



**SISTEM PENDUKUNG KEPUTUSAN SELEKSI CALON PEGAWAI
SIPIL BARU DI POLDA SUMATERA UTARA MENGGUNAKAN
METODE MULTIFACTOR EVALUATION PROCESS (MFEP)**

SKRIPSI

**Diajukan Untuk Memenuhi Persyaratan Ujian
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi**

Oleh :
**Yuda Syahputra
1924370303**

**PROGRAM STUDI SISTEM KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2021**

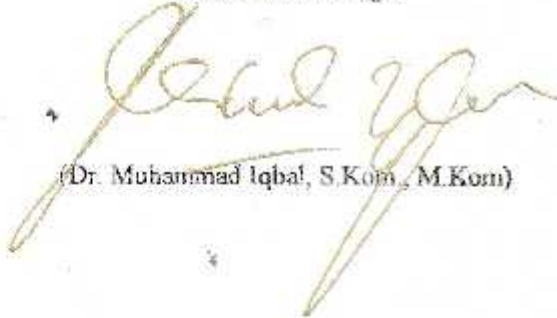
LEMBAR PENGESAHAN SKRIPSI
SISTEM PENDUKUNG KEPUTUSAN SELEKSI CALON PEGAWAI SIPIL BARU
DI POLDA SUMATERA UTARA MENGGUNAKAN METODE MULTIFACTOR
EVALUATION PROCESS (MFEP)

Disusun oleh :

Nama : Yuda Syahpatra
NPM : 1924370303
Program Studi : Sistem Komputer

Skripsi Telah Disetujui Oleh Dosen Pembimbing Skripsi
pada tanggal 24-02-2021

Dosen Pembimbing 1



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

Dosen Pembimbing 2




(M. Donni Lesmana Sihaban, S.Kom., M.Kom)

Mengetahui,

Dekan Fakultas Sains dan Teknologi



Ketua Program Studi Komputer



(Eko Hariyanto, 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	(1)RAKREDITASI)
ELEKTRO PROGRAM STUDI	(1)RAKREDITASI)
ARSITEKTUR PROGRAM STUDI	(1)RAKREDITASI)
SISTEM KOMPUTER PROGRAM	(1)RAKREDITASI)
STUDI TEKNIK KOMPUTER	(1)RAKREDITASI)
PROGRAM STUDI	(1)RAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR

Saya yang bertanda tangan di bawah ini :

Nama Lengkap : YUDA SYAHPUTRA
 Tempat/Tgl. Lahir : AHK KANOPAN / 07 Desember 1978
 Nomor Pokok Mahasiswa : 1924370303
 Program Studi : Sistem Komputer
 Konsentrasi : Sistem Kendali Komputer
 Jumlah Kredit yang telah dicapai : 133 SKS, IPK 3.12
 Nomor Hp : 081255219293

Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut :


No.	Judul
1	SISTEM PENDUKUNG KEPUTUSAN SELEKSI CATON PECAJAWA SIPIL BARU DI POLDA SUMATERA UTARA MENGGUNAKAN METODE MULTIFACTOR EVALUATION PROCESS (MHEP)

Catatan : Diisi Oleh Dosen Jika Ada Perubahan Judul

*Coret Yang Tidak Perlu

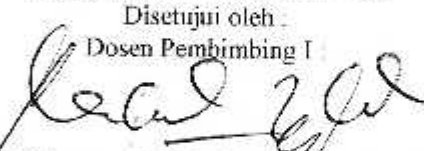

 (Cahyo Pramono SE., MM.)


Medan, 22 Juli 2020

Permohon,

 (Yuda Syahputra)

Tanggal :
 Disahkan oleh :
 Dekan

 (Hamdani, S.T., MT.)

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

 (M. Donni Lesmana Sijaan, 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 : YUDA SYAHPUTRA
NPM : 1924370303
Program Studi : Sistem Komputer
Jenjang Pendidikan : Strata Satu
Dosen Pembimbing : Muhammad Donni Lesmana Siahaan, S.Kom, M.Kom
Judul Skripsi : Sistem Pendukung Keputusan Seleksi Calon Pegawai Sipil Baru di Polda Sumatera Utara Menggunakan Metode Multifactor Evaluation Process (MFEP)

Tanggal	Pembahasan Materi	Status	Keterangan
17 Juli 2020	Acc Sempro	Revisi	
06 Agustus 2020	Acc Bab II dan III lanjut ke bab berikutnya	Revisi	
06 Agustus 2020	Acc Bab II dan III lanjut ke bab berikutnya	Revisi	
18 September 2020	ACC SEMINAR HASIL	Disetujui	
23 Februari 2021	acc meja hijau	Disetujui	
12 Oktober 2021	acc jilid	Disetujui	

Medan, 29 Desember 2021
Dosen Pembimbing,



Muhammad Donni Lesmana Siahaan,
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 : YUDA SYAHPUTRA
NPM : 1924370303
Program Studi : Sistem Komputer
Jenjang : Strata Satu
Pendidikan :
Dosen Pembimbing : Dr Muhammad Iqbal, S.Kom., M.Kom.
Judul Skripsi : Sistem Pendukung Keputusan Seleksi Calon Pegawai Sipil Baru di Polda Sumatera Utara Menggunakan Metode Multifactor Evaluation Process (MFEP)

Tanggal	Pembahasan Materi	Status	Keterangan
16 Juli 2020	Acc samprom	Revisi	
05 Agustus 2020	Acc Bab II	Revisi	
25 Agustus 2020	Acc bab IV dan V	Revisi	
25 Agustus 2020	Acc seminar hasil	Disetujui	
24 Februari 2021	Acc sidang	Disetujui	
19 Oktober 2021	Acc Jilid	Disetujui	

Medan, 29 Desember 2021
Dosen Pembimbing,



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

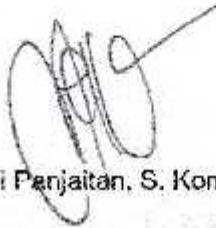
KARTU BEBAS PRAKTIKUM
Nomor. 1143/BL/LAKO/2021

Tanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

	: YUDA SYAHPUTRA
	: 1924370303
Semester	: Akhir
Kelas	: SAINS & TEKNOLOGI
Prodi	: Sistem Komputer

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

Medan, 29 Desember 2021
Ka. Laboratorium



Elva Sari Penjaitan, S. Kom., M.Kom.





SURAT BEBAS PUSTAKA
NOMOR: 3719/PERP/BP/2021

Perpustakaan Universitas Pembangunan Panca Budi menerangkan bahwa berdasarkan data pengguna perpustakaan
ma saudara/i:

: YUDA SYAHPUTRA

: 1924370303

Semester : Akhir



S : SAINS & TEKNOLOGI

Prodi : Sistem Komputer

annya terhitung sejak tanggal 23 Februari 2021, dinyatakan tidak memiliki tanggungan dan atau pinjaman buku
tidak lagi terdaftar sebagai anggota Perpustakaan Universitas Pembangunan Panca Budi Medan.

Medan, 23 Februari 2021

Diketahui oleh,
Kepala Perpustakaan

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

Dokumen: FM-PERPUS-06-01

isi : 01

Efektif : 04 Juni 2015

Hal : Permohonan Meja Hijau

Medan, 29 Desember 2021
 Kepada Yth : Bapak/Ibu Dekan
 Fakultas SAINS & TEKNOLOGI
 UNPAB Medan
 Di -
 Tempat


Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama : YUDA SYAHPUTRA
 Tempat/Tgl. Lahir : Aek Kanopan / 7 Desember 1978
 Nama Orang Tua : SUPARDI
 N. P. M : 1924370303
 Fakultas : SAINS & TEKNOLOGI
 Program Studi : Sistem Komputer
 No. HP : 081265219293
 Alamat : Komplek Abdul Hamid Nasution Blok XV No. K.3 Desa
 Lalang Kec. Sunggal Kab. Deli Serdang

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul Sistem Pendukung Keputusan Seleksi Calon Pegawai Sipil Baru di Polda Sumatera Utara Menggunakan Metode Multifactor Evaluation Process (MFEP), Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntut 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 dijiid (x 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jiid kertas jeruk 5 exemplar untuk penguji (bentuk dan warna penjilidan 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 7 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 yang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan rincian sbh :

1. [102] Ujian Meja Hijau	: Rp.	1,000,000
2. [170] Administrasi Wisuda	: Rp.	1,750,000
Total Biaya	: Rp.	2,750,000

Ukuran Toga : 

Diketahui/Disetujui oleh :

Hormat saya



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



YUDA SYAHPUTRA
 1924370303

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 (ast) - Mts.ybs.

Plagiarism Detector v. 1857 - Originality Report 2/24/2021 3:06:17 PM

Author: YUDA SYAHPUTRA_1924370303_SISTEM KOMPUTER.doc Submitted to: Universitas Pembangunan Panca Budi_License04

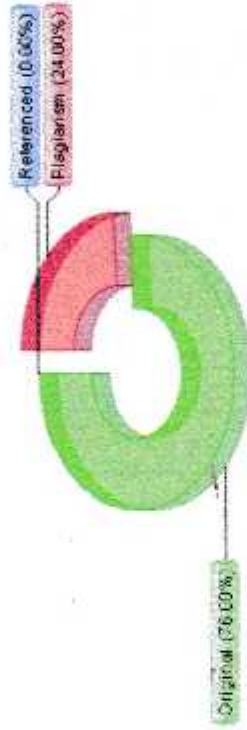
Comparison Method: Rewrite

Check type: Internet Check



Enabled document body analysis

Relation chart



Plagiarism Detector v. 1857

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-LJMA-067/2

Revisi

: 00

Tgl Eff

: 23 Jan 2019

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Yuda Syahputra
NPM : 1924370303
Program Studi : Sistem Komputer
Judul Skripsi : Sistem Pendukung Keputusan Seleksi Calon Pegawai Sipil Baru Di Polda Sumatera Utara Menggunakan Metode Multifactor Evaluation Process (MFEP).

Dengan ini menyatakan bahwa :

1. Skripsi ini merupakan hasil karya tulis saya sendiri dan bukan merupakan hasil karya orang lain (plagiat).
2. Memberikan izin hak bebas Royalti Non-Eksklusif kepada UNPAB untuk menyimpan, mengalih media / formatkan, mengelola, mendistribusikan, dan mempublikasikan karya skripsinya melalui internet atau media lain bagi kepentingan akademis.

Pernyataan ini saya buat dengan penuh tanggungjawab dan saya bersedia menerima konsekuensi apapun sesuai dengan aturan yang berlaku apabila dikemudian hari diketahui bahwa pernyataan ini tidak benar.

Medan, 24 Februari 2021


METERAI
TEMPEL
D4B19AJX738250398 Yuda Syahputra)

SURAT PERNYATAAN ORISINALITAS

Dengan ini menyatakan bahwa dalam skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar kesarjanaan di dalam perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis di acu dalam skripsi ini dan disebutkan dalam daftar pustaka.

Medan, 24 Februari 2021

Yang membuat pernyataan



Yuda Svahputra
1924370303

ABSTRAK

Sistem informasi memiliki peranan penting untuk menghasilkan informasi yang lebih cepat dan akurat karena dapat mendukung keberhasilan suatu instansi pemerintah dalam mencapai tujuannya. Pada masa sekarang ini instansi pemerintah telah memanfaatkan teknologi informasi dalam mendukung proses operasionalnya sehingga meningkatkan efisiensi tenaga dan waktu. Perkembangan teknologi komputer, ilmu pengetahuan dan juga informasi dimasa sekarang ini berkembang sangat cepat seiring dengan penemuan-penemuan baru. dalam penentuan pegawai sipil yang berkualitas terdapat beberapa faktor yang menjadi penilaian. Penilaian ini didasarkan pada hasil proses seleksi. Sehingga tidak menutup kemungkinan bahwa pengambilan suatu keputusan merupakan suatu yang sangat penting dalam menentukan keputusan yang harus diambil. Dengan tujuan untuk membangun dan memberikan sebuah sistem penunjang keputusan yang mempunyai kemampuan analisa pemilihan pegawai sipil baru dengan menggunakan Metode Multifactor Evaluation Process (MFEP). Metode ini bekerja dengan memecah-mecah suatu situasi yang kompleks, tidak terstruktur, kedalam bagian-bagian komponennya. Dimana masing-masing kriteria dalam hal ini faktor - faktor kriteria dan keputusan alternatif dalam hal ini para calon pegawai dibandingkan satu dengan yang lainnya sehingga memberikan output nilai intensitas prioritas yang menghasilkan nilai terhadap setiap pegawai sipil baru. Sistem penunjang keputusan ini membantu dan memberikan alternatif dalam melakukan penilaian setiap calon pegawai sipil baru dengan menentukan kriteria dan nilai bobot. Hal ini berguna untuk memudahkan pengambilan keputusan yang terkait dengan masalah penyeleksian pegawai sipil baru, sehingga akan di dapatkan calon pegawai yang paling layak diterima sesuai dengan yang diinginkan.

Kata Kunci : Sistem Pendukung Keputusan, Multifactor Evaluation Process, Polda Sumut

DAFTAR ISI

HALAMAN JUDUL	
LEMBAR PENGESAHAN	i
LEMBAR PERSETUJUAN.....	ii
PERNYATAAN KEASLIAN TULISAN	iii
ABSTRAK.....	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiv

BAB I PENDAHULUAN

1.1. Latar Belakang Masalah.....	1
1.2. Perumusan Masalah	2
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	3

BAB II LANDASAN TEORI

2.1 Rekayasa Perangkat Lunak	4
2.1.1 Perangkat Lunak.....	6
2.1.2 Proses Rekayasa Perangkat Lunak	8

2.2	<i>Unified Modelling Language (UML)</i>	10
2.2.1	Sejarah <i>Unified Modelling Language (UML)</i>	10
2.2.2	<i>Use Case Diagram</i>	11
2.2.3	<i>Class Diagram</i>	15
2.2.4	<i>Sequence Diagram</i>	17
2.2.5	<i>Communication Diagram</i>	20
2.2.6	<i>Statechart Diagram</i>	21
2.2.7	<i>Activity Diagram</i>	22
2.2.8	<i>Deployment Diagram</i>	23
2.3	Sistem Pendukung Keputusan	24
2.3.1	Definisi Sistem Pendukung Keputusan.....	24
2.3.2	Karakteristik Sistem Pendukung Keputusan.....	26
2.3.3	Arsitektur Sistem Pendukung Keputusan	28
2.4	Multifactor Evaluation Process (MFEP).....	29
2.4.1	Kelebihan Multifactor <i>Evaluation Process (MFEP)</i>	30
2.4.2	Kekurangan <i>Multifactor Evaluation Process (MFEP)</i>	30
2.4.3	Prosedur Perhitungan MFEP.....	31
2.5	Bahasa Pemograman.....	33
2.5.1	Konsep Dasar HTML.....	33
2.5.2	Sejarah HTML.....	34
2.5.3	Struktur Dokumen HTML.....	36
2.5.4	<i>Hypertext Processor (PHP)</i>	37
2.5.5	Konsep Dasar PHP	38
2.6	<i>Database Management System (DBMS)</i>	40

2.6.1	Definisi MySQL	41
2.6.2	Kelebihan dari MySQL	41
2.7	Teori <i>Seleksi</i>	42

BAB III METODOLOGI PENELITIAN

3.1	Kerangka Penelitian	43
3.2	Tahapan Penelitian	44
3.2.1	Penelitian Pendahuluan	44
3.2.2	Pengumpulan Data	44
3.2.2.1	Waktu Penelitian	44
3.2.2.2	Tempat Penelitian	45
3.2.2.3	Metode Penelitian	45
3.3	Analisa	46
3.3.1	Analisa Data.....	46
3.3.2	Analisa Proses	48
3.3.3	Analisa Sistem	49
3.4	Perancangan UML	53
3.4.1	<i>Use Case Diagram</i>	54
3.4.2	<i>Class Diagram</i>	61
3.4.3	<i>Sequence Diagram dan Collaboration Diagram</i>	64
3.4.4	<i>Statechart Diagram</i>	68
3.4.5	<i>Activity Diagram</i>	70
3.4.6	<i>Deployment Diagram</i>	71
3.5	Perancangan <i>Interface</i>	71

3.5.1	Tampilan Halaman Login	72
3.5.2	Tampilan Halaman Beranda.....	72
3.5.3	Tampilan Halaman Data Kriteria	73
3.5.4	Tampilan Halaman Data Alternatif.....	74
3.5.5	Tampilan Halaman Hasil Analisa.....	74
3.5.6	Tampilan Halaman Edit Password	75

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1	Implementasi.....	76
4.1.1	Persiapan Implementasi.....	76
4.1.2	<i>Batasan</i> Implementasi.....	77
4.1.3	Implementasi <i>Sistem</i>	77
4.1.3.1	Persiapan <i>Web Server</i>	77
4.1.3.2	Import Database.....	83
4.2	Pengujian	86
4.2.1	Pengujian Aplikasi.....	86
4.2.2	Pengujian Interface	87

BAB V PENUTUP

5.1	Kesimpulan	92
5.2	Keterbatasan Sistem.....	92
5.3	Saran	93

DAFTAR PUSTAKA	95
-----------------------------	-----------

DAFTAR GAMBAR

Gambar 2.1 Tahapan Umum Rekayasa Perangkat Lunak	9
Gambar 2.2 : Arsitektur DSS	29
Gambar 2.3 Contoh <i>Source Code</i> HTML	36
Gambar 2.4 Ilustrasi Proses Kerja PHP	38
Gambar 2.5 Contoh <i>Source Code</i> PHP	39
Gambar 2.6 Contoh Hasil dari <i>Source Code</i> PHP.....	40
Gambar 3.1 Kerangka Kerja Penelitian	43
Gambar 3.2 <i>Use Case Diagram</i>	61
Gambar 3.3 Class Diagram	62
Gambar 3.4 <i>Sequence Diagram</i> User untuk Mengelola Data Kriteria.....	65
Gambar 3.5 <i>Collaboration Diagram</i> User untuk Mengelola Data Kriteria	66
Gambar 3.6 <i>Sequence Diagram</i> User untuk Mengelola Data Alternatif	66
Gambar 3.7 <i>Collaboration Diagram</i> User untuk Mengelola Data Kriteria.....	67
Gambar 3.8 <i>Sequence Diagram</i> User untuk Mengelola Data Solusi	67
Gambar 3.9 <i>Collaboration Diagram</i> Admin untuk Mengelola Data Solusi	68
Gambar 3.10 <i>Statechart</i> Diagram Form Tambah Kriteria	68
Gambar 3.11 <i>Statechart</i> Diagram Form Tambah Alternatif	69
Gambar 3.12 <i>Statechart</i> Diagram Form Analisa Hasil	69
Gambar 3.13 Activity Diagram Aplikasi SPK.....	70
Gambar 3.14 Deployment Diagram	71
Gambar 3.15 Halaman Login.....	72
Gambar 3.16 Halaman Beranda	73

Gambar 3.17 Halaman Data Kriteria	73
Gambar 3.18 Halaman Data Alternatif	74
Gambar 3.19 Halaman Hasil Analisa.....	74
Gambar 3.20 Halaman Edit Password	75
Gambar 4.1 Ekstraksi File Xampp.....	78
Gambar 4.2 File Installer.....	78
Gambar 4.3 Pilihan Next Setup.....	79
Gambar 4.4 Packages yang akan Di install.....	79
Gambar 4.5 Pilihan Folder Instalasi.....	80
Gambar 4.6 Pesan Pemberitahuan.....	80
Gambar 4.7 Proses Instalasi	81
Gambar 4.8 Pesan Peringatan akses Mysql.....	81
Gambar 4.9 Pesan Peringatan akses Apache HTTP Server	82
Gambar 4.10 Pesan Pemberitahuan.....	82
Gambar 4.11 Hasil Instalasi	83
Gambar 4.12 <i>Interface</i> aplikasi Xampp	84
Gambar 4.13 <i>Interface</i> halaman awal Xampp.....	84
Gambar 4.14 <i>Interface</i> Halaman Membuat Database	85
Gambar 4.15 <i>Interface</i> Halaman Import Database.....	85
Gambar 4.16 <i>Interface</i> halaman struktur database.....	86
Gambar 4.17 <i>Interface</i> halaman <i>login</i>	87
Gambar 4.18 <i>Interface</i> halaman utama	88
Gambar 4.19 <i>Interface</i> Halaman <i>Input</i> Kriteria	88
Gambar 4.20 <i>Interface</i> Halaman <i>Input</i> Alternatif.....	89

Gambar 4.21 Interface Analisa Hasil	89
Gambar 4.22 <i>Interface</i> Hasil Akhir.....	90
Gambar 4.23 <i>Interface</i> halaman ubah <i>password</i>	90
Gambar 4.24 <i>Interface</i> Halaman <i>About</i>	91

DAFTAR TABEL

Tabel 2.1 Simbol – Simbol Use Case Diagram.....	12
Tabel 2.2 Simbol-simbol Class Diagram	16
Tabel 2.3 Simbol-simbol sequence diagram	17
Tabel 2.4 Simbol-simbol <i>commutation diagram</i>	20
Tabel 2.5 Simbol-simbol statechart diagram	21
Tabel 2.6 Simbol-simbol <i>activity diagram</i>	22
Tabel 2.7 Simbol-simbol deployment diagram.....	24
Tabel 2.8: Pemberian Nilai Bobot Untuk Faktor	31
Tabel 2.9: Evaluasi Faktor	31
Tabel 3.1 Data Faktor Pendukung.....	47
Tabel 3.2 Bobot Faktor Pendukung	49
Tabel 3.3 Proses Perhitungan.....	50
Tabel 3.4 . Evaluasi Untuk Nama Peserta Calon 1	51
Tabel 3.5 . Evaluasi Untuk Nama Peserta Calon 2	51
Tabel 3.6 . Evaluasi Untuk Nama Peserta Calon 3	51
Tabel 3.7 Hasil Total Bobot Evaluasi	52
Tabel 3.8 Hasil Setelah Sorting.....	53
Tabel 3.9 Definisi Aktor	54
Tabel 3.10 Skenario Login.....	55
Tabel 3.11 Skenario Manajemen Alternatif.....	56
Tabel 3.12 Skenario Manajemen Kriteria	57
Tabel 3.13 Skenario Manajemen User	58

Tabel 3.14 Skenario Home.....	59
Tabel 3.15 Skenario Hasil.....	60
Tabel 3.16 Definisi <i>Class</i>	62
Tabel 3.17 Tabel User.....	63
Tabel 3.18 Tabel Kriteria.....	64
Tabel 3.19 Tabel Alternatif.....	64
Tabel 4.1: Pengujian Aplikasi.....	87

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 **“SISTEM PENDUKUNG KEPUTUSAN SELEKSI CALON PEGAWAI SIPIL BARU DI POLDA SUMATERA UTARA MENGGUNAKAN METODE MULTIFACTOR EVALUATION PROCESS (MFEP) ”**.

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 Dr. Muhammad Iqbal, S.Kom., M.Kom. selaku Dosen Pembimbing I dan Bapak Muhammad Donni Lesmana, S.Kom., M.Kom., selaku pembimbing II.
4. Bapak/Ibu Dosen beserta seluruh Staf Universitas Pembangunan Panca Budi Medan yang telah mendidik dan membimbing penulis selama mengikuti perkuliahan.

5. 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.
6. 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, 24 Februari

2021

Penulis,

Yuda Syahputra
NPM : 1924370303

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam penyeleksian masih sering terdapat beberapa calon yang diterima tidak sesuai dengan yang diinginkan. Alasan tersebut menjadikan pembuatan suatu model dalam pengambilan keputusan merupakan hal penting sehingga keputusan yang di ambil merupakan keputusan yang akurat, cermat dan tentu saja menguntungkan bagi yang bersangkutan (Muhammad Dahria, dkk. 2014).

Dan dalam menentukan calon pegawai sipil yang layak dan dinyatakan lolos untuk masuk dalam proses ke dunia pekerjaan di lingkungan Polda Sumatera Utara, maka penulis akan menganalisa serta merancang Sistem Pendukung Keputusan dalam bentuk aplikasi berbasis web. Untuk keputusan yang berpengaruh secara strategis lebih dianjurkan menggunakan sebuah pendekatan kuantitatif seperti Metode Multi Factor Evaluation Process (MFEP). Multi Factor Evaluation Process yang di mana metode ini bekerja dengan memecah-mecah suatu situasi yang kompleks, tidak terstruktur, kedalam bagian-bagian komponennya (E. Iswandy. 2015).

Dalam rangka mengembangkan dan meningkatkan kualitas mutu dari pegawai sipil yang berada di lingkungan kepolisian daerah sumatera utara, dan untuk meningkatkan kemampuan dan pengetahuan pegawai sipil, serta mengembangkan potensi yang dimiliki oleh calon pegawai sipil tersebut. Kualitas pegawai yang tidak seperti diharapkan muncul setelah pegawai sipil lulus dari proses seleksi dan ini tentu sangat merugikan bagi instansi terkait dan lamanya proses seleksi yang dilakukan tentu akan menjadi beban instansi terkait dengan harus mengorbankan waktu dan anggaran yang bertambah untuk proses yang masih dilakukan secara manual.

Maka dari itu dibutuhkanlah calon pegawai sipil dengan permasalahan yang sering dihadapi masyarakat sekitar. Yakni dengan memberikan seleksi kepada calon pegawai sipil berdasarkan kriteria permasalahan yang ada di lingkungan sekitar berdasarkan keterampilan atau keahlian yang dibutuhkan khususnya pada lingkungan kepolisian daerah Sumatera Utara, dimana kriteria tersebut memiliki bobot terpenting dalam proses seleksi calon pegawai sipil, dan disusul oleh bakat, kreativitas dan prestasi yang lebih menonjol dengan calon pegawai lainnya.

Dengan permasalahan yang telah ada di atas, maka penulis akan mengangkat tema ini sebagai tugas akhir penulis dengan judul “ **SISTEM PENDUKUNG KEPUTUSAN SELEKSI CALON PEGAWAI SIPIL BARU DI POLDA SUMATERA UTARA MENGGUNAKAN METODE MULTIFACTOR EVALUATION PROCESS (MFEP)**”.

1.2 Perumusan Masalah

Berdasarkan hal di atas, maka dalam penelitian ini masalah yang diteliti dirumuskan dalam bentuk perumusan masalah yang meliputi :

1. Bagaimana sistem pendukung keputusan yang dibangun dapat menghasilkan kualitas pegawai sipil seperti yang diharapkan ?
2. Bagaimana sistem pendukung keputusan yang dibangun dapat melakukan efisiensi waktu dalam perekrutan ?
3. Bagaimana sistem pendukung keputusan yang dibangun dapat menghasilkan calon pegawai sipil yang memiliki keterampilan sesuai kebutuhan Polda Sumut ?

1.3 Batasan Masalah

Mengingat adanya keterbatasan kemampuan, waktu, dan untuk menghindari meluasnya permasalahan didalam penyusunan atau penulisan laporan penelitian ini maka penulis menitik beratkan pada permasalahan yang bersifat praktis, untuk itu penulis memberikan batasan pembahasan yaitu bagaimana Sistem Pendukung Keputusan yang dibangun dengan bahasa pemrograman PHP dapat berjalan dengan baik di Polda Sumut.

1.4 Tujuan Penelitian

Dalam penelitian ini terdapat beberapa tujuan yang ingin dicapai, adapun diantaranya adalah :

1. Menganalisa Sistem Pendukung Keputusan untuk dapat membantu penetapan pegawai sipil yang lebih berkualitas.
2. Untuk merancang Sistem Pendukung Keputusan agar bisa dimanfaatkan oleh pihak Polda Sumatera Utara.
3. Memudahkan penggunaan Aplikasi Sistem Pengambil Keputusan.

1.5 Manfaat Penelitian

Berikut merupakan manfaat dari penelitian yang ingin dicapai :

1. Membantu Polda Sumut untuk dapat menghasilkan pegawai sipil yang berkualitas seperti yang diharapkan.
2. Membantu Polda Sumut untuk dapat melakukan efisiensi waktu dan anggota yang terlibat dalam proses penseleksian.
3. Membantu Polda Sumut untuk dapat menghasilkan calon pegawai sipil yang memiliki keterampilan sesuai kebutuhan Polda Sumut.

BAB II

LANDASAN TEORI

2.1 Rekayasa Perangkat Lunak

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. Perangkat lunak banyak dibuat dan pada akhir sering tidak digunakan karena tidak memenuhi kebutuhan pelanggan bahkan karena masalah non-teknis seperti keinginan pemakai perangkat lunak (*user*) untuk mengubah cara kerja dari manual ke otomatis, atau ketidakmampuan user menggunakan komputer (Rosa A.S dan Shalahuddin, 2013).

Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut (Rosa A.S dan Shalahuddin, 2013) :

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

Dari kriteria diatas maka perangkat lunak yang baik adalah perangkat lunak yang dapat memenuhi kebutuhan pelanggan (*customer*) atau *user* (pemakai perangkat lunak) atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak (Rosa A.S dan Shalahuddin, 2013).

Fase pendukung (*support phase*) fokus pada perubahan yang terasosiasi pada perbaikan kesalahan (*error*), adaptasi yang dibutuhkan pada lingkungan perangkat lunak yang terlibat, dan perbaikan yang terjadi akibat perubahan kebutuhan pelanggan (*customer*).

Fase pendukung terdiri dari empat tipe perubahan antara lain (Rosa A.S dan Shalahuddin, 2013) :

1. Koreksi (*correction*)

Walaupun dengan jaminan kualitas yang terbaik, akan selalu ada kecacatan atau keinginan pelanggan (*customer*) yang tidak tertangani oleh perangkat lunak.

2. Adaptasi (*adaptation*)

Pada saat tertentu lingkungan asli (seperti CPU, sistem operasi, aturan bisnis, karakteristik produk luar) dimana perangkat lunak dikembangkan akan mengalami perubahan.

3. Perbaikan (*enhancement*)

Sejalan dengan dijalankannya perangkat lunak, maka pelanggan (*customer*) atau pemakainya (*user*) akan mengenali fungsi tambahan yang dapat mendatangkan manfaat.

4. Pencegahan (*prevention*)

Keadaan perangkat lunak komputer sangat dimungkinkan untuk perubahan. Oleh karena itu, pemeliharaan pencegahan (*preventive*) atau sering juga disebut juga dengan rekayasa ulang sistem.

Tantangan yang dihadapi dari proses rekayasa perangkat lunak adalah 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 dengan skala besar dan kompleks sekalipun dapat sampai ke tangan pelanggan (*customer*) atau *user* dengan cepat dan kualitas tetap terjaga.

2.1.1 Perangkat Lunak

Perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*). Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat disebut perangkat lunak (*software*).

Sistem perangkat lunak berarti sebuah sistem yang memiliki komponen berupa perangkat lunak yang memiliki hubungan satu sama lain untuk memenuhi kebutuhan pelanggan. (Rosa A.S dan Shalahuddin, 2013).

Rekayasa Perangkat Lunak (*Software engineering*) merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomis yang dipercaya dan bekerja secara efisien menggunakan mesin. (Rosa A.S dan Shalahuddin, 2013).

Penelitian rekayasa merupakan penelitian yang menerapkan ilmu pengetahuan menjadi suatu rancangan, guna mendapatkan suatu kinerja sesuai dengan persyaratan yang ditentukan. Penelitian diarahkan untuk membuktikan bahwa

rancangan tersebut memenuhi spesifikasi yang ditentukan. Penelitian berawal dari menentukan spesifikasi rancangan yang memenuhi spesifikasi yang ditentukan, memilih alternatif yang terbaik, dan membuktikan bahwa rancangan yang dipilih dapat memenuhi persyaratan yang ditentukan secara efisien, efektif dan dengan biaya yang murah.

Karakter perangkat lunak adalah sebagai berikut (Rosa A.S dan Shalahuddin, 2013) :

1. Perangkat lunak dibangun dengan rekayasa (*software engineering*) bukan diproduksi secara manufaktur atau pabrikan.
2. Perangkat lunak tidak pernah usang (*wear out*) karena kecacatan dalam perangkat lunak dapat diperbaiki.
3. Barang produksi pabrikan biasanya komponen barunya akan terus diproduksi, sedangkan perangkat lunak biasanya terus diperbaiki seiring bertambahnya kebutuhan.

Aplikasi dari perangkat lunak adalah sebagai berikut (Rosa A.S dan Shalahuddin, 2013) :

1. Perangkat lunak sistem (*system software*).
Merupakan kumpulan program dalam hal ini program yang satu ditulis untuk memenuhi kebutuhan program lainnya.
2. Perangkat lunak waktu nyata (*real-time software*).
Merupakan perangkat lunak yang memonitor, menganalisis, mengontrol sesuatu secara waktu nyata (*real-time*).
3. Perangkat lunak bisnis (*business software*).
Merupakan perangkat lunak pengelola informasi bisnis seperti akuntansi, penjualan pembayaran dan penyimpanan.

4. Perangkat lunak untuk keperluan rekayasa dan keilmuan (*engineering and scientific software*).

Merupakan perangkat lunak yang mengimplementasikan algoritma yang terkait dengan keilmuan ataupun perangkat lunak yang membantu keilmuan.

5. Perangkat lunak tambahan untuk membantu mengerjakan suatu fungsi dari perangkat lunak yang lainnya (*embedded software*).

Misalnya perangkat lunak untuk mencetak dokumen ditambahkan agar perangkat lunak yang memerlukan agar dapat mencetak laporan.

6. Perangkat lunak komputer personal (*personal komputer software*).

Merupakan perangkat lunak untuk PC misalnya perangkat lunak untuk pemroses teks, pemroses grafik.

7. Perangkat lunak berbasis web (*web based software*).

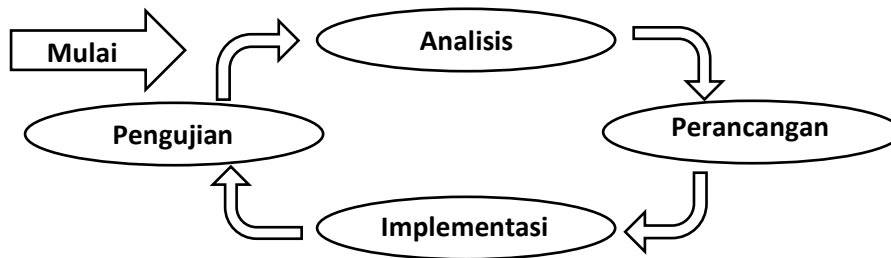
Merupakan perangkat lunak yang dapat diakses dengan mengiakan browser.

8. Perangkat lunak berinteligensi buatan (*artificial intelligence software*).

Merupakan perangkat lunak yang menggunakan algoritma tertentu untuk mengelola data sehingga seakan-akan memiliki inteligensi seiring bertambahnya data yang diproses.

2.1.2 Proses Rekayasa Perangkat Lunak

Proses perangkat lunak dilakukan selama pembangunan perangkat lunak. Proses-proses yang dilakukan dalam rekayasa perangkat lunak secara garis besar adalah sebagai berikut :



Sumber : Rosa A.S dan Shalahuddin, 2013

Gambar 2.1 Tahapan umum Rekayasa Perangkat Lunak

Proses-proses pada gambar diatas dapat dilakukan berulang kali sampai perangkat lunak memenuhi kebutuhan pelanggan atau *user*. Untuk membangun perangkat lunak yang benar-benar baik maka diperlukan tahapan-tahapan rekayasa perangkat lunak. Perangkat lunak yang dibangun tanpa melalui tahapan rekayasa perangkat lunak yang baik maka akan membuat baju tanpa pola dan rencana model baju yang baik. Jika baju dibuat tanpa pola maka yang terjadi adalah baju hasil tambal sulam. (Rosa A.S dan Shalahuddin, 2013).

Proses perangkat lunak (*software process*) adalah sekumpulan aktifitas yang memiliki tujuan untuk mengembangkan atau mengubah perangkat lunak. Secara umum proses perangkat lunak terdiri dari (Rosa A.S dan Shalahuddin, 2013) :

1. Pengumpulan Spesifikasi (*Specification*)

Mengetahui apa saja yang harus dapat dikerjakan sistem perangkat lunak dan batasan pengembangan perangkat lunak.

2. Pengembangan (*Development*)

Pengembangan perangkat lunak untuk menghasilkan sistem perangkat lunak.

3. Validasi (*Validation*)

Memeriksa apakah perangkat lunak sudah memenuhi keinginan pelanggan (*Customer*).

4. Evolusi (*Evolution*)

Mengubah perangkat lunak untuk memenuhi perubahan kebutuhan pelanggan (*customer*).

2.2 *Unified Modelling Language (UML)*

Unified Modelling Language (UML) muncul karena adanya kebutuhan pemodelan untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML hanya berpungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metode tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Rosa A.S dan Shalahuddin, 2013).

2.2.1 *Sejarah Unified Modelling Language (UML)*

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembang bahasa pemrograman berorientasi objek selanjutnya.

Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian diikuti dengan perkembangan bahasa pemrograman berorientasi objek

yang lainnya seperti C objek, C++, Eiffel, dan CLOS. Metode yang pertama diperkenalkan oleh Sally Shlaer dan Stephen Mellor (Shlaer-Mellor, 1988) dan Peter Coad dan Edward Yourdon (Coad-Yourdon, 1991), diikuti oleh Grady Booch (Booch, 1991), James R. Rumbaugh, Michael R. Blaha, William Lorenzen, Frederik Eddy, William Premerlani (Rumbaugh-Blaha-Premerlani-Eddy-Lorenzen, 1991).




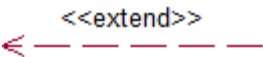
Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Usaha penyatuan ini banyak mengambil dari metodologi-metodologi yang berkembang saat itu. Maka dibuatlah bahasa yang merupakan gabungan dari beberapa konsep seperti konsep *ObjectModelling Technique* (OMT) dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities, Collaborators* (CRC) dari Rebecca Wirfs-Brock (1990), Konsep pemikiran Ivar Jacobson, dan beberapa konsep lainnya dimana James R. Rumbaugh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama *Rational Software Corporation* menghasilkan bahasa yang disebut dengan *Unified Modeling Language* (UML) (Rosa A.S dan Shalahuddin, 2013).

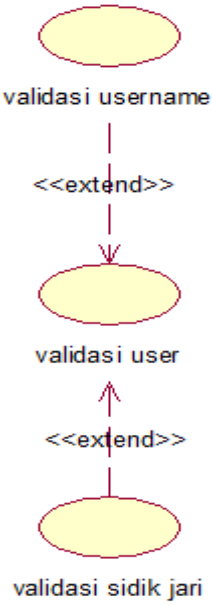

2.2.2 Use Case Diagram

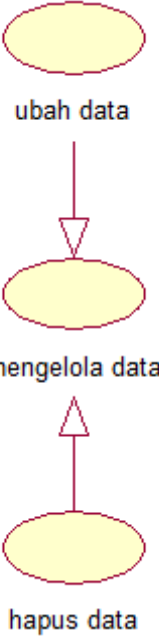
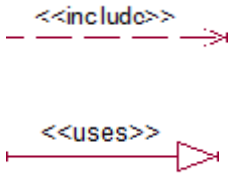
Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa A.S dan Shalahuddin, 2013).

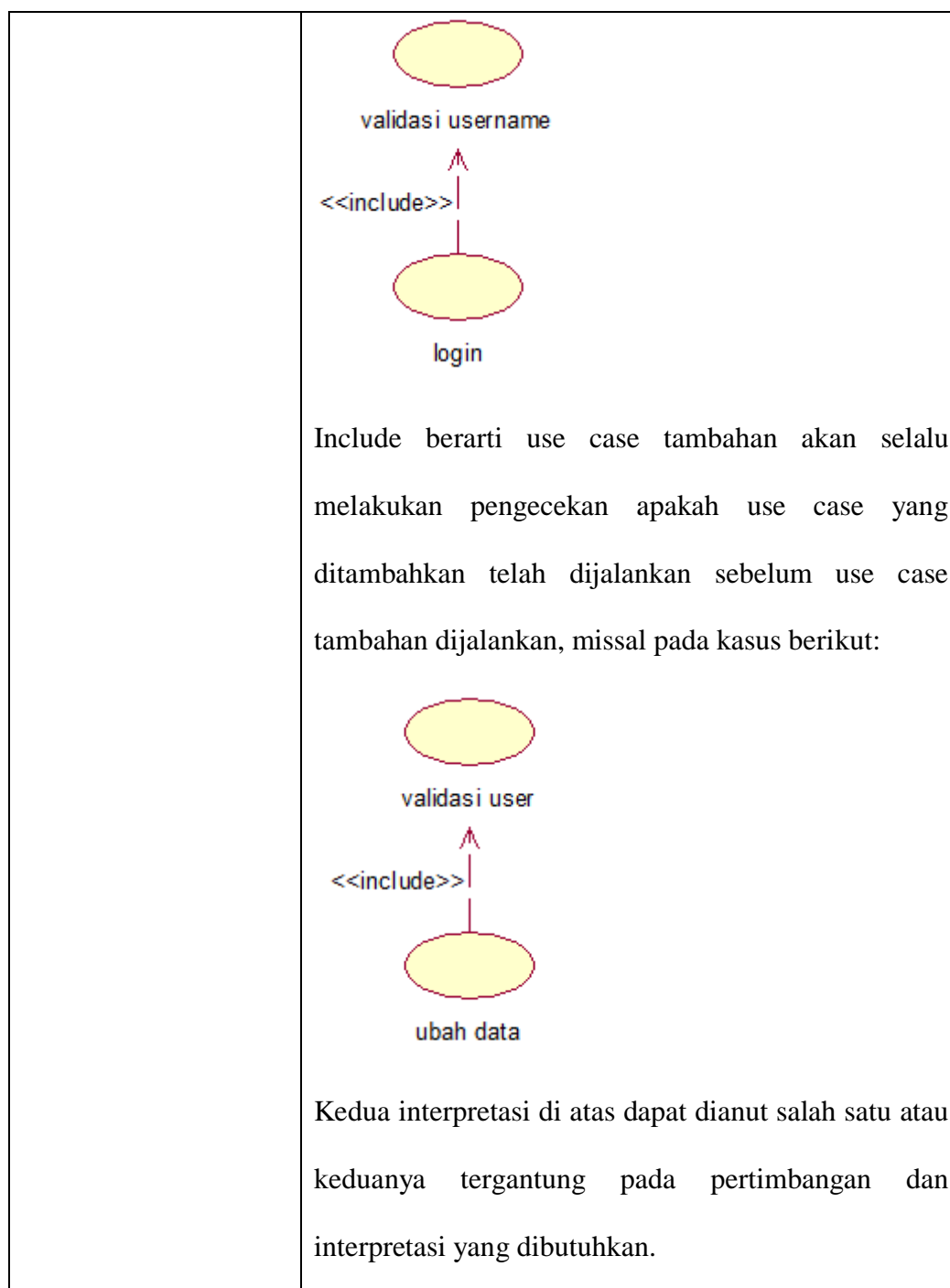
Berikut adalah simbol-simbol yang ada pada *use case* diagram itu (Rosa A. S dan M. Salahuddin, 2013) :

Tabel 2.1 Simbol – Simbol Use Case Diagram

Simbol	Deskripsi
Use case  nama use case	Fungsionalitas yang disediakan system sebagai unit-unit yang saling bertukar pesan antar unit atau actor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case
Aktor  nama aktor	Orang, proses, atau system lain yang berinteraksi dengan system informasi yang akan dibuat di luar system informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
Asosiasi / association 	Komunikasi antar actor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor.
Ekstensi / extend 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misalnya :

	 <pre> graph TD A([validasi username]) -.-> <<extend>> B([validasi user]) B -.-> <<extend>> C([validasi sidik jari]) </pre> <p>Arah panah mengarah pada use case yang ditambahkan; biasanya use case yang menjadi extendnya merupakan jenis yang sama dengan use case yang menjadi induknya.</p>
<p>Generalisasi / generalization</p> 	<p>Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah use case yang satu adalah fungsi yang lebih umum dari lainnya, misalnya :</p>

	 <p>Arah panah mengarah pada use case yang menjadi generalisasinya (umum).</p>
<p>Menggunakan / include / uses</p> 	<p>Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.</p> <p>Ada dua sudut pandang yang cukup besar mengenai include di use case :</p> <p>Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan, missal pada kasus berikut :</p>



Sumber : (Rosa A. S dan M. Salahuddin, 2013)

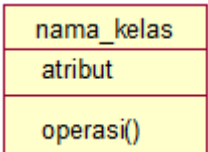
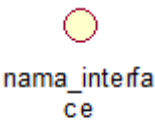


2.2.3 Class Diagram



Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Diagram kelas dibuat agar pembuat program atau

programer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron (Rosa A.S dan Shalahuddin, 2013).

Berikut adalah simbol-simbol yang ada pada *class diagram* (Rosa A. S dan M. Salahuddin, 2013) :

Tabel 2.2 Simbol-simbol Class Diagram

Simbol	Deskripsi
	Kelas pada struktur system
	Sama dengan konsep interface dalam pemrograman beroerientasi objek
	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity

Kebergantungan / Dependency 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi / aggregation 	Relasi antar kelas dengan makna semua bagian (whole-part)


Sumber : (Rosa A. S dan M. Salahuddin, 2013)

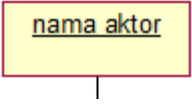

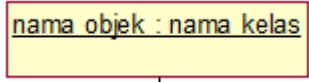

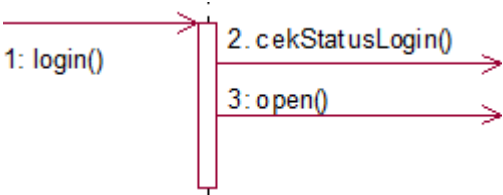

2.2.4 Sequence Diagram


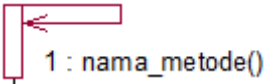

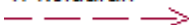
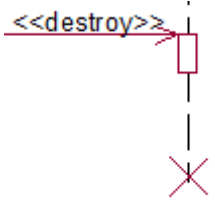
Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirim dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu (Rosa A.S dan Shalahuddin, 2013).

Berikut ini adalah simbol-simbol yang ada pada *sequence diagram* (Rosa A. S dan M. Salahuddin, 2013) :

Tabel 2.3 Simbol-simbol sequence diagram

Simbol	Deskripsi
Actor  nama aktor Atau	Orang, proses, atau system lain yang berinteraksi dengan system informasi yang akan dibuat di luar system informasi yang akan dibuat sendiri, jadi walaupun symbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal

<p><u>nama aktor</u></p> 	<p>frase nama actor.</p>
<p>Garis hidup / lifeline</p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> <p><u>nama objek : nama kelas</u></p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif yang berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p>  <p>Maka cek Status Login () dan open () dilakukan di dalam metode login ()</p> <p>Actor tidak memiliki waktu aktif</p>
<p>Pesan tipe create</p> <p><code><<create>></code></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p>	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau</p>

<p>1 : nama_metode</p> 	<p>dirinya sendiri,</p>  <p>Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe send</p> <p>1: masukan</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah yang mengarah pada objek yang dikirim</p>
<p>Pesan tipe return</p> <p>1: keluaran</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan type destroy</p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy</p>

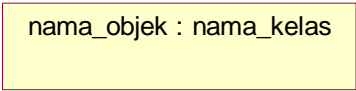

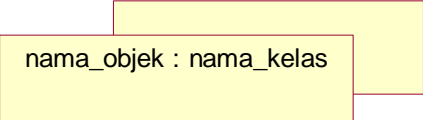


Sumber : (Rosa A. S dan M. Salahuddin, 2013)

2.2.5 *Communication Diagram*

Communication diagram adalah diagram kolaborasi tetapi dibuat untuk setiap sekuen. Dan menggambarkan interaksi antar objek/bagian dalam bentuk urutan pengiriman pesan (Rosa A.S dan Shalahuddin, 2013).

Berikut ini adalah simbol-simbol yang ada pada *Communication diagram* (Rosa A. S dan M. Salahuddin, 2013) :

Tabel 2.4 Simbol-simbol *communation diagram*

Symbol	Deskripsi
Objek 	objek yang melakukan interaksi pesan
Link 	relasi antar objek yang menghubungkan objek satu dengan lainnya atau dengan dirinya sendiri 
Arah Pesan 	arah pesan yang terjadi, jika pada suatu link ada dua arah pesan berbeda maka arah juga digambarkan dua rah pada dua sisi link 



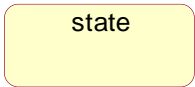
Sumber : (Rosa A. S dan M. Salahuddin, 2013)

2.2.6 Statechart Diagram

Statechart diagram digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek. Jika diagram *sequence* digunakan untuk interaksi antar objek maka *statechart* diagram digunakan untuk interaksi di dalam sebuah objek (Rosa A.S dan Shalahuddin, 2013).

Berikut ini adalah simbol yang digunakan dalam *statechart diagram* (Rosa A. S dan M. Salahuddin, 2013):

Tabel 2.5 Simbol-simbol statechart diagram

Simbol	Deskripsi
Start / status awal (initial state) 	<i>Start</i> atau Initial State adalah <i>state</i> atau keadaan awal pada saat sistem mulai hidup.
End / status Akhir (Final State) 	<i>End</i> atau <i>final state</i> adalah keadaan akhir dari daur hidup suatu sistem.
Event —Even —>	<i>Event</i> adalah kegiatan menyebabkan berubahnya status mesin.
State 	State atau status Keadaan sistem pada waktu tertentu. State dapat berubah jika pada event tertentu yang memicu perubahan tersebut.






Sumber : (Rosa A. S dan M. Salahuddin, 2013)

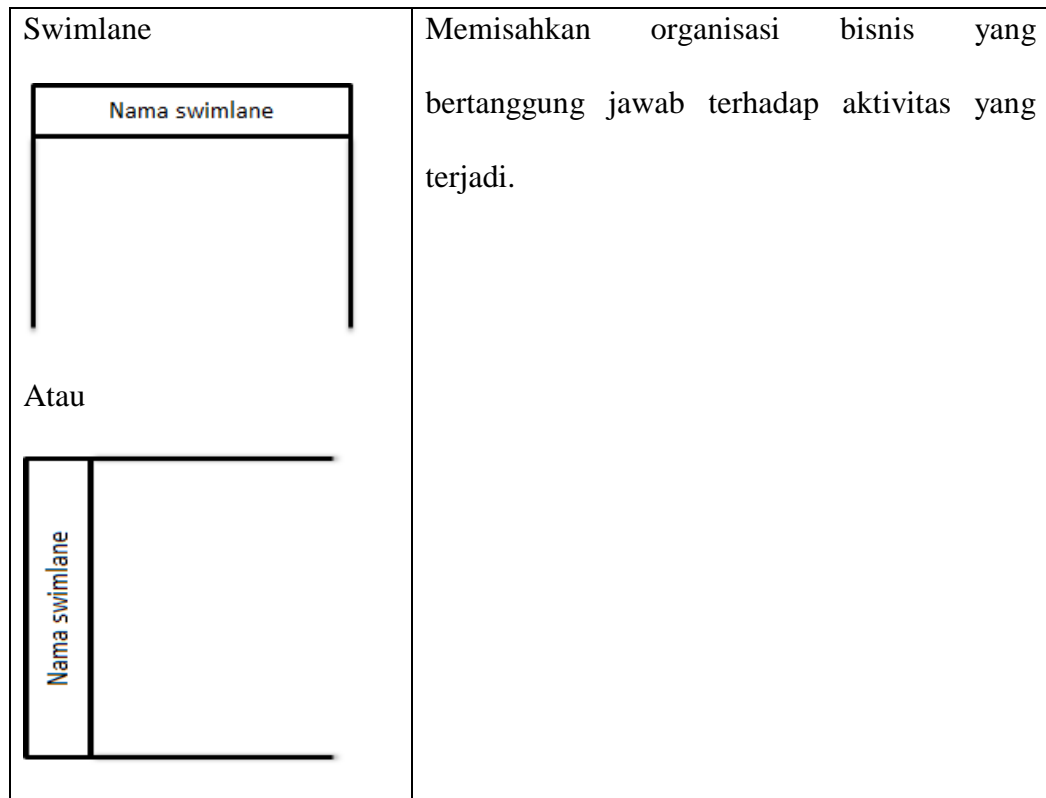
2.2.7 Activity Diagram

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 kreativitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa A.S dan Shalahuddin, 2013).

Berikut ini adalah simbol-simbol yang ada pada *activity diagram* (Rosa A. S dan M. Salahuddin, 2013):

Tabel 2.6 Simbol-simbol *activity diagram*

Simbol	Deskripsi
	Status awal aktivitas sistem, sebuah <i>activity</i> diagram memiliki sebuah status awal.
	Aktivitas yang dilakukan sistem aktivitas biasanya diawali dengan kata kerja.
	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.



Sumber : (Rosa A. S dan M. Salahuddin, 2013)

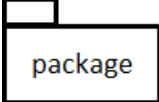
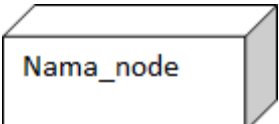


2.2.8 *Deployment Diagram*

Deployment/physical diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut (Rosa A.S dan Shalahuddin, 2013) :

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device, node, dan hardware*.
2. Sistem *client/server*.
3. Sistem terdistribusi murni.
4. Rekayasa ulang aplikasi.

Berikut ini adalah simbol yang digunakan dalam *deployment diagram* (Rosa A. S dan M. Salahuddin, 2013) :

Tabel 2.7 Simbol-simbol deployment diagram

Simbol	Deskripsi
Package 	Package Merupakan sebuah bungkus dari satu atau lebih <i>node</i>
Node 	Biasanya mengacu pada perangkat keras, perangkat lunak yang tidak dibuat sendiri, jika dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
Kebergantungan / dependency 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
Link 	Relasi antar <i>node</i>

Sumber : (Rosa A. S dan M. Salahuddin, 2013)

2.3 Sistem Pendukung Keputusan

2.3.1 Definisi Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Sistem pendukung keputusan juga merupakan sistem informasi berbasis

komputer untuk manajemen pengambilan keputusan yang menangani masalah-masalah semi struktur, (Ahmad Khaidir, 2014).

Keputusan adalah kegiatan memilih suatu strategi atau tindakan dalam masalah tersebut. Tindakan memilih strategi atau aksi yang diyakini manager akan memberikan solusi terbaik atas suatu itu disebut pengambilan keputusan. Tujuan dari pengambilan keputusan adalah untuk mencapai target atau aksi tertentu yang harus dilakukan.

Kriteria atau ciri-ciri keputusan yaitu, (Kusrini, 2007) :

1. Banyak pilihan atau alternatif,
2. Ada kendala atau syarat,
3. Mengikuti suatu model atau tingkah laku, baik yang terstruktur maupun tidak terstruktur,
4. Banyak input atau variabel,
5. Ada faktor resiko,
6. Dibutuhkan kecepatan, ketepatan dan keakuratan.

Sistem Pendukung Keputusan (SPK) adalah sistem informasi berbasis komputer yang memberikan kemudahan berupa fleksibilitas dalam menganalisa dan membantu manager untuk tetap fokus dimasa datang (Kusrini, 2007). SPK diterapkan untuk membantu *top manager* dan *midle manager* untuk membantu strategi keputusan terhadap permasalahan yang tidak terstruktur, termasuk kejadian dan kebiasaan yang ada di luar organisasi.

Tujuan SPK adalah sebagai *second opinion* atau *informatif sources*, sebagai bahan pertimbangan seorang manager sebelum memutuskan kebijakan tertentu (Kusrini, 2007). SPK lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur.

SPK memiliki masukan berupa ringkasan laporan, proses transaksi data dan data internal yang lainnya, masukan bisa berasal dari luar organisasi. Sedangkan untuk keluarannya bersifat fleksibel sesuai permintaan dari manager to manager sebagai pembuat keputusan.

SPK dapat meningkatkan efektifitas pengambilan keputusan, mengurangi kebutuhan akan pelatihan, peningkatan kontrol manajemen, memfasilitasi komunikasi, mengurangi usaha yang harus dikerjakan pengguna, mengurangi biaya, dan memberikan banyak pilihan tujuan pengambilan keputusan.

2.3.2 Karakteristik Sistem Pendukung Keputusan

Oleh karena tidak ada konsensus mengenai apa sebenarnya DSS, jelas tidak ada kesepakatan mengenai karakteristik standar DSS. Berikut karakteristik yang diharapkan ada di DSS, (Kusrini, 2007) :

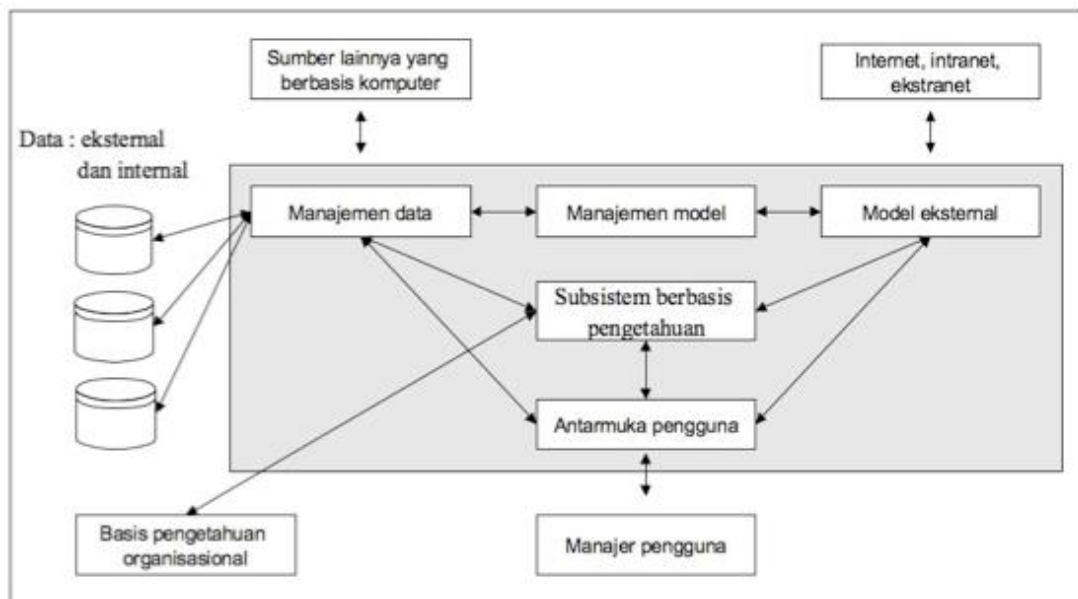
1. Dukungan kepada pengambil keputusan, terutama pada situasi semi terstruktur dan tak terstruktur, dengan menyertakan penilaian manusia dan informasi terkomputerisasi. Masalah-masalah tersebut tidak bisa dipecahkan oleh sistem komputer lain atau oleh metode atau alat kuantitatif standar.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer.
3. Dukungan untuk individu dan kelompok. Masalah yang kurang terstruktur sering memerlukan keterlibatan individu dari departemen dan tingkat organisasional yang berbeda atau bahkan dari organisasional yang lain.
4. Dukungan untuk keputusan independen atau sekunsial. Keputusan bisa dibuat satu kali, beberapa kali, atau berulang kali (dalam interval yang sama).

5. Dukungan di semua fase proses pengambilan keputusan : intelegensi, desain, pilihan , dan implementasi.
6. Dukungan diberbagai proses dan gaya pengambilan keputusan.
7. Adaptivitas sepanjang waktu. Pengambil keputusan seharusnya reaktif, bisa menghadapi perubahan kondisi secara cepat, dan mengadaptasi DSS untuk memenuhi perubahan tersebut.
8. Pengguna merasa seperti di rumah. Ramah pengguna, kapabilitas grafis yang sangat kuat, dan antar muka manusia-mesin yang interaktif dengan satu bahasa alami bisa sangat meningkatkan efektifitas DSS.
9. Peningkatan efektifitas pengambilan keputusan (akurasi, timelines, kualitas) ketimbang pada efisiensinya (biaya pengambilan keputusan).
10. Kontrol penuh oleh pengambil keputusan terhadap semua langkah proses pengambilan keputusan dalam memecahkan suatu masalah.
11. Pengguna akhir bisa mengembangkan dan memodifikasi sendiri sistem sederhana. Sistem yang lebih besar bisa dibangun dengan bantuan ahli sistem informasi.
12. Bisanya model-model digunakan untuk menganalisis situasi pengambilan keputusan. Kapabilitas pemodelan memungkinkan eksperimen dengan berbagai strategi yang berbeda dibawah konfigurasi yang berbeda.
13. Akses disediakan untuk berbagai sumber data, format, dan tipe mulai dari sistem informasi geografis (GIS) sampai sistem berorientasi objek.
14. Dapat digunakan sebagai alat standalone oleh seorang pengambil keputusan pada suatu lokasi atau didistribusikan disuatu organisasi.

2.3.3 Arsitektur Sistem Pendukung Keputusan

Aplikasi Sistem Pendukung Keputusan (SPK) mempunyai suatu sistem yang terdiri dari beberapa subsistem yang menyusunnya, diantaranya, (Kusrini, 2007) :

1. *Subsistem Manajemen Data*, subsistem manajemen data memasukkan satu database yang berisi data yang relevan untuk suatu situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen database (DBMS/ Database Manajemen Sistem).
2. *Subsistem Manajemen Model*, merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lain yang memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat.
3. *Subsistem Antar Pengguna*, pengguna berkomunikasi dengan dan memerintahkan sistem pendukung keputusan melalui subsistem tersebut. Pengguna adalah bagian yang dipertimbangkan dari sistem.
4. *Subsistem Manajemen Berbasis Pengetahuan*, subsistem tersebut mendukung semua subsistem lain atau bertindak langsung sebagai suatu komponen independen dan bersifat opsional.



Sumber : Kusrini, 2007

Gambar 2.2 : Arsitektur DSS

2.4 Multifactor Evaluation Process (MFEP)

Multifactor Evaluation Process (MFEP) merupakan suatu metode yang memecahkan suatu situasi yang kompleks tidak terstruktur, kedalam bagian-bagian komponennya. Dasar berpikirnya metode MFEP adalah pengambilan keputusan dilakukan dengan memberikan pertimbangan subyektif dan intuitif terhadap faktor yang dianggap penting. Pertimbangan-pertimbangan tersebut berupa pemberian bobot (*weighting system*) atas *multifactor* yang terlibat dan dianggap penting tersebut.

Metode MFEP sangat memperhitungkan tingkat kualitas dari suatu faktor. selain itu, MFEP mempunyai kemampuan untuk memecahkan masalah yang memiliki hasil yang terfokus atau dapat memecahkan situasi yang kompleks agar lebih terstruktur, dan multi kriteria yang didasarkan pada perbandingan faktor-faktor penentu dari setiap faktor-faktor yang telah ditentukan.

2.4.1 Kelebihan Multifactor Evaluation Process (MFEP)

Metode MFEP telah banyak penggunaannya dalam berbagai skala bidang kehidupan. Adapun keunggulan metode MFEP adalah sebagai berikut (Ahmad Khaidir, 2014) :

1. Faktor utama dapat ditentukan dengan melakukan pemberian bobot lebih tinggi dari pada faktor yang lainnya.
2. Dapat menentukan proses penseleksian berdasarkan kebutuhan yang ada pada instansi terkait.
3. Memperhitungkan daya tahan atau ketahanan output analisis sensitivitas pengambilan keputusan.
4. Metode MFEP memiliki keunggulan dari segi proses pengambilan keputusan dan akomodasi untuk atribut-atribut baik kuantitatif maupun kualitatif.
5. Metode MFEP juga mampu menghasilkan hasil lebih terfokus dibandingkan dengan metode-metode lainnya.
6. Metode pengambilan keputusan MFEP memiliki sistem yang mudah dipahami dan digunakan.

2.4.2 Kekurangan Multifactor Evaluation Process (MFEP)

Kekurangan metode MFEP diantaranya adalah (Ahmad Khaidir, 2014) :

1. Bila nilai dari bobot utama sudah rendah maka persentase gagal untuk suatu aktor akan sangat tinggi.
2. MFEP tidak cocok untuk diterapkan pada proses pengambilan keputusan bersifat umum atau kuantitatif.

2.4.3 Prosedur Perhitungan MFEP

Pada dasarnya terdapat beberapa tahapan ataupun prosedur yang harus dilakukan dalam proses perhitungan bobot dengan metode MFEP. Adapun tahap-tahap dalam proses perhitungan bobot antara lain (Muhammad Dahria dkk, 2014) :

1. Tentukan faktor-faktor pendukung keputusan

Tabel 2.8: Pemberian Nilai Bobot Untuk Faktor

Faktor	Nilai Bobot
Faktor 1	-
Faktor 2	-
Faktor 3	-
Total Bobot	1

2. Langkah selanjutnya adalah memberikan pembobotan pada faktor-faktor yang digunakan dimana total pembobotan harus sama dengan 1 $\sum \text{pembobotan} = 1$ tentukan bobot pada masing-masing faktor berdasarkan kualitas yang dibutuhkan, semakin besar bobot suatu faktor maka akan semakin besar pula persentase dari faktor tersebut.
3. Setelah dilakukan pembobotan calon pegawai yang ada akan ditimbang selanjutnya calon pegawai yang ada akan dievaluasi dan diberikan nilai bobot untuk setiap kriterianya seperti tercantum dalam tabel di bawah ini.

Tabel 2.9: Evaluasi Faktor

Faktor	Calon 1	Calon 2	Calon 3	Calon 4	Calon 5	Calon 6
Faktor 1
Faktor 2
Faktor 3

4. Dengan adanya informasi diatas, didapat jumlah total nilai evaluasi untuk setiap alternatif atau peserta seleksi. Setiap peserta seleksi memiliki sebuah nilai evaluasi bagi faktor-faktor yang menjadi bahan pertimbangannya, untuk mendapatkan nilai total evaluasi setiap pelamar dengan cara perhitungan sebagai berikut :

Perhitungan nilai bobot evaluasi:

$$\mathbf{Nbe = Nbf \times Nef}$$

Keterangan:

Nbe → Nilai bobot evaluasi

Nbf → Nilai bobot faktor

Nef → Nilai evaluasi faktor

Perhitungan total nilai evaluasi (Tne):

$$Tne = Nbe1 + Nbe2 + Nbe3 + \dots Nben$$

Keterangan :

Tnb → Total nilai evaluasi

5. Setelah dilakukan pengenalan rumus maka proses perhitungan berdasarkan tiap-tiap alternatif yang ada :

Faktor	Bobot Faktor		Evaluasi Faktor	Bobot Evaluasi
Faktor 1	...	x
Faktor 2	...	x
Faktor 3	...	x
Total				...

6. Setelah dilakukan perhitungan untuk semua calon kriteria yang ada maka didapatkanlah nilai total nilai evaluasi dari masing-masing kriteria/calon anggota yang ada.
7. Setelah didapatkan nilai total nilai evaluation (Tne) maka hal yang harus dilakukan adalah melakukan proses sorting berdasarkan nilai terkecil hingga terbesar hingga di dapat batas angka maksimal yang mencapai target.

2.5 Bahasa Pemograman

Yang dimaksud dengan program adalah kumpulan instruksi yang digunakan untuk mengatur komputer agar melakukan suatu tindakan tertentu. Dengan kata lain, program merupakan salah satu bagian penting pada komputer, yang mengatur komputer agar melakukan tindakan yang sesuai dengan yang dikehendaki oleh pembuatnya.

Suatu program dapat ditulis dengan mengikuti kaidah bahasa pemrograman tertentu. Bahasa pemrograman dapat dianalogikan dengan bahasa yang digunakan manusia (bahasa manusia). Sebagaimana diketahui, ada bermacam-macam bahasa manusia, seperti bahasa Inggris, bahasa Indonesia, bahasa Batak. Kumpulan instruksi dalam bahasa manusia yang berupa sejumlah kalimat dapat anda analogikan dengan suatu program. Manusia dapat mengerjakan suatu instruksi berdasarkan kalimat-kalimat dan komputer bisa menjalankan suatu instruksi menurut program (Abdul Kadir, 2012).

2.5.1 Konsep Dasar HTML

Hypertext Markup Language (HTML) adalah sebuah bahasa dasar yang digunakan untuk merancang halaman web statis, html hanyalah sebuah gambaran tekstual sederhana dari sebuah isi dengan suatu maksud, (Sri Winiarti, 2013).

Seperti contoh, bentuk kode dari judul kepala "Tujuan dari HTML" di atas adalah:

```
<h2 id="htmllooks">Bagaimana Bentuk HTML</h2>
```

Bagian "<h2>" adalah sebuah marker (biasanya dinyatakan sebagai "tag") yang berarti "apa yang harus dipertimbangkan pada berikutnya di judul kepala tingkat kedua". "</h2>" adalah tag yang menandakan dimana judul kepala tingkat kedua berakhir (biasanya dinyatakan sebagai "penutup tag"). Pembuka tag, penutup tag dan segala sesuatu yang berada di antaranya disebut "elemen". Kebanyakan orang dalam penggunaan istilah elemen dan tag saling bergantian, namun demikian penggunaan tersebut kurang tepat. Id = "htmllooks" disebut sebagai atribut.

2.5.2 Sejarah HTML

Hypertext Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman web. HTML merupakan pengembangan dari standar pemformatan dokumen teks yaitu *Standard Generalized Markup Language* (SGML). HTML sebenarnya adalah dokumen ASCII atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu.

HTML dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN dan dipopulerkan pertama kali oleh browser Mosaic. Selama awal tahun 1990 HTML mengalami perkembangan yang sangat pesat. Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya. Namun perkembangan resmi dikeluarkan pada bulan November 1995 oleh IETF (Internet Engineering Task Force). HTML 2.0 ini merupakan penyempurnaan dari HTML+ (1993). HTML 3.0 (1995) memberikan kemampuan lebih daripada versi sebelumnya. Sebuah usaha dari World Wide Web Consortium's (W3C) HTML

Working Group pada tahun 1996 menghasilkan HTML 3.2. HTML versi ini secara resmi diterbitkan pada bulan Januari 1997. HTML versi terbaru adalah HTML 4.01 yang dikeluarkan secara resmi oleh W3C pada tanggal 24 April 1998. HTML merupakan perbaikan dari HTML 4.0 yang lebih dulu diterbitkan. (18 Desember 1997).

Pada tahun 1995, "HTML 2.0" telah dituliskan, dengan mengambil ide dari naskah HTML yang asli. Sebuah proposal pengganti dinamakan HTML+ juga dituliskan oleh Dave Raggett, dimana digunakan sebagai sebuah dasar untuk elemen-elemen baru yang diimplementasikan pada browser (seperti metode untuk memasukkan gambar ke dalam dokumen dipelopori oleh NCSA Mosaic).

Naskah HTML 3.0 kemudian muncul selanjut pada tahun tersebut, namun pengerjaan atas versi ini dihentikan karena kurangnya bantuan dalam pengarahan dari pembuat browser. HTML 3.2 meninggalkan beberapa fitur baru yang ada pada versi 3.0, namun mengadopsi kreasi-kreasi baru dari browser terkenal pada saat itu, Mosaic dan Netscape Navigator.

Pada tahun 1997, W3C publikasikan HTML 4.0 sebagai sebuah rekomendasi yang mengadopsi ekstensi yang lebih berspesifik browser, namun juga telah berusaha dalam melakukan rasionalisasi dan pembersihan pada HTML. Dalam versi ini dilakukan dengan menandakan beberapa elemen sebagai elemen "deprecated", yang berarti elemen-elemen yang tidak terpakai. Meskipun elemen-elemen deprecated masih tetap berada dalam versi ini, pada versi berikutnya mereka akan dihapus. Versi ini juga merupakan sebuah pengantoran untuk lebih baik dan semantik dalam penggunaan HTML pada dokumen (dijelaskan lebih detil pada artikel Model standar web).

HTML 4.01 dipublikasikan pada tahun 1999 dengan beberapa kesalahan tulis yang kemudian dijilidkan pada tahun 2001. Versi ini merupakan versi terakhir dari HTML, meskipun HTML 5 yang saat ini sedang disusun. Pada tahun 2000, W3C publikasikan lagi spesifikasi XHTML 1.0 yang merupakan sebuah HTML yang distrukturkan ulang dan dijadikan dokumen XML yang valid.

2.5.3 Struktur Dokumen HTML

Dokumen pertama kali dimulai dengan sebuah elemen tipe dokumen, atau *doctype*. *Doctype* mendeskripsikan tipe apa dari HTML yang sedang digunakan, sehingga *user agent* dapat menentukan bagaimana dokumen diterjemahkan.

```
1 <html>
2   <head>
3     <title>Example page</title>
4   </head>
5   <body>
6     <h1>Hello world</h1>
7   </body>
8 </html>
9
```

Gambar 2.3 Contoh Source Code HTML

Setelah itu bisa dilihat tag pembuka dari elemen HTML dari contoh di atas. Ini merupakan [wrapper] pada seluruh dokumen. Tag penutup html merupakan hal yang paling akhir pada setiap dokumen HTML. Diawali dengan <namatag> dan diakhiri dengan </namatag> (terdapat tanda "/").

Dalam elemen HTML terdapat elemen head. Elemen ini merupakan sebuah wrapper untuk menampung informasi tentang dokumen (metadata). Di dalam elemen head terdapat elemen title yang menentukan judul "Example page" pada menu bar. Kalimat yang terletak di dalam tag ini akan muncul pada bagian paling atas browser Anda (pada title bar).

Setelah elemen head, selanjutnya adalah elemen body. Elemen ini merupakan wrapper yang berisi konten yang sebenarnya dari halaman. Pada contoh di atas, hanya terdapat elemen header tingkat pertama (h1) yang berisi kata "Hello world.". Di dalam tag ini bisa diletakkan berbagai *page attribute* seperti warna latar belakang, warna teks, warna link, warna visited link, warna active link dan lain-lain.

Dengan demikian dapat dikatakan bahwa informasi yang berupa kode-kode akan diapit oleh tag awal dan tag akhir, dan sebuah apitan tag bisa juga di apit oleh tag lainnya. Dalam HTML ada beberapa tag yang tidak perlu diakhiri atau ditutup, misal tag
. Selain itu ada beberapa tag yang memiliki atribut-atribut untuk pengaturan teks maupun halaman, misalnya tag anchor yang sering dituliskan dengan lambang <a> dan memiliki atribut href, rel name dan sebagainya (Diar Puji Oktavian, 2013).

2.5.4 Hypertext Processor (PHP)

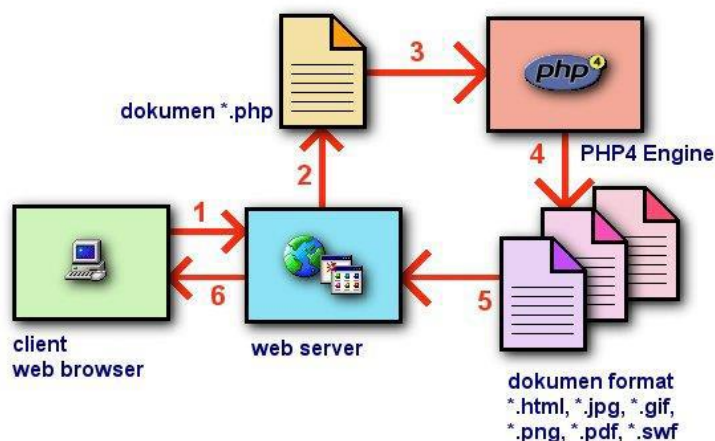
PHP (Hypertext Preprocessor) merupakan bahasa pemrograman scripting web server-side. Dengan pemrograman server-side, sebuah website akan lebih dinamis. PHP biasanya terpasang pada file HTML yang disimpan dengan eksistensi *.php, (Sri Winiarti, 2013).

PHP (*Hypertext Preprocessor*) adalah kode/skrip yang akan dieksekusi pada server-side. PHP akan membuat suatu aplikasi dapat diintegrasikan kedalam HTML, sehingga suatu halaman web tidak lagi bersifat dinamis. Sifat server-side berarti pengerjaan skrip di lakukan di server, baru kemudian hasilnya di kirim ke browser (Riyanto, 2014).

Salah satu kelebihan dari PHP adalah mampu berkomunikasi dengan berbagai *database* yang terkenal. Dengan demikian, menampilkan data yang bersifat dinamis, yang diambil dari *database*, merupakan hal yang mudah untuk diimplementasikan.

2.5.5 Konsep Dasar PHP

Salah satu server side scripting (berjalan pada sisi server) yang populer saat ini adalah PHP. Selain dapat digunakan untuk berbagai sistem operasi, koneksi *database* yang sangat mudah, dan gratis menyebabkan PHP banyak digunakan.



Sumber : php.net

Gambar 2.4. Ilustrasi Proses Kerja PHP

Adapun beberapa PHP mempunyai ciri-ciri khusus, yaitu (Diar Puji Oktavian, 2013) :

1. Hanya dapat dijalankan menggunakan web server, misal : Apache.
2. Kode PHP diletakkan dan dijalankan di web server.
3. Kode PHP dapat digunakan untuk mengakses database, seperti : MySQL, Postgre SQL, Oracle, dan lain-lain.
4. Merupakan software yang bersifat open source.
5. Gratis untuk didownload dan digunakan

6. Memiliki sifat multiplatform, artinya dapat dijalankan menggunakan sistem operasi apapun, seperti : Linux, Unix, Windows, dan lain-lain.

Sedangkan penulisan tag php ada empat macam, yaitu:

1. Standar Style

Format penulisannya : `<?php ?>`

2. Short Style

Format penulisannya : `<? ?>`

3. Javascript Style

Format penulisannya : `<script language="php"> </script>`

4. ASP Style

Format penulisannya : `<% %>`

Dan gambar dibawah ini adalah contoh sederhana dari pemrograman PHP yang akan digunakan.

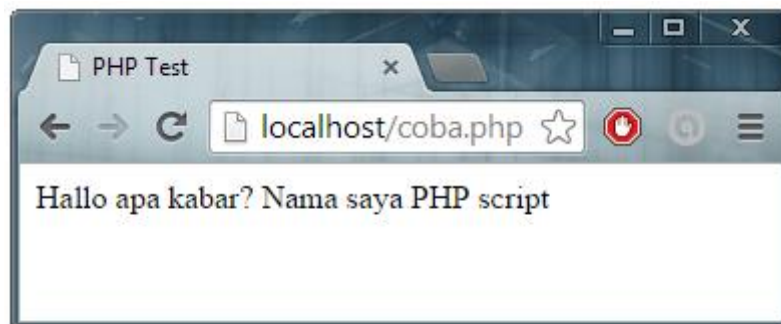
```

1  <html>
2  <head>
3      <title>PHP Test</title>
4  </head>
5  <body>
6      <?php
7          echo("Hallo apa kabar? Nama saya PHP script");
8      ?>
9  </body>
10 </html>
11

```

Gambar 2.5. Contoh Source Code PHP

Apabila dijalankan pada webserver maka akan menghasilkan tampilan sebagai berikut :



Gambar 2.6. Contoh Hasil dari *Source Code* PHP

2.6 *Database Management System (DBMS)*

Basis data adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Dengan basis data, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basis data menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Datapun harus mudah ditambahkan kedalam basis data, dimodifikasi dan dihapus (Janner Simarmata dan Iman Paryudi, 2005).

Basis data merupakan himpunan kelompok data yang saling berkaitan atau merupakan kumpulan data yang dapat digambarkan sebagai aktifitas dari suatu atau lebih organisasi yang berelasi.

DBMS merupakan suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola dan menampilkan data. DBMS juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, keakuratan dan konsistensi data (Rosa A.S M. Shalahuddin, 2013).

DBMS (*Database Management System*) merupakan perantara bagi pemakai dengan basis data dalam media penyimpanan. DBMS berisi kumpulan data yang saling berelasi dengan set program untuk mengakses data tersebut. Suatu sistem

aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut (Rosa A.S dan Shalahuddin, 2013) :

- a. Menyediakan fasilitas untuk mengelola akses data.
- b. Mampu menangani integritas data.
- c. Mampu menangani akses data yang dilakukan
- d. Mampu menangani *backup* data.

2.6.1 Definisi MySQL

MySQL adalah DBMS yang didistribusikan secara gratis dibawah lisensi dari *General Public License (GPL)*, dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh untuk dijadikan program induk turunan bersifat *close source* (komersial).

MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus, (Rosa A.S dan Shalahuddin, 2013).

2.6.2 Kelebihan dari MySQL

Program database yang digunakan dalam pembahasan penelitian ini adalah database MySQL, dengan beberapa pertimbangan (Diar Puji Oktavian, 2013) :

1. Bersifat open source dan gratis (Free).
2. Memiliki berbagai macam fitur dan terus dikembangkan.
3. Memiliki banyak komunitas.
4. Bahasa pemrograman web PHP sangat mendukung (support) database MySQL.

2.7 Teori Seleksi

Berbagai bentuk perubahan yang terjadi dengan cepat menjadi salah satu faktor pendorong departemen SDM untuk terus melakukan usaha-usaha pada praktik-praktik inovatif SDM secara kontinyu. Akan tetapi tekanan yang dihadapi departemen SDM untuk melakukan efektifitas biaya dalam memberikan kontribusi secara nyata pada departemen SDM bagi kelangsungan hidup instansi terkait masih menjadi polemik dan sulit diterapkan pada praktik-praktik sesungguhnya (Maricar dkk, 2016).

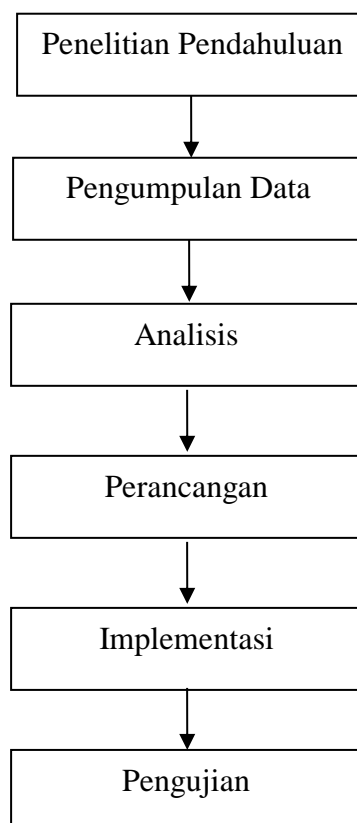
Salah satu sumber daya yang dimiliki organisasi untuk mendukung hal tersebut adalah karyawan (Sumber Daya Manusia/SDM). Oleh sebab itu karyawan tersebut harus dipekerjakan secara efektif, efisien dan manusiawi. Agar kondisi yang adaptif tersebut dapat tercipta, maka organisasi perlu membuat perencanaan Sumber Daya Manusia (SDM) sebaik-baiknya sehingga dapat memenuhi kebutuhan sumber daya manusia organisasi, baik jangka pendek, menengah maupun jangka panjang. Dengan demikian, organisasi tidak akan mengalami kesulitan dalam memenuhi Sumber Daya Manusia yang dibutuhkan (Nasution, A. B. 2018).

BAB III

METODOLOGI PENELITIAN

3.1. Kerangka Penelitian

Kerangka Penelitian adalah urutan kegiatan yang akan dilakukan dalam suatu penelitian. Agar langkah-langkah yang diambil penulis dalam perancangan ini tidak melenceng dari pokok pembahasan dan lebih mudah dipahami, maka urutan langkah-langkah penelitian 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 berikut :



Gambar 3.1 Kerangka Kerja Penelitian

3.2. Tahapan Penelitian

Tahapan penelitian ini menjelaskan langkah-langkah dalam melakukan pencatatan data serta mengumpulkan beberapa laporan yang diperlukan untuk dijadikan pedoman dalam pembuatan penelitian ini, yaitu :

3.2.1. Penelitian Pendahuluan

Penelitian pendahuluan ini merupakan langkah pertama dalam melakukan suatu penelitian dengan cara menganalisa terlebih dahulu masalah-masalah yang akan dikembangkan. Dengan tujuan setelah diterapkan aplikasi yang akan dikembangkan ini dapat membantu pengguna (*user*) pihak terkait yang dalam hal ini adalah pihak Polda Sumatera Utara dalam melakukan proses seleksi calon pegawai baru.

3.2.2. Pengumpulan Data

Merupakan urutan-urutan dalam melakukan penelitian mulai dari awal sampai akhir penelitian. Dalam melakukan penelitian untuk mendapatkan data informasi, maka metode yang digunakan dalam proses pengumpulan data dilakukan dengan tahap-tahap sebagai berikut :

3.2.2.1. Waktu Penelitian

Untuk dapat menyelesaikan penelitian ini dimulai pada waktu penulis telah menerima surat persetujuan atas judul yang telah diajukan kepada program studi untuk melakukan penelitian.

3.2.2.2. Tempat Penelitian

Adapun tempat penelitian dan pengambilan data dilakukan di Kantor SDM Polda Sumatera Utara sebagai sumber informasi kami dalam penelitian ini yang ber alamat di Jati Kota Medan Propinsi Sumatera Utara.

3.2.2.3. Metode Penelitian

Untuk menyelesaikan permasalahan yang ada, metode pengumpulan data yang saya lakukan sebagai penulis adalah dengan cara :

1. Penelitian Lapangan (*Field Research*)

a. Pengamatan (*Observasi*)

Dalam tahapan awal yang dilakukan oleh penulis merupakan *observasi* lapangan dengan cara melihat serta mempelajari hal-hal yang mempengaruhi proses seleksi calon pegawai baru dan mengumpulkan data-data yang bersangkutan dengan seleksi calon pegawai baru. Penelitian dilaksanakan di Kantor SDM Polda Sumatera Sumut.

b. Wawancara (*Interview*)

Wawancara ini dilakukan untuk mendapatkan data dan informasi yang berhubungan dengan seleksi calon pegawai Polda Sumatera Utara. Yaitu dengan mewawancari Kompol. Yessi Kurniati selaku Staf koordinasi SDM Polda Sumut serta hal-hal apa saja yang mempengaruhi calon pegawai untuk dapat dikatakan lulus atau ditetapkan sebagai anggota Polisi.

2. Penelitian Pustaka (*Library Research*)

Metode yang dilakukan adalah dengan cara mencari, mengumpulkan dan mempelajari data dari buku-buku, *browsing* internet,

serta literatur-literatur yang berhubungan dengan permasalahan yang dijadikan bahan objek penelitian.

3. Penelitian Laboratorium (*Laboratory Research*)

Merupakan tahap penelitian yang dilakukan dengan cara *research* menggunakan komputer guna untuk mempraktekkan langsung hasil dari analisis yang bertujuan untuk menguji kebenaran sistem yang dirancang.

3.3. Analisa

Tahapan analisa data merupakan tahapan yang paling penting dalam pengembangan sebuah sistem, karena pada tahap inilah nantinya dilakukan evaluasi kinerja, identifikasi terhadap masalah yang ada, rancangan sistem dan langkah-langkah yang dibutuhkan untuk perancangan yang diinginkan sampai pada analisa yang diharapkan, (Soetanto, H. 2019).

Penelitian ini menggunakan bahasa pemrograman PHP sebagai alat bantu untuk pengambilan keputusan. PHP merupakan *Software Open Source* dan *cross Platform* yang dapat digunakan dalam berbagai bidang. Perancangan pada analisa data ini terdiri dari analisa data faktor beserta bobotnya. Hal tersebut dapat dijelaskan pada uraian dibawah ini :

3.3.1. Analisa Data

Masalah yang dibahas dalam penelitian ini adalah mengenai seleksi calon pegawai. Setelah melakukan wawancara dengan pihak yang biasa menangani masalah seleksi calon pegawai yaitu Ibu Kompol Yessi Kurniati, penulis mendapatkan kejelasan tentang beberapa faktor yang mempengaruhi calon pegawai polisi dapat dinyatakan lulus. Beberapa faktor pendukung beserta bobotnya dapat kita lihat pada tabel 3.1 Berikut ini :

Tabel 3.1 Data Faktor Pendukung

Nama Faktor	Bobot Faktor	Keterangan
Kesehatan	40%	Faktor kesehatan meliputi kesehatan fisik misalnya bebas narkoba, tidak lumpuh, tidak buta warna dan segala hal yang berhubungan dengan fisik atau tidak ada permasalahan pada fisik sama sekali.
Psikologi	25%	Faktor psikologi menjadi salah satu faktor penting lainnya setelah faktor kesehatan karena itu kesehatan psikologi calon peserta tersebut harus sudah teruji dan siap untuk menanggung beban seberat apapun dan dalam kondisi apapun.
Akademik	20%	Faktor akademik ditentukan pada saat calon peserta mengikuti tes akademik yang akan dilakukan oleh pihak Polda Sumatera Utara.
Kedisiplinan	15%	Faktor kedisiplinan diliput secara umum ketika calon peserta mulai mengikuti tes sampai pada saat calon peserta sampai pada tahap ini.

3.3.2. Analisa Proses

Dalam pengembangan sistem pendukung keputusan ini pengetahuan dan informasi diperoleh dari beberapa sumber, yaitu dari Pihak yang bersangkutan serta dari jurnal tentang seleksi maupun jurnal tentang metode yang dibahas. Pengetahuan akan *direpresentasikan* dengan menggunakan sebuah metode yang berguna untuk menemukan kesimpulan terhadap hasil proses seleksi. Pada prosesnya dengan menggunakan metode ini penentuan faktor utama sangatlah dibutuhkan oleh karena itu untuk menentukan kualitas dari suatu faktor yang telah di tentukan maka perlulah diberikan pembobotan sesuai dengan kapasitas bobot masing-masing.

Berikut Beberapa prosedur seleksi nilai calon pegawai Polri yang akan diterapkan pada sistem ini :

1. Seleksi 1

Tahapan awal pemeriksaan yaitu memisahkan berkas hasil nilai text calon pegawai Polri yang memiliki nilai bobot test dibawah nilai bobot minimum. Jika calon pegawai Polri memiliki salah satu nilai faktor test dari tiga faktor test dengan nilai dibawah nilai minimum faktor, maka calon pegawai Polri itu dinyatakan gagal.

2. Seleksi 2

Pada tahap kedua, bagian kepegawaian Polda akan melihat seluruh kemungkinan dalam kriteria penempatan yang dapat diperoleh dari hasil akhir nilai test calon pegawai Polri. Seperti yang sudah dijelaskan pada bagian kriteria bahwa sudah menjadi kesepakatan dibawah nilai minimum dari seluruh kriteria yang ada akan dinyatakan gagal.

3. Seleksi 3

Tahap terakhir dari perhitungan penyeleksian nilai akhir calon pegawai Polri adalah mengurutkan dari nilai terbaik sampai batas pencarian jumlah nilai terbaik sampai batas pencarian jumlah pegawai polri baru yang akan diterima kerja. Nilai akhir dari seluruh calon pegawai Polri yang lolos sampai pada tahapan terakhir ini akan dimasukkan pada kriteria penempatan. Penyeleksian akhir dimulai dengan merata – ratakan nilai faktor calon pegawai Polri sesuai dengan prioritas test factor dari setiap kriteria yang ada.

3.3.3. Analisa Sistem

Pada penerapan metode *Multifactor Evaluation Process* dibutuhkan pihak yang memahami untuk menentukan bobot dari suatu faktor pendukung yang pada Proses pencarian bobot dengan melakukan wawancara terhadap pihak yang bersangkutan, (Turaina, R., & Efendy, C. G. 2016)

Menentukan Faktor yang dijadikan acuan dalam pengambilan keputusan beserta bobotnya.

Tabel 3.2 Bobot Faktor Pendukung

No	Nama Faktor Pendukung	Bobot
1	Kesehatan	40%
2	Psikologi	25%
3	Akademik	20%
4	Kedisiplinan	15%

Setelah dilakukan pembobotan, calon pegawai polisi yang melamar akan ditimbang dan dievaluasi dan diberikan nilai bobot untuk masing masing setiap kriteria seperti contoh dibawah ini :

Tabel 3.3 Proses Perhitungan

Faktor	Calon 1	Calon 2	Calon 3	Ke-n
Kesehatan	6	3	5	-
Psikologi	5	3	9	-
Akademik	7	8	3	-
Kedisiplinan	3	9	7	-

Dengan adanya informasi tersebut diatas, didapat jumlah total nilai Evaluasi untuk setiap alternatif atau peserta seleksi. Setiap peserta seleksi mempunyai sebuah nilai evaluasi bagi ketiga faktor-faktor yang menjadi pertimbangannya, untuk mendapatkan nilai total evaluasi setiap pelamar dengan cara perhitungan sebagai berikut :

Perhitungan nilai bobot evaluasi

$$\mathbf{Nbe = Nbf \times Nef}$$

Keterangan:

Nbe → Nilai Bobot Evaluasi

Nbf → Nilai Bobot Faktor

Nef → Nilai Evaluation Faktor

Perhitungan total nilai evaluasi

$$Tne = Nbe1 + Nbe2 + Nbe3, \dots$$

Keterangan :

Tne → Total Nilai Evaluasi

Nbe → Nilai Bobot Evaluasi

Tabel 3.4 . Evaluasi Untuk Nama Peserta Calon 1

Faktor	Bobot Faktor		Evaluasi Faktor	Bobot Evaluasi
Kesehatan	0.4	X	6	2.4
Psikologi	0.25	X	5	1.25
Akademik	0.20	X	7	1.4
Kedisiplinan	0.15	X	3	0.45
Total	1			5.5

Tabel 3.5 . Evaluasi Untuk Nama Peserta Calon 2

Faktor	Bobot Faktor		Evaluasi Faktor	Bobot Evaluasi
Kesehatan	0.4	X	3	1.2
Psikologi	0.25	X	3	0.75
Akademik	0.20	X	8	1.6
Kedisiplinan	0.15	X	9	1.35
Total	1			4.9

Tabel 3.6 . Evaluasi Untuk Nama Peserta Calon 3

Faktor	Bobot Faktor		Evaluasi Faktor	Bobot Evaluasi
Kesehatan	0.4	X	5	2
Psikologi	0.25	X	9	2.25
Akademik	0.20	X	3	0.6
Kedisiplinan	0.15	X	7	1.05
Total	1			5.9

Langkah perhitungannya :

Nilai Evaluasi setiap faktor bobot dikalikan evaluasi faktor, seperti dibawah ini:

$$\begin{aligned} \text{Alternatif Calon 1} &= (0.4 \times 6) + (0.25 \times 5) + (0.20 \times 7) + (0.15 \times 3) \\ &= 2.4 + 1.25 + 1.4 + 0.45 \\ &= 5.5 \end{aligned}$$

$$\begin{aligned} \text{Alternatif Calon 2} &= (0.4 \times 3) + (0.25 \times 3) + (0.20 \times 8) + (0.15 \times 9) \\ &= 1.2 + 0.75 + 1.6 + 1.35 \\ &= 4.9 \end{aligned}$$

$$\begin{aligned} \text{Alternatif Calon 3} &= (0.4 \times 5) + (0.25 \times 9) + (0.20 \times 3) + (0.15 \times 7) \\ &= 2 + 2.25 + 0.6 + 1.05 \\ &= 5.9 \end{aligned}$$

Tabel 3.7 Hasil Total Bobot Evaluasi

No	Nama	Total Bobot Evaluasi
1	Calon 1	5.5
2	Calon 2	4.9
3	Calon 3	5.9

Dari proses perhitungan di atas setelah didapatkan total bobot evaluasi maka akan dilakukan sorting berdasarkan nilai Total Bobot Evaluasi yang terbesar ke yang terkecil setelah didapatkan total bobot yang terbesar maka diperlukan kecocokan

kuota yang tersedia dengan alternatif yang ada dan yang berada dibawah kuota yang telah terpenuhi maka alternatif tersebut dinyatakan gagal, semua itu dapat kita lihat tabel dibawah ini setelah dilakukan sorting berdasarkan total bobot evaluasi.

Tabel 3.8 Hasil Setelah Sorting

No	Nama	Total Bobot Evaluasi
1	Calon 3	5.9
2	Calon 1	5.5
3	Calon 2	4.9

Dari hasil perhitungan dengan metode MFEP menentukan bahwa alternatif dengan nilai tertinggi adalah solusi terbaik berdasarkan kriteria yang telah dipilih, dalam contoh yang digunakan hasil nilai tertinggi pertama adalah nama peserta Amin dan nilai tertinggi kedua dimiliki oleh alternatif yang bernama Budi.

3.4. Perancangan UML

Perancangan UML (*Unified Modelling Language*) adalah untuk menentukan cara kerja program “Sistem Pendukung Keputusan Seleksi Calon Pegawai Baru Di Polda Sumatera Utara Menggunakan Metode Multifactor Evaluation Process (MFEP)” yang menggunakan *Use Case Diagram, Class Diagram, Sequence Diagram, State Diagram, Collaboration Diagram, Deployment Diagram* dan *Activity Diagram*. Untuk perancangan UML ini menggunakan program *Star UML*.

Adapun perancangan UML (*Unified Modelling Language*) untuk perancangan sistem pendukung keputusan untuk menseleksi calon pegawai polisi adalah sebagai berikut :

3.4.1. Use Case Diagram

Use case menggambarkan bagaimana seseorang akan menggunakan atau memanfaatkan sistem, sedangkan aktor adalah seseorang atau sesuatu yang berinteraksi dengan sistem. *Use case* diagram menggambarkan bagaimana proses-proses yang dilakukan oleh aktor terhadap sistem.

a. Definisi Actor

Definisi aktor adalah aktivitas yang dilakukan oleh para aktor dalam menggunakan *system* dapat kita lihat pada tabel 3.9 Berikut ini :

Tabel 3.9 Definisi Aktor

No.	Aktor	Deskripsi
1.	Admin/User	Aktor yang memonitoring segala isi sistem dan juga sebagai aktor yang menggunakan isi sistem tersebut

b. Definisi Use case

Definisi dalam *Use Case* diagram adalah kegiatan-kegiatan yang akan terjadi didalam sistem antara para aktor dengan *Use Case* dapat kita lihat pada Tabel 3.10 Berikut ini :

Tabel 3.10 Skenario Login

Identifikasi	
No	Use case 01
Nama	Login
Tujuan	Untuk menjaga database hanya pihak user yang bisa masuk ke dalam database.
Diskripsi	Aktor yang memonitoring segala isi sistem pendukung keputusan.
Aktor	User
Skenario	
Kondisi Awal	Menampilkan menu login, <i>username</i> dan <i>password</i>
Aksi Aktor	Reaksi Sistem
User melakukan login untuk masuk ke dalam database	Sistem akan menampilkan menu login dan memprosesnya
Kondisi Akhir	Tampilan akan menampilkan home, logout.

Tabel 3.11 Skenario Manajemen Alternatif

Identifikasi	
No	Use case 02
Nama	Manajemen Alternatif
Tujuan	Untuk memanajemen alternatif yang akan diproses oleh sistem.
Diskripsi	Digunakan untuk melihat, mengedit, membuat, dan menghapus alternatif yang ada.
Aktor	User
Skenario	
Kondisi Awal	Menu entri sudah aktif
Aksi Aktor	Reaksi Sistem
User melakukan proses entri alternatif	Sistem akan memproses perintah user untuk membuat, mengedit atau menghapus alternatif yang ada
Kondisi Akhir	Tampilan akan menampilkan menu manajemen alternatif.

Tabel 3.12 Skenario Manajemen Kriteria

Identifikasi	
No	Use case 03
Nama	Manajemen Kriteria
Tujuan	Untuk memanajemen kriteria yang akan diproses oleh sistem.
Diskripsi	Digunakan untuk melihat, mengedit, membuat, dan menghapus kriteria yang ada.
Aktor	User
Skenario	
Kondisi Awal	Menu entri sudah aktif
Aksi Aktor	Reaksi Sistem
User melakukan proses entri alternatif	Sistem akan memproses perintah user untuk membuat, mengedit atau menghapus kriteria yang ada.
Kondisi Akhir	Tampilan akan menampilkan menu manajemen kriteria.

Tabel 3.13 Skenario Manajemen User

Identifikasi	
No	Use case 04
Nama	Manajemen User
Tujuan	Untuk memanajemen User yang berhak mengakses sistem ini.
Diskripsi	Digunakan untuk melihat dan mengedit user yang memiliki hak akses.
Aktor	User
Skenario	
Kondisi Awal	Menu entri sudah aktif
Aksi Aktor	Reaksi Sistem
User melakukan proses manajemen user	Sistem akan memproses perintah user untuk melihat dan mengedit user yang ada.
Kondisi Akhir	Tampilan akan menampilkan menu manajemen user

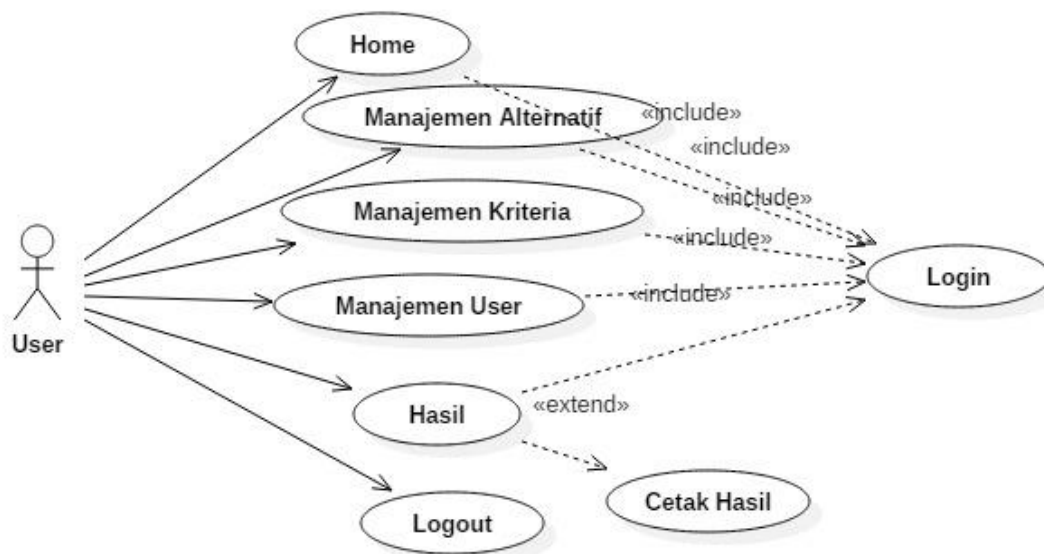
Tabel 3.14 Skenario Home

Identifikasi	
No	Use case 05
Nama	Home
Tujuan	Untuk melihat sebuah tampilan awal dalam web.
Diskripsi	Menampilkan halaman web.
Aktor	User
Skenario	
Kondisi Awal	Tampilan program sudah aktif.
Aksi Aktor	Reaksi Sistem
User mencari informasi penyakit	Sistem akan melakukan apa yang diminta user.
Kondisi Akhir	Sistem akan menampilkan data yang di minta user.

Tabel 3.15 Skenario Hasil

Identifikasi	
No	Use case 06
Nama	Hasil
Tujuan	Untuk melihat hasil proses perhitungan SPK dengan metode MFEP
Diskripsi	Menampilkan Hasil proses MFEP
Aktor	User
Skenario	
Kondisi Awal	Menu hasil aktif
Aksi Aktor	Reaksi Sistem
User akan melihat daftar alternatif yang lulus dan tidak	Sistem akan menampilkan hasil alternatif yang sudah diproses.
Kondisi Akhir	User akan mengetahui informasi tentang lulus tidak nya alternatif tersebut.

Gambar *Use Case* diagram, yaitu menggambarkan scenario atau interaksi yang dapat dilakukan oleh seorang aktor, aktor disini hanya berupa admin, karena sistem ini dibangun untuk mendukung pihak pimpinan dalam menentukan keputusan jadi tidak diwajibkan untuk diketahui dalam user yang banyak. Adapun *use case diagram* tersebut dapat kita lihat pada Gambar 3.2 berikut ini :



Gambar 3.2 Use Case Diagram

3.4.2. Class Diagram

Class diagram sering digunakan untuk menampilkan kelas tabel, *field* dan operasi dari program ke *database* yang digunakan.

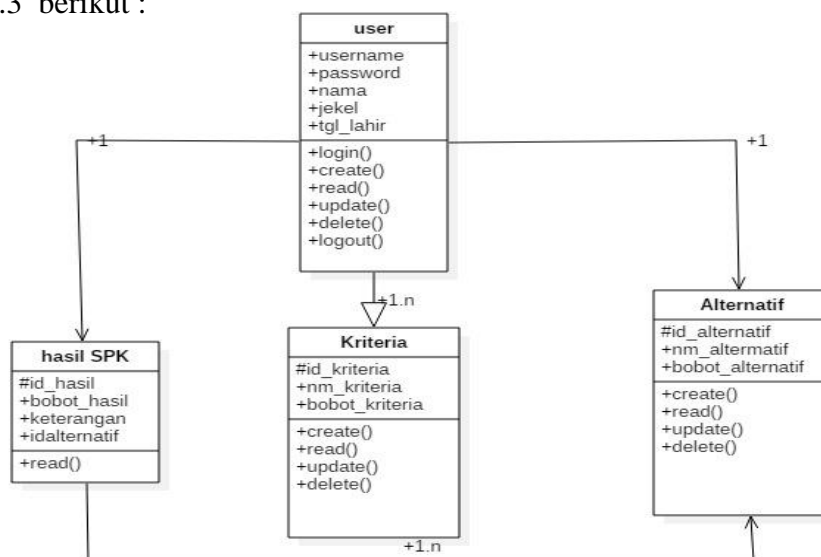
a. Definisi Class

Spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Untuk lebih jelasnya dapat kita lihat pada Tabel 3.16 berikut ini :

Tabel 3.16 Definisi Class

No.	Class	Deskripsi
1.	Admin/User	Merupakan tabel yang berisikan data user seperti <i>username</i> dan <i>password</i> dan informasi tentang user untuk melakukan login.
2.	tbl_alternatif	Merupakan tabel yang berisikan data dari alternatif-alternatif yang ada yang dalam hal ini adalah calon pegawai ke polisia.
3.	tbl_kriteria	Merupakan tabel yang berisikan data faktor pendukung yang menentukan hasil seleksi calon pegawai polisi.
4.	tbl_hasil	Berikan data-data hasil proses MFEP yang telah diproses dan disimpan didalam tabel ini dengan bobot yang baru.

Adapun class diagram dari sistem yang akan dibuat dapat kita lihat pada gambar 3.3 berikut :



Gambar 3.3 Class Diagram

b. Perancangan *Database*

Dalam perancangan *database* sistem pendukung keputusan untuk menentukan bobot hasil dari alternatif yang ada, terhadap empat tabel *class diagram*. Dalam tabel tersebut terdiri dari beberapa *record*, *record* terbagi atas beberapa *field*, dan setiap *record* akan menampung data untuk menghasilkan informasi. Bentuk dari *database* tersebut yaitu sebagai berikut :

1. Tabel User

Tabel yang digunakan sebagai tempat penyimpanan data *username* dan *password* User. Dimana *username* dan *password* nantinya dapat digunakan sebagai login pada sistem sebagai operator. Adapun bentuk tabel *User* yang ada di *database* dapat kita lihat pada Tabel 3.17 berikut ini :

Tabel 3.17 Tabel User

No.	Field	Type	Keterangan
1.	Username	Varchar (35)	<i>Primery Key</i>
2.	Password	Varchar (35)	
3	Nama	Varchar	
4	Jekel	Varchar	
5	Tgl_lahir	Date	

2. Tabel Kriteria

Tabel Faktor merupakan tabel yang digunakan sebagai tempat penyimpanan data-data dari jenis beberapa kriteria penentu yang ada yang

diinputkan oleh *User*. Adapun bentuk tabel yang ada di *database* dapat kita lihat pada Tabel 3.18 berikut ini :

Tabel 3.18 Tabel Kriteria

No.	Field	Type	Keterangan
1.	id_kriteria	Varchar (15)	Primery Key
2.	nama_kriteria	Varchar (50)	
3	Bobot_kriteria	Float	

3. Tabel Alternatif

Tabel Alternatif merupakan tabel yang digunakan sebagai tempat penyimpan data dari alternatif-alternatif yang sudah diproses dan memiliki evaluasi bobot pada tiap-tiap alternatif yang diinputkan oleh *User*. Adapun bentuk tabel Alternatif yang ada di *database* dapat kita lihat pada tabel 3.19 berikut ini :

Tabel 3.19 Tabel Alternatif

No.	Field	Type	Keterangan
1.	id_alternatif	Int	<i>Primary Key</i>
2	nm_alternatif	Varchar (100)	
3	Bobot Alternatif	Float	

3.4.3. *Sequence Diagram* dan *Collaboration Diagram*

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

perilaku pada sebuah *scenario* yang diterapkan pada sistem untuk model penerapan pada bahasa pemograman.

Collaboration diagram adalah bentuk lain dari *sequence diagram*. Bila *sequence diagram* diorganisir menurut waktu maka *collabration diagram* diorganisir menurut ruang atau *space*.

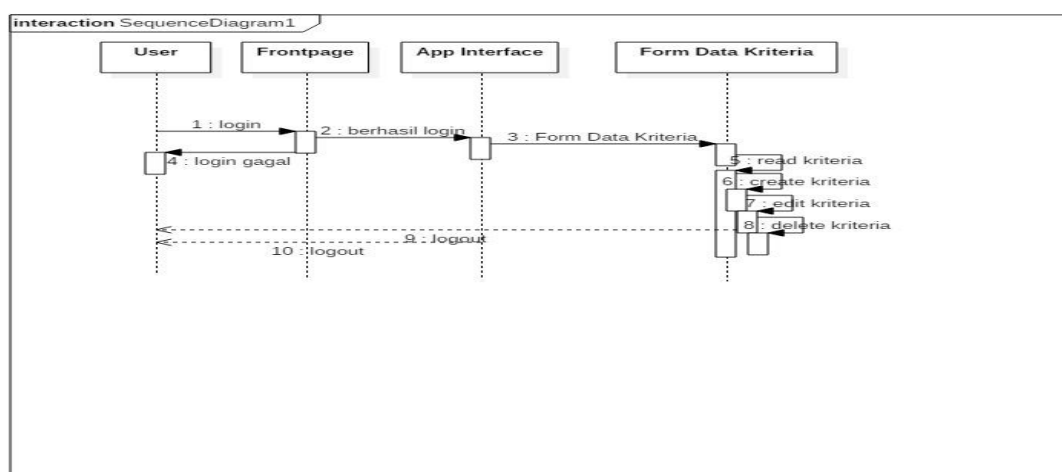
Berikut gambar-gambar *sequence diagram* dan *collaboration diagram* yang terdapat pada sistem, diantaranya :

a. *Sequence Diagram dan Collaboration Diagram* User untuk Mengelola Data Kriteria

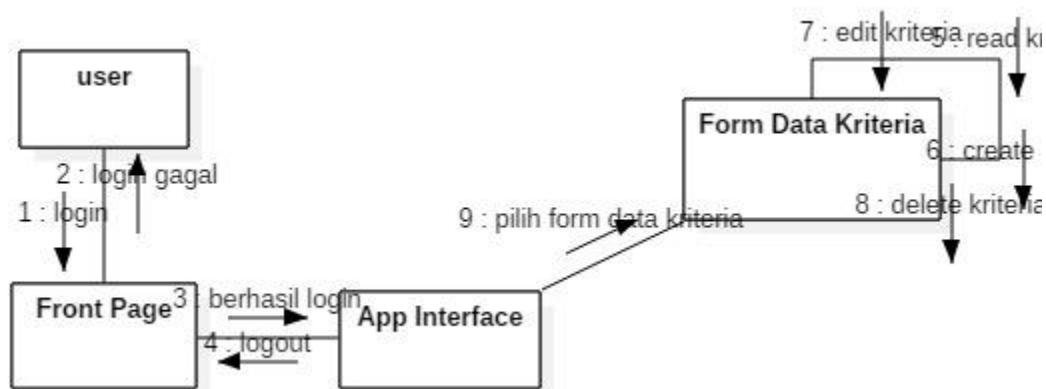
Diagram *sequence diagram* ini menggambarkan perilaku dan skenario yang dilakukan oleh user untuk melihat, menambah, mengedit, menghapus data kriteria dalam *database*.

Diagram *collaboration* ini menjelaskan tahapan-tahapan terjadinya suatu aktivitas aktor user yang memperlihatkan bagaimana hubungan antara beberapa objek untuk melakukan melihat, menambah, mengedit dan menghapus data kriteria dalam *database*, (Verina, W, dkk, 2018).

Aktivitas tersebut dapat kita lihat pada gambar Berikut ini :



Gambar 3.4 *Sequence Diagram* User untuk Mengelola Data Kriteria



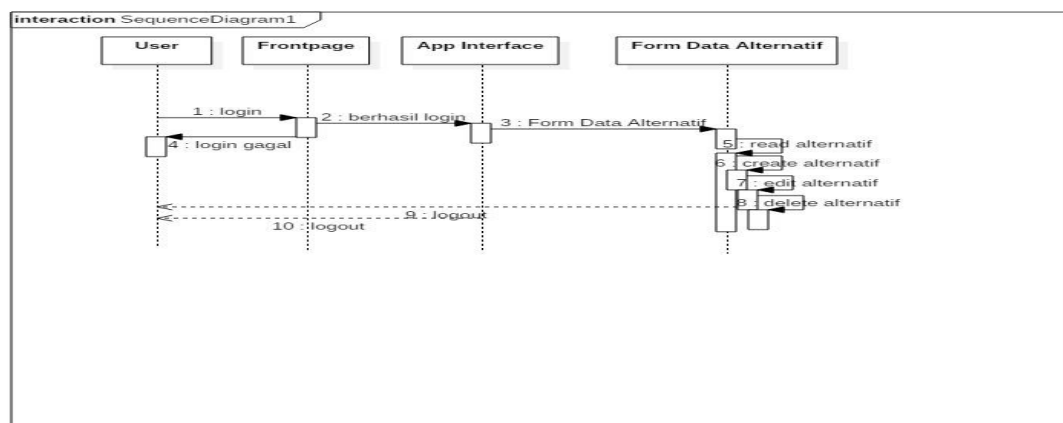
Gambar 3.5 Collaboration Diagram User untuk Mengelola Data Kriteria

b. Sequence Diagram dan Collaboration Diagram User untuk Mengelola Data Alternatif

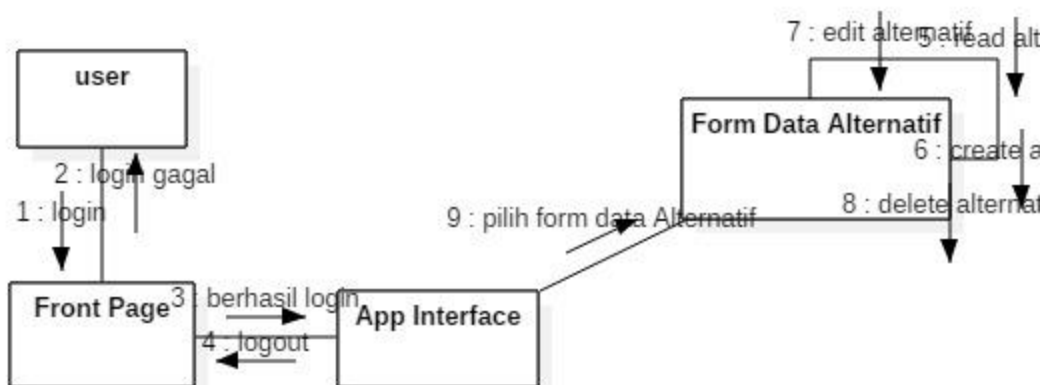
Diagram ini menggambarkan perilaku dan skenario yang dilakukan oleh user untuk melihat, menambah, mengedit, menghapus data alternatif dalam *database*.

Diagram collaboration ini menjelaskan tahapan-tahapan terjadinya suatu aktivitas aktor user yang memperlihatkan bagaimana hubungan antara beberapa objek untuk melakukan melihat, menambah, mengedit dan menghapus data kriteria dalam *database*.

Aktivitas tersebut dapat kita lihat pada gambar Berikut ini :



Gambar 3.6 Sequence Diagram User untuk Mengelola Data Alternatif



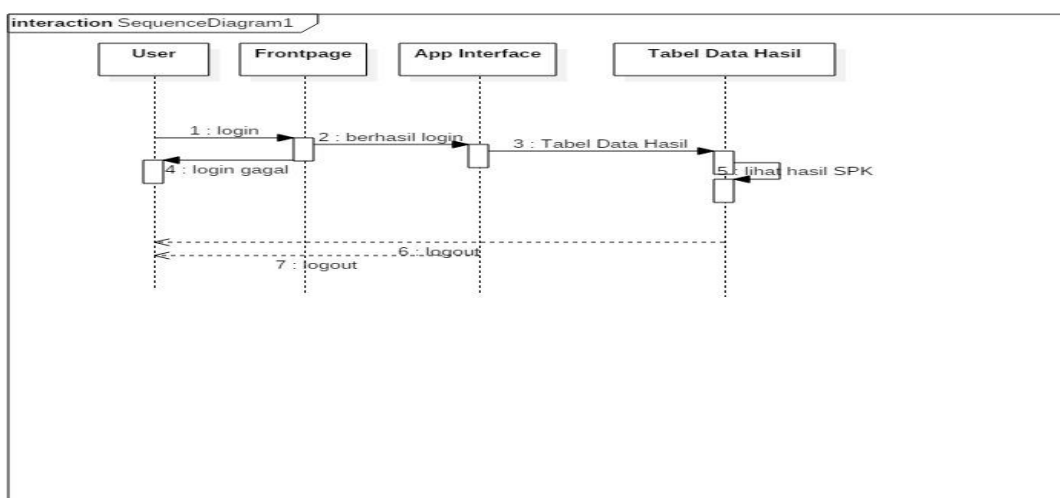
Gambar 3.7 Collaboration Diagram User untuk Mengelola Data Kriteria

c. Sequence Diagram dan Collaboration Diagram User Untuk Mengelola Data Solusi

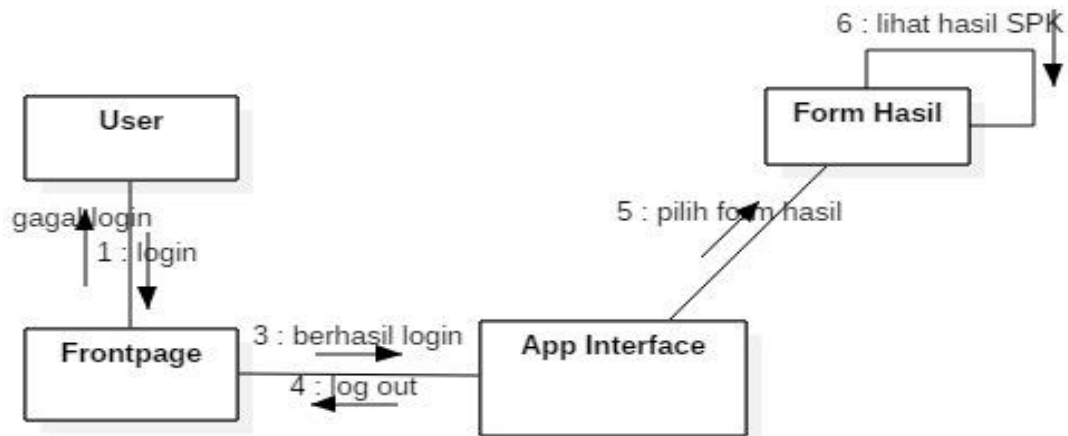
Sequence Diagram ini menggambarkan perilaku dan skenario yang dilakukan oleh user untuk melihat hasil perhitungan bobot yang telah diproses dan disimpan ke dalam *database*.

Collaboration Diagram ini menjelaskan tahapan-tahapan terjadinya suatu aktifitas aktor user yang memperlihatkan bagaimana hubungan antara beberapa objek untuk melihat hasil perhitungan MFEP yang telah tersimpan di dalam *database*.

Aktivitas tersebut dapat kita lihat pada gambar 3.8 Berikut ini :



Gambar 3.8 Sequence Diagram User untuk Mengelola Data Solusi



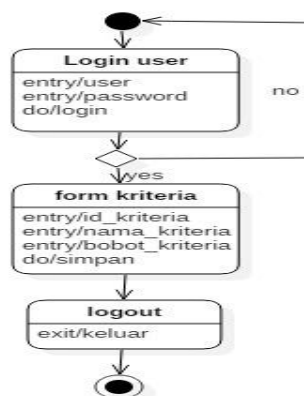
Gambar 3.9 Collaboration Diagram Admin untuk Mengelola Data Solusi

3.4.4. Statechart Diagram

Statechart Diagram biasanya digunakan untuk memodelkan perilaku dinamis suatu *class* atau obyek. Diagram ini memperlihatkan urutan *state* (aktivitas) yang dilalui sebuah obyek, kejadian yang menyebabkan sebuah transisi dari suatu *state* ke *state* yang lain, dan aksi yang menyebabkan perubahan *state* atau aktivitas. Beberapa *Statechart Diagram* pada sistem, antara lain :

1. Statechart Diagram pada Form Tambah Kriteria

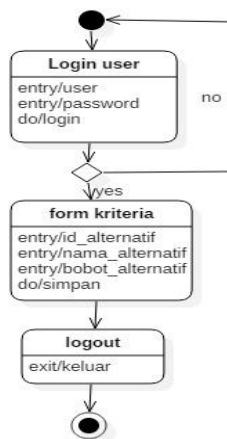
Diagram ini menjelaskan *state* yang terjadi pada Form Tambah Kriteria, yang meliputi proses menginputkan data-data kriteria yang didapatkan dari pihak yang bersangkutan yang dalam hal ini adalah pihak Polda Sumatera Utara.



Gambar 3.10 Statechart Diagram Form Tambah Kriteria

2. Statechart Diagram pada Form Alternatif

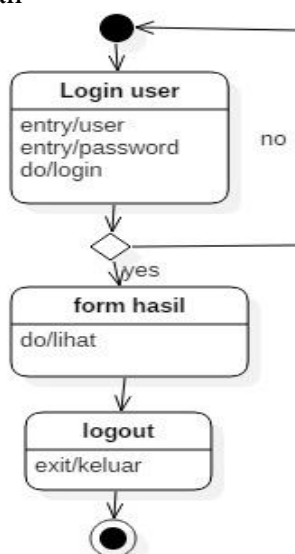
Diagram ini menjelaskan *state-state* yang terjadi pada Form Alternatif, yang meliputi proses menginputkan data-data alternatif hingga penentuan masing-masing bobot pada tiap-tiap alternatif yang ada.



Gambar 3.11 Statechart Diagram Form Tambah Alternatif

3. Statechart Diagram pada Form Analisa Hasil

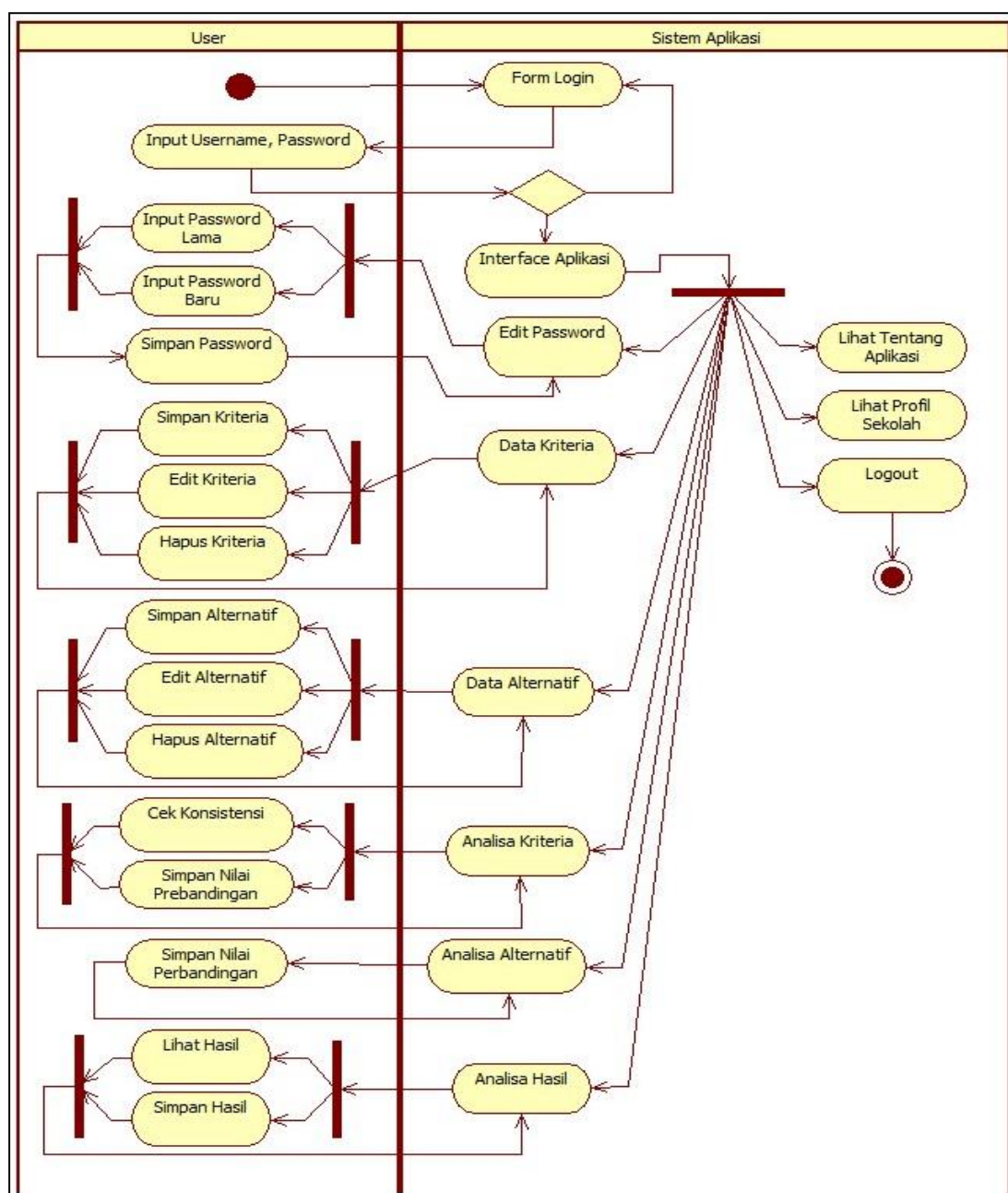
Diagram ini menjelaskan *state-state* yang terjadi pada Form Analisa Hasil, meliputi proses perhitungan dan menampilkan hasil yang sudah diproses oleh sistem hingga menjadi informasi yang dapat bermanfaat oleh pihak yang bersangkutan



Gambar 3.12 Statechart Diagram Form Analisa Hasil

3.4.5. Activity Diagram

Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang dirancang, dari mana masing-masing aliran berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Berikut ini *activity* diagram *user* menggambarkan segala aktivitas yang bisa dilakukan oleh *user* pada Aplikasi Sistem Pendukung Keputusan.

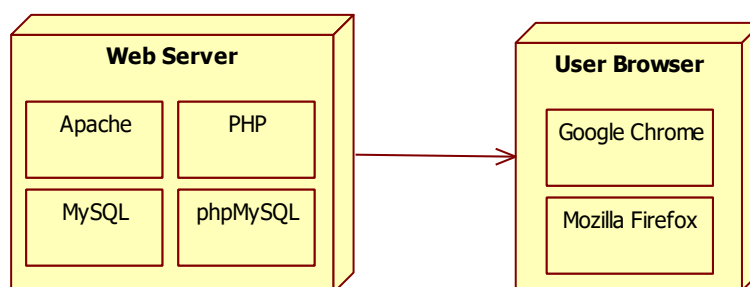


Gambar 3.13 Activity Diagram Aplikasi SPK

3.4.6. *Deployment Diagram*

Deployment atau *physical diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, *server* atau piranti keras), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah *server*, *workstation*, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya *TCP/IP*) dan *requirement* dapat juga didefinisikan dalam diagram ini.

Pada perancangan sistem ini menggunakan *web server Apache* dan *database server MySQL*. sedangkan *user* dapat melakukan akses dengan menggunakan *web browser* seperti Mozilla Firefox, Internet Explorer dan Google Chrome.



Gambar 3.14 Deployment Diagram

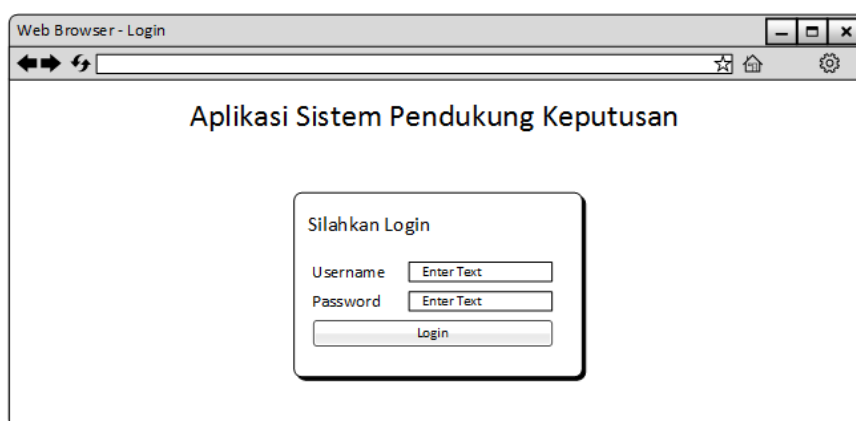
3.5. Perancangan *Interface*

Desain antarmuka merupakan suatu sistem yang mempertimbangkan keinginan pengguna, yaitu aplikasi yang memenuhi kebutuhan dan kemudahan untuk digunakan. Desain ini juga menggambarkan tampilan halaman *website* yang akan dirancang. Pada halaman awal terdapat perintah (*link*) yang dapat memanggil dan membuka halaman selanjutnya, dan begitu pula pada halaman-halaman yang lainnya.

Beberapa desain yang akan dibuat pada sistem sebagai gambaran kasarnya dapat kita lihat pada gambar-gambar berikut ini :

3.5.1. Tampilan Halaman Login

Halaman login adalah halaman awal yang akan muncul ketika Aplikasi Sistem Pendukung Keputusan ini dibuka. Halaman login ini ditujukan untuk memberikan keamanan terhadap sistem yang dibangun. Dan tampilannya dapat dilihat pada gambar berikut :

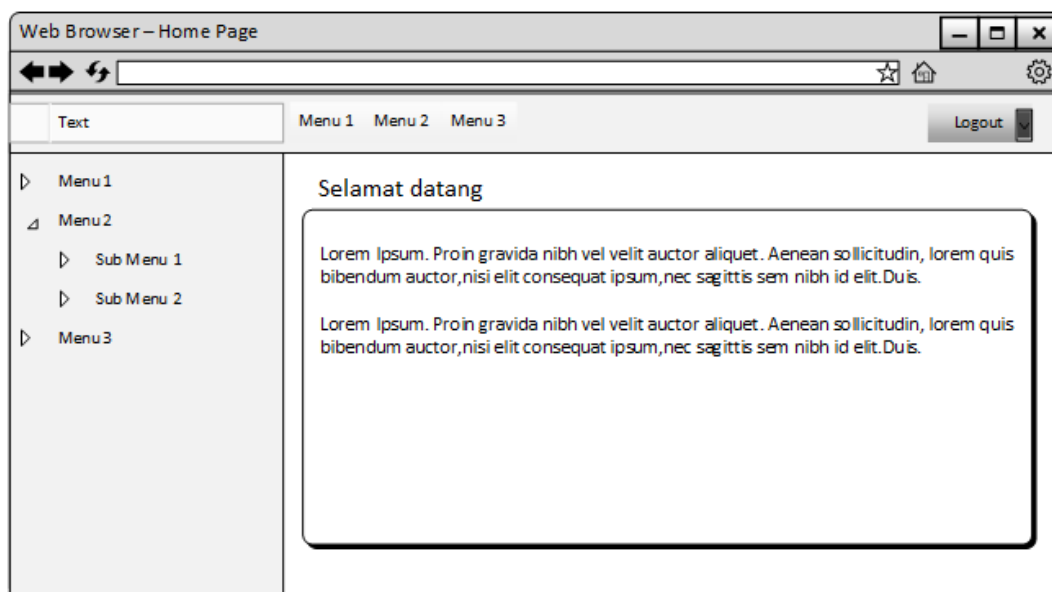


The image shows a web browser window titled "Web Browser - Login". The address bar is empty. The main content area displays the title "Aplikasi Sistem Pendukung Keputusan" centered at the top. Below the title is a login form with the heading "Silahkan Login". The form contains three input fields: "Username" with a placeholder "Enter Text", "Password" with a placeholder "Enter Text", and a "Login" button.

Gambar 3.15 Halaman Login

3.5.2. Tampilan Halaman Beranda

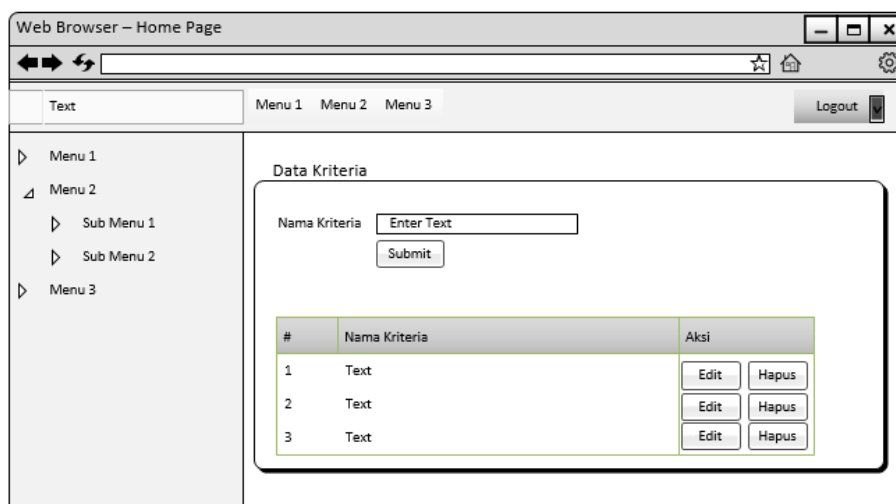
Halaman beranda atau *Home page* merupakan halaman depan dari Aplikasi Sistem Pendukung Keputusan. Dan tampilannya dapat dilihat pada gambar berikut :



Gambar 3.16 Halaman Beranda

3.5.3. Tampilan Halaman Data Kriteria

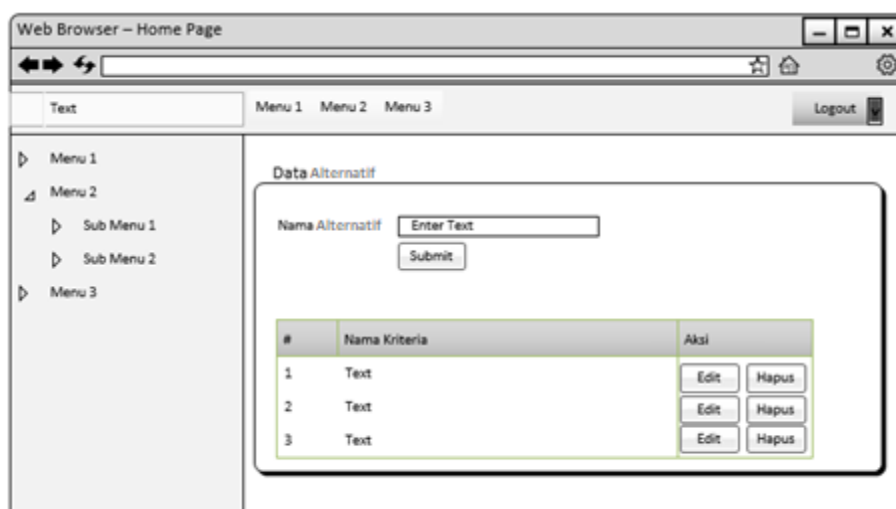
Halaman data kriteria ini merupakan halaman untuk menambah, melihat, mengedit dan menghapus data kriteria. Dan tampilannya dalam dilihat pada gambar berikut :



Gambar 3.17 Halaman Data Kriteria

3.5.4. Tampilan Halaman Data Alternatif

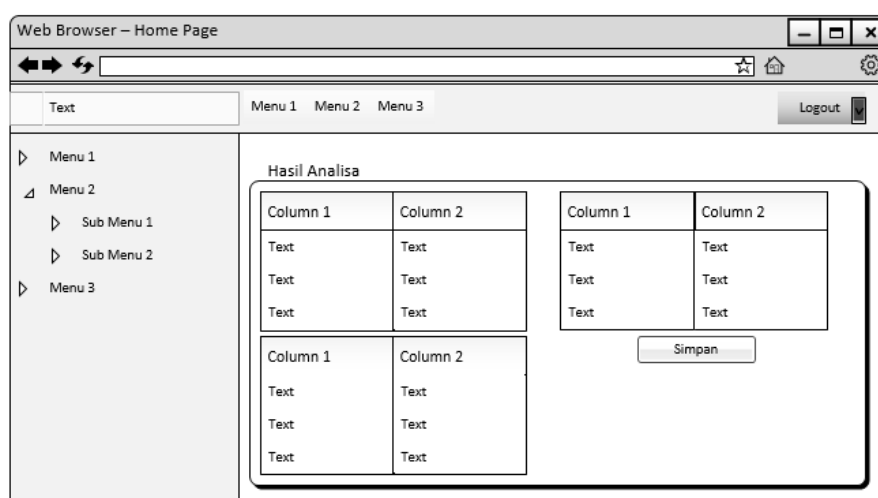
Halaman data alternatif ini merupakan halaman untuk menambah, melihat, mengedit dan menghapus data alternatif. Dan tampilannya dalam dilihat pada gambar berikut :



Gambar 3.18 Halaman Data Alternatif

3.5.5. Tampilan Halaman Hasil Analisa

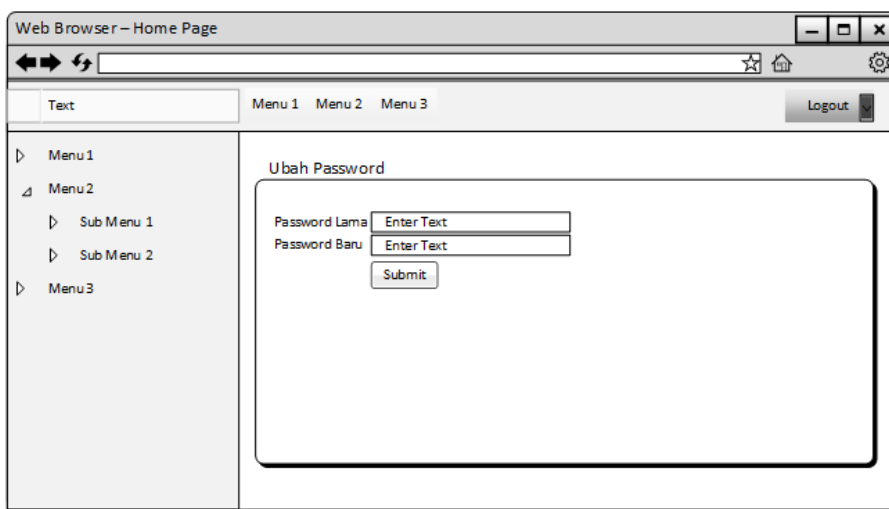
Halaman hasil analisa merupakan halaman untuk menganalisa data kriteria dan data alternatif yang telah dihitung untuk mendapatkan hasil yang diinginkan. Dan tampilannya dalam dilihat pada gambar berikut :



Gambar 3.19 Halaman Hasil Analisa

3.5.6. Tampilan Halaman Edit Password

Halaman edit password digunakan untuk user yang menggunakan aplikasi agar dapat mengedit password yang diinginkannya. Dan tampilannya dalam dilihat pada gambar berikut :



The image shows a web browser window titled "Web Browser - Home Page". The browser's address bar is empty. Below the address bar, there is a navigation bar with a search box containing the text "Text", and three menu items: "Menu 1", "Menu 2", and "Menu 3". A "Logout" button is located on the right side of the navigation bar. The main content area of the browser displays a form titled "Ubah Password". The form contains two text input fields: "Password Lama" (Old Password) and "Password Baru" (New Password), both with placeholder text "Enter Text". Below these fields is a "Submit" button.

Gambar 3.20 Halaman Edit Password

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi

4.1.1 Persiapan Implementasi

Implementasi ini didemokan dengan menggunakan komputer dan *webserver* lokal. Untuk mengimplementasikan aplikasi ini dibutuhkan perangkat keras dan perangkat lunak yang memadai, dalam mengimplementasikan aplikasi Sistem Pendukung Keputusan ini penulis menggunakan perangkat keras (*hardware*) dengan spesifikasi sebagai berikut :

1. System Manufacture : Axer
2. System Model : Aspire 4738G
3. Processor : Intel(R) Core(TM) i3 CPU M370 @ 2.40GHz
4. Memory : 3072MB (RAM)
5. Harddisk : 500 GB

Serta perangkat lunak (software) yang digunakan untuk implementasi ini yaitu sebagai berikut :

1. Sistem Operasi Windows 10 Home 64-bit (10.0, Build 18363)
2. Browser Google Chrome
3. Web Server Xampp Control Panel v3.2.2, dengan rincian paket server sebagai berikut :
 - a. Apache
 - b. MySQL
 - c. FileZila
 - d. Mercury
 - e. Tomcat

4.1.2 Batasan Implementasi

Batasan implementasi dimaksudkan sebagai ruang lingkup sistem yang telah dibuat. Sehingga pengguna mengerti hal apa saja yang dapat dilakukan dalam sistem yang dibuat. Dan batasan dalam implementasi aplikasi Sistem Pendukung Keputusan adalah sebagai berikut :

1. Aplikasi Sistem Pendukung Keputusan ini dibuat dengan bahasa pemrograman PHP dan *database* MySQL.
2. Dalam aplikasi ini, pengguna dapat melakukan perhitungan dengan Sistem Pendukung Keputusan dengan cara *login* terlebih dahulu.

4.1.3. Implementasi Sistem

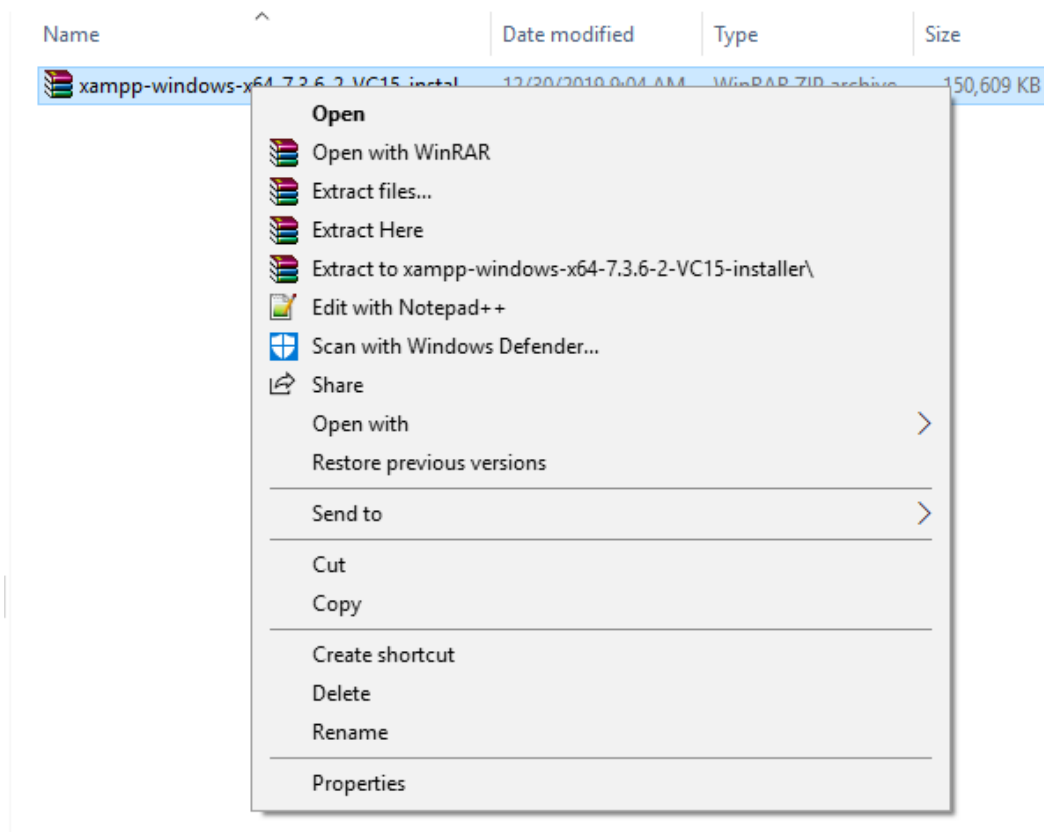
Implementasi sistem merupakan realisasi dari hasil perancangan yang telah diterapkan. Perancangan sistem yang sudah dibuat dikonversikan ke dalam bentuk program.

4.1.3.1. Persiapan *Web Server*

Untuk mengimplementasikan aplikasi Sistem Pendukung Keputusan, dibutuhkanlah aplikasi pendukung yang harus diinstall terlebih dahulu dan dikonfigurasi dengan benar agar program yang dibuat dapat berjalan dan berfungsi sebagaimana mestinya. Dan *aplikasi server* yang digunakan adalah Xampp.

Dan berikut ini adalah langkah-langkah untuk instalasi *webserver Xampp* :

1. Silahkan buat folder baru sebagai tempat penyimpanan hasil ekstrak file .RAR nya nanti



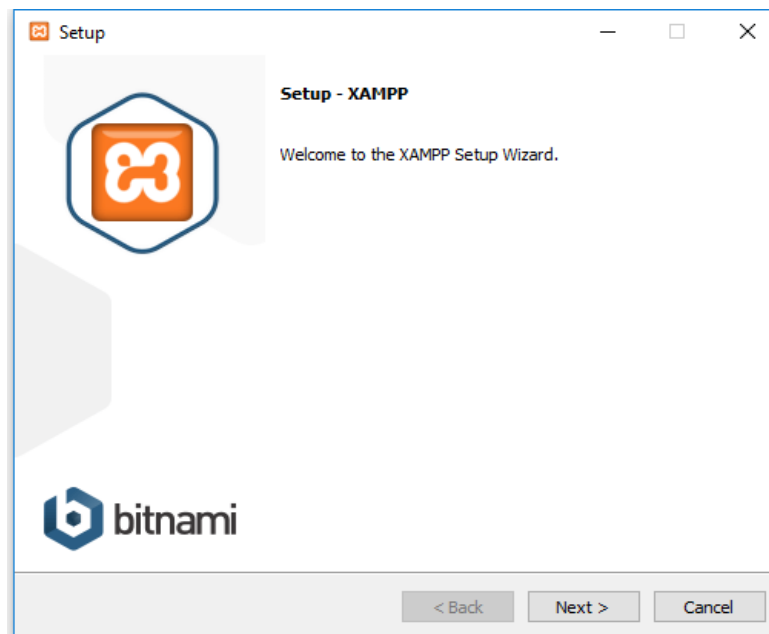
Gambar 4.1 Ekstraksi File Xampp

2. Berikut hasil file yang sudah di ekstrak tersebut lalu jalankan file .exe xampp

Name	Date modified	Type	Size
xampp-windows-x64-7.3.6-2-VC15-installer	7/22/2019 3:16 AM	Application	152,376 KB
xampp-windows-x64-7.3.6-2-VC15-installer.rar	12/30/2019 9:04 AM	WinRAR ZIP archive	150,609 KB

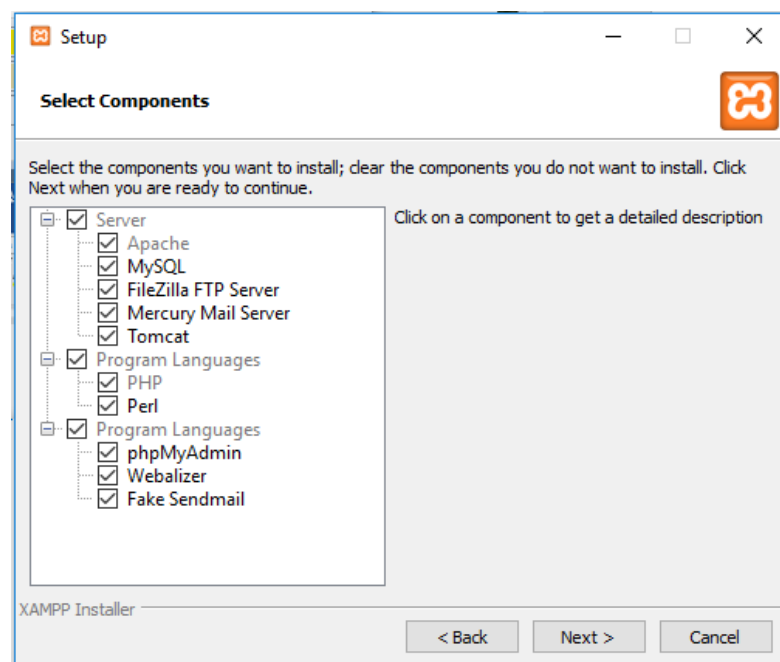
Gambar 4.2 File Installer

3. Lalu pada pilihan welcome to the XAMPP setup Wizard klik next.



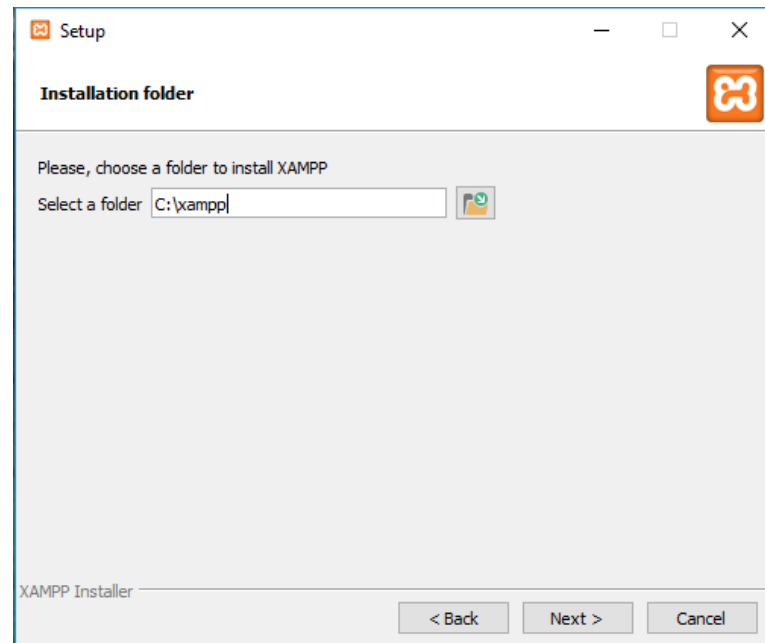
Gambar 4.3 Pilihan Next Setup

4. Lalu akan tampil beberapa pilihan packages apa saja yang akan di instal dalam aplikasi ini lalu pilih next.



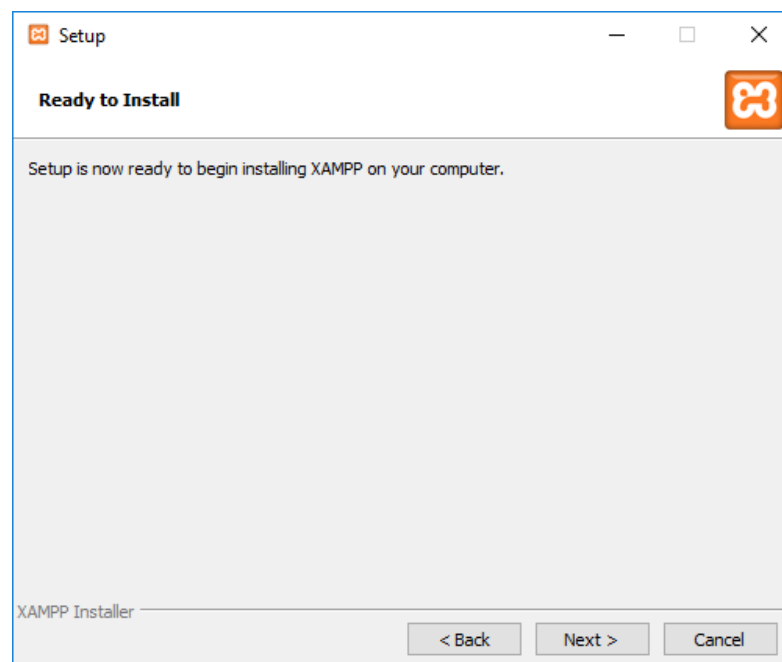
Gambar 4.4 Packages yang akan Di install

5. Lalu akan tampil menu di folder mana aplikasi xampp akan diinstal, lalu pilih next.



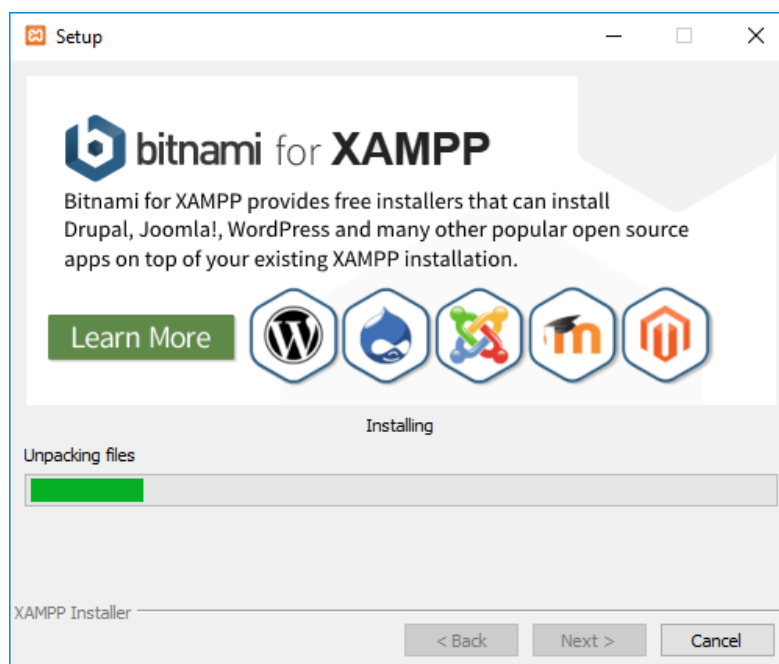
Gambar 4.5 Pilihan Folder Instalasi

6. Lalu akan tampil menu pemberitahuan apakah sudah anda yakin akan menginstall aplikasi xampp ke komputer maka pilih next.



Gambar 4.6 Pesan Pemberitahuan

7. Lalu proses instalasi akan berjalan



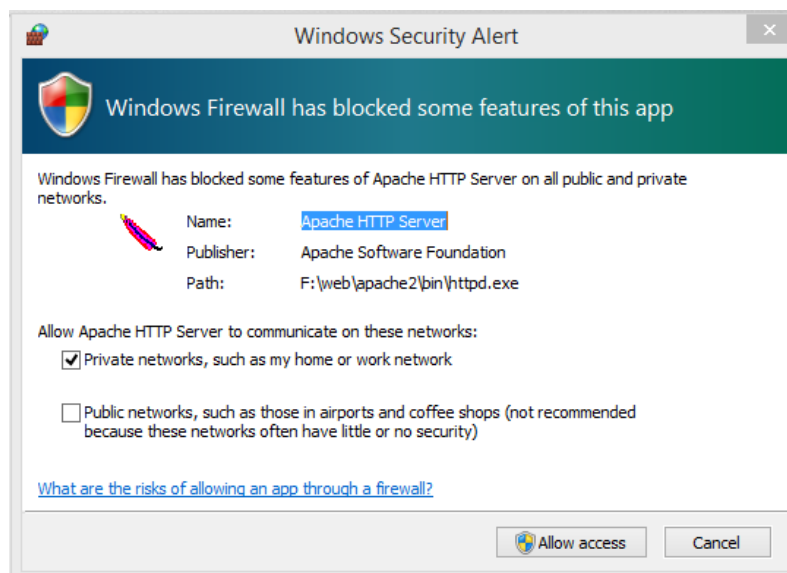
Gambar 4.7 Proses Instalasi

8. Setelah selesai proses instalasi maka secara otomatis firewall anda akan memblok akses mysql yang ada pada aplikasi ini pilih Allow access



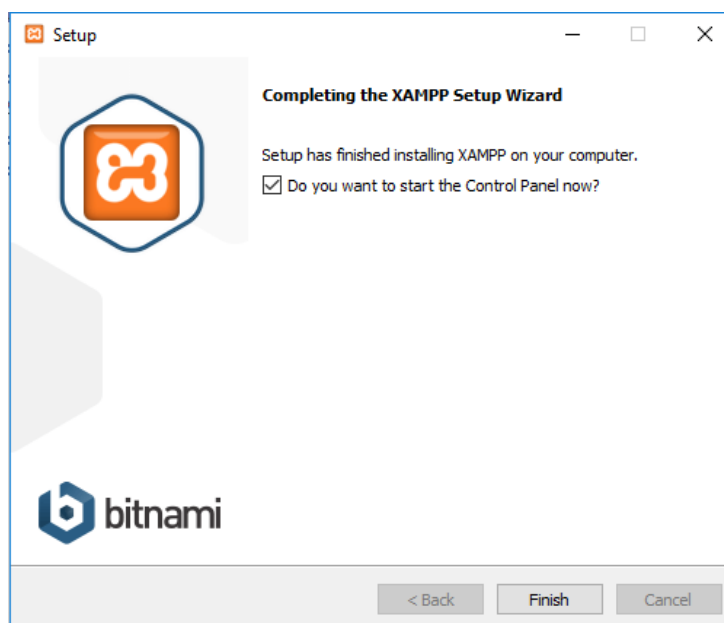
Gambar 4.8 Pesan Peringatan akses Mysql

9. Lalu akan tampil juga pesan peringatan bahwa aplikasi ini berhak mengakses sistem Apache HTTP Server yang ada pada perangkat ini lalu pilih Allow Access



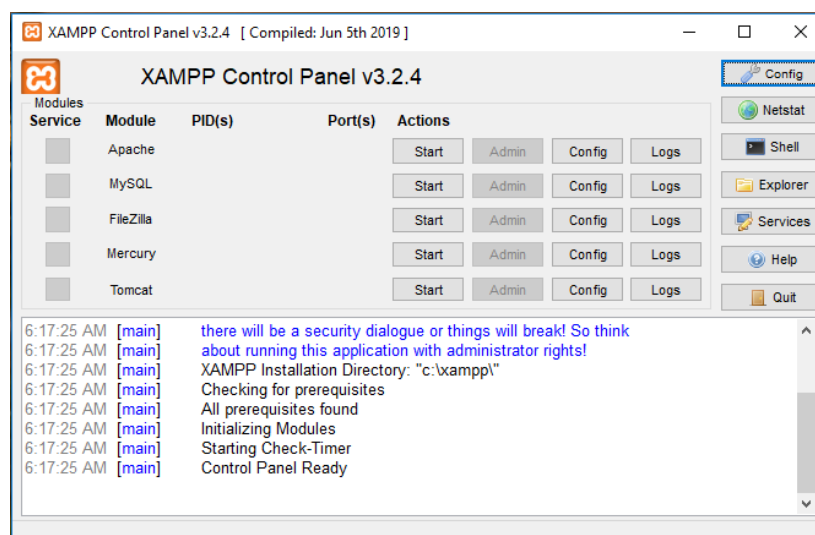
Gambar 4.9 Pesan Peringatan akses Apache HTTP Server

10. Maka akan tampil tab pemberitahuan tentang aplikasi yang telah kita install pada perangkat ini pilih finish.



Gambar 4.10 Pesan Pemberitahuan

11. Bila proses instalasi telah berjalan dengan sukses maka akan tampil dan aktifkan koneksi sistem apache dan mysql yang ada pada perangkat ini dengan klik start.



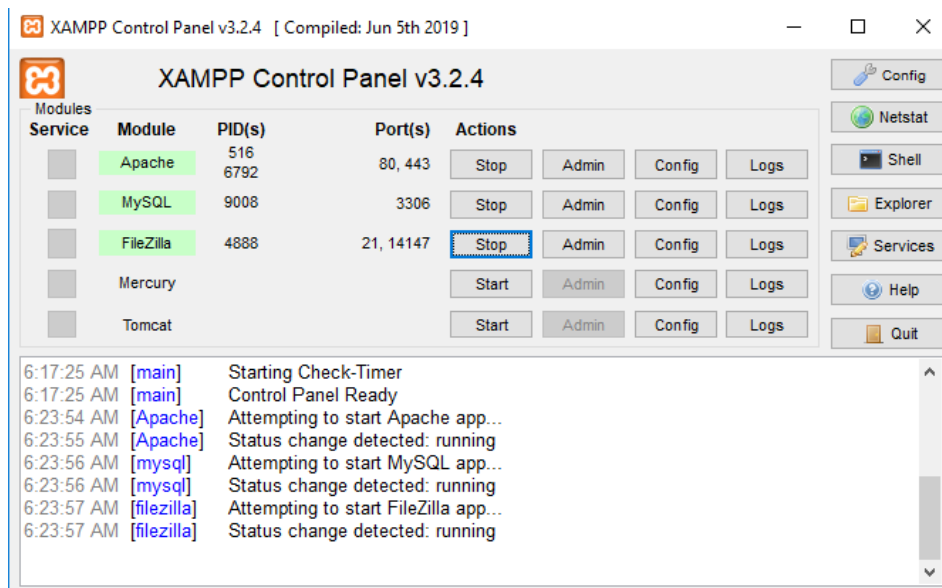
Gambar 4.11 Hasil Instalasi

4.1.3.2. Import Database

Database ini digunakan penulis sebagai media penyimpan dalam aplikasi Sistem Pendukung Keputusan dengan menggunakan bahasa pemrograman PHP.

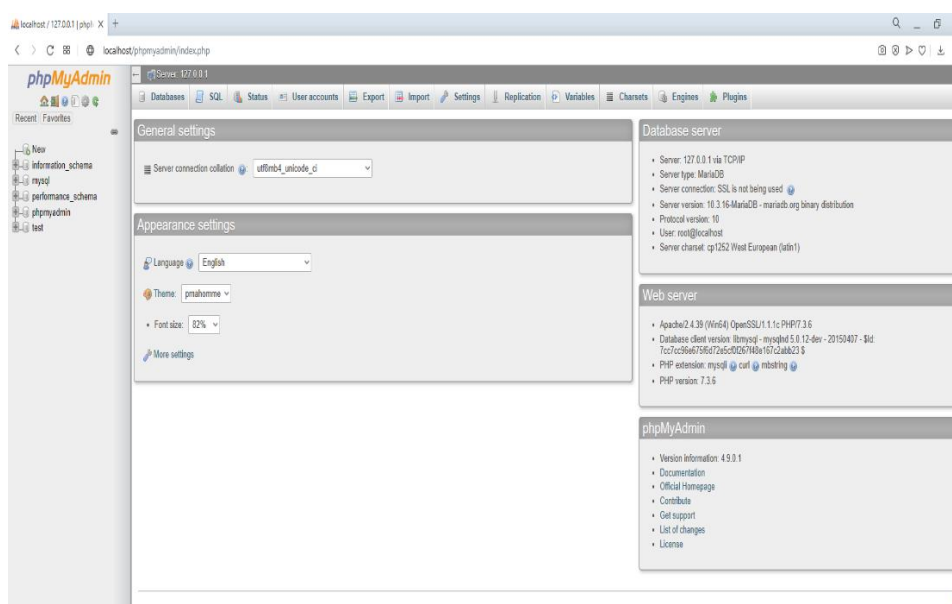
Langkah-langkah untuk import database dapat dikerjakan sebagai berikut :

1. Sebelumnya pastikan Apache dan MySQL telah berjalan dengan cara membuka aplikasi Xampp yang telah tersedia sebelumnya seperti gambar berikut :



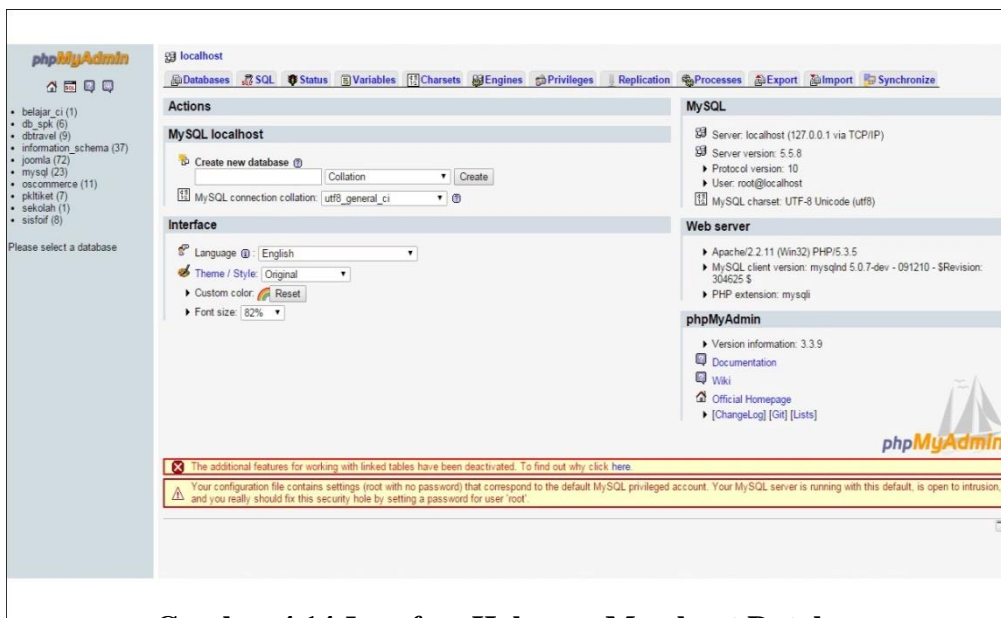
Gambar 4.12 Interface aplikasi Xampp

2. Kemudian buka browser Google Chrome dan masukkan alamat lokal. Sehingga tampil interface halaman awal server lokal dari Xampp.



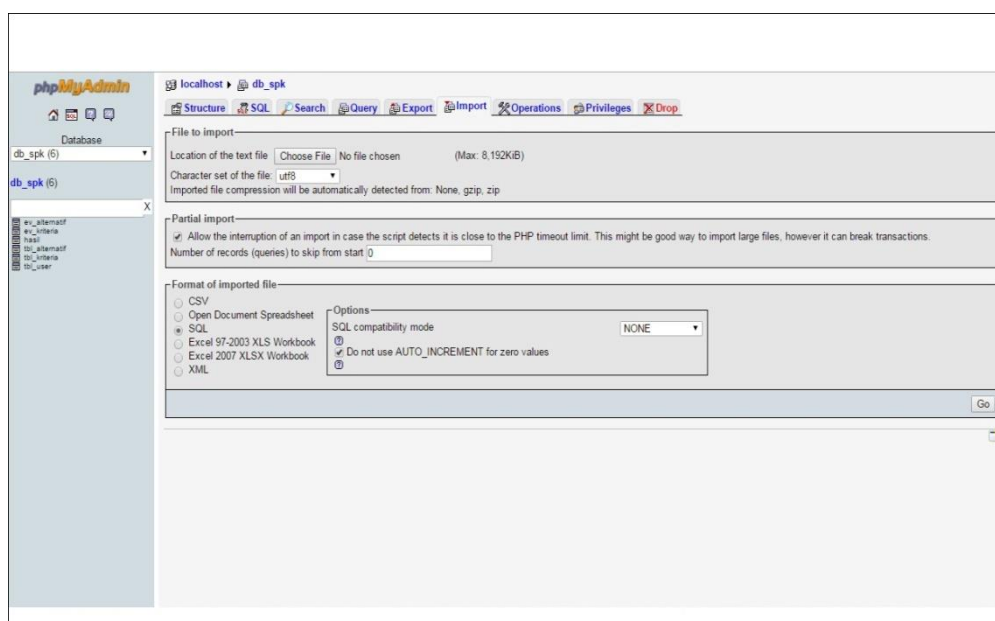
Gambar 4.13 Interface halaman awal Xampp

3. Kemudian untuk dapat menuju ke alamat lokal database maka kita perlu langsung menuju ke alamat lokal database dengan cara mengetikkan alamat localhost/phpmyadmin pada ruang alamat yang tersedia lalu masukkan nama database yang akan dibuat yang mana dalam hal ini adalah db_spk.:



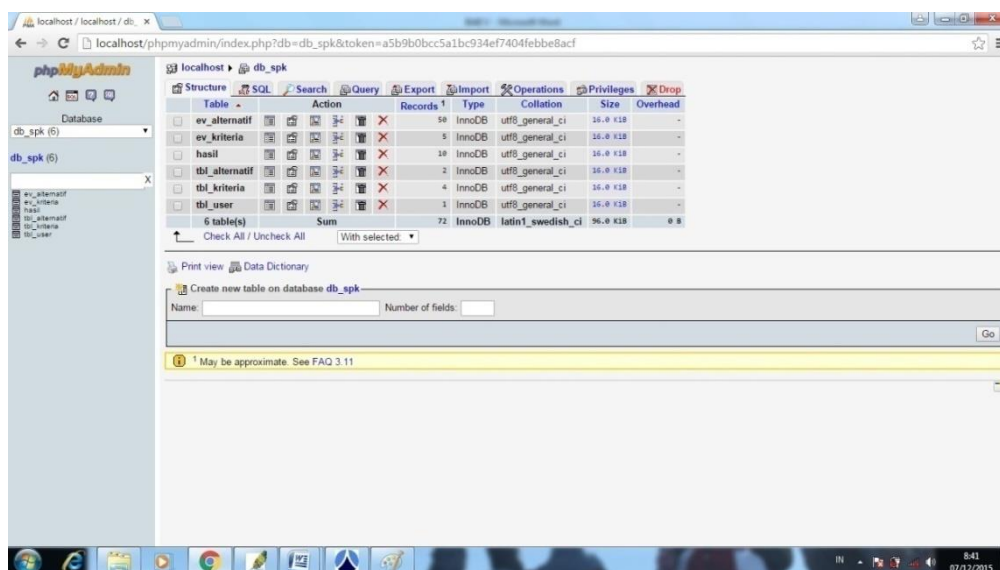
Gambar 4.14 Interface Halaman Membuat Database

4. Setelah itu, buka database yang telah dibuat. Kemudian klik menu Import untuk mengimport database yang sebelumnya telah di backup.



Gambar 4.15 Interface Halaman Import Database

5. Dan jika telah berhasil import, dapat dilihat struktur databasenya seperti gambar berikut :



Gambar 4.16 *Interface* halaman struktur database

4.2. Pengujian

Dalam hal pengujian ini banyak hal yang harus menjadi bahan pertimbangan apakah sistem yang sudah dibangun sudah dapat berjalan sesuai yang diharapkan berdasarkan metode yang digunakan atau berdasarkan susunan secara umumnya.

4.2.1 Pengujian Aplikasi

Pengujian aplikasi ini bertujuan untuk apakah hasil penelitian yang dijalankan dapat berjalan sesuai dengan yang diharapkan atau muncul sebuah permasalahan baru lagi yang belum dapat teratasi, untuk itu perlulah dibuat sebuah tabel pengujian aplikasi yang menjelaskan valid atau tidak nya proses aplikasi yang diharapkan, berikut adalah tabel pengujian aplikasi tersebut:

Tabel 4.1: Pengujian Aplikasi

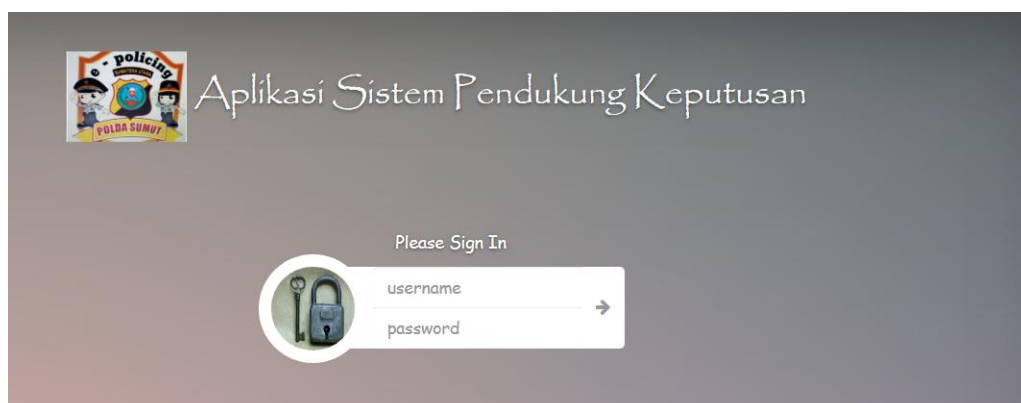
No	Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login user	Proses login berjalan lancar	Sesuai harapan	Valid
2	Proses Data kriteria	Proses data sukses dilakukan	Sesuai harapan	Valid
3	Proses Data Alternatif	Proses data sukses dilakukan	Sesuai harapan	Valid
4	Proses Perhitungan	Berjalan sesuai metode	Sesuai Harapan	Valid
5	Hasil Perhitungan	Mendapatkan Hasil Maksimal	Sesuai harapan	Valid
6	Logout User	Proses Logout Berjalan Lancar	Sesuai harapan	Valid
7	Edit User	Proses Edit user sukses	Sesuai Harapan	Valid

4.2.2 Pengujian Interface

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 dapat dilihat. Berikut ini adalah implementasi dari perancangan aplikasi Sistem Pendukung Keputusan menggunakan metode MFEP (*Multifactor Evaluation Process*).

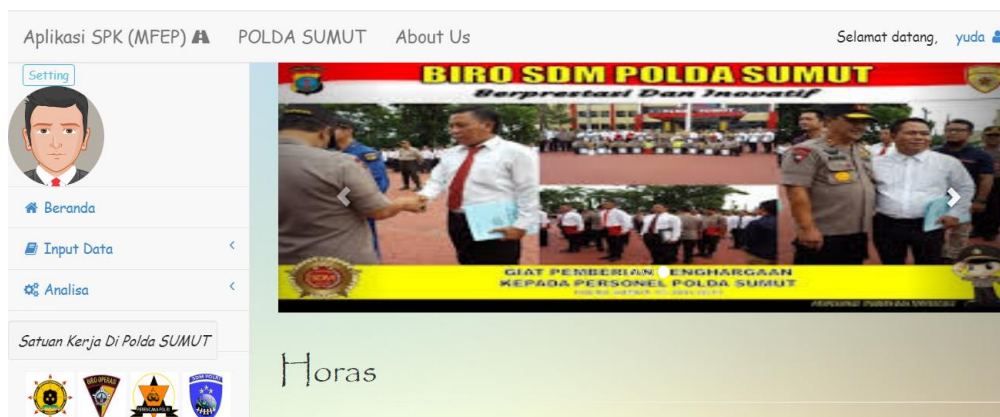
1. Halaman pertama yang muncul ketika diakses adalah halaman *login*.

Halaman *login* dapat dilihat pada gambar berikut :



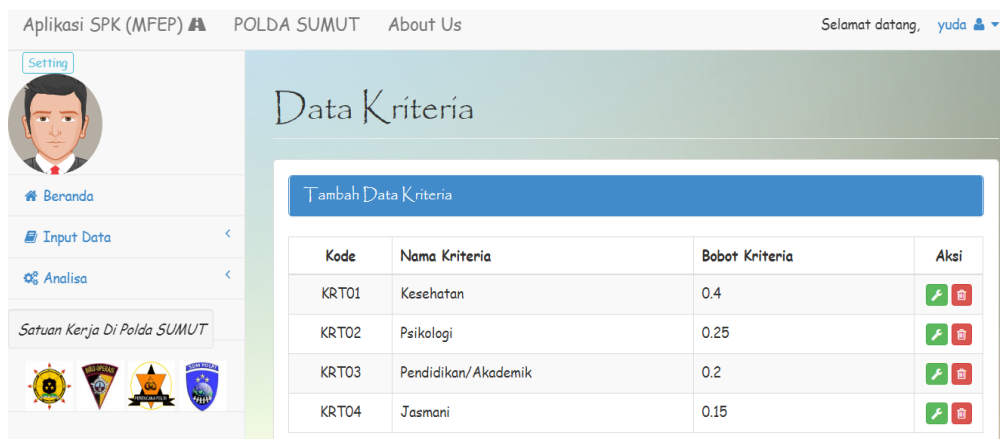
Gambar 4.17 Interface halaman login

2. Setelah berhasil *login*, maka akan tampil *interface* dari halaman utama seperti gambar berikut :



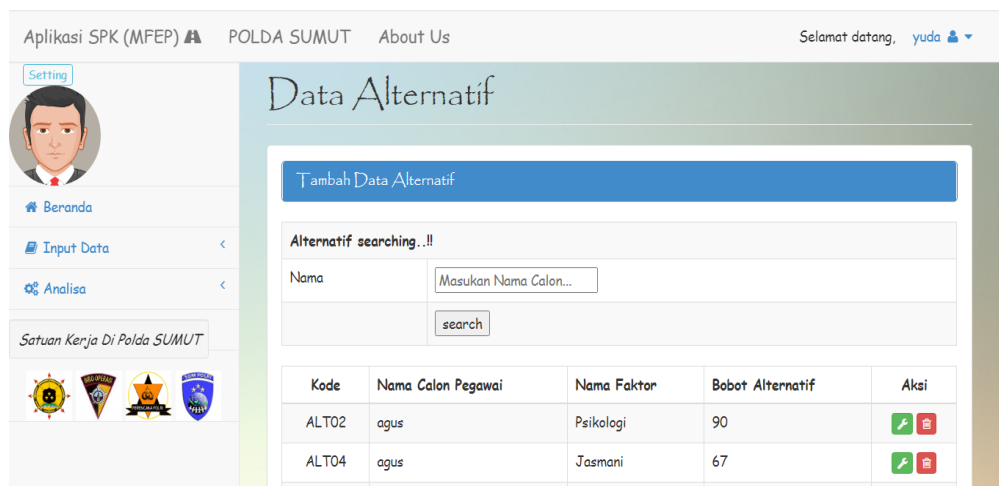
Gambar 4.18 Interface halaman utama

3. Untuk melakukan perhitungan Sistem Pendukung keputusan. Sebelumnya pengguna meng-*input*-kan data kriteria dan alternatif yang dapat dilihat pada gambar berikut :



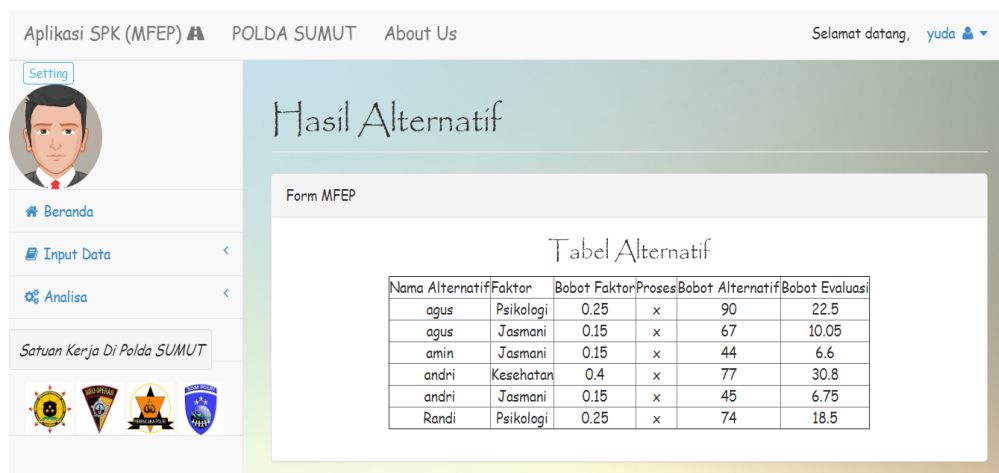
Gambar 4.19 Interface Halaman Input Kriteria

4. Untuk melakukan perhitungan Sistem Pendukung keputusan. Sebelumnya pengguna meng-*input*-kan alternatif yang dapat dilihat pada gambar berikut :



Gambar 4.20 Interface Halaman Input Alternatif

5. Tampilan ini adalah analisa proses dari metode MFEP mulai dari proses perhitungan sampai dengan proses untuk mendapatkan hasil untuk di tampilkan di menu hasil.



Gambar 4.21 Interface Analisa Hasil

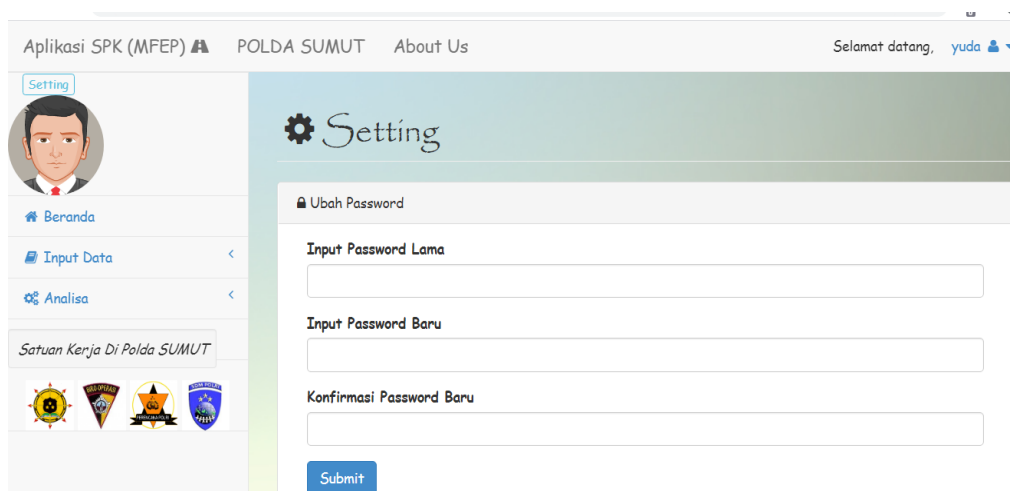
6. Dan terakhir untuk melihat hasil proses yang telah dilakukan perhitungan sebelumnya berdasarkan aspek-aspek yang telah ditentukan



ID Hasil	Nama Alternatif	Hasil
3		82.20000107586384
2		78.75000108778477
4	andri	74.90000095963478
5		64.65000103414059

Gambar 4.22 Interface Hasil Akhir

7. Dalam aplikasi ini juga terdapat fasilitas mengubah *password* pengguna. Sehingga pengguna dapat mengubah *password* sesuai keinginannya sendiri. Berikut gambarnya :



Gambar 4.23 Interface halaman ubah password

8. Dan berikut ini *interface* dari halaman tentang aplikasi (*AboutUs*). Dapat dilihat pada gambar berikut :



Gambar 4.24 Interface Halaman About

Setelah melakukan implementasi dan pengujian sistem, maka dapat disimpulkan bahwa aplikasi Sistem Pendukung Keputusan dengan metode MFEP (*Multifactor Evaluation Process*) ini berjalan sebagaimana mestinya dan sesuai dengan perancangan.

BAB V

PENUTUP

5.1 Kesimpulan

Sebagai penutup sajian pembahasan dalam penulisan dapat diambil kesimpulan-kesimpulan sekaligus memberikan saran untuk memajukan sistem yang dibuat, dengan adanya kesimpulan dan saran ini dapatlah diambil suatu perbandingan yang akhirnya dapat memberikan perbaikan-perbaikan pada masa yang akan datang.

Adapun kesimpulan yaitu sebagai berikut :

1. Aplikasi ini terbukti mampu mempermudah pihak terkait dalam melakukan proses seleksi dengan menggunakan proses *Multi factor evaluation process* untuk mengetahui informasi dan laporan kelulusan calon pegawai berdasarkan kualitas yang diharapkan.
2. Sistem pendukung keputusan yang dibangun dapat mempersingkat waktu proses seleksi yang dibutuhkan karena melakukan perhitungan yang sudah terkomputerisasi.
3. Sistem Pendukung Keputusan yang dibangun dapat menyesuaikan berdasarkan bobot tertinggi yang diharapkan atau berdasarkan suatu keahlian khusus yang diharapkan atau diutamakan untuk diterima bergabung bersama Polda Sumatera Utara.

5.2 Keterbatasan Sistem

Dari hasil yang analisa yang telah dilakukan penulis, aplikasi ini masih terdapat keterbatasan-keterbatasan tersebut adalah :

1. Sistem pendukung keputusan yang dibangun ini hanya dapat digunakan oleh pihak Polda Sumatera Utara karena berdasarkan hasil proses wawancara dengan pihak yang bersangkutan.
2. Sistem Pendukung Keputusan yang dibangun ini hanya baru bisa melakukan proses perhitungan setelah mendapatkan nilai dari proses manual yang telah dilakukan calon pegawai Polda Sumatera Utara sebelumnya bersama pihak Polda Sumatera Utara.
3. Sistem Pendukung Keputusan yang dibangun ini belum memiliki keamanan data yang maksimal apabila dilakukan proses hosting dikarenakan kesensitifan data kepolisian yang tidak boleh di publikasikan.

5.3 Saran

Aplikasi Sistem Pendukung Keputusan ini masih membutuhkan beberapa pengembangan untuk menjadikannya lebih sempurna dan mempunyai hasil yang lebih akurat.

Beberapa aspek yang perlu dikembangkan antara lain :

- 1 Sistem ini hanya dapat dijalankan pada instansi pihak Polda Sumatera Utara, karena berdasarkan hasil wawancara dengan pihak Polda Sumatera Utara.
- 2 Sistem ini baru dapat berjalan apabila sudah mendapatkan data proses seleksi manual dari pihak Polda Sumatera Utara.
- 3 Sistem ini belum memiliki keamanan akses data yang maksimal.
- 4 Sistem ini baru dapat dijalankan di aplikasi *web server*.

Diharapkan adanya peneliti lain yang mengembangkan sistem pendukung keputusan ini dengan metode-metode yang lain agar dapat melihat perbandingan hasilnya.

Saran :

- 1 Untuk dapat digunakan di instansi Polda daerah lain maka harus dilakukan wawancara secara langsung kepada pihak Polda terkait berdasarkan permasalahan yang dihadapi daerah tersebut.
- 2 Untuk pengembangan selanjutnya diharapkan dibuatkan aplikasi yang secara otomatis menseleksi calon pegawai terkait.
- 3 Untuk masalah keamanan penulis mengharapkan untuk pengembangan selanjutnya menggunakan keamanan berbasis kriptografi karena data-data yang disimpan ini adalah data-data yang sangat penting.
- 4 Untuk pengembangan selanjutnya penulis mengharapkan agar aplikasi ini dapat dijalankan melalui media smartphone atau dapat dijalankan dengan aplikasi mobile untuk lebih memudahkan inputan data.

..

DAFTAR PUSTAKA

- Abdul Rohman Amat Ramat, Alvino Octaviano. (2016). Aplikasi Pemesanan Tiket Bus Berbasis Web (Studi Kasus Pada PO. Harapan Jaya). Vol. 1, No. 1. ISSN: 2541-1004.
- Ali Subhan Afrizal. (2017). Rancang Bangun Aplikasi Pembelajaran Dasar Pemograman Berbasis Mobile Phone. Vol. VI. No. 1. ISSN: 2407-2192.
- Bay Haqi. (2017). Membuat Aplikasi Antrean Dengan Java NetBeans IDE 8.0.2 dan Database MySQL. Jakarta:PT Elex Media Komputindo.
- Dahmiri, D., Yuliusman, Y., Amin, M., Musyayaddah, M., & Arsyadt, A. (2019). Pelatihan Kewirausahaan Melalui Usaha Kue Dengan Bahan Dasar Ubi Kayu. Jurnal Karya Abdi Masyarakat, 3(2), 133-139.
- Efmi Mariyana. (2018). Pemanfaat Android Dalam Perancangan Aplikasi Kumpulan Doa. (ISSN: 2502-096X), 54-67.
- Fachri, B., & Harahap, F. H. (2020). Simulasi Penggunaan Intrusion Detection System (IDS) Sebagai Keamanan Jaringan dan Komputer. Jurnal Media Informatika Budidarma, 4(2), 413-420.
- Juansyah Andi. (2015). Pembangunan Aplikasi Child Tracker Berbasis Assited-Global Positioning System (A-GPS) dengan Platform Android. Vol. 1 No. 1 Agustus. ISSN: 2089-9033.
- Ganda Yoga Swara, Yunes Pebriadi. (2016). Rekayasa Perangkat Lunak Pemesanan Tiket Bioskop Berbasis Web. 4 No 2 (ISSN: 2338-2724), 27-39.
- Laila Septiana. (2016). Perancangan Sistem Pakar Diagnosa Penyakit ISPA Dengan Metode Certainty Factor Berbasis Factor. Vol. 13. No. 2. ISSN: 1978-2136.
- Jonathan Tiku Ali, Andi Patombongi. (2016). Perancangan Game Edukasi Pembelajaran Membaca Berbasis Andorid. 1 No 1 (ISNN: 2502-5899), 1-8.
- Nico Alvio Maiyedra. (2018). Perancangan Sistem Pakar Mendiagnosa Penyakit Kulit Pada Anak Dengan Menggunakan Metode Backward Chaining. 6 No 2 (ISSN: 2338-1523).
- Putra, P. H., Zarlis, M., & Mawengkang, H. (2020). Analysis of the effect number input and hidden layer variations on the addition kohonen algorithm to backpropagation method. In IOP Conference Series: Materials Science and Engineering (Vol. 725, No. 1, p. 012096). IOP Publishing.
- Rachmat Agusli, Muchmad Iqbal, Fendi Saputra. (2020). Sistem Pakar Diagnosa Penyakit Pada Ibu Hamil Dengan Metode Certainty Faktor Berbasis Web. Vol. 2. No. 1.
- Sapta Irwan Permana, Yusuf Sumaryana. 2018. Sistem Pakar Untuk Mendiagnosa Penyakit Kulit Dengan Metode Forward Chaining. Vol. 1, No. 1. ISSN: 2613-9138.
- Sari, R. M., & Tasril, V. (2020). Prediksi Jumlah APBD Kota Payakumbuh dengan metode K-Means. Jurnal Ipteks Terapan, 14(1), 45-50.
- Sibagariang Swono. 2015. Sistem Pakar Diagnisa Penyakit Sapi Dengan Metode Certainty Factor Berbasis Android. Vol. IV. NO. 2. ISSN: 2337-3601.

Sudirman. (2016). Analisis Komunikasi Data Dengan XML Dan JSON Pada Webservice. 1 No 2 (ISSN: 2502-7131), 1-6.

Teuku Feraldy Ramadhani, Iskandar Fitri, Endah Tri Esti Handayani. (2018). Sistem Pakar Diagnosa Penyakit ISPA Berbasis Web Dengan Metode Forward Cahining. Vol. 5. No. 2. ISSN: 2541-6448.

Yossi Octavina, Abdul Fadlil. (2014). Sistem Pakar Untuk Mendiagnosa Penyakit Pada Saluran Pernafasan Dan Paru Menggunakan Metode Certainty. Vol. 2. No. 2. ISSN: 2338-5197.