



**SISTEM PENDUKUNG KEPUTUSAN PENILAIAN PEGAWAI  
BERPRESTASI MENGGUNAKAN METODE *WEIGHTED*  
*PRODUCT* BERBASIS DESKTOP**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh  
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi  
Universitas Pembangunan Panca Budi  
Medan

**SKRIPSI**

**OLEH:**

**NAMA : TOHIRUDDIN AMRŶ SIREGAR**  
**NPM : 1624370802**  
**PROGRAM STUDI : SISTEM KOMPUTER**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PEMBANGUNAN PANCA BUDI  
MEDAN  
2020**

## ABSTRAK

**TOHIRUDDIN AMRY SIREGAR**  
**Sistem Pendukung Keputusan Penilaian Pegawai Berprestasi Menggunakan**  
**Metode *Weighted Product* Berbasis Desktop**  
**2019**

Pegawai merupakan seorang pekerja yang ditetapkan dalam suatu kantor. Setiap kantor dapat memiliki banyak pegawai. Tetapi dalam proses kerja, tidak semua pegawai memiliki kinerja yang seperti diinginkan perusahaan. Ada yang gesit dan ada juga pegawai yang tidak mengalami peningkatan. Untuk memicu kinerja pegawai, perusahaan dapat melakukan dengan beberapa cara termasuk memilih pegawai berprestasi dan memberikan pegawai tersebut imbalan atas kinerja yang dilakukannya selama bekerja di perusahaan. Metode *Weighted Product* dapat membantu perusahaan dalam menentukan pegawai yang berprestasi dalam melakukan peningkatan kinerja perusahaan. Pegawai-pegawai terbaik dapat dipilih dan ditempatkan pada divisi-divisi tertentu agar dapat memicu kegiatan perusahaan tersebut agar menjadi lebih baik. Metode *Weighted Product* terbukti dapat menentukan pegawai berprestasi pada perusahaan.

**Kata Kunci:** pegawai, perkantoran, WP, SPK

## DAFTAR ISI

<b>KATA PENGANTAR</b> .....	<b>i</b>
<b>DAFTAR ISI</b> .....	<b>ii</b>
<b>DAFTAR GAMBAR</b> .....	<b>iv</b>
<b>DAFTAR TABEL</b> .....	<b>v</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
<b>BAB II LANDASAN TEORI</b> .....	<b>5</b>
2.1 Sistem.....	5
2.1.1 Elemen Dalam Sistem .....	6
2.1.2 Klasifikasi Sistem.....	7
2.1.3 Jenis Sistem .....	7
2.2 Pengambilan Keputusan.....	8
2.3 Sistem Pendukung Keputusan.....	9
2.3.1 Tujuan Sistem Pendukung Keputusan.....	10
2.3.2 Karakteristik Sistem Pendukung Keputusan .....	11
2.3.3 Kriteria atau Ciri-ciri Pengambilan Keputusan .....	12
2.3.4 Karakteristik Sistem Pendukung Keputusan .....	12
2.3.5 Tahapan Sistem Pendukung Keputusan .....	13
2.3.6 Proses Pengambilan Keputusan.....	14
2.4 Pemilihan Pegawai .....	16
2.5 Manajemen Model .....	17
2.5.1 Basis Model .....	17
2.5.2 Sistem Manajemen Basis Model .....	18
2.5.3 Model Direktori .....	18
2.6 Metode <i>Weighted Product (WP)</i> .....	18
2.7 Flowchart .....	21
2.7.1 Sistem Flowchart .....	22
2.7.2 Program Flowchart .....	22
2.7.3 Flow Direction Symbols.....	24
2.7.4 Simbol Flowchart .....	25
2.7.5 Input-Output Symbols .....	26
2.8 <i>Unified Modelling Language</i> .....	27
2.8.1 <i>Use Case Diagram</i> .....	28
2.8.2 <i>Activity Diagram</i> .....	30
2.8.3 <i>Class Diagram</i> .....	31
2.9 Database .....	32
2.10 Tipe Data.....	34

2.10.1	Tipe Data Numerik.....	34
2.10.2	Tipe Data String.....	35
2.10.3	Tipe data tanggal.....	36
2.11	Visual Basic.NET.....	36
<b>BAB III METODE PENELITIAN.....</b>		<b>41</b>
3.1	Tahapan Penelitian.....	41
3.2	Sistem Yang Berjalan.....	44
3.3	Sistem Yang Diusulkan.....	44
3.4	Analisis Prosedur Pada Sistem Yang Diusulkan.....	44
3.5	Prosedur Penilaian Jasa.....	45
3.6	Rancangan Penelitian.....	45
	3.6.1 <i>Use Case Diagram</i> .....	45
	3.6.2 <i>Activity Diagram</i> .....	46
	3.6.3 Flowchart.....	48
3.7	Perancangan Antarmuka.....	49
	3.7.1 Rancangan Menu Beranda.....	49
	3.7.2 Rancangan Menu <i>Weighted Product</i> .....	50
	3.7.3 Rancangan Menu Info.....	51
	3.7.4 Rancangan Menu About.....	52
3.8	Perancangan Kriteria.....	52
3.9	Analisis Metode <i>Weighted Product</i> .....	54
	3.9.1 Kriteria, Nilai dan Bobot.....	54
	3.9.2 Penyelesaian Perhitungan.....	55
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>		<b>58</b>
4.1	Kebutuhan Sistem.....	58
	4.1.1 Perangkat Lunak.....	58
	4.1.2 Perangkat Keras.....	59
4.2	Implementasi Sistem.....	59
	4.2.1 Hasil Tampilan Beranda.....	60
	4.2.2 Hasil Tampilan Menu Info.....	60
	4.2.3 Hasil Tampilan Menu Profil.....	61
	4.2.4 Halaman Menu <i>Weighted Product</i> .....	62
<b>BAB V PENUTUP.....</b>		<b>64</b>
5.1	Kesimpulan.....	64
5.2	Saran.....	64

## DAFTAR PUSTAKA

## KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, karena dengan berkat dan kasih anugerah-Nya penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini sebagaimana mestinya. Judul skripsi ini adalah “**Sistem Pendukung Keputusan Penilaian Pegawai Berprestasi Menggunakan Metode *Weighted Product* Berbasis Desktop**”. Dalam kesempatan ini, penulis mengucapkan rasa terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini. Penulis ingin mengucapkan terima kasih kepada:

1. Orang tua saya yang selalu memberikan semangat, dukungan dan motivasi dalam penyusunan skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E., M.M. selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Bapak Ir. Bhakti Alamsyah, M.T., Ph.D., selaku Rektor I Universitas Pembangunan Panca Budi Medan.
4. Bapak Hamdani, S.T., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
5. Bapak Eko Hariyanto, S.Kom., M.Kom., selaku Ketua Program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
6. Bapak Andysah Putera Utama Siahaan, S.Kom., M.Kom., selaku Dosen Pembimbing I yang telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini.
7. Bapak Subhan Hartanto, S.Kom., M.Kom., selaku Dosen Pembimbing II yang telah memberikan ilmu pengetahuan, serta bimbingan dalam penyelesaian skripsi ini.
8. Dosen-dosen pada Program Studi Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
9. Seluruh staff dan karyawan pada Universitas Pembangunan Panca Budi Medan.
10. Seluruh teman-teman penulis dari program studi Sistem Komputer, Fakultas Sains dan Teknologi, Universitas Pembangunan Panca Budi, Medan

Penulis juga menyadari bahwa penyusunan skripsi ini belum sempurna baik dalam penulisan maupun isi disebabkan keterbatasan kemampuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang sifatnya membangun dari pembaca untuk kesempurnaan isi skripsi ini.

Medan, 16 Juni 2020  
Penulis

Tohiruddin Amry Siregar  
1624370802

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pengertian pegawai adalah orang pribadi yang bekerja pada pemberi kerja, berdasarkan perjanjian atau kesepakatan kerja baik secara tertulis maupun tidak tertulis, untuk melaksanakan suatu pekerjaan dalam jabatan atau kegiatan tertentu dengan memperoleh imbalan yang dibayarkan berdasarkan periode tertentu, penyelesaian pekerjaan, atau ketentuan lain yang ditetapkan pemberi kerja, termasuk orang pribadi yang melakukan pekerjaan dalam jabatan negeri.

Sistem Pendukung Keputusan adalah bagian dari sistem informasi berbasis komputer yang digunakan untuk mendukung pengambilan keputusan pada suatu organisasi atau perusahaan. Sistem pendukung keputusan digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur.

Metode *Weighted Product* adalah salah satu metode dalam sistem pengambilan keputusan dimana pengambilan sebuah keputusan dapat dilakukan secara lebih cepat dan tepat, sesuai dengan kriteria yang diinginkan atau setidaknya mendekati kriteria yang diinginkan. Alternatif-alternatif pilihan yang diharapkan dapat memberikan daftar referensi kepada pembuat keputusan sebelum benar-benar mengambil suatu keputusan akhir. Pengambilan keputusan dilakukan dengan pendekatan sistematis terhadap permasalahan melalui proses pengumpulan keputusan data menjadi informasi serta ditambah dengan faktor-faktor yang perlu

dipertimbangkan dalam pengambilan keputusan. Pengambilan keputusan merupakan proses pemilihan alternatif tindakan untuk mencapai tujuan atau sasaran tertentu.

Karena permasalahan pegawai berprestasi yang belum ditemukannya proses perhitungannya, maka penulis mencoba meneliti masalah pegawai berprestasi dengan metode *Weighted Product*. Sehingga diharapkan nantinya dapat memberikan rekomendasi terbaik untuk proses pegawai berprestasi tersebut. Dari permasalahan yang telah diuraikan, maka penulis memilih judul “**Sistem Pendukung Keputusan Penilaian Pegawai Berprestasi Menggunakan Metode *Weighted Product***”.

## 1.2 Rumusan Masalah

Meninjau dari pokok permasalahan yang telah diuraikan pada latar belakang masalah di atas dapat diambil rumusan sebagai berikut:

1. Bagaimana menentukan kriteria penilaian pegawai berprestasi sesuai dengan kinerja pegawai?
2. Bagaimana menerapkan metode *Weighted Product* untuk menentukan pegawai berprestasi?

### 1.3 Batasan Masalah

Agar masalah yang dibahas tidak menjadi luas, maka penulis membuat batasan masalah sebagai berikut:

- 1 Jumlah kriteria yang digunakan adalah sebanyak lima kriteria yaitu Absen, Komunikasi, Penampilan, Hasil Kerja dan Loyalitas.
- 2 Program aplikasi yang digunakan adalah Microsoft Visual Basic.Net 2010 berbasis *desktop* (tidak *online*).
- 3 Jumlah data yang digunakan adalah sebanyak 7 data.
- 4 Data yang digunakan merupakan data simulasi.

### 1.4 Tujuan Penelitian

Adapun tujuan penelitian yang dilakukan berdasarkan latar belakang yang sudah dibahas antara lain:

1. Untuk membuat sistem pendukung keputusan untuk menilai pegawai berprestasi.
2. Mengimplementasikan metode *Weighted Product* dalam sistem pendukung keputusan untuk menilai pegawai berprestasi.

### 1.5 Manfaat Penelitian

Adapun manfaat penelitian yang diharapkan berdasarkan latar belakang yang sudah dibahas antara lain:

1. Menghemat biaya dan waktu yang digunakan dalam menyeleksi pegawai.
2. Memberi pengetahuan tentang penerapan metode *Weighted Product*.

3. Membantu proses pemilihan pegawai berprestasi berdasarkan kategori perusahaan kepada pegawai.
4. Dapat menambah ilmu metode *Weighted Product* dalam membantu menentukan kinerja berprestasi berdasarkan kategori perusahaan.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Sistem**

Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau *variable* yang terorganisir, saling berinteraksi, terpadu dan saling tergantung satu sama lain. Teori sistem secara umum yang pertama kali diuraikan oleh *Kenneth Boulding* terutama menekankan pentingnya perhatian terhadap setiap bagian yang membentuk sebuah sistem (Yakub, 2012). Konsep lain yang terkandung di dalam definisi tentang sistem adalah konsep sinergi. Konsep ini mengandaikan bahwa di dalam suatu *system output* dari suatu organisasi diharapkan lebih besar dari pada *output* individual atau *output* masing-masing bagian. Sistem merupakan kumpulan elemen yang saling berkaitan yang bertanggung jawab memproses masukan (*input*) sehingga menghasilkan keluaran (*output*) (Kusrini, 2007).

Suatu sistem adalah kumpulan elemen atau komponen yang diorganisasikan untuk tujuan bersama. Sistem tersebut terkadang menggambarkan organisasi atau rencana itu sendiri dan kadang-kadang menggambarkan bagian-bagian dalam sistem. Sistem komputer terdiri dari komponen perangkat keras yang telah dipilih dengan cermat sehingga sistem tersebut bekerja dengan baik bersama-sama dan komponen perangkat lunak atau program yang berjalan di komputer. Komponen perangkat lunak utama itu sendiri merupakan sistem operasi yang mengelola dan

menyediakan layanan untuk program lain yang dapat dijalankan di komputer (Jogiyanto, 2016).

### **2.1.1 Elemen Dalam Sistem**

Elemen pembentuk suatu sistem dapat dibagi menjadi tujuh bagian, yaitu:

1. Tujuan, sistem dibuat untuk mencapai tujuan (output) tertentu yang ingin dicapai.
2. Masukan, semuanya yang masuk ke dalam sistem akan diproses, baik itu obyek fisik maupun abstrak.
3. Proses, yaitu transformasi dari masukan menjadi keluaran yang lebih memiliki nilai, misalnya produk atau informasi. Namun juga bisa dapat berupa hal yang tak berguna, misalnya limbah.
4. Keluaran, ini adalah hasil dari pemrosesan dimana wujudnya bisa dalam bentuk informasi, saran, cetakan laporan, produk, dan lain-lain.
5. Batas, sesuatu yang memisahkan antara sistem dan daerah di luar sistem. Dalam hal batas akan menentukan konfigurasi, ruang lingkup, dan hal-hal lainnya.
6. Pengendalian dan Umpan Balik, mekanismenya dapat dilakukan dengan memakai feedback terhadap keluaran untuk mengendalikan masukan maupun proses.
7. Lingkungan, segala sesuatu di luar sistem yang berpengaruh pada sistem, baik menguntungkan maupun merugikan.

### 2.1.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang. Klasifikasi sistem tersebut diantaranya adalah sistem abstrak (*abstract system*), sistem fisik (*physical system*), sistem tertentu (*deterministic system*), sistem tak tentu (*probabilistic system*), sistem tertutup (*close system*), dan sistem terbuka (*open system*) (Yakub, 2012). Berikut ini adalah penjelasan dari pembagian klasifikasi sistem:

1. Sistem tak tentu (*probabilistic system*), adalah suatu sistem yang kondisi masa depan tidak dapat di prediksi karena mengandung unsur probabilitas.
2. Sistem abstrak (*abstract system*), adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.
3. Sistem fisik (*physical system*), adalah sistem yang ada secara fisik.
4. Sistem tertentu (*deterministic system*), adalah sistem yang beroperasi dengan tingkah laku yang tidak dapat diprediksi, interaksi antara bagian dapat di deteksi dengan pasti sehingga keluaran dapat diprediksi.
5. Sistem tertutup (*close system*), adalah sistem yang tidak bertukar materi informasi, atau energi dengan lingkungan.

Sistem terbuka (*open system*), adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan.

### 2.1.3 Jenis Sistem

Ada berbagai tipe sistem berdasarkan kategori:

1. Atas dasar keterbukaan:
  - a. sistem terbuka, di mana pihak luar dapat mempengaruhinya.
  - b. sistem tertutup.
2. Atas dasar komponen:
  - a. Sistem fisik, dengan komponen materi dan energi.
  - b. Sistem non-fisik atau konsep, berisikan ide-ide.

## **2.2 Pengambilan Keputusan**

Pengambilan keputusan di dalam suatu organisasi merupakan hasil suatu proses komunikasi dan partisipasi yang terus menerus dari keseluruhan organisasi. Hasil keputusan tersebut dapat merupakan pernyataan yang disetujui antar alternatif atau antar prosedur untuk mencapai tujuan tertentu. Pendekatannya dapat dilakukan, baik melalui pendekatan yang bersifat individual atau kelompok, sentralisasi atau desentralisasi, partisipasi atau tidak berpartisipasi, maupun demokratis atau konsensus (Kusrini, 2006).

Persoalan pengambilan keputusan, pada dasarnya adalah bentuk pemilihan dari berbagai alternatif tindakan yang mungkin dipilih yang prosesnya melalui mekanisme tertentu dengan harapan akan menghasilkan sebuah keputusan yang terbaik. Penyusunan model keputusan adalah suatu cara untuk mengembangkan hubungan matematis yang mencerminkan hubungan yang terjadi diantara faktor-faktor yang terlibat. Pada umumnya para penulis sependapat bahwa kata keputusan (*decision*) berarti pilihan (*choice*) yaitu pilihan dari dua atau lebih kemungkinan. Pengambilan keputusan hampir tidak merupakan pilihan antara yang benar dan

yang salah tetapi justru yang sering terjadi ialah pilihan antara yang hampir benar dan yang mungkin salah. Keputusan yang diambil biasanya dilakukan berdasarkan pertimbangan situasional bahwa keputusan tersebut adalah keputusan terbaik.

### **2.3 Sistem Pendukung Keputusan**

Sistem pendukung keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. Sistem pendukung keputusan dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang (Keen & Scott-Morton, 1978).

Sistem pendukung keputusan seperti itu disebut aplikasi sistem pendukung keputusan. Aplikasi sistem pendukung keputusan digunakan dalam pengambilan keputusan. Aplikasi sistem pendukung keputusan menggunakan CBIS (*Computer Base Informasi Systems*) yang fleksibel, interkatif, dan dapat di adaptasi yang dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur. Aplikasi sistem pendukung keputusan menggunakan data, memberikan antar muka pengguna yang mudah, dan dapat menggabungkan pemikiran pengambil keputusan. Sistem pendukung keputusan lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas (Turban et al., 2005).

Sistem pendukung keputusan tidak dimaksudkan untuk mengotomatisasikan pengambilan keputusan tetapi memberikan perangkat interaktif yang memungkinkan pengambilan keputusan untuk melakukan berbagai analisis menggunakan model-model yang tersedia. Jadi dapat disimpulkan bahwa Sistem Pendukung Keputusan (*Decision Support Systems*) adalah dukungan bagi pengambilan keputusan baik untuk individu maupun grup yang memberikan pilihan pada pengambilan keputusan yang lebih baik dan lebih konsisten dalam satu cara yang dibatasi oleh waktu (Hatta et al., 2016).

### **2.3.1 Tujuan Sistem Pendukung Keputusan**

Adapun tujuan dari sistem pendukung keputusan sebagai berikut:

1. Membantu manajemen dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksudkan untuk menggantikan fungsi manajer.
3. Meningkatkan efektifitas keputusan yang diambil manajer lebih dari pada perbaikan efisiensinya.
4. Kecepatan komputasi, komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
5. Peningkatan produktivitas, membangun satu kelompok pengambilan keputusan terutama para pakar bisa sangat mahal. Pendukung terkomputerisasi bisa mengurangi ukuran kelompok dan memungkinkan para anggotanya untuk berada di berbagai lokasi yang berbeda-beda.

6. Dukungan kualitas, komputer bisa meningkatkan kualitas yang dibuat.

### **2.3.2 Karakteristik Sistem Pendukung Keputusan**

Berikut ini beberapa karakteristik sistem pendukung keputusan:

1. Mendukung proses pengambilan keputusan suatu organisasi atau perusahaan.
2. Adanya *interface* manusia atau mesin dimana manusia *user* tetap memegang kontrol proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur serta mendukung beberapa keputusan yang saling berinteraksi.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
6. Memiliki dua komponen utama yaitu data dan model.

### **2.3.3 Kriteria atau Ciri-ciri Pengambilan Keputusan**

Adapun kriteria atau ciri-ciri dari keputusan adalah sebagai berikut:

1. Banyak pilihan/alternative.
2. Ada kendala.

3. Mengikuti suatu pola/model tingkah laku, baik yang terstruktur maupun tidak terstruktur.
4. Banyak input/variable.
5. Ada faktor resiko. Dibutuhkan kecepatan, ketepatan, dan keakuratan.

#### **2.3.4 Karakteristik Sistem Pendukung Keputusan**

Disamping berbagai kemampuan dan karakteristik seperti dikemukakan di atas, sistem pendukung keputusan juga memiliki keterbatasan, antara lain:

1. Mendukung proses pengambilan keputusan suatu organisasi atau perusahaan.
2. Adanya *interface* manusia atau mesin dimana manusia *user* tetap memegang kontrol proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur serta mendukung beberapa keputusan yang saling berinteraksi.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
6. Memiliki dua komponen utama yaitu data dan model.

Sistem pendukung keputusan (SPK) dirancang sedemikian rupa sehingga bersifat interaktif dengan pemakainya (Mulyono, 1996).

### **2.3.5 Tahapan Sistem Pendukung Keputusan**

Adapun beberapa fase atau tahapan dalam proses pengambilan keputusan yaitu sebagai berikut:

1. *Face intelligence*

Tahap ini merupakan proses penelusuran dan pendeteksian dari ruang lingkup problematika secara proses pengenalan masalah. Data masukan diperoleh, diproses dan diuji dalam rangka mengidentifikasi masalah.

2. *Fase design*

Tahap ini merupakan proses menemukan, mengembangkan dan menganalisis alternatif tindakan yang bisa dilakukan. Tahap ini meliputi menguji kelayakan solusi.

3. *Fase choice*

Pada tahap ini dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Hasil pemilihan tersebut kemudian diimplementasikan dalam proses pengambilan keputusan.

### **2.3.6 Proses Pengambilan Keputusan**

Ada tiga fase dalam proses pengambilan keputusan diantaranya sebagai berikut:

1. Intelligence

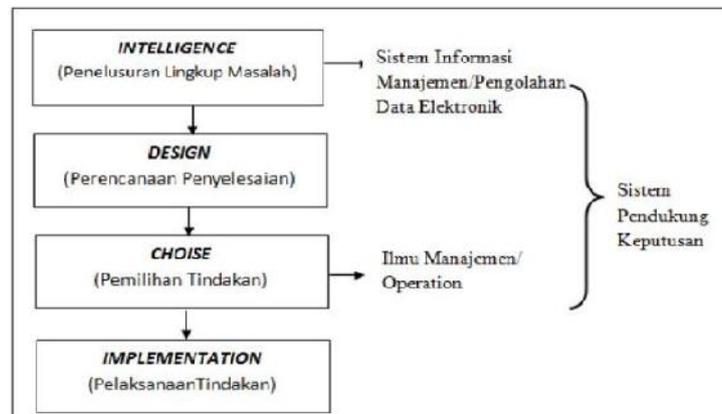
Tahapan ini merupakan proses penelusuran dan pendeteksian dari ruang lingkup problematika secara proses pengenalan masalah. Data masukan diperoleh, diproses dan diuji dalam rangka mengidentifikasi masalah.

## 2. Design

Tahap ini proses menemukan, mengembangkan dan menganalisis alternatif tindakan yang bisa dilakukan. Tahap ini meliputi menguji kelayakan solusi.

## 3. Choice

Proses pemilihan diantara berbagai alternatif tindakan dijalankan. Hasil pemilihan diimplementasikan dalam proses pengambilan keputusan.



**Gambar 2.1** Fase proses pengambilan keputusan

Sumber: (Nofriansyah, 2014)

Secara garis besar sistem pendukung keputusan dibangun oleh tiga komponen utama yaitu:

### 1. Subsistem data (Database)

Subsistem data merupakan komponen sistem pendukung keputusan yang berguna sebagai penyedia data bagi sistem. Data tersebut disimpan untuk

diorganisasikan oleh suatu sistem yang disebut dengan sistem manajemen basis data (Database Management System).

## 2. Subsistem Model (Model Base)

Model adalah suatu tiruan dari alam nyata. Kendala yang sering dihadapi dalam merancang model adalah bahwa model yang dirancang tidak mampu mencerminkan seluruh variabel alam nyata, sehingga keputusan yang diambil tidak sesuai dengan kebutuhan oleh karena itu, dalam menyimpan berbagai model harus diperhatikan dan harus dijaga fleksibilitasnya. Hal ini yang harus diperhatikan adalah pada setiap model yang disimpan hendaknya ditambahkan rincian keterangan dan penjelasan yang komprehensif mengenai model yang dibuat.

## 3. Subsistem Dialog (User System Interface)

Subsistem dialog adalah fasilitas yang mampu mengintegrasikan sistem yang terpasang dengan pengguna secara interaktif, yang dikenal dengan subsistem dialog. Melalui subsistem dialog sistem diimplementasikan sehingga pengguna dapat berkomunikasi dengan sistem yang dibuat.

## **2.4 Pemilihan Pegawai**

Model merupakan abstraksi dunia nyata menjadi bentuk simbolik dengan tujuan menyederhanakan, meminimalkan resiko agar lebih efektif. Beberapa bentuk model diantaranya.

### 1. Model Ikonik

Model ikonik adalah perwakilan fisik dari beberapa hal, baik dalam bentuk ideal ataupun dalam skala yang berbeda. Model ikonik memiliki beberapa karakteristik yang sama dengan hal yang diwakili, terutama untuk menerangkan kejadian pada waktu yang spesifik.

## 2. Model Analog

Model analog bisa mewakili situasi dinamik, yaitu keadaan yang berubah menurut waktu. Model ini lebih sering dipakai daripada model ikonik karena kemampuannya untuk mengetengahkan karakteristik dari kejadian yang dikaji. Model analog banyak berkesesuaian dengan penjabaran hubungan kuantitatif antara sifat dan kelas-kelas yang berbeda.

## 3. Model Matematik (model simbolik)

Format model simbolik berupa bentuk angka, simbol, dan rumus. Jenis model simbolik yang umum dipakai adalah suatu persamaan (equation). Bentuk persamaan adalah tepat, singkat dan mudah dimengerti. Simbol persamaan tidak saja mudah dimanipulasi daripada kata-kata, tetapi juga lebih cepat dimengerti maksudnya. Suatu persamaan adalah bahasa universal dalam penelitian operasional dan ilmu sistem, dimana digunakan suatu logika simbolis.

## **2.5 Manajemen Model**

### **2.5.1 Basis Model**

Berisi model statistik, keuangan, pengetahuan manajemen atau model kuantitatif lain yang menyediakan kemampuan analisa seperti mencari, menjalankan, menggabungkan memeriksa model (AH. Kridalaksana, 2018).

Dilihat dari tingkat manajemen penggunaannya, model dalam basis model dibedakan menjadi empat 4 bagian yaitu:

1. Model stratejik

Mendukung tanggung jawab perencanaan strategik dari top management

Contoh: Pengembangan perusahaan, pemilihan lokasi pabrik, perencanaan merger

2. Model taktikal

Digunakan oleh manajemen tingkat menengah dalam membantu pengalokasian dan pengontrolan sumber daya organisasi. Contoh:

Perencanaan kebutuhan tenaga kerja, pembelajaran rutin, dll

3. Model Operasional

Mendukung manajemen level bawah untuk pelaksanaan aktivitas sehari-hari dalam cakupan waktu yang singkat

4. Model Building Block

Model ini merupakan model yang akan digunakan untuk membangun model yang lain. Contoh: Analisis Regresi, penghitungan NPV, dll.

### **2.5.2 Sistem Manajemen Basis Model**

Ada beberapa hal yang berkaitan dengan manajemen basis model. Beberapa diantaranya adalah software pembuat model, pembaharuan model, pengubahan model dan manipulasi data.

### 2.5.3 Model Direktori

Katalog semua model dalam basis model yang terdiri dari defenisi model dan fungsi utama untuk menjawab pertanyaan tentang keberadaan dan kemampuan model. Model eksekusi, intelegensi dan perintah. Eksekusi berfungsi mengontrol jalannya aktivitas nyata. Intelegensi menggabungkan operasi beberapa model, sedangkan perintah berfungsi menerima dan menerjemahkan instruksi model dari model lain.

### 2.6 Metode *Weighted Product* (WP)

Metode *Weighted Product* merupakan salah satu metode yang sederhana dengan perkalian untuk menghubungkan rating atribut, dimana setiap rating atribut harus dipangkatkan dengan bobot atribut yang bersangkutan. Hal tersebut dinamakan normalisasi (Mufizar, 2018).

Langkah-langkah penyelesaian *Weighted Product* adalah sebagai berikut:

1. Menentukan kriteria-kriteria terlebih dahulu yang akan dijadikan tolak ukur dalam pengambilan keputusan.
2. Menormalisasikan setiap nilai alternative dengan perbaikan bobot  $\sum W_j=1$  adalah dengan rumus:

$$W_j = \frac{w}{\sum w} \dots\dots\dots(1)$$

3. Menghitung nilai bobot preferensi pada setiap alternatif dengan variabel  $W$  adalah pangkat bernilai positif untuk atribut keuntungan dan bernilai negatif untuk atribut biaya. Preferensi untuk alternatif  $S_i$  diberikan sebagai berikut:

$$S_i = \prod_{j=1}^n x_{ij}^{w_j} \dots\dots\dots(2)$$

Dimana:

$S_i$  = nilai dari setiap alternatif

$n$  = banyaknya kriteria

$x_{ij}$  = nilai dari setiap baris dan kolom

$w_j$  = nilai atribut yang dimiliki dari setiap kriteria

$\Pi$  = produk

4. Melakukan perhitungan yang diperoleh dari nilai terbesar yang dipilih sebagai alternatif terbaik. Dengan Rumus:

$$V_i = \frac{S_i}{\prod_{j=1}^n (x_j^*)^{w_j}} \dots\dots\dots(3)$$

Dimana :

$V$  : Preferensi alternatif sebagai vektor  $V$

$X$  : Nilai Kriteria

$W$  : Bobot kriteria

$i$  : Alternatif

$j$  : Kriteria

$n$  : Banyaknya kriteria

**Tabel 2.1 Pembobotan Kriteria**

<b>Kriteria</b>	<b>Bobot</b>
Harga	5
Kualitas	4
Bahan	3
Motif	2
Warna	1

**Tabel 2.2 Skala pembobotan Kriteria**

<b>Kriteria</b>	<b>Skala</b>	<b>Bobot</b>
Harga	0 – 100 ribu	5
	101 ribu – 200 ribu	4
	201 ribu – 300 ribu	3
	301 ribu – 400 ribu	2
	≥ 500 ribu	1
Kualitas	Sangat Rendah	1
	Rendah	2
	Sedang	3
	Tinggi	4
	Sangat Tinggi	5
Bahan	Sangat Kasar	1
	Kasar	2
	Sedang	3
	Lembut	4
	Sangat Lembut	5
Motif	Polos	1
	Bercorak	2
	Bordir	3
	Sablon	4
	Bordir Sablon	5
Warna	Sangat Buram	1

	Buram	2
	Sedang	3
	Terang	4
	Sangat Terang	5

Tabel 2.1 dan tabel 2.2 adalah contoh pembuatan kriteria dan pengisian kriteria dengan bobot berskala 1 hingga 5.

## 2.7 Flowchart

*Flowchart* adalah penggambaran secara fisik dari langkah – langkah dan urutan-urutan prosedur dari suatu program. Tujuan utama dari penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas dengan menggunakan symbol-simbol yang standart. Tahapan penyelesaian masalah yang disajikan harus jelas, sederhana, efektif dan tepat. Dalam penulisan *flowchart* dikenal 2 model yaitu sistem *flowchart* dan program *flowchart* (Raymond Mcleod & Schell, 2007).

### 2.7.1 Sistem Flowchart

Sistem *flowchart* merupakan diagram alir yang menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antar alat tersebut. Sistem *flowchart* ini tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah tetapi hanya untuk menggambarkan prosedur dalam sistem yang dibentuk.

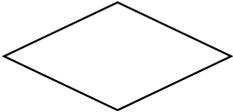
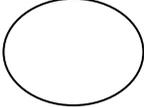
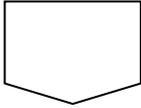
### 2.7.2 Program Flowchart

Program *flowchart* merupakan diagram alir yang menggambarkan urutan logika dari suatu prosedur pemecahan masalah. Untuk menggambarkan program *flowchart* telah tersedia simbol-simbol standar, namun demikian seperti halnya pada sistem *flowchart*, pemrograman dapat menambah simbol-simbol tersebut asalkan pemrograman melengkapinya dengan penggambaran program *flowchart* dengan kamus simbol.

Berikut ini adalah gambar dari simbol-simbol standar yang digunakan pada program *flowchart*.

**Tabel 2.3 Simbol-simbol *Flowchart***

No	Simbol	Nama	Fungsi
1		Terminator	Permulaan / akhir program
2		Garis alir	Arah aliran program
3		<i>Preperation</i>	Proses inisialisasi / pemberian nilai awal
4		<i>Proses</i>	Proses perhitungan / proses pengolahan data
5		<i>Input/output</i>	Proses input/output data, parameter, informasi

6		<i>Predefined procces</i>	Permulaan sub program / proses menjalankan sub program
7		<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
8		<i>On page connector</i>	Penghubung bagian – bagian flowchart yang berada pada satu halaman
9		<i>Off page connector</i>	Penghubung bagian – bagian flowchart yang berada pada halaman yang berbeda

Sumber: (Kurniawan, 2018)

*Flowchart* adalah salah satu bagan yang menjelaskan dengan rinci langkah – langkah dari proses suatu program”. *Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma. *Flowchart* membantu seorang analis dan *programmer* dalam memecahkan suatu masalah serta dapat membantu dalam menganalisis alternatif – alternatif dalam pengoperasiannya.

Fungsi *Flowchart* untuk menggambarkan sebuah proses agar mempermudah pemahaman dan mudah dilihat berdasarkan urutan langkahnya berdasarkan proses yang satu ke proses yang lainnya. *Flowchart* disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut:

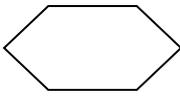
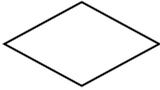
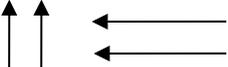
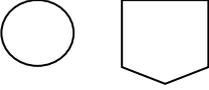
1. *Flow Direction Symbols* (Simbol Penghubung/ Alur)
2. *Processing Symbols* (Simbol Proses)
3. *Input-Output Symbols* (Simbol Input-Output)

### 2.7.3 Flow Direction Symbols

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut.

**Tabel 2.4 *Flow Direction Symbols***

Simbol	Keterangan
Terminal 	Untuk menggambarkan awal dan akhir proses aliran dokumen
Processing 	Dipakai untuk pengolahan aritmatika dan pemindahan data

Preparation 	Dipakai untuk memberikan nilai awal dari suatu variable atau counter
Decision 	Dipakai untuk mewakili operasi perbandingan logika
Predefined process 	Dipakai untuk proses yang detailnya dijelaskan secara terpisah, misalnya dalam bentuk subroutine
Garis alir 	Dipakai untuk menunjukkan aliran dari program
Penghubung 	Menunjukkan penghubung di halaman yang masih sama atau di halaman lainnya
Input / Output 	Mewakili data input atau output

Sumber: (Kurniawan, 2018)

#### 2.7.4 Simbol Flowchart

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur. Simbol-simbol tersebut adalah sebagai berikut:

**Tabel 2.5 Processing Symbols**

	Process	Simbol yang menunjukkan pengolahan yang dilakukan Komputer
	Decision	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban / aksi
	Predefined Process	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan didalam storage
	Terminal	Simbol untuk permulaan atau akhir dari suatu program
	Manual Input	Simbol untuk pemasukan data secara manual on-line keyboard

Sumber: (Kurniawan, 2018)

### 2.7.5 Input-Output Symbols

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media input atau output. Simbol-simbol tersebut sebagai berikut:

**Tabel 2.6 Input-Output Symbols**

	Process	Simbol yang menunjukkan pengolahan yang dilakukan Komputer
	Decision	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban / aksi
	Predefined Process	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan didalam storage
	Terminal	Simbol untuk permulaan atau akhir dari suatu program
	Manual Input	Simbol untuk pemasukan data secara manual on-line keyboard

Sumber: (Kurniawan, 2018)

## 2.8 *Unified Modelling Language*

*Unified Modeling Language* adalah bahasa pemodelan standar yang memungkinkan pengembang menentukan, memvisualisasikan, membuat, dan mendokumentasikan artefak sistem perangkat lunak (Technopedia, 2019). Dengan demikian, *UML* membuat artefak ini dapat diskalakan, aman, dan kuat dalam eksekusi. *UML* adalah aspek penting yang terlibat dalam pengembangan perangkat lunak berorientasi objek. Ini menggunakan notasi grafis untuk membuat model visual dari sistem perangkat lunak. Arsitektur *UML* didasarkan pada fasilitas meta-objek, yang mendefinisikan dasar untuk membuat bahasa pemodelan. Mereka cukup tepat untuk menghasilkan seluruh aplikasi. *UML* yang sepenuhnya dapat dieksekusi dapat digunakan untuk berbagai platform menggunakan teknologi yang berbeda dan dapat digunakan dengan semua proses sepanjang siklus pengembangan perangkat lunak. *UML* dirancang untuk memungkinkan pengguna mengembangkan bahasa pemodelan visual yang ekspresif, siap pakai. Selain itu, mendukung konsep pengembangan tingkat tinggi seperti kerangka kerja, pola, dan kolaborasi (Wasserkrug et al., 2019).

Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain diluarnya (Sukmawati & Priyadi, 2019).

### 2.8.1 *Use Case Diagram*

*Use Case Diagram* adalah model tentang bagaimana berbagai jenis pengguna berinteraksi dengan sistem untuk memecahkan masalah. Dengan demikian, ini menggambarkan tujuan pengguna, interaksi antara pengguna dan sistem, dan perilaku sistem yang diperlukan dalam memenuhi tujuan-tujuan ini. Model *use case* terdiri dari sejumlah elemen model. Elemen model yang paling penting adalah kasus penggunaan, aktor dan hubungan di antara mereka. Diagram use-case digunakan untuk menggambarkan secara grafis subset dari model untuk menyederhanakan komunikasi. Biasanya akan ada beberapa diagram kasus penggunaan yang terkait dengan model yang diberikan, masing-masing menunjukkan subset elemen model yang relevan untuk tujuan tertentu. Elemen model yang sama dapat ditampilkan pada beberapa diagram use-case, tetapi setiap instance harus konsisten. Jika alat digunakan untuk mempertahankan model *use case*, kendala konsistensi ini otomatis sehingga setiap perubahan pada elemen model (mengubah nama misalnya) akan secara otomatis tercermin dalam setiap diagram use-case yang menunjukkan elemen itu (UTM, 2019).

Model use-case dapat berisi paket yang digunakan untuk menyusun model untuk menyederhanakan analisis, komunikasi, navigasi, pengembangan, pemeliharaan, dan perencanaan. Faktanya, sebagian besar model *use case* adalah tekstual, dengan teks yang ditangkap dalam Spesifikasi *Use Case* yang terkait dengan setiap elemen model use-case. Spesifikasi ini menjelaskan alur peristiwa use case. Model *use case* berfungsi sebagai utas pemersatu sepanjang pengembangan sistem. Ini digunakan sebagai spesifikasi utama dari persyaratan

fungsional untuk sistem, sebagai dasar untuk analisis dan desain, sebagai input untuk perencanaan iterasi, sebagai dasar mendefinisikan kasus uji dan sebagai dasar untuk dokumentasi pengguna. (Kurniawan, 2018).

*Use case diagram* merupakan suatu diagram yang berisi *use case*, *actor*, serta *relationship* diantaranya. *Use Case Diagram* dapat digunakan untuk kebutuhan apa saja yang diperlukan dalam suatu sistem, sehingga sistem dapat digambarkan dengan jelas bagaimana proses dari sistem tersebut, bagaimana cara aktor menggunakan sistem, serta apa saja yang dapat dilakukan pada suatu sistem. Adapun simbol dari *use case* adalah sebagai berikut:

**Tabel 2.7 Simbol Use Case Diagram**

No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
3		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .

5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: (Kurniawan, 2018)

### 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 mereka berakhir (Ladjamudin, 2017).

*Activity diagram* menurut adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case*. Diagram ini juga dapat digantikan dengan sejumlah teks.

**Tabel 2.8 Simbol Activity Diagram**

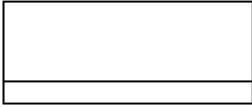
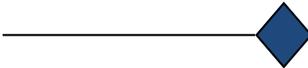
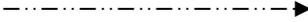
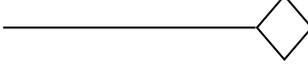
No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk /diawali.
4		<i>Activity</i> <i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber: (Kurniawan, 2018)

### 2.8.3 Class Diagram

*Class diagram* digunakan untuk menggambarkan perbedaan yang mendasar antara *class*, hubungan antara *class*, dan di mana *sub-sistem class* tersebut (Jogiyanto, 2016). Simbol yang digunakan dalam *class diagram* adalah sebagai berikut:

Tabel 2.9 Simbol Class Diagram

Simbol	Nama	Fungsi
	<i>Class</i>	Menggambarkan <i>Class</i> baru pada diagram.
	<i>Association</i>	Menggambarkan relasi antar asosiasi
	<i>Composition</i>	Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap <i>class</i> tempat dia bergantung tersebut.
	<i>Depedency</i>	Umumnya penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain.
	<i>Aggregation</i>	<i>Aggregation</i> mengindikasikan keseluruhan bagian <i>relationship</i> dan biasanya disebut sebagai relasi.

Sumber: (Kurniawan, 2018)

## 2.9 Database

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi". *Database* digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam

komputer. Untuk mengolah *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*) (Hung et al., 2018).

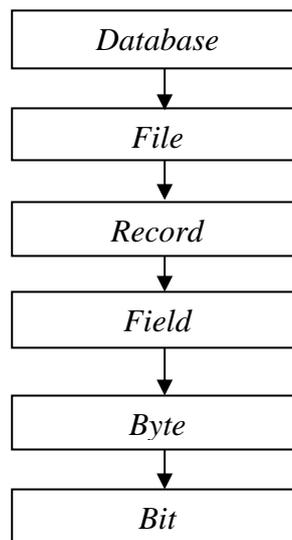
DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* (pengguna) untuk membuat, memelihara, mengontrol dan mengakses *database* secara praktis dan efisien. Dengan DBMS, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada. Sedangkan RDBMS atau *Relationship Database System* merupakan salah satu jenis DBMS yang mendukung adanya *relationship* atau hubungan antar label. Disamping RDBMS, terdapat jenis DBMS lain, misalnya: *Hierarchy DBMS*, *Object Oriented DBMS* dan sebagainya.

Berdasarkan tingkat kompleksitas nilai data, tingkatan data dapat disusun kedalam sebuah hierarki, mulai dari yang paling sederhana hingga yang paling kompleks. Urutan atau hierarki *database* adalah sebagai berikut:

1. *Database* adalah sekumpulan dari bermacam-macam tipe *record* yang memiliki hubungan antar *record*.
2. *File* adalah sekumpulan rekaman data yang berkaitan dengan suatu objek.
3. *Record* adalah kumpulan elemen-elemen yang saling berkaitan dengan di informasikan tentang suatu *entity* secara lengkap. Suatu record terdiri atas satu atau beberapa *field* yang membentuk satu kesatuan.
4. *Field* adalah unit terkecil yang disebut data yang tidak dapat dipecah lagi menjadi unit lain yang bermakna.
5. *Byte* adalah bagian terkecil yang dialamatkan dalam memori.
6. *Bit* adalah sistem biner yang terdiri atas dua macam nilai, yaitu 0 dan 1. Sistem biner merupakan dasar yang dapat digunakan untuk komunikasi

antara manusia dan mesin, yang merupakan serangkaian komponen elektronik dan hanya dapat membedakan 2 macam keadaan, yaitu ada tegangan dan tidak ada tegangan yang masuk ke rangkaian tersebut.

Hierarki *database* dapat dilihat sebagai berikut:



**Gambar 2.2 Hierarki Data**

## 2.10 Tipe Data

Data yang terdapat dalam sebuah tabel berupa *field-field* yang berisi nilai dari data tersebut. Nilai data dalam *field* memiliki tipe sendiri-sendiri. MySQL mengenal beberapa tipe data *field* yaitu:

### 2.10.1 Tipe Data Numerik

Tipe data *numerik* dibedakan dalam dua macam, yaitu *integer* dan *floating point*. *Integer* digunakan untuk data bilangan bulat sedangkan *floating point* digunakan untuk bilangan desimal. Tipe data numerik selengkapnya dapat dilihat pada tabel 2.10.

**Tabel 2.10 Tipe Data Numerik**

<b>Tipe Data</b>	<b>Kisaran Nilai</b>
TINYINT	(-128)-127 atau (0-255)
SMALLINT	(-32768)-32767 atau (0-65535)
MEDIUMINT	(-3888608)-8388607 atau 0-16777215
INT, INTEGER	(-2147683648)-(-21447683647) atau 0-4294967295
FLOAT	(-3.4 E+38)-(-1.17E-38), 0 dan 1.175E-38-3.4e+38
DOUBLE	(-1.79E+308)-(-2.225E-308), 0 dan 2.225E-308 – 1.79E+308

### 2.10.2 Tipe Data String

*String* adalah rangkaian karakter. Tipe data ini sering digunakan untuk menyimpan data-data yang panjang dan berbentuk huruf dan angka. Tipe-tipe data yang termasuk dalam tipe data *string* dapat dilihat pada tabel 2.11.

**Tabel 2.11 Tipe Data String**

<b>Tipe Data</b>	<b>Kisaran Nilai</b>
CHAR	1-255 karakter
VARCHAR	1-255 karakter
TINYTEXT	1-255 karakter
TEXT	1-65535 karakter
MEDIUMTEXT	1-16777215 karakter
LONGTEXT	1-424967295 karakter

### 2.10.3 Tipe data tanggal

Untuk tanggal dan jam, tersedia tipe-tipe data *field* berupa *DATETIME*, *DATE*, *TIMESTAMP*, *TIME* dan *YEAR*. Masing-masing tipe mempunyai kisaran nilai tertentu. MYSQL akan memberikan peringatan kesalahan (*error*) apabila tanggal atau waktu yang dimasukkan salah. Kisaran nilai dan besar memori penyimpanan yang diperlukan untuk masing-masing tipe dapat dilihat pada tabel 2.12.

**Tabel 2.12 Tipe Data Tanggal**

<b>Tipe Data</b>	<b>Kisaran Nilai</b>
DATETIME	1000-01-01 00:00 sampai 9999-12-31 23:59:59
DATE	1000-01-01 sampai 9999-12-31
TIMESTAMP	1970-01-01 00:00:00 sampai 2037
TIME	-839:59:59 sampai 838:59:59
YEAR	1901 sampai 2155

### 2.11 Visual Basic.NET

*Visual Studio 2010* pada dasarnya adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. *Visual Studio 2010 (Visual Basic.NET 2010)* selain disebut dengan bahasa pemrograman juga sering disebut sebagai sarana (tool) untuk menghasilkan program-program aplikasi berbasis *windows*. *Visual Basic.NET* adalah *Visual*

*Basic* yang direkayasa kembali untuk digunakan pada *platform .NET* sehingga aplikasi yang dibuat menggunakan *Visual Basic.NET* dapat berjalan pada sistem komputer apapun dan dapat mengambil data dari *server* dengan tipe apapun asalkan terinstal *.NET Framework*. Pada umumnya *Visual Basic.NET* terpaket dalam *Visual Studio.NET*. Pada distribusinya terdapat berbagai versi *Visual Studio..NET* yaitu versi *Professional*, *Premium*, dan yang paling lengkap adalah versi *Ultimate*.

*Visual Basic* adalah sebuah bahasa pemrograman yang berpusat pada *object* (*Object Oriented Programming*) digunakan dalam pembuatan aplikasi *windows* yang berbasis *Graphical User Interface* hal ini menjadikan *Visual Basic* menjadi bahasa pemrograman yang wajib diketahui dan dikuasai oleh setiap programmer. Beberapa karakteristik obyek tidak dapat dilakukan oleh *Visual Basic* misalnya seperti *Inheritance* tidak bisa *module* dan *polymorphism* secara terbatas bisa dilakukan dengan deklarasi *class module* yang mempunyai *interface* tertentu (Irwin, 2018).

Beberapa kemampuan atau manfaat dari *Visual Studio 2010* (*Visual Basic.NET*) diantaranya seperti:

1. Untuk membuat program aplikasi berbasis *windows*.
2. Untuk membuat objek-objek pembantu program seperti misalnya: kontrol *ActiveX*, *file Help*, aplikasi internet dan sebagainya.
3. Menguji program (*debugging*) dan menghasilkan program berakhiran *EXE* yang bersifat *executable* atau dapat langsung dijalankan.

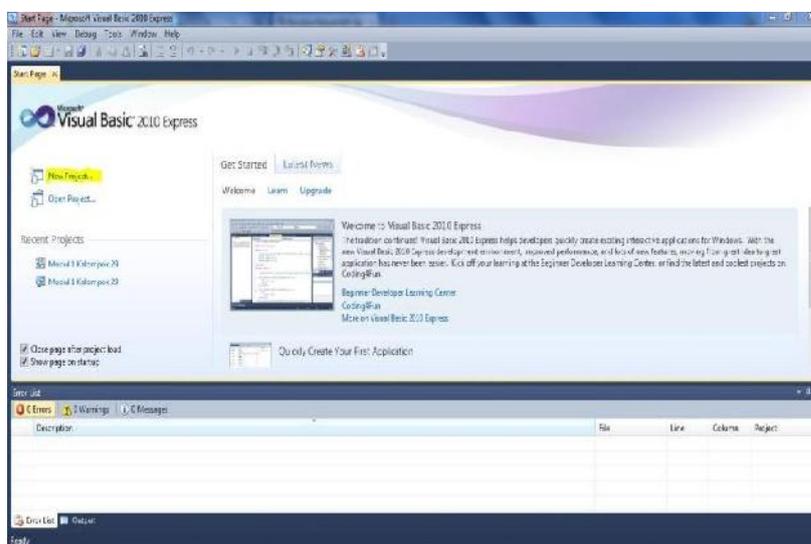
*Visual Basic.Net* layak untuk dijadikan pilihan bahasa pemrograman visual, karena memiliki cukup banyak kelebihan. Beberapa kelebihan *Visual Basic.Net* yaitu:

1. Sederhana dan mudah dipahami.
2. Mendukung *Graphical User Interface (GUI)*.
3. Menyederhanakan *deployment*.
4. Menyederhanakan pengembangan perangkat lunak.
5. Mendukung penuh *Object Oriented Programming (OOP)*.
6. Mempermudah pengembangan aplikasi berbasis *Web*.
7. Migrasi ke *Visual Basic.NET* dapat dilakukan dengan mudah.
8. Banyak digunakan oleh programmer-programmer di seluruh dunia.

*Visual Studio 2010* adalah bahasa yang cukup mudah untuk dipelajari. Bagi programmer pemula yang baru ingin belajar program lingkungan *Visual Studio* dapat membantu membuat program dalam sekejap mata. Sedang bagi programmer tingkat lanjut kemampuan yang besar dapat digunakan untuk membuat program-program yang kompleks misalnya lingkungan *net-working* atau *client server*. Bahasa *Visual Studio* cukup sederhana dan menggunakan kata-kata bahasa *Inggris* yang umum digunakan. Kita tidak perlu lagi menghafalkan sintaks-sintaks maupun format-format bahasa yang bermacam-macam di dalam *Visual Basic* semuanya sudah disediakan dalam pilihan-pilihan yang tinggal diambil sesuai dengan kebutuhan. Selain itu sarana pengembangannya yang bersifat *visual* memudahkan

kita untuk mengembangkan aplikasi berbasis *windows* bersifat *mouse-driven* (digerakkan dengan *mouse*) dan berdaya guna tinggi (Irwin, 2018).

Kompiler yang dimasukkan kedalam *Visual Studio 2010* antara lain Visual Basic, Visual C#, Visual C++, Visual InterDev, Visual J++, Visual F#, dan Visual Source Safe, dan banyak yang lainnya. Dan semua itu sudah terpaket dan diperuntukkan kedalam platform .Net Framework 4.0 atau versi yang lebih tinggi. Visual studio ini dapat digunakan untuk membuat aplikasi yang berbasis desktop yang merupakan platform windows, namun juga dapat dijalankan dalam bentuk Microsoft Intermediate Language diatas .Net Framework. Selain itu Visual Studio juga dapat digunakan untuk membuat aplikasi yang dapat dijalankan dengan windows mobile yang berjalan diatas Net Compact Framework.



**Gambar 2.3** Tampilan awal *Visual Studio 2010*

Sumber: (Lee, 2014)

Perangkat lunak ini memiliki beberapa versi. Beberapa versi *Visual Studio* yang ada di pasaran, antara lain:

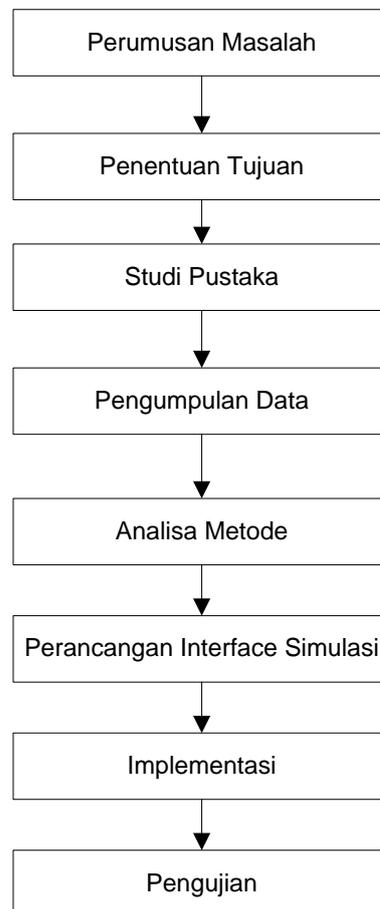
1. *Visual Studio 2010* Express Edition yang bisa digunakan secara gratis tanpa memberikan royalti kepada Microsoft Inc.
2. *Visual Studio* Standard Edition.
3. *Visual Studio 2010* Professional Edition.
4. *Visual Studio 2010* Ultimate Edition.

## BAB III

### METODE PENELITIAN

#### 3.1 Tahapan Penelitian

Adapun tahapan penelitian yang dilakukan oleh penulis ini dengan judul Sistem Pendukung Keputusan Penilaian Pegawai Berprestasi Menggunakan Metode *Weighted Product* Berbasis Website adalah sebagai berikut:



**Gambar 3.1 Tahapan Penelitian**

Ada beberapa tahapan yang perlu dilakukan dalam mencapai hasil yang diinginkan. Penulis melakukan susunan perancangan untuk menentukan pegawai terbaik dengan memberikan beberapa tahapan yang harus dilakukan. Tahapan berikut merupakan langkah-langkah yang harus dikerjakan dalam melakukan penelitian dalam menentukan pegawai terbaik.

1. Studi Literatur

Studi literatur yang dilakukan yaitu pencarian sumber daya yang dilakukan berdasarkan sumber-sumber yang berhubungan dengan penentuan pegawai terbaik dengan metode *Weighted Product*. Referensi dapat diambil dari buku-buku, jurnal *web*.

2. Pengumpulan Data

Pengumpulan data adalah pencarian terhadap sesuatu karena ada perhatian dan keinginan terhadap hasil suatu aktivitas. Metode pengumpulan data dalam penulisan ini dibagi menjadi 3, yaitu :

- a. Wawancara (*Interview*).

Wawancara ini dilakukan dengan cara mengadakan komunikasi yang dapat memberikan informasi dan data-data yang diperoleh mengenai kemanaan data.

- b. Pengamatan (*Observation*)

Penulis melakukan pengamatan langsung pada setiap penilaian pegawai yang berprestasi.

- c. Penelitian Kepustakaan (*Library Research*)

Mencari referensi dengan mengumpulkan bahan-bahan pustaka yang dilakukan di perpustakaan kampus, maupun perpustakaan umum, juga melakukan pencarian lewat internet, dengan mengunjungi situs-situs seperti *google Book online* yang dapat membantu pembahasan materi.

### 3. Analisa

Analisa dilakukan untuk menentukan bagaimana permasalahan dapat diselesaikan. Salah satu analisa dilakukan untuk menentukan kriteria apa yang layak dalam menentukan pegawai terbaik. Kriteria-kriteria tersebut mendukung permasalahan tersebut dapat diselesaikan dengan metode *Weighted Product*.

### 4. Pembahasan

Pembahasan dilakukan dengan melakukan perhitungan sistem pendukung keputusan dengan menggunakan metode *Weighted Product* dalam menentukan pegawai terbaik. Hasil diperoleh berdasarkan kriteria dan bobot preferensi yang sudah ditentukan pada perancangan kriteria.

### 5. Implementasi dan pengujian

Implementasi dan pengujian merupakan ujicoba hasil program aplikasi dan hasil perhitungan sistem pendukung keputusan metode *Weighted Product* dalam menentukan pegawai terbaik.

### **3.2 Sistem Yang Berjalan**

Dari tahap analisis masalah dapat di ketahui dengan jelas masalah-masalah apa saja yang sering muncul dalam penilaian jasa. Berdasarkan hasil penelitian yang dilakukan, didapatkan permasalahan sebagai berikut:

- a. Penilaian jasa belum memiliki perhitungan secara komputerisasi.
- b. Banyaknya perusahaan yang komplain dikarenakan tidak adanya laporan penilaian jasa dari perusahaan.
- c. Perhitungan nilai jasa yang menentukan kualitas dari pegawai, masih menggunakan semi komputer (menggunakan Microsoft Excel).

### **3.3 Sistem Yang Diusulkan**

Dari hasil wawancara dan observasi yang dilakukan pada perusahaan, di dapat beberapa hal yang dianggap penting dalam kebutuhan sistem pendukung keputusan penilaian jasa. Untuk implementasi sistem dibutuhkan perangkat komputer di dalam dengan menggunakan data dummy untuk digunakan oleh admin, admin harus mempunyai kemampuan untuk menjalankan sistem pendukung keputusan tersebut. Kebutuhan dari hasil analisis ini harus dapat dilaksanakan, diukur, diuji, terkait dengan kebutuhan di lapangan yang teridentifikasi, serta didefinisikan sampai tingkat detail yang memadai untuk desain sistem.

### **3.4 Analisis Prosedur Pada Sistem Yang Diusulkan**

Analisis prosedur pada sistem yang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem pendukung keputusan tersebut, sehingga

kelebihan dan kekurangan sistem pendukung keputusan dapat diketahui dengan cara menerapkan metode *Weighted Product*.

### **3.5 Prosedur Penilaian Jasa**

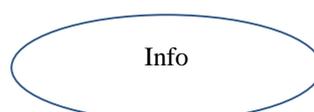
Prosedur penilaian jasa yang harus dipenuhi oleh pegawai adalah sebagai berikut: pekerjaan yang dikerjakan hasil bernilai absen, komunikasi, penampilan, hasil kerja dan loyalitas yang tepat yang dilaksanakan dalam satu bulan. Jika pegawai tersebut memenuhi ketentuan maka akan mendapatkan nilai jasa yang baik pada bulan selanjutnya dan jika pegawai tersebut tidak memenuhi ketentuan maka tidak akan mendapatkan sanksi.

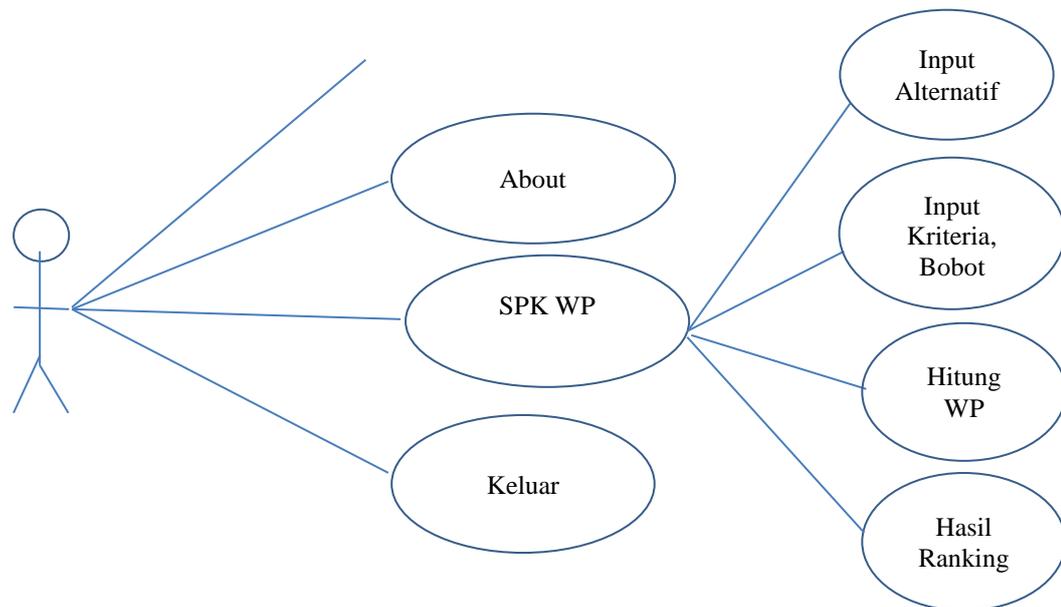
### **3.6 Rancangan Penelitian**

Beberapa diagram akan digunakan dalam menentukan dan menggambarkan rancangan penelitian. Diagram ini akan menjelaskan alur dari program aplikasi yang akan digunakan dalam penelitian ini. Fungsi dari masing-masing keadaan akan dijelaskan terutama tentang metode *Multi-factor Evaluation Process*.

#### **3.6.1 Use Case Diagram**

*Use case diagram* merepresentasikan sistem secara keseluruhan. Pada diagram ini digambarkan hubungan sistem dengan entitas luar yang terlibat. Adapun *use case diagram* dari sistem pendukung keputusan penentuan pegawai terbaik yang akan dibuat dapat digambarkan pada gambar 3.2



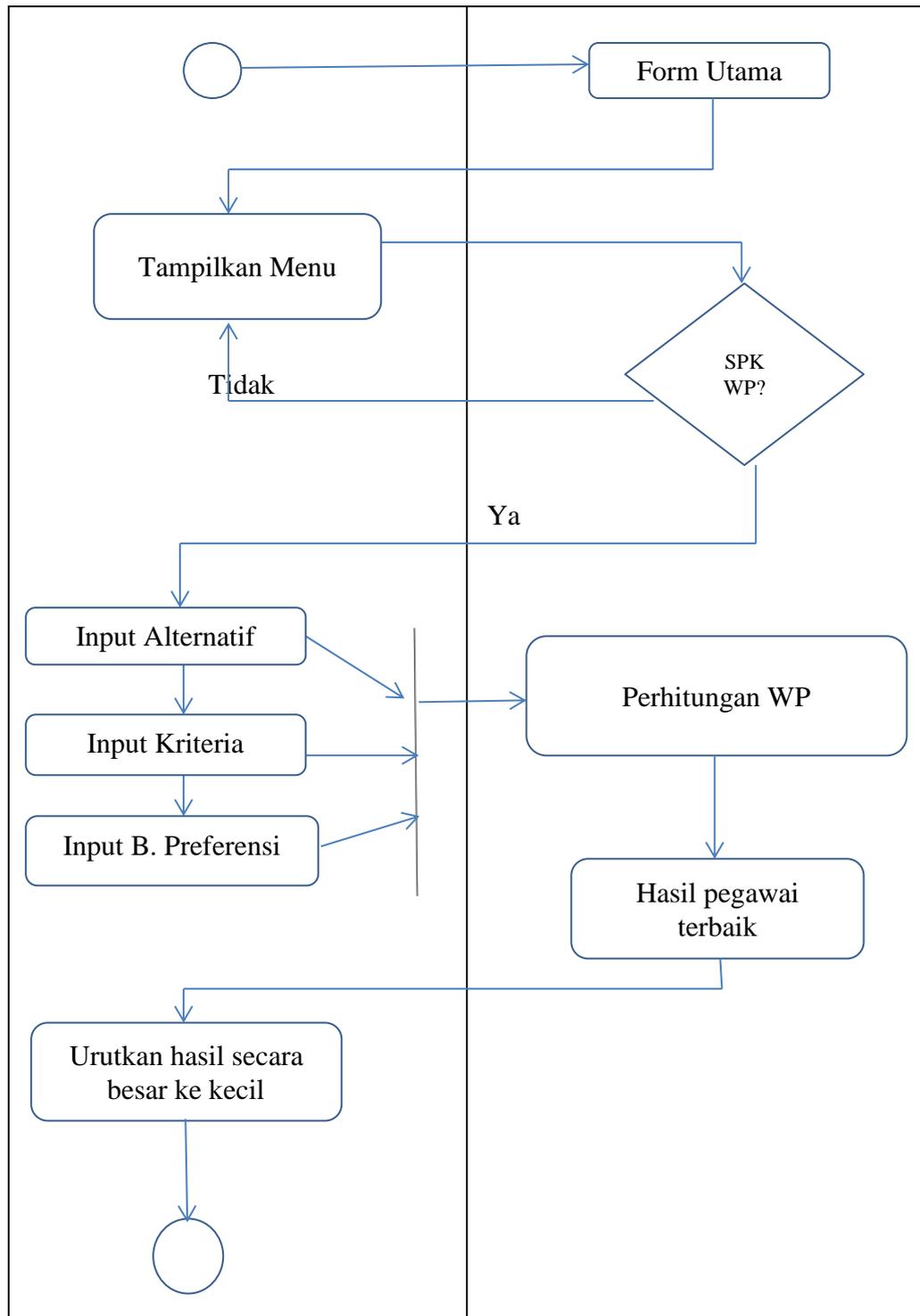


**Gambar 3.2 Use Case Diagram**

### 3.6.2 Activity Diagram

*Activity diagram* menggambarkan kegiatan dari sistem yang dilakukan pengguna untuk menentukan pegawai terbaik dengan menggunakan metode *Weighted Product*. *Activity diagram* dari sistem pendukung keputusan bertujuan memberikan hasil bahwa alternatif mana yang layak mendapatkan posisi pegawai terbaik. Gambar 3.3 adalah *Activiti Diagram* dari perancangan penelitian ini.

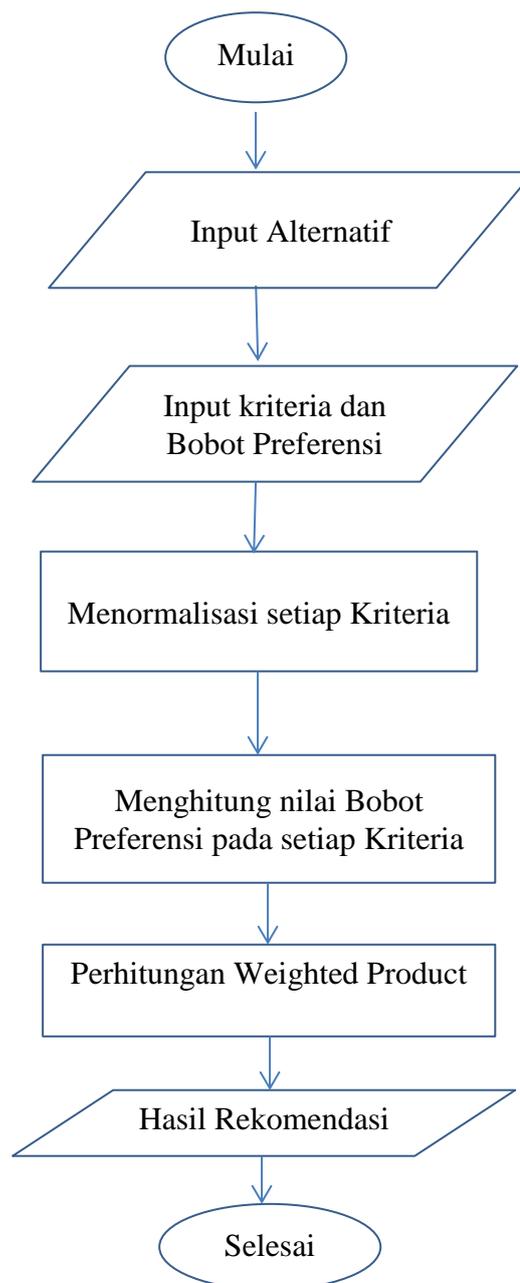




**Gambar 3.3 Activity Diagram**

### 3.6.3 Flowchart

*Flowchart* adalah urutan langkah kerja atau diagram alir suatu proses yang digambarkan dan disusun secara sistematis. Dari sistem pendukung keputusan pegawai terbaik yang akan dibuat dapat digambarkan pada gambar 3.4.



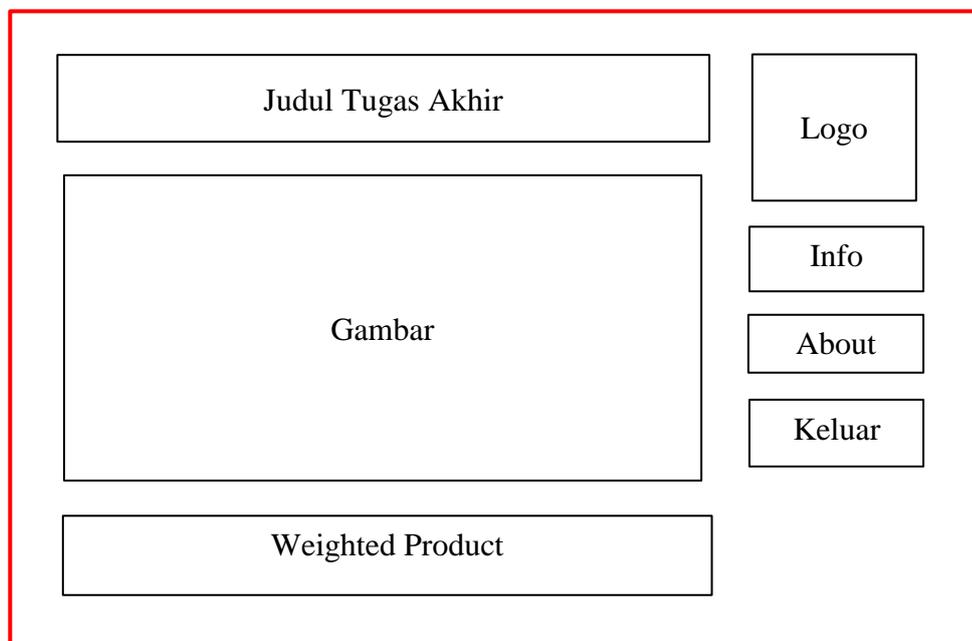
**Gambar 3.4 Flowchart SPK WP**

### 3.7 Perancangan Antarmuka

Perancangan merupakan bagian yang paling penting dalam merancang sistem. Perancangan ini dilakukan untuk memudahkan pemrograman yang dilakukan dengan menggunakan bahasa pemrograman *Microsoft Visual Basic.Net 2010*. Berikut ini merupakan tahapan perancangan sistem pendukung keputusan penentuan pegawai terbaik.

#### 3.7.1 Rancangan Menu Beranda

Menu beranda adalah menu pembuka dimana pengguna akan menjumpainya pertama sekali. Beberapa komponen terdapat pada menu utama tersebut. Gambar 3.5 adalah hasil perancangan menu utama.



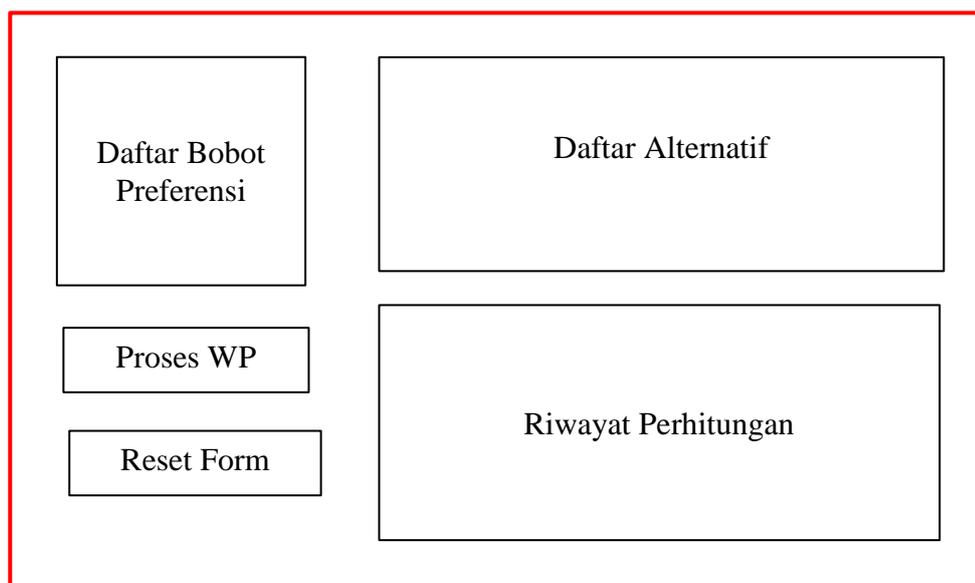
**Gambar 3.5 Rancangan Menu Utama**

Menu ini memiliki berapa komponen antara lain:

1. Logo
2. Judul Tugas Akhir
3. Gambar
4. Profil
5. About
6. Weighted Product
7. Keluar

### 3.7.2 Rancangan Menu *Weighted Product*

Rancangan menu *Weighted Product* adalah bagian utama program aplikasi aplikasi yang berfungsi untuk memproses alternatif sehingga menghasilkan penentuan pegawai terbaik. Gambar 3.6 adalah rancangan menu SPK MFEP.



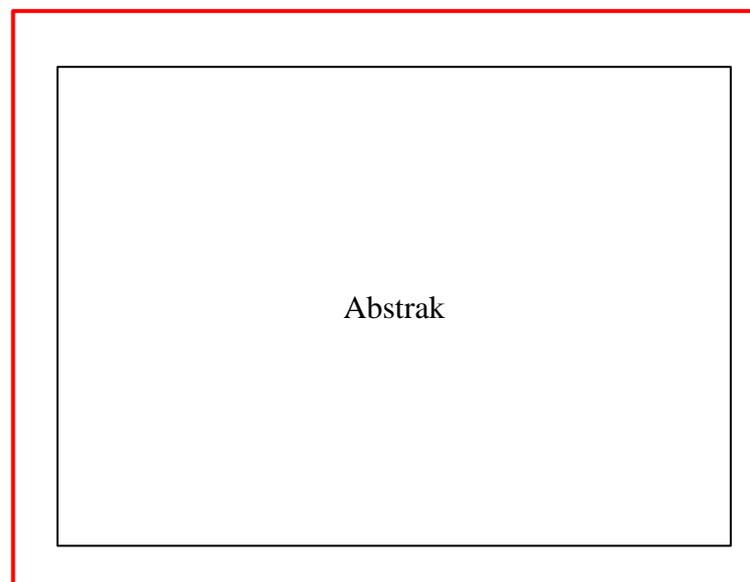
**Gambar 3.6 Rancangan Menu *Weighted Product***

Menu SPK MFEP memiliki beberapa bagian antara lain:

1. Daftar Alternatif
2. Daftar Bobot Preferensi
3. Riwayat Perhitungan
4. Tombol Proses WP
5. Tombol Reset

### 3.7.3 Rancangan Menu Info

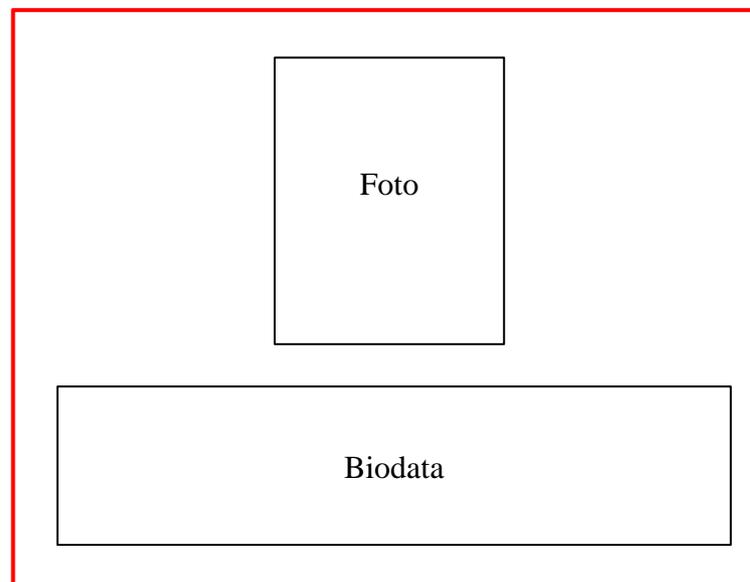
Rancangan menu info adalah abstrak dari tugas akhir penulis. Abstrak menjelaskan tahapan singkat dan hasil dari penelitian sistem pendukung keputusan metode *Weighted Product*. Gambar 3.7 adalah hasil perancangan menu info.



**Gambar 3.7 Rancangan Menu Info**

### 3.7.4 Rancangan Menu About

Menu ini akan menampilkan biodata penulis. Pada menu ini akan ditampilkan foto dari penulis. Menu ini terdiri dari objek logo dan biodata. Gambar 3.8 adalah hasil perancangan dari menu About.



**Gambar 3.8 Rancangan Menu About**

### 3.8 Perancangan Kriteria

Kriteria yang digunakan dalam penelitian ini adalah sebanyak lima kriteria yaitu Absen, Komunikasi, Penampilan, Hasil Kerja dan Loyalitas. Setiap kriteria memiliki bobot yang akan berguna untuk memberikan acuan dan nilai terhadap masing-masing alternatif. Tabel-tabel berikut merupakan data lengkap penggunaan kriteria pada penelitian tentang sistem pendukung keputusan penentuan pegawai terbaik.

**Tabel 3.1 Penilaian Jasa Kriteria**

<b>Inisial</b>	<b>Bobot</b>	<b>Bobot Kriteria</b>
C1	4	Sangat Bagus
	3	Bagus
	2	Cukup
	1	Kurang
C2	5	Sangat Bagus
	4	Bagus
	3	Cukup Bagus
	2	Kurang Bagus
	1	Tidak Bagus
C3	3	Sangat Bagus
	2	Bagus
	1	Kurang Bagus
C4	5	Bagus
	1	Tidak Bagus
C5	3	Sangat Bagus
	2	Bagus
	1	Kurang Bagus

**Tabel 3.2 Indikator Penilaian**

<b>Bobot</b>	<b>Keterangan</b>	<b>Indikator Penilaian</b>
4	Sangat Bagus	Tidak Absen
3	Bagus	Izin 1 / Bulan
2	Cukup	Absen 1 / Bulan
1	Kurang	Lebih Dari 1 Absen / Bulan
5	Sangat Bagus	4 (Salam, Sapa, Sopan, Santun)
4	Bagus	4 (Salam, Sapa, Sopan)
3	Cukup Bagus	4 (Salam, Sapa)
2	Kurang Bagus	4 (Salam)
1	Tidak Bagus	4 (Salam, Sapa, Sopan, Santun) Tidak dilaksanakan
3	Sangat Bagus	Bersih, Sepatu Tidak Kotor, Berpakaian Rapi, Rambut Rapi
2	Bagus	Sepatu Tidak Kotor, Berpakaian Rapi, Rambut Tidak Rapi
1	Kurang Bagus	Berpakaian Tidak Rapi
5	Bagus	Pekerjaan Selesai
1	Tidak Bagus	Pekerjaan Tidak Selesai

3	Sangat Bagus	Jam Kerja Lebih dari 30 Menit
2	Bagus	Jam Kerja Lebih dari 15 Menit
1	Kurang Bagus	Tidak Ada Kelebihan Jam Kerja

### 3.9 Analisis Metode Weighted Product.

Dalam sistem pendukung keputusan proses penilaian jasa yang baik dengan menggunakan metode *Weighted Product* diperlukan kriteria-kriteria dan bobot untuk melakukan perhitungan sehingga akan di dapat alternatif terbaik.

#### 3.9.1 Kriteria, Nilai dan Bobot

Penentuan penilaian jasa dengan metode *Weighted Product* terdapat kriteria yang dibutuhkan untuk menentukan siapa yang akan terpilih sebagai nilai jasa yang terbaik sesuai dengan kriteria yang telah ditentukan. Adapun kriteria yang digunakan adalah sebagai berikut:

1. Kriteria Penilaian yang digunakan sebagai acuan:

Kriteria	Inisial	Bobot
Absen	C1	4
Komunikasi	C2	5
Penampilan	C3	3
Hasil Kerja	C4	5
Loyalitas	C5	3

Nilai Tertinggi untuk penilaian pada kriteria yaitu 100

2. Pengambilan keputusan memberikan bobot referensi:

<b>W</b>	20
----------	----

3. Sebelum dilakukan perbaikan bobot terlebih dahulu maka perangkat diperoleh dari jumlah W, dibagi masing-masing nilai dari kriteria yang sudah ditentukan, dan hasilnya adalah:

<b>W1</b>	0,20
<b>W2</b>	0,25
<b>W3</b>	0,15
<b>W4</b>	0,25
<b>W5</b>	0,15
<b>Total Bobot</b>	1

### 3.9.2 Penyelesaian Perhitungan

Berikut ini adalah hasil perhitungan dari analisa metode *Weighted Product* yang sudah diberikan pada bagian sebelumnya.

Badu	Absen	C1	80	0,2	2,402249
	Komunikasi	C2	95	0,25	3,121986
	Penampilan	C3	85	0,15	1,94721
	Hasil Kerja	C4	80	0,25	2,990698
	Loyalitas	C5	75	0,15	1,910993
				Nilai S1 (Badu)	83,46289
Susi	Absen	C1	70	0,2	2,338943
	Komunikasi	C2	75	0,25	2,942831
	Penampilan	C3	70	0,15	1,891319
	Hasil Kerja	C4	60	0,25	2,783158
	Loyalitas	C5	70	0,15	1,891319
				Nilai S2 (Susi)	68,52549
Joko	Absen	C1	78	0,2	2,390116

	Komunikasi	C2	75	0,25	2,942831
	Penampilan	C3	80	0,15	1,929583
	Hasil Kerja	C4	75	0,25	2,942831
	Loyalitas	C5	80	0,15	1,929583
				Nilai S2 (Joko)	77,06844
Rudi	Absen	C1	80	0,2	2,402249
	Komunikasi	C2	70	0,25	2,892508
	Penampilan	C3	82	0,15	1,936743
	Hasil Kerja	C4	79	0,25	2,981308
	Loyalitas	C5	80	0,15	1,929583
				Nilai S2 (Joko)	77,41674
Adrian	Absen	C1	82	0,2	2,414142
	Komunikasi	C2	82	0,25	3,009217
	Penampilan	C3	80	0,15	1,929583
	Hasil Kerja	C4	85	0,25	3,03637
	Loyalitas	C5	80	0,15	1,929583
				Nilai S2 (Joko)	82,12927
Adit	Absen	C1	90	0,2	2,459509
	Komunikasi	C2	80	0,25	2,990698
	Penampilan	C3	80	0,15	1,929583
	Hasil Kerja	C4	75	0,25	2,942831
	Loyalitas	C5	80	0,15	1,929583
				Nilai S2 (Joko)	80,59597
Suci	Absen	C1	78	0,2	2,390116
	Komunikasi	C2	90	0,25	3,08007
	Penampilan	C3	85	0,15	1,94721
	Hasil Kerja	C4	85	0,25	3,03637
	Loyalitas	C5	80	0,15	1,929583
				Nilai S2 (Joko)	83,98672

Langkah berikutnya adalah menentukan hasil ranking dari metode *Weighted Product*. Setelah nilai vektor S didapat, maka selanjutnya adalah menjumlahkan seluruh S untuk menghitung V, perhitungannya sebagai berikut:

A.	V1 (Badu)	83,5							=	83,5	=	0,1509
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
B.	V2 (Susi)	68,5							=	68,5	=	0,1239
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
C.	V3 (Joko)	77,1							=	77,1	=	0,1393
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
D.	V4 (Rudi)	77,4							=	77,4	=	0,1399
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
E.	V5 (Adrian)	82,1							=	82,1	=	0,1485
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
F.	V6 (Adit)	80,6							=	80,6	=	0,1457
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		
G.	V7 (Suci)	84,0							=	84,0	=	0,1518
		83,5	68,5	77,1	77,4	82,1	80,6	84,0		553,2		

Maka dari perhitungan tersebut, dapat diperoleh nilai terbesar terdapat pada V7(Suci), maka yang berhak mendapatkan pegawai terbaik adalah Suci.

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

Menu yang terdapat di dalam aplikasi ada berupa seorang admin dan *user* (pengguna). Sebelum mengaplikasikan aplikasi pengguna harus membuka menu yang terdapat pada program aplikasi. Menu tersebut terdiri dari Beranda, Info, Profil, dan SPK WP. Pembuatan program aplikasi adalah perancangan *interface* dan penulisan kode program sesuai dengan sistem yang telah dirancang. Untuk membuat program aplikasi penentuan pegawai terbaik menggunakan *Microsoft Visual Basic.NET 2010*.

#### **4.1 Kebutuhan Sistem**

Kebutuhan sistem penting dipersiapkan agar program aplikasi nantinya dapat berjalan dengan baik dan benar. Kebutuhan sistem terdiri dari kebutuhan perangkat lunak dan perangkat keras.

##### **4.1.1 Perangkat Lunak**

Pembuatan program aplikasi adalah perancangan *interface* dan penulisan kode program sesuai dengan sistem yang telah dirancang. Untuk membuat program aplikasi ada beberapa *software* pendukung yang penting, antara lain:

- Microsoft Windows 10
- Microsoft Visual Basic.Net 2010
- Snipping Tool

- Microsoft Excel 2019
- Microsoft Word 2019

#### **4.1.2 Perangkat Keras**

Sistem yang telah terkomputerisasi ini dapat dijalankan apabila telah dilakukan beberapa hal, yaitu proses instalasi sudah dilakukan serta *hardware* yang mendukung dalam menjalankan program ini. Spesifikasi *hardware* yang dibutuhkan untuk mengimplementasikan sistem agar dapat berjalan dengan baik adalah sebagai berikut:

- Processor Intel i5 2.7 GHz
- RAM 4GB
- HDD 500GB
- Mouse
- Keyboard
- Printer Canon G2000

#### **4.2 Implementasi Sistem**

Hasil pembuatan program aplikasi dapat dilihat pada implementasi sistem tersebut. Tampilan pada sistem pendukung keputusan ini dibuat dengan simple dan mudah digunakan sehingga pengguna tidak mengalami kendala-kendala dalam menjalankan program aplikasi tersebut.

#### 4.2.1 Hasil Tampilan Beranda

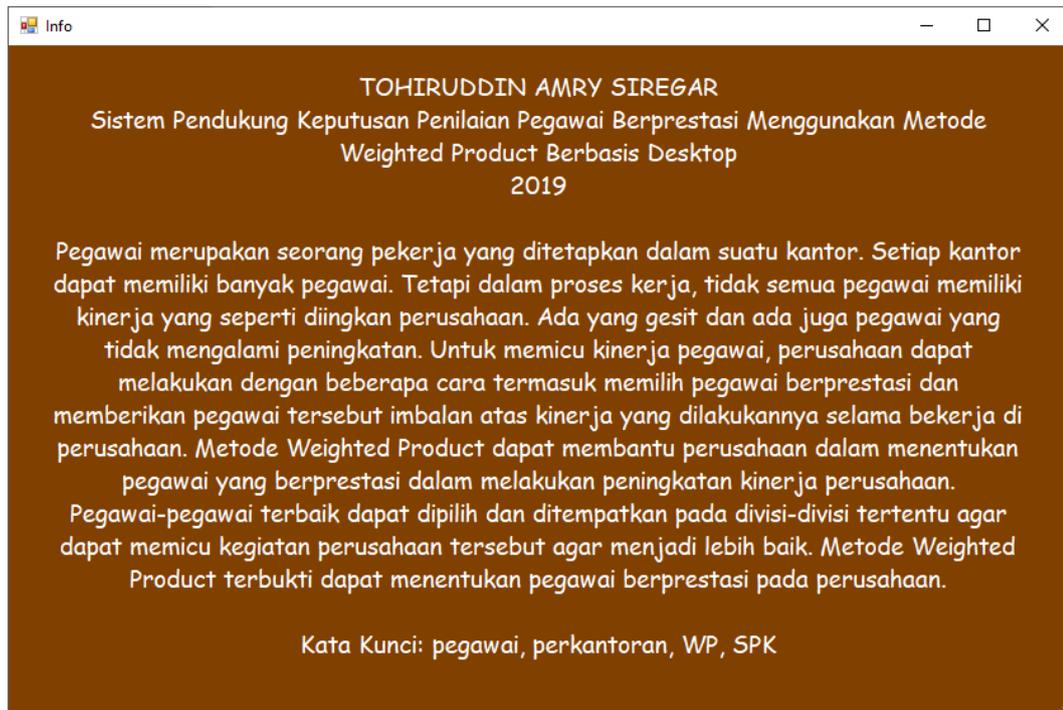
Beranda merupakan menu yang pertama akan tampil jika program aplikasi dieksekusi. Gambar 4.1 adalah tampilan menu beranda.



**Gambar 4.1 Halaman Beranda**

#### 4.2.2 Hasil Tampilan Menu Info

Halaman info adalah menu yang menampilkan penjelasan singkat tentang penelitian ini. Informasi dikemas dalam suatu abstrak yang menjelaskan rumusan, tujuan dan manfaat serta hasil yang diperoleh secara singkat dan padat. Gambar 4.2 adalah tampilan menu info.



**Gambar 4.2 Halaman Info**

### **4.2.3 Hasil Tampilan Menu Profil**

Halaman Profil adalah tampilan tentang penulis. Halaman ini menampilkan informasi tentang nama, NPM, fakultas dan program studi. Gambar 4.3 adalah tampilan Profil.



**Gambar 4.3 Halaman Profil**

#### **4.2.4 Halaman Menu *Weighted Product***

Halaman ini merupakan halaman dimana sistem pendukung keputusan metode *Weighted Product* melakukan perangkingan dan menentukan pegawai terbaik. Gambar 4.4 adalah hasil tampilan menu *Weighted Product*.

Weighted Product

Bobot Absen

Bobot Komunikasi

Bobot Penampilan

Bobot Hasil Kerja

Bobot Loyalitas

Jumlah

Proses Weighted Product

Reset Form

	Pegawai	Absen	Komunikasi	Penampilan	Hasil Kerja	Loyalitas	Vektor S	Vektor V
▶ 1	Badu	80	95	85	80	75		
2	Susi	70	75	70	60	70		
3	Joki	78	75	80	75	80		
4	Rudi	80	70	82	79	80		
5	Adnan	82	82	80	85	80		
6	Adit	90	80	80	75	80		
7	Suci	78	90	85	85	80		
*								

**Gambar 4.4** Halaman *Weighted Product*

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Berdasarkan hasil program aplikasi sistem pendukung keputusan penentuan pegawai berprestasi menggunakan metode *Weighted Product*, maka didapat beberapa kesimpulan, antara lain:

1. Dengan diterapkannya metode *Weighted Product* pada proses menentukan pegawai berprestasi sesuai dengan kinerja, dapat membantu perusahaan dalam menentukan pegawai-pegawai mereka.
2. Bobot preferensi dapat memberikan variasi hasil terhadap alternatif yang diolah.
3. Metode *Weighted Product* memberikan tingkat akurasi yang baik dan sesuai dengan hasil yang diperoleh pada penentuan pegawai terbaik yang dilakukan secara manual.

#### **5.2 Saran**

Berikut adalah saran dari penulis agar Sistem Pendukung Keputusan Penilaian Pegawai Berprestasi Menggunakan Metode *Weighted Product* ini dapat bermanfaat dan dikembangkan menjadi lebih baik lagi:

1. Kriteria yang digunakan sebaiknya ditambahkan agar memberikan tingkat akurasi yang baik.

2. Bobot preferensi dapat dikembangkan sehingga memberikan akurasi dan hasil keputusan yang lebih baik.
3. Hendaknya program aplikasi dapat dikembangkan menjadi berbasis *web* dan *online*.

## DAFTAR PUSTAKA

- Barus, S., Sitorus, V. M., Napitupulu, D., Mesran, M., & Supiyandi, S. (2018). Sistem Pendukung Keputusan Pengangkatan Guru Tetap Menerapkan Metode Weight Aggregated Sum Product Assesment (WASPAS). *Jurnal Media Informatika Budidarma*, 2(2).
- Batubara, Supina, Sri Wahyuni, and Eko Hariyanto. "Penerapan Metode Certainty Factor Pada Sistem Pakar Diagnosa Penyakit Dalam." *Seminar Nasional Royal (SENAR)*. Vol. 1. No. 1. 2018.
- Fachri, Barany, Agus Perdana Windarto, and Ikhsan Parinduri. "Penerapan Backpropagation dan Analisis Sensitivitas pada Prediksi Indikator Terpenting Perusahaan Listrik." *JEPIN (Jurnal Edukasi dan Penelitian Informatika)* 5.2 (2019): 202-208.
- Ginting, G., Fadlina, M., Siahaan, A. P. U., & Rahim, R. (2017). Technical approach of TOPSIS in decision making. *Int. J. Recent Trends Eng. Res*, 3(8), 58-64.
- Hatta, H. R., Rizaldi, M., & Khairina, D. M. (2016). Penerapan Metode Weighted Product Untuk Pemilihan Lokasi Lahan Baru Pemakaman Muslim Dengan Visualisasi Google Maps. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 2(3), 85–94. <https://doi.org/10.25077/TEKNOSI.v2i3.2016.85-94>
- Hung, N. V., van Hung, P., & Anh, B. T. (2018). Database Design For E-Governance Applications: A Framework For The Management Information Systems Of The Vietnam Commitee For Ethnic Minority Affairs (CEMA). *International Journal of Civil Service Reform and Practice*, 3(1).
- Jogiyanto, H. M. (2016). *Analisis Dan Desain Sistem Informasi, Pendekatan Terstruktur Teori Dan Praktek Aplikasi Bisnis*. Andi Offset.
- Keen, P. G. W., & Scott-Morton, M. S. (1978). *Decision Support Systems: An Organizational Perspective*. Addison-Wesley.
- Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 77. <https://doi.org/10.25126/jtiik.201851610>
- Kusrini. (2006). *Sistem Pakar Teori dan Aplikasi*. Andi Offset.
- Kusrini. (2007). *Strategi Perancangan dan Pengelolaan Basis Data*. Andi Offset.
- Kurniawan, H. (2018). Pengenalan Struktur Baru untuk Web Mining dan
- Khairul, K., IlhamiArsyah, U., Wijaya, R. F., & Utomo, R. B. (2018, September). implementasi augmented reality sebagai media promosi penjualan rumah. In *Seminar Nasional Royal (SENAR)* (Vol. 1, No. 1, pp. 429-434).

- Personalisasi Halaman Web. *Jurnal Teknik dan Informatika*, 5(2), 13-19.
- Putera, A., Siahaan, U., & Rahim, R. (2016). Dynamic key matrix of hill cipher using genetic algorithm. *Int. J. Secur. Its Appl*, 10(8), 173-180.
- Putra, Randi Rian. "implementasi metode backpropagation jaringan saraf tiruan dalam memprediksi pola pengunjung terhadap transaksi." *JurTI (Jurnal Teknologi Informasi)* 3.1 (2019): 16-20.
- Ladjamudin, A.-B. bin. (2017). *Analisis dan Desain Sistem Informasi*. Graha Ilmu.
- Lee, C. (2014). *Buku Pintar Pemrograman Visual Basic 2010*. Elex Media Komputindo.
- Mufizar, T. (2018). *Implementasi Metode Weighted Product (WP) Pada Sistem Pendukung Keputusan Seleksi Calon Karyawan BPJS Kesehatan Tasikmalaya*.
- Nofriansyah, D. (2014). *Konsep Data Mining vs Sistem Pendukung Keputusan*. Deepublish.
- Raymond Mcleod, J., & Schell, G. P. (2007). Sistem Informasi Manajemen. In *Riset Akuntansi Dan Bisnis* (Vol. 7, Issue 1). <https://rudiatko.files.wordpress.com/2008/10/kuliahsim-1-2.pdf>
- Rahim, R., Supiyandi, S., Siahaan, A. P. U., Listyorini, T., Utomo, A. P., Triyanto, W. A., ... & Khairunnisa, K. (2018, June). TOPSIS Method Application for Decision Support System in Internal Control for Selecting Best Employees. In *Journal of Physics: Conference Series* (Vol. 1028, No. 1, p. 012052). IOP Publishing.
- Sukmawati, R., & Priyadi, Y. (2019). Perancangan Proses Bisnis Menggunakan UML Berdasarkan Fit/Gap Analysis Pada Modul Inventory Odoo. *INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi*, 3(2),
- Sari, R. D., Supiyandi, A. P. U., Siahaan, M. M., & Ginting, R. B. (2017). A Review of IP and MAC Address Filtering in Wireless Network Security. *Int. J. Sci. Res. Sci. Technol*, 3(6), 470-473.
- Sidik, A. P. (2018). Algoritma RSA dan Elgamal sebagai Algoritma Tambahan untuk Mengatasi Kelemahan Algoritma One Time Pad pada Skema Three Pass Protocol.
- Siahaan, A. P. U., Aryza, S., Nasution, M. D. T. P., Napitupulu, D., Wijaya, R. F., & Arisandi, D. (2018). Effect of matrix size in affecting noise reduction level of filtering. 104. <https://doi.org/10.29407/intensif.v3i2.12697>
- Technopedia. (2019). *Unified Modeling Language (UML)*. Technopedia. <https://www.techopedia.com/definition/3243/unified-modeling-language-uml>
- Turban, E., Aronson, J. E., & Liang, T. (2005). *Decision Support Systems and Intelligent Systems*. Andi.

- Tasril, V., Wijaya, R. F., & Widya, R. (2019). aplikasi pintar belajar bimbingan dan konseling untuk siswa sma berbasis macromedia flash. *Jurnal Informasi Komputer Logika*, 1(3).
- UTM. (2019). *Concept: Use-Case Model*. Univesidad Technologica de La Mixteca. [http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use\\_case\\_model\\_CD178AF9.html](http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model_CD178AF9.html)
- Wasserkrug, S., Dalvi, N., Munson, E. V., Gogolla, M., Sirangelo, C., Fischer-Hübner, S., Ives, Z., Velegakis, Y., Bevan, N., Jensen, C. S., & Snodgrass, R. T. (2019). Unified Modeling Language. In *Encyclopedia of Database Systems* (pp. 3232–3239). Springer US. [https://doi.org/10.1007/978-0-387-39940-9\\_440](https://doi.org/10.1007/978-0-387-39940-9_440)
- Wijaya, Rian Farta, et al. "Aplikasi Petani Pintar Dalam Monitoring Dan Pembelajaran Budidaya Padi Berbasis Android." *Rang Teknik Journal* 2.1 (2019).
- Yakub. (2012). *Pengantar Sistem Informasi*. Graha Ilmu.