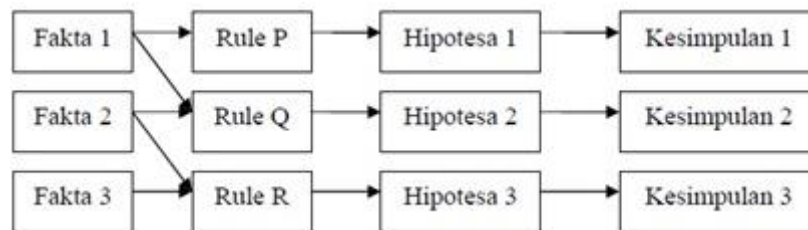
**F. Metode *Forward Chaining***

Runut maju (*forward chaining*) berarti menggunakan himpunan aturan kondisi – aksi. (Kusrini, 2006 : 36).

Menurut Russel P (2003) Metode forward chaining adalah metode pencarian atau teknik pelacakan kedepan yang dimulai dengan informasi yang ada dan penggabungan rule untuk menghasilkan suatu kesimpulan atau tujuan. Metode inferensi adalah mekanisme berfikir dan pola-pola penalaran yang digunakan oleh sistem untuk mencapai suatu kesimpulan. Metode ini akan menganalisa masalah tertentu dan selanjutnya akan mencari jawaban atau kesimpulan yang terbaik. Penalaran dimulai dengan mencocokkan kaidah-kaidah dalam basis pengetahuan dengan fakta-fakta yang ada dalam basis data.

*Forward chaining* merupakan metode inferensi yang melakukan penalaran dari suatu masalah kepada solusinya. Jika klausa premis sesuai dengan situasi

(bernilai TRUE), maka proses akan menyatakan konklusi. *Forward chaining* adalah *data-driven* karena inferensi dimulai dengan informasi yang tersedia dan baru konklusi diperoleh. Jika suatu aplikasi menghasilkan tree yang lebar dan tidak dalam, maka gunakan forward chaining.



Gambar 2.2: Penalaran metode forward chaining

Sumber: Artificial Intelligence: Teknik dan Aplikasinya (kusumadewi,Sri. 2003)

### G. *PHP (Profesional Home Page)*

Menurut Bimo (2003) *PHP* adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah – perintah yang di berikan akan sepenuhnya dijalankan diserver tetapi akan disertakan pada dokumen HTML. *PHP* menawarkan koneksitas yang baik dengan bebarapa basis data antara lain Oracle, Sybase, mSQL, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBASE.

#### **Keunggulan PHP**

- a. PHP mudah dibuat dan dijalankan, maksudnya PHP dapat berjalan dalam Web Server dan dalam Sistem Operasi yang berbeda pula.
- b. PHP adalah software open-source yang gratis dan bebas didistribusikan kembali di bawah lisensi GPL (GNU Public License). User dapat download kode-kode PHP tanpa harus mengeluarkan uang atau khawatir dituntut oleh pihak pencipta PHP.
- c. PHP bisa dioperasikan pada platform Linux ataupun Windows.
- d. PHP sangat efisien, karena PHP hanya memerlukan resource system yang sangat sedikit dibanding dengan bahasa pemrograman lain.

- e. Ada banyak Web Server yang mendukung PHP, seperti Apache, PWS, IIS, dan lain-lain.
- f. PHP juga didukung oleh banyak database, seperti MySQL, PostgreSQL, Interbase, SQL, dan lain-lain.
- g. Bahasa pemrograman PHP sintaknya sederhana, singkat dan mudah untuk dipahami.
- h. HTML-embedded, artinya PHP adalah bahasa yang dapat ditulis dengan menempelkan pada sintak-sintak HTML. <http://dir.unikom.ac.id>

## **H. MySQL**

*MySQL* adalah *multiuser database* yang menggunakan bahasa *Structured Query Language (SQL)*.

*SQL* merupakan bahasa standart yang digunakan untuk mengakses *database server*. Bahasa ini pada awalnya dikembangkan oleh IBM, namun telah di adopsi dan di gunakan sebagai standar industri. Dengan menggunakan *SQL*, proses akses database menjadi lebih *user-friendly* dibandingkan dengan menggunakan *dBASE* atau *Clipper* yang masih menggunakan perintah – perintah pemrograman. (Bimo, 2003 : 65)

### **Keunggulan MySQL**

Beberapa keunggulan dari MySQL yaitu :

- a. Cepat, handal dan Mudah dalam penggunaannya MySQL lebih cepat tiga sampai empat kali dari pada database server komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.
- b. Didukung oleh berbagai bahasa Database server MySQL dapat memberikan pesan error dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
- c. Mampu membuat tabel berukuran sangat besar 24 Ukuran maksimal dari setiap tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.

- d. Lebih Murah,MySQL bersifat open source dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan Windows platform
- e. Melekatnya integrasi PHP dengan MySQL Keterikatan antara PHP dengan MySQL yang sama-sama software open-source sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan database server lainnya. <http://dir.unikom.ac.id>

### **I. DFD (*Data Flow Diagram*)**

Menurut Ladjamudin (2005:64) dalam bukunya yang berjudul Analisis dan Desain Sistem Informasi *Data Flow Diagram* adalah diagram aliran data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil.

Menurut Jogiyanto (2005) dalam bukunya Basia Data ada beberapa simbol digunakan pada DFD untuk mewakili :

#### 1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

#### 2. Arus Data (*Data Flow*)

Arus Data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

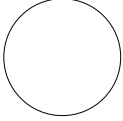



#### 3. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

#### 4. Simpanan Data (*Data Store*)

*Data Store* merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer.

Tabel 2.2 : Simbol-simbol DFD

No	Symbol	Nama	Keterangan
1		Proses	Menunjukkan kegiatan/ kerja yang dilakukan oleh orang, mesin atau komputer.
2		Simbol Data Flow ( arus data)	Menunjukkan arus dari proses.
3		Eksternal Entity	Menunjukkan entitas/entity
4		Data Store	Simpanan data

Sumber : Analisis dan Desain Sistem Informasi (Ladjamudin, 2005)


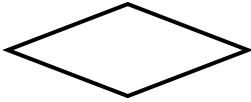


#### J. ERD (*Entity Relationship Diagram*)

Menurut Ladjamudin (2005 : 84) *Entity Relationship Diagram* (ERD) adalah diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam system bisnis.

Simbol – simbol dari *Entity Relationship Diagram* adalah sebagai berikut :

1. Persegi Panjang : Menyatakan himpunan / entitas
2. Lingkaran / elips : Menyatakan atribut (atribut yang berfungsi key digaris bawah).
3. Belah Ketupat : Menyatakan himpunan relasi
4. Garis : Sebagai Penghubung antar himpunan relasi dengan Himpunan entitas dan himpunan entitas dengan atributnya.

Tabel 2.3 : Simbol-simbol *Entity Relationship Diagram*

No	Symbol	Nama	Keterangan
1		<i>Entity / Entitas</i>	Suatu kumpulan objek atau sesuatu yang dapat dibedakan atau dapat didefinisikan
2		<i>Relationship</i>	Hubungan yang dapat terjadi antara satu entitas atau lebih
3		Atribut	Karakteristik dari entitas atau Relationship yang menyediakan penjelasan detail entitas atau relation
4		<i>Link</i>	Baris sebagai penghubung antara himpunan, relasi dan himpunan entitas dan atributnya

Sumber : Analisis dan Desain Sistem Informasi (Ladjamudin, 2005)

### K. *Flowchart*

Menurut Ladjamudin (2006:265) dalam bukunya *Rekayasa Perangkat Lunak* Flowchart adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah penyelesaian suatu masalah. Flowchart cara penyajian suatu algoritma. Ada dua macam flowchart yang menggambarkan proses dengan computer, yaitu :

1. Sistem *Flowchart*

Bagan yang memperlihatkan urutan proses dalam system dengan menunjukkan alat media input, output serta jenis media penyimpanan dalam proses pengolahan data.

2. Program *Flowchart*

Bagan yang memperlihatkan urutan instruksi yang digambarkan dengan symbol tertentu untuk memecahkan masalah dalam suatu program.



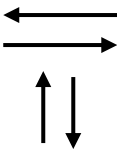

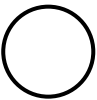
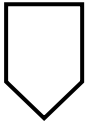
**a. Simbol – simbol *Flowchart***

Flowchart disusun dengan symbol. Symbol ini dipakai sebagai alat bantu menggambarkan proses di dalam halaman. Symbol – symbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok yaitu :

1. *Flow Direction Symbols* (Simbol Penghubung/alur)

Simbol yang digunakan untuk menghubungkan antara symbol yang satu dengan yang lainnya. Simbol ini juga disebut *connecting line*, Simbol tersebut adalah :

Tabel 2.4 : *Flow Direction Symbols*



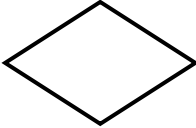

No	Symbol	Nama	Keterangan
1		<i>Arus / Flow</i>	Untuk menyatakan jarkannya arus suatu proses
2		<i>Communication link</i>	Untuk menyatakan bahwa adanya transisi suatu data atau nformasi dari suatu lokasi ke lokasi lainnya
3		<i>Connector</i>	Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembaran sama
4		<i>Offline Connector</i>	Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembaran yang berbeda


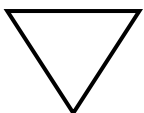

Sumber : Analisis dan Desain Sistem Informasi (Ladjamudin, 2005)

2. *Processing Symbols* (Simbol Proses)

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur. Symbol – symbol tersebut adalah :

Tabel 2.5 : *Processing Symbols*

No	Symbol	Nama	Keterangan
1		Proses	Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi
2		Symbol manual	Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer (manual)
3		<i>Decision / Logika</i>	Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan 2 kemungkinan jawaban, YA / TIDAK
4		<i>Predefined Process</i>	Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal

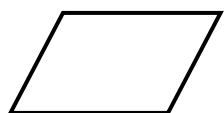
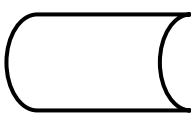
5		Terminal	Untuk menyatakan permulaan atau akhir suatu program
6		Offline Storage	Untuk menunjukkan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu
7		Manual Input	Untuk memasukkan data secara manual dengan menggunakan online keyword


Sumber : Analisis dan Desain Sistem Informasi (Ladjamudin, 2005)

### 3. *Input / Output Symbols* (Simbol Input – output)

Symbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Symbol – symbol tersebut adalah :

Tabel 2.6 : *Input – Output Symbols*

No	Symbol	Nama	Keterangan
1		<i>Input / output</i>	Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
2		<i>Disk Storage</i>	Untuk menyatakan input berasal dari disk atau optput disimpan ke disk

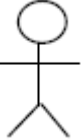
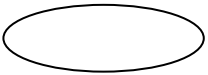

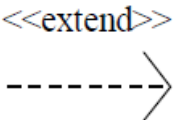
3		Document	Untuk dokumen	menyetak
---	---	----------	------------------	----------


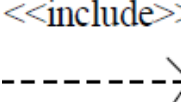
Sumber : Analisis dan Desain Sistem Informasi (Ladjamudin, 2005)

#### L. Use case Diagram

Menurut Shalahuddin (2013:155) *use case* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem. Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut :

Tabel 2.7 : Simbol *Use case* Diagram

No	Simbol	Nama	Deskripsi
1		<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.
2		<i>Use case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
3		<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.
4		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

5		<i>Generalization</i>	<p>Hubungan generalisasi dan Spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p> <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan</p>
6		<i>Include</i>	<p><i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p>

Sumber : Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek (Rosa dan Shalahuddin, 2013)

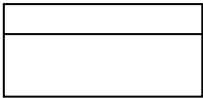


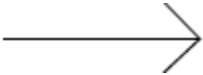



Komponen pembentuk diagram *use case* adalah:

- i. Aktor (*actor*), menggambarkan pihak-pihak yang berperan dalam sistem.
- ii. *Use case*, aktivitas / sarana yang disiapkan oleh bisnis / sistem.
- iii. Hubungan (*link*), aktor mana saja yang terlibat dalam *use case* ini.

### M. *Class Diagram*

Shalahuddin (2013:141) menjelaskan tentang *class diagram* yaitu menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Adapun simbol-simbol yang digunakan dalam *class diagram* adalah sebagai berikut :

Tabel 2.8 : Simbol *Class* Diagram

No	Simbol	Nama	Deskripsi
1		<i>Class</i>	Kelas pada stuktur sistem
2		<i>Interface</i>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3		<i>Association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4		<i>Directed association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
5		<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
6		<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.
7		<i>Aggregation</i>	Relasi antar kelas dengan makna semua bagian ( <i>whole-part</i> ).

Sumber : Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek (Rosa dan Shalahuddin, 2013)