

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

PT. PRIMA TETAP JAYA merupakan sebuah perusahaan yang bergerak dibidang supplier, leveransir dan jasa pembuatan mesin mesin pabrik minyak kepala sawit. Perusahaan ini sudah sangat di percayai oleh banyak perusahaan lainnya karena hasil kerja yang memuaskan dan tidak mengecewakan pelanggan. PT. PRIMA TETAP JAYA selalu berusaha mencegah terjadinya kesalahan agar tidak merugikan pelanggan, tetapi masih adanya permasalahan yang peneliti temukan di PT. PRIMA TETAP JAYA yaitu pihak perusahaan sering merasa dirugikan karna tidak semua karyawan bisa memperbaiki mesin bubut, lambatnya karyawan dalam menentukan kerusakan pada mesin bubut, lambatnya pengerjaan proyek yang dikarenakan mesin tidak bekerja optimal, dan masih menggunakan cara mendiagnosa semua satu persatu kerusakan mesin yang menghabiskan banyak waktu untuk menemukan masalah memperbaiki mesin bubut, selain permasalahan pada lambatnya karyawan dalam mendiagnosa terdapat kerusakan yang sering terjadi pada mesin bubut diantaranya mesin tidak dapat bekerja dengan baik, beban motor yang berlebihan, kepala tetap, eretan, kepala lepas dan kerusakan lainnya. Oleh karena itu pihak perusahaan harus bijaksana menghadapi masalah kerusakan mesin bubut seketika mesin rusak, karena kerusakan pada mesin bubut menjadi terhentinya proses kerja, sehingga dibutuhkan solusi untuk memecahkan masalah dengan merancang sebuah sistem pakar diagnosa kerusakan mesin bubut, yang bertujuan membangun sebuah sistem pakar mendiagnosa kerusakan mesin bubut dengan menggunakan metode *Dempster Shafer*

Metode *Dempster Shafer* dapat digunakan untuk mencari persentase kemungkinan kerusakan pada mesin bubut dengan mendiagnosa gejala yang dirasakan. Diharapkan dengan penggunaan metode ini dapat meminimalisir ketidakpastian sehingga dapat menghasilkan diagnosa yang valid. Metode *Dempster Shafer* sesuai untuk diimplementasikan dalam sebuah sistem pakar yang memiliki kemampuan untuk mendiagnosis suatu kerusakan.

Sistem pakar (expert system) adalah cabang dari kecerdasan buatan dan juga merupakan bidang ilmu yang muncul seiring perkembangan ilmu komputer saat ini. Sistem ini adalah sistem komputer yang bisa meniru kemampuan seorang pakar, sistem ini bekerja untuk mengadopsi pengetahuan manusia ke komputer yang menggabungkan dasar pengetahuan (knowledge base) menggunakan sistem inferensi untuk menggantikan fungsi seorang pakar dalam menyelesaikan suatu masalah.

Berdasarkan deskripsi masalah yang dibahas diatas maka dibangunlah Sistem Pakar yang mengadopsi metode *Dempster Shafer* dalam pemecahan masalah kerusakan pada mesin bubut, sistem ini diharapkan dapat membantu pihak industri untuk lebih mudah dalam melakukan diagnosa kerusakan pada mesin bubut. Berdasarkan latar belakang yang telah peneliti paparkan, maka peneliti mengangkat judul skripsi **“E-DIAGNOSIS KERUSAKAN MESIN BUBUT DENGAN METODE DEMPSTER SHAFER PADA PT . PRIMA TETAP JAYA”**.

## 1.2 Rumusan Masalah

Permasalahan permasalahan yang akan di pecahkan dalam penelitian ini adalah sebagai berikut :

1. Bagaimana merancang sebuah sistem pakar untuk mendiagnosa kerusakan pada mesin bubut ?
2. Bagaimana menerapkan metode *Dempster Shafer* dalam proses mendiagnosa kerusakan pada mesin bubut ?
3. Bagaimana membuat laporan hasil diagnosa kerusakan pada mesin bubut pada PT. PRIMA TETAP JAYA ?

## 1.3 Batasan Masalah

Berdasarkan uraian diatas maka batasan masalahnya adalah sebagai berikut:

1. Aplikasi yang dibangun hanya untuk mendiagnosa kerusakan pada mesin bubut.
2. Nilai pengujian dari hasil perhitungan berdasarkan rumus *Dempster Shafer* yang akan berakhir pada suatu solusi ( Kerusakan yang dialami pada mesin bubut).
3. Aplikasi dibangun dengan menggunakan bahasa pemrograman PHP dan database MySQL

## 1.4 Tujuan Penelitian

Adapun yang menjadi tujuan penelitian dalam membangun aplikasi sistem pakar ini adalah sebagai berikut :

1. Membangun sistem yang baru berbasis web yang digunakan untuk mendiagnosa kerusakan mesin bubut.
2. Membuat aplikasi sistem pakar mengenai kerusakan mesin bubut dengan metode *Dempster Shafer*.
3. Membangun sistem yang berbasis web untuk menampilkan laporan mengenai hasil diagnosa kerusakan mesin bubut.

### **1.5 Manfaat Penelitian**

Adapun yang menjadi manfaat penelitian dalam membangun aplikasi sistem pakar ini adalah sebagai berikut :

1. Dapat bermanfaat membantu para karyawan dalam mendiagnosa kerusakan mesin bubut sehingga lebih efektif dan efisien.
2. Dapat bermanfaat bagi perusahaan dalam memecahkan masalah ketidakpastian, sehingga membantu dalam mengambil keputusan dalam mendiagnosa kerusakan mesin bubut.
3. Dapat memberikan kemudahan bagi perusahaan dalam mengetahui laporan hasil diagnosa sehingga mempercepat waktu dalam pengolahan laporan diagnosa masalah mesin bubut.
4. Dapat bermanfaat bagi masyarakat dalam mendiagnosa kerusakan mesin bubut tanpa harus membuang waktu.

### **1.6 Metodologi Penelitian**

Langkah langkah yang dilakukan dalam penelitian ini adalah sebagai berikut :

1. Studi Kepustakaan

Pada tahap ini dilakukan studi kepustakaan yaitu proses mengumpulkan informasi dengan melakukan pengumpulan, mempelajari, dan membaca berbagai bahan referensi yang berkaitan dengan aplikasi diagnosa serta metode *Dempster Shafer*. Adapun literatur yang digunakan meliputi buku, artikel, paper, jurnal, makalah, internet dan sumber lainnya.

## 2. Analisis dan Perancangan

Pada tahap ini dilakukan analisis spesifikasi aplikasi diagnosa kerusakan mesin bubut dan melakukan perancangan aplikasi diagnosa kerusakan mesin bubut, seperti perancangan proses dan antarmuka yang meliputi desain database sebagai *Back End*, dan sketsa tampilan sebagai *Front End*, dan lain sebagainya.

## 3. Pengkodean

Pada tahap ini dilakukan pengkodean aplikasi diagnosa kerusakan mesin bubut sesuai dengan analisis spesifikasi dan perancangan yang telah ditentukan.

## 4. Pengujian Aplikasi

Pada tahap ini dilakukan pengujian terhadap aplikasi diagnosa kerusakan mesin bubut yang telah dibangun, dan tingkat keakuratan dari sistem aplikasi yang telah dibuat.

## 5. Penyusunan Laporan

Pada tahap ini dilakukan penulisan dokumentasi dan laporan dari aplikasi diagnosa kerusakan mesin bubut yang dikembangkan.

### **1.7 Sistematika Penulisan**

Sistematika penulisan Tugas Skripsi ini dibagi atas beberapa bab, di mana masing-masing bab dibagi atas beberapa sub agar mempermudah penjelasan

mengenai penelitian yang dilakukan dan mempermudah pembaca dalam memahami isi penelitian. Adapun sistematika penulisan Tugas Skripsi ini adalah sebagai berikut :

## **BAB I PENDAHULUAN**

Pendahuluan berisi tentang Latar Belakang Masalah, Rumusan Masalah, Tujuan, Manfaat, Batasan Masalah, Metodologi Penelitian dan Sistematika Penulisan dalam pembuatan Tugas Skripsi.

## **BAB II LANDASAN TEORI**

Bab ini berisi teori-teori pengetahuan dasar yang di peroleh dari studi kepustakaan atau literatur dan dokumentasi internet yang digunakan untuk memahami permasalahan yang dibahas pada penelitian ini. Teori – teori pengetahuan dasar yang disajikan antara lain tentang Aplikasi kerusakan mesin bubut serta metode *Dempster Shafer*.

## **BAB III METODE PENELITIAN**

Bab ini menguraikan tahapan-tahapan sistematis yang digunakan untuk melakukan kajian penelitian. Tahapan-tahapan tersebut merupakan kerangka yang dijadikan pedoman penelitian untuk mencapai tujuan yang telah ditetapkan. Tahapan tersebut dimulai dari waktu dan tempat penelitian serta alat dan bahan yang digunakan dalam aplikasi diagnosa kerusakan mesin bubut dengan menggunakan metode *Dempster Shafer* pada PT PRIMA TETAP JAYA.

## **BAB IV HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil dan pembahasan dari aplikasi diagnosa kerusakan mesin bubut menggunakan metode *Dempster Shafer* dalam mengetahui diagnosa kerusakan mesin bubut pada PT PRIMA TETAP JAYA yang telah dibuat.

## **BAB V KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dari keseluruhan uraian bab bab penulisan skripsi dan saran yang diajukan untuk pengembangan lebih lanjut.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Mesin Bubut**

Mesin bubut merupakan suatu mesin yang biasanya dibuat dari logam, manfaatnya membentuk macam macam barang kerja semacam membubut rata, muka, tirus, ulir kanan, ulir kiri, *eksentrik*, alur serta *kartel* dengan metode menyayat, dengan gerakan utama berbalik. Tidak hanya buat memperindah penampilan suatu barang kerja, pula berguna buat membuat suatu barang memiliki guna lain. Dengan dibubut, suatu *spare part* bisa dipasangkan ke barang lain yang lebih dahulu sama sekali tidak dapat dipasangkan sama sekali tidak dapat dipasangkan. Perawatan memiliki peranan yang berarti, terdapat kalanya sangat memastikan kelancaran ataupun kemacetan dikala kerja, hingga terjalin aktivitas *maintenance* itu baru dilaksanakan sehabis peralatan peralatan yang dipunyai rusak. Hendaknya aktivitas pemeliharaan wajib bisa menjamin jika sepanjang proses kerja berlangsung, tidak hendak terjalin kemacetan yang ditimbulkan oleh perlengkapan mesin tersebut.

Adapun tujuan pemeliharaan mesin adalah :

1. Untuk membantu mengurangi pemakaian dan penyimpangan diluar jangkauan.
2. Dapat digunakan setiap saat.
3. Menjaga ongkos pemeliharaan serendah mungkin.
4. Menghindari kegiatan pemeliharaan yang dapat membahayakan para pekerja.
5. Mengadakan kerjasama yang erat dengan kantor.
6. Jenis pemeliharaan.

Adapun waktu menjalankan atau bekerja pada mesin adalah :



1. Menghidupkan mesin setelah yakin tidak ada bagian-bagian atau alat yang tidak mengganggu bagian-bagian yang bergerak.
2. Meneliti apakah bagian-bagian mesin bekerja normal.
3. Menjalankan mesin setelah alat-alat dan benda kerja terikat dan siap untuk dijalankan.
4. Jangan meninggalkan mesin yang sedang bekerja.

Hal hal yang harus diperhatikan setelah selesai bekerja :

1. Membersihkan dan merapikan alat-alat ke tempat semula.
2. Membersihkan mesin dari bekas-bekas sayatan atau tatal.
3. Membersihkan lantai sekitar mesin.
4. Melumasi bidang-bidang mesin setelah bagian-bagian lainnya.

## **2.2 Sistem Pakar**

Sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para pakar (Istiqomah & Fadlil, 2013). Menurut (Nas, 2019) sistem pakar merupakan sistem yang berbasis komputer yang menggunakan pengetahuan seorang pakar dalam menyelesaikan sebuah masalah. Sistem pakar dapat berfungsi sebagai konsultan yang memberi saran kepada pengguna sekaligus sebagai asisten bagi pakar (Wahyuni & Prijodiprodjo, 2013), Menurut (Merlina & Hidayat, 2012) Sistem pakar terdiri atas dua bagian pokok, yaitu lingkungan pengembangan (development environment) dan lingkungan konsultasi (consultation environment),

Secara Umum, Sistem pakar merupakan program komputer yang menilai dan perilaku manusia ataupun organisasi yang memiliki pengetahuan dan

pengalaman dalam bidang tertentu. Biasanya sistem seperti ini memiliki ilmu pengetahuan yang berisi pengalaman dan pengetahuan untuk menerapkan ilmu dasar untuk setiap situasi. Sistem pakar yang canggih dapat meningkatkan dengan menambah basis pengetahuan. Diantaranya banyak sistem pakar yang ada, salah satunya yaitu sistem diagnosa medis. Pengetahuan yang digunakan pada sistem pakar ini merupakan salah satu yang bisa menyelesaikan dan memecahkan masalah gejala diagnosa.

### **2.2.1 Ciri ciri Sistem Pakar**

Ciri-ciri sistem pakar yaitu sebagai berikut :

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data yang tidak pasti.
3. Dapat mengemukakan rangkaian alasan-alasan yang diberikan dengan cara yang bisa dipahami.
4. Berdasarkan aturan atau rule tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap agar bisa menghasilkan informasi yang lebih baik dan akurat.
6. Pengetahuan dan mekanisme penalaran jelas terpisah.
7. Keluarnya bersifat anjuran.

### **2.2.2 Konsep Dasar Sistem Pakar**

Ada enam hal yang menjadi konsep dasar dari sebuah Sistem Pakar, yaitu :

1. Keahlian (Expertise)

Keahlian dapat diperoleh dari pelatihan/training, membaca atau dari pengalaman. Keahlian itu meliputi :

- a. Fakta-fakta tentang area.
- b. Teori-teori tentang area.
- c. Aturan-aturan tentang apa yang harus dilakukan dalam situasi permasalahan.
- d. Strategi global untuk memecahkan.

#### 1. Pakar (Expert)

Sulit untuk mendefinisikan apakah yang dimaksud dengan pakar itu. Masalahnya adalah berapa banyak keahlian yang harus dimiliki oleh seseorang agar dapat dikualifikasikan sebagai pakar. Namun berikut ini dijelaskan beberapa kualifikasi yang harus dimiliki oleh seorang pakar:

- a. Dapat mengenal dan merumuskan
- b. Dapat memecahkan masalah dengan cepat dan
- c. Dapat menjelaskan suatu
- d. Dapat menentukan
- e. Belajar dari pengalaman

#### 2. Pemindehan Kepekaran (Transferring Expertise)

Tujuan dari sistem pakar adalah memindahkan keahlian dari seorang pakar ke komputer dan kemudian ke manusia lain yang bukan pakar. Proses ini meliputi empat kegiatan, yaitu :

- a. Memperoleh pengetahuan pakar
- b. Merepresentasikan pengetahuan ke dalam komputer
- c. Mengolah pengetahuan sehingga dapat menghasilkan kesimpulan.
- d. Memindahkan pengetahuan ke pengguna

#### 3. Menarik Kesimpulan (Inferencing)

Keistimewaan dari sistem pakar adalah kemampuan nalarnya. Komputer diprogram sehingga dapat membuat kesimpulan. Pengambilan keputusan ini dilaksanakan dalam komponen yang disebut inference engine.

#### 4. Aturan (Rule)

Kebanyakan sistem pakar adalah sistem berbasis rule, pengetahuan disimpan dalam bentuk rule-rule sebagai prosedur pemecahan masalah.

#### 5. Kemampuan Menjelaskan (Explanatin Capability)

Keistimewaan lain dari sistem pakar adalah kemampuan menjelaskan darimana asal sebuah solusi/rekomendasi diperoleh.

### **2.2.3 Keuntungan dan Kelemahan Sistem Pakar**

Pengguna sistem pakar secara umum memberikan keuntungan yang dimanfaatkan langsung oleh pengguna. Adapun keuntungan dari pengguna sistem pakar antara lain :

1. Masyarakat yang bukan pakar atau masyarakat awam bisa memanfaatkan keahlian di bidang tertentu tanpa kehadiran langsung seorang ahli.
2. Mengambil dan melestarikan keahlian langka.
3. Menghemat waktu dalam penyelesaian masalah yang kompleks.
4. Adanya kemungkinan untuk menggabungkan berbagai bidang pengetahuan dari berbagai pakar untuk dikombinasikan.
5. Pengetahuan dari seorang pakar dapat didokumentasikan tanpa ada batas waktu.
6. Sebagai media pembelajaran.
7. Mempunyai kemampuan untuk bekerja dengan informasi yang tidak lengkap dan tidak pasti.

8. Dapat beroperasi dalam lingkup yang berbahaya.
9. Dapat digunakan untuk mengakses basis data dengan cara cerdas.
10. Bertambahnya efisiensi pekerjaan tertentu, serta hasil solusi pekerjaan.

Sistem pakar tidak hanya memiliki keuntungan namun juga memiliki kelemahan, Adapun kelemahan pada sistem pakar yaitu :

1. Biaya yang diperlukan untuk pembuatan dan pemeliharaan sistem relatif mahal.
2. Berkurangnya daya kerja dan produktivitas manusia, karena semua dikerjakan secara otomatis oleh sistem.
3. Sulit di kembangkan karena erat kaitannya dengan ketersediaan para ahli.
4. Harus ada satu admin yang selalu update informasi dalam bidang yang sesuai dengan sistem pakar.
5. Membutuhkan waktu yang lama untuk mempelajari sistem.

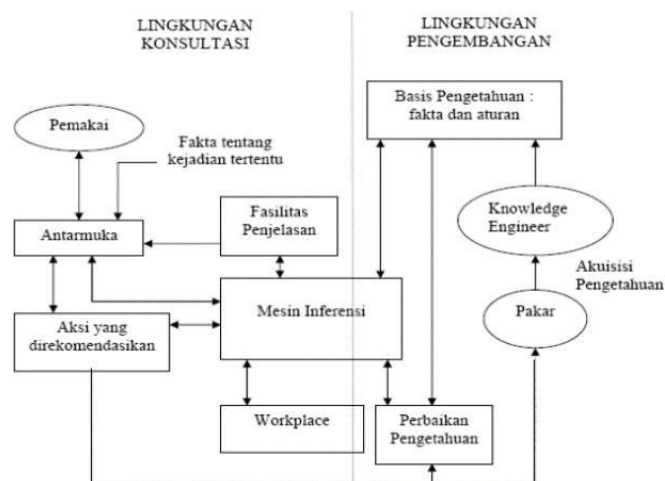
#### **2.2.4 Tujuan Sistem Pakar**

Keahlian dari para ahli ke komputer untuk kemudian dialihkan ke orang lain yang bukan ahli. Seorang pakar merupakan seseorang yang mampu menjelaskan suatu tanggapan, dan mempelajari hal yang baru seputar topik permasalahan, menyusun kembali pengetahuan jika dipandang perlu, dan memecahkan aturan-aturan jika dibutuhkan dan menentukan relevan atau tidaknya keahlian mereka. Proses keahlian pengetahuan membutuhkan 4 aktivitas, adalah sebagai berikut :

1. Tambahkan pengetahuan (dari para ahli atau sumber lainnya).
2. Pengetahuan (ke komputer).
3. Lakukan inferensi pengetahuan.
4. Pengalihan pengetahuan ke user

### 2.2.5 Struktur Sistem Pakar

Adapun definisi Struktur Sistem Pakar dua bagian penting dari sistem pakar, yaitu lingkungan pengembangan (development environment) dan lingkungan konsultasi (consultation environment). Lingkungan pengembangan digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan kedalam knowledge base (basis pengetahuan). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar.



Gambar 2.1 Komponen komponen sistem pakar

Keterangan :

#### 1. Akuisisi Pengetahuan

Sub sistem ini digunakan untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan meletakkannya ke dalam basis pengetahuan dengan format tertentu dalam bentuk representasi pengetahuan.

## 2. Basis Pengetahuan (Knowledge Base)

Basis pengetahuan yang diperlukan untuk memahami, memformulasikan dan menyelesaikan masalah. Basis pengetahuan terdiri Dari dua elemen dasar, yaitu :

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

## 3. Mesin Inferensi (Inference Engine)

Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada.

## 4. 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.

## 5. Antar Muka Pemakai (User Interface)

Digunakan sebagai media komunikasi antara pengguna dan sistem pakar. Komunikasi disajikan dalam bahasa alami (natural language) dan dilengkapi dengan grafik, menu, dan formulir elektronik.

## 6. Sub Sistem Penjelasan (Explanation Subsystem/Justifier)

Berfungsi memberi penjelasan kepada pengguna, kesimpulan dapat diambil. Kemampuan seperti ini sangat penting bagi pengguna untuk mengetahui proses pemindahan keahlian pakar maupun dalam pemecahan masalah.

## 7. Sistem Perbaikan Pengetahuan (Knowledge Refining System)

Kemampuan memperbaiki pengetahuan (Knowledge Refining System) dari seorang pakar diperlukan untuk menganalisa pengetahuan, belajar dari kesalahan

masa lalu, kemudian memperbaiki pengetahuannya sehingga dapat dipakai di masa yang akan datang.

#### 8. Pengetahuan (User)

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

### 2.3 Metode *Dempster Shafer*

Metode *Dempster Shafer* pertama kali diperkenalkan oleh Dempster, yang melakukan percobaan model ketidakpastian sebagai probabilitas tunggal. Kemudian pada tahun 1976 Shafer mempublikasikan teori Dempster itu pada sebuah buku yang berjudul *Mathematical Theory Of Evident. Dempster Shafer Theory Of Evidence*, menunjukkan suatu cara untuk memberikan bobot keyakinan sesuai fakta yang dikumpulkan. Pada teori ini dapat membedakan ketidakpastian dan ketidaktahuan. Teori *Dempster Shafer* adalah representasi, kombinasi dan propogasi ketidakpastian, dimana teori ini memiliki beberapa karakteristik yang secara instutitif sesuai dengan cara berfikir seorang pakar, namun dasar matematika yang kuat. Secara umum teori *Dempster Shafer* ditulis dalam suatu interval : [Belief, Plausibility]. Belief (Bel) adalah ukuran kekuatan *evidence* dalam mendukung suatu himpunan proposisi. Jika bernilai 0 maka mengindikasikan bahwa tidak ada *evidence*, dan jika bernilai 1 menunjukkan adanya kepastian. Plausibility (Pls) akan mengurangi tingkat kepastian dari *evidence*. Plausibility bernilai 0 sampai 1. Jika yakin akan X', maka dapat dikatakan bahwa  $Bel(X') = 1$ , sehingga rumus di atas nilai dari  $Pls(X) = 0$ . Menurut Giarratano dan Riley fungsi Belief dapat diformulasikan dan ditunjukkan pada persamaan (2.1) :



$$Bel(X) = \sum_{y \subset x} m(Y) \dots \dots \dots (2.1)$$

$$Pls(x) = 1 - \sum_{y \subset x} m(Y) \dots \dots \dots (2.2)$$

Dimana :

Bel (X) = Belief (X)

Pls (X) = Plausibility (X)

m (X) = mass function dari (X)

m (Y) = mass function dari (Y)

Teori *Dempster Shafer* menyatakan adanya *frame of discrement* yang dinotasikan dengan simbol ( $\Theta$ ). *Frame of discrement* merupakan semesta pembicaraan dari sekumpulan hipotesis sehingga sering disebut dengan *environment* yang ditunjukkan pada persamaan (2.3) :

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_N\} \dots \dots \dots (2.3)$$

Dimana :

$\Theta$  = frame of discrement atau environment

$\theta_1, \dots, \theta_N$  = element/ unsur bagian dalam environment

*Environment* mengandung elemen elemen yang menggambarkan kemungkinan sebagai jawaban, dan hanya ada satu yang akan sesuai dengan jawaban yang dibutuhkan. Kemungkinan ini dalam teori *Dempster Shafer* disebut dengan *power set* dan dinotasikan dengan  $P(\Theta)$ , setiap elemen dalam *power set* ini memiliki nilai interval antara 0 sampai 1. Sehingga dapat dirumuskan pada persamaan (2.4)

$$m : P(\Theta) [0,1] \dots\dots\dots(2.4)$$

$$\sum_{x \in P(\Theta)} m(x) = 1$$

Dimana

$P(\Theta)$  = power set

$m(X)$  = mass function (X)

Massa *function* ( $m$ ) dalam teori *Dempster shafer* adalah tingkat kepercayaan dari suatu *evidence* (gejala), sering disebut dengan *evidence measure* sehingga dinotasikan dengan ( $m$ ). Tujuannya adalah mengaitkan ukuran kepercayaan elemen elemen  $\theta$ . Tidak semua *evidence* secara langsung mendukung tiap tiap elemen. Untuk itu perlu adanya probabilitas fungsi *densitas* ( $m$ ). Nilai  $m$  tidak hanya mendefinisikan elemen elemen  $\theta$  saja, namun juga semua subsetnya.

Sehingga jika  $\theta$  berisi  $n$  elemen, maka subset  $\theta$  adalah  $2^n$ . Jumlah semua  $m$  dalam subset  $\theta$  sama dengan 1. Apabila tidak ada informasi apapun untuk memilih hipotesis, maka nilai :  $m\{\theta\} = 1,0$ . Apabila diketahui  $X$  adalah subset dari  $\theta$ , dengan  $m_1$  sebagai fungsi densitasnya, dan  $Y$  juga merupakan subset dari  $\theta$  dengan  $m_2$  sebagai fungsi densitasnya, maka dapat dibentuk fungsi kombinasi  $m_1$  dan  $m_2$  sebagai  $m_3$ , yaitu ditunjukkan pada persamaan (2.5) :

$$m_3(Z) = \frac{\sum_{x \cap Y = Z} m_1(X)m_2(Y)}{1 - \sum_{x \cap Y = \emptyset} m_1(X)m_2(Y)} \dots\dots\dots(2.5)$$

$m_3(Z)$  = mass *function* dari *evidence* (Z)

$m_1(X)$  = mass *function* dari *evidence* (X), yang diperoleh dari nilai keyakinan suatu *evidence* dikalikan dengan nilai *disbelief* dari *evidence* tersebut.

$m_3(Y)$  = mass *function* dari *evidence* (Y), yang diperoleh dari nilai keyakinan suatu

*evidence* dikalikan dengan nilai *disbelief* dari *evidence* tersebut.

$\sum_{X \cap Y = Z} m_1(X)m_2(Y)$  = merupakan nilai kekuatan dari *evidence* Z yang diperoleh dari kombinasi nilai keyakinan sekumpulan *evidence*.

## 2.4 Unified Modeling Language (UML)

UML (Unified Modeling Language) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Tujuan Penggunaan UML adalah :


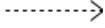




1. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
2. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
3. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.





UML bisa juga berfungsi sebagai sebuah (blue print) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi secara detail tentang koding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (reverse engineering).

### 2.4.1 Use case Diagram

*Use case* diagram merupakan suatu diagram yang berisi *use case*, actor, serta relationship (hubungan) di dalamnya. *Use case* diagram adalah titik awal dalam memahami atau menganalisis kebutuhan sistem dan dapat menggambarkan dengan detail bagaimana suatu sistem memproses atau melakukan sesuatu pada saat perancangan sistem.

Tabel 2.1 Simbol *Use case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.






7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

### 2.4.2 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

**Tabel 2.2 Tabel Activity Diagram**


NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------


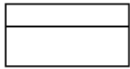


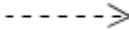

1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity</i> <i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

### 2.4.3 Class Diagram

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

**Tabel 2.3 Simbol Class Diagram**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).

2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

## 2.5 XAMPP

Menurut Yogi wicaksono dalam (Hadi, 2017) bahwa, “XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan

menggunakan pengolah data MySQL dikomputer local”. (Yudhanto & Prasetyo, 2018) menyebutkan XAMPP adalah “Kompilasi dari program aplikasi gratis terfavorit dikalangan developer atau programmer yang berguna untuk pengembangan website berbasis PHP dan MYSQL”. XAMPP berperan sebagai server web pada komputer yang kita gunakan. XAMPP juga dapat disebut sebuah CPanel server virtual, yang dapat membantu kita melakukan preview sehingga dapat memodifikasi website tanpa harus online atau terhubung dengan internet.

Dengan menggunakan aplikasi XAMPP, kita sudah mendapatkan software lengkap yang bisa dijalankan pada sistem operasi windows. XAMPP akan membuat konfigurasi secara lengkap dan otomatis termasuk aplikasi management Database yaitu PHPMyadmin.

## 2.6 HTML

Menurut (Anhar, 2010) di dalam bukunya yang berjudul Panduan Menguasai PHP dan MYSQL secara Otodidak, HTML adalah “sekumpulan simbol simbol atau tag tag yang dituliskan dalam sebuah file yang digunakan untuk menampilkan halaman pada *web browser*”.

Dengan menuliskan kode kode HTML, pada baris kode HTML tersebut mengandung perintah perintah teks yang biasa disebut Tag untuk menampilkan tulisan, warna, media ( suara, gambar, video, serta animasi) serta link yang dapat menghubungkan ke situs lain atau pun link yang kita buat sendiri untuk topik topik lainnya.

Hal ini juga di sampaikan oleh (Yudhanto & Prasetyo, 2018) di dalam bukunya yang berjudul Panduan Mudah Belajar *Framework Laravel*, bahwa “HTML adalah sebuah penKita atau mark”. Dan seiring perkembangannya, HTML



kini sudah lebih maju dengan versi HTML yang lebih baru, yaitu HTML5, atau HTML versi 5.

## **2.7 CSS**

Jika HTML di analogikan sebagai tulang pada website yang berfungsi membuat kerangka tertentu pada website sesuai baris kode HTML yang dituliskan. Maka CSS digunakan sebagai format untuk membuat format tampilan kerangka HTML yang telah di buat menjadi lebih bagus dan elegan, responsive dan adaptif.

“CSS adalah sebuah bahasa stylesheet yang digunakan untuk mengatur tampilan suatu website, baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan” (Suryana & Koesheryatin, 2014). Menurut (Yudhanto & Prasetyo, 2018), CSS adalah “Komponen pembangun website yang digunakan untuk memformat HTML agar menjadi lebih bagus dan efektif dalam tampilan”.

## **2.8 PHP**

PHP atau Hypertext Preprocessor sebuah bahasa pemograman yang memiliki kemampuan / fungsi untuk memanggil dirinya sendiri. PHP merupakan salah satu bahasa pemograman web yang sangat populer yang digunakan untuk mengembangkan web. Menurut (Yudhanto & Prasetyo, 2018) PHP adalah bahasa pemograman script server side yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan web. Sedangkan menurut (Achmad Solichin, n.d), PHP merupakan salah satu bahasa server side scripting yang dapat menghasilkan halaman web yang dinamis. Bahasa server side scripting merupakan program yang dikompilasi di server.

## 2.9 MySQL

SQL merupakan kepanjangan dari Structure Query Language yang digunakan untuk mengolah database. Menurut Budi Raharjo dalam (Agusli et al., 2017) MySQL adalah software Relation Database Management System (RDBMS) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (multi-user) dan dapat melakukan suatu proses secara sinkron atau berbarengan (multi-threaded).