

BAB II

LANDASAN TEORI

2.1. Penelitian Terdahulu

Penelitian yang dilakukan oleh Ulfa (2011), telah berhasil membuat aplikasi untuk memprediksi resiko pemberian kredit menggunakan jaringan syaraf tiruan menggunakan metode PCA (*Principal Componen Analysis*) dengan proses training mencapai 77,33%.

Penelitian yang dilakukan oleh Yohanes, Wahyudi dan Rahmi (2015), telah berhasil membuat aplikasi untuk penentuan upah minimum kota menggunakan *Backpropagation*. Pada tahap uji coba data dibagi menjadi dua bagian yaitu data latih dan data uji, dimana data latih digunakan untuk mencari jumlah iterasi, jumlah *hidden layer*, dan nilai *learning rate* yang optimal. Pengujian data latih memberikan hasil yakni jumlah iterasi optimal diperoleh pada saat iterasi 80, sedangkan untuk jumlah *hidden layer* yang optimal adalah sebanyak satu *hidden layer* dan untuk nilai *learning rate* optimal yakni pada saat bernilai 0.8. Dari hasil pengujian data uji diperoleh nilai MSE (error) sebesar 0.07280534710552478.

Penelitian yang dilakukan oleh Maulana dan Muslim (2015), telah berhasil membuat aplikasi untuk sistem prediksi tagihan listrik usaha *laundry* dengan jaringan syaraf tiruan *Backpropagation*. Data jumlah pakaian yang di proses oleh usaha *laundry* (Kg) dan data curah hujan bulanan (mm) ini dijadikan sebagai data latih. Sedangkan data tagihan listrik dijadikan data target. Sistem

prediksi tagihan listrik usaha jasa *laundry* menghasilkan prediksi dengan nilai error berdasarkan *Mean Absolute Error* terendah sebesar 3,52% atau memiliki tingkat akurasi sebesar 96,48%.

Penelitian ini dikembangkan dari penelitian terdahulu seperti tampak pada tabel 2.1. :

Tabel 2.1. Perbandingan Penelitian Terdahulu

No.	Nama	Judul Penelitian	Metode Yang Digunakan	Hasil
1	Fifi Maria Ulfa (2011)	Penerapan Jaringan Syaraf Tiruan RBF dan Metode PCA dalam Memprediksi Resiko Pemberian Kredit	Metode Analisis Komponen Utama (<i>Principal Component Analysis</i>)	Membuat aplikasi untuk memprediksi resiko pemberian kredit menggunakan jaringan syaraf tiruan menggunakan metode PCA dengan proses training mencapai 77,33%.
2	Ervin Yohannes, Wayan Firdaus Mahmudy, Asyrofa Rahmi (2015)	Penentuan Upah Minimum Kota Berdasarkan Tingkat Inflasi Menggunakan <i>Backpropagation Neural Network</i>	Metode <i>Backpropagation</i>	Hasil yang peroleh yaitu saat data uji dengan menggunakan iterasi, jumlah <i>hidden layer</i> , <i>learning rate</i> dari MSE yakni 0.07280534710552478
3	Muhamad Irvan Maulana, Much Aziz Muslim (2015)	Sistem Prediksi Tagihan Listrik Usaha Jasa <i>Laundry</i> Menggunakan Jaringan Syaraf Tiruan <i>Backpropagation</i>	Metode <i>Backpropagation</i>	Hasil yang peroleh yaitu aplikasi prediksi tagihan listrik usaha jasa <i>laundry</i> menunjukkan tingkat akurasi sebesar 96,48%

Berdasarkan penelitian – penelitian diatas, yang menjadi rujukan pada penelitian ini adalah penelitian yang dilakukan oleh Ulfa (2011). Penelitian tersebut berhasil melakukan prediksi resiko pemberian kredit melalui akurasi klasifikasi jaringan RBF (*Radial Basis Funtion*). Aplikasi yang digunakan

adalah menggunakan Matlab 7.8. Pada penelitian skripsi yang dibuat yakni dapat melakukan peringatan pengambilan kredit berdasarkan data gaji pegawai yang diterima pada tahun 2015. Proses pengambilan kredit ditentukan berdasarkan kriteria *warning* yang sudah dibuat beserta laporan *warning* kredit pegawai. Data pegawai tahun 2015 akan dilakukan *training* dan *testing* menggunakan jaringan syaraf tiruan yakni algoritma *Backpropagation*. *Software* yang digunakan adalah *JAVA NetBeans IDE 8.1*.

2.2. Gambaran Umum Tempat Penelitian

2.2.1. Sejarah SMP N 4 Kupang

Sekolah Menengah Pertama Negeri 4 Kupang adalah salah satu dari sekian banyak sekolah di Kota Kupang. Sekolah ini berdiri pada tahun 13 September 1979 dan ditandatangani oleh Menteri P & K RI. SMP N 4 berlokasi di jalan Untung Surapati No. 19 Kupang.

Perjalanan perubahan nama sekolah, tahun 1947 – 1950 bernama MILO, tahun 1950 – 1951 bernama SM/ MS, 1951 – 1958 bernama SMP Negeri Airnona, tahun 1958 – 1979 bernama SMP N 1 Kupang, 1979 – 1990 bernama SMP N 3 Kupang, dan terakhir 1990-sekarang bernama SMP N 4 Kupang Visi dan Misi UTD PMI Kupang

1. Visi

Mewujudkan karakteristik siswa yang santun dalam mutu iptek dan terampil

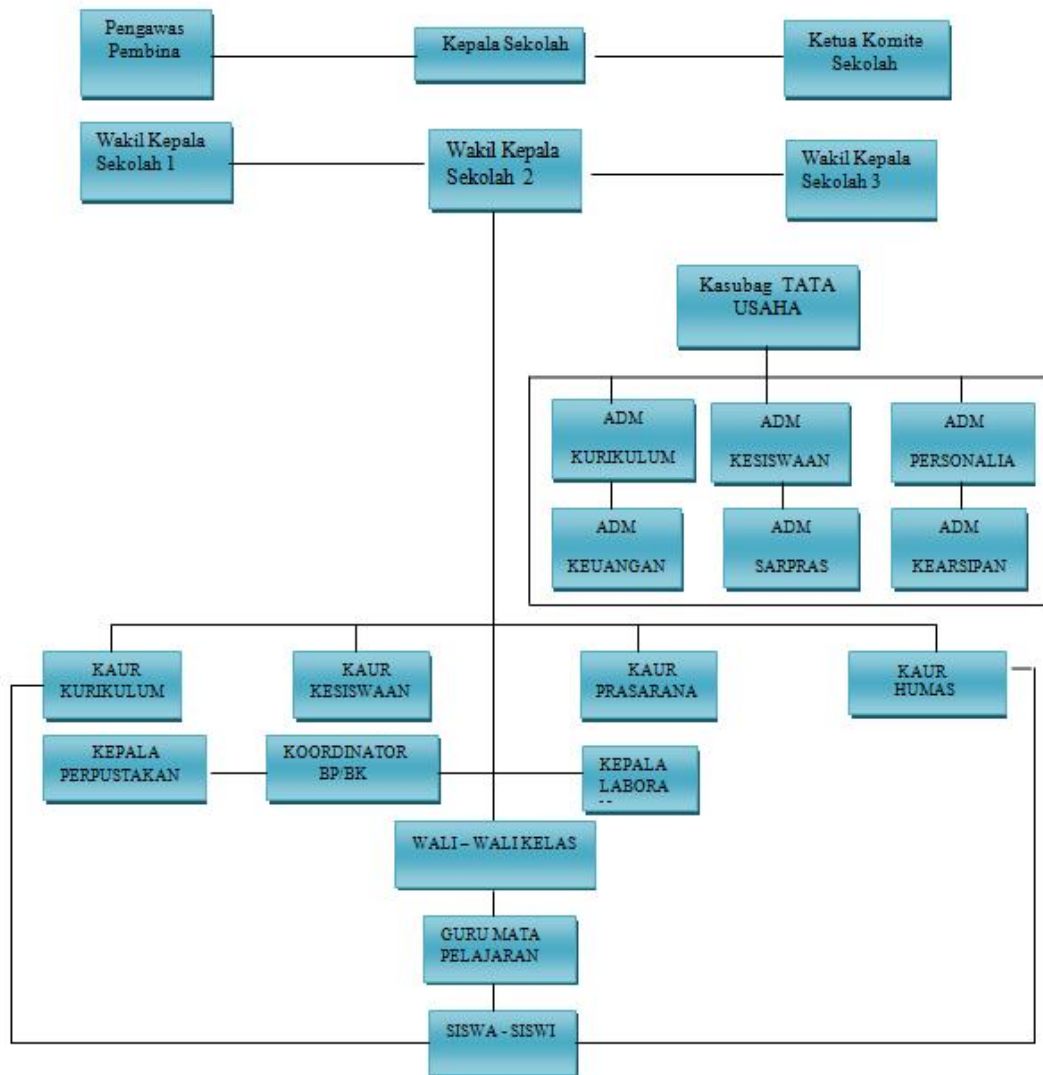
2. Misi

Mewujudkan manajemen mutu pendidikan melalui disiplin kerja, kerja cerdas, kerja tulus dan kerja ikhlas dalam pelayanan prima.

3. Tujuan

- a. Memiliki manajemen sekolah berbasis Evaluasi Diri Sekolah (EDS)
- b. Memperoleh perubahan yang lebih baik sesuai kurikulum, sistimatis, berarah, terpadu, demand driven (berdasarkan kebutuhan) dan realistik.
- c. Menumbuhkan rasa percaya diri dan saling menghargai sesama
- d. Meningkatkan kualitas layanan bagi siswa berkebutuhan khusus
- e. Menciptakan nuansa sekolah yang aman, damai dan tentram
- f. Menghasilkan output pembelajar yang berkualitas dan bermutu yang kompetitif.

2.2.2. Struktur Organisasi SMP N 4 Kupang



Gambar 2.1. Struktur Organisasi SMP N 4 Kupang

2.3. Jaringan Syaraf Tiruan

2.3.1. Pengertian Jaringan Syaraf Tiruan

Jaringan syaraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh system saraf secara biologis, seperti proses informasi pada otak manusia. Elemen kunci dari paradigma ini adalah

struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang saling berhubungan (*neuron*), bekerja serentak untuk menyelesaikan masalah tertentu.

Cara kerja JST seperti cara kerja manusia yaitu belajar melalui contoh. Sebuah JST dikonfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau aplikasi data, melalui proses pembelajaran. Belajar dalam sistem biologis melibatkan penyesuaian terhadap koneksi *synaptic* yang ada antara *neuron*. Hal ini berlaku juga untuk JST.

JST menemukan cara menyelesaikan masalah dengan sendirinya sehingga kita bisa melakukan hal-hal yang kita tidak tahu persis bagaimana melakukannya. Selain itu, kita bisa menyelesaikan masalah tanpa harus mengetahui terlebih dahulu bagaimana menyelesaikan masalah tersebut. JST dan pemrograman konvensional tidak saling bersaing, tetapi saling melengkapi. Ada masalah-masalah yang harus diselesaikan menggunakan pendekatan algoritmik seperti operasi aritmatika dan masalah-masalah yang harus diselesaikan menggunakan JST (Sutojo, 2011).

2.3.2. Karakteristik Jaringan Syaraf Tiruan

Dengan meniru sistem jaringan biologis (manusia), maka sistem jaringan saraf tiruan memiliki 3 karakteristik utama (Sutojo, 2011) yaitu :

a. Arsitektur Jaringan

Merupakan pola keterhubungan antara *neuron*. Keterhubungan *neuron - neuron* inilah yang membentuk suatu jaringan.

b. Algoritma Jaringan

Merupakan metode untuk menentukan nilai bobot hubungan. Ada dua metode pada algoritma jaringan saraf tiruan, yaitu metode bagaimana JST tersebut melakukan Pelatihan (Pembelajaran) dan metode bagaimana JST tersebut melakukan Pengenalan (Aplikasi).

c. Fungsi Aktivasi

Merupakan fungsi untuk menentukan nilai keluaran berdasarkan nilai total masukan pada *neuron*. Fungsi aktivasi suatu algoritma jaringan dapat berbeda dengan fungsi aktivasi algoritma jaringan lain.

2.3.3. *Backpropagation*

Backpropagation merupakan model *neural network* dengan banyak lapisan yang sering digunakan pada perkiraan *time series*. *Backpropagation Neural Network* mempunyai kemampuan yang kuat dari interpolasi non-linear. Ini dapat berisi pencerminan pemetaan matematis hukum internal dari data eksperimen. Metode ini secara luas digunakan dalam aplikasi teknik untuk prediksi dan optimalisasi (Yin, 2011). BPNN (*Backpropagation Neural Network*) disusun berdasarkan

algoritma *descent gradien*, yang mana diusulkan oleh Rumelhart dan McClelland. Ini adalah model jaringan banyak *layer*, yang intinya adalah BPA (*Backpropagation Algorithm*). BPNN terdiri dari *input layer*, beberapa *hidden layers* dan *output layer* (Chen, 2013).

Algoritma pelatihan/simulasi *Backpropagation* terdiri atas dua langkah, yaitu perambatan maju dan perambatan mundur. Langkah perambatan maju dan perambatan mundur ini dilakukan pada jaringan untuk setiap pola yang diberikan selama jaringan mengalami pelatihan. Jaringan *Backpropagation* terdiri atas tiga lapisan unit pengolah.

Perambatan maju dimulai dengan memberikan pola masukan ke lapisan masukan. Pola masukan ini merupakan nilai aktivasi pada unit-unit di lapisan berikutnya. Pada setiap lapisan, tiap unit pengolah melakukan penjumlahan bobot dan menerapkan fungsi sigmoid untuk menghitung keluarannya. Normalisasi data adalah suatu proses yang menentukan nilai sehingga nilai berada pada range tertentu. Proses ini bertujuan untuk menghasilkan data yang baik dan berkualitas. Metode normalisasi data adalah *minimum-maximum*. Adapun rumus matematika untuk metode ini.

$$x' = \left(0.8 x \frac{x - \text{min value}}{\text{max value} - \text{min value}} \right) + 0.1 \dots\dots\dots (1)$$

x = data

x' = hasil normalisasi

min value = nilai minimum

max value = nilai maksimum

Fungsi sigmoid digunakan untuk memenuhi beberapa syarat kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak naik turun.

Rumus fungsi aktivasi :

$$y = \frac{1}{1 + e^{-x}} \dots\dots\dots(2)$$

$$y' = f(x)(1 - f(x)) \dots\dots\dots(3)$$

Keterangan :

e = bilangan eural yaitu 2.71828

x = hasil penjumlahan dari sinyal – sinyal input

y = fungsi untuk mengaktivasi nilai x

y' = turunan dari f(x)

Yang dilakukan pada perambatan mundur adalah menghitung bobot-bobot pada semua interkoneksi. Error dihitung pada semua unit pengolah dan bobot pun diubah pada semua sambungannya. Perhitungan dimulai dari lapisan keluaran dan mundur sampai lapisan masukan. Hasil keluaran dari perambatan maju dibandingkan hasil keluaran yang diinginkan. Berdasarkan perbedaan ini, kemudian dihitung error untuk lapisan keluaran. Kemudian bobot-bobot setiap sambungan yang menuju ke lapisan keluaran disesuaikan. Setelah itu, dihitung nilai error pada lapisan tersembunyi dan hitung perubahan bobot yang menuju ke lapisan tersembunyi.

Perambatan Maju :

1. Langkah 1 : Tiap unit masukan ($x_i, i = 1, 2, 3, \dots, n$) menerima isyarat masukan x_i dan diteruskan ke unit – unit tersembunyi.
2. Langkah 2 : Tiap unit tersembunyi ($z_j, j = 1, \dots, p$) menjumlahkan isyarat masukan terbobot.

$$z_net_j = v_{j0} + \sum_{i=1}^n x_i \cdot v_{ji} \dots\dots\dots (4)$$

Keterangan :

Z_net_j = nilai untuk menghitung *hidden layer*

V_{j0} = bobot bias antara *input layer* dengan *hidden layer*

x_i = nilai input *layer* ke i

V_{ji} = bobot antara *input layer* dengan *hidden layer*

dengan menerapkan fungsi aktivasi hitung dengan :

$$z_j = f(z_net_j) = \frac{1}{1 + \exp(-z_net_j)} \dots\dots\dots (5)$$

Keterangan :

Z_j = nilai *hidden layer*

Z_net_j = nilai untuk menghitung

dan dikirim isyarat ke unit – unit keluaran.

3. Langkah 3 : Tiap unit keluaran (y_k , $k = 1, 2, \dots, m$) menjumlahkan isyarat masukan berbobot.

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j \cdot w_{kj} \dots\dots\dots(6)$$

Keterangan :

y_{net_k} = net masukan unit k

w_{k0} = nilai penimbang sambungan pada bias untuk unit y_k ($w_{k0} = 0$)

z_j = nilai aktivasi dari unit z_j

w_{kj} = nilai penimbang sambungan dari z_{ij} ke y_k

dengan menerapkan sinyal *output* fungsi aktivasi hitung :

$$y_k = f(y_{net_k}) = \frac{1}{1 + \exp(-y_{net_k})} \dots\dots\dots(7)$$

Keterangan :

y_k = unit ke – k pada lapisan keluaran

y_{net_k} = net masukan unit k

Perambatan Mundur

4. Langkah 4 : Hitung faktor tiap unit keluaran (y_k , $k = 1, 2, 3, \dots, m$) menerima target berkaitan dengan pola pelatihan masukannya. Hitung error informasi :

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \dots\dots\dots(8)$$

Keterangan :

d_k = faktor pengaturan nilai
penimbang sambungan pada
lapisan keluaran

t_k = nilai target data

y_k = unit ke – k pada lapisan keluaran

y_{net_k} = net masukan unit k

hitung koreksi perubahan bobot w_{kj}

$$\Delta w_{kj} = \alpha \cdot \delta_k \cdot z_j \dots\dots\dots(9)$$

Keterangan :

Δw_{kj} = selisih antara $w_{kj}\{t\}$ dengan $w_{kj}(t-1)$

α = learning rate $0 < \alpha < 1$

d_k = faktor pengaturan nilai penimbang
sambungan pada lapisan keluaran

z_j = nilai aktivasi dari unit z_j

5. Langkah 5 : Tiap unit tersembunyi (z_j , $k = 1, 2, \dots, m$),

dan ($j = 0, 1, \dots, p$) menghitung faktor

menjumlahkan delta masukannya (dari

unit – unit di lapisan atasnya yakni z_j).

$$\delta_{net_j} = \sum_{k=1}^m \delta_k \cdot w_{kj} \dots\dots\dots(10)$$

Keterangan :

δ_{net_j} = nilai untuk menghitung kesalahan
di *hidden layer*

d_k = faktor pengaturan nilai sambungan
pada lapisan keluaran

w_{kj} = nilai penimbang sambungan dari z_j
ke unit y_k

faktor δ layer tersembunyi :

$$\delta_j = \delta_{net_j} \cdot f'(z_{net_j}) = \delta_{net_j} \cdot z_j \cdot (1 - z_j) \dots (11)$$

Keterangan :

δ_{net_j} = nilai untuk menghitung kesalahan
di *hidden layer*

z_{net_j} = net masukan untuk unit j

z_j = nilai aktivasi dari unit z_j

hitung perubahan bobot:

$$\Delta v_{ji} = \alpha \cdot \delta_j \cdot x_i \dots (12)$$

Keterangan :

Δv_{ji} = selisih antara $v_{ij}(t)$ dengan $V_{ij}(t+1)$

α = *learning rate* $0 < \alpha < 1$

d_j = faktor pengaturan nilai penimbang
sambungan pada lapisan
tersembunyi

x_i = unit ke - i pada lapisan masukan

Perbaharui bobot dan prasikap :

6. Langkah 6 : Tiap unit keluaran ($j = 1, 2, \dots, p$)

dan ($i = 0, 1, \dots, n$)

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{jk} \dots \dots \dots (13)$$

Keterangan :

w_{kj} = nilai penimbang sambungan dari z_{ij}
ke unit y_k

Δw_{kj} = selisih antara $w_{kj}(t)$ dengan
 $w_{kj}(t+1)$

Tiap unit tersembunyi ($k = 1, 2, \dots, m$)
memperbaharui bobot – bobot dan
prasikapnya ($j = 0, 1, \dots, p$)

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \dots \dots \dots (14)$$

7. Langkah 7 : *Mean Square Error* (MSE)

$$MSE = \frac{1}{n} \sum_{k=1}^n (y_k - t_k)^2 \dots \dots \dots (15)$$

Dimana :

n = jumlah data

y_k = nilai output data

t_k = nilai target data

Dalam pengujian *Backpropagation Neural Network*, ada proses yang dinamakan denormalisasi. Denormalisasi adalah sebuah proses yang mengubah nilai hasil normalisasi ke data asli.

$$x'' = \frac{(\text{max value} - \text{min value}) * (x' - 0.1)}{0.8} + \text{min value} \dots (16)$$

Dimana :

Max value = nilai maksimum

Min value = nilai minimum

x'' = hasil denormalisasi (Heaton, 2010).

2.4. Pegawai Negeri Sipil

2.4.1. Pengertian Pegawai Negeri Sipil

Menurut UU No. 43 Tahun 1999 perubahan UU No. 8 Tahun 1974 tentang Pokok – Pokok Kepegawaian adalah :

1. Pegawai negeri adalah unsur aparatur negara, abdi negara, dan abdi masyarakat yang dengan kesetiaan dan ketaatan kepada Pancasila dan Undang-Undang Dasar 1945, negara dan pemerintah, menyelenggarakan tugas pemerintahan dan pembangunan.
2. Pegawai negeri adalah mereka yang telah memenuhi syarat-syarat yang ditentukan dalam peraturan perundang-undangan yang berlaku, diangkat oleh pejabat yang berwenang dan disertai tugas dalam sesuatu jabatan negeri atau disertai tugas negara lainnya yang ditetapkan berdasarkan sesuatu peraturan perundang-undangan dan digaji menurut peraturan perundang-undangan yang berlaku.

2.4.2. Jenis Pegawai Negeri Sipil

Pasal 2 Undang-Undang Republik Indonesia Nomor 43 Tahun 1999 Tentang Perubahan Atas Undang-Undang Nomor 8 Tahun 1974 Tentang Pokok- Pokok Kepegawaian, yang menjelaskan Pegawai Negeri terdiri dari:

1. Pegawai Negeri Sipil
2. Anggota Tentara Nasional Indonesia
3. Anggota Kepolisian Negara Republik Indonesia Pegawai Negeri Sipil terdiri dari:

1. Pegawai Negeri Sipil pusat
2. Pegawai Negeri Sipil daerah
3. Pegawai Negeri Sipil lain yang ditetapkan dengan peraturan pemerintah

2.4.3. Pengertian Gaji

Gaji adalah segala sesuatu dalam bentuk uang yang diberikan kepada karyawan/pegawai sebagai balas jasa terhadap suatu perusahaan. Sebab itu gaji adalah alat untuk memenuhi berbagai kebutuhan karyawan (Samsudin, 2010).

Gaji dapat berperan dalam meningkatkan motivasi pegawai untuk bekerja lebih efektif, meningkatkan kinerja, meningkatkan produktivitas dalam perusahaan, serta mengimbangi kekurangan dan keterlibatan komitmen yang menjadi ciri angkatan kerja masa

kini. Perusahaan yang tergolong *modern*, saat ini banyak mengaitkan gaji dengan kinerja.

2.4.4. Fungsi dan Tujuan Gaji

Gaji merupakan penghasilan yang diterima oleh seorang Pegawai Negeri Sipil dan digunakan untuk memenuhi kebutuhannya. Gaji merupakan salah satu motivasi penting yang ikut mendorong pegawai untuk berprestasi, sehingga tinggi rendahnya gaji yang diberikan akan mempengaruhi kinerja dan kesetiaan karyawan, maka gaji memiliki fungsi dan tujuan bagi pegawai.

Fungsi gaji bukan hanya membantu manajer personalia dalam menentukan gaji yang adil dan layak saja, tetapi masih ada fungsi-fungsi yang lain antara lain :

1. Untuk menarik pekerja yang mempunyai kemampuan ke dalam organisasi.
2. Untuk mendorong pekerja agar menunjukkan prestasi yang tinggi.
3. Untuk memelihara prestasi pekerja selama periode yang panjang

Selain itu adapun tujuan dari penggajian antara lain :

1. Ikatan kerja sama

Dengan pemberian gaji terjalinlah ikatan kerja sama formal antara majikan dengan karyawan. Karyawan harus mengerjakan

tugas - tugasnya dengan baik, sedangkan pengusaha atau majikan wajib membayar gaji sesuai dengan perjanjian yang disepakati.

2. Kepuasan kerja

Dengan balas jasa, karyawan akan dapat memenuhi kebutuhan-kebutuhan fisik, status sosial, dan egoistiknya sehingga memperoleh kepuasan kerja dari jabatannya (Hasibuan, 2010).

2.4.5. Gaji Diterima (*Take Home Pay*)

Take home pay berbeda dengan gaji. *Take home pay* adalah jumlah nilai keseluruhan gaji/upah dan dikurangi jika ada potongan atas kewajiban pegawai.

Makna *take home pay* yakni pembayaran yang benar-benar diterima oleh pegawai setelah menambahkan pendapatan-pendapatan rutin maupun isidentil yang merupakan hak pegawai dikurangi dengan hal-hal yang sudah diatur oleh pemerintah dan kebijakan dari perusahaan dimana pegawai tersebut bekerja.

Pada dasarnya prinsip penentuan gaji sudah memikirkan faktor seimbang, kelayakan, angka kewajaran dan selalu diusahakan agar bisa menjadi sumber untuk memotivasi bagi pegawai untuk bekerja sesuai hak dan kewajibannya.

2.4.6. Pengkreditan

Menurut Undang-Undang Republik Indonesia Nomor 10 Tahun 1998 tentang Perubahan Atas Undang – Undang Nomor 7

Tahun 1992 tentang perbankan pasal 21 ayat 11 dijelaskan kredit adalah penyediaan uang atau tagihan yang dapat dipersamakan dengan itu, berdasarkan persetujuan atau kesepakatan pinjam meminjam antara bank dengan pihak lain yang mewajibkan pihak peminjam melunasi utangnya setelah jangka waktu tertentu dengan pemberian bunga.

Kredit dilihat dari Tujuan Penggunaan :

- a. Kredit Investasi, merupakan kredit yang diberikan oleh bank kepada debitur untuk pengadaan barang-barang modal yang mempunyai nilai ekonomis lebih dari satu tahun.
- b. Kredit Modal Kerja, merupakan kredit yang digunakan untuk memenuhi kebutuhan modal kerja yang biasanya habis dalam satu siklus usaha.
- c. Kredit Konsumtif, merupakan kredit yang diberikan kepada nasabah untuk membeli barang dan jasa untuk keperluan pribadi dan tidak untuk digunakan keperluan usaha (Ismail, 2010).

2.5. Java

2.5.1. Sejarah Java

Java adalah sebuah bahasa pemrograman yang dikeluarkan oleh perusahaan *Sun Microsystems*. Sejarah bahasa pemrograman *Java* dimulai sejak tahun 1991, yaitu ketika sebuah proyek perusahaan *Sun Microsystems* dengan nama “*The Green Project*”.

Pelopop proyek ini adalah James Googling dan Patrick Naughton, Mike Sheridan, dan Bill Joy, beserta sembilan pemrogram lainnya dari perusahaan *Sun Microsystems*.

Tim ini ingin mendesain sebuah bahasa pemrograman komputer yang berukuran kecil yang dapat digunakan untuk peralatan elektronika konsumen seperti *switchboxes* TV kabel. Dikarenakan peralatan-peralatan ini menggunakan konsumsi daya dan *memory* yang rendah, maka bahasa pemrograman tersebut harus berukuran sangat kecil. Juga karena setiap vendor menggunakan CPUs (*Central Processing Unit*) yang berbeda, maka bahasa tersebut harus bersifat *multiplatform*, tidak terikat hanya pada satu arsitektur (*Architecture Neutral*). Pada tahun 1995, nama Oak diganti dengan nama *java* karena ada produk lain yang telah mematenkan nama tersebut. Pemberian nama *java* merupakan diambil dari nama sejenis kopi favorit James Gosling bersamaan dengan lahirnya nama *Java* ini, Sun juga mengganti nama *web-browser* mereka dari *Web Runner* menjadi *HotJava*. Sun kemudian mengumumkan bahasa *Java* sebagai bahasa yang mampu menjalankan *Web* secara interaktif dan aman.

Insinyur-insinyur Sun yang tergabung dalam proyek ini berlatar belakang sistem operasi *Unix*. Sehingga mereka mendasari bahasa pemrograman mereka dengan *C++* dari pada *Pascal*. Secara

husus mereka membuat bahasa mereka berorientasi obyek (*object oriented*) (Shalahuddin, 2010).

2.5.2. Pengertian *Java*

Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh *Sun Microsystems* sejak tahun 1991. Bahasa ini dikembangkan dengan model yang mirip dengan bahasa C++ dan *Smalltalk*, namun dirancang agar lebih mudah dipakai dan *platform independent*, yaitu dapat dijalankan di berbagai jenis sistem operasi dan arsitektur komputer.

Java bukan sekedar bahasa pemrograman, tetapi merupakan suatu sistem *platform* yang disediakan oleh *java* untuk membangun suatu sistem baik yang berskala kecil yakni *game*, *workstation program* sampai ke *mobile device*.

Microsystems mendefinisikan *java* yang dikategorikan menjadi tiga bagian, yaitu:

1. J2SE (*Java 2 Platform Standard Edition*) yang berfungsi sebagai pembuatan aplikasi-aplikasi dekstop dan *applet*.
2. J2EE (*Java 2 Platform Enterprise Edition*) yang dirancang untuk aplikasi bisnis dari *server*.
3. J2ME (*Java 2 Platform Micro Edition*) yang berfungsi sebagai pembuat aplikasi-aplikasi yang dapat dijalankan di lingkungan perangkat-perangkat mikro seperti *handphone* dan PDA (Irawan, 2013).

Tidak hanya *applet*, *Java* juga dapat membuat berbagai aplikasi GUI (*Graphical User Interface*) yang dapat berjalan pada berbagai *platform* (sistem operasi manapun), termasuk *Microsoft Windows* dan *Linux X-Windows*. Untuk keperluan ini, *Java* telah menyediakan kumpulan *class* di dalam *Java Foundation Class* (JFC), dengan *user interface* yang dinamakan SWING (Irawan, 2013).

2.6. Basis Data

Basis data (*database*) diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Prinsip utama basis data adalah pengaturan data dengan tujuan utama fleksibilitas dan kecepatan dalam pengambilan data kembali. Adapun tujuan basis data diantaranya meliputi *speed*, *space and accuracy*, menangani data dalam jumlah besar, kebersamaan pemakaian dan meniadakan duplikasi (Yakub, 2012).

2.6.1. Manfaat Basis Data

Beberapa manfaat basis data adalah untuk kecepatan dan kemudahan, efisien ruang penyimpanan, keakuratan, ketersediaan, kelengkapan, keamanan, dan kebersamaan. Berikutnya dipaparkan :

- a. Kecepatan dan kemudahan, pemanfaatan basis data memungkinkan untuk dapat menyimpan, mengubah dan menampilkan kembali data tersebut dengan lebih cepat dan mudah.
- b. Efisiensi ruang penyimpanan, dengan basis data efisiensi/optimalisasi penggunaan ruang penyimpanan dapat

dilakukan karena penekanan jumlah redudansi data, baik dengan jumlah pengkodean.

- c. Keakuratan, pembentukan relasi antar data bersama dengan penerapan aturan tipe, domain dan keunikan data dapat diterapkan dalam sebuah basis data.
- d. Ketersediaan, dapat memilih data utama, transaksi, data histori hingga data kadaluarsa.
- e. Kelengkapan, lengkap atau tidaknya data dalam sebuah basis data bersifat relatif.
- f. Keamanan, untuk menentukan siapa-siapa yang berhak menggunakan basis data berserta objek-objek didalamnya dan menentukan jenis-jenis operasi apa saja yang boleh dilakukan.
- g. Kebersamaan pemakai, basis data dapat digunakan oleh beberapa pemakai dan beberapa lokasi. Basis data yang dikelola oleh sistem yang mendukung *multi user* dapat memenuhi kebutuhan, akan tetapi harus menghindari inkonsistensi data (Yakub, 2012).

2.7. MySQL

MySQL merupakan RDBMS atau *server database* yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat diakses oleh banyak *user* (Raharjo, 2011).

MySQL dibuat dan dikembangkan oleh MySQL AB yang berada di Swedia. MySQL dapat digunakan untuk membuat dan mengelola *database* beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan,

mengubah dan menghapus data yang berada dalam *database*. *MySQL* merupakan sistem manajemen *database* yang bersifat relational. Artinya data-data yang dikelola dalam *database* akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat. *MySQL* dapat digunakan untuk mengelola *database* mulai dari yang kecil sampai dengan yang sangat besar. *MySQL* juga dapat menjalankan perintah-perintah *Structured Query Language* (SQL) untuk mengelola *database-database* yang ada di dalamnya.

2.8. UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat membuat model untuk semua jenis aplikasi piranti lunak dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun (Sugiarti, 2013).

2.8.1. *Use Case Diagram*

Diagram *use case* adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar, dan menjelaskan sistem secara fungsional yang terlihat *user*. Sebuah *use case* mempresentasikan

sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, dan sebagainya.

Sebuah *use case* dapat dipanggil (*include*) oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat *meng-extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain (Sugiarti, 2013).

2.8.2. Activity Diagram

Activity diagram menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana aktifitas berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu, *activity diagram* tidak menggambarkan *behaviour internal* sebuah sistem secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktifitas dari level atas secara umum. Sebuah aktifitas dapat direalisasikan oleh satu *use case* atau lebih. Aktifitas menggambarkan proses yang berjalan, sementara *use case*

menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktifitas.

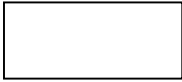
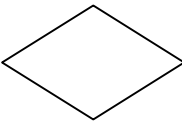
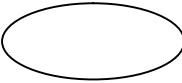
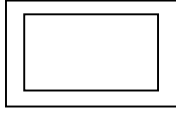
Activity diagram digambar menggunakan segiempat dengan sudut membulat untuk menggambarkan aktifitas. *Decision* untuk menggambarkan *behaviour* pada kondisi tertentu. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktifitas tertentu (Satzinger, 2010).

2.9. ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah suatu komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut yang mempresentasikan seluruh fakta dari dunia nyata yang ditinjau. *Entity Relationship Diagram* menggambarkan data dan hubungan antar data secara global dengan menggunakan *Entity Relationship Diagram* (Jogiyanto, 2010).

Berikut simbol – simbol pada *Entity Relationship Diagram* sebagai berikut:

Tabel 2.3. Simbol - simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Nama	Keterangan
		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <i>field-field</i> nya dipergunakan dalam aplikasi program.
2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya.
3		Atribut	Atribut merupakan karakteristik dari sebuah entitas.
4		Garis Relasi	Menunjukkan hubungan antar entitas.
5		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat.

1. Entitas adalah segala sesuatu yang dapat digambarkan oleh data. Entitas juga dapat diartikan sebagai individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Ada dua macam entitas yaitu entitas kuat dan entitas

lemah. Entitas kuat merupakan entitas yang tidak memiliki ketergantungan dengan entitas lainnya. Contohnya entitas anggota. Sedangkan entitas lemah merupakan entitas yang kemunculannya tergantung pada keberadaan entitas lain dalam suatu relasi.

2. Atribut merupakan pendeskripsian karakteristik dari entitas. Atribut digambarkan dalam bentuk lingkaran atau *elips*. Atribut yang menjadi kunci entitas atau *key* diberi garis bawah.
3. Relasi atau hubungan menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.