



**PENERAPAN SISTEM PENUNJANG KEPUTUSAN DALAM
PENERIMAAN ANGGOTA KOPERASI RSU DENGAN
MENGUNAKAN METODE AHP DAN PEMOGRAMAN PHP MYSQL**

Disusun dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir Dalam
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH

NAMA : FITRIANI
N.P.M : 1614370417
PROGRAM STUDI : SISTEM KOMPUTER

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2021**

PENGESAHAN SKRIPSI

JUDUL : PENERAPAN SISTEM PENUNJANG KEPUTUSAN DALAM PEMILIHAN
ANGGOTA KOPERASI RSU DENGAN MENGGUNAKAN METODE AHP DAN
PEMOGRAMAN PHP MYSQL

NAMA : FITRIANI
N.P.M : 1614370417
FAKULTAS : SAINS & TEKNOLOGI
PROGRAM STUDI : Sistem Komputer
TANGGAL KELULUSAN : 31 Mei 2021

DIKETAHUI

DEKAN



Hamdani, ST., MT.

KETUA PROGRAM STUDI



Eko Hariyanto, S.Kom., M.Kom

**DISETUJUI
KOMISI PEMBIMBING**

PEMBIMBING I



Dr Muhammad Iqbal, S.Kom., M.Kom.

PEMBIMBING II



Radiyan Rahim, S.Kom., M.Kom

SURAT PERNYATAAN ORISINALITAS SKRIPSI

Yang bertanda tangan dibawah ini :

Nama : FITRIANI
NPM : 16143703417
Program Studi : SISTEM KOMPUTER
JudulSkripsi : PENERAPAN SISTEM PENUNJANG KEPUTUSAN
DALAM PENERIMAAN ANGGOTA KOPERASI RSU
DENGAN MENGGUNAKAN METODE AHP DAN
PEMOGRAMAN PHP MYSQL

Menyatakan dengan sebenarnya bahwa skripsi yang ditulis ini tidak mempunyai persamaan dengan skripsi lain.

Demikian pernyataa ini dibuat tanpa adanya paksaan dari pihak manapun. Apabila pernyataan ini tidak benar, maka akan diberikan sanksi oleh pimpinan fakultas.

Medan, 2September2021

Yang MembuatPernyataan



FITRIANI

16143703417



KARTU BEBAS PRAKTIKUM
Nomor. 1245/BL/LAKO/2021

Yang bertanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

Nama : FITRIANI
N.P.M. : 1614370417
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

Benar dan telah menyelesaikan urusan administrasi di Laboratorium Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 27 April 2021
Ka. Laboratorium

Melva Sari Panjaitan, S. Kom., M.Kom.





YAYASAN PROF. DR. H. KADIRUN YAHYA
PERPUSTAKAAN UNIVERSITAS PEMBANGUNAN PANCA BUDI
Jl. Jend. Gatot Subroto KM. 4,5 Medan Sunggal, Kota Medan Kode Pos 20122

SURAT BEBAS PUSTAKA
NOMOR: 4076/PERP/BP/2021

Kepala Perpustakaan Universitas Pembangunan Panca Budi menerangkan bahwa berdasarkan data pengguna perpustakaan atas nama saudara/i:

Nama : FITRIANI
N.P.M. : 1614370417
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

Bahwasannya terhitung sejak tanggal 27 April 2021, dinyatakan tidak memiliki tanggungan dan atau pinjaman buku sekaligus tidak lagi terdaftar sebagai anggota Perpustakaan Universitas Pembangunan Panca Budi Medan.

Medan, 27 April 2021
Diketahui oleh,
Kepala Perpustakaan,



Sugiarjo, S.Sos., S.Pd.I

SURAT KETERANGAN PLAGIAT CHECKER

Dengan ini saya Ka.LPMU UNPAB menerangkan bahwa surat ini adalah bukti pengesahan dari LPMU sebagai pengesah proses plagiat checker Tugas Akhir/ Skripsi/Tesis selama masa pandemi *Covid-19* sesuai dengan edaran rektor Nomor : 7594/13/R/2020 Tentang Pemberitahuan Perpanjangan PBM Online.

Demikian disampaikan.

NB: Segala penyalahgunaan/pelanggaran atas surat ini akan di proses sesuai ketentuan yang berlaku UNPAB.



No. Dokumen : PM-UJMA-06-02	Revisi : 00	Tgl Eff : 23 Jan 2019
-----------------------------	-------------	-----------------------

Plagiarism Detector v. 1864 - Originality Report 4/22/2021 10:35:23 AM

Analyzed document: FITRIANI_1614370417_SISTEM KOMPUTER.docx Licensed to: Universitas Pembangunan Panca Budi_License03

Comparison Preset: Rewrite Detected language:
Check type: Internet Check



Detailed document body analysis:
Relation chart:



Category	Percentage
Original	30.00%
Plagiarism	68.00%
Unchecked	2.00%

Distribution graph:



Top sources of plagiarism: 42



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808
MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : FITRIANI
NPM : 1614370417
Program Studi : Sistem Komputer
Jenjang Pendidikan : Strata Satu
Dosen Pembimbing : Dr Muhammad Iqbal, S.Kom., M.Kom.
Judul Skripsi : Penerapan sistem penunjang keputusan dalam pemilihan anggota koperasi RSU dengan metode AHP dan pemograman PHP MySQL

Tanggal	Pembahasan Materi	Status	Keterangan
04 Juli 2020	Acc Sempro	Revisi	
07 Februari 2021	Acc Bab II	Revisi	
07 Februari 2021	Acc Bab 3	Revisi	
07 Februari 2021	Acc Bab IV	Revisi	
07 Februari 2021	Acc Bab V	Revisi	
07 Februari 2021	Acc Seminar Hasil	Disetujui	
24 April 2021	Acc Sidang Meja Hijau	Disetujui	
07 Juni 2021	Acc Jilid	Disetujui	

Medan, 18 Juni 2021
Dosen Pembimbing,



Dr Muhammad Iqbal, S.Kom., M.Kom.



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808

MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : FITRIANI
 NPM : 1614370417
 Program Studi : Sistem Komputer
 Jenjang : Strata Satu
 Pendidikan :
 Dosen Pembimbing : Radian Rahim, S.Kom., M.Kom
 Judul Skripsi : Penerapan sistem penunjang keputusan dalam pemilihan anggota koperasi RSU dengan metode AHP dan pemrograman PHP MySQL

Tanggal	Pembahasan Materi	Status	Keterangan
28 Mei 2020	acc	Revisi	
05 September 2020	Acc sempro	Disetujui	
29 Oktober 2020	lanjutkan bab berikutnya	Revisi	
16 November 2020	lanjut bab berikutnya	Revisi	
06 Februari 2021	acc semhas	Disetujui	
18 April 2021	acc sidang meja hijau	Disetujui	
07 Juni 2021	selesai jilid	Disetujui	

Medan, 18 Juni 2021
 Dosen Pembimbing,



Radian Rahim, S.Kom., M.Kom



UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

Saya yang bertanda tangan di bawah ini :

Nama Lengkap : FITRIANI
 Tempat/Tgl. Lahir : SEI SKALA / 27 Juli 1978
 Nomor Pokok Mahasiswa : 1614370417
 Program Studi : Sistem Komputer
 Konsentrasi : Sistem Kendali Komputer
 Jumlah Kredit yang telah dicapai : 141 SKS, iPK 3.33
 Nomor Hp : 085261025327
 Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut :

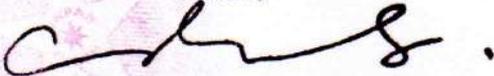
No.	Judul
1.	Penerapan sistem penunjang keputusan dalam pemilihan anggota koperasi RSU dengan metode AHP dan pemrograman PHP MySQL

Catatan : Diisi Oleh Dosen Jika Ada Perubahan Judul

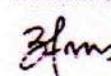
*Coret Yang Tidak Perlu

Medan, 23 Februari 2021

Rektor I,


 (Cahyo Pramono, S.E., M.M.)

Pemohon,


 (Fitriani)

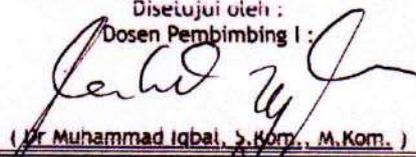
Tanggal :

Disahkan oleh :
 Dekan


 (Hamdani, S.E., M.I.)

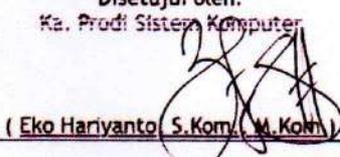
Tanggal :

Disetujui oleh :
 Dosen Pembimbing I :


 (Dr. Muhammad Iqbal, S.Kom., M.Kom.)

Tanggal :

Disetujui oleh :
 Ka. Prodi Sistem Komputer


 (Eko Hariyanto, S.Kom., M.Kom.)

Tanggal :

Disetujui oleh :
 Dosen Pembimbing II :


 (Radian Rahim, S.Kom., M.Kom.)

No. Dokumen: FM-UPBA-18-02

Revisi: 0

Tgl. Eff: 22 Oktober 2018

Hal : Permohonan Meja Hijau

Medan, 26 April 2021
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama : FITRIANI
Tempat/Tgl. Lahir : Sei Skala / 27 Juli 1978
Nama Orang Tua : SUHARDI
N. P. M : 1614370417
Fakultas : SAINS & TEKNOLOGI
Program Studi : Sistem Komputer
No. HP : 085261025327
Alamat : Dusun Kantil Desa Padang Brahrang Kecamatan Selesai
Kabupaten Langkat

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Penerapan sistem penunjang keputusan dalam pemilihan anggota koperasi RSU dengan metode AHP dan pemograman PHP MySQL**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntun ujian perbaikan nilai mata kuliah untuk perbaikan indek prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkripnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 exemplar untuk penguji (bentuk dan warna penjiilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangani dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	0
2. [170] Administrasi Wisuda	: Rp.	1,500,000
3. [202] Bebas Pustaka	: Rp.	100,000
4. [221] Bebas LAB	: Rp.	5,000
Total Biaya	: Rp.	1,605,000

Ukuran Toga :

S

Diketahui/Disetujui oleh :



Hamdani, ST., MT.
Dekan Fakultas SAINS & TEKNOLOGI

Hormat saya



FITRIANI
1614370417

Catatan :

- 1. Surat permohonan ini sah dan berlaku bila ;
 - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
 - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2. Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.

SURAT PERNYATAAN

Saya Yang Bertanda Tangan Dibawah Ini :

Nama : FITRIANI
N. P. M : 1614370417
Tempat/Tgl. Lahir : Sei Skala / 27 Juli 1978
Alamat : Dusun Kantil Desa Padang Brahrang Kecamatan Selesai Kabupaten Langkat
No. HP : 085261025327
Nama Orang Tua : SUHARDI/WARINA FAUZIAH GINTING
Fakultas : SAINS & TEKNOLOGI
Program Studi : Sistem Komputer
Judul : Penerapan sistem penunjang keputusan dalam pemilihan anggota koperasi RSU dengan metode AHP dan pemrograman PHP MySQL

Bersama dengan surat ini menyatakan dengan sebenar - benarnya bahwa data yang tertera diatas adalah sudah benar sesuai dengan ijazah pada pendidikan terakhir yang saya jalani. Maka dengan ini saya tidak akan melakukan penuntutan kepada UNPAB. Apabila ada kesalahan data pada ijazah saya.

Demikianlah surat pernyataan ini saya buat dengan sebenar - benarnya, tanpa ada paksaan dari pihak manapun dan dibuat dalam keadaan sadar. Jika terjadi kesalahan, Maka saya bersedia bertanggung jawab atas kelalaian saya.



SURAT PERNYATAAN

Saya Yang bertanda tangan dibawah ini:

Nama : FITRIANI
Npm : 16143703417
Program Studi : SISTEM KOMPUTER
Judul Skripsi : PENERAPAN SISTEM PENUNJANG KEPUTUSAN DALAM
PENERIMAAN ANGGOTA KOPERASI RSU DENGAN
MENGUNAKAN METODE AHP DAN PEMOGRAMAN
PHP MYSQL

Dengan Ini Menyatakan Bahwa :

1. tugas akhir atau skripsi saya bukan hasil plagiat
2. saya tidak akan menuntut perbaikan nilai indeks prestasi kumulatif (ipk) setelah ujian sidang meja hijau
3. skripsi saya dapat dipublikasikan oleh pihak lembaga, dan saya tidak akan menuntut akibat publikasih tersebut.

demikian pernyataan ini saya buat dengan sebenar-benarnya, Terimakasih.

Medan, Juli 2021
Yang Membuat Pernyataan



Fitriani



UNIVERSITAS PEMBANGUNAN PANCA BUDI

BIRO PELAYANAN ADMINISTRASI AKADEMIK (BPAA)

Jl. Jend. Gatot Subroto Km. 4,5 Telp. (061)8455571 Fax. (061)8458077 Po. Box 1099

MEDAN – INDONESIA

website: www.pancabudi.ac.id email : unpab@pancabudi.ac.id

SURAT REKOMENDASI DOKUMEN PERMOHONAN SIDANG MEJA HIJAU

Kepala Biro Pelayanan Administrasi Akademik UNPAB menerangkan bahwa surat ini adalah bukti pengesahan dari BPAA sebagai proses rekomendasi dokumen permohonan sidang meja hijau selama masa pandemi Covid-19 sesuai dengan edaran Rektor Nomor : 7594/13/R/2020 Tentang Pemberitahuan Perpanjangan PBM Online.

Dengan ini disampaikan bahwa Saudara/i :

Nama : Fitriani
NPM : 1614370417
Fakultas/Jurusan : Sains dan Teknologi / Sistem Komputer
No Hp : 085261025327
Ukuran Toga : S

Telah dilakukan pemeriksaan dokumen permohonan sidang meja hijau dan sesuai dengan persyaratan yang ditentukan UNPAB.

Demikian disampaikan untuk dapat dipergunakan semestinya.

Medan, 30 April 2021

Ka. BPAA

Wirda Fitriani, S.Kom., M.Kom

NB : Segala penyalahgunaan atau pelanggaran atas surat ini akan diproses sesuai ketentuan yang berlaku di UNPAB

ABSTRAK

Pada saat ini masih banyak kecurangan yang sering terjadi dalam dunia kerja mulai dari penerimaan karyawan, kenaikan jabatan dan juga penilaian terhadap karyawan, sehingga banyak kesalahan yang timbul contohnya menempatkan posisi jabatan pada orang yang tidak ada kompetensinya sehingga dapat merugikan sebuah organisasi atau perusahaan. Penggunaan metode Analytical Hierarchy Process (AHP) telah cukup meluas diberbagai aplikasi baik dibidang industri, elektronika, pendidikan dan lain sebagainya. Selain itu metode Analytical Hierarchy Process (AHP) dapat pula diterapkan untuk penentuan penerimaan karyawan koperasi RSUD. Dengan sistem pendukung keputusan yang menerapkan metode AHP berbasis web dalam pengambilan keputusan sangat membantu panitia dalam menghasilkan keputusan yang mutlak. Dengan telah ditentukannya kriteria dalam penerimaan anggota baru dapat membantu panitia menghasilkan anggota baru yang berbakat pada setiap bidang keahliannya. Dengan adanya aplikasi sistem pengambilan keputusan ini yang menerapkan metode AHP dapat menghasilkan anggota baru yang berdedikasi tinggi serta memiliki loyalitas yang besar pada Koperasi RSUD sesuai dengan kriteria yang telah ditetapkan dalam penerimaan.

***Kata kunci** : Karyawan, SPK, AHP, PHP, Mysql*

KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Allah SWT, Tuhan Yang Maha Esa yang telah memberikan rahmat-Nya kepada peneliti, sehingga Skripsi ini dapat diselesaikan oleh peneliti tepat pada waktunya dengan judul **“PENERAPAN SISTEM PENUNJANG KEPUTUSAN DALAM PENERIMAAN ANGGOTA KOPERASI RSU DENGAN MENGGUNAKAN METODE AHP DAN PEMOGRAMAN PHP MYSQL ”**.

Skripsi ini disusun dengan maksud guna memenuhi salah satu syarat memperoleh gelar Sarjana Komputer pada Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan. Dalam penyusunan skripsi ini penulis masih banyak ketidak sempurnaan atas apa yang penulis lakukan tetapi penulis menyadari sebagai manusia memiliki keterbatasan kemampuan dan hal ini tidak dapat penulis hindari, penulis berharap adanya saran dan kritik demi sempurnanya skripsi ini .

Pada kesempatan ini, penulis menyampaikan rasa terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. Bapak Dr. H. Muhammad Isa Indrawan, SE, MM, selaku Rektor Universitas Pembangunan Panca Budi Medan.
2. Bapak Hamdani, ST., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
3. Bapak Eko Haryanto, S.Kom., M.Kom., selaku Ketua Program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
4. Bapak Dr. Muhammad Iqbal, S.Kom., M.Kom., selaku Dosen Pembimbing I yang telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini. dan Bapak Radian Rahim, S.Kom., M.Kom. selaku Dosen pembimbing II saya yang juga telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini
5. Bapak/Ibu Dosen beserta seluruh staf Universitas Pembangunan Panca Budi Medan. yang telah mendidik dan membimbing penulis selama mengikuti perkuliahan
6. Teristimewa kepada Kedua Orang Tua Ayah dan Ibu dan terima kasih atas semua pengorbanannya , yang telah banyak memberikan bimbingan dan bantuan baik moril maupun materil selama peneliti mengikuti pendidikan hingga selesainya Tugas Akhir ini.
7. Kepada seluruh rekan-rekan di program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan yang telah memberikan dukungan moril kepada penulis.

Penulis menyadari bahwa Skripsi ini masih kurang sempurna. Oleh karena itu, penulis sangat mengharapkan dan menghargai saran maupun kritikan dari pembaca dan semua pihak yang mengarah kepada perbaikan Skripsi ini. Akhir kata, penulis berharap semoga penyusunan Skripsi ini dapat bermanfaat bagi pembaca.

Medan, 20 Maret 2021

FITRIANI
1614370417

DAFTAR ISI

	Halaman
COVER	
LEMBAR JUDUL	
LEMBAR PENGESAHAN	
ABSTRAK	
KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vii
DAFTAR LAMPIRAN	viii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	3
BAB II LANDASAN TEORI.....	4
2.1 Rekayasa Perangkat Lunak.....	4
2.1.1 Tahapan Umum Rekayasa Perangkat Lunak	5
2.1.2 Tantangan Rekayasa Perangkat Lunak.....	6
2.2 Sistem Pendukung Keputusan (Decision Support System).....	7
2.2.1 Pengertian	7
2.2.2 Keuntungan DSS (Decision Support System	7
2.2.3 System	8
2.2.4 Model	9
2.2.5 Karakteristik dan Kemampuan Decision Support System.....	12
2.6 Metode dalam DSS (Decision Support System).....	15
2.7 Analytic Hierarchy Process (AHP)	18
2.8 PHP	21
2.8.1 Definisi dan Sejarah PHP.....	21
2.8.2 Dasar-Dasar Pemrograman PHP.....	23
2.9 Database MySQL.....	26
2.9.1 Definisi Database MySQL	26
2.10 UML (Unified Modelling Language).....	27
2.10.1 Use Case Diagram	28
2.10.2 Class Diagram	33
2.10.3 Sequence Diagram	35

2.10.4 Collaboration Diagram	37
2.10.5 State Machine Diagram	38
2.10.6 Activity Diagram	39
BAB III METODOLOGI PENELITIAN	40
3.1 Kerangka Penelitian	40
3.2 Tahap penelitian	41
3.2.1 Literatur	41
3.2.2 Pengumpulan Data	41
3.3. Analisis sistem	42
3.3.1 Kebutuhan Sistem	42
3.3.2 Analisa Sistem yang Sedang Berjalan	43
3.3.3 Analisis Sistem yang Akan Dibangun	44
3.4 Penerapan AHP	44
3.4.1 Hierarki	44
3.4.2 Alternatif	45
3.4.3 Kriteria	46
3.4.4 Penilaian Kriteria AHP	47
3.4.5 Penilaian Sub Kriteria AHP	52
3.4.6 Hasil Perhitungan	53
3.5 Struktur Database	55
3.6 Perancangan aplikasi	57
3.6.1 Perancangan Aplikasi Menggunakan Use Case Diagram	57
3.6.3 Perancangan Aplikasi Menggunakan Activity Diagram	60
3.6.4 Perancangan Aplikasi Menggunakan Sequence Diagram	62
3.6.5 Perancangan Aplikasi Menggunakan Collaboration Diagram	63
3.6.6 Perancangan Aplikasi Menggunakan Deployment Diagram	64
3.7 Perancangan antar muka	65
BAB IV IMPLEMENTASI DAN PENGUJIAN	72
4.1 Implementasi Sistem	72
4.1.1 Kebutuhan Hardware	72
4.1.2 Kebutuhan Software	73
4.2 Implementasi program.	73
4.2.1 Proses Instalasi XAMPP 1.7.4	73
4.2.2 Tampilan Database Sistem Pengambilan Keputusan	77
4.2.3 Proses Instalasi Notepad++	78
4.2.5 Form Lihat Promosi	79
4.2.6 Form Pemesanan Produk	80
4.2.7 Form Input Bukti Pembayaran	80
4.2.8 Layout Halaman Admin	81
4.2.9 Form Login Admin	82
4.2.10 Form Input Data Kategori	82

4.2.11	Form Input Data Produk.....	83
4.2.12	Form Input Data Pengiriman Produk.....	83
4.2.13	Form Laporan Data Customer.....	84
4.2.14	Form Laporan Data Produk.....	85
4.2.15	Form Laporan Data Pengiriman.....	85
4.2.16	Form Laporan Penjualan Per Hari.....	86
4.2.17	Form Laporan Data Penjualan Per Bulan.....	87
4.2.18	Form Laporan Data Penjualan Per Tahun.....	87
BAB V PENUTUP		90
5.1	Kesimpulan	90
5.2	Keterbatasan Sistem.....	90
5.3	Saran	91

DAFTAR PUSTAKA

LAMPIRAN-LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Karakteristik dan Kemampuan DSS	12
Gambar 3.1 Kerangka Kerja Penelitian	40
Gambar 3.2 Struktur Hirarki SPK Penerimaan	45
Gambar 3.3 Use Case Diagram	58
Gambar 3.4 Activity Diagram Admin	60
Gambar 3.5 Activity Diagram User	61
Gambar 3.6 Sequence Diagram Admin Kelola User	62
Gambar 3.7 Sequence Diagram Admin Kelola Sub Kriteria	63
Gambar 3.8 Collaboration Diagram User	64
Gambar 3.9 Deployment Diagram	65
Gambar 3.10 Menu Awal Website.....	66
Gambar 3.11 Halaman Index Admin	67
Gambar 3.12 Halaman Panduan	67
Gambar 3.13 Form Registrasi User	68
Gambar 3.14 Form Kriteria.....	69
Gambar 3.15 Form Seleksi	69
Gambar 3.16 Halaman Perbandingan Kriteria	70
Gambar 3.17 Halaman Hasil Keputusan	70
Gambar 3.18 Halaman Index User.....	71
Gambar 3.19 Halaman Data User	71
Gambar 3.20 Desain Form Promosi.....	55
Gambar 3.22 Desain Form Input Pembayaran.....	55
Gambar 3.23 Desain Form Input Kategori Produk	56
Gambar 3.24 Desain Form Input Produk	57
Gambar 3.25 Desain Form Input Pengiriman.....	58
Gambar 3.26 Desain Tampilan Website.....	59
Gambar 3.27 Tampilan Halaman Admin	60
Gambar 4.1 Welcome XAMPP Setup.....	74
Gambar 4.2 Pilih Lokasi Install XAMPP.....	74
Gambar 4.3 Opsi Penginstalan XAMPP	75
Gambar 4.4 Proses Instalasi XAMPP	75
Gambar 4.5 Proses Instalasi XAMPP Selesai	76
Gambar 4.6 XAMPP Control Panel.....	76
Gambar 4.7 Tampilan XAMPP.....	77
Gambar 4.8 Tampilan Database db_penerimaan	78
Gambar 4.9 Welcome Notepad++ Setup	78
Gambar 4.10 License Agreement.....	79
Gambar 4.11 Pemilihan Lokasi Instalasi	79
Gambar 4.12 Pemilihan Instalasi Komponen	80
Gambar 4.13 Pemilihan Instalasi Komponen	80
Gambar 4.14 Proses Ekstraksi.....	81
Gambar 4.15 Instalasi Notepad++ Selesai	81

Gambar 4.16 Halaman Awal Website.....	82
Gambar 4.17 Halaman Index Admin	83
Gambar 4.18 Halaman Panduan.....	83
Gambar 4.19 Halaman Registrasi User.....	84
Gambar 4.20 Halaman Kriteria Pengambilan Keputusan	84
Gambar 4.21 Halaman Sub Kriteria Pengambilan Keputusan.....	85
Gambar 4.22 Halaman Daftar User.....	85
Gambar 4.23 Halaman Seleksi.....	86
Gambar 4.24 Halaman Proses Keputusan Kriteria	86
Gambar 4.25 Halaman Proses Keputusan Sub Kriteria	87
Gambar 4.26 Halaman Proses Keputusan Perbandingan	87
Gambar 4.27 Halaman Hasil Seleksi	88
Gambar 4.28 Halaman Index User.....	88
Gambar 4.29 Halaman Lengkapi Data User	89
Gambar 4.30 Halaman Seleksi User	89

DAFTAR TABEL

	Halaman
Tabel 2.1 Skala Penilaian Perbandingan Berpasangan	21
Tabel 2.2 Simbol – Simbol Use Case Diagram	29
Tabel 2.3 Simbol – Simbol Class Diagram	33
Tabel 2.4 Simbol – Simbol Sequence Diagram	34
Tabel 2.5 Simbol – Simbol Collaboration Diagram	35
Tabel 2.6 Simbol – Simbol State Machine Diagram	37
Tabel 2.7 Simbol – Simbol Activity Diagram	38
Tabel 2.8 Simbol – Simbol Deployment Diagram	39
Tabel 3.1 Skala Penilaian Perbandingan Berpasangan	48
Tabel 3.2 Matriks Perbandingan Berpasangan Kriteria	48
Tabel 3.3 Perhitungan prioritas nilai kriteria	50
Tabel 3.4 Perhitungan matrik kriteria penjumlahan setiap baris.....	50
Tabel 3.5 Perhitungan CR kriteria.....	51
Tabel 3.6 Index Random (IR)	52
Tabel 3.7 Matrik perbandingan berpasangan sub kriteria	52
Tabel 3.8 Matrik prioritas sub kriteria	52
Tabel 3.9 Hasil penjumlahan setiap baris.....	53
Tabel 3.10 Perhitungan CR sub kriteria.....	53
Tabel 3.11 Hasil perhitungan setiap prioritas kriteria dan subkriteria.....	54
Tabel 3.12 Contoh Data	54
Tabel 3.13 Hasil Perhitugan Contoh Data.....	54
Tabel 3.14 Tabel Admin	55
Tabel 3.15 Tabel User	55
Tabel 3.16 Tabel Kriteria	56
Tabel 3.17 Tabel Sub Kriteria	57
Tabel 3.18 Deskripsi Aktor	58
Tabel 3.19 Deskripsi Use Case	59

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Biografi Penulis	L1
Lampiran 2. Lembar Pengesahan Skripsi	L2
Lampiran 3. Lembar Permohonan Mengajukan Judul Skripsi.....	L3
Lampiran 4. Lembar Bimbingan Skripsi Doping Satu.....	L4
Lampiran 5. Lembar Bimbingan Skripsi Doping Dua.....	L5
Lampiran 6. Lembar Riset.....	L6
Lampiran 7. Permohonan Meja Hijau.....	L7
Lampiran 8. Plagiat Checker.....	L8
Lampiran 9. Kartu Bebas Praktikum.....	L9
Lampiran10.Listing Program	L10

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pada saat ini masih banyak kecurangan yang sering terjadi dalam dunia kerja mulai dari penerimaan karyawan, kenaikan jabatan dan juga penilaian terhadap karyawan, sehingga banyak kesalahan yang timbul contohnya menempatkan posisi jabatan pada orang yang tidak ada kompetensinya sehingga dapat merugikan sebuah organisasi atau perusahaan, masalah klasik ini bisa di kurangi dengan bantuan teknologi yang membuat system penilaian lebih mudah cepat dan transparan.

Untuk mendapatkan karyawan yang berkualitas, serta untuk mendapatkan karyawan bertalenta terbaik maka ditentukan berdasarkan aspek-aspek yang di tentukan saat pengambilan keputusan, dalam pengambilan keputusan juga harus menggunakan metode yang sesuai.

Penggunaan metode Analytical Hierarchy Process (AHP) telah cukup meluas diberbagai aplikasi baik dibidang industri, elektronika, pendidikan dan lain sebagainya. Selain itu metode Analytical Hierarchy Process (AHP) dapat pula diterapkan untuk penentuan penerimaan karyawan koperasi RSU

Analytical Hierarchy Process (AHP) merupakan proses dalam pengambilan keputusan dengan menggunakan perbandingan berpasangan (kemampuan) .Hasil perhitungan AHP yang diterapkan ini akan menghasilkan keluaran nilai intensitas prioritas pelamar sehingga pelamar yang memiliki nilai tertinggi mendapat peluang tertinggi untuk diterima

Berdasarkan uraian diatas, maka penelitian ini berjudul , “ Penerapan Sistem Penunjang Keputusan Dalam Penerimaan Anggota Koperasi RSU Dengan Menggunakan Metode Ahp Dan Pemograman Php Mysql “

1.2. Perumusan Masalah

Berdasarkan latar belakang di atas, maka di dapat kan beberapa rumusan masalah sebagai berikut :

1. Bagaimana membuat sistem pendukung keputusan menggunakan metode AHP dalam memberikan solusi untuk menyelesaikan permasalahan didalam penerimaan karyawan ?
2. Bagaimana mengembangkan sistem yang dapat menghemat waktu dalam proses penerimaan karyawan ?
3. Bagaimana membantu pihak HRD dalam penerimaan Karyawan yang tepat dan cocok untuk bekerja di perusahaan?

1.3. Batasan Masalah

Permasalahan yang diangkat pada penelitian ini memiliki batasan masalah sebagai berikut :

1. Pengambilan data dilakukan dan sesuai ketentuan yang ada di Koperasi RSU
2. Penilaian yang digunakan untuk penerimaan karyawan adalah kriteria-kriteria yang di tentukan oleh perusahaan.

1.4. Tujuan Penelitian

Dalam penulisan ini ada beberapa tujuan yang akan dicapai, diantaranya :

1. Menerapkan metode AHP dalam system penunjang keputusan.
2. sistem pendukung keputusan terkomputerisasi yang mampu membantu pihak perusahaan atau instansi – instansi yang terkait.
3. Membantu dalam penetapan penerimaan karyawan di dalam instansi.

1.5. Manfaat Penelitian

1. Digunakan sebagai dasar pengambilan keputusan yang terstruktur untuk mendukung prioritas penerimaan karyawan pada RSUD
2. Membantu HRD untuk memutuskan penerimaan karyawan sesuai dengan penilaian kriteria karyawan.
3. Menambah wawasan penulis sendiri tentang sistem pendukung keputusan.

BAB II

LANDASAN TEORI

2.1 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak merupakan suatu disiplin ilmu yang bertujuan mengembangkan sistem perangkat lunak yang efektif dari segi biaya. Perangkat lunak bersifat abstrak dan tidak nyata. Perangkat lunak tidak terbuat dari unsur, mengikuti hukum fisika atau proses manufaktur. Dalam beberapa hal, kenyataan ini menyederhanakan rekayasa perangkat lunak karena tidak ada pembatasan fisik terhadap potensi perangkat lunak. Akan tetapi, dalam hal lain, tidak adanya batasan natural ini berarti bahwa perangkat lunak dengan mudah dapat menjadi sangat kompleks dan demikian sangat sulit dipahami (Dra.Yuhilza Hanum,M.Eng.,2003:4).

Rekayasa perangkat lunak merupakan suatu media yang sering digunakan dalam perancangan sebuah *software* atau perangkat lunak. Perangkat lunak banyak dibangun dan pada akhirnya sering tidak digunakan karena tidak memenuhi kebutuhan pelanggan atau bahkan karena masalah non-teknis seperti keengganan pemakai perangkat lunak (*user*) untuk mengubah cara kerja dari manual keotomatis, atau ketidak mampuan *user* menggunakan komputer. Oleh karena itu dibutuhkan suatu media agar perangkat lunak yang dibangun tidak menjadi perangkat lunak yang tidak *friendly* atau tidak berguna. Media yang sering digunakan dalam perancangan suatu perangkat lunak yaitu “Rekayasa Perangkat Lunak (*Software Engineering*)”.

Rekayasa perangkat lunak (*software engineering*) merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin (Rosa A.S dan M. Shalahuddin 2013:4).

(Rosa A.S dan M. Shalahuddin, 2013:5) menyatakan bahwa rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria sebagai berikut :

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan.
2. Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi.
3. Efisien dari segi sumber daya dan pengguna.
4. Kemampuan untuk dipakai sesuai dengan kebutuhan.

Dari kriteria diatas maka perangkat lunak yang baik adalah perangkat lunak yang mampu memenuhi kebutuhan pengguna (*user*) atau berorientasi pada pemakai perangkat tersebut.

2.1.1 Tahapan Umum Rekayasa Perangkat Lunak

(Rosa A.S dan M. Shalahuddin, 2013:5) membagi proses rekayasa perangkat lunak menjadi tiga bagian tanpa melihat area dari aplikasi, ukuran proyek proyek perangkat lunak, dan kompleksitas perangkat lunak yang dibuat, yaitu:

1. Proses pendefinisian yang fokus pada “*what*”. Artinya yaitu harus mencari tahu atau mengidentifikasi informasi apa yang harus diproses, seperti apa fungsi dan peformansi yang diinginkan, seperti apa perilaku sistem yang diinginkan, apa kriteria validasi yang dibutuhkan atau mendefenisikan sistem.
2. Proses pengembangan yang fokus pada “*how*”. Artinya selama tahap pembangunan perangkat lunak seorang perekayasa perangkat lunak berusaha untuk mendefenisikan hal berikut:
 - a. Bagaimana data distrukturkan

- b. Bagaimana fungsi-fungsi yang dibutuhkan diimplementasikan didalam arsitektur perangkat lunak
 - c. Bagaimana detail prosedural diimplementasikan
 - d. Bagaimana karakter antar muka tampilan
 - e. Bagaimana desain ditranslasikan kebahasa pemograman, dan
 - f. Bagaimana pengujian akan dijalankan
3. Proses pendukung. Artinya terasosiasi pada perbaikan kesalahan (*error*), adaptasi yang dibutuhkan pada lingkungan perangkat lunak yang terlibat dan perbaikan yang terjadi akibat perubahan kebutuhan *user*. Proses pendukung terdiri dari empat fase, yaitu;
- a. Koreksi
 - b. Adaptasi
 - c. Perbaikan
 - d. Pencegahan

2.1.2 Tantangan Rekayasa Perangkat Lunak

(Rosa A.S dan M. Shalahuddin, 2013:7) menyatakan tantangan yang dihadapi dalam proses rekayasa perangkat lunak yaitu sebagai berikut :

1. Tantangan warisan dimana perangkat lunak dikembangkan selama bertahun-tahun oleh orang-orang yang berbeda, hal ini dapat menyebabkan ketidakpahaman atau perubahan tujuan pembuatan perangkat lunak.
2. Tantangan *heterogenitas*, dimana perangkat lunak harus dapat beradaptasi dengan teknologi yang terus berkembang dengan semakin luasnya lingkungan distribusi perangkat lunak.

3. Tantangan pengiriman bahwa perangkat lunak dengan skala besar dan kompleks sekalipun dapat sampai ketangan pemakai (*user*) dengan cepat dan kualitas tetap terjaga.

2.2 Sistem Pendukung Keputusan (*Decision Support System*)

2.2.1 Pengertian

Sistem berbasis model yang terdiri dari prosedur-prosedur dalam pemrosesan data dan pertimbangan untuk membantu manajer dalam pengambilan keputusan (Irfan Subakti, 2002:19. “Sistem Pendukung Keputusan (*Decision Support System*)”. Institut Teknologi Sepuluh Nopember Surabaya).

(Irfan Subakti, 2002:19. “Sistem Pendukung Keputusan (*Decision Support System*)”. Institut Teknologi Sepuluh Nopember Surabaya) menyatakan ada juga definisi yang menyatakan bahwa *Decision Support System (DSS)* adalah sistem berbasis komputer yang terdiri dari 3 komponen interaktif, yaitu :

1. *Sistem bahasa*, mekanisme yang menyediakan komunikasi diantara *user* dan berbagai komponen dalam *DSS*.
2. *Knowledge system*, penyimpanan knowledge domain permasalahan yang ditanamkan dalam *DSS*, baik sebagai data ataupun prosedur.
3. *Sistem pemrosesan permasalahan*, link diantara dua komponen, mengandung satu atau lebih kemampuan memanipulasi masalah yang dibutuhkan untuk pengambilan keputusan.

2.2.2 Keuntungan DSS (*Decision Support System*)

(Irfan Keuntungan DSS (*Decision Support System* Subakti, 2002:21. “Sistem Pendukung Keputusan (*Decision Support System*)”. Institut Teknologi Sepuluh Nopember Surabaya) menyatakan keuntungan dari DSS ini adalah :

1. Mampu mendukung pencarian solusi dari masalah yang kompleks.

2. Respon cepat pada situasi yang tak diharapkan dalam kondisi yang berubah-ubah.
3. Mampu untuk menerapkan berbagai strategi yang berbeda pada konfigurasi berbeda secara cepat dan tepat.
4. Pandangan dan pembelajaran baru.
5. Memfasilitasi komunikasi.
6. Meningkatkan kontrol manajemen dan kinerja.
7. Menghemat biaya.
8. Keputusan lebih tepat.
9. Meningkatkan efektifitas manajerial, menjadikan manajer dapat bekerja lebih singkat dengan sedikit usaha.
10. Meningkatkan produktivitas analisis.

2.2.3 System

(Irfan Subakti, 2002:9. “Sistem Pendukung Keputusan (*Decision Support System*)”. Institut Teknologi Sepuluh Nopember Surabaya) menyatakan sistem adalah kumpulan dari objek-objek seperti orang, resources, konsep, dan prosedur yang ditujukan untuk melakukan fungsi tertentu atau memenuhi suatu tujuan. Sebuah sistem terdiri dari :

1. *Input* adalah semua elemen yang masuk ke sistem. Contohnya adalah bahan baku yang masuk ke pabrik kimia, pasien yang masuk ke rumah sakit, atau memasukan data ke komputer.
2. *Proses* adalah proses transformasi elemen-elemen dari *input* menjadi *output*.
3. *Output* adalah produk jadi atau hasil dari suatu proses sistem.
4. *Feedback* adalah aliran informasi dari komponen *output* kepengambil keputusan yang memperhitungkan *output* atau kinerja sistem.

5. *Environment* adalah lingkungan dari sebuah sistem terdiri dari berbagai elemen yang terletak diluar *input*, *output*, atau pun proses. Namun, mereka dapat mempengaruhi kinerja dan tujuan sistem. Bila suatu elemen memiliki hubungan dengan tujuan sistem serta pengambil keputusan secara signifikan tak mungkin memanipulasi elemen ini, maka elemen tersebut harus dimasukkan sebagai bagian dari *environment*.
6. *Boundary* adalah pemisah antara suatu sistem dengan *environment*-nya. Sistem ada didalam *boundary*, dimana *environment* ada diluarnya.
7. *Closed System* (Sistem Tertutup) adalah sistem yang mempresentasikan derajat kemandirian dari sistem. Sistem ini secara penuh mandiri, tak tergantung sama sekali.
8. *Open System* (Sistem Terbuka) sangat tergantung pada lingkungannya. Sistem ini menerima input (informasi, energi, material) dari lingkungannya dan bisa juga memberikan *output*-nya kembali ke lingkungan tersebut.

Ada dua ukuran utama dari sistem, adalah :

1. *Efektivitas* adalah derajat seberapa banyak tujuan sistem tercapai. Ini mengacu pada hasil atau output dari suatu sistem.
2. *Efisiensi* adalah ukuran penggunaan input (resources) untuk mencapai tujuan.

2.2.4 Model

Persyaratan *user* harus harus dituliskan dalam bahasa natural karena harus di mengerti oleh orang-orang yang bukan pakar teknik. Walaupun demikian, persyaratan sistem yang lebih rinci dapat dinyatakan dengan cara yang lebih teknis. Salah satu teknik yang banyak digunakan adalah pendokumentasian spesifikasi sistem sebagai suatu set model sistem. Model-model ini merupakan representasi grafis yang

mendeskripsikan masalah yang akan ditangani dan sistem yang akan dikembangkan. Karena digunakan representasi grafis, model seringkali lebih dipahami dibandingkan dengan deskripsi bahasa naturalnya yang rinci tentang persyaratan sistem. Representasi ini juga merupakan jembatan antara proses analisis dan perancangan. (Dra.Yuhilza Hanum,M.Eng.,2003:141).

(Irfan Subakti, 2002:10. “Sistem Pendukung Keputusan (*Decision Support System*)”. Institut Teknologi Sepuluh Nopember Surabaya) menyatakan karakteristik utama dari DSS adalah adanya kemampuan pemodelan. Model adalah representasi sederhana atau penggambaran kenyataan. Terdapat 3 (tiga) jenis model, yaitu :

1. *Iconic* adalah replika fisik dari sistem, biasanya dalam skala tertentu dari bentuk aslinya. GUI pada OOPL adalah contoh dari model ini.
2. *Analog* adalah tak seperti sistem yang sesungguhnya tetapi berlaku seperti itu. Lebih abstrak dari pada model *Iconic* dan merupakan representasi simbolis dari kenyataan.
3. *Matematis* adalah kompleksitas hubungan dalam sistem organisasi tak dapat dipresentasikan dengan *iconic* atau *analog*, karena kalau pun bisa akan memakan waktu lama dan sulit.

1. Keuntungan Pemodelan

Di bawah ini adalah alasan utama mengapa DSS menggunakan model :

- a. Biaya analisis model lebih murah dari pada percobaan yang dilakukan pada sistem yang sesungguhnya.
- b. Model memungkinkan untuk menyingkat waktu. Operasi bertahun-tahun dapat disimulasikan dalam hitungan menit komputer.

- c. Memanipulasi model (perubahan variabel) lebih mudah dilakukan dari pada bila diterapkan pada sistem nyata. Selanjutnya percobaan yang dilakukan akan lebih mudah dilakukan dan tak mengganggu jalannya operasi harian organisasi.
- d. Akibat yang ditimbulkan dari adanya kesalahan-kesalahan sewaktu proses *trial and error* lebih kecil dari pada penggunaan model langsung di sistem nyata.
- e. Lingkungan sekarang makin berada dalam ketidak pastian. Penggunaan pemodelan menjadikan seorang *manager* dapat menghitung resiko yang ada pada proses-proses tertentu.
- f. Penggunaan model matematis bisa menjadi analisis dilakukan pada kemungkinan-kemungkinan solusi yang banyak sekali, bahkan bisa tidak terhitung. Dengan adanya komunikasi dan teknologi canggih sekarang ini, manajer akan sering kali memiliki alternatif-alternatif pilihan.
- g. Model meningkatkan proses pembelajaran dan meningkatkan pelatihan.

2. Proses Pemodelan

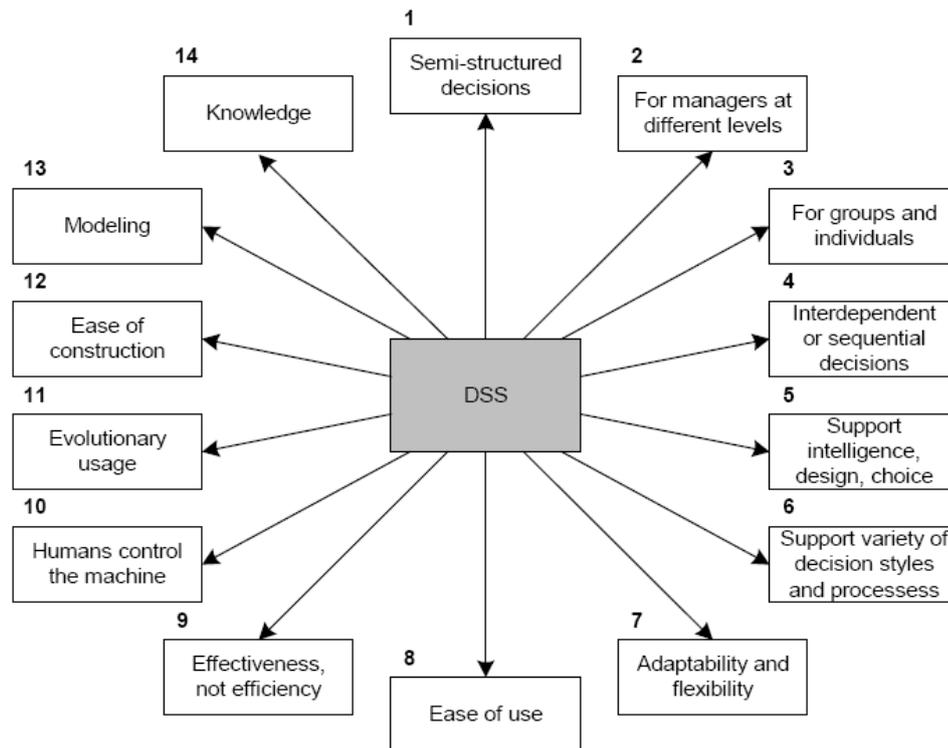
Berikut ini adalah proses yang terjadi dalam pemodelan :

- a. *Trial and error* dengan sistem nyata. Tapi ini tak berjalan bila :
 - Terlalu banyak alternatif untuk dicoba.
 - Akibat samping dari error yang terjaid besar pengaruhnya.
 - Lingkungan itu sendiri yang selalu berubah.
- b. *Simulation*
- c. *Optimistic*
- d. *Heuristic*.

2.2.5 Karakteristik dan Kemampuan *Decision Support System*

Dibawah ini adalah karakteristik dan kemampuan ideal dari suatu *DSS*, yaitu :
(Irfan Subakti, 2002:20. “Sistem Pendukung Keputusan (*Decision Support System*)”.

Institut Teknologi Sepuluh Nopember Surabaya).



Sumber : Modul Kuliah Sistem Pendukung Keputusan – Irfan Subakti

Gambar 2.1 Karakteristik dan Kemampuan DSS

1. DSS menyediakan dukungan bagi pengambil keputusan utamanya pada situasi semi terstruktur dan tak terstruktur dengan memadukan pertimbangan manusia dan informasi terkomputerisasi. berbagai masalah tak dapat diselesaikan (atau tak dapat diselesaikan secara memuaskan) oleh sistem terkomputerisasi lain, seperti EDP atau MIS, tidak juga dengan metode atau *tool* kuantitatif standar.
2. Dukungan disediakan untuk berbagai level manajerial yang berbeda, mulai dari pimpinan puncak sampai manajer lapangan.

3. Dukungan disediakan bagi individu dan juga bagi group. berbagai masalah organisasional melibatkan pengambilan keputusan dari orang dalam *group*. Untuk masalah yang strukturnya lebih sedikit seringkali hanya membutuhkan keterlibatan beberapa individu dari departemen dan level organisasi yang berbeda.
4. DSS menyediakan dukungan ke berbagai keputusan yang berurutan atau saling berkaitan.
5. DSS mendukung berbagai *fase* proses pengambilan keputusan: *intelligence*, *design*, *choice* dan *implementation*.
6. DSS mendukung berbagai proses pengambilan keputusan dan *style* yang berbeda-beda, ada kesesuaian diantara DSS dan atribut pengambil keputusan individu (contohnya *vocabulary* dan *style* keputusan).
7. DSS selalu bisa beradaptasi sepanjang masa. Pengambil keputusan harus reaktif, mampu mengatasi perubahan kondisi secepatnya dan beradaptasi untuk membuat DSS selalu bisa menangani perubahan ini. DSS adalah *fleksibel*, sehingga *user* dapat menambahkan, menghapus, mengkombinasikan, mengubah, atau mengatur kembali elemen-elemen dasar (menyediakan respon cepat pada situasi yang tak diharapkan). Kemampuan ini memberikan analisis yang tepat waktu dan cepat setiap saat.
8. DSS mudah untuk digunakan. *User* harus merasa nyaman dengan sistem ini. *User-friendliness*, *fleksibilitas*, dukungan grafis terbaik, dan antarmuka bahasa yang sesuai dengan bahasa manusia dapat meningkatkan *efektivitas* DSS. Kemudahan penggunaan ini diimplikasikan pada mode yang *interaktif*.
9. DSS mencoba untuk meningkatkan *efektivitas* dari pengambilan keputusan (akurasi, jangka waktu, kualitas), lebih daripada *efisiensi* yang bisa diperoleh (biaya membuat keputusan, termasuk biaya penggunaan komputer).

10. Pengambil keputusan memiliki kontrol menyeluruh terhadap semua langkah proses pengambilan keputusan dalam menyelesaikan masalah. DSS secara khusus ditujukan untuk mendukung dan tak menggantikan pengambil keputusan. Pengambil keputusan dapat menindak lanjuti rekomendasi komputer sembarang waktu dalam proses dengan tambahan pendapat pribadi atau pun tidak.
11. DSS mengarah pada pembelajaran, yaitu mengarah pada kebutuhan baru dan penyempurnaan sistem, yang mengarah pada pembelajaran tambahan, dan begitu selanjutnya dalam proses pengembangan dan peningkatan DSS secara berkelanjutan.
12. *User* / pengguna harus mampu menyusun sendiri sistem yang sederhana. Sistem yang lebih besar dapat dibangun dalam organisasi *user* tadi dengan melibatkan sedikit saja bantuan dari spesialis di bidang *Information Systems* (IS).
13. DSS biasanya mendayagunakan berbagai model (standar atau sesuai keinginan *user*) dalam menganalisis berbagai keputusan. Kemampuan pemodelan ini menjadikan percobaan yang dilakukan dapat dilakukan pada berbagai konfigurasi yang berbeda. Berbagai percobaan tersebut lebih lanjut akan memberikan pandangan dan pembelajaran baru.
14. DSS dalam tingkat lanjut dilengkapi dengan komponen *knowledge* yang bisa memberikan solusi yang efisien dan efektif dari berbagai masalah yang pelik.

2.2.6 Metode dalam DSS (*Decision Support System*)

A. *Fuzzy Multiple Attribute Decision Making* (FMADM)

(Heri Sulistiyo,2009:2) menyatakan suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari FMADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan

dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subyektif, pendekatan obyektif dan pendekatan integrasi antara subyektif dan obyektif. Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan (Heri Sulistiyo, 2009:2). Ada beberapa metode yang dapat digunakan untuk menyelesaikan masalah FMADM. antara lain:

- a. *Simple Additive Weighting Method (SAW)*
- b. *Weighted Product (WP)*
- c. ELECTRE
- d. *Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)*
- e. *Analytic Hierarchy Process (AHP).*

B. *Simple Additive Weighting (SAW)*

Metode SAW (*Simple Additive Weighting*) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada (Heri Sulistiyo, 2009:2).

C. *Perference Ranking Organization Method for Enrichment Evaluation (PROMETHEE)*

(Dewi Safitri Hutabarat, 2013:14) menyatakan metode *Promethee* termasuk kedalam kelompok pemecahan masalah *Multi Criteria Decision Making (MCDM)* atau pengambilan keputusan kriteria majemuk yang merupakan disiplin ilmu yang sangat penting dalam pengambilan keputusan atau suatu masalah yang memiliki lebih dari satu kriteria (multikriteria). Menurut Brans dan Marcchal (1999:15), *Promethee* yang merupakan singkatan dari *Preference Ranking Organization Method for Enrichment*

Evaluation adalah metode *outranking* yang menawarkan cara yang fleksibel dan sederhana kepada *user* (pembuat keputusan) untuk menganalisis masalah-masalah multikriteria. Prinsip yang digunakan adalah penetapan prioritas alternatif yang telah ditetapkan berdasarkan pertimbangan dengan kaidah dasar:

Max $\{f_1(x), f_2(x), f_3(x), \dots, f_i(x), \dots, f_k(x)\}$ Ix Dimana k adalah sejumlah kumpulan alternatif dan f_i ($i=1,2,\dots, k$) merupakan nilai / ukuran relatif kriteria untuk masing-masing alternatif. Termasuk dalam keluarga dari metode *outranking* yang dikembangkan oleh B.Roy (Brans et. al,1999) dan meliputi dua fase:

1. Membangun hubungan *outranking* dari K, dimana K adalah sejumlah kumpulan alternaif.
2. Eksploitasi dari hubungan ini memberikan jawaban optimasi kriteria dalam paradigma permasalahan multi kriteria.

Dalam *fase* pertama, nilai hubungan *outranking* berdasarkan pertimbangan dominasi masing-masing kriteria. Indeks preferensi ditentukan dan nilai *outranking* secara grafis disajikan berdasarkan preferensi dari pembuat keputusan.

D. Metode AHP (*Analytical Hierarchy Process*)

AHP adalah sebuah metode memecah permasalahan yang kompleks atau rumit dalam situasi yang tidak terstruktur menjadi bagian-bagian komponen. Mengatur

bagian atau variabel ini menjadi suatu bentuk susunan *hierarki*, kemudian memberikan nilai *numerik* untuk penilaian subjektif terhadap kepentingan relatif dari setiap variabel dan *mensintesis* penilaian untuk variabel mana yang memiliki prioritas tertinggi yang akan mempengaruhi penyelesaian dari situasi tersebut. AHP menggabungkan pertimbangan dan penilaian pribadi dengan cara yang logis dan dipengaruhi imajinasi, pengalaman, dan pengetahuan untuk menyusun *hierarki* dari suatu masalah yang berdasarkan logika, intuisi dan juga pengalaman untuk memberikan pertimbangan. Dalam menyelesaikan permasalahan dengan AHP ada beberapa prinsip dasar yang harus dipahami. (Yosep Agus Pranoto, 2014:15)

E. Metode *Technique For Order Preference by Similiarity to Ideal Solution* (TOPSIS)

(Nuri Guntur Perdana, dkk, 2013:266) *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) didasarkan pada konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif. Langkah-langkah penyelesaian masalah MADM dengan TOPSIS :

- a. Membuat matriks keputusan yang ternormalisasi.
- b. Membuat matriks keputusan yang ternormalisasi terbobot.
- c. Menentukan matriks solusi ideal positif & matriks solusi ideal negatif.
- d. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif & matriks solusi ideal negatif.
- e. Menentukan nilai *preferensi* untuk setiap alternatif.

TOPSIS adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang (1981). TOPSIS menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal

positif dan terjauh dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Euclidean* untuk menentukan kedekatan relatif dari suatu alternatif dengan solusi optimal. Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi negatif-ideal terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut. TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif. Berdasarkan perbandingan terhadap jarak relatifnya, susunan prioritas alternatif bisa dicapai. Metode ini banyak digunakan untuk menyelesaikan pengambilan keputusan. Hal ini disebabkan konsepnya sederhana, mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relatif dari alternatif-alternatif keputusan.

2.7 Analytic Hierarchy Process (AHP)

Ketika keputusan yang akan diambil bersifat kompleks dengan risiko yang besar seperti perumusan kebijakan, pengambil keputusan sering memerlukan alat bantu dalam bentuk analisis yang bersifat ilmiah, logis, dan terstruktur/konsisten. Salah satu alat analisis tersebut adalah berupa *decision making model* (model pembuatan keputusan) yang memungkinkan mereka untuk membuat keputusan untuk masalah yang bersifat kompleks (Hilyah Magdalena, 2012:51).

Ciri khas suatu DSS adalah digunakannya model yang salah satu fungsinya untuk penyederhanaan masalah. AHP yang dikembangkan oleh Thomas L Saaty merupakan model *hierarki* fungsional dengan input utamanya persepsi manusia. Dengan adanya *hierarki* masalah yang kompleks atau tidak terstruktur dipecah dalam sub-sub masalah kemudian disusun menjadi suatu bentuk *hierarki* (Abdul Syukur, dkk, 2010:105).

AHP sering digunakan sebagai metode pemecahan masalah dibanding dengan metode yang lain karena alasan-alasan sebagai berikut : (Hilyah Magdalena, 2012:51).

1. Struktur yang *hierarki*, sebagai konsekuensi dari kriteria yang dipilih, sampai pada subkriteria yang paling dalam.
2. Memperhitungkan *validitas* sampai dengan batas toleransi *inkonsistensi* sebagai kriteria dan alternatif yang dipilih oleh pengambil keputusan.
3. Memperhitungkan daya tahan *output* analisis *sensitivitas* pengambilan keputusan.

Adapun langkah-langkah dalam metode AHP adalah sebagai berikut : (Abdul Syukur, dkk, 2010:105).

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan.
2. Membuat struktur *hierarki*, yang diawali dengan tujuan umum, dilanjutkan dengan sub-sub tujuan, kriteria dan kemungkinan alternatif pada tingkatan kriteria yang paling bawah.
3. Membuat matriks perbandingan berpasangan yang menggambarkan kontribusi *relative* pengaruh setiap elemen terhadap masing-masing tujuan kriteria yang setingkat di atasnya. Perbandingan berdasarkan "*judgement*" dari pengambil keputusan dengan menilai tingkat kepentingan suatu elemen dibandingkan elemen lainnya.
4. Melakukan perbandingan berpasangan sehingga diperoleh *judgment* seluruhnya sebanyak $n \times [(n-1)/4]$.
5. Menghitung nilai *eigen* dan menguji konsistensinya, jika tidak konsisten maka pengambilan data diulangi.
6. Mengulangi langkah 3, 4, dan 5 untuk seluruh tingkat *hierarki*.

7. Menghitung vektor *eigen* dari setiap matriks perbandingan berpasangan. Nilai vektor *eigen* merupakan bobot setiap elemen. Langkah ini untuk mensistensis *judgment* dalam penentuan prioritas elemen-elemen pada tingkat *hierarki* terendah sampai pencapaian tujuan.
8. Memeriksa konsistensi *hierarki*.jika nilainya lebih dari 10% maka penilaian data *judgment* harus diperbaiki.

Dengan naluri manusia dapat memperkirakan besaran sederhana melalui inderanya. Proses yang mudah adalah dengan membandingkan dua hal dengan keakuratan perbandingan yang dapat dipertanggung jawabkan. Untuk menilai perbandingan tingkat kepentingan elemen, Saaty(1980) menetapkan skala *kuantitatif* 1 sampai 9 seperti tabel dibawah ini :

Intensitas Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting dari pada elemen lainnya.
5	Elemen yang satu lebih penting dari pada elemen lainnya
7	Satu elemen jelas lebih mutlak penting dari pada elemen lainnya.
9	Satu elemen mutlak penting dari pada elemen lainnya.
2, 4, 6, 8	Nilai-nilai antara dua nilai pertimbangan yang berdekatan.

Tabel 2.1 Skala penilaian perbandingan berpasangan

Skala nilai diatas digunakan untuk mengisi nilai matrik perbandingan berpasangan yang akan menghasilkan prioritas (bobot/nilai kepentingan) masing-masing kriteria dan subkriteria

2.8 PHP

2.8.1 Definisi dan Sejarah PHP

PHP Merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML. Dikenal juga dengan bahasa pemrograman server side. Dengan menggunakan PHP maka *maintanance* suatu situs web menjadi lebih mudah. Proses update data dapat dilakukan dengan menggunakan *script* PHP. (Betha Sidik, 2012:4).

Penggunaan PHP memungkinkan web dapat dibuat dinamis sehingga *maintanance* situs web tersebut menjadi lebih mudah dan efisien. PHP merupakan software *open source* yang disebarakan dan dilisensikan secara gratis serta dapat di download secara bebas dari situs resminya <http://www.php.net> (Kasiman Peranginangin, 2006:2).

PHP merupakan sebuah bahasa *scripting* sisi server yang menjadi satu dengan HTML. Sebagian besar sintaknya mirip dengan bahasa C, Java, dan Perl. Tujuan bahasa ini diciptakan adalah untuk membantu pemrogram web dalam membuat halaman web dinamis. Banyak *tool* yang dapat digunakan untuk membuat dokumen PHP, mulai dari *text editor* biasa, seperti Notepad, Wordpad, Notepad++, Editplus, dan lain-lain, sampai aplikasi populer untuk PHP seperti Dreamweaver, PHP Desaigner, dan sebagainya. (Riyanto, 2014:9).

PHP merupakan bahasa pemrograman yang paling sering digunakan oleh programming web karena merupakan bahasa pemrograman *open source*, sehingga para programming tidak perlu membeli lisensi untuk membuat aplikasi web. Rasmus Ledorf merupakan orang yang membuat PHP pada tahun 1995. pada waktu itu, nama PHP adalah FI (*Form Interpreted*) yang merupakan sekumpulan *script*, digunakan untuk mengolah data form dari web. Pada perkembangan berikutnya, Rasmus akhirnya melepas kode sumber tersebut dan diberi nama PHP (*Personal Home Page*). Dengan berubahnya kode program menjadi *open source*, maka banyak programming tertarik dalam mengembangkan PHP. (I Komang Setia Buana, 2014:1).

File yang hanya berisi kode HTML tidak mendukung pembuatan aplikasi pendukung yang melibatkan database karena HTML dirancang untuk menyajikan informasi yang bersifat statis. Oleh karena itu, selanjutnya muncul pemikiran untuk membuat suatu perantara yang memungkinkan aplikasi bisa menghasilkan sesuatu yang bersifat dinamis dan berinteraksi dengan *database*. Akhirnya, lahirlah berbagai perantara seperti PHP, ASP, dan JSP. (Abdul Kadir, 2009:4)

Rilis pertama PHP adalah PHP 2.0, pada bulan November 1997. Pada rilis ini, interpreter sudah bisa diimplementasikan ke bahasa C. Di dalam versi ini sudah ditambahkan modul-modul ekstensi yang menyebabkan kemampuan PHP lebih baik. Pada tahun yang sama, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Sehingga pada bulan Juni 1998, perusahaan tersebut akhirnya merilis versi baru dari PHP dengan nama PHP 3.0 dan mengubah kepanjangan PHP menjadi *Hypertext Preprocessor*. Pertengahan tahun 1999, Zend merilis versi baru dari PHP dan diberi nama PHP 4.0. PHP 4.0 merupakan versi PHP yang paling banyak dipakai oleh programming karena bisa digunakan untuk membuat aplikasi web yang lebih kompleks tetapi memiliki kecepatan proses dan

stabilitas yang tinggi. Pada perkembangan berikutnya, Zend akhirnya merilis versi baru lagi dengan nama PHP 5.0 pada bulan Juni 2014. Versi ini merupakan versi mutakhir dari PHP dan sudah menerapkan model pemrograman berorientasi objek yang merupakan model yang banyak dikembangkan di semua bahasa pemrograman. PHP group kini sudah merilis versi baru dari PHP, yaitu versi 5.1.6.

PHP merupakan aplikasi perangkat lunak opensource, dimana kepanjangan PHP adalah *Hypertext Preprocessor* yang diatur dalam aturan *general purpose license* (GPL). Pemrograman PHP merupakan pemrograman yang sangat cocok dikembangkan di lingkungan web karena bisa diletakkan pada *script* HTML ataupun sebaliknya. PHP tergolong sebagai pemrograman web dinamis karena mampu menghasilkan website yang bisa diubah secara terus menerus hasilnya atau kontennya tanpa harus masuk ke dalam *coding* (I Komang Setia Buana, 2014:9)

PHP juga merupakan bahasa pemrograman berbasis server. Ini berarti setiap pemrograman PHP harus diletakkan di server terlebih dahulu, kemudian diterjemahkan oleh web server dan hasilnya dikirimkan ke browser client. Kemampuan dan fitur PHP yang paling mendukung banyak basis data yaitu *MSSQL*, *MySQL*, *Oracle*, dan *Postgre SQL* (I Komang Setia Buana, 2014:9).

2.8.2 Dasar-Dasar Pemrograman PHP

A. Sintak Dasar PHP

Untuk menuliskan sintak pemrograman PHP, membutuhkan program editor text tambahan. Salah satu editor text yang paling umum dipakai adalah notepad++ digunakan untuk mengetikkan sintak PHP. Notepad++ tersebut bisa diunggah secara gratis. Ketika menuliskan program PHP selalu diawali dan diakhiri menggunakan tag khusus. Tag khusus tersebut memiliki 4 cara penulisan, yaitu (I Komang Setia Buana, 2014) :

1. Cara I

```
<? echo "Sintak PHP" ; ?>
```

2. Cara II

```
<? php echo "Sintak PHP" ; ?>
```

3. Cara III

```
<? = echo "Sintak PHP" ; ?>
```

4. Cara IV

```
<% echo "Sintak PHP" ; %>
```

Anda bisa menggunakan cara yang mana saja untuk memulai dan menutup program PHP, namun cara yang ketiga dan keempat tidak disarankan karena ada beberapa server yang tidak mengenali tag tersebut.

B. Tipe Data

Tipe data merupakan jenis data yang digunakan untuk pengolahan di dalam komputer, dimana data ini sangat dibutuhkan di dalam pemrograman komputer. Tipe data diibaratkan sebagai luas dan jenis ruangan yang digunakan untuk menampung barang-barang yang diinginkan. Bagus tidaknya suatu program, bergantung pintar tidaknya seorang programming untuk menentukan tipe data yang mana yang cocok untuk menampung data yang dimasukkan. Jika salah memilih tipe data, maka program akan menjadi error atau akan terjadi pemborosan memori, karena tipe data ini erat kaitannya dengan memori di dalam komputer (I Komang Setia Buana, 2014).

1. Pengenalan Tipe Data

Di dalam membuat program PHP, tidak diperlukan pendeklarasian tipe data pada suatu variabel, seperti dilakukan pada bahasa pemrograman lain, tetapi lebih ditentukan oleh *runtime* program PHP. Sehingga jika ingin memberikan nilai pada suatu variabel, kita tidak perlu mendeklarasikan tipe data, cukup hanya dengan mengisi tanda dolar (\$) di depan sebuah variabel.

Cara penulisan tipe data (I Komang Setia Buana, 2014) :

\$nama_variabel

Oleh karena penulisan tipe data hanya menggunakan tanda dolar (\$), jenis tipe data tidak akan diketahui. Karena itu, untuk mengetahui tipe data apa saja yang kita gunakan pada saat menuliskan sintak, bisa menggunakan fungsi *gettype* ().

2. Jenis-Jenis Tipe Data

PHP mendukung delapan jenis tipe data yang dibagi atas tiga bagian (I Komang Setia Buana, 2014), yaitu :

1. Tipe data dasar
 - a. *Boolean*
 - b. *Integer*
 - c. *Float*
 - d. *String*
2. Tipe *compound* / tipe campuran
 - a. *Array*
 - b. *Object*
 - a. Tipe *special* / tipe *Resource*
 - b. *Null*
3. khusus

C. Kelebihan PHP Dari Bahasa Pemrograman Lain

Kelebihan PHP dari bahasa pemrograman lain adalah (I Komang Setia Buana, 2014) :

1. Bahasa pemrograman php adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung php dapat ditemukan dimana – mana dari mulai IIS sampai dengan *apache*, dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, php adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.9 Database MySQL

2.9.1 Definisi Database MySQL

MySQL merupakan database server yang paling sering digunakan dalam pemrograman PHP. MySQL digunakan untuk menyimpan data di dalam database dan memanipulasi data-data yang diperlukan. Manipulasi data tersebut berupa, menambah, mengubah, dan menghapus data yang berada dalam database. (I Komang Setia Buana, 2014:2:)

Database sering didefinisikan sebagai kumpulan data yang terkait. Secara teknis, yang berada dalam sebuah database adalah sekumpulan tabel atau objek lain (indeks, view, dan lain-lain). Tujuan utama pembuatan database adalah untuk

memudahkan dalam mengakses data. Data dapat ditambahkan, diubah, dihapus, atau dibaca dengan relatif mudah dan cepat (Abdul Kadir, 2009:14).

MySQL merupakan software yang tergolong database server dan bersifat *open source*. *Open source* menyatakan bahwa software ini dilengkapi dengan *source* kode, selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di internet secara gratis. Hal menarik lainnya adalah MySQL juga bersifat multiplatform. MySQL dapat dijalankan pada berbagai sistem operasi. (Abdul Kadir,2009:15).

2.10 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah keluarga notasi grafis yang didukung oleh meta model tunggal yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berbasis objek (OOP). Definisi ini merupakan definisi yang sederhana. Pada kenyataannya, pendapat orang-orang tentang UML berbeda satu sama lain. Hal ini dikarenakan oleh sejarahnya sendiri dan oleh perbedaan persepsi tentang apa yang membuat sebuah proses rancang bangun perangkat lunak efektif. (Martin Fowler, 2004:1)

UML merupakan bahasa *visual* untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung (Rosa A.S dan M. Shalahuddin, 2013:137).

UML muncul karena adanya kebutuhan pemodelan *visual* untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi,

penggunaan UML tidak terbatas pada teknologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

UML terdiri atas 13 diagram yang dikelompokkan dalam tiga kategori berdasarkan sudut pandangnya, yaitu (Rosa A.S dan M. Shalahuddin, 2013:140):

1. Structure Diagrams

- a. Class Diagram*
- b. Object Diagram*
- c. Component Diagram*
- d. Composite Diagram*
- e. Package Diagram*
- f. Deployment Diagram*

2. Behavior Diagrams

- a. Use Case Diagram*
- b. Activity Diagram*
- c. State Machine Diagram*

3. Intraction Diagrams

- a. Sequence Diagram*
- b. Communication Diagram*
- c. Timing Diagram*
- d. Interaction Diagram*

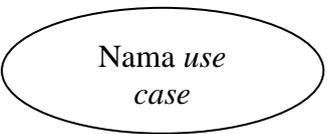
2.10.1 Use Case Diagram

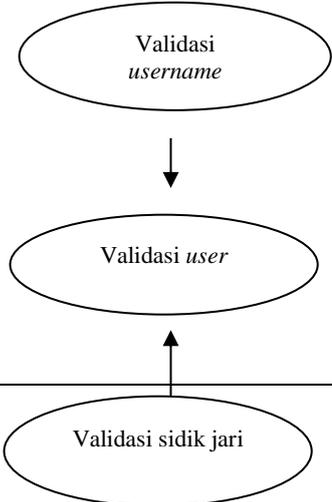
Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang dibuat (Rosa A.S dan M. Shalahuddin, 2013:155).

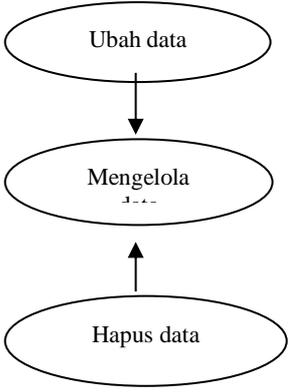
Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefenisikan sesimpel mungkin dan dapat dipahami. Terdapat dua hal utama pada *use case*, yaitu (Rosa A.S dan M. Shalahuddin, 2013:155):

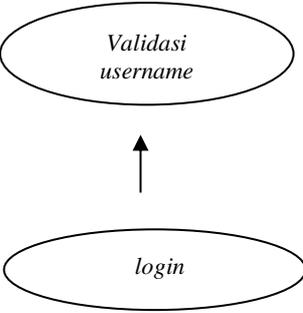
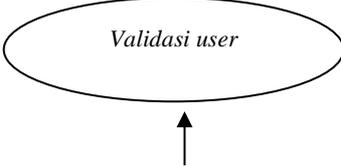
1. *Actor* merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibangun. Jadi, walaupun simbol dari *actor* adalah gambar orang, tapi *actor* belum tentu merupakan orang
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau *actor*.

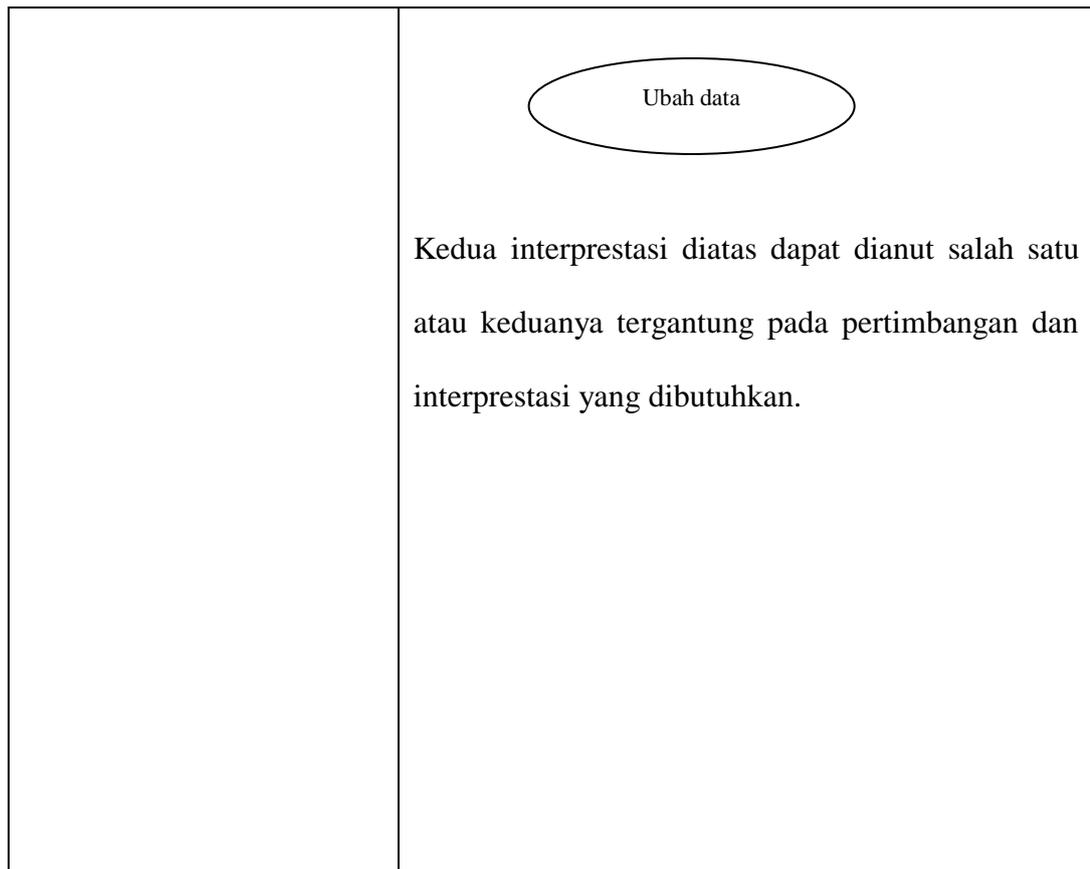
Tabel 2.2 Simbol – Simbol *Use Case* Diagram

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yanaktorg saling bertukar pesan antar unit atau <i>actor</i>; biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i></p>
<p><i>Actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang,</p>

 <p>Nama <i>actor</i></p>	<p>tapi <i>actor</i> belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama <i>actor</i>.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara <i>actor</i> dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan <i>actor</i>.</p>
<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri meski tanpa <i>use case</i> tambahan itu mirip dengan prinsip.</p> <p><i>Inheritance</i> pada pemogram berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, missalnya:</p> <div style="text-align: center;">  <pre> graph BT A([Validasi sidik jari]) --> B([Validasi user]) B --> C([Validasi username]) </pre> </div>

	<p>Arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
<p><i>Generalisasi/generalization</i></p> <p>→</p>	<p>Hubungan <i>generalisasi</i> dan <i>spesialisasi</i> (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misalnya;</p>  <pre> graph TD A([Ubah data]) --> B([Mengelola data]) C([Hapus data]) --> B </pre> <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya.</p>

<p>Menggunakan <i>/include/</i> <i>uses</i></p> <p><code><<include>></code> →</p> <p><code><<uses>></code> →</p>	<p>Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> :</p> <p>Include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut :</p>  <pre>graph BT; login([login]) --> validasi_username([Validasi username]);</pre> <p>Include berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut;</p>  <pre>graph BT; validasi_user([Validasi user]) --> validasi_user([Validasi user]);</pre>
--	--



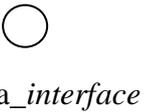
2.10.2 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun aplikasi. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa A.S dan M. Shalahuddin, 2013:141).

Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. *Class diagram* dibuat agar seorang *programmer* membuat kelas-kelas sesuai rancangan didalam diagram kelas sehingga antara dokumentasi perancangan dan perangkat lunak *sinkron*. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat

membuat kelas-kelas didalam program perangkat lunak sesuai dengan perancangan *class diagram*.

Tabel 2.3 Simbol – Simbol *Class Diagram*

Simbol	Deskripsi
<p>Kelas</p> 	<p>Kelas pada struktur <i>system</i></p>
<p>Antarmuka / <i>interface</i></p> 	<p>Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek</p>
<p><i>Asosiasi / association</i></p> 	<p>Relasi antar kelas dengan makna umum, <i>asosiasi</i> biasanya juga disertai dengan <i>multiplicity</i></p>
<p><i>Asosiasi berarah / directed association</i></p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, <i>asosiasi</i> biasanya juga disertai dengan <i>multiplicity</i></p>
<p><i>Generalisasi</i></p> 	<p>Relasi antar kelas dengan makna <i>generalisasi-spesialisasi</i> (umum-khusus)</p>

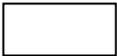
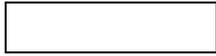
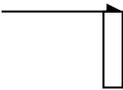
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

2.10.3 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinisiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case* (Rosa A.S dan M. Shalahuddin, 2013:165).

Tabel 2.4 Simbol – Simbol *Sequence Diagram*

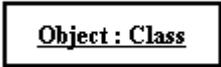
Simbol	Deskripsi
<i>Actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem

<p>atau</p> 	
<p><i>Lifeline</i></p> <p>.....</p>	Menyatakan kehidupan suatu objek
<p>Objek</p> 	Merupakan objek yang berinteraksi pesan
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
<p>Pesan tipe <i>create</i></p> 	Menyatakan suatu objek memuat objek yang lain, arah panah mengarah pada objek yang dibuat
<p>Pesan tipe <i>call</i></p> 	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain
<p>Pesan tipe <i>send</i></p> 	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya
<p>Pesan tipe <i>return</i></p> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembali ke objek tertentu
<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek yang lain

2.10.4 Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence* diagram, tapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, dimana *message* dari level tertinggi memiliki nomor satu. *Messages* dari level yang sama memiliki prefiks yang sama (Sri Darwiyanti dan Romi Satria Wahono, 2010).

Tabel 2.5 Simbol – Simbol *Collaboration* Diagram

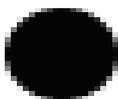
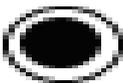
Simbol	Deskripsi
Objek 	Objek yang melakukan interaksi pesan
<i>Link</i> 	Relasi antara objek yang menghubungkan objek satu dengan lainnya atau dengan diri sendiri
<i>Stimulus</i> 	Arah pesan yang terjadi, jika pada suatu link ada dua arah pesan yang berbeda maka arah juga digambarkan dua arah pada dua sisi link

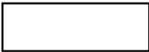
2.10.5 State Machine Diagram

State machine diagram merupakan pengembangan dari diagram *Finite State Automata* dengan penambahan beberapa fitur dan konsep baru. Diagram *Finite State Automata* biasanya digunakan dalam ilmu Automata (Rosa A.S dan M. Shalahuddin, 2013:163).

State machine diagram digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek. Jika diagram *sequence* digunakan untuk interaksi didalam sebuah objek maka diagram status digunakan untuk interaksi dalam sebuah objek. Perubahan tersebut digambarkan dalam suatu *graf* berarah.

Tabel 2.6 Simbol – Simbol *State Machine* Diagram

Simbol	Deskripsi
<p><i>Start / status awal (initial state)</i></p> 	<p><i>Start</i> atau <i>initial state</i> adalah <i>state</i> atau keadaan awal pada saat sistem mulai hidup</p>
<p><i>End / status akhir (final state)</i></p> 	<p><i>End</i> atau <i>final state</i> adalah <i>state</i> keadaan akhir dari daur hidup suatu sistem</p>

<p><i>Event</i></p> 	<p><i>Event</i> adalah kegiatan yang menyebabkan berubahnya status mesin</p>
<p><i>State</i></p> 	<p><i>State</i> atau status adalah keadaan sistem pada waktu tertentu</p>

2.10.6 Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem, bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa A.S dan M. Shalahuddin, 2013:161).

Tabel 2.7 Simbol – Simbol *Activity* Diagram

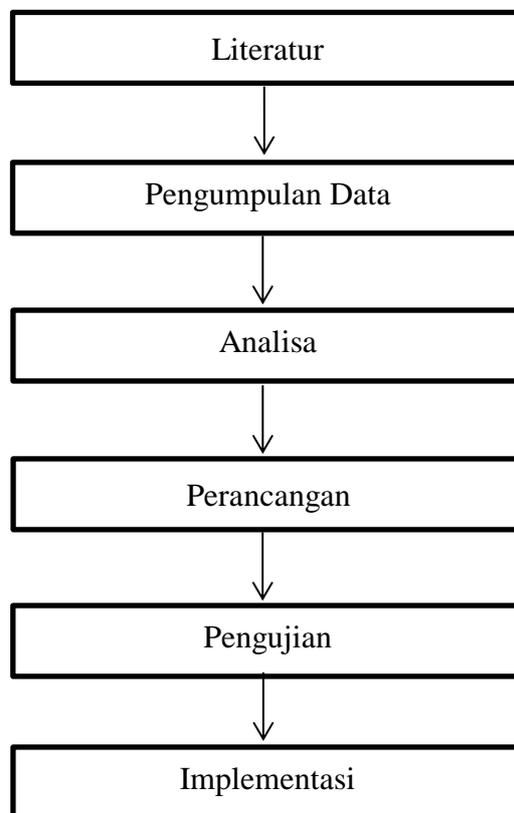
Simbol	Deskripsi
<p>Status awal</p> 	<p>Status awal aktivitas sistem</p>
<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja</p>
<p><i>Decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu</p>
<p><i>Join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabung menjadi satu</p>

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Adapun kerangka penelitian dalam metodologi penelitian ini dibuat agar langkah-langkah yang diambil penulis dalam perancangan ini tidak melenceng dari pokok pembahasan dan lebih mudah dipahami, maka urutan langkah-langkah akan dibuat secara sistematis sehingga dapat dijadikan pedoman yang jelas dan mudah untuk menyelesaikan permasalahan yang ada. Urutan langkah-langkah yang akan dibuat pada penelitian ini dapat dilihat pada Gambar 3.1 berikut.



Gambar 3.1 Kerangka Kerja Penelitian

3.2 Tahapan Penelitian

Tahap penelitian ini menjelaskan langkah-langkah dalam mengumpulkan beberapa beberapa laporan yang diperlukan untuk dapat dijadikan pedoman dalam pembuatan penelitian ini, yaitu :

3.2.1 Literatur

Literatur merupakan langkah pertama dalam melakukan suatu penelitian. Penelitian dilaksanakan dengan mengumpulkan beberapa modul atau pedoman untuk penghitungan menentukan nilai dari masing-masing kriteria dengan menggunakan metode *Analytic Hierarchy Process* (AHP) yang bertujuan untuk membantu proses pengambilan keputusan secara akurat dengan membandingkan berbagai nilai kriteria yang ada dalam mengambil keputusan. Sehingga membantu mengambil keputusan secara cepat dan tepat.

Setelah dikumpulkannya teori-teori yang bersangkutan maka ditentukan langkah-langkah dari metode penelitian ini. Penelitian ini bertujuan untuk mengumpulkan, menganalisa, pengujian dan implementasi dari penelitian ini, sehingga penelitian ini dapat dirancang dengan baik.

3.2.2 Pengumpulan Data

Rancangan urutan-urutan dalam melakukan penelitian dari awal sampai akhir penelitian. Tahapan-tahapan dalam penelitian ini dapat dijelaskan sebagai berikut : Penelitian Kepustakaan (*Library Research*). Pada tahap ini dikumpulkan berbagai dokumentasi, hasil-hasil penelitian dan teori-teori yang diarahkan untuk mendapatkan konsep-konsep penelitian yang berkaitan dengan permasalahan yang ada sebagai landasan dalam tahap-tahap penelitian selanjutnya. Teori-teori dan hal penelitian yang dimiliki dan memiliki hubungan erat dengan masalah yang akan diteliti dijadikan

landasan teori sebagai kerangka berpikir untuk menyelesaikan permasalahan yang ada mulai dari tahap awal sampai akhir penulisan laporan ini.

adalah perluasan aktivitas yang saling menyesuaikan.

3.3 Analisa Sistem

Analisis sistem dilakukan bertujuan untuk mengetahui masalah yang terjadi pada sistem pengambilan keputusan, sebagai dasar melakukan pengembangan sistem dari data-data yang telah diperoleh dari teori-teori yang telah dipelajari. Analisis sistem dilakukan agar pengembangan dan perancangan sistem yang baru dapat dilakukan dengan baik dan sesuai dengan kebutuhan *user* tersebut.

Dalam melakukan analisis terhadap sistem yang sedang berjalan, dalam sistem pengambilan keputusan ini, penulis menganalisa apakah aplikasi berjalan secara efektif pada saat ini. Banyaknya permasalahan dalam pengambilan keputusan yang dilakukan secara manual menurut beberapa pertimbangan yang telah ditentukan, serta dengan mengenal aplikasi sistem pengambilan keputusan ini diharapkan dapat menghasilkan calon-calon anggota baru yang berjiwa pemimpin, cerdas, serta dapat bertanggung jawab.

3.3.1 Kebutuhan Sistem

Dalam proses perancangan aplikasi sistem pengambilan keputusan ini di butuhkan perangkat-perangkat dasar maupun perangkat-perangkat pendukung dalam bentuk software dan hardware. Adapun perangkat-perangkat tersebut adalah :

1. Perangkat Keras (*Hardware*)

Satu unit Personal Komputer, dengan spesifikasi minimum sebagai berikut :

- a. Processor : Intel (R) Core (TM) i3-2310M CPU 2.10GHz
- b. Memory : 2048MB RAM

c. Monitor : Dell 21 inc

2. Perangkat Lunak (*Software*)

a. Sistem operasi windows 7.

b. Bahasa pemrograman PHP dengan menggunakan editor *notepad++* sebagai media pembuatan aplikasi.

c. XAMPP adalah *software* untuk lokal web server.

a. MySQL adalah perangkat lunak untuk menyimpan data.

b. *Adobe photoshop* digunakan untuk membantu mendisain tampilan sistem pengambilan keputusan.

c. *Astah Community*

3.3.2. Analisa Sistem yang Sedang Berjalan

Dalam perancangan aplikasi sistem pengambilan keputusan untuk seleksi penerimaan anggota koperasi RSU , terlebih dahulu dikaji terhadap apa yang dilakukan sebelumnya. Berikut ini merupakan prosedur yang terjadi dalam proses seleksi :

1. Para calon anggota mendaftarkan diri kepada panitia penerimaan.
2. Panitia penerimaan memberikan bukti pendaftaran berupa nomor pendaftaran untuk memverifikasi pendaftaran secara *online*.
3. Para calon anggota mengisi semua data yang ada pada sistem.
4. Setelah semua data di isi, para anggota dapat keluar dari sistem.
5. Para calon anggota baru wajib mengikuti semua seleksi yang telah diterapkan oleh panitia penerimaan dari awal perekrutan sampai akhir perekrutan.
6. Beberapa penilaian yang telah dilakukan melalui seleksi, akan dilakukan beberapa pertimbangan melalui musyawarah oleh seluruh panitia penerimaan.

3.3.3. Analisis Sistem yang Akan Dibangun

Merujuk pada hasil analisis proses yang sedang berjalan, maka proses aplikasi yang akan dibangun adalah sebagai berikut:

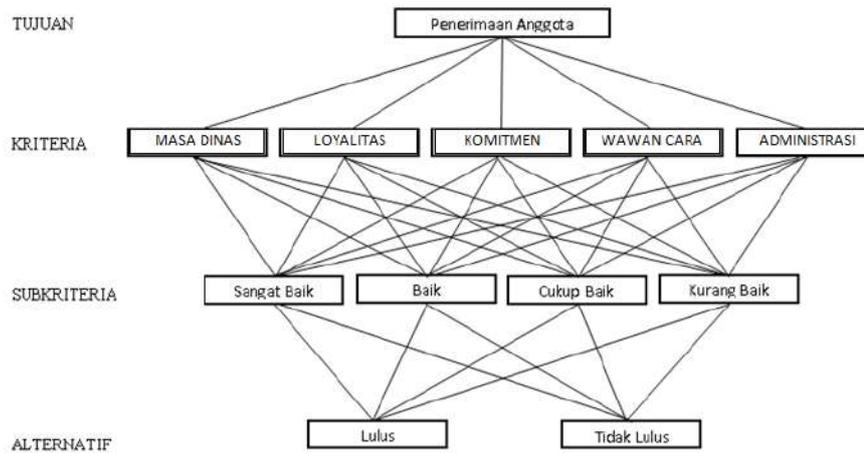
1. Para calon anggota baru mendaftarkan diri kedalam sistem agar bisa dilakukan proses pengambilan keputusan.
2. Panitia penerimaan memasukan data penilaian seleksi yang telah dilakukan berdasarkan kriteria pendukung kedalam sistem aplikasi.
3. Data dari masukan tadi akan di analisa oleh sistem.
4. Setelah di analisa maka keluarlah keputusan.
5. Para calon anggota baru bisa melihat hasil keputusan yang telah dilakukan apakah berhak diterima atau tidak.

3.4 Penerapan AHP

3.4.1. Hierarki

Pada dasarnya, AHP merupakan metode yang digunakan untuk memecahkan masalah yang kompleks dan tidak terstruktur ke dalam kelompok-kelompoknya, dengan mengatur kelompok tersebut ke dalam struktur *hirarki*. Struktur *hirarki* adalah suatu model yang memberikan kesempatan bagi perorangan atau kelompok untuk membangun gagasan-gagasan dan mendefinisikan persoalan dengan cara membuat asumsi mereka masing-masing dan memperoleh pemecahan yang diinginkan darinya. Dengan cara menyusun tujuan utama sebagai *level* teratas akan disusun *level hirarki* yang berada di bawahnya yaitu kriteria-kriteria yang cocok untuk mempertimbangkan atau menilai alternatif yang kita berikan dan menentukan

alternatif tersebut. Tiap-tiap kriteria mempunyai intensitas yang berbeda-beda seperti pada gambar berikut ini:



Gambar 3.2 Struktur *Hirarki* SPK Penerimaan

3.4.2. Alternatif

Alternatif merupakan keputusan akhir, dimana alternatif dengan nilai tertinggi akan dipilih. Alternatif yang akan digunakan dalam AHP penerimaan anggota baru koperasi yaitu :

1. Terima

Calon anggota yang penilaian diatas standar, maka bisa dinyatakan lulus dalam seleksi. Hasil keputusan yang telah dilakukan akan tersimpan didalam database.

2. Tidak Terima

Calon anggota yang tidak memenuhi kriteria dari penilaian bisa dikatakan tidak lulus dalam penerimaan

3.4.3. Kriteria

Kriteria merupakan dasar penilaian, dimana masing-masing kriteria diberi prioritas-prioritas yang nantinya akan mempengaruhi pengambilan keputusan. Dimana data yang menjadi kriteria pengambilan keputusan :

1. Masa Dinas

Berisi dari data-data yang didapatkan dari hasil tes yang telah dilakukan sebelumnya. Kriteria tes pengetahuan memiliki *subkriteria* antara lain :

- a. Sangat baik : *nilai* $\geq 85\%$
- b. Baik : *nilai* $\geq 70\%$ - $\leq 84\%$
- c. Cukup Baik : *nilai* $\geq 55\%$ - $\leq 69\%$
- d. Kurang Baik : *nilai* $\leq 54\%$

2. Loyalitas

Berisi dari data-data yang didapatkan dari hasil tes yang telah dilakukan sebelumnya. Kriteria tes performance memiliki *subkriteria* antara lain :

- a. Sangat baik : *nilai* $\geq 85\%$
- b. Baik : *nilai* $\geq 70\%$ - $\leq 84\%$
- c. Cukup Baik : *nilai* $\geq 55\%$ - $\leq 69\%$
- d. Kurang Baik : *nilai* $\leq 54\%$

3. Komitmen

Berisi dari data-data yang didapatkan dari hasil tes yang telah dilakukan sebelumnya. Kriteria tes kelompok memiliki *subkriteria* antara lain :

- a. Sangat baik : *nilai* $\geq 85\%$
- b. Baik : *nilai* $\geq 70\%$ - $\leq 84\%$
- c. Cukup Baik : *nilai* $\geq 55\%$ - $\leq 69\%$
- d. Kurang Baik : *nilai* $\leq 54\%$

4. Tes Wawancara

Berisi dari data-data yang didapatkan dari hasil tes yang telah dilakukan sebelumnya. Kriteria tes wawancara memiliki *subkriteria* antara lain :

- a. Sangat baik : *nilai* $\geq 85\%$
- b. Baik : *nilai* $\geq 70\%$ - $\leq 84\%$
- c. Cukup Baik : *nilai* $\geq 55\%$ - $\leq 69\%$
- d. Kurang Baik : *nilai* $\leq 54\%$

5. Tes Kehadiran

Berisi dari data-data yang didapatkan dari hasil tes yang telah dilakukan sebelumnya. Kriteria tes kehadiran memiliki *subkriteria* antara lain :

- a. Sangat baik : *nilai* $\geq 85\%$
- b. Baik : *nilai* $\geq 70\%$ - $\leq 84\%$
- c. Cukup Baik : *nilai* $\geq 55\%$ - $\leq 69\%$
- d. Kurang Baik : *nilai* $\leq 54\%$

3.4.4. Penilaian Kriteria AHP

Dengan naluri manusia dapat memperkirakan besaran sederhana melalui inderanya. Proses yang mudah adalah dengan membandingkan dua hal dengan keakuratan perbandingan yang dapat dipertanggung jawabkan. Untuk menilai perbandingan tingkat kepentingan elemen, Saaty(1980) menetapkan skala kuantitatif 1 sampai 9 seperti tabel dibawah ini :

Tabel 3.1 Skala Penilaian Perbandingan Berpasangan

Intensitas Kepentingan	Keterangan
---------------------------	------------

1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting dari pada elemen lainnya.
5	Elemen yang satu lebih penting dari pada elemen lainnya
7	Satu elemen jelas lebih mutlak penting dari pada elemen lainnya.
9	Satu elemen mutlak penting dari pada elemen lainnya.
2, 4, 6, 8	Nilai-nilai antara dua nilai pertimbangan yang berdekatan.

Skala nilai diatas digunakan untuk mengisi nilai matrik perbandingan berpasangan yang akan menghasilka prioritas (bobot/nilai kepentingan) masing-masing kriteria dan subkriteria

1. *Matriks* perbandingan berpasangan nilai kriteria

Pada tahap ini dilakukan penilaian perbandingan antara satu kriteria dengan kriteria yang lain. Hasil penilaian kriteria dapat dilihat dalam tabel berikut :

Tabel 3.2 *Matriks* Perbandingan Berpasangan Kriteria

Kriteria	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi
Masa Dinas	1	3	4	5	0,50
Loyalitas	0,33	1	3	4	0,33
Komitmen	0,25	0,33	1	3	0,25
Wawancara	0,20	0,25	0,33	1	0,20
Administrasi	2	3,00	4,00	5,00	1

Jumlah	3,78	7,58	12,33	18,00	2,28
--------	------	------	-------	-------	------

Angka 1 pada kolom Masa Dinas pada baris *pengetahuan* menggambarkan tingkat kepentingan yang sama antara Masa Dinas dengan Masa Dinas. Sedangkan angka 3 pada kolom Masa Dinas baris Masa Dinas menunjukkan bahwa Masa Dinas sedikit lebih penting dibandingkan Masa Dinas. Angka 0,33 kolom Masa Dinas baris *Loyalitas* merupakan hasil perhitungan angka 1 pada baris dan kolom Masa Dinas dibagi dengan 3 angka pada baris Masa Dinas kolom *Loyalitas*. Sedangkan angka yang lain diperoleh dengan cara yang sama.

2. Membuat *matriks* nilai kriteria

Matriks ini diperoleh dengan rumus sebagai berikut :

- a. Nilai baris kolom baru = nilai baris kolom lama (tabel matrik perbandingan berpasangan) / jumlah masing kolom lama (tabel matrik perbandingan berpasangan).
- b. Nilai kolom jumlah tabel matrik nilai kriteria diperoleh dari penjumlahan nilai pada tiap baris tabel tersebut. $(0,26 + 0,40 + 0,32 + 0,28 + 0,22 = 1,48)$.
- c. Nilai prioritas pada tabel matrik nilai kriteria diperoleh dari pembagian kolom jumlah dibagi jumlah kriteria.

Hasil perhitungan bisa dilihat pada tabel 3.3 berikut :

Tabel 3.3 Perhitungan prioritas nilai kriteria

Kriteria	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi	Jumlah	Prioritas
Masa Dinas	0,26	0,40	0,32	0,28	0,22	1,48	0,2962
Loyalitas	0,09	0,13	0,24	0,22	0,15	0,83	0,1663

Komitmen	0,07	0,04	0,08	0,17	0,11	0,47	0,0935
Wawancara	0,05	0,03	0,03	0,06	0,09	0,26	0,0512
Administrasi	0,53	0,40	0,32	0,28	0,44	1,96	0,3929

3. Membuat *matriks* penjumlahan setiap baris

Matrik ini dibuat dengan mengalikan nilai prioritas pada tabel matrik nilai kriteria dengan matrik perbandingan berpasangan. Hasil perhitungannya dapat dilihat pada tabel berikut :

Tabel 3.4 Perhitungan matrik kriteria penjumlahan setiap baris

Kriteria	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi	Jumlah
Masa Dinas	0,30	0,50	0,37	0,26	0,20	1,62
Loyalitas	0,10	0,17	0,28	0,20	0,13	0,88
Komitmen	0,07	0,06	0,09	0,15	0,10	0,47
Wawancara	0,06	0,04	0,03	0,05	0,08	0,26
Administrasi	0,59	0,50	0,37	0,26	0,39	2,11

Nilai 0,30 pada baris dan kolom Masa Dinas didapat dari perkalian antara nilai prioritas baris Masa Dinas pada tabel matrik nilai kriteria dikalikan dengan nilai baris – kolom Masa Dinas pada tabel matrik perbandingan berpasangan. Nilai yang lain diperoleh dengan cara yang sama. Nilai jumlah didapat dari penjumlahan nilai pada masing-masing baris kriteria.

4. Perhitungan rasio konsistensi

Perhitungan ini digunakan untuk memastikan bahwa nilai rasio konsistensi (CR) $\leq 0,1$ jika nilainya lebih besar dari 0,1 maka matrik perbandingan berpasangan perlu diperbaiki. Untuk menghitung rasio konsistensi dibuat tabel 4.4 seperti berikut :

Tabel 3.5 Perhitungan CR kriteria

Kriteria	Jumlah per Baris	Prioritas	Hasil
Masa Dinas	1,62	0,30	1,92
Loyalitas	0,88	0,17	1,05
Komitmen	0,47	0,09	0,57
Wawancara	0,26	0,05	0,31
Administrasi	2,11	0,39	2,51

Kolom jumlah per baris diperoleh dari kolom jumlah pada tabel penjumlahan setiap baris, sedangkan kolom prioritas diperoleh dari kolom prioritas tabel matrik nilai kriteria. dari tabel diatas diperoleh nilai berikut :

Jumlah (penjumlahan dari nilai hasil) = 6,35,

n (jumlah kriteria) = 5

λ maks (jumlah/n) = 1,2706

CI (λ maks - n) / (n-1) = -0,932

CR (CI/IR (lihat pada tabel index random) = -0.832

Tabel index random (IR) dapat dilihat sebagai berikut :

Tabel 3.6 Index Random (IR)

1	2	3	4	5	6	7	8	9	10
0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

Oleh karena nilai $CR \leq 0,1$, maka rasio konsistensi perhitungan dapat diterima.

3.4.5. Penilaian *Sub Kriteria AHP*

1. *Matriks* perbandingan berpasangan sub kriteria tes *Masa Dinas* Hasil penilaian

kriteria dapat dilihat dalam tabel berikut :

Tabel 3.7 Matrik perbandingan berpasangan sub kriteria

Masa Dinas	Sangat Baik	Baik	Cukup Baik	Kurang Baik
Sangat Baik	1	3	4	6
Baik	0,33	1	3	5
Cukup Baik	0,25	0,33	1	3
Kurang Baik	0,17	0,20	0,33	1
Jumlah	1,75	4,53	8,33	15,00

2. Membuat *matriks* nilai sub kriteria

Tabel 3.8 Matrik prioritas sub kriteria

Masa Dinas	Sangat Baik	Baik	Cukup Baik	Kurang Baik	Jumlah	Prioritas	Prioritas Sub kriteria
Sangat Baik	0,57	0,66	0,48	0,40	2,11	0,5283	1,0000
Baik	0,19	0,22	0,36	0,33	1,10	0,2761	0,5226
Cukup Baik	0,14	0,07	0,12	0,20	0,54	0,1341	0,2538
Kurang Baik	0,10	0,04	0,04	0,07	0,25	0,0615	0,1164

Nilai pada kolom sub kriteria diperoleh dari pembagian antara nilai pada kolom prioritas dibagi dengan nilai tertinggi pada prioritas tersebut.

3. Membuat matrik penjumlahan setiap baris

Tabel 3.9 Hasil penjumlahan setiap baris

Masa Dinas	Sangat Baik	Baik	Cukup Baik	Kurang Baik	Jumlah
Sangat Baik	0,53	0,83	0,54	0,37	2,26
Baik	0,18	0,28	0,40	0,31	1,16
Cukup Baik	0,13	0,09	0,13	0,18	0,54
Kurang Baik	0,09	0,06	0,04	0,06	0,25

4. Perhitungan Rasio Konsistensi

Tabel 3.10 Perhitungan CR sub kriteria

Masa Dinas	Jumlah Per Baris	Prioritas	Hasil
Sangat Baik	2,26	0,53	2,79
Baik	1,16	0,28	1,44
Cukup Baik	0,54	0,13	0,68
Kurang Baik	0,25	0,06	0,31

Jumlah (penjumlahan dari nilai hasil) = 5.22

n (jumlah kriteria) = 4

λ maks (jumlah/ n) = 1,3041

CI (λ maks - n) / ($n-1$) = -0,899

CR (CI/IR (lihat pada tabel index random) = -0.998

Oleh karena nilai CR $\leq 0,1$, maka rasio konsistensi perhitungan dapat diterima.

3.4.6. Hasil Perhitungan

Prioritas hasil perhitungan kemudian dituangkan ke dalam *matriks* hasil yang terlihat pada Tabel 3.11.

Tabel 3.11 Hasil perhitungan setiap prioritas kriteria dan subkriteria

Masa Dinas 0,2962	Loyalitas 0,1663	Komitmen 0,0935	Wawancara 0,0512	Administrasi 0,3929
Sangat Baik 1,0000				
Baik 0,5226	Baik 0,6032	Baik 0,5170	Baik 0,5708	Baik 0,5950
Cukup Baik 0,2538	Cukup Baik 0,3642	Cukup Baik 0,2189	Cukup Baik 0,2390	Cukup Baik 0,3458
Kurang Baik 0,1164	Kurang Baik 0,1564	Kurang Baik 0,1010	Kurang Baik 0,1116	Kurang Baik 0,2060

Seandainya diberikan dari data tiga orang calon anggota sebagai berikut :

Tabel 3.12 Contoh Data

	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi
A	Baik	Cukup Baik	Sangat Baik	Cukup Baik	Sangat Baik
B	Sangat Baik	Sangat Baik	Baik	Cukup Baik	Baik
C	Cukup Baik	Baik	Cukup Baik	Sangat Baik	Baik

Maka hasilnya adalah :

Tabel 3.13 Hasil Perhitungan Contoh Data

	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi	Total
A	0.1548	0.0606	0.0205	0.0512	0.3929	0.6799
B	0.2962	0.1663	0.0205	0.0292	0.2338	0.7459
C	0.0752	0.1003	0.0935	0.0292	0.1359	0.4341

Nilai 0.1548 pada pelamar A pada kolom pengetahuan didapatkan dari hasil perkalian nilai prioritas pengetahuan dengan nilai sub prioritas pada sub kriteria pengetahuan, begitupun dengan nilai lainnya dilakukan dengan cara yang sama. Total

didapatkan dari nilai penjumlahan pada tiap baris penilaian. Maka dari hasil total score tertinggi yang nantinya akan diprioritaskan untuk dapat diterima.

3.5. Struktur Database

Struktur *database* merupakan kumpulan dari data-data beserta tipenya yang merupakan komponen penting dalam membuat suatu program. Struktur *database* merupakan struktur data yang saling berhubungan satu sama lain sehingga sangat diperlukan dalam menjalankan program dan juga menyimpan data dalam suatu sistem *database* seperti di bawah ini :

1. Nama tabel : *admin*
 Primary key : *username*
 Fungsi : Menyimpan data *admin*

Tabel 3.14 Tabel *Admin*

<i>Field</i>	<i>Key</i>	<i>Type</i>	<i>Length</i>	<i>Keterangan</i>
<i>Username</i>	Pk	<i>varchar</i>	20	<i>Username</i>
<i>Nama_lengkap</i>		<i>varchar</i>	30	<i>Password</i>
<i>Password</i>		<i>varchar</i>	30	Nama lengkap

2. Nama tabel : *user*
 Primary Key : *id_user*
 Fungsi : Menyimpan data *user*

Tabel 3.15 Tabel *User*

<i>Field</i>	<i>Key</i>	<i>Type</i>	<i>Length</i>	<i>Keterangan</i>
<i>Id_user</i>	Pk	<i>Int</i>	5	<i>Id_user</i>
<i>username</i>		<i>varchar</i>	20	<i>Username</i>
<i>password</i>		<i>varchar</i>	30	<i>Password</i>
<i>nama_lengkap</i>		<i>varchar</i>	50	Nama lengkap
<i>nip</i>		<i>varchar</i>	15	Nip

bagian		<i>varchar</i>	20	bagian
golongan		<i>varchar</i>	20	golongan
no_hp		<i>varchar</i>	15	No handphone
<i>Email</i>		<i>varchar</i>	30	<i>email</i>
Alamat		<i>text</i>		alamat
Waktu_daftar		<i>timestamp</i>		Waktu daftar

3. Nama tabel : kriteria

Primary Key : id_kriteria

Fungsi : Menyimpan data kriteria

Tabel 3.16 Tabel Kriteria

<i>Field</i>	<i>Key</i>	<i>Type</i>	<i>Length</i>	Keterangan
<i>Id_kriteria</i>	Pk	<i>Varchar</i>	5	<i>Id kriteria</i>
nama_kriteria		<i>varchar</i>	25	Nama Kriteria
Masa Dinas		<i>double</i>		Nilai Masa Dinas
Loyalitas		<i>double</i>		Nilai Loyalitas
Komitmen		<i>double</i>		Nilai Komitmen
Wawancara		<i>double</i>		Nilai Wawancara
Administrasi		<i>double</i>		Nilai Administrasi
nilai_kriteria		<i>double</i>		Nilai Kriteria

Nama tabel : sub_kriteria

Primary Key : id

Fungsi : Menyimpan data sub kriteria

Tabel 3.17 Tabel Sub Kriteria

<i>Field</i>	<i>Key</i>	<i>Type</i>	<i>Length</i>	Keterangan
<i>Id</i>	Pk	<i>Int</i>	5	<i>Id kriteria</i>
<i>Id_kriteria</i>		<i>varchar</i>	10	Nama Kriteria
<i>Sb</i>		<i>Double</i>		Nilai Sangat Baik
<i>B</i>		<i>Double</i>		Nilai Baik
<i>Cb</i>		<i>Double</i>		Nilai Cukup Baik
<i>Kb</i>		<i>Double</i>		Nilai Kurang Baik

3.6. Perancangan Aplikasi

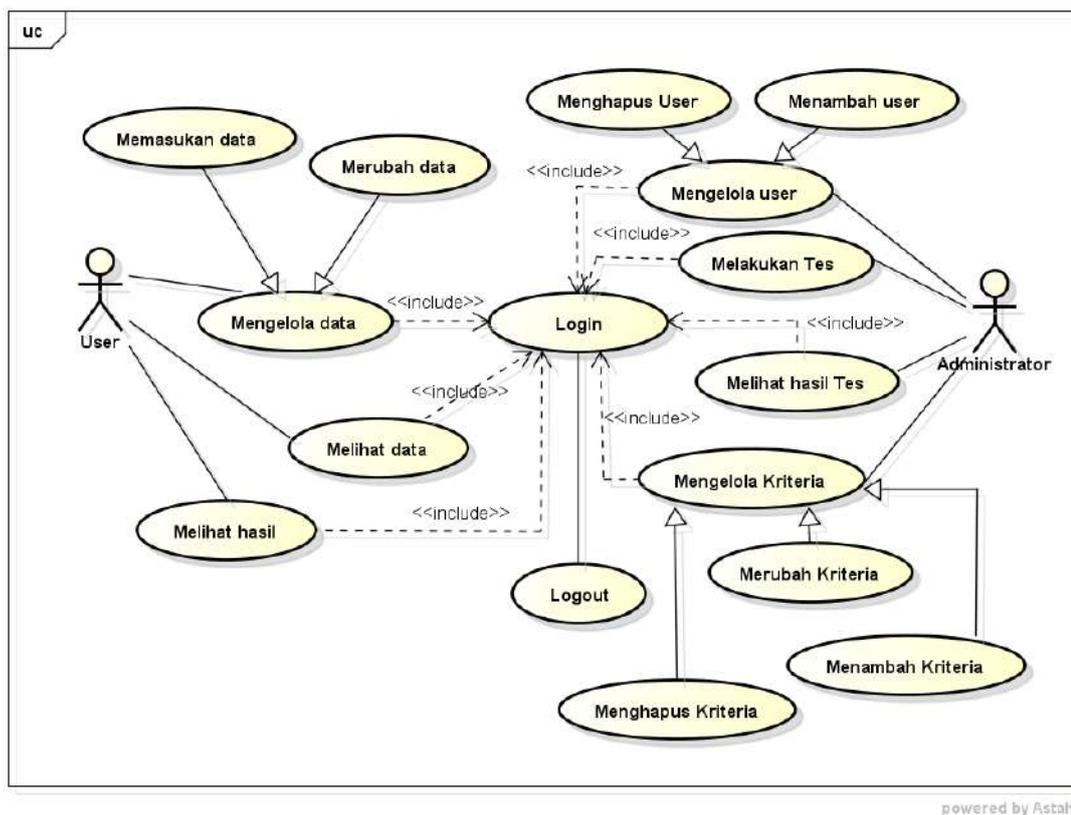
Perancangan aplikasi sistem pengamilan keputusan ini, dirancang menggunakan alat bantu berupa UML (*Unified Modelling Language*) agar mempermudah memindahkan konsep sistem yang dirancang kedalam bentuk program, dimana perancangannya digambarkan dalam bentuk diagram-diagram berikut :

1. *Use Case diagram.*
2. *Activity diagram.*
3. *State Machine diagram.*
4. *Sequence diagram*
5. *Class Diagram*
6. *Deployment Diagram*

3.6.1. Perancangan Aplikasi Menggunakan *Use Case Diagram*

Use case diagram adalah gambaran *graphical* dari beberapa atau semua actor, *use case*, dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun digunakan untuk menjelaskan bagaimana langkah-langkah yang seharusnya dikerjakan oleh sistem. *Use case diagram* menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem

tersebut berinteraksi dengan dunia luar. Adapun *use case* pada aplikasi ini adalah sebagai berikut :



Gambar 3.3 Use Case Diagram

A. Deskripsi Aktor

Berikut adalah deskripsi aktor pada sistem perancangan aplikasi sistem pengambilan keputusan pada tabel 3.18 dibawah ini :

Tabel 3.18 Deskripsi Aktor

Aktor	Deskripsi
<i>User</i>	Fungsinya untuk melihat hasil keputusan yang telah diinputkan administrator yang diproses oleh sistem serta dapat merubah segala data diri yang berhubungan dengan dirinya.

<i>Administrator</i>	Aktor yang berfungsi untuk mengelola segala sesuatu yang ada pada sistem aplikasi, seperti menambah, merubah, serta menghapus data.
----------------------	---

a. Deskripsi *Use Case*

Berikut adalah deskripsi *use case* pada sistem perancangan aplikasi sistem pengambilan keputusan pada tabel 3.19 dibawah ini :

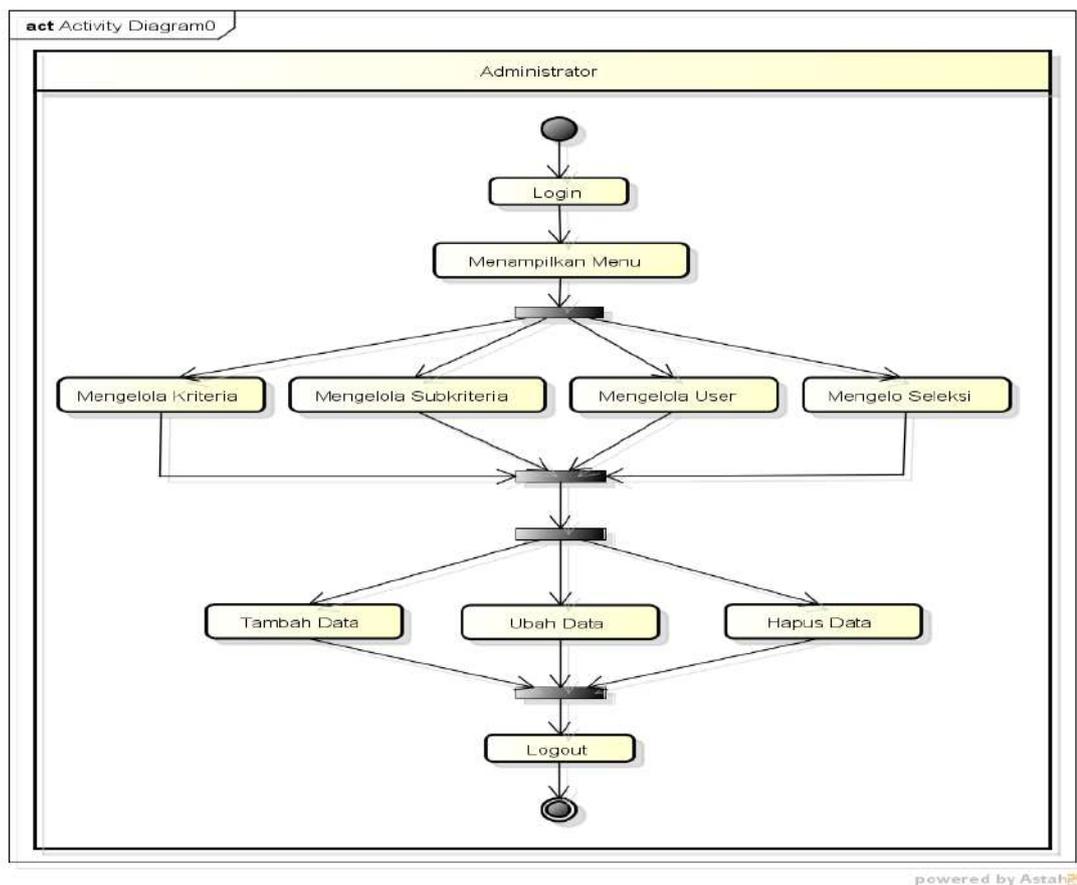
Tabel 3.19 Deskripsi *Use Case*

<i>Use Case</i>	Deskripsi	Aktor
<i>Login</i>	Hak akses yang digunakan untuk masuk kedalam sistem.	<i>User,</i> <i>Administrator</i>
<i>Logout</i>	Digunakan setelah menggunakan sistem	<i>User,</i> <i>Administrator</i>
Mengelola data	Digunakan untuk memasukan data diri serta merubah data diri.	<i>User</i>
Memasukan data	Digunakan untuk memasukan data diri yang berkaitan dengan user.	<i>User</i>
Merubah data	Digunakan untuk merubah segala sesuatu yang berhubungan dengan data diri <i>user</i> .	<i>User</i>
Melihat data	<i>User</i> dapat melihat data diri yang telah dimasukan atau yang sudah di rubah.	<i>User</i>
Melihat hasil	<i>User</i> bisa melihat hasil keputusan yang telah dimasukan oleh administrator yang diproses sistem	<i>User</i>

Mengelola <i>user</i>	Kegiatan yang digunakan untuk menambahkan data <i>user</i> , serta menghapus data	Administrator
Menambah <i>user</i>	Digunakan untuk menambahkan <i>user</i>	Administrator
Menghapus <i>user</i>	Digunakan untuk menghapus data <i>user</i>	Administrartor
Mengelola Kriteria	Digunakan untuk menambah, merubah, serta menghapus data mengenai data kriteria.	Administrator
Menambah kriteria	Digunakan untuk menambah data kriteria	Administrator
Merubah kriteria	Digunakan untuk merubah data kriteria.	Administrator
Menghapus Kriteria	Digunakan untuk menghapus kriteria.	Administrator.
Melakukan Tes	Digunakan untuk melakukan proses pengambilan keputusan dengan memasukan kritria yang telah ada	Administrator
Melihat hasil tes	Digunakan untuk melihat hasil tes pengambilan keputusan yang telah diproses sistem	Administrator

3.6.2. Perancangan Aplikasi Menggunakan *Activity Diagram*

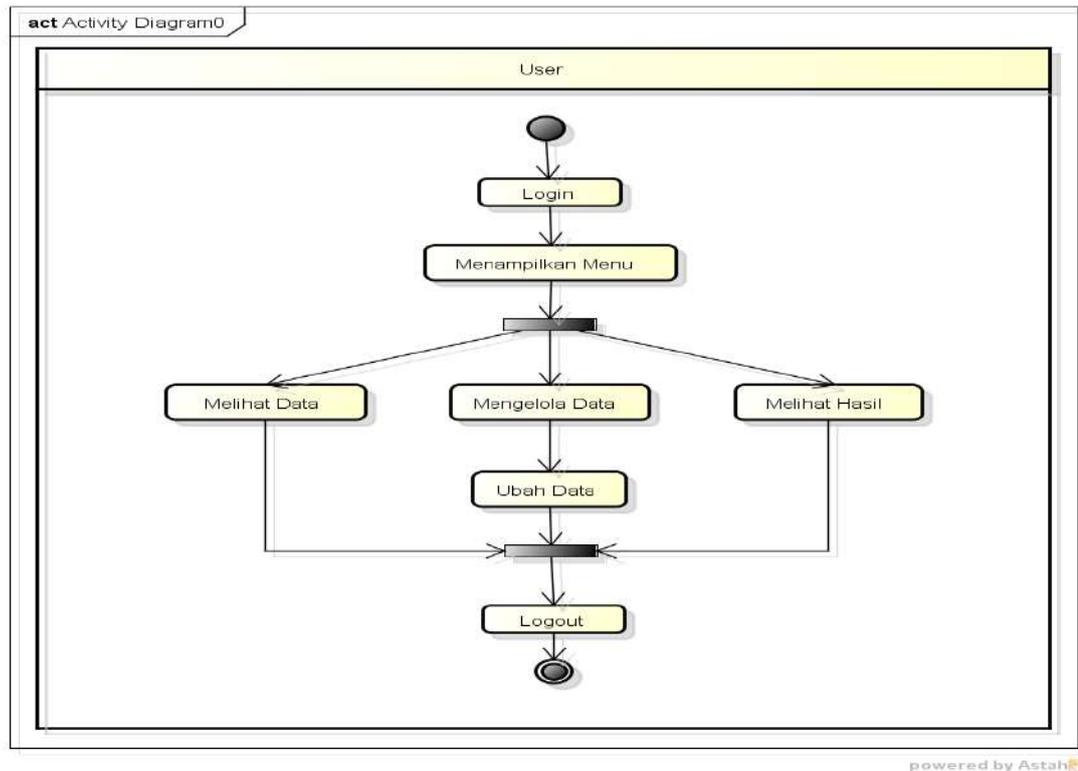
A. *Activity Diagram Admin*



Gambar 3.4 Activity Diagram Admin

Pada Gambar 3.4 di atas, *administrator* menggambarkan segala aktivitas yang bisa dilakukan *admin* terhadap aplikasi yang sedang berjalan serta bisa memilih aktivitas yang akan dilakukan melalui menu-menu pilihan yang ada. Sistem menggambarkan segala aktifitas yang digunakan untuk segala proses pengelolaan data.

B. Activity Diagram User



Gambar 3.5 Activity Diagram User

Pada Gambar 3.5 di atas, *user* menggambarkan segala aktivitas yang bisa dilakukan *user* terhadap aplikasi yang sedang berjalan serta bisa memilih aktivitas yang akan dilakukan melalui menu-menu pilihan yang ada. Sistem menggambarkan segala aktifitas yang digunakan untuk segala proses pengelolaan data.

Sequence Diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

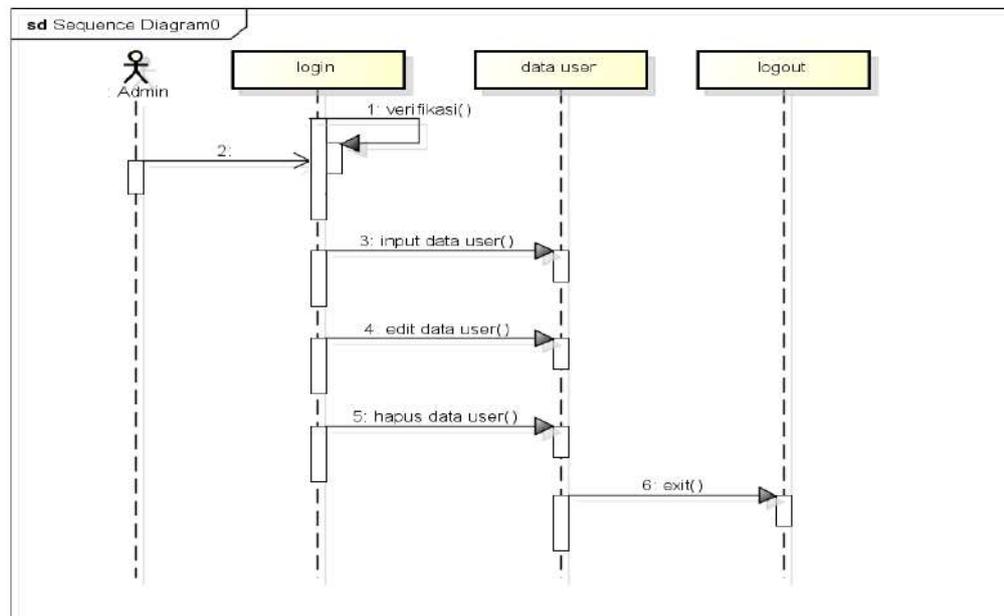
Pada *sequence diagram* ini menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan pada sistem sebagai respon dari sebuah *event* untuk menghasilkan output tertentu.

Gambar 3.6 *Sequence Diagram Admin Kelola User*

A. Sequence Diagram Admin Kelola User

Admin terlebih dahulu melakukan login untuk dapat masuk menggunakan form *input user*. *Admin* dapat mengelola data *user*, mulai dari menambahkan, merubah, serta menghapus data. Setelah selesai, *admin* dapat keluar dari sistem. Untuk lebih jelasnya bisa dilihat seperti gambar dibawah ini :

B. Sequence Diagram Admin Kelola Kriteria



Admin terlebih dahulu melakukan login untuk dapat masuk menggunakan form *input kriteria*. *Admin* dapat mengelola data *kriteria*, mulai dari menambahkan, merubah, serta menghapus data. Setelah selesai, *admin* dapat keluar dari sistem. Untuk lebih jelasnya bisa dilihat seperti gambar dibawah ini :

C. Sequence Diagram Admin Kelola Sub kriteria

Admin terlebih dahulu melakukan login untuk dapat masuk menggunakan form *input subkriteria*. *Admin* dapat mengelola data *subkriteria*, mulai dari menambahkan,

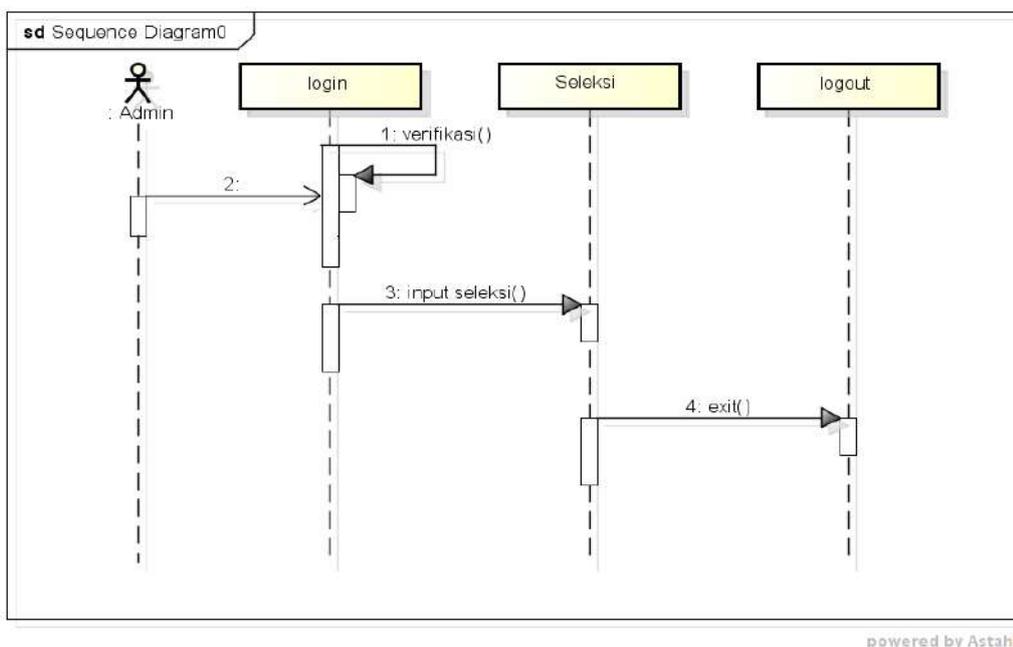
merubah, serta menghapus data. Setelah selesai, *admin* dapat keluar dari sistem. Untuk lebih jelasnya bisa dilihat seperti gambar dibawah ini :

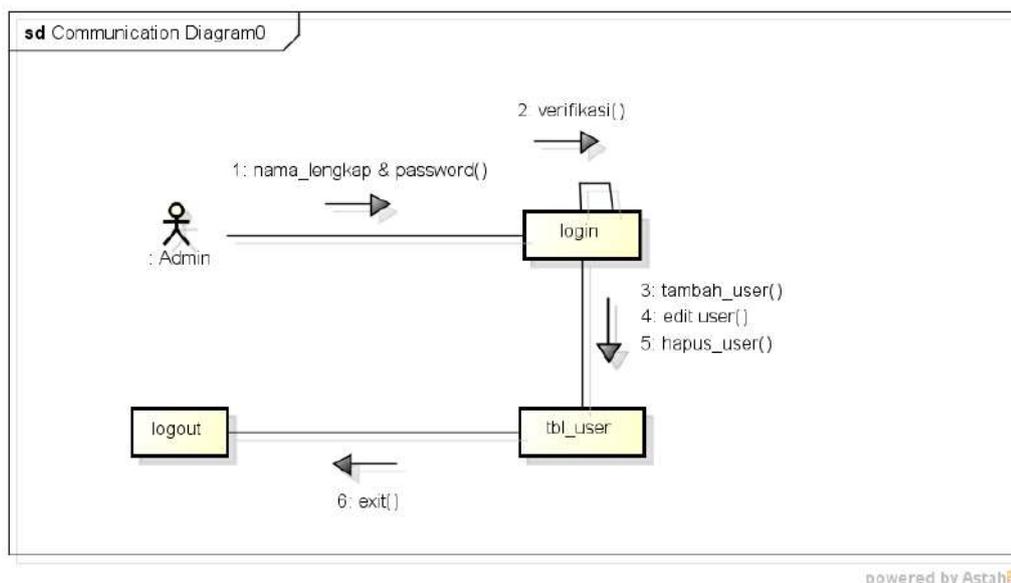
3.6.3 Perancangan Aplikasi Menggunakan *Collaboration Diagram*

Merupakan cara alternatif untuk menampilkan suatu *scenario*. Menampilkan interaksi objek yang terorganisasi di sekitar objek dan hubungannya dengan objek yang lain.

Gambar 3.7 *Sequence Diagram Admin Kelola Sub Kriteria*

A. *Collaboration Diagram User*

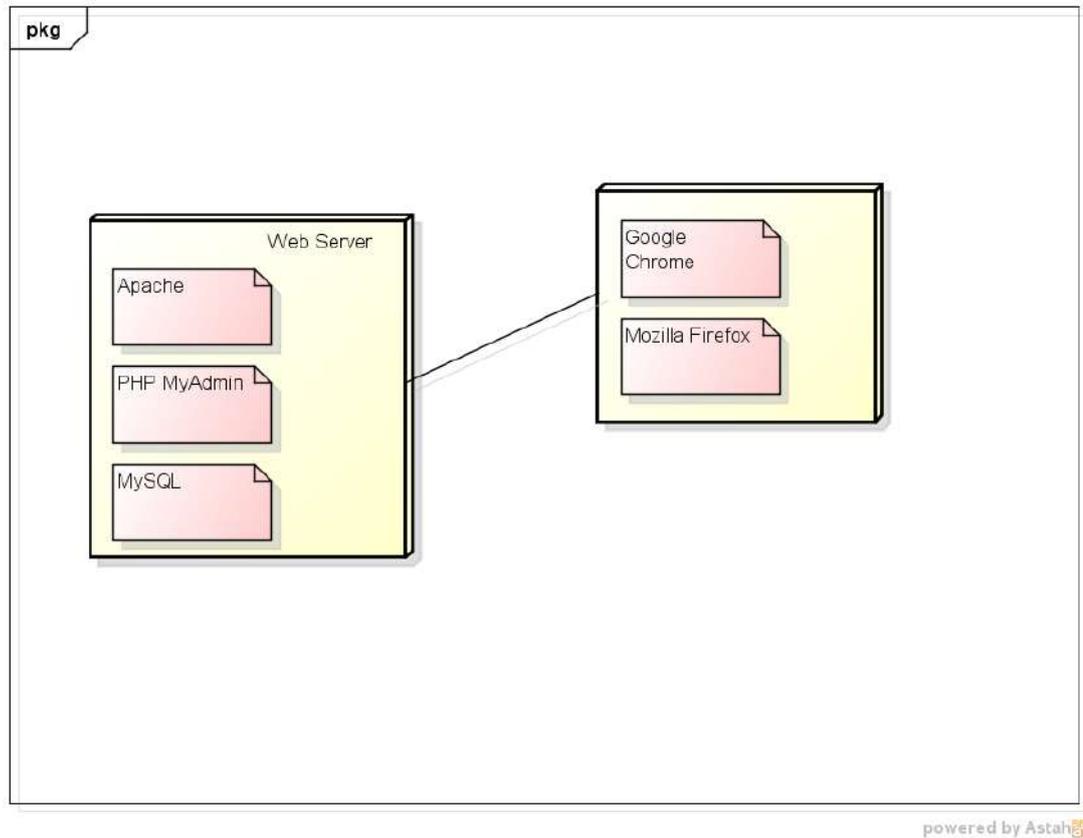




Gambar 3.8 Collaboration Diagram User

3.6.4 Perancangan Aplikasi Menggunakan *Deployment Diagram*

Deployment Diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi atau tata letak sebuah sistem secara fisik. Bagian utamanya adalah *node* yang merupakan nama umum untuk semua jenis sumber komputasi. Ada 2 tipe node yaitu *processor* atau *execution environment* (node yang bisa mengeksekusi sebuah komponen dan software yang menjadi host atau mengandung software lain) dan *device* (yang menjadi interface dengan dunia luar). Setiap node diwakili bentuk sebuah kubus, dan garis yang menghubungkan antar dua kubus menunjukkan hubungan diantara kedua node tersebut.



Gambar 3.9 *Deployment* Diagram

3.7. Perancangan Antar Muka

Perancangan antar muka adalah bentuk rancangan tampilan sementara dari pembuatan aplikasi sistem pengambilan keputusan Perancangan ini dibuat untuk memberikan penjelasan tentang tampilan yang dihadapkan pada aktor di saat menggunakan aplikasi sistem pengambilan keputusan ini. Sehingga dapat mempermudah dalam mengimplementasikan aplikasi serta akan memudahkan pembangunan aplikasi yang memenuhi prinsip perancangan anta muka yang baik. Perancangan aplikasi sistem pengambilan keputusan yang akan dibangun adalah sebagai berikut :

1. Perancangan Halaman *Index* Web

Perancangan tampilan *index* web adalah perancangan menu yang akan ditampilkan kepada *user* di saat pertama membuka aplikasi. Pada halaman website ini adalah halaman login yang akan membantu pengguna baik *user* maupun administrator untuk masuk kedalam sistem.

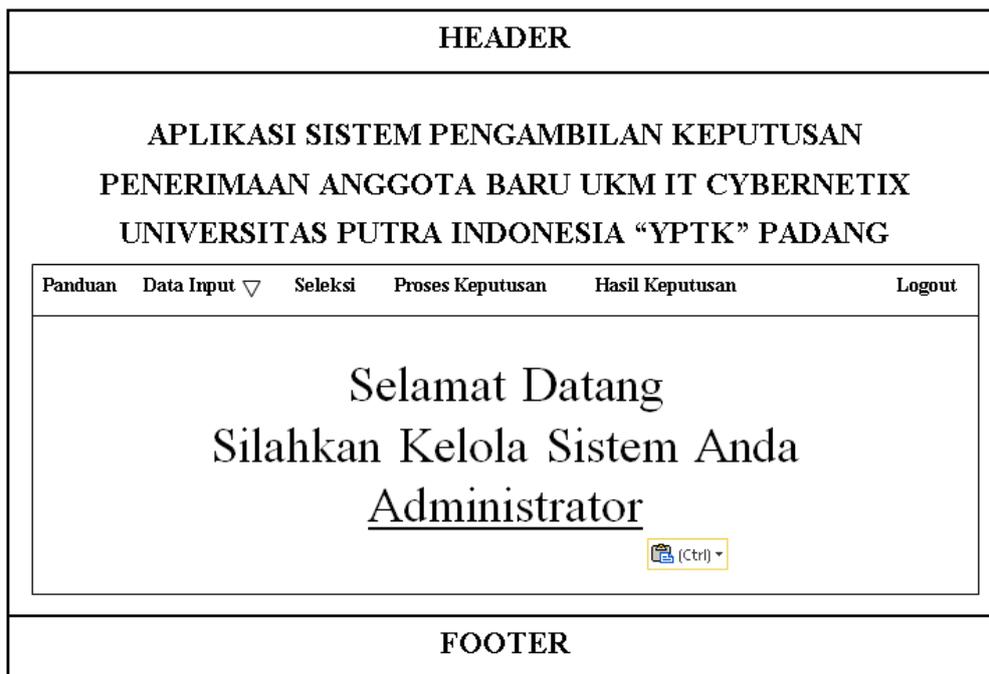
The diagram illustrates the layout of the initial website menu. It is structured as follows:

- HEADER:** A horizontal bar at the top containing the text "HEADER".
- Main Content Area:** A large rectangular area below the header, containing:
 - APLIKASI SISTEM PENGAMBILAN KEPUTUSAN:** The title of the application, centered at the top of the main content area.
 - Login Form:** A box on the left side with the title "Silahkan Login". It contains:
 - A text input field labeled "Username".
 - A text input field labeled "Password".
 - Two buttons: "Sign In" and "Cancel".
 - GAMBAR:** A large empty rectangular box on the right side, labeled "GAMBAR".
- FOOTER:** A horizontal bar at the bottom containing the text "FOOTER".

Gambar 3.10 Menu Awal Website

2. Perancangan Halaman *Index Admin*

Merupakan halaman yang akan ditampilkan kepada *admin* setelah login. Halaman *index admin* dapat digambarkan seperti gambar berikut .:



Gambar 3.11 Halaman *Index Admin*

3. Perancangan Halaman Panduan *Admin*

Merupakan halaman yang digunakan sebagai informasi yang berisi menampilkan aturan dalam aturan penentuan perbandingan berpasangan.

HEADER	
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN	
Panduan Data Input ▾ Seleksi Proses Keputusan Hasil Keputusan Logout	
<u>Tabel Aturan Perbandingan Berpasangan</u>	
Nilai	Keterangan
1	Sama Penting (Equal)
3	Sedikit Lebih Penting (Moderate)
5	Jelas Lebih Penting (Strong)
7	Sangat Jelas Lebih Penting (Very Strong)
9	Mutlak Lebih Penting (Extreme)
2,4,6,8	Apabila ragu-ragu antara 2 nilai yang berdekatan
FOOTER	

Gambar 3.12 Halaman Panduan

4. Perancangan *Form Registrasi User Admin*

Merupakan halaman yang digunakan untuk menambahkan para calon anggota baru yang akan diberi hak akses untuk melihat hasil keputusan yang telah dilakukan. Administrator hanya akan mengisi id *user* dan password untuk login *user*, dan untuk selebihnya, si calon anggota lah yang akan melengkapi data.

HEADER					
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN					
Panduan	Data Input ▾	Seleksi	Proses Keputusan	Hasil Keputusan	Logout
<u>Form Registrasi User</u>					
<input type="text" value="Username"/>					
<input type="text" value="Password"/>					
<input type="text" value="Nama Lengkap"/>					
<input type="text" value="No Bp"/>					
FOOTER					

Gambar 3.13 *Form Registrasi User*

5. Perancangan Halaman *Form Kriteria Admin*

Merupakan halaman yang digunakan untuk menambahkan, merubah, serta menghapus data kriteria yang dilakukan oleh *administrator*. Berikut rancangan dari sistem ini dapat dilihat seperti berikut :

HEADER					
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN					
Panduan	Data Input ▾	Seleksi	Proses Keputusan	Hasil Keputusan	Logout
Daftar Kriteria					
Kode	Keterangan	Keterangan			
FOOTER					

Gambar 3.14 *Form Kriteria*

6. Perancangan Halaman Seleksi *Admin*

Perancangan halaman seleksi ini merupakan halaman yang digunakan untuk melakukan proses seleksi yang dilakukan oleh *administrator* berdasarkan data-data yang telah diterima dari panitia penerimaan terlebih dahulu, dari keterangan data tersebut barulah bisa diberi penilaian. Penilaian yang tersedia dari halaman ini berupa sub kriteria penilaian, yaitu : sangat baik, baik, cukup baik, dan kurang baik.

HEADER														
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN														
Panduan	Data Input ▾	Seleksi	Proses Keputusan	Hasil Keputusan	Logout									
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Keterangan</th> </tr> </thead> <tbody> <tr> <td>Sangat Baik >= 85%</td> </tr> <tr> <td>Baik >=70% s/d <= 84%</td> </tr> <tr> <td>Cukup Baik >= 55% s/d <= 69%</td> </tr> <tr> <td>Kurang Baik <= 54%</td> </tr> </tbody> </table>	Keterangan	Sangat Baik >= 85%	Baik >=70% s/d <= 84%	Cukup Baik >= 55% s/d <= 69%	Kurang Baik <= 54%	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Silahkan Masukan Kriteria</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">-- Masukan Nama Calon Anggota --</td> </tr> <tr> <td style="height: 20px;"></td> </tr> <tr> <td style="height: 20px;"></td> </tr> <tr> <td style="text-align: center; color: blue; font-weight: bold;">PROSES</td> </tr> </tbody> </table>				Silahkan Masukan Kriteria	-- Masukan Nama Calon Anggota --			PROSES
Keterangan														
Sangat Baik >= 85%														
Baik >=70% s/d <= 84%														
Cukup Baik >= 55% s/d <= 69%														
Kurang Baik <= 54%														
Silahkan Masukan Kriteria														
-- Masukan Nama Calon Anggota --														
PROSES														
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Aturan</th> </tr> </thead> <tbody> <tr> <td>1. Xxxxxxxxxxxxxx</td> </tr> <tr> <td>2. Xxxxxxxxxxxxxx</td> </tr> <tr> <td>3. Xxxxxxxxxxxxxx</td> </tr> </tbody> </table>	Aturan	1. Xxxxxxxxxxxxxx	2. Xxxxxxxxxxxxxx	3. Xxxxxxxxxxxxxx										
Aturan														
1. Xxxxxxxxxxxxxx														
2. Xxxxxxxxxxxxxx														
3. Xxxxxxxxxxxxxx														
FOOTER														

Gambar 3.15 *Form Seleksi*

7. Perancangan Halaman *Admin* Proses Keputusan Kriteria

Perancangan halaman ini digunakan untuk menentukan perbandingan antara satu kriteria dengan kriteria lainnya. Hasil yang di masukan nantinya akan diproses sehingga mendapatkan nilai akhir seperti yang diinginkan. Rancangan halaman ini adalah sebagai berikut :

HEADER					
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN					
Panduan Data Input ▾ Seleksi Proses Keputusan Hasil Keputusan Logout					
<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> Matrik Perbandingan Matrik Kriteria Matrik Sub Kriteria </div> <div style="border: 1px solid black; padding: 2px;"> Keterangan 01 : Sangat Baik 02 : Baik 03 : Cukup Baik 04 : Kurang Baik </div>	MATRIK PERBANDINGAN SUB KRITERIA				
	SP01	SP02	SP03	SP04	Aksi
SP01	1				Edit
SP02		1			Edit
SP03			1		Edit
SP04				1	Edit
PROSES					
FOOTER					

Gambar 3.16 Halaman Perbandingan Kriteria

8. Perancangan Halaman *Admin* Hasil Keputusan

Perancangan halaman ini digunakan untuk melihat daftar calon anggota yang diterima maupun ditolak. Adapun rancangan dari halaman hasil keputusan ini sebagai berikut :

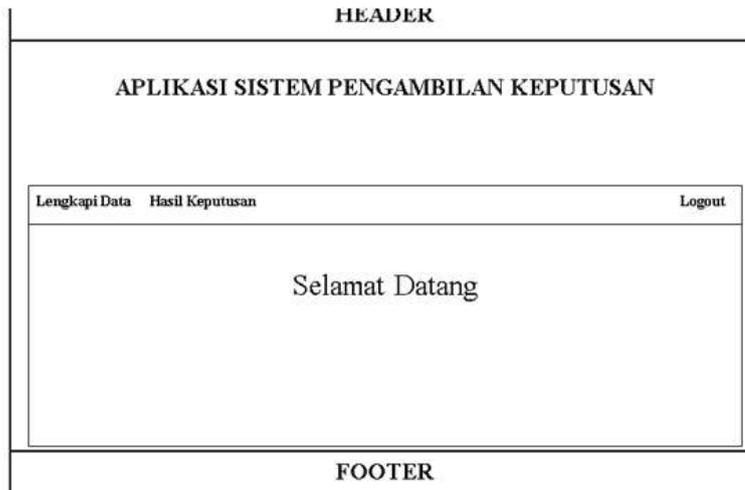
HEADER					
APLIKASI SISTEM PENGAMBILAN KEPUTUSAN					
Panduan Data Input ▾ Seleksi Proses Keputusan Hasil Keputusan Logout					
<div style="border: 1px solid black; height: 40px; margin-bottom: 5px;"></div> <div style="border: 1px solid black; height: 40px; margin-bottom: 5px;"></div> <div style="border: 1px solid black; height: 40px; margin-bottom: 5px;"></div>					

Gambar 3.17 Halaman Hasil Keputusan

9. Perancangan Halaman *Index User*

Merupakan halaman yang akan ditampilkan kepada *admin* setelah login.

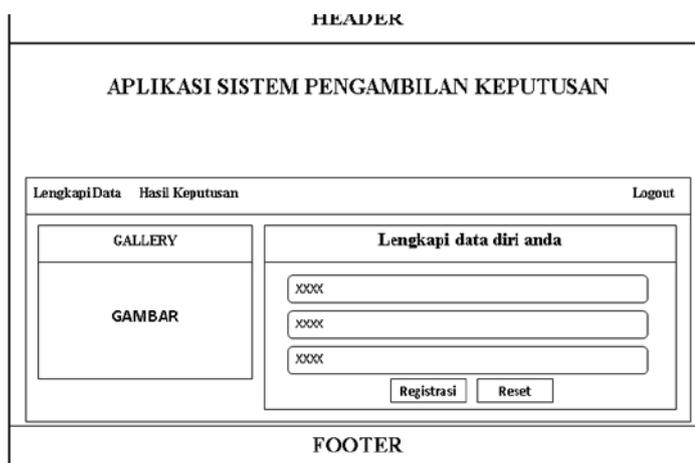
Halaman *index admin* dapat digambarkan seperti gambar berikut :



Gambar 3.18 Halaman *Index User*

10. Perancangan Halaman *Data User*

Merupakan halaman yang akan digunakan *user* untuk melengkapi data dirinya, yang nantinya akan digunakan untuk proses seleksi. Halaman data *user* dapat digambarkan sebagai berikut :



Gambar 3.19 Halaman *Data User*

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi Sistem

Implementasi sistem merupakan bagian dari siklus pengembangan sistem itu sendiri. Untuk melakukan sebuah implementasi maka diperlukan aplikasi perancangan *interface* dan penulisan *coding* sesuai dengan sistem yang dirancang ataupun yang telah dianalisa.

Perancangan sistem disini menggunakan model UML (*Unified Modelling Language*) yang digunakan untuk menggambarkan cara kerja sistem pengambilan keputusan. Setelah perancangan sistem selesai maka di lakukan implementasi kedalam program yakni dengan menggunakan bahasa pemrograman PHP dan *database* MySQL.

Implmentasi sistem dilakukan setelah tahapan perancangan sistem dan pengujian dilakukan. Implementasi sistem dapat dilakukan setelah sistem yang dibuat dapat berjalan sebagaimana mestinya. Perancangan *interface* dilakukan untuk interaksi *user* dengan sistem yang telah dibuat.

Adapun kebutuhan *hardware* ataupun *software* dalam pengujian sistem ini adalah sebagai berikut:

4.1.1. Kebutuhan *Hardware*

Dalam implementasi ini menggunakan bantuan dari perangkat keras (*hardware*), dimnana perangkat keras yang digunakan adalah sebagai berikut:

1. Processor : Intel (R) Core (TM) i3-2310M CPU 2.10GHz
2. Memory : 2048MB RAM
3. Monitor : Dell 21 inc

4.1.2. Kebutuhan *Software*

Dalam implementasi sistem ini juga menggunakan bantuan dari berbagai perangkat lunak, diantara perangkat lunak yang digunakan adalah sebagai berikut:

1. Sistem operasi windows 7.
2. Bahasa pemrograman PHP dengan menggunakan editor *notepad++* sebagai media pembuatan aplikasi.
3. XAMPP adalah *software* untuk lokal web server.
4. MySQL adalah perangkat lunak untuk menyimpan data.
5. *Adobe photoshop* digunakan untuk membantu mendisain tampilan sistem pengambilan keputusan.
6. *Astah Community*

4.2. Implementasi Program

Implementasi program merupakan realisasi dari hasil perancangan yang telah dibuat dengan UML, Perancangan sistem yang sudah dibuat dikonversikan ke dalam bentuk program aplikasi. Untuk mengimplementasikan program sistem pengambilan keputusan penerimaan anggota baru, dibutuhkanlah aplikasi pendukung yang harus diinstal terlebih dahulu dan dikonfigurasi dengan benar agar program yang dibuat dapat berjalan dan berfungsi sebagaimana mestinya. Berikut ini aplikasi pendukung.

4.2.1. Proses Instalasi XAMPP 1.7.4

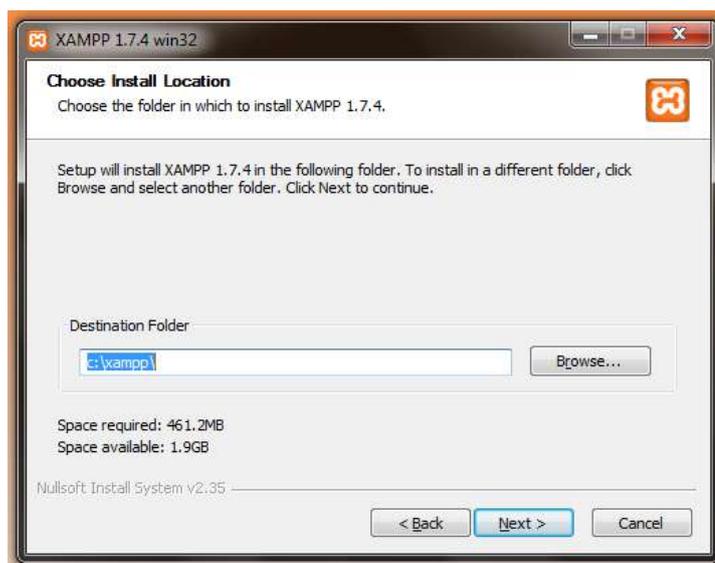
Langkah awal yang harus dilakukan adalah men-*download* file XAMPP . Kemudian lakukan instalasi. Langkah-langkah instalasinya adalah sebagai berikut :

1. Jalankan *file* *xampp-win32-1.7.4-4-VC6-installer.exe*, sebagai contoh penulis menggunakan xampp versi 1.8.2.
2. Ketika proses instalasi dimulai klik *next*.



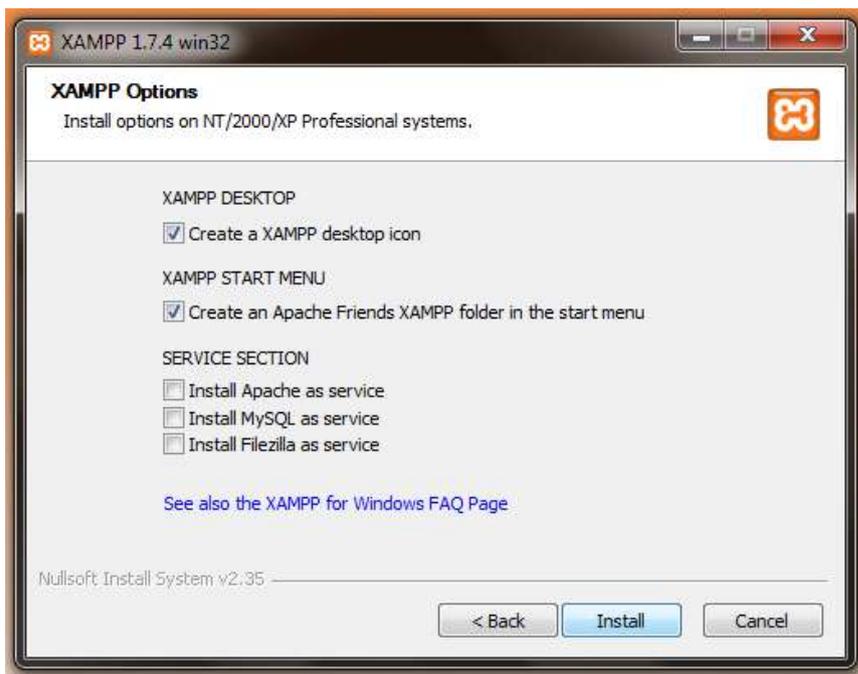
Gambar 4.1 Welcome XAMPP Setup

3. Selanjutnya pilih lokasi penginstalan XAMPP, kemudian klik next.



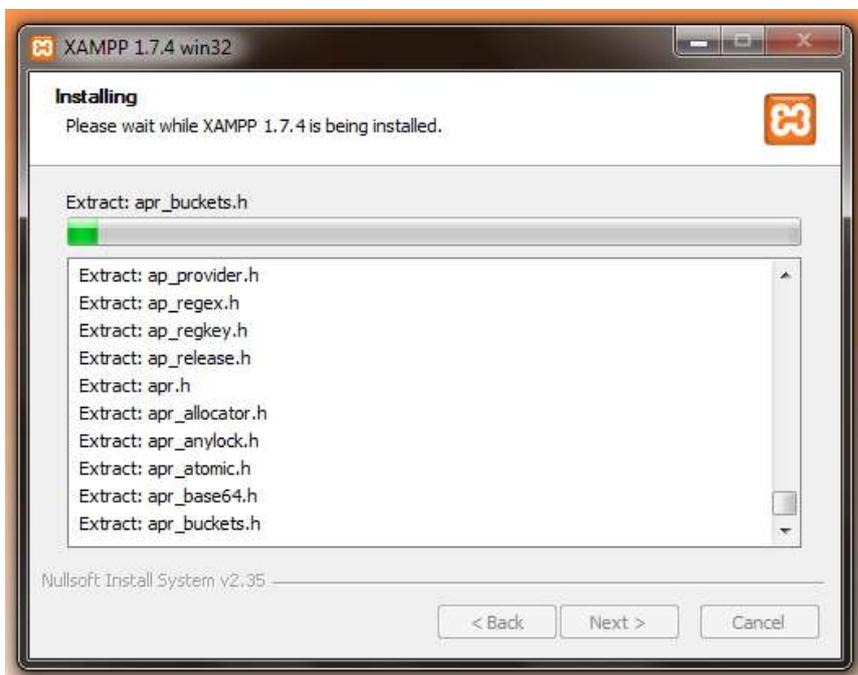
Gambar 4.2 Pilih Lokasi Install XAMPP

4. Anda bisa memilih opsi sesuai dengan kebutuhan. Klik install untuk memulai proses instalasi.



Gambar 4.3 Opsi Penginstalan XAMPP

5. Tunggu beberapa saat sampai proses instalasi XAMPP selsesai.



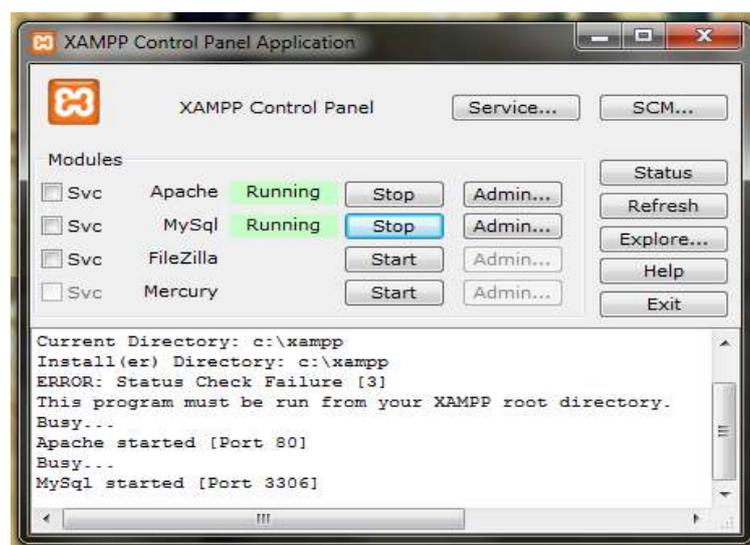
Gambar 4.4 Proses Instalasi XAMPP

6. Instalasi selesai. Klik finish.



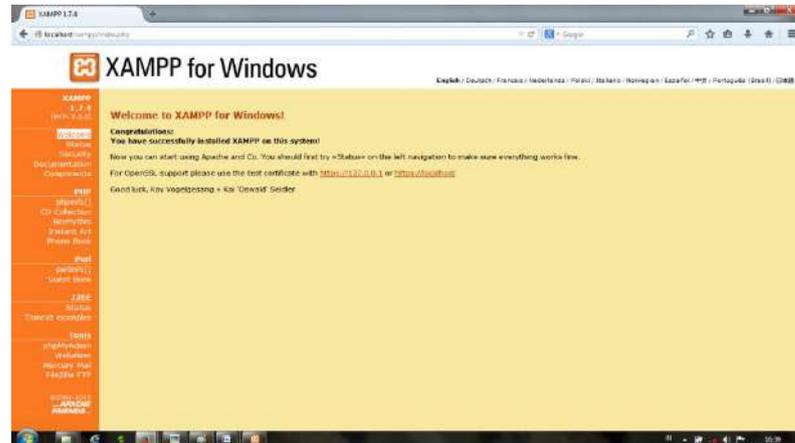
Gambar 4.5 Proses Instalasi XAMPP Selesai

7. Sampai tahan ini berarti kita sudah menginstal XAMPP. Selanjutnya kita akan menggunakan XAMPP sebagai server lokal.
8. Jalankan XAMPP Control Panel yang ada di desktop. Atau melalui start menu -> all program -> apachefriends -> XAMPP Control Panel. Kemudian klik tombol Start pada modul Apache dan MySQL.



Gambar 4.6 XAMPP Control Panel

9. Setelah berhasil menjalankan apache dan MySQL. Selanjutnya buka web browser, lalu ketikkan <http://localhost> pada address bar. Jika berhasil maka akan muncul halaman XAMPP seperti dibawah ini, berarti XAMPP sudah terinstal dengan benar.

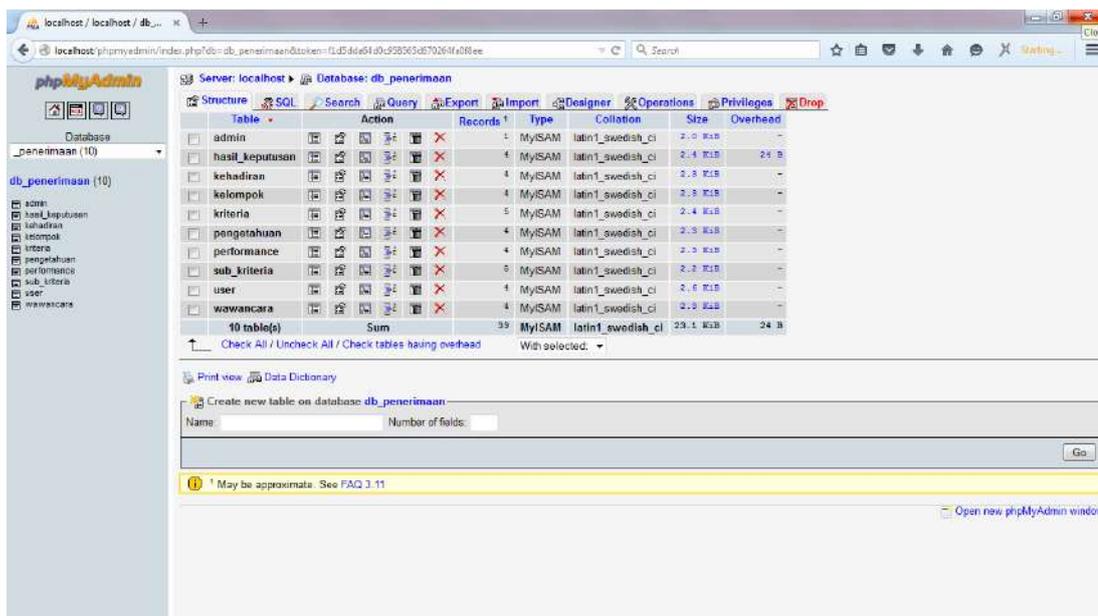


Gambar 4.7 Tampilan XAMPP

10. Selajutnya kita bisa menggunakan XAMPP untuk menjadikan website yang telah dibuat dan mengelola database MySQL melalui menu phpmyadmin.

4.2.2. Tampilan Database Sistem Pengambilan Keputusan

Berikut ini merupakan gambar dari *database* sistem pakar yang diakses oleh *admin* melalui PHP Myadmin, *database* db_penerimaan yang dibuat oleh *admin* untuk menampung data-data yang berhubungan segala data yang diperlukan.

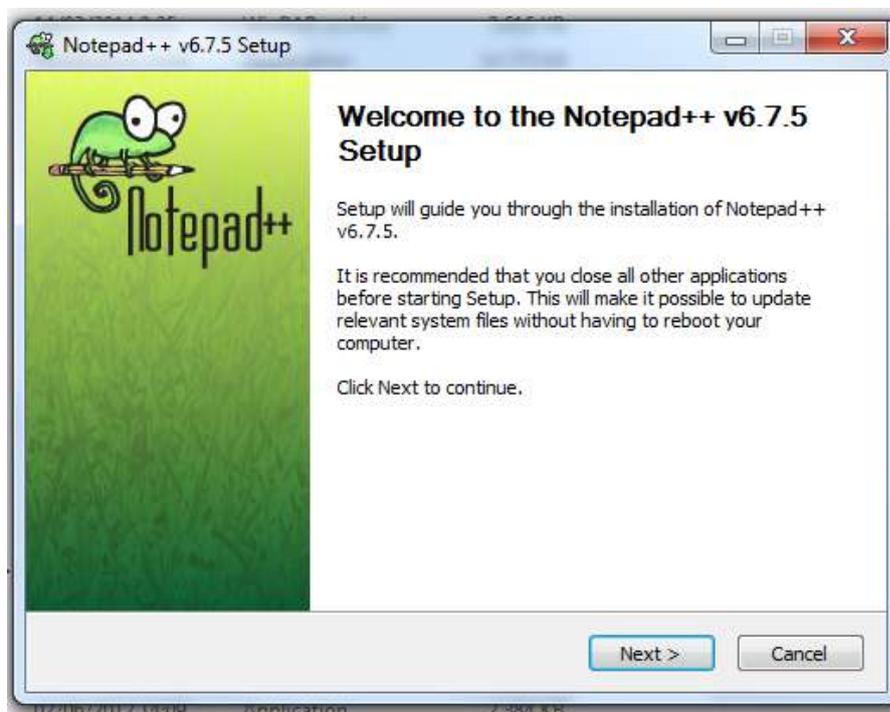


Gambar 4.8 Tampilan *Database db_penerimaan*

4.2.3. Proses Instalasi Notepad++

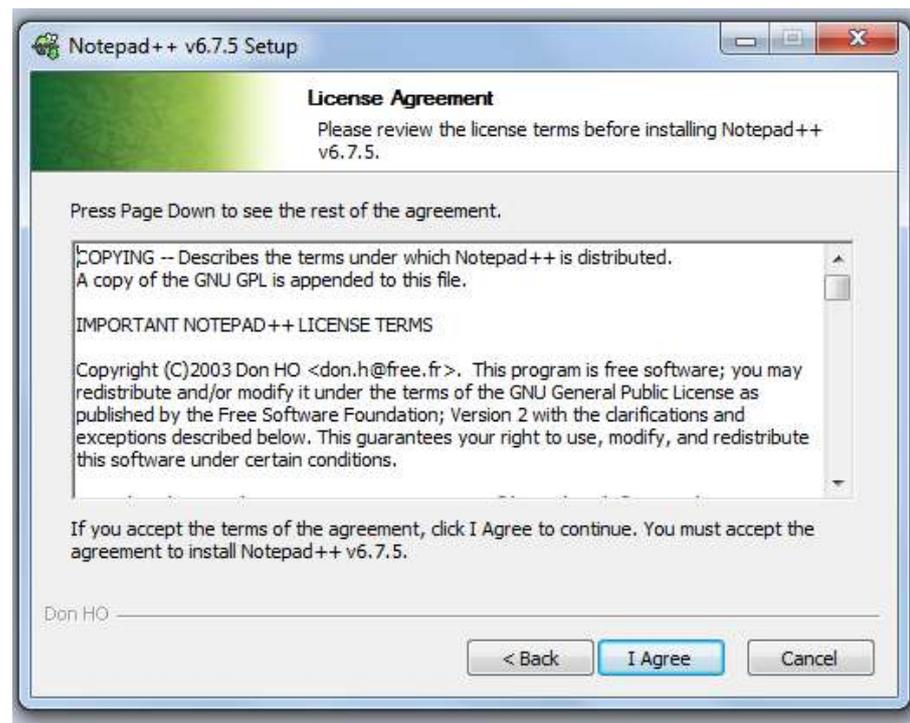
Install selanjutnya adalah instalasi notepad++, proses instalasi adalah sebagai berikut :

1. Klik Next, sehingga muncul gambar seperti dibawah ini :



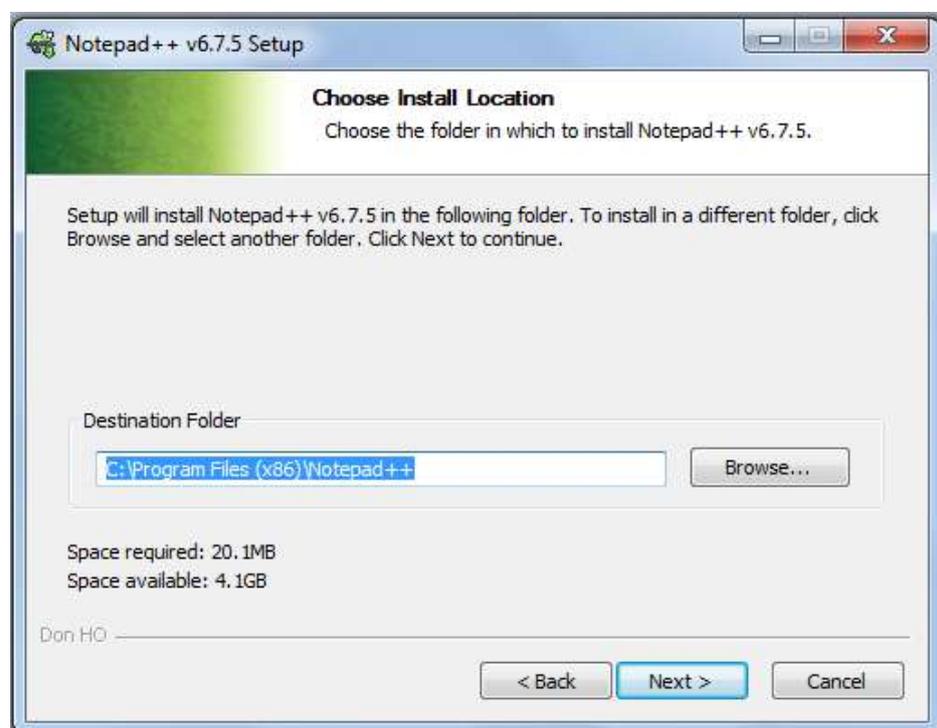
Gambar 4.9 *Welcome Notepad++ Setup*

2. Klik next, sehingga muncul gambar dibawah ini.



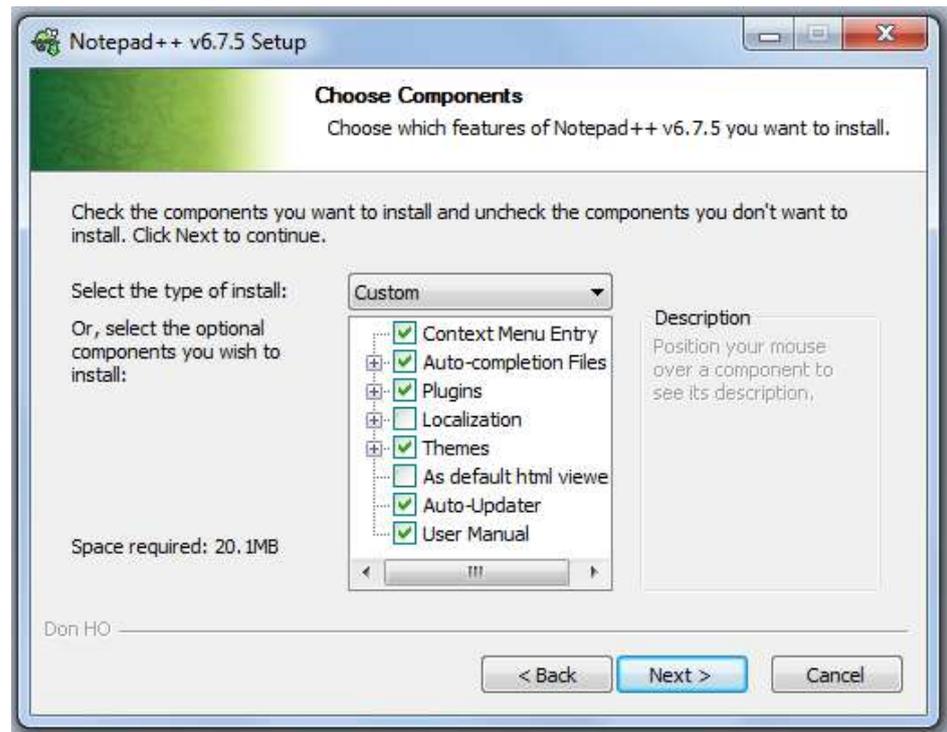
Gambar 4.10 *License Agreement*

3. Klik I Agree, kemudian klik next.



Gambar 4.11 *Pemilihan Lokasi Instalasi*

4. Klik Next,



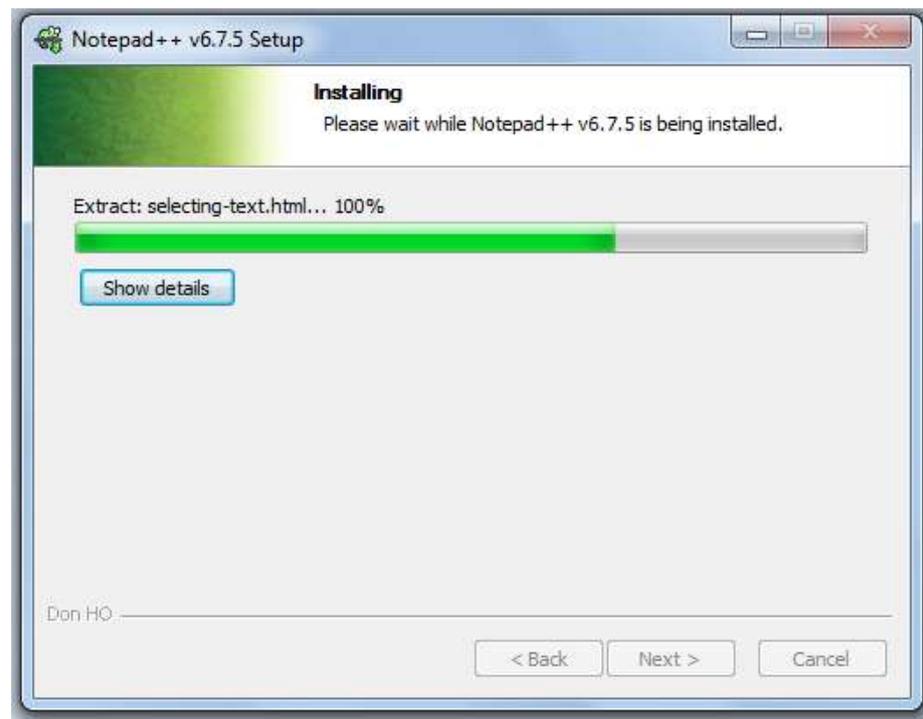
Gambar 4.12 Pemilihan Instalasi Komponen

5. Klik Next



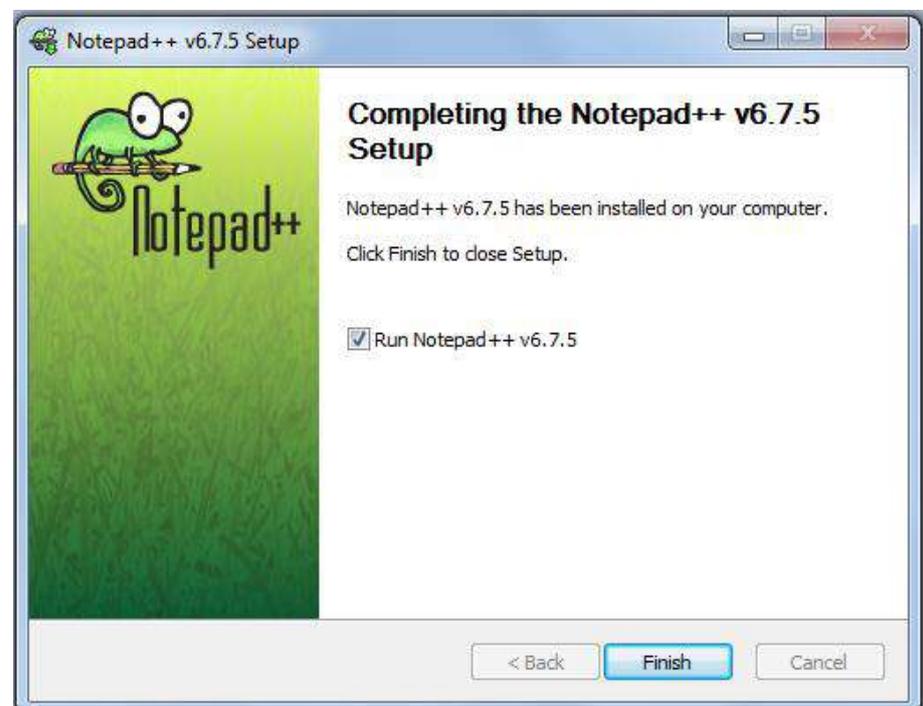
Gambar 4.13 Pemilihan Instalasi Komponen

7. Klik Install, tunggu hingga proses ekstraksi selesai.



Gambar 4.14 Proses Ekstraksi

8. Klik Finish, maka instalasi notepad++ berhasil.



Gambar 4.15 Instalasi Notepad++ Selesai

4.3. Implementasi Pengujian Sistem Pengambilan Keputusan

Pengujian dan implementasi sistem bertujuan untuk melihat apakah sistem yang dirancang sudah sesuai dengan apa yang diinginkan atau belum, setelah dilakukannya pengujian dan implementasi, kualitas sebuah sistem akan terlihat. Berikut ini adalah implementasi dari perancangan sistem pengambilan keputusan penerimaan anggota baru UKM IT Cybernetix.

1. Halaman Awal Website.

Halaman awal website merupakan halaman yang akan ditampilkan ketika pertama kali membuka sistem aplikasi pengambilan keputusan ini. form login yang akan digunakan *user*, baik *administrator* maupun calon anggota.



Gambar 4.16 Halaman Awal Website

2. Halaman Index Admin

Halaman index *admin* merupakan halaman awal yang digunakan *administrator* setelah login. Halaman ini berisikan keterangan selamat datang kepada *administrator*.



Gambar 4.17 Halaman Index Admin

3. Halaman Panduan

Halaman ini berisikan panduan yang digunakan oleh *admin* dalam membandingkan kriteria yang satu dengan yang lainnya.



Gambar 4.18 Halaman Panduan

4. Halaman Registrasi User

Halaman ini berisikan data diri *user* yang akan dimasukkan kedalam *database* oleh *admin*. *Admin* hanya memasukkan *username* beserta *password* yang nantinya akan digunakan calon anggota untuk melengkapi data dirinya masing-masing.

620fitri/admin/index2.php?page=user

Form Registrasi User

Username

Password

Nama Lengkap

Nip

Baglan

Gotongan

No. Handphone

Alamat

Gambar 4.19 Halaman Registrasi *User*

5. Halaman Kriteria Pengambilan Keputusan

Halaman ini berisikan data kriteria yang digunakan dalam pengambilan keputusan beserta nilainya yang telah didapatkan melalui proses perbandingan.

Kode Kriteria	Nama Kriteria	Nilai Kriteria
K001	Masa Dinas	0.2964
K002	Loyalitas	0.1659
K003	Wawancara	0.0934
K004	Administrasi	0.0512
K005	Komitmen	0.3931

Gambar 4.20 Halaman Kriteria Pengambilan Keputusan

6. Halaman Sub Kriteria Pengambilan Keputusan

Halaman ini berisikan data sub kriteria yang digunakan dalam pengambilan keputusan beserta nilainya yang telah didapatkan melalui proses perbandingan.



Kode Kriteria	Sangat Baik	Baik	Cukup Baik	Kurang Baik
K001	1.0000	0.5217	0.2534	0.1171
K002	1.0000	0.6031	0.3633	0.1561
K003	1.0000	0.5170	0.2185	0.1001
K004	1.0000	0.5707	0.2385	0.1107
K005	1.0000	0.5949	0.3449	0.2056

Gambar 4.21 Halaman Sub Kriteria Pengambilan Keputusan

7. Halaman Daftar User

Halaman ini berisikan daftar nama calon anggota yang telah dimasukan serta yang telah melengkapi data dirinya.



No	Username	Password	Nama Lengkap	Bagian	No Handphone	Email	Aksi
1	irfan	123	IRFANDI RICON	TES	085263446902	irfandirico@yahoo.co.id	X
2	ade	123	Ade Eka Putra	TES	085263446821	arikumiawan994@yahoo.co.id	X
3	job	123	job	TES	0987543	zildan@gmail.com	X
4	lek	123	lek	TES	00776	082390764529	X

Gambar 4.22 Halaman Daftar User

8. Halaman Seleksi

Halaman ini berisikan data dari *user* yang akan dilakukan seleksi. Pada halaman ini terdapat beberapa aturan dan keterangan dari penilaian yang berada pada sisi kiri website. Keterangan ini jugalah nantinya yang akan dijadikan pedoman dalam penilaian.

Gambar 4.23 Halaman Seleksi

9. Halaman Proses Keputusan Kriteria

Halaman ini berisikan data perbandingan kriteria yang digunakan berupa nilai-nilai seperti pada halaman panduan yang telah ditampilkan. Nilai inilah yang nantinya akan diproses sesuai dengan rumus perhitungan metode *Analytic Hierarchy Process* (AHP) yang telah dimasukkan kedalam sistem aplikasi

Kriteria	K001	K002	K003	K004	K005	Aksi
K001	1	3	4	5	0.5	
K002	0.33	1	3	4	0.33	
K003	0.25	0.33	1	3	0.25	
K004	0.2	0.25	0.33	1	0.2	
K005	2	3	4	5	1	

Gambar 4.24 Halaman Proses Keputusan Kriteria

10. Proses Keputusan Sub Kriteria

Halaman ini berisikan data perbandingan sub kriteria yang digunakan berupa nilai-nilai seperti pada halaman panduan yang telah ditampilkan. Nilai

ini yang nantinya akan diproses sesuai dengan rumus perhitungan metode *Analytic Hierarchy Process* (AHP) yang telah dimasukkan kedalam sistem aplikasi.

Matriks Perbandingan Sub Kriteria

Masa Dinas	SP01	SP02	SP03	SP04	Aksi
SP01	1	3	4	6	
SP02	0.33	1	3	5	
SP03	0.25	0.33	1	3	
SP04	0.17	0.2	0.33	1	

KETERANGAN
01 : SANGAT BAIK
02 : BAIK
03 : CUKUP BAIK
04 : KURANG BAIK

Proses

Loyalitas	SPR01	SPR02	SPR03	SPR04	Aksi
-----------	-------	-------	-------	-------	------

Gambar 4.25 Halaman Proses Keputusan Sub Kriteria

11. Hasil Proses Keputusan Perbandingan

Halaman ini berisikan hasil dari proses perhitungan yang telah dilakukan oleh sistem baik itu perbandingan kriteria, maupun perbandingan sub kriteria dengan menggunakan rumus perhitungan yang sama.

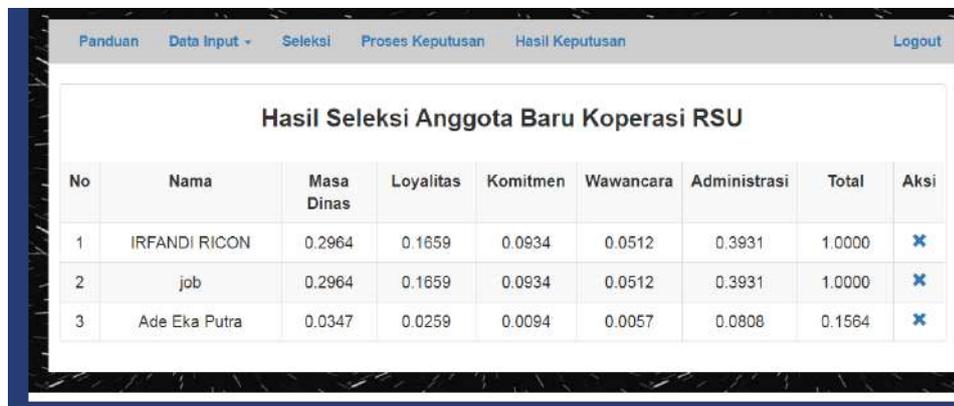
Hasil Seleksi Anggota Baru Koperasi RSU

No	Nama	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi	Total	Aksi
1	IRFANDI RICON	0.2964	0.1659	0.0934	0.0512	0.3931	1.0000	
2	job	0.2964	0.1659	0.0934	0.0512	0.3931	1.0000	
3	Ade Eka Putra	0.0347	0.0259	0.0094	0.0057	0.0808	0.1564	

Gambar 4.26 Halaman Proses Keputusan Perbandingan

12. Halaman Hasil Seleksi

Halaman ini berisi daftar nama calon anggota yang telah dilakukan proses seleksi oleh *administrator*.



No	Nama	Masa Dinas	Loyalitas	Komitmen	Wawancara	Administrasi	Total	Aksi
1	IRFANDI RICON	0.2964	0.1659	0.0934	0.0512	0.3931	1.0000	✕
2	job	0.2964	0.1659	0.0934	0.0512	0.3931	1.0000	✕
3	Ade Eka Putra	0.0347	0.0259	0.0094	0.0057	0.0808	0.1564	✕

Gambar 4.27 Halaman Hasil Seleksi

13. Halaman Index User

Halaman index *user* merupakan halaman awal yang digunakan calon anggota setelah login. Halaman ini berisikan keterangan selamat datang kepada calon anggota.



Gambar 4.28 Halaman Index User

14. Halaman Lengkapi Data User

Halaman ini berisikan data diri dari calon anggota yang telah terdaftar yang nantinya data diri yang telah dimasukan dapat diproses oleh *administrator*.

Gambar 4.29 Halaman Lengkapi Data *User*

15. Halaman Hasil Seleksi *User*

Halaman ini berisikan keterangan dari data yang telah diproses oleh *administrator* menggunakan sistem apakah calon anggota terima atau tidak pada penerimaan anggota baru.

Gambar 4.30 Halaman Seleksi *User*

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari analisa, perancangan aplikasi sistem pengambilan keputusan dalam penerimaan anggota baru pada Koperasi RSU menggunakan metode *Analytic Hierarchy Process* (AHP) dengan bahasa pemrograman php & mysql, maka dapat diambil kesimpulan, yaitu :

1. Dengan sistem pendukung keputusan yang menerapkan metode AHP berbasis web dalam pengambilan keputusan sangat membantu panitia dalam menghasilkan keputusan yang mutlak.
2. Dengan telah ditentukannya kriteria dalam penerimaan anggota baru dapat membantu panitia menghasilkan anggota baru yang berbakat pada setiap bidang keahliannya.
3. Dengan adanya aplikasi sistem pengambilan keputusan ini yang menerapkan metode AHP dapat menghasilkan anggota baru yang berdedikasi tinggi serta memiliki loyalitas yang besar pada Koperasi RSU sesuai dengan kriteria yang telah ditetapkan dalam penerimaan.

5.2 Keterbatasan Sistem

Dalam perancangan aplikasi pengambilan keputusan ini ada beberapa keterbatasan sistem, antara lain :

1. Sistem perhitungan yang dilakukan hanya diproses oleh administrator.
2. Proses pemasukan nilai perbandingan masih dilakukan secara manual.
3. Sistem penilaian dapat diproses apabila semua tes seleksi telah dilakukan yang menghasilkan nilai/range dari rata-rata penilaian.

5.3 Saran

Demi tercapainya peningkatan efektivitas kerja di Koperasi RSU dalam menentukan keputusan penerimaan anggota baru, maka penulis mengemukakan beberapa saran yang diharapkan dapat menjadi bahan pertimbangan. Adapun saran-saran tersebut adalah :

1. Melatih sumber daya manusia dalam mengembangkan sistem yang baru sehingga sistem dapat berjalan dengan lancar.
2. Perlu adanya koordinasi dan kerja sama yang lebih baik antara sub bagian yang satu dengan yang lain sehingga sistem yang baru dapat dilaksanakan sesuai dengan tujuan yang ingin dicapai.
3. Pemakaian Aplikasi MySQL pada sistem pengambilan keputusan untuk mengambil keputusan penerimaan calon anggota sebaiknya didukung oleh perangkat yang memadai. Hal ini diperlukan agar sistem dapat bekerja dengan lancar tanpa terjadi hal-hal yang tidak di inginkan dalam menjalankan aplikasi tersebut.
4. Memerlukan waktu untuk penyesuaian dalam mengembangkan sistem ini karena sistem ini baru pertama kali di gunakan dalam Koperasi RSU

DAFTAR PUSTAKA

- Andi, S, Rosa A. dan M. Shalahuddin, 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung : Informatika. Sidik
- Andriani, Y., Ramli, N. M., Syamsumir, D. F., Kassim, M. N. I., Jaafar, J., Aziz, N. A., ... & Mohamad, H. (2019). Phytochemical analysis, antioxidant, antibacterial and cytotoxicity properties of keys and cores part of *Pandanus tectorius* fruits. *Arabian Journal of Chemistry*, 12(8), 3555-3564.
- Betha, 2012. *Pemrograman Web PHP*, Bandung : Informatika.
- B. Nugroho. (2005). *Database Relasional dengan MySQL*. C.V Andi Offset : Yogyakarta. Fahmi, I. (2016). *Teori dan Teknik Pengambilan Keputusan: Kualitatif dan Kuantitatif*. Jakarta: Rajawali Pers.
- Fowler, Martin. 2004. *UML Distilled*. Yogyakarta : Andi.
- Hanum, Yuhilza. *Software Engineering (Rekayasa Perangkat Lunak)*. Jakarta :Erlangga. Komang, I Setia Buana.2014. *Jago Pemrograman PHP*. Bandung: Dunia Komputer.
- Magdalena, Hilyah. 2012. *Sistem pendukung keputusan untuk menentukan mahasiswa lulusan terbaik di perguruan tinggi (studi kasus STMIK ATMA LUHUR Pangkal Pinang)*.Yogyakarta: Seminar Nasional Teknologi Informasi dan Komunikasi.
- Mulyanto, Aunur R. 2008. *Rekayasa Perangkat Lunak, Jilid 1*. Jakarta: Direktur Pembinaan Sekolah Menengah Kejuruan. Oktavian, Diar, Puji. 2013. *Membuat Powerfull Menggunakan PHP*. Yogyakarta: Mediakom.
- Peranginangin, Kasiman. 2014. *Aplikasi Web dengan PHP & MySQL*. Yogyakarta
- Ramadhan, Z., Zarlis, M., Efendi, S., & Siahaan, A. P. U. (2018). Perbandingan Algoritma Prim dengan Algoritma Floyd-Warshall dalam Menentukan Rute Terpendek (Shortest Path Problem). *JURIKOM (Jurnal Riset Komputer)*, 5(2), 135-139.
- Rusdina, R., Syafarina, G. A., & Amin, M. (2020). PROTOTYPE APLIKASI BANJAR BERBASIS ANDROID STUDIO SEBAGAI SALAH SATU PETUNJUK WISATA DI BANJARMASIN. *Technologia: Jurnal Ilmiah*, 11(1), 59-63
- Santosa, A., Sitopu, M. W., Sirait, D. N., & Nasution, D. (2021). Analysis of Damage to Localizer Equipment (Case Study of Sultan Iskandar Muda Airport, Banda Aceh). *Budapest International Research and Critics Institute (BIRCI-Journal): Humanities and Social Sciences*, 4(3), 7054-7061.
- Subakti, Irfan. 2002. *Sistem Pendukung Keputusan (Decision Support System)*. Teknik Informatika Institut Teknologi Sepuluh Nopember.

Syukur, Abdul, Tyas Catur P dkk., 2010. Penerapan Metode Analytical Hierarchi Process dalam penerimaan karyawan pada PT. Pasir Besi Indonesia. Pascasarjana Teknik Informatika Univeristas Dian Nuswantoro

Syafi'I, M. (2005). Aplikasi Database Dengan PHP 5 MySQL PostgreSQL Oracle. Yogyakarta : Andi.

Trisnani, A. A., Anwar, D. U., Ramadhani, W., Manurung, M. M., & Siahaan, A. P. U. (2018). Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi Menerapkan Metode Vise Kriterijumska Optimizajica I Kompromisno Resenje (VIKOR). JURIKOM (Jurnal Riset Komputer), 5(2), 85-90.

Triyono, S., Putra, R. M., Waluyo, S., & Amin, M. (2019, November). The effect of three different containers of nutrient solution on the growth of vegetables cultured in DFT hydroponics. In IOP Conference Series: Earth and Environmental Science (Vol. 355, No. 1, p. 012092). IOP Publishing.