

## PERANCANGAN SISTEM PAKAR DIAGNOSA PENYAKIT LUPUS ERITMATOSUS SISTEM(LES) DENGAN METODE FORWARD CHAINING MENGUNAKAN PEMROGRAMAN PHP DAN MySQL

Novi Yona Sidratul Munti  
Pasca Sarjana, Universitas Putra Indonesia “YPTK” Padang  
e-mail: sikumbang\_ona@yahoo.com

### Abstract

Tujuan penelitian adalah merancang dan membangun rekayasa perangkat lunak sistem pakar berbasis web yang mampu mendiagnosa penyakit *Lupus Eritmatosus Sistem (LES)* pada manusia untuk mendapatkan solusi dan informasi dengan mudah dan cepat. Hasil yang ditampilkan berupa kondisi *user* terkait dengan *Lupus Eritmatosus Sistem (LES)*. Hasil juga dilengkapi dengan penjelasan penyakit dan solusi pengobatan yang ditampilkan dalam bentuk website menggunakan pemrograman PHP dengan database MySQL. Kesimpulan dalam penelitian ini adalah bahasa pemrograman PHP dan MySQL terbukti mampu diimplementasikan dalam rekayasa sistem pakar untuk mendiagnosa penyakit *Lupus Eritmatosus Sistem (LES)*. Metode forward chaining terbukti mampu melakukan penelusuran /pelacakan gejala penyakit *Lupus Eritmatosus Sistem (LES)* dengan mudah dan cepat. Sistem online dapat membantu *user* mendapatkan informasi tentang jenis-jenis penyakit, gejala dan solusi pengobatan pada penyakit *Lupus Eritmatosus Sistem (LES)*.

**Kata kunci:** Sistem pakar, *Lupus Eritmatosus Sistem (LES)*, Forward Chaining, PHP MySQL

### 1. PENDAHULUAN

Pada masa era globalisasi sekarang ini perkembangan teknologi khususnya komputer sudah banyak dimanfaatkan pada perusahaan atau pun dibidang ilmu pengetahuan. Dengan demikian peranan teknologi informasi itu sangat penting dalam setiap aspek kehidupan, baik itu dalam bidang ekonomi, ilmu pengetahuan, Pemerintah dan bidang teknologi itu sendiri.

Sistem pakar adalah Program berbasis pengetahuan yang menyediakan solusi-solusi dengan kualitas pakar untuk masalah yang spesifik. Sistem pakar merupakan Program komputer yang meniru proses pemikiran dan pengetahuan pakar dalam menyelesaikan suatu masalah tertentu. Perancangan sistem pakar banyak digunakan dalam bidang kesehatan karena sistem pakar dipandang sebagai cara penyimpanan pengetahuan pakar pada bidang tertentu dalam Program komputer sehingga keputusan dapat diberikan dalam melakukan penalaran secara cerdas.

Lupus adalah Penyakit Autoimun yang muncul ketika tubuh terkena zat asing tertentu, seperti Bakteri dan Virus. Penyakit Lupus adalah peradangan kronis yang terjadi ketika sistem

imun tubuh menyerang organ dan jaringan tubuh. Peradangan yang disebabkan oleh lupus dapat berefek pada berbagai sistem di dalam tubuh, antara lain Sendi, Kulit, Ginjal, Sel Darah, Jantung dan Paru-Paru.

Berdasarkan permasalahan di atas maka penulis tertarik mengangkat permasalahan tentang Lupus dan direalisasikan kedalam bentuk Skripsi yang diberi Judul :”**PERANCANGAN SISTEM PAKAR DIAGNOSA PENYAKIT LUPUS ERITMATOSUS SISTEM (LES) DENGAN METODE FORWARD CHAINING MENGGUNAKAN PEMROGRAMAN PHP DAN MySQL** “ .

### 2. KAJIAN LITERATUR

#### 2.1 Rekayasa Perangkat Lunak (RPL)

Rekayasa Perangkat lunak adalah seluruh perintah yang digunakan untuk memproses informasi. Perangkat lunak dapat berupa program atau prosedur. Program adalah kumpulan perintah yang dimengerti oleh komputer sedangkan prosedur adalah perintah yang dibutuhkan oleh pengguna dalam memproses informasi (O'Brien, 1999). Pengertian RPL sendiri adalah sebagai berikut:

Suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai

dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, disain , pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan (Mulyanto, 2008 :2)

Karakteristik Perangkat Lunak lebih dikenal sebagai elemen logik dari pada fisik, oleh karena itu perangkat lunak memiliki karakteristik yang berbeda dari perangkat keras :

1. Perangkat lunak dikembangkan atau direkayasa, jadi tidak diproduksi dalam pengertian klasik.
2. Merupakan produk yang unik (tidak ada seri produksi).
3. Perangkat lunak tidak pernah akan rusak atau aus karena selalu diperbaharui
4. Tidak terlihat (invisible).
5. Perangkat lunak pada umumnya dibangun sesuai keinginan, jadi tidak dibentuk dari komponen yang sudah ada.
7. *Fleksibel*, sehingga mudah dimodifikasi.
8. Dihubungkan (*linked*) dengan sistem komputer.

### 2.1.1. Tujuan Rekayasa Perangkat lunak

Secara umum tujuan rekayasa perangkat lunak tidak berbeda dengan rekayasa yang lain, menurut Mulyanto (2008:2) mengungkapkan: “*tujuan dari rekayasa perangkat lunak*” adalah sebagai berikut:

1. Memperoleh biaya produksi perangkat lunak yang rendah.
2. Menghasilkan perangkat lunak yang kinerjanya tinggi, handal dan tepat waktu.
3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis platform.
4. Menghasilkan perangkat lunak yang biaya perawatannya rendah.
5. Mampu melaksanakan analisis kebutuhan, perancangan, implementasi dan pengujian perangkat lunak dengan pendekatan terstruktur.
6. Mampu menyusun dokumentasi pengembangan perangkat lunak.

### 2.2 Sejarah Singkat Artificial intelligence

Kecerdasan Buatan (*artificial intelligence*) merupakan inovasi baru dibidang ilmu pengetahuan. Pada tahun 1950-an para ilmuwan dan peneliti mulai memikirkan bagaimana cara agar mesin dapat melakukan pekerjaan seperti yang biasa dilakukan para manusia. Alan Turing, seorang matematikawan dari Inggris pertama kali mengusulkan adanya tes untuk melihat bisa tidaknya sebuah mesin dikatakan cerdas. Hasil tes tersebut kemudian dikenal Turing Tes, dimana si mesin tersebut menyamar seolah-olah sebagai seorang didalam

suatu permainan yang mampu memberikan respon terhadap serangkaian pertanyaan yang diajukan. Turing beranggapan bahwa jika mesin dapat membuat seorang percaya bahwa dirinya mampu berkomunikasi dengan orang lain, maka dapat dikatakan bahwa mesin tersebut cerdas (seperti layaknya manusia).

Kecerdasan buatan atau “*Artificial Intelligence* “ itu sendiri dimunculkan oleh seorang profesor dari Massachusetts Institute of Technology yang bernama John McCarthy pada tahun 1956 pada Dartmouth Conference yang dihadiri para peneliti AI. Pada konferensi tersebut juga didefinisikan tujuan utama dari kecerdasan buatan, yaitu: mengetahui dan memodelkan proses-proses berfikir manusia dan mendesain mesin agar dapat menirukan kelakuan manusia tersebut.

Pada awalnya, kecerdasan buatan hanya ada di Universitas-Universitas dan laboratorium penelitian, serta hanya sedikit produk yang dihasilkan dan dikembangkan. Menjelang akhir 1970-an dan 1980-an, mulai dikembangkan secara penuh dan hasilnya berangsur-angsur dipublikasikan di khalayak umum. Permasalahan di dalam kecerdasan buatan akan selalu bertambah dan berkembang seiring dengan laju perkembangan zaman menuju arah globalisasi dalam setiap aspek kehidupan manusia, yang membawa persoalan-persoalan yang semakin beragam pula. Program kecerdasan buatan lebih sederhana dalam pengoperasiannya, sehingga banyak membantu pemakai. Program konvensional dijalankan secara prosedural dan kaku, rangkaian tahap solusinya sudah didefinisikan secara tepat oleh pemrogramnya. Sebaliknya, pada program kecerdasan buatan untuk mendapatkan solusi yang memuaskan dilakukan pendekatan trial and error, mirip seperti apa yang dilakukan oleh manusia (Anita desiani, Arhami, 2005).

### 2.2.1. Defenisi Kecerdasan Buatan (Artificial Intelligence)

Definisi kecerdasan buatan yang telah didefinisikan oleh beberapa ahli:

1. H. A. Simon (1987) :

“*Kecerdasan buatan (Artificial Intelligence) merupakan kawasan penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang dalam pandangan manusia adalah cerdas*”.

2. Rich and Knight (1991):

“*Kecerdasan Buatan (Artificial Intelligence) merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-*

hal yang pada saat ini dapat dilakukan lebih baik oleh manusia”.

3. Encyclopedia Britannica:

“Kecerdasan Buatan (*Artificial Intelligence*) merupakan cabang dari ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan, dan memproses informasi berdasarkan metodeheuristic atau dengan berdasarkan sejumlah aturan” (Sutojo, Mulyanto dan Suhartono, 2010).

4. Rich and Knight (1991):

“Kecerdasan buatan (*Artificial Intelligence*) merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang saat ini dapat dilakukan lebih baik oleh manusia”.

Kecerdasan tiruan (*Artificial Intelligence*) adalah suatu area dalam ilmu komputer. Istilah tersebut mencakup banyak definisi (Ray, 1996) tetapi sebagian pakar setuju bahwa kecerdasan tiruan (*Artificial Intelligence*) berkaitan dengan dua ide dasar. Pertama, ide yang melibatkan pembelajaran proses pemikiran manusia (untuk memahami apa yang dimaksud dengan kecerdasan); kedua, berkaitan dengan representasi dan duplikasi proses tersebut melalui mesin (misalnya, komputer dan robot).

Kecerdasan tiruan juga merupakan tingkah laku mesin yang jika dilakukan oleh manusia akan disebut cerdas. Suatu definisi yang membangkitkan pemikiran dinyatakan oleh Rich dan Knight (1991): “Kecerdasan tiruan adalah studi tentang bagaimana membuat komputer melakukan hal yang pada saat itu lebih baik dilakukan oleh manusia”.

Bidang-bidang kecerdasan tiruan (Turban, 2005) adalah:

1. Sistem Pakar
2. Pemrosesan Bahasa Alami
3. Speech (*Voice*) Understanding
4. Sistem Robotik dan Sistem Sensor
5. Computer Vision dan Recognition
6. Intelligent Computer-Aided Instruction
7. Komputasi Saraf
8. Game Playing

### 2.3 Sistem Pakar (Expert System)

Sistem pakar dapat membantu aktivitas para pakar sebagai asisten yang berpengalaman dan mempunyai pengetahuan yang dibutuhkan. Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (*inference rules*) dengan

basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam basis pengetahuan, yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu.

Biasanya pakar manusia mampu melakukan hal berikut : *Mengenal* dan *merumuskan persoalan*, Memecahkan persoalan dengan cepat dan tepat, Menjelaskan solusi tersebut, Belajar dari pengalaman, Menyusun ulang pengetahuan, Membagi-bagi aturan jika diperlukan, Menetapkan relevansi Keahlian adalah pengetahuan ekstensif yang spesifik terhadap tugas yang dimiliki pakar.

Keahlian sering dicapai dari pelatihan, membaca, dan mempraktikkan. Keahlian mencakup pengetahuan eksplisit, misalnya teori yang dipelajari dari buku teks atau kelas, dan pengetahuan implisit yang diperoleh dari pengalaman. **Pengembangan sistem pakar** dibagi menjadi dua generasi. Kebanyakan **sistem pakar generasi pertama** menggunakan aturan jika-maka untuk merepresentasikan dan menyimpan pengetahuannya. **Sistem pakar generasi kedua** jauh lebih fleksibel dalam mengadopsi banyak representasi pengetahuan dan metode pertimbangan.

Ada beberapa definisi tentang sistem pakar :

1. Menurut *Durkin*: sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan oleh seorang pakar.
2. Menurut *Ingizio*: sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.
3. Menurut *Giarratano* dan *Riley*: sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar.

**Sistem pakar menurut Turban** (1995). Disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultationenvironment*).Lingkungan pengembangan sistem pakar digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar.

#### 2.3.1. Manfaat Sistem pakar

Manfaat dari sistem pakar antara lain :

1. Memungkinkan orang awam mengerjakan pekerjaan orang ahli.
2. Bisa melakukan proses secara berulang secara otomatis .
3. Menyimpan pengetahuan dan keahlian orang pakar.
4. Mampu dan mengambil dan melestarikan keahlian para pakar (terutama termasuk keahlian yang langka ).
5. Mampu beroperasi dalam lingkungan yang berbahaya.
6. Tidak memerlukan biaya pada saat tidak digunakan, sedangkan pada pakar manusia sehari-hari memerlukan biaya.
7. Dapat digandakan (diperbanyak ) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya.
8. Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama .
9. Menghemat waktu dalam pengambilan keputusan.
10. Meningkatkan kreativitas dan produktifitas karna dapat memberi nasehat yang konsisten dan mengurangi masalah.
11. Meningkatkan kapabilitas sistem terkomputerisasi yang lain. integrasi Sistem Pakar dengan sistem komputer lain membuat lebih efektif , dan bisa mencakup lebih banyak aplikasi.
12. Mampu menyediakan pelatihan. pengguna pemula yang berkerja dengan Sistem Pakar akan menjadi lebih berpengalaman. Fasilitas penjelas dapat berfungsi sebagai guru.

### 2.3.2. Ciri-ciri Sistem Pakar

Menurut Arhami, Muhamad (2005), dalam pembuatan sistem pakar harus diketahui ciri-ciri dan kategori masalah sistem pakar. Pada umumnya sistem pakar bersifat:

1. Memiliki informasi yang handal, baik dalam menampilkan langkah-langkah antara maupun dalam menjawab pertanyaan-pertanyaan tentang proses penyelesaian.
2. Mudah dimodifikasi, yaitu dengan menambah atau menghapus suatu kemampuan dari basis pengetahuannya.
3. Dapat digunakan dalam berbagai jenis komputer.
4. Memiliki kemampuan untuk beradaptasi.

Selain Arhami (Kusrini, 2008) juga menyatakan beberapa ciri-ciri dari sistem pakar, yaitu :

1. Terbatas pada bidang keahlian yang spesifik.
2. Dapat mengemukakan rangkaian alasan-alasan yang diberikan dengan cara yang Dapat dipahami.

3. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
4. Berdasarkan rule atau kaidah tertentu.
5. Dirancang untuk dikembangkan secara bertahap.
6. Keluaran bersifat anjuran atau nasehat.
7. Keluaran tergantung dari dialog dengan *user*.

### 2.3.3. Keuntungan Sistem Pakar

Ada beberapa kelebihan yang dapat diperoleh dari pembuatan sistem pakar menurut Arhami, Muhamad (2005), yaitu:

1. Menjadikan pengetahuan dan nasihat lebih mudah didapat.
2. Meningkatkan output dan produktivitas.
3. Menyimpan kemampuan dan keahlian pakar.
4. Meningkatkan reabilitas.
5. Memberikan respons (jawaban) yang cepat.
6. Merupakan panduan yang cerdas.
7. Dapat bekerja dengan informasi yang kurang lengkap dan mengandung ketidakpastian.
8. Basis data cerdas, bahwa sistem pakar dapat digunakan untuk mengakses basis data dengan cara cerdas.

### 2.3.4. Kelemahan Sistem Pakar

Selain sistem pakar mempunyai kelebihan, menurut Arhami, Muhamad (2005), sistem pakar juga mempunyai kelemahan, yaitu:

1. Masalah dalam mendapatkan pengetahuan di mana pengetahuan tidak selalu bisa didapatkan dengan mudah, karena kadangkalapakar dari masalah yang kita buat tidak ada, dan walaupun ada kadangkadangkad pendekatan yang dimiliki oleh pakar berbeda-beda.
2. Untuk membuat suatu sistem pakar yang benar-benar berkualitas tinggi sangatlah sulit dan memerlukan biaya yang sangat besar untuk pengembangan dan pemeliharanya.
3. Boleh jadi sistem pakar tidak dapat membuat keputusan.
4. Sistem pakar tidak 100% menguntungkan, karena seseorang yang terlibat dalam sistem pakar tidak selalu benar. Oleh karena itu perlu diuji ulang secara teliti sebelum digunakan.

### 2.3.5. Komponen Sistem Pakar

Menurut (Kusrini, 2008). Komponen utama pada sistem pakar meliputi:

1. Basis Pengetahuan (*Knowledge Base*)  
Basis pengetahuan merupakan inti dari suatu sistem pakar, yaitu berupa representasi

pengetahuan dari pakar. Basis pengetahuan tersusun atas fakta dan kaidah. Fakta adalah informasi tentang objek, peristiwa, atau situasi. Kaidah adalah cara untuk membangkitkan suatu fakta baru dari fakta yang sudah diketahui.

## 2. Mesin Inferensi (*Inference Engine*)

Mesin inferensi berperan sebagai otak dari sistem pakar. Mesin inferensi berfungsi untuk memandu proses penalaran terhadap suatu kondisi, berdasarkan pada basis pengetahuan yang tersedia. Di dalam mesin inferensi terjadi proses untuk memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan dalam rangka mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi penalaran dan strategi pengendalian. Strategi penalaran terdiri dari strategi penalaran pasti (*Exact Reasoning*) dan strategi penalaran tak pasti (*Inexact Reasoning*). *Exact reasoning* akan dilakukan jika semua data yang dibutuhkan untuk menarik suatu kesimpulan tersedia, sedangkan *inexact reasoning* dilakukan pada keadaan sebaliknya. Strategi pengendalian berfungsi sebagai panduan arah dalam melakukan proses penalaran. Terdapat tiga teknik pengendalian yang sering digunakan, yaitu *forward chaining*, *backward chaining*, dan gabungan dari kedua teknik pengendalian tersebut.

## 3. Basis Data (*Data Base*)

Basis data terdiri atas semua fakta yang diperlukan, di mana fakta-fakta tersebut digunakan untuk memenuhi kondisi dari kaidah-kaidah dalam sistem. Basis data menyimpan semua fakta, baik fakta awal pada saat sistem mulai beroperasi, maupun fakta-fakta yang diperoleh pada saat proses penarikan kesimpulan sedang dilaksanakan. Basis data digunakan untuk menyimpan data hasil observasi dan data lain yang dibutuhkan selama pemrosesan.

### 2.3.6. Klasifikasi Sistem Pakar

Berdasarkan kegunaannya Arhami, Muhamad (2005), sistem pakar dapat diklasifikasikan menjadi sebagai berikut:

1. **Diagnosis.**  
Sistem ini digunakan untuk memecahkan atau menemukan masalah atau kerusakan yang sedang terjadi.
2. **Pengajaran.**  
Sistem ini digunakan untuk proses belajar mengajar, sistem ini diharapkan dapat mengetahui letak dan memberikan cara untuk memperbaiki kekurangan tersebut.
3. **Interpretasi**

Digunakan untuk menganalisa data yang tidak lengkap, tidak teratur, dan juga kontradiktif.

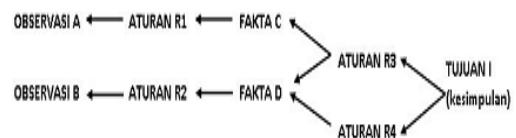
4. **Prediksi**  
Kemampuan untuk memprediksi yang akan terjadi.
5. **Perencanaan**  
Kemampuan untuk merencanakan sistem yang akan dipakai.
6. **Kontrol**  
Digunakan untuk mengontrol kegiatan yang membutuhkan ketepatan waktu tinggi.

Terdapat dua pendekatan untuk mengontrol inferensi dalam sistem pakaberbasis aturan, yaitu:

## 1. Pelacakan Kebelakang (*Backward Chaining*).

Pelacakan kebelakang adalah pendekatan yang dimotori tujuan (*goal-driven*). Dalam pendekatan ini pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya proses pelacakan menggunakan premis untuk aturan tersebut sebagai tujuan baru dan mencari aturan lain dengan tujuan baru sebagai kesimpulannya. Proses berlanjut sampai semua kemungkinan ditemukan (Arhami, 2005).

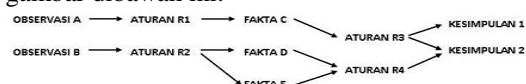
Prosesnya diperlihatkan pada gambar dibawah ini:



Gambar 2.1 Proses Backward Chaining

## 2. Pelacakan Kedepan (*Forward Chaining*).

Pelacakan kedepan (*forward chaining*) adalah pendekatan yang dimotori data (*data-driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan dan selanjutnya mencoba menggambarkan kesimpulan, pelacakan kedepan mencari fakta yang sesuai dengan bagian IF dari aturan IF-THEN. Prosesnya ditunjukkan pada gambar dibawah ini:



Gambar 2.2 Proses Forward Chaining

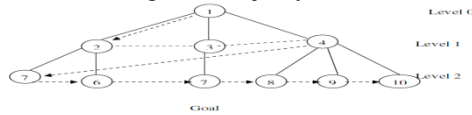
Kedua metode inferensi tersebut dipengaruhi oleh tiga macam penelusuran, yaitu :  
*a. Depth-First Search*

Melakukan penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkatan dalam yang berurutan.

**Gambar 2.3 Diagram alir teknik penelusuran Depth-first search**

**b. Breadth-First Fearch**

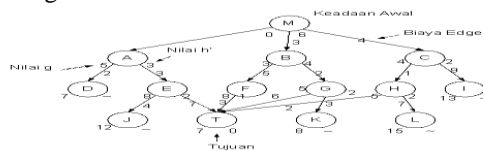
Bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat selanjutnya.



**Gambar 2.4 Diagram alir teknik penelusuran Breadth-first fearch**

**c. Best-First Search (BFS)**

Metode yang membangkitkan simpul dari simpul sebelumnya. Best-first search memilih simpul baru yang memiliki biaya terkecil diantara semua leaf nodes (simpul-simpul pada level terdalam) yang pernah dibangkitkan.



**Gambar 2.5 Diagram alir teknik penelusuran Best-first search**

**d. Workplace**

Merupakan area dari sekumpulan memori kerja (*working memory*). *Workplace* digunakan untuk merekan antara hasil kesimpulan yang dicapai.

**a. Fasilitas Penjelasan**

Merupakan komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pemakai.

**b. 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 mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya (Arhami, 2005).

**2.4 Sekilas Tentang Web**

*Web* adalah salah satu aplikasi yang berisikan dokumen-dokumen multi media (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (*hypertext transfer protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut browser (Rudyanto Arief, 2001). Beberapa jenis browser

yang populer saat ini adalah: *internet Explorer* yang di produksi oleh *microsoft*, *mozilla Firefok*, dan *Safari* yang di produksi oleh *Apple*.

Browser (perambah) adalah aplikasi yang mampu menjalankan dokumen-dokumen web dengan cara di terjemahkan. Prosesnya di lakukan oleh komponen yang terdapat di dalam aplikasi browser yang biasa di sebut *web engine*. Semua dokumen *web* di tampilkan oleh browser dengan cara di terjemahkan (Rudyanto Arief, 2001)

Situs *web* adalah dokumen-dokumen *web* yang terkumpul menjadi satu kesatuan yang memiliki *Unifed Resource Locator* (URL) atau *domain* dan biasanya di-publish di *internet* atau *intranet* (Rudyanto Arief, 2001).

Sebelum mulai membuat desain tampilan web, ada baiknya memahami lebih dahulu fungsi sebuah *website*, sehingga desain yang dibuat disesuaikan dengan fungsi *website* tersebut. Jasmadi (2004) mengungkapkan: “secara umum *website* mempunyai fungsi”, yaitu:

1. Fungsi Komunikasi  
Sebagian besar *website* mempunyai fungsi komunikasi. Beberapa fasilitas yang memberikan fungsi komunikasi ini, seperti : *web base email*, halaman *form contact*, *chatting* dan lain-lain.
2. Fungsi Informasi  
*Website* mempunyai fungsi informasi seperti *news*, *profile company*, *library*, referensi, dan lain-lain.
3. Fungsi Entertainment  
*Website* mempunyai fungsi hiburan. Beberapa contoh *website* dengan fungsi ini, misalnya web-web yang menyediakan *online game*, *online music*, *online movie*, dan sebagainya.
4. Fungsi Transaksi  
Sebuah *website* dapat dijadikan saran untuk melakukan transaksi bisnis, seperti : *online order*, pembayaran menggunakan kartu kredit, dan lain-lain.

“Sebuah *website* dapat dijadikan sarana untuk melakukan transaksi bisnis, seperti: *online order*, *pembayaran menggunakan kartu kredit*, dan lain-lain” (Jasmadi, 2004).

*Website* atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

## 2.5 Personal Home Page (PHP)

PHP merupakan bahasa (*scripting language*) yang ditempelkan (*embedded*) ketika digunakan dalam halaman *web*. Maksudnya adalah bahwa kode PHP ditempelkan di dalam kode *html* (Rulianto kurniawan, 2010). *Software* PHP bekerja bersama dengan *Web Server*. *Web Server* adalah *software* yang mengirim halaman *web* kepada dunia.

*Javascript* adalah bahasa skrip yang ditempelkan pada kode HTML dan diproses di sisi klien. Dengan adanya bahasa ini, kemampuan dokumen HTML menjadi semakin luas (Rulianto kurniawan, 2010). Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk memvalidasi masukan-masukan pada formulir sebelum formulir dikirimkan ke server. *Javascript* bukanlah bahasa Java dan merupakan dua bahasa yang berbeda. *Javascript* diinterpretasikan oleh klien (kodenya bisa dilihat pada sisi klien), sedangkan kode Java dikompilasi oleh pemrogram dan hasil kompilasinyalah yang dijalankan oleh klien.

Selain dari php juga terdapat suatu system yang di namakan *Cascading Style Sheet* (CSS). *Cascading Style Sheet* (CSS) merupakan sebuah *document* yang berguna untuk melakukan pengaturan pada komponen halaman *web*, inti dari *document* ini adalah memformat halaman *web* standar menjadi bentuk *web* yang memiliki kualitas yang lebih indah dan menarik. CSS biasanya digunakan untuk melakukan pengaturan global yang berkaitan dengan objek tetap, misalnya memberikan warna pada halaman *web*, pengaturan lebar dan kecil bagian *web* serta menentukan bentuk *font* jenis huruf yang digunakan secara menyeluruh dalam halaman *web*.

### 2.5.1. Pengenalan PHP

*PHP* merupakan perangkat lunak *open source*, yang mana penulisan kode program *PHP* menyatu dengan *HTML* yang berjalan pada *server-side*. Hal ini berarti semua sintak yang telah ditulis akan sepenuhnya dijalankan pada *server*; hanya hasil dari *script* yang telah dieksekusi saja yang dikirimkan pada sisi *client/browser* tanpa bisa mengetahui kode yang digunakan. *PHP* merupakan bahasa pemrograman yang sesuai untuk membuat aplikasi website dinamis seperti *CMS* karena memiliki performa yang tinggi, mudah dipelajari, *multiplatform*, aman, *open source*, serta mudah dikoneksikan dengan berbagai macam *database*.

Dan dengan pengembangan yang dilakukan *PHP* kini memiliki kinerja yang lebih tinggi daripada versi-versi sebelumnya,

berkemampuan *OOP* (*Object Oriented Programming*), *syntax highlighting*, *array multidimensi*, dan dapat diperluas melalui mekanisme *extension*. *PHP* merupakan salah satu bahasa *Server-Side* yang didesain khusus untuk perancangan aplikasi web dan tergolong aman untuk digunakan.

### 2.5.2. Sejarah PHP

“Rasmus Lardorf merasa kurang puas dengan sistem yang ada pada saat itu, sehingga dia menciptakan suatu model *interface* (antar muka) yang dapat digunakan untuk menampung informasi tentang para pengunjung situsnya. Pertama kali, Rasmus membuat *interface* dengan menggunakan PERL dan selanjutnya dia mengembangkan dengan menggunakan bahasa C untuk memberikan fleksibilitas pada *interface/parser* tersebut” (Sakur, 2010).

Pada mulanya, *interface* tersebut diberi nama *Personal Home Page*, yang memiliki kemampuan untuk mencatat seluruh informasi dari pengunjung situs *online*-nya. Kemudian, *interface* atau *parser* tersebut dimodifikasi dengan mendukung *database mSQL* atau *Mini Structure Query Language* dengan menggunakan *parser SQL*, pengembangan ini diberi nama *FI* (*Form Interpreter*). Kemudian *PHP/FI version 2.0* diluncurkan dan merupakan awal kelahiran dari *PHP* yang saat ini sudah mencapai versi 5.x.x. Selanjutnya *PHP* ini dikembangkan oleh tim untuk memberikan kemampuan yang seimbang dengan aplikasi lainnya.

“Saat ini, Zend menjadi pengembang utama dan telah mendistribusikan Zend Engine-nya untuk perkembangan *PHP*. Sampai sekarang, penggunaan *PHP* sudah sangat banyak karena kemudahan dan keandalannya di dalam proses pemrograman” (Saputra, 2010).

### 2.5.3. Kelebihan PHP

Saputra (2010) mengungkapkan: “dalam penggunaan *PHP* ini memiliki kelebihan atau keunggulan”, seperti berikut:

1. Mudah dipelajari, alasan tersebut menjadi salah satu alasan utama untuk menggunakan *PHP*. Karena baik pemula ataupun yang telah mahir mampu merancang suatu *website*.
2. *Multiplatform*, yang berarti *PHP* dapat digunakan ke dalam berbagai *platform OS* (*Operating System*) dan hampir semua *browser* juga mendukung *PHP*.
3. *Free* atau gratis, bersifat *open source*.
4. *PHP* memiliki tingkat akses yang cepat.
5. Didukung oleh beberapa macam *web server*.
6. Mendukung berbagai macam *database*.

### 2.5.4. Cara Kerja PHP

“PHP merupakan bahasa *Server Side Scripting*, di mana PHP selalu membutuhkan

web browser. Client tersebut akan dikirimkan ke server, kemudian server akan mengembalikan pada halaman sesuai instruksi yang diminta” (Saputra, 2012). Berikut uraian dari penjelasan di atas

1. *Server* membaca permintaan dari *client/browser*.
2. Kemudian dilanjutkan untuk mencari halaman/*page* pada *server*.
3. *Server* melakukan instruksi yang diberikan oleh *PHP* untuk melakukan modifikasi pada halaman/*page*.

Selanjutnya hasil modifikasi tersebut akan dikembalikan kepada *client/browser*.

### 2.6 Grafik User Interface (GUI) MySQL Dengan Aplikasi XAMPP

#### 1. Membuat Database di MySQL

Database MySQL berikut ini dibuat dengan program Xamp, jika menggunakan program MySQL lainnya pada umumnya hampir sama, perbedaan hanya pada posisi dan tampilanya saja. Maka untuk program selain Xampp disesuaikan.

- a) Jalankan program Apache dan MySQLnya.
- b) Buka internet explorer atau program browser lainnya.
- c) Ketik di alamatnya / address : <http://localhost/phpmyadmin/>.
- d) Buat nama database di create new database , lalu klik tombol create .



Gambar 2.6 Create Database Pada MySQL XAMPP

Ada beberapa Tab yang ada pada MySQL Xampp.

#### Gambar 2.7 Tab Pada MySQL XAMPP

##### a. Struktire

Pada tab ini terdapat table yang telah dibuat pada database, dan tempat untuk membuat table baru.

##### b. SQL

Pada tab SQL, terdapat tempat untuk memasukkan program SQL.

##### c. Search

Fungsi Search adalah tempat mencari data yang

sudah pernah disimpan dalam table atau database.

##### d. Query

Tempat untuk memasukkan program SQL ke dalam

base. Untuk itu terdapat beberapa operasi yang dapat dilakukan pada database seperti :

##### b) Import

Memanggil database yang pernah disimpan untuk dimasukkan kedalam database mysql yang diinginkan.

##### c) Operations

Pada tab operation terdapat beberapa operasi yang dapat dilakukan pada database seperti : membuat table baru, membuat comment database, mengganti nama database, dan mengcopy database.

##### d) Privileges

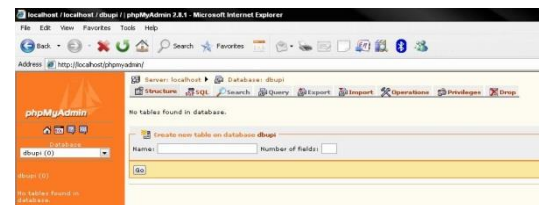
Tempat informasi user yang dapat mengakses database.

##### e) Drop

Tab untuk menghapus database yang ada.

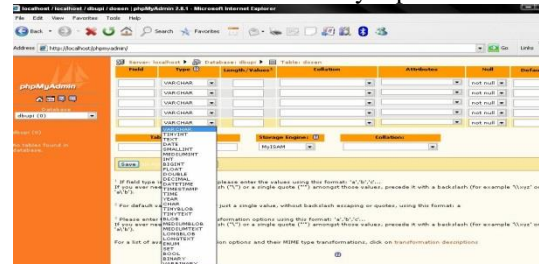
Ada tiga cara untuk membuat table di MySQL XAMPP, antara lain :

- a) Membuat tanpa program MySQL
  - Klik pada tab struktur.
  - Buat nama table yang akan anda buat pada name di create new table on database, dan jumlah kolom pada table tersebut di number of fields, dan klik tombol go.



Gambar 2.8 Membuat Table Tanpa Program MySQL

- Buat nama masing-masing field / kolom dari table yang akan dibuat, dikolom field. Pilih Tipe dari field tersebut di type, isi panjang karakter di Length/values (kecuali date, datetime, time).
- Pilih option primary key jika field tersebut primary.
- Klik tombol save untuk menyimpan.





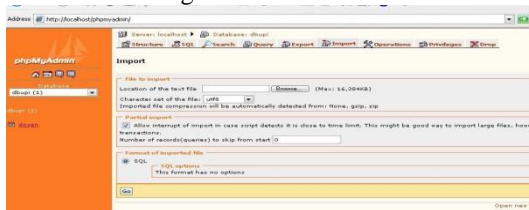
**Gambar 2.9 Mengisi Field Pada Table**

- Jika akan menambah field, maka isikan banyak field yang akan ditambah pada kolom Add, pilih posisinya dimana (diakhir table, diawal table atau setelah field tertentu). Dan klik tombol go.
- b) Membuat dengan program MySQL
  - Pilih tab SQL.
  - ketik program MySQL ditempat Run SQL query/queries on database dan klik tombol go.



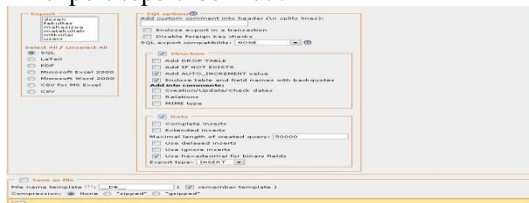
**Gambar 2.9 Membuat Table Dengan Program MySQL**

- c) Membuat table MySQL dengan file yang telah disimpan sebelumnya, dibuat di notepad atau dibuat dengan program lainnya
  - Pilih tab Import.
  - Klik tombol Browse kemudian pilih file yang disimpan tersebut.
  - Klik tombol go.



**Gambar 2.10 Membuat Table Dengan File Yang Telah Disimpan Sebelumnya**

- e) Menyimpan Database MySQL. Database yang telah dibuat di MySQL sesuai dengan cara yang diatas, sebenarnya sudah tersimpan tempat lain perlu dilakukan backup database. Adapun caranya sebagai berikut :
  - a) Pilih tab Export maka akan tampil form export seperti berikut :



**Gambar 2.11 Menyimpan Database MySQL**

- Pilih bentuk file yang akan disimpan seperti SQL, Latex, pdf, dll (agar dapat diImport lagi pada Mysql sebaiknya pilih SQL).
- Check list pada struktur dan pada data sesuai yang ingin disimpan (jika ingin sama persis dengan yang aslinya check list semuanya pada struktur dan data.

- Check list save as file dan masukkan nama file pada file name template.
- Klik tombol go.

**2.7 Sekilas Tentang Dreamweaver**

Dreamweaver merupakan perangkat lunak yang ditujukan untuk membuat suatu situs web. Versi pertama dirilis pada tahun 1997, dan sejak itu Dreamweaver menjadi web editor yang banyak digunakan oleh para web developer. Hal itu antara lain karena kemudahan dalam penggunaannya, kelengkapan fiturnya dan juga dukungannya terhadap teknologi terkini. Dreamweaver merupakan salah satu perangkat lunak yang dikembangkan oleh Macromedia Inc . Dan sekarang resmi Milik Adobe.

Setelah Dreamweaver terinstall di komputer, kita dapat mengaksesnya di menu Program Files. Untuk membukanya kita dapat membuka dengan memilih menu Start > All Programs > Macromedia > Macromedia Dreamweaver 8. Maka akan terbuka window start-up Dreamweaver

Dari window start-up Dreamweaver, terlihat 3 (tiga) kolom, yaitu *Open a Recent Item*, *Create New* dan *Create From Samples*. Pada kolom *Open a Recent Item* ditampilkan file-file yang terakhir dibuka (*recent files*). Kita dapat membuka file tersebut dengan mengklik salah satu nama file. Dan pada kolom ini juga terdapat menu *Open* yang dapat dipilih jika ingin membuka file atau dokumen yang belum ada di *recent item*.

Selanjutnya pada kolom *Create New* terdapat pilihan untuk membuat file baru. Pilihan file baru yang dapat dibuat dengan Dreamweaver antara lain file HTML, ColdFusion, PHP, ASP VBScript, Javascript, XML, CSS dan Dreamweaver Site. Pada kolom ketiga terdapat kategori *Create From Samples* dimana dapat kita pilih jika ingin membuat halaman web berdasarkan tampilan (template) yang sudah tersedia. Jadi pilihan ini dapat digunakan jika menginginkan membuat halaman web dengan cepat dan mudah.



Sumber : <http://faltines.blogspot.com/2012/03/sekilas-dreamweaver.html>

**Gambar 2.12 Halaman Awal Dreamweaver 8**

Pada halaman awal Dreamweaver 8 terdapat beberapa menu yang dapat dipilih, yaitu :

1. *Open a Recent Item*

Pada menu ini akan ditampilkan beberapa file yang sebelumnya pernah kita buka dengan menggunakan Dreamweaver 8. Atau di paling bawah ada *Open* yang dapat digunakan untuk membuka file yang lain.

2. *Create New*

Pada menu ini kita dapat memilih dokumen baru apa yang akan kita buat dengan menggunakan Dreamweaver 8. Ada banyak pilihan, diantaranya HTML, ColdFusion, PHP, ASP, JavaScript, CSS.

3. *Create From Samples*

Pada menu ini kita dapat membuat file berdasarkan contoh yang sudah diberikan oleh Dreamweaver.

**2.8 MySQL**

*MySQL* adalah salah satu jenis database server yang sangat dikenal dan digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan data. *MySQL* juga bersifat open source dan mudah digunakan (m.rudyanto arif, 2011) Karena komputer sangat unggul dalam menangani sejumlah besar data, sistem manajemen database (*Database Management Systems/DBMS*) memainkan suatu peran yang penting dalam komputasi, baik sebagai utilitas *stand-alone* maupun bagian dari aplikasi lainnya. *MySQL* adalah database server model relasional yang gratis di bawah lisensi *GNU General Public License*. Dengan sifatnya yang *open source*, memungkinkan *user* untuk melakukan modifikasi pada *source code*-nya untuk memenuhi kebutuhan spesifik mereka sendiri. *MySQL* merupakan database server multiuser dan *multi-threaded* yang tangguh (*robust*).

**2.9 UML (Unified Modelling Language)**

*Unified Modeling Language (UML)* adalah salah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi objek” (Munawar, 2005). Hal ini disebabkan karena *UML* menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

*UML* tidak hanya merupakan sebuah bahasa pemrograman visual saja, namun juga

dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti *JAVA*, *C++*, *Visual Basic*, atau bahkan dihubungkan secara langsung ke dalam sebuah *object-oriented database*.

**2.9.1. Use Case Diagram.**

*“Use Case adalah deskripsi fungsi dari sebuah sistem dari prespektif pengguna”* (Munawar, 2005). *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut *Scenario*. Setiap *Scenario* mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan demikian secara singkat bisa dikatakan *use case* adalah serangkaian *scenario* yang digabungkan bersama-sama oleh tujuan umum pengguna.

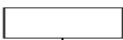
*“Model use case adalah bagian dari model requirement”* (Munawar, 2005). Termasuk disini adalah *problem domain object* dan penjelasan tentang *user interface*. *Use case* memberikan spesifikasi fungsi-fungsi yang ditawarkan oleh sistem dari prespektif *user*.

Setiap langkah dalam *use case* adalah sebuah elemen dalam interaksi antara *actor* dan sistem. Setiap langkah harus berupa pernyataan sederhana dan dengan jelas menunjukkan siapa yang menjalankan langkah tersebut. Langkah tersebut harus menunjukkan tujuan *actor*, bukan mekanisme yang harus dilakukan *actor*. *Use case diagram* menunjukkan 3 aspek dari sistem yaitu aktor, *use case* dan *system* atau *sub system boundary*. *“Aktor mewakili peran orang, system yang lain atau alat ketika berkomunikasi dengan use case. Use case adalah abstraksi dari interaksi antara system dan aktor”* (Munawar, 2005)

**2.9.2. Sequence Diagram**

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah *scenario*.

*“Simbol-simbol yang ada pada sequence diagram”* (Munawar, 2005) adalah:

1. Obyek atau l 

Sumber: Munaw

Gambar 2.10 menunjukkan pada sebuah *Sequence Diagram*

Obyek diletakkan di dekat bagian atas diagram dengan urutan dari kiri ke kanan. Setiap *participant* terhubung dengan garis titik-titik yang disebut *lifeline*. Sepanjang *lifeline* ada kotak yang disebut *activation*. *Activation* mewakili sebuah eksekusi operasi dari *participant*.

## 2. Message

Sebuah *message* bergerak dari satu *participant* ke *participant* yang lain dan dari satu *lifeline* ke *lifeline* yang lain. Sebuah *message* bisa jadi *simple*, *synchronous* atau *asynchronous*. *Message* yang *simple* adalah sebuah perpindahan (*transfer*) control dari satu *participant* ke *participant* yang lainnya. Jika sebuah *participant* mengirimkan sebuah *message synchronous*, maka jawaban atas *message* tersebut akan ditunggu sebelum diproses dengan urusannya. Namun jika *message asynchronous* yang akan dikirimkan, maka jawaban atas *message* tersebut tidak perlu ditunggu.

### 2.9.3. Collaboration Diagram

“*Collabration diagram* adalah perluasan dari obyek diagram (*Obyek diagram* menunjukkan obyek-obyek hubungannya satu dengan yang lain)” (Munawar, 2005). *Collaboration diagram* menunjukkan *message-message* obyek yang dikirimkan satu sama lain. *Collaboration diagram* adalah bentuk lain dari *sequence diagram*. Bila *sequence diagram* diorganisir menurut waktu maka *collabration diagram* diorganisir menurut ruang/ *space*.

*Collaboration diagram* merupakan asosiasi diantara obyek-obyek. Panah didekat garis asosiasi menunjukkan *message*, sedangkan *content message* ditunjukkan dengan *label*. Angka pada *message* menunjukkan urutan *message*. *Collaboration diagram* memungkinkan untuk memodelkan pengiriman sebuah *message* ke banyak obyek pada *class* yang sama. “*Demikian juga hanya untuk*

*menunjukkan adanya obyek aktif yang mengendalikan aliran dari message*” (Munawar, 2005).

## 2.9 LUPUS ERITMATOSUS SISTEM (LES)

### 2.10.1. Pengertian Lupus Eitmatosus Sistem (LES)

*Lupus eritematosus sistemik* (LES) adalah penyakit autoimun sistemik yang ditandai dengan adanya autoantibodi terhadap autoantigen, pembentukan kompleks imun, dan disregulasi sistem imun, menyebabkan kerusakan pada beberapa organ tubuh.

### 2.10.2. Etiologi

Faktor Resiko terjadinya SLE di antaranya yaitu :

#### 1. Faktor Genetik

- a. Jenis kelamin, frekuensi pada wanita dewasa 8 kali lebih sering dari pada pria dewasa.
- b. Umur, biasanya lebih sering terjadi pada usia 20-40 tahun Etnik, Faktor keturunan, dengan Frekuensi 20 kali lebih sering dalam keluarga yang terdapat anggota dengan penyakit tersebut.

#### 2. Faktor Resiko Hormon

Hormon estrogen menambah resiko SLE, sedangkan androgen mengurangi resiko ini.

#### 3. Sinar UV

Sinar Ultra violet mengurangi supresi imun sehingga terapimenjadi kurang efektif, sehingga SLE kambuh atau bertambahberat. Ini disebabkan sel kulit mengeluarkan sitokin dan prostaglandin sehingga terjadi inflamasi di tempat tersebut maupun secara sistemik melalui peredaran pembuluh darah

#### 4. Imunitas

Pada pasien SLE, terdapat hiperaktivitas sel B atau intoleransi terhadap sel T

#### 5. Obat

Obat tertentu dalam presentase kecil sekali pada pasien tertentudan diminum dalam jangka waktu tertentu dapat mencetuskan lupus obat (Drug Induced Lupus Erythematosus atau DILE). Jenis obat yang dapat menyebabkan Lupus Obat adalah :

- a. Obat yang pasti menyebabkan Lupus obat : Kloropromazin, metildopa, hidralasin, prokainamid, dan isoniazid.
  - b. Obat yang mungkin menyebabkan Lupus obat : dilantin, penisilamin, dan kuinidin. Hubungannya belum jelas : garam emas, beberapa jenis antibiotic dan griseofurvin
- #### 6. Infeksi

Pasien SLE cenderung mudah mendapat infeksi dan kadang- kadang penyakit ini kambuh setelah infeksi

#### 7. Stres

Stres berat dapat mencetuskan SLE pada pasien yang sudah memiliki kecendrungan akan penyakit ini.

### 2.10.3. Gejala Penyakit Lupus

Gejala-gejala penyakit dikenal sebagai Lupus Eritomatosus Sistemik (LES) alias Lupus. Eritomatosus artinya kemerahan. sedangkan sistemik bermakna menyebar luas ke berbagai organ tubuh. Istilahnya disebut LES atau Lupus. Gejala-gejala yang umum dijumpai adalah.

1. Kulit yang mudah gosong akibat sinar matahari serta timbulnya gangguan pencernaan.
2. Gejala umumnya penderita sering merasa lemah, kelelahan yang berlebihan, demam dan pegal-pegal. Gejala ini terutama didapatkan pada masa aktif, sedangkan pada masa remisi (nonaktif) menghilang.
3. Pada kulit, akan muncul ruam merah yang membentang di kedua pipi, mirip kupu-kupu. Kadang disebut (butterfly rash). Namun ruam merah menyerupai cakram bisa muncul di kulit seluruh tubuh, menonjol dan kadang-kadang bersisik. Melihat banyaknya gejala penyakit ini, maka wanita yang sudah terserang dua atau lebih gejala saja, harus dicurigai mengidap Lupus.
4. Anemia yang diakibatkan oleh sel-sel darah merah yang dihancurkan oleh penyakit LUPUS ini
5. Rambut yang sering rontok dan rasa lelah yang berlebihan

Contoh Penderita Penyakit Lupus Eritmatosus Sistem (LES).



**Gambar 2.17 Penyakit Lupus**

#### 2.10.4. Klasifikasi Lupus

Penyakit Lupus dapat diklasifikasikan menjadi 3 macam yaitu *discoïd lupus*, *Systemic Lupus Erythematosus*, dan lupus yang diinduksi oleh obat.

##### 1. *Discoïd Lupus*

Lesi berbentuk lingkaran atau cakram dan ditandai oleh batas eritema yang meninggi, skuama, sumbatan folikuler, dan telangiectasia. Lesi ini timbul di kulit kepala, telinga, wajah, lengan, punggung, dan dada.

Penyakit ini dapat menimbulkan kecacatan karena lesi ini memperlihatkan atrofi dan jaringan parut di bagian tengahnya serta hilangnya apendiks kulit secara menetap (Hahn, 2005).

##### 2. *Systemic Lupus Erythematosus*

SLE merupakan penyakit radang atau inflamasi multisistem yang disebabkan oleh banyak faktor (Isenberg and Horsfall, 1998) dan dikarakterisasi oleh adanya gangguandisregulasi sistem imun berupa peningkatan sistem imun dan produksi autoantibodi yang berlebihan (Albar, 2003). Terbentuknya autoantibodi terhadap dsDNA, berbagai macam ribonukleoprotein intraseluler, sel-sel darah, dan fosfolipid dapat menyebabkan kerusakan jaringan (Albar, 2003) melalui mekanisme pengaktifan komplemen (Epstein, 1998).

##### 3. Lupus yang diinduksi oleh obat

Lupus yang disebabkan oleh induksi obat tertentu khususnya pada asetilator lambat yang mempunyai gen HLA DR-4 menyebabkan asetilasi obat menjadi lambat, obat banyak terakumulasi di tubuh sehingga memberikan kesempatan obat untuk berikatan dengan protein tubuh. Hal ini direpson sebagai benda asing oleh tubuh sehingga tubuh membentuk kompleks antibodi antinuklear (ANA) untuk menyerang benda asing tersebut (Herfindal *et al.* 2000).

#### 2.10.5. Patofisiologi

Penyakit SLE terjadi akibat terganggunya regulasi kekebalan yang menyebabkan

Peningkatan autoantibodi yang berlebihan. Gangguan imunoregulasi ini ditimbulkan oleh kombinasi antara faktor-faktor genetik, hormonal ( sebagaimana terbukti oleh awitan penyakit yang biasanya terjadi selama usia reproduktif) dan lingkungan (cahaya matahari, luka bakar termal). Obat-obat tertentu seperti hidralazin, prokainamid, isoniazid, klorpromazin dan beberapa preparat antikonvulsan di samping makanan seperti kecambah alfalfa turut terlibat dalam penyakit SLE- akibat senyawa kimia atau obat-obatan.

Pada SLE, peningkatan produksi autoantibodi diperkirakan terjadi akibat fungsi sel T supresor yang abnormal sehingga timbul penumpukan kompleks imun dan kerusakan jaringan. Inflamasi akan menstimulasi antigen yang selanjutnya serangsang antibody tambahan dan siklus tersebut berulang kembali.

#### 2.10.6. Kriteria SLE

Pada tahun 1982, *American Rheumatism Association* (ARA) menetapkan kriteria baru untuk klasifikasi SLE yang

diperbarui pada tahun 1997. Kriteria SLE ini mempunyai selektivitas 96%. Diagnosa SLE dapat ditegakkan jika pada suatu periode pengamatan ditemukan 4 atau lebih kriteria dari 11 kriteria yaitu :

1. Arthritis, arthritis nonerosif pada dua atau lebih sendi perifer disertai rasa nyeri, bengkak, atau efusi dimana tulang di sekitar persendiantidak mengalami kerusakan
2. Tes ANA diatas titer normal = Jumlah ANA yang abnormalditemukan dengan immunofluoroscence atau pemeriksaan serupajika diketahui tidak ada pemberian obat yang dapat memicu ANAsebelumnya
3. Bercak Malar / Malar Rash (Butterfly rash) = Adanya eritemaberbatas tegas, datar, atau berelevasi pada wilayah pipi sekitarhidung (wilayah malar)
4. Fotosensitif bercak reaksi sinar matahari = peka terhadap sinar UV /matahari, menyebabkan pembentukan atau semakin memburuknyaruam kulit
5. Bercak diskoid = Ruam pada kulit
6. Salah satu Kelainan darah;
  - a) anemia hemolitik,
  - b) Leukosit  $< 4000/mm^3$ ,
  - c) Limfosit $<1500/mm^3$ ,
  - d) Trombosit  $<100.000/mm^3$
7. Salah satu Kelainan Ginjal;
  - a) Proteinuria  $> 0,5 g / 24 jam$ ,
  - b) Sedimen seluler = adanya elemen abnormal dalam air kemih yang berasal dari sel darah merah/putih maupun sel tubulus ginjal
8. Salah satu Serositis :
  - a) Pleuritis,
  - b) Perikarditis
9. Salah satu kelainan Neurologis;
  - a) Konvulsi / kejang,
  - b) Psikosis
10. Ulser Mulut, Termasuk ulkus oral dan nasofaring yang dapatditemukan
11. Salah satu Kelainan Imunologi
  - a) Sel LE+
  - b) Anti dsDNA diatas titer normal
  - c) Anti Sm (Smith) diatas titer normal
  - d) Tes serologi sifilis positif palsu

### 3. METODE PENELITIAN



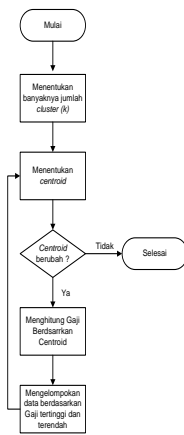
### 4. HASIL DAN PEMBAHASAN

#### 4.1 Analisa Data

Dalam mengumpulkan data dilakukan dengan cara melakukam wawancara serta mengambil data kepada administrasi kantor tersebut. Faktor-faktor yang mempengaruhi gaji karyawan banyak jenisnya, diantaranya faktor *Intern* dan faktor *Ekstern*. Namun pada penelitian ini faktor yang digunakan adalah faktor *Intern*. Faktor-faktor ini nantinya akan dikelompokkan sehingga membantu dalam memahami data gaji karyawan. Penganalisaan dilakukan berdasarkan fakta dan data yang didapat di PT Indomex Dwijaya Lestari.

#### 4.2 Analisa *Clustering* dengan Menggunakan Algoritma *K-Means*

Analisis *cluster* merupakan pengelompokan objek-objek data hanya berdasarkan pada informasi yang terdapat pada data, yang menjelaskan objek dan relasinya. Tahapan algoritma *K-Means* dapat dilihat pada gambar 4.1 :



Gambar 4.1 : Diagram Proses Flowchart K-Means

### 4.3 Penetapan Data

Sebelum melakukan proses pengelompokan data siswa, terlebih dahulu tentukan data yang akan dilakukan pengolahan agar tujuan penelitian dapat tercapai. Adapun langkah-langkahnya sebagai berikut :

### 4.4 Menentukan Atribut Yang Digunakan

Pada penelitian ini faktor yang mempengaruhi gaji karyawan tetap dan karyawan kontrak yaitu :

1. Bagian kerja karyawan
2. Status kerja karyawan
3. Gaji Pokok karyawan
4. Hari kerja karyawan
5. Total gaji karyawan

### 4.5 Pengkodean Data

Untuk membuktikan bahwa faktor rekap gaji karyawan dipengaruhi oleh faktor – faktor yang telah ditentukan seperti yang sudah dijelaskan sebelumnya maka dibutuhkan suatu alat bantu untuk membuktikannya yaitu berupa rekap gaji karyawan dengan melihat total gaji karyawan. Adapun data yang diolah yaitu karyawan tetap sebanyak 32 orang, sedangkan karyawan kontrak 28 orang, pada tabel berikut adalah data yang mewakili saja, terlihat pada tabel 4.1 :

Tabel 4.1: Sampel Data Karyawan

Tabel 4.1 : Sampel Data Awal Karyawan Tetap dan Karyawan Kontrak

No	Nama	Bagian	Status	Gaji Pokok	Jumlah Hari Kerja	Yang Diterima
1	RUDI DARMAWI	DIREKSI	TETAP	Rp. 3.000.000	25	Rp. 3.529,075
2	RIKO MAYEK	GENERAL MANAGER	TETAP	Rp. 2.500.000	25	Rp. 3.100,675
3	MAWARDI	MEKANIK	TETAP	Rp. 2.220.000	25	Rp. 2.715,075
4	MASLAN	MEKANIK	TETAP	Rp. 2.220.000	25	Rp. 2.715,075
5	MULFARI	MANAGER OP/PRO	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
6	MAWARDI K	MANAGER A/DEK	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
7	AZHARI AGUSTI	MANAGER AKUNTANSI	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
8	JULIANO	MANAGER PEMASARAN	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
9	GESTIANI BUULOLO	PERSONALIA DAN UMUM	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
10	SARDEWI	PRODUKSI // Z/AS	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
11	ANDHIKA	PENGAWAS PRODUKSI	TETAP	Rp. 2.300.000	25	Rp. 2.815,075
12	YOLA	ADM PEMBELIAN	TETAP	Rp. 2.050.000	20	Rp. 2.560,075
13	MERI	ADM PENJUALAN	TETAP	Rp. 2.050.000	20	Rp. 2.560,075
14	RIMA	ADM PEMJUALAN	TETAP	Rp. 2.050.000	20	Rp. 2.560,075
15	WULAN	ADM PENJUALAN	TETAP	Rp. 2.050.000	20	Rp. 2.560,075
16	FAUZI	SDM	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
17	RINA SIALANGAN	KASIR	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
18	MESI KOSWITA	ADM BANK	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
19	FERDIANTO	GUIDANG	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
20	MAISARIL	KA GUDANG BARANG JADI	TETAP	Rp. 2.014.000	25	Rp. 2.525,555
21	VIARA Z	ADM PIUTANG	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
22	RETHA	ADM GUDANG	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
23	OKY	ADM GUDANG	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
24	WANDI	SATPAM	TETAP	Rp. 2.050.000	25	Rp. 2.560,075

No	Nama	Bagian	Status	Gaji Pokok	Jumlah Hari Kerja	Yang Diterima
25	DODI	SATPAM	TETAP	Rp. 2.050.000	25	Rp. 2.560,075
26	OYONG	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
27	RIZKI	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
28	RUSDI	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
29	RAMHAT	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
30	DEFIS	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
31	RUDI	DRIVER	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
32	ADRIZON	SDM	TETAP	Rp. 2.014.000	20	Rp. 2.523,335
33	SLAMET	MEKANIK	KONTRAK	Rp. 1.800.725	25	Rp. 1.833,765
34	PUTRA	DRIVER	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
35	BADI	ANGGOTA GUDANG	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
36	VITRA	ANGGOTA GUDANG	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
37	HARIS	ANGGOTA GUDANG	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
38	EPI	SALES	KONTRAK	Rp. 1.800.725	18	Rp. 1.833,765
39	MADI	SALES	KONTRAK	Rp. 1.800.725	18	Rp. 1.833,765
40	HENDRA	SALES	KONTRAK	Rp. 1.800.725	18	Rp. 1.833,765
41	YUPI	SALES	KONTRAK	Rp. 1.800.725	18	Rp. 1.833,765
42	TALISI	SATPAM	KONTRAK	Rp. 2.050.000	25	Rp. 2.088,025
43	CHANI	SATPAM	KONTRAK	Rp. 2.050.000	25	Rp. 2.088,025
44	ARIF	BOLER	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
45	DONI	BOLER	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
46	IRJON	FORMILIA	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
47	AGUS	FORMILIA	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
48	SYAFRIZAL	PACKING	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
49	ADHAR	PACKING	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
50	YNSA	PACKING	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
51	DIEA	ANGGOTA PRODUKSI	KONTRAK	Rp. 2.050.000	28	Rp. 2.088,025
52	SIEKA	ANGGOTA PRODUKSI	KONTRAK	Rp. 2.050.000	28	Rp. 2.088,025
53	MAYAR	CLENING SERVICE	KONTRAK	Rp. 1.800.725	25	Rp. 1.833,765
54	LASTRI	CLENING SERVICE	KONTRAK	Rp. 1.800.725	15	Rp. 1.833,765
55	MERI	CLENING SERVICE	KONTRAK	Rp. 1.800.725	25	Rp. 1.833,765
56	ADRIZON	CLENING SERVICE	KONTRAK	Rp. 1.800.725	25	Rp. 1.833,765
57	YULENDI	SDM	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
58	ADRIZON	SDM	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
59	MAISARIL	KA GUDANG BARANG JADI	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765
60	OKY	KA GUDANG BARANG JADI	KONTRAK	Rp. 1.800.725	20	Rp. 1.833,765

### 4.6 Transformasi Data

Transformasi data dilakukan karena jenis data tidak berupa *numeric* maka data harus ditransformasikan terlebih dahulu dengan cara melakukan frekuensi pada data yang terbanyak muncul dengan mengurutkan frekuensi tertinggi ke terendah dan lakukan inisial data.

### 4.6 Proses Clustering Menggunakan Algoritma K-Means

Data yang sudah ditetapkan akan dilakukan pengolahan data dengan proses *clustering* dengan menggunakan algoritma *K-Means* sehingga didapatkanlah hasil pengelompokan terbaik yang nantinya akan membantu dalam pemahaman data karyawan.

**Tabel 4.2: Sampel Data Siswa Proses Clustering**

Tabel 4.7 Sampel Data Karyawan Proses Clustering

No	Nama	Bagian	Status	Gaji Pokok	Jumlah Hari Kerja	Yang Diterima
1	RUDI DARMAWI	1	1	1	1	3
2	RIKO MAYER	2	1	2	1	3
3	MAWARDI	3	1	3	1	2
4	MASLAN	3	1	3	1	2
5	MULFAJRI	4	1	4	1	2
6	MAWARDIK	5	1	4	1	2
7	AZHARI AGUSTI	6	1	4	1	2
8	JULIANO	7	1	4	1	2
9	GESTIANI BUULOLO	8	1	4	1	2
10	SARDEWI	9	1	4	1	2
11	ANDHIKA	10	1	4	1	2
12	YOLA	11	1	5	2	2
13	MERI	12	1	5	2	2
14	RIMA	12	1	5	2	2
15	WULAN	12	1	5	2	2
16	FAUZI	13	1	5	1	2
17	RNA SIALANGAN	14	1	5	1	2
18	MESI KOSWITA	15	1	5	1	2
19	FERDIANTO	16	1	5	1	2
20	MAISARIL	17	1	6	1	2
21	VIARA Z	18	1	5	1	2
22	RETHA	19	1	5	1	2
23	OKY	19	1	5	1	2
24	WANDI	20	1	5	1	2
25	DODI	20	1	5	1	2
26	OYONG	21	1	6	2	2
27	RICHI	21	1	6	2	2
28	RUDSDI	21	1	6	2	2
29	RAHMAT	21	1	6	2	2
30	DEFIS	21	1	6	2	2
31	RUDI	21	1	6	2	2
32	ADRIZON	13	1	6	2	2
33	SLAMET	3	2	7	1	1
34	PUTRA	21	2	7	2	1
35	BADI	22	2	7	2	1
36	VITRA	22	2	7	2	1
37	HARIS	22	2	7	2	1
38	ERI	23	2	7	3	1
39	HADI	23	2	7	3	1
40	HENDRA	23	2	7	3	1
41	YOPI	23	2	7	3	1
42	JAMRI	20	2	5	1	2
43	CHAN	20	2	5	1	2
44	ARIF	24	2	7	2	1
45	DONI	24	2	7	2	1
46	JHON	25	2	7	2	1
47	AGUS	25	2	7	2	1
48	SYAFRIZAL	26	2	7	2	1
49	AZHAR	26	2	7	2	1
50	VINA	26	2	7	2	1
51	DIKA	27	2	5	4	2
52	SISKA	27	2	5	4	2
53	MAYAR	28	2	7	1	1
54	LASTRI	28	2	7	1	1
55	MERI	28	2	7	1	1
56	ADRIZON	28	2	7	1	1
57	YULENDRI ADI	13	2	7	2	1
58	ADRIZON	13	2	7	2	1
59	MAISARIL	29	2	7	2	1
60	OKY	29	2	7	2	1

1. Penentuan jumlah *cluster*

Penentuan jumlah *cluster* dilakukan untuk mengetahui hasil dari total gaji karyawan. Maka dalam penelitian ini jumlah *cluster* yang digunakan adalah sebanyak 4 *cluster* ( $k=4$ ), sehingga nanti akan diketahui *cluster* terbaik dalam melakukan penelitian ini.

2. Menentukan *centroid*

Pusat awal *cluster* atau *centroid* ditentukan secara *random* atau acak, dimana nilai *cluster* 0 diambil dari baris ke-11, nilai *cluster* 1 pada baris ke-47. Berikut nilai titik pusat awal *cluster* (*centroid*) yang dapat dilihat pada tabel :

NO	NAMA	BAGIAN	STATUS	GAJI POKOK	JUMLAH H.K	TOTAL
11	ANDHIKA	10	1	4	1	2
47	AGUS	25	2	7	2	1

Tabel 4.7 Titik Pusat Awal Cluster

Cluster 0 : (11 ; 10 ; 1 ; 4 ; 1 ; 2 )

Cluster 1 : (47 ; 25 ; 2 ; 7 ; 2 ; 1 )

### 3. Menghitung jarak dari centroid

Menghitung jarak antara titik centroid dengan titik tiap objek dengan menggunakan *Euclidian Distance*. Ada pun penghitungan centroid awal secara manual. Perhitungannya adalah sebagai berikut:

Perhitungan dengan rumus :

$$D(i, j) = \sqrt{(X_{1i} - X_{1j})^2 + (X_{2i} - X_{2j})^2 + \dots + (X_{ki} - X_{kj})^2}$$

Perhitungan jarak dari data ke-1 terhadap pusat cluster :

$$Centroid1 = \frac{\sqrt{(1-10)^2 + (1-1)^2 + (1-4)^2 + (1-1)^2 + (2-2)^2}}{5} = 90$$

$$Centroid2 = \frac{\sqrt{(2-10)^2 + (1-1)^2 + (2-4)^2 + (1-1)^2 + (3-2)^2}}{6} = 69$$

$$Centroid3 = \frac{\sqrt{(3-10)^2 + (1-1)^2 + (3-4)^2 + (1-1)^2 + (2-2)^2}}{5} = 50$$

$$Centroid4 = \frac{\sqrt{(3-10)^2 + (1-1)^2 + (3-4)^2 + (1-1)^2 + (2-2)^2}}{5} = 50$$

$$Centroid5 = \frac{\sqrt{(4-10)^2 + (1-1)^2 + (4-4)^2 + (1-1)^2 + (2-2)^2}}{3} = 36$$

Perhitungan jarak dari data ke-2 terhadap pusat cluster :

$$Centroid1 = \frac{\sqrt{(1-25)^2 + (1-2)^2 + (1-7)^2 + (1-2)^2 + (2-1)^2}}{5} = 615$$

$$Centroid2 = \frac{\sqrt{(2-25)^2 + (1-2)^2 + (2-7)^2 + (1-2)^2 + (3-1)^2}}{6} = 560$$

$$Centroid3 = \frac{\sqrt{(3-25)^2 + (1-2)^2 + (3-7)^2 + (1-2)^2 + (2-1)^2}}{5} = 503$$

$$Centroid4 = \frac{\sqrt{(3-25)^2 + (1-2)^2 + (3-7)^2 + (1-2)^2 + (2-1)^2}}{5} = 503$$

$$Centroid5 = \frac{\sqrt{(4-25)^2 + (1-2)^2 + (4-7)^2 + (1-2)^2 + (2-1)^2}}{4} = 453$$

Alokasikan masing-masing objek ke centroid terdekat, untuk melakukan mengalokasikan objek ke dalam masing-

masing cluster dengan cara mengelompokkan berdasarkan jarak minimum objek ke pusat cluster dengan memberikan kode "1" jika hasil cluster mendekati nol. Perhitungan dilakukan terus sampai data ke-60 terhadap pusat cluster. Setelah dilakukan proses perhitungan maka akan didapatkan data seperti Tabel 4.3:

Tabel 4.3 : Hasil Perhitungan Gaji dan Pengelompokan Data Iterasi ke-1

no	NAMA	Bagian	STATUS	Gaji Pokok	JUMAH H.K	Total	centroid 1	centroid 2	centroid 3	centroid 4	centroid 5	cluster
11	ANDHIKA	10	1	4	1	2	90	69	50	50	36	1
47	AGUS	25	2	7	2	1	615	560	503	503	453	2

### 1. Melakukan Pencarian Iterasi 2

Pada pembahasan ini dilakukan pencarian iterasi 2, dengan melakukan perhitungan sebagai berikut :



Pusat cluster 1	23,361	2	6,833	2,111	1,056
Pusat cluster 2	9,25	1,083	4,667	1,25	1,958

a. Perhitungan jarak data 1 terhadap pusat cluster.

Perhitungan pusat cluster terhadap data 1 dengan penjelasan sebagai berikut:

$$D(i, j) = \sqrt{(X_{i1} - X_{1j})^2 + (X_{i2} - X_{2j})^2 + \dots + (X_{in} - X_{nj})^2}$$

Perhitungan gaji dari data ke-1 terhadap pusat cluster :

Centroid 1 =

$$\sqrt{(1 - 23,361)^2 + (1 - 2)^2 + (1 - 6,833)^2 + (1 - 2,111)^2 + (2 - 1,056)^2}$$

= 537.174

Centroid 2 =

$$\sqrt{(2 - 23,361)^2 + (1 - 2)^2 + (2 - 6,833)^2 + (1 - 2,111)^2 + (3 - 1,056)^2}$$

= 485.674

Centroid 3 =

$$\sqrt{(3 - 23,361)^2 + (1 - 2)^2 + (3 - 6,833)^2 + (1 - 2,111)^2 + (2 - 1,056)^2}$$

= 432.396

Centroid 4 =

$$\sqrt{(3 - 23,361)^2 + (1 - 2)^2 + (3 - 6,833)^2 + (1 - 2,111)^2 + (2 - 1,056)^2}$$

= 432.396

Centroid 5 =

$$\sqrt{(4 - 23,361)^2 + (1 - 2)^2 + (4 - 6,833)^2 + (1 - 2,111)^2 + (2 - 1,056)^2}$$

= 386.007

Perhitungan gaji dari data ke-2 terhadap pusat cluster :

Centroid 1 =

$$\sqrt{(1 - 9,25)^2 + (1 - 1,083)^2 + (1 - 4,667)^2 + (1 - 1,125)^2 + (2 - 1,958)^2}$$

= 81,578

Centroid 2 =

$$\sqrt{(2 - 9,25)^2 + (1 - 1,083)^2 + (2 - 4,667)^2 + (1 - 1,125)^2 + (2 - 1,958)^2}$$

= 60,828

Centroid 3 =

$$\sqrt{(3 - 9,25)^2 + (1 - 1,083)^2 + (3 - 4,667)^2 + (1 - 1,125)^2 + (2 - 1,958)^2}$$

= 41,911

Centroid 4 =

$$\sqrt{(3 - 9,25)^2 + (1 - 1,083)^2 + (3 - 4,667)^2 + (1 - 1,125)^2 + (2 - 1,958)^2}$$

= 41,911

centroid 2				
23,189	2	6,838	2,108	1,054
= 41,911				

Centroid 5 =

$$\sqrt{(4 - 9,25)^2 + (1 - 1,083)^2 + (4 - 4,667)^2 + (1 - 1,125)^2 + (2 - 1,958)^2}$$

= 28,078

Hingga seterusnya data diperhitungkan sampai data yang 60 dan hasil dari data perhitungan yang dilakukan sebagai berikut.

Tabel 4.4: Hasil Perhitungan Gaji dan Pengelompokan Data Iterasi ke-2

no	NAMA	Bagian	STATUS	Gaji Pokok	JUALAH Yang Ditrima	centroid 1					
						23,361	2	6,833	2,111	1,056	
1	DUDI DARMA	2	1	2	2	537,174	23,371	2	6,831	2,115	1,052
2	SIHO MAVER	2	1	2	1	485,674	22,038	60,828	7,799	22,038	7,799
3	MAWARDI	3	1	3	1	432,396	20,794	41,911	6,474	20,794	6,474
4	MASLAN	3	1	3	1	432,396	20,794	41,911	6,474	20,794	6,474
5	MALFARI	4	1	4	1	386,007	19,647	28,078	5,293	19,647	5,293
6	MAWARDI K	5	1	4	1	348,285	18,642	18,576	4,310	18,642	4,310
7	AZHAR AGUS	6	1	4	1	312,561	17,679	11,076	3,329	17,679	3,329
8	BEJLAND	7	1	4	1	278,840	16,699	5,576	2,362	16,699	2,362
9	GESTIANI BUK	8	1	4	1	247,118	15,720	2,076	1,442	15,720	1,442
10	SARDENY	9	1	4	1	217,396	14,744	0,576	0,760	14,744	0,760
11	SOLA	10	1	5	2	189,674	13,772	1,076	1,038	13,772	1,038
12	MERI	10	1	5	2	158,952	12,800	3,745	1,935	12,800	1,935
13	RIIMA	10	1	5	2	134,340	11,991	8,245	2,871	11,991	2,871
14	MULAN	10	1	5	2	134,340	11,991	8,245	2,871	11,991	2,871
15	HAZDI	10	1	5	2	113,840	10,670	14,245	3,774	10,670	3,774
16	ZINA SALANG	14	1	5	1	94,118	9,701	22,745	4,769	9,701	4,769
17	MESI KOSWIT	15	1	7	1	73,063	8,540	18,576	6,211	8,540	6,211
18	PERDIANTO	16	2	7	2	54,229	7,364	33,328	7,362	7,364	7,362
19	MARSAKIPIL	17	2	7	2	45,901	6,561	47,828	8,294	6,561	8,294
20	UARAZ	18	2	7	2	28,785	5,369	84,338	6,183	5,369	6,183
21	BETHA	19	2	7	2	19,063	4,369	102,838	10,140	4,369	10,140
22	DEW	19	2	7	2	19,063	4,369	102,838	10,140	4,369	10,140
23	MANDI	20	2	7	2	11,940	3,369	123,338	11,105	3,369	11,105
24	ODDI	20	2	7	2	11,940	3,369	123,338	11,105	3,369	11,105
25	CHONG	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
26	DEFI	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
27	ANDI	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
28	BAHMANAT	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
29	DEFIS	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
30	DEFI	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
31	ANDI	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
32	ANDSON	15	1	6	1	111,174	10,544	15,911	3,989	10,544	3,989
33	SUAMET	8	1	6	1	418,396	20,433	40,911	6,399	20,433	6,399
34	PULTRA	21	2	7	2	5,618	2,370	145,838	12,076	2,370	12,076
35	BADI	20	2	7	2	1,896	1,377	170,338	15,051	1,377	15,051
36	PULTRA	21	2	7	2	1,896	1,377	170,338	15,051	1,377	15,051
37	PABIS	22	2	7	2	1,896	1,377	170,338	15,051	1,377	15,051
38	DEFI	23	2	7	2	0,951	0,975	199,338	14,118	0,975	14,118
39	HADI	23	2	7	2	0,951	0,975	199,338	14,118	0,975	14,118
40	DENDRA	23	2	7	2	0,951	0,975	199,338	14,118	0,975	14,118
41	TOPY	23	2	7	2	0,951	0,975	199,338	14,118	0,975	14,118
42	MANDI	20	2	7	2	11,940	3,369	123,338	11,105	3,369	11,105
43	CHANI	20	2	7	2	11,940	3,369	123,338	11,105	3,369	11,105
44	DEP	24	2	7	2	0,481	0,677	225,338	15,011	0,677	15,011
45	DEWI	24	2	7	2	0,481	0,677	225,338	15,011	0,677	15,011
46	HON	25	2	7	2	2,729	1,621	255,838	15,995	1,621	15,995
47	DEFAERIAL	26	2	7	2	7,007	2,647	288,338	16,980	2,647	16,980
48	ALHAR	26	2	7	2	7,007	2,647	288,338	16,980	2,647	16,980
49	LINA	26	2	7	2	7,007	2,647	288,338	16,980	2,647	16,980
50	LINA	26	2	7	2	7,007	2,647	288,338	16,980	2,647	16,980
51	SISA	27	2	5	4	11,063	4,589	333,378	17,988	4,589	17,988
52	SISA	27	2	5	4	11,063	4,589	333,378	17,988	4,589	17,988
53	MAYAR	28	2	5	3	26,118	5,111	353,495	18,801	5,111	18,801
54	LASTRI	28	2	7	2	22,785	4,773	358,838	18,943	4,773	18,943
55	MERI	28	2	7	2	22,785	4,773	358,838	18,943	4,773	18,943
56	ADRIANOW	28	2	7	2	22,785	4,773	358,838	18,943	4,773	18,943
57	VALENTINI AD	13	1	6	1	111,174	10,544	15,911	3,989	10,544	3,989
58	ANDSON	15	1	6	1	111,174	10,544	15,911	3,989	10,544	3,989
59	MAISARIL	29	2	7	2	11,840	5,643	397,838	19,943	5,643	19,943
60	TOPY	29	2	7	2	11,840	5,643	397,838	19,943	5,643	19,943

centroid 1				
8,91304	1,043	4,565	1,217	2

Lakukan iterasi, kemudian tentukan posisi *centroid* baru dengan cara menghitung rata-rata dari data-data yang berada pada *centroid* yang sama.

2. Melakukan Pencarian Iterasi 3

Pada pembahasan ini dilakukan pencarian iterasi 3, dengan melakukan perhitungan sebagai berikut :

Pusatcluster 1	8,913	1,043	4,565	1,217	2
Pusatcluster 2	23,189	2	6,838	2,108	1,054

b. Perhitungan gaji data 1 terhadap pusat *cluster*.

Perhitungan pusat *cluster* terhadap data 1 dengan penjelasan sebagai berikut:

$$D_{ij} = \sqrt{(X_{1i} - X_{1j})^2 + (X_{2i} - X_{2j})^2 + \dots + (X_{ki} - X_{kj})^2}$$

Perhitungan gaji dari data ke-1 terhadap pusat *cluster* :

*Centroid 1* =

$$\sqrt{(1 - 8,913)^2 + (1 - 1,043)^2 + (1 - 4,565)^2 + (1 - 1,217)^2 + (2 - 2)^2}$$

= 75,376

*Centroid 2* =

$$\sqrt{(2 - 8,913)^2 + (1 - 1,043)^2 + (2 - 4,565)^2 + (1 - 1,217)^2 + (3 - 2)^2}$$

= 55,420

*Centroid 3* =

$$\sqrt{(3 - 8,913)^2 + (1 - 1,043)^2 + (3 - 4,565)^2 + (1 - 1,217)^2 + (2 - 2)^2}$$

= 37,463

*Centroid 4* =

$$\sqrt{(3 - 8,913)^2 + (1 - 1,043)^2 + (3 - 4,565)^2 + (1 - 1,217)^2 + (2 - 2)^2}$$

= 37,463

*Centroid 5* =

$$\sqrt{(4 - 8,913)^2 + (1 - 1,043)^2 + (4 - 4,565)^2 + (1 - 1,217)^2 + (2 - 2)^2}$$

= 24,507

Perhitungan gaji dari data ke-2 terhadap pusat *cluster* :

*Centroid 1* =

$$\sqrt{(1 - 23,189)^2 + (1 - 2)^2 + (1 - 6,838)^2 + (1 - 2,108)^2 + (1 - 1,054)^2}$$

= 529,563

*Centroid 2* =

$$\sqrt{(2 - 23,189)^2 + (1 - 2)^2 + (2 - 6,838)^2 + (1 - 2,108)^2 + (3 - 1,054)^2}$$

= 478,401

*Centroid 3* =

$$\sqrt{(3 - 23,189)^2 + (1 - 2)^2 + (3 - 6,838)^2 + (1 - 2,108)^2 + (2 - 1,054)^2}$$

= 425,455

*Centroid 4* =

$$\sqrt{(3 - 23,189)^2 + (1 - 2)^2 + (3 - 6,838)^2 + (1 - 2,108)^2 + (2 - 1,054)^2}$$

= 425,455

*Centroid 5* =

$$\sqrt{(4 - 23,189)^2 + (1 - 2)^2 + (4 - 6,838)^2 + (1 - 2,108)^2 + (2 - 1,054)^2}$$

= 379,401

Hingga seterusnya data diperhitungkan sampai data yang 60, Setelah didapatkan *cluster* baru pada iterasi ke-2 maka dilakukan *clustering* pada iterasi ke-3 yang mana hasil *cluster* tetap mengalami perubahan terlihat hasil perhitungan pada tabel 4.5 Dan hasil dari data perhitungan yang dilakukan sebagai berikut

Tabel 4.5 : Hasil Perhitungan Gaji dan Pengelompokan Data Iterasi ke-3

No/Nama	Jenis	STATUS	Gaji Pokok	UANGAN	Insentif	Benefit	Centroid 1	Centroid 2	Centroid 3	Centroid 4	Centroid 5
1	ALDO DAMBAWI	1	2	2	2	2	8121	1041	4345	1217	2
2	RENED MAFER	1	2	2	1	2	8121	1041	4345	1217	2
3	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
4	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
5	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
6	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
7	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
8	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
9	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
10	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
11	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
12	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
13	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
14	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
15	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
16	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
17	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
18	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
19	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
20	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
21	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
22	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
23	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
24	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
25	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
26	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
27	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
28	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
29	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
30	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
31	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
32	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
33	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
34	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
35	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
36	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
37	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
38	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
39	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
40	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
41	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
42	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
43	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
44	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
45	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
46	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
47	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
48	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
49	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
50	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
51	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
52	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
53	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
54	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
55	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
56	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
57	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
58	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
59	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2
60	MANSUR	1	2	2	1	2	8121	1041	4345	1217	2

Setelah dilakukan iterasi ke-3 ternyata hasilnya masih berbeda, maka lakukan

iterasi ke-4 untuk membandingkan letak posisi hasil *cluster* 1 dan *cluster* 2. Jika posisinya tidak berubah maka iterasi berikutnya tidak perlu dilakukan lagi. Untuk melakukan iterasi ke-4 maka tentukan *cluster* baru terlebih dahulu.

Didapatkan hasil *cluster* baru sebagai berikut

centroid 2				
8.591	1	4.455	1.182	2.045
centroid 2				
23	2	6.842	2.105	1.053

#### 4. Melakukan Pencarian Interasi 4

Pada pembahasan ini dilakukan pencarian interasi 4, dengan melakukan perhitungan sebagai berikut :

c. Perhitungan gaji data 1 terhadap pusat *cluster*.

Pusat <i>cluster</i> 1	8,591	1,000	4,455	1,182	2,045
Pusat <i>cluster</i> 2	23,000	2	6,842	2,105	1,053

Perhitungan pusat *cluster* terhadap data 1 dengan penjelasan sebagai berikut:

$$D_{ij} = \sqrt{(X_{1i} - X_{1j})^2 + (X_{2i} - X_{2j})^2 + \dots + (X_{ki} - X_{kj})^2}$$

Perhitungan gaji dari data ke-1 terhadap pusat *cluster* :

*centroid* 1 =

$$\sqrt{(1 - 8,591)^2 + (1 - 1)^2 + (1 - 4,455)^2 + (1 - 1,182)^2 + (2 - 2,045)^2} = 69,591$$

*Centroid* 2 =

$$\sqrt{(2 - 8,591)^2 + (1 - 1)^2 + (2 - 4,455)^2 + (1 - 1,182)^2 + (3 - 2,045)^2} = 50,409$$

*Centroid* 3 =

$$\sqrt{(3 - 8,591)^2 + (1 - 1)^2 + (3 - 4,455)^2 + (1 - 1,182)^2 + (2 - 2,045)^2} = 33,409$$

*Centroid* 4 =

$$\sqrt{(3 - 8,591)^2 + (1 - 1)^2 + (3 - 4,455)^2 + (1 - 1,182)^2 + (2 - 2,045)^2} = 33,409$$

*Centroid* 5 =

$$\sqrt{(4 - 8,591)^2 + (1 - 1)^2 + (4 - 4,455)^2 + (1 - 1,182)^2 + (2 - 2,045)^2} = 21,318$$

Perhitungan gaji dari data ke-2 terhadap pusat *cluster* :

*Centroid* 1 =

$$\sqrt{(1 - 23)^2 + (1 - 2)^2 + (1 - 6,842)^2 + (1 - 2,105)^2 + (2 - 1,053)^2} = 521,249$$

*Centroid* 2 =

$$\sqrt{(2 - 23)^2 + (1 - 2)^2 + (2 - 6,842)^2 + (1 - 2,105)^2 + (3 - 1,053)^2} = 470,460$$

*Centroid* 3 =

$$\sqrt{(3 - 23)^2 + (1 - 2)^2 + (3 - 6,842)^2 + (1 - 2,105)^2 + (2 - 1,053)^2} = 417,881$$

*Centroid* 4 =

$$\sqrt{(3 - 23)^2 + (1 - 2)^2 + (3 - 6,842)^2 + (1 - 2,105)^2 + (2 - 1,053)^2} = 417,881$$

*Centroid* 5 =

$$\sqrt{(4 - 23)^2 + (1 - 2)^2 + (4 - 6,842)^2 + (1 - 2,105)^2 + (2 - 1,053)^2} = 372,197$$

Hingga seterusnya data diperhitungkan sampai data yang 60, Setelah didapatkan *cluster* baru pada iterasi ke-3 maka dilakukan *clustering* pada iterasi n ke-4 yang mana hasil *cluster* tidak mengalami perubahan terlihat hasil perhitungan pada table 4.6 sehingga perulangan dihentikan. Dan Hasil dari data perhitungan yang dilakukan sebagai berikut.

Tabel 4.6 : Hasil Perhitungan Gaji dan Pengelompokan Data Interasi ke-4

No	Berkas	STATUS	Gaji Pokok	Jumlah Hari Kerja	Overtime	centroid 2				perhitungan 1	perhitungan 2	1010
						8.591	1	4.455	1.182			
1	1	1	1	1	1	69.591	8.542	512.249	12.811	8.542	12.811	0
2	1	1	2	1	1	50.409	7.100	470.460	21.660	7.100	21.660	0
3	1	1	3	1	1	33.409	5.760	417.881	20.442	5.760	20.442	0
4	1	1	4	1	1	21.318	4.417	372.197	18.262	4.417	18.262	0
5	1	1	4	1	1	13.146	3.024	318.197	16.308	3.024	16.308	0
6	1	1	4	1	1	8.651	2.037	300.197	17.126	2.037	17.126	0
7	1	1	4	1	1	2.775	1.669	267.197	16.340	1.669	16.340	0
8	1	1	4	1	1	0.991	0.766	208.197	15.369	0.766	15.369	0
9	1	1	4	1	1	0.409	0.345	167.197	14.394	0.345	14.394	0
10	1	1	4	1	1	2.127	1.482	160.197	14.424	1.482	14.424	0
11	1	1	5	1	1	6.775	2.852	149.301	11.121	2.852	11.121	0
12	1	1	5	1	1	12.991	3.946	138.301	11.298	3.946	11.298	0
13	1	1	5	1	1	11.991	3.548	126.301	11.298	3.548	11.298	0
14	1	1	5	1	1	29.775	4.442	126.312	10.151	4.442	10.151	0
15	1	1	5	1	1	29.391	3.442	87.312	9.911	3.442	9.911	0
16	1	1	7	1	1	47.391	8.899	67.144	8.364	8.899	8.364	0
17	1	1	7	1	1	64.116	8.008	49.000	7.000	8.008	7.000	0
18	1	1	7	1	1	79.851	6.842	36.000	6.001	6.842	6.001	0
19	1	1	7	1	1	87.775	5.888	25.000	5.001	5.888	5.001	0
20	1	1	7	1	1	117.991	4.844	18.000	4.001	4.844	4.001	0
21	1	1	7	1	1	117.991	3.844	9.000	3.001	3.844	3.001	0
22	1	1	7	1	1	118.426	11.817	9.000	3.000	11.817	3.000	0
23	1	1	7	1	1	118.426	11.817	9.000	3.000	11.817	3.000	0
24	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
25	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
26	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
27	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
28	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
29	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0
30	1	1	7	1	1	168.117	12.776	4.000	2.001	12.776	2.001	0

Tabel 4.6 Hasil Perhitungan Gaji dan Pengelompokan Data Interasi ke-4 (Lanjutan)

NAMA	BAGIAN	STATUS	GAJI POKOK	JUMLAH HARI KERJA	TOTAL
RUUDI	1	1	1	1	2
ADRIAN	1	1	1	1	2
JAMRI	1	1	1	1	2
PUTRA	1	1	1	1	2
AGUS	1	1	1	1	2
VIARA	1	1	1	1	2
WITRA	1	1	1	1	2
WULAN	1	1	1	1	2
YOLA	1	1	1	1	2
MERI	1	1	1	1	2
RIMA	1	1	1	1	2
WULAN	1	1	1	1	2
FAUZI	1	1	1	1	2
RUUDI	1	1	1	1	2
ADRIAN	1	1	1	1	2
JAMRI	1	1	1	1	2
PUTRA	1	1	1	1	2
AGUS	1	1	1	1	2
VIARA	1	1	1	1	2
WITRA	1	1	1	1	2
WULAN	1	1	1	1	2
YOLA	1	1	1	1	2
MERI	1	1	1	1	2
RIMA	1	1	1	1	2
WULAN	1	1	1	1	2
FAUZI	1	1	1	1	2
RUUDI	1	1	1	1	2
ADRIAN	1	1	1	1	2
JAMRI	1	1	1	1	2
PUTRA	1	1	1	1	2
AGUS	1	1	1	1	2
VIARA	1	1	1	1	2
WITRA	1	1	1	1	2
WULAN	1	1	1	1	2
YOLA	1	1	1	1	2
MERI	1	1	1	1	2
RIMA	1	1	1	1	2
WULAN	1	1	1	1	2
FAUZI	1	1	1	1	2

Pada interasi ke-3 dan interasi ke-4 hasil cluster tidak berubah maka perulangan tidak dilakukan lagi. Dapat ditarik kesimpulan bahwa cluster 0 terdiri 22 orang karyawan, cluster 1 terdiri dari 38 orang karyawan . Terlihat pada Tabel 4.7

NAMA	BAGIAN	STATUS	GAJI POKOK	JUMLAH HARI KERJA	TOTAL
GAJI KARYAWAN TINGGI					
RUUDI DARMAWI	1	1	1	1	2
RIKO MAYER	2	1	2	1	3
MAWARDI	3	1	3	1	2
MASLAN	3	1	3	1	2
MULFAJRI	4	1	4	1	2
MAWARDI K	5	1	4	1	2
AZHARI AGUSTI	6	1	4	1	2
JULIANO	7	1	4	1	2
GESTIANI BULOLO	8	1	4	1	2
SARDEWI	9	1	4	1	2
ANDHIKA	10	1	4	1	2
YOLA	11	1	5	2	2
MERI	12	1	5	2	2
RIMA	12	1	5	2	2
WULAN	12	1	5	2	2
FAUZI	13	1	5	1	2

JAMRI	20	2	7	2	1
CHAN	20	2	7	2	1
ARIF	24	2	7	2	1
DONI	24	2	7	2	1
JHON	25	2	7	2	1
AGUS	25	2	7	2	1
SYAFRIZAL	26	2	7	2	1
AZHAR	26	2	7	2	1
VDNA	26	2	7	2	1
DIKA	27	2	5	4	2
SISKA	27	2	5	4	2
MAYAR	28	2	5	1	1
LASTRI	28	2	7	1	1
RINA SIALANGAN	14	1	5	1	2
MESI KOSWITA	15	1	7	1	2
ADRIZON	13	1	6	1	2
SLAMET	3	1	6	1	2
YULENDRI ADI	13	1	6	1	2
ADRIZON	13	1	6	1	2
GAJI KARYAWAN RENDAH					
FERDIANTO	16	2	7	2	1
MAISASRIL	17	2	7	2	1
VIARA Z	18	2	7	2	1
RETHA	19	2	7	2	1
OKY	19	2	7	2	1
WANDI	20	2	7	2	1
DODI	20	2	7	2	1
OYONG	21	2	7	2	1
RICHI	21	2	7	2	1
RUDDI	21	2	7	2	1
RAHMAT	21	2	7	2	1
DEFIS	21	2	7	2	1
RUDI	21	2	7	2	1
PUTRA	21	2	7	2	1
BADI	22	2	7	2	1
VITRA	22	2	7	2	1
HARIS	22	2	7	2	1
ERI	23	2	7	3	1
HADI	23	2	7	3	1
HENDRA	23	2	7	3	1
YOPI	23	2	7	3	1
JAMRI	20	2	7	2	1
MERI	28	2	7	1	1
ADRIZON	28	2	7	1	1
MAISARIL	29	2	7	2	1
OKY	29	2	7	2	1

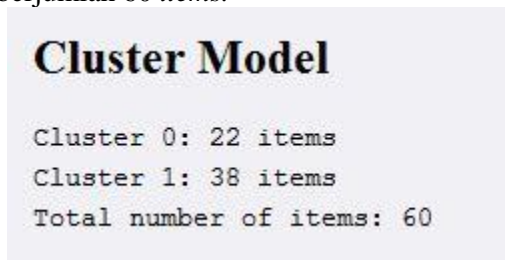
### 5. IMPLEMENTASI DAN PENGUJIAN

Setelah dilakukan penganalisaan data seperti penjelasan pada bab sebelumnya maka langkah selanjutnya adalah melakukan implementasi. Implementasi dilakukan dengan menggunakan aplikasi *RapidMiner* 6.2 yang akan membantu mempermudah dalam pengelompokan gaji karyawan tetap dan kontrak yang akan membantu dalam proses pengambilan keputusan.

#### 5.1 Hasil Pengujian Clustering

Pada *Cluster Model (Clustering)* dapat dilihat jumlah dari ke-4 cluster. Pada *Cluster Model (Clustering)* ini dapat dilihat jumlah dari ke-2 cluster adalah

cluster 1 sebanyak 22 items. Cluster 2 sebanyak 38 items. Keseluruhan data berjumlah 60 items.



**Gambar 5.1** Tampilan *Cluster Model* (Text View)

Pada *centroid table* dapat kita lihat bahwa cluster 1 merupakan kelompok karyawan yang bergaji rendah, dan cluster 2 kelompok karyawan yang bergaji tinggi. Hal ini dapat dibaca pada centroid masing-masing cluster pada *attribute* nilai rata-rata.

Attribute	cluster_0	cluster_1
Bagian	8.591	23
STATUS	1.136	1.658
Gaji Pokok	4.500	6.289
JUMLAH HAI	1.318	1.868
Yang Diterin	1.955	1.447

**Gambar 5.2** Tampilan *Centroid Table*

## 6. KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Berdasarkan analisa dan pengujian menggunakan *Software RapidMiner* pada PT Indomex Dwijaya Lestari, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Metode *Clustering K-Means* dapat membantu mengelompokkan gaji karyawan berdasarkan faktor-faktor yang mempengaruhi kinerja karyawan dalam bekerja sehingga membantu proses administrasi PT Indomex Dwijaya Lestari.
2. Didapatkan 2 kategori gaji karyawan, yaitu bergaji kategori tinggi dan bergaji kategori rendah dimana masing-masing cluster memiliki nilai rata-rata yang berbeda.

3. Penggunaan *Algoritma k-means Clustering* dapat membantu dan mempermudah dalam pencarian informasi
4. Aplikasi *RapidMiner* mampu mengatasi persoalan-persoalan dalam proses pengelompokan gaji karyawan, dari pada melakukan pencarian hasil secara manual karena kesalahan dalam proses pencarian oleh *Administrator*.

### 6.2 Saran

Untuk pengembangan dalam penelitian berikutnya maka dituliskan saran-saran sebagai berikut :

1. Faktor-faktor yang mempengaruhi rendah atau tingginya gaji karyawan dapat ditambahkan sehingga pemahaman terhadap data karyawan lebih luas lagi.
2. Pada pengembangan penelitian selanjutnya dapat dilakukan dengan jumlah data yang lebih besar lagi.
3. Aplikasi *RapidMiner* ini merupakan suatu *software* yang *userfriendly* dan dapat digunakan untuk membantu mengelompokkan data.
4. *Software RipadMiner* dimanfaatkan pada bidang yang lebih luas lagi. Karena dengan menggunakan *software* ini dalam aplikasi pekerjaan sehari-hari dapat membantu aktifitas pekerjaan .

## 1. DAFTAR PUSTAKA

- Ahmad Sanmorino. 2012. Clustering Batik Images using *Fuzzy C-Means Algorithm Based on Log-Average Luminance*. Faculty of Computer Science, Universitas Indonesia.
- Amita Verma, Ashwani kumar. 2014. *Performance Enhancement of K-Means Clustering Algorithms for High Dimensional Data sets*. International Journal of Advanced Research in Computer Science and Software Engineering Research Paper Available online at: [www.ijarcse.com](http://www.ijarcse.com)

- Angga Ginanjar Mabur, Riani Lubis. 2012. Program Studi Teknik Informatika Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia. Available online at: [www.ijarcsms.com](http://www.ijarcsms.com)
- Emerensye S. Y. Pandie. 2012. Implementasi Algoritma *data mining K-Nearset Neighbor* (K-NN) dalam pengambilan keputusan pengajuan kredit. *Jurusan Ilmu Komputer, Fakultas Sains dan Teknik, Universitas Nusa Cendana*
- J Oyelade O , O Oladipupo O, C Obagbuwa I (2010), "Penerapan algoritma *K-Means Clustering* untuk prediksi kinerja akademik mahasiswa"
- Noticewala Maitry dan Dinesh Vaghela. 2014. *Survey on Different Density Based Algorithms on Spatial Dataset. International Journal of Advance Research in Computer Science and Management Studies Research Article / Paper / Case Study*
- Nurhayati. 2014. Metode Rough Set untuk melihat perilaku suami yang menjadi akseptor KB vasetomi. Program Studi Teknik Informatika, STMIK Kaputama Binjai.
- Ramesh Singh Yadava1, P.K.Mishra2. 2012. *Performance Analysis of High Performance k-Mean Data Mining Algorithm for Multicore Heterogeneous Compute Cluster. International Journal of Information and Communication Technology Research* ISSN 2223-4985
- Sankar Rajagopal. 2011. Customer data Clustering using data mining technique. Enterprise DW/BI Consultant Tata Consultancy Services, Newark, DE, USA