

**TESIS**

**GABUNGAN METODE MODIFIED DELPHI-AHP-PROMETHEE  
UNTUK SELEKSI PESERTA PELATIHAN (STUDI KASUS: BALAI  
LATIHAN KERJA BANTUL)**

***COMBINATION OF MODIFIED DELPHI-AHP-PROMETHEE FOR  
SELECTION OF TRAINING PARTICIPANTS (CASE STUDY: BALAI  
LATIHAN KERJA BANTUL)***



**DEWI ANISA ISTIQOMAH  
15/388466/PPA/04905**

**PROGRAM STUDI S2 ILMU KOMPUTER  
JURUSAN ILMU KOMPUTER DAN ELEKTRONIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS GADJAH MADA  
YOGYAKARTA**

**2018**

## **TESIS**

**GABUNGAN METODE MODIFIED DELPHI-AHP-PROMETHEE  
UNTUK SELEKSI PESERTA PELATIHAN (STUDI KASUS: BALAI  
LATIHAN KERJA BANTUL)**

***COMBINATION OF MODIFIED DELPHI-AHP-PROMETHEE FOR  
SELECTION OF TRAINING PARTICIPANTS (CASE STUDY: BALAI  
LATIHAN KERJA BANTUL)***

Diajukan untuk memenuhi salah satu syarat memperoleh derajat  
*Master of Computer Science*



**DEWI ANISA ISTIQOMAH  
15/388466/PPA/04905**

**PROGRAM STUDI S2 ILMU KOMPUTER  
JURUSAN ILMU KOMPUTER DAN ELEKTRONIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS GADJAH MADA  
YOGYAKARTA**

**2018**

## **HALAMAN PENGESAHAN**

### **TESIS**

#### **GABUNGAN METODE MODIFIED DELPHI-AHP-PROMETHEE UNTUK SELEKSI PESERTA PELATIHAN (STUDI KASUS: BALAI LATIHAN KERJA BANTUL)**

Telah dipersiapkan dan disusun oleh

**DEWI ANISA ISTIQOMAH  
15/388466/PPA/04905**

Telah dipertahankan di depan Tim Penguji  
pada tanggal <masukan tanggal ujian>

Susunan Tim Penguji

Dr. Sigit Priyanta, S.Si., M.Kom.      <Masukan nama penguji>  
Pembimbing I                                  <Masukan posisi Dosen>

Dr. Yohanes Suyanto, M.I.Kom.      <Masukan nama Penguji>  
Pembimbing II                                <Masukan posisi Dosen>

<Masukan nama Penguji>  
<Masukan posisi Dosen>

Tesis ini telah diterima sebagai salah satu persyaratan  
Untuk memperoleh gelar *Master of Computer Science*  
Tanggal <masukan tanggal ujian>

Dr. Tri Kuntoro Priyambodo, M.Sc.  
Pengelola Program Studi S2 Ilmu Komputer

## **PERNYATAAN**

Dengan ini saya menyatakan bahwa Tesis ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Master di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, <Tanggal, bulan, tahun>

DEWI ANISA ISTIQOMAH

*Karya ini penulis persembahkan untuk orang tua, kakak-kakak, adik-adik, dan seluruh keluarga besar yang selalu memberikan harapan bagi penulis, juga segenap rekan-rekan serta pembaca sekalian.*

Katakanlah: “Kalau sekiranya lautan menjadi tinta untuk (menulis) kalimat-kalimat TuhanKu, sungguh habislah lautan itu sebelum habis (ditulis) kalimat kalimat TuhanKu, meskipun Kami datangkan tambahan sebanyak itu (pula)”.  
(Q.S. Al-Kahfi: 109)

## PRAKATA

Puji syukur kehadirat Allah SWT atas limpahan rahmat, karunia, serta petunjuk-Nya sehingga tugas akhir berupa penyusunan tesis ini telah terselesaikan dengan baik.

Dalam penyusunan tugas akhir ini penulis telah banyak mendapatkan arahan, bantuan, serta dukungan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Prof. Dr. Subanar selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Gadjah Mada.
2. Prof. Dr. Dra. Sri Wahyuni, M.S. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Gadjah Mada.
3. Drs. G.P. Dalijo. Dipl.Comp. selaku Ketua Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Gadjah Mada.
4. Drs. Sri Mulyana, M.Kom selaku Dosen Wali Akademik penulis.
5. Drs. Yohanes Suyanto, M.Kom selaku Dosen Pembimbing yang telah berkenan meluangkan waktu, dan pikiran dalam penyelesaian tugas akhir ini.
6. Nur Rokhman, S.Si, M.Kom dan Aina Musdholifah, S.Si selaku tim penguji.
7. Dr. Pekik Purwantoro, M.Si yang telah menyumbangkan template naskah skripsi FMIPA UGM berbasis LaTeX dan OpenOffice untuk menyusun naskah skripsi ini.
8. Segenap Dosen dan civitas akademik di lingkungan Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada.
9. Kedua Orang Tua, kakak dan adik yang selalu memberikan arahan selama belajar dan menyelesaikan tugas akhir ini.

Akhir kata penulis berharap semoga tesis ini dapat memberikan manfaat bagi kita semua, terutama bagi perkembangan ilmu pengetahuan serta perkembangan Ilmu Komputer dan Teknologi Informasi.

Yogyakarta, <Tanggal, bulan, tahun>

Penulis

## DAFTAR ISI

<b>PRAKATA.....</b>	iv
<b>DAFTAR SINGKATAN.....</b>	x
<b>INTISARI.....</b>	xi
<b>ABSTRACT.....</b>	xii
<b>BAB I</b>	
<b>PENDAHULUAN.....</b>	13
1.1 Latar Belakang .....	13
1.2 Rumusan Masalah.....	15
1.3 Batasan Masalah.....	15
1.4 Tujuan Penelitian.....	16
1.5 Manfaat Penelitian.....	16
1.6 Keaslian Penelitian.....	16
1.7 Metodologi Penelitian.....	17
1.8 Sistematika Penulisan.....	19
<b>BAB II</b>	
<b>TINJAUAN PUSTAKA.....</b>	21
<b>BAB III</b>	
<b>LANDASAN TEORI.....</b>	36
3.1 Metode MADM (Multiple Attribute Decision-Making) .....	36
3.2 Metode Modified Delphi .....	38
3.3 Metode AHP (Analytic Hierarchy Process) .....	40
3.4 Tahapan Dalam AHP .....	42
3.5 PROMETHEE (Preference Ranking Organization Method For Enrichment Evaluation).....	44
<b>BAB IV</b>	
<b>ANALISIS DAN RANCANGAN SISTEM.....</b>	47
4.1 Analisis Data .....	47
4.2 Pertimbangan Penggunaan Metode Modified Delphi-AHP-PROMETHEE.....	53
4.3 Analisis Kebutuhan Sistem .....	54
4.4 Perancangan Sistem.....	56
4.5 Perancangan Basis Data.....	65
4.6 Perancangan Antarmuka.....	79
4.7 Perancangan Pemodelan.....	81
4.8 Rangkaian Penggunaan Pemodelan.....	91
<b>BAB V</b>	
<b>IMPLEMENTASI.....</b>	112
5.1 Implementasi Metode Modified Delphi.....	112

5.2 Implementasi Metode AHP.....	130
5.3 Implementasi Metode PROMETHEE.....	136
<b>BAB VI</b>	
<b>HASIL PENELITIAN DAN PEMBAHASAN.....</b>	<b>149</b>
6.1 Perbandingan Metode AHP-PROMETHEE Dan Modified Delphi-AHP-PROMETHEE.....	149
6.2 Perbandingan Metode Modified Delphi-AHP-PROMETHEE Dan Modified Delphi-AHP-TOPSIS.....	156
<b>BAB VII</b>	
<b>KESIMPULAN DAN SARAN.....</b>	<b>159</b>
7.1 Kesimpulan.....	159
7.2 Saran.....	159

## DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka.....	27
Tabel 3.1 Skala untuk Perbandingan Kuantitatif Alternatif (Bhushan dan Rai, 2004).....	43
Tabel 4.1 Identitas pendaftar.....	47
Tabel 4.2 Hasil penilaian pendaftar.....	49
Tabel 4.3 Kriteria seleksi personil dari studi literatur.....	51
Tabel 4.4 Deskripsi use case .....	54
Tabel 4.5 Struktur tabel users.....	68
Tabel 4.6 Struktur tabel uploads.....	68
Tabel 4.7 Struktur tabel educations.....	69
Tabel 4.8 Struktur tabel educational_background.....	70
Tabel 4.9 Struktur tabel courses.....	70
Tabel 4.10 Struktur tabel course_experience.....	71
Tabel 4.11 Struktur tabel registrants.....	71
Tabel 4.12 Struktur tabel roles.....	72
Tabel 4.13 Struktur tabel role_user.....	72
Tabel 4.14 Struktur tabel permissions.....	72
Tabel 4.15 Struktur tabel permission_role .....	73
Tabel 4.16 Struktur tabel vocational.....	73
Tabel 4.17 Struktur tabel sub_vocational.....	74
Tabel 4.18 Struktur tabel registrations.....	74
Tabel 4.19 Struktur tabel selection_schedules.....	75
Tabel 4.20 Struktur tabel selections.....	76
Tabel 4.21 Struktur tabel criterias.....	77
Tabel 4.22 Struktur tabel choice.....	78
Tabel 4.23 Struktur tabel result_selection.....	79
Tabel 4.24 Struktur tabel pairwise_comparisons.....	79
Tabel 4.25 Fungsi Preferensi (Brans dan Mareschal, 2005).....	85
Tabel 4.26 Hasil kesesuaian kriteria pada Delphi putaran pertama.....	91
Tabel 4.27 Hasil kriteria pada Delphi putaran kedua.....	93
Tabel 4.28 Matriks perbandingan berpasangan kriteria.....	96
Tabel 4.29 Normalisasi matriks perbandingan berpasangan kriteria.....	96
Tabel 4.30 Hasil bobot kriteria.....	96
Tabel 4.31 Matriks perbandingan berpasangan subkriteria untuk kriteria FI.....	97
Tabel 4.32 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FI.....	97
Tabel 4.33 Hasil bobot subkriteria untuk kriteria FI.....	98
Tabel 4.34 Matriks perbandingan berpasangan subkriteria untuk kriteria FP.....	98
Tabel 4.35 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FP.....	99
Tabel 4.36 Hasil bobot subkriteria untuk kriteria FP.....	99

Tabel 4.37 Matriks perbandingan berpasangan subkriteria untuk kriteria FE.....	100
Tabel 4.38 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FE.....	100
Tabel 4.39 Hasil bobot subkriteria untuk kriteria FE.....	101
Tabel 4.40 Bobot global kriteria.....	101
Tabel 4.41 Nilai peserta seleksi pelatihan.....	103
Tabel 4.42 Selisih (1,2) untuk setiap kriteria.....	104
Tabel 4.43 Fungsi preferensi.....	105
Tabel 4.44 Indeks preferensi.....	106
Tabel 4.45 Derajat preferensi d(1,2) untuk setiap kriteria.....	107
Tabel 4.46 Hasil Leaving flow dan entering flow.....	108
Tabel 4.47 Hasil net flow.....	110
Tabel 4.48 Peringkat lengkap PROMETHEE II.....	111
Tabel 5.1 Potongan source code implementasi pengelolaan kriteria dari studi literatur.....	113
Tabel 5.2 Potongan source code implementasi kuesioner kriteria.....	116
Tabel 5.3 Potongan source code implementasi hasil Delphi putaran pertama.....	120
Tabel 5.4 Potongan source code implementasi Delphi putaran kedua.....	122
Tabel 5.5 Potongan implementasi hierarki kriteria.....	127
Tabel 5.6 Potongan source code implementasi metode AHP.....	131
Tabel 5.7 Potongan source code implementasi tambah nilai peserta seleksi pelatihan.....	137
Tabel 5.8 Impelemensi pengelolaan fungsi preferensi.....	140
Tabel 5.9 Potongan source code implementasi penilaian peserta seleksi pelatihan .....	143
Tabel 5.10 Potongan source code implementasi hasil peringkat peserta seleksi .....	148
Tabel 6.1 Perbandingan kriteria/subkriteria berdasarkan metode Modified Delphi dan pembuat keputusan.....	149
Tabel 6.2 Hasil perhitungan bobot dengan AHP untuk kriteria berdasarkan pembuat keputusan.....	152
Tabel 6.3 Hasil perhitungan bobot dengan AHP untuk kriteria/subkriteria berdasarkan Modified Delphi.....	152
Tabel 6.4 Perbandingan hasil AHP-PROMETHEE dan Modified Delphi-AHP-PROMETHEE.....	154
Tabel 6.5 Perbandingan hasil Modified Delphi-AHP-PROMETHEE dan Modified Delphi-AHP-TOPSIS.....	156

## DAFTAR GAMBAR

Gambar 3.1 Sistem Hierarki untuk MADM (Dubois dan Prade, 1980).....	38
Gambar 3.2 Struktur Umum Hierarki AHP (Bhushan dan Rai, 2004).....	42
Gambar 3.3 Format untuk Perbandingan Berpasangan (Bhushan dan Rai, 2004).....	42
Gambar 4.1 Use case diagram sistem.....	57
Gambar 4.2 Activity diagram mengelola kriteria pustaka.....	58
Gambar 4.3 Activity diagram mengisi kuesioner kriteria.....	59
Gambar 4.4 Activity diagram melihat hasil kriteria tahap 1.....	59
Gambar 4.5 Activity diagram mengelola kriteria tahap 2 .....	60
Gambar 4.6 Activity diagram mengelola hierarki kriteria.....	61
Gambar 4.7 Activity diagram mengelola bobot kriteria .....	62
Gambar 4.8 Activity diagram mengelola tipe preferensi.....	63
Gambar 4.9 Activity diagram mengelola nilai alternatif.....	64
Gambar 4.10 Activity diagram melihat hasil ranking.....	65
Gambar 4.11 Entity Relationship Diagram (ERD).....	67
Gambar 4.12 Implementasi basis data.....	69
Gambar 4.13 Rancangan antarmuka halaman utama .....	80
Gambar 4.14 Proses model keputusan seleksi peserta pelatihan.....	83
Gambar 4.15 Flowchart model Modified Delphi.....	88
Gambar 4.16 Flowchart model AHP.....	89
Gambar 4.17 Flowchart model PROMETHEE.....	90
Gambar 4.18 Hierarki kriteria seleksi peserta pelatihan.....	95
Gambar 5.1 Implementasi halaman pengelolaan kriteria dari studi literatur.....	112
Gambar 5.2 Implementasi halaman kuesioner kriteria .....	115
Gambar 5.3 Implementasi hasil isian kuesioner kriteria.....	116
Gambar 5.4 Implementasi ahli mengisi kuesioner kriteria.....	119
Gambar 5.5 Implementasi halaman hasil Delphi putaran pertama.....	119
Gambar 5.6 Halaman implementasi Delphi putaran kedua.....	122
Gambar 5.7 Implementasi saat proses diskusi menentukan kriteria.....	126
Gambar 5.8 Implementasi halaman hierarki kriteria.....	126
Gambar 5.9 Implementasi halaman perbandingan berpasangan.....	130
Gambar 5.10 Implementasi halaman bobot.....	131
Gambar 5.11 Implementasi halaman tambah nilai peserta seleksi pelatihan.....	137
Gambar 5.12 Implementasi halaman pengelolaan fungsi preferensi.....	140
Gambar 5.13 Implementasi halaman penilaian peserta seleksi pelatihan.....	142
Gambar 5.14 Implementasi halaman hasil peringkat peserta seleksi .....	148
Gambar 6.1 Hierarki kriteria berdasarkan pembuat keputusan.....	151

## **DAFTAR SINGKATAN**

KB	: Kesan baik
KS	: Kesungguhan
PO	: Potensi
PE	: Pertimbangan ekonomi
KO	: Komitmen
PD	: Percaya diri
KK	: Keterampilan komunikasi
PM	: Penampilan
PK	: Pertimbangan keluarga
ME	: Mental
MO	: Motivasi
US	: Usia
PT	: Pendidikan terakhir
SI	: Sikap
KJ	: Kejujuran
KT	: Keterampilan teknis
PG	: Pengetahuan
RK	: Rekomendasi
RS	: Rencana setelah selesai pelatihan
PP	: Pengalaman pelatihan
IK	: Intensitas keikutsertaan pelatihan di BLK Bantul
FI	: Faktor intelektual
FP	: Faktor personal
FE	: Faktor eksternal

## **INTISARI**

### **GABUNGAN METODE MODIFIED DELPHI-AHP-PROMETHEE UNTUK SELEKSI PESERTA PELATIHAN (STUDI KASUS: BALAI LATIHAN KERJA BANTUL)**

DEWI ANISA ISTIQOMAH  
15/388466/PPA/04905

Contoh untuk Intisari. Pada umumnya sistem perangkat lunak terdiri dari beberapa *concern*, premis dari masalah ini adalah sebaran *concern*, di mana kebutuhan rancangan tertentu cenderung memotong-melintasi grup inti fungsional modul. Teknik orientasi-objek yang menerapkan *concern* tersebut cenderung menghasilkan kode yang tersebar, daya baca yang sulit, serta susah untuk dikembangkan. Metodologi baru, *aspect-oriented programming* (AOP), memberikan fasilitas modularisasi pemotong-lintasan/*cross-cutting concern*. Dengan menggunakan AOP, terdapat cara untuk membuat penerapan sistem yang lebih mudah untuk dirancang, dipahami, dan dipelihara. Lebih jauh lagi, AOP menjanjikan produktivitas yang lebih tinggi, peningkatan kualitas, dan kemampuan lebih baik untuk menambahkan *feature* baru.

Contoh paragraf kedua. AspectJ adalah bahasa pemrograman yang digunakan secara luas untuk menerapkan program-program berorientasi aspek di Java. Namun demikian, AspectJ masih belum memiliki bahasa pemodelan yang dapat memenuhi perancangan program berorientasi aspek. Aspect Oriented Design Model (AODM), sebagai sebuah model perancangan baru pada pengembangan program dalam AspectJ, hanya memperluas konsep-konsep UML (Unified Modeling Language) yang telah ada dengan menggunakan mekanisme perluasan UML untuk memberikan konsep orientasi-aspek yang ada di dalam AspectJ. AODM menyediakan spesifikasi model rancangan orientasi-aspek untuk ditransformasikan menjadi model rancangan UML biasa.

## **ABSTRACT**

### **COMBINATION OF MODIFIED DELPHI-AHP-PROMETHEE FOR SELECTION OF TRAINING PARTICIPANTS (CASE STUDY: BALAI LATIHAN KERJA BANTUL)**

DEWI ANISA ISTIQOMAH  
15/388466/PPA/04905

Example of Abstract. Most software systems consist several concerns, the premise of such thing is separation of concerns, where certain design requirements tend to cut across group of core functional modules. Object-oriented techniques for implementing such concerns result in systems that are invasive to implement, tough to understand, and difficult to evolve. The new aspect-oriented programming (AOP) methodology facilities modularization of crosscutting concerns. Using AOP, there is a way to create implementations that are easier to design, understand, and maintain. Further, AOP promises higher productivity, improved quality, and better ability to implement newer features.

Second paragraf. AspectJ is a well-established programming language that is widely used to implement aspect-oriented programs in Java. However, there is no modeling language available for the design aspect oriented programs in AspectJ. Aspect Oriented Design Model (AODM), as a new design model for development of AspectJ programs, extends existing UML (Unified Modeling Language) concepts using standard UML extension mechanisms to provide aspect-oriented concepts as in AspectJ. The AODM specifies how an aspect-oriented design model maybe transformed into an ordinary UML design model.

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pelatihan dibutuhkan untuk mempersiapkan tenaga kerja yang profesional, berkualitas dan berkompetensi sesuai dengan kebutuhan pembangunan serta pasar kerja. Faktor-faktor yang menunjang efektivitas pelatihan, yaitu materi atau isi pelatihan, metode pelatihan, pelatih/instruktur, peserta pelatihan, sarana pelatihan dan evaluasi pelatihan (Rivai dan Sagala, 2014). Peserta pelatihan merupakan salah satu faktor yang menunjang efektivitas pelatihan, sehingga dibutuhkan seleksi peserta pelatihan. Hal ini guna mendapatkan peserta pelatihan yang sesuai dengan tujuan dan sasaran. Tahapan dalam proses seleksi yaitu pengisian form aplikasi, pengujian/*testing*, wawancara, pengecekan referensi dan pembuatan keputusan seleksi (Noe dkk., 2012). Tahapan tersebut juga dilakukan dalam seleksi peserta pelatihan Balai Latihan Kerja (BLK).

BLK merupakan lembaga pelatihan kerja milik pemerintah yang mempunyai maksud dan tujuan dalam upaya mengurangi pengangguran. Seleksi peserta pelatihan BLK Bantul masih manual. Hal ini menimbulkan beberapa permasalahan, yaitu kurang efektif dan efisien dalam pengelolaan data. Data peserta seleksi harus diinputkan oleh administrator ke dalam *spreadsheet* berdasarkan data yang diisikan peserta seleksi dalam formulir pendaftaran. Data yang belum tertata dengan baik membuka peluang bagi seseorang untuk dapat mengikuti pelatihan berulang-ulang.

Penilaian pemilihan yang dilakukan belum objektif dan transparan karena penilaian tidak mematuhi kriteria seleksi dan belum mempertimbangkan pembobotan kriteria. Peserta seleksi yang mempunyai hubungan kekerabatan dengan pegawai BLK Bantul mempunyai peluang yang lebih besar untuk diterima. Hal ini memungkinkan adanya peserta yang mendaftar berulang kali

tetapi tidak juga diterima karena hasil nilainya standar dan tidak mempunyai hubungan kekerabatan. Maka dari itu, BLK Bantul memerlukan sistem yang mampu membantu pembuat keputusan dalam pemilihan peserta pelatihan berdasarkan kriteria-kriteria seleksi.

Dalam beberapa literatur untuk memecahkan masalah seleksi personil digunakan metode *Multiple Attibute Decision-Making* (MADM). Metode MADM yang telah digunakan dalam masalah seleksi personil, antara lain metode SAW (Afshari dkk., 2010), AHP (Gibney dan Shang, 2007), AHP-Dynamic Programming (Özdemir, 2013), AHP-ROC-SAW (Sudipa dan Hartati, 2017), AHP-TOPSIS (Efendi dkk., 2017), *fuzzy AHP-Yager's Weighted Goals* (Güngör dkk., 2009), Geometric Mean-SAW, WPM, AHP, TOPSIS (Kumar dkk., 2013), VIKOR-Information Entropy Weight (El-santawy, 2012), modified fuzzy VIKOR-Worst Case (Alguliyev dkk., 2015), *fuzzy* TOPSIS (Kelemenis dan Askounis, 2010), PROMETHEE (ÖZTÜRK, 2013), *fuzzy* AHP-fuzzy ELECTRE (Rouyendegh dan Erkan, 2013), *fuzzy* AHP-fuzzy VIKOR (Salehi, 2016) dan AHP-PROMETHEE (Bogdanovic dan Miletic, 2014). Dari penelitian tersebut, dapat disimpulkan bahwa penentuan kriteria merupakan langkah krusial yang dapat menentukan kesesuaian hasil akhir.

Penelitian ini menggabungkan metode *Modified Delphi-AHP-PROMETHEE*. Alasan menggabungkan ketiga metode tersebut yaitu karena dalam masalah seleksi peserta pelatihan di BLK Bantul belum ada perumusan kriteria, pembobotan kriteria dan penentuan peserta yang diterima didominasi ditentukan berdasarkan hasil seleksi wawancara. Sedangkan, hasil seleksi tertulis dan data diri peserta seleksi belum dikalkulasi dalam penilaian. Hal ini membutuhkan metode untuk merumuskan kriteria, pembobotan kriteria dan menentukan peringkat peserta seleksi pelatihan. Metode AHP digunakan untuk menentukan pembobotan terhadap kriteria dan subkriteria, karena AHP fokus pada pendekatan relatif/prioritas masing-masing kriteria. Namun, dalam AHP belum dijelaskan cara menentukan kriteria, sehingga digunakan metode *Modified*

Delphi (Afshari dkk., 2012) untuk menentukan kriteria yang berpengaruh. Metode *Modified Delphi* dipilih karena metode ini mempertimbangkan studi literatur dan pendapat para ahli. Metode AHP membutuhkan gabungan dengan metode *outranking*, karena gabungan metode menunjukkan hasil yang lebih baik dalam pemilihan personil. Metode yang digunakan yaitu PROMETHEE untuk menentukan peringkat terhadap alternatif. PROMETHEE dapat memberikan peringkat parsial dan lengkap setelah mengidentifikasi alternatif yang tidak dapat dibandingkan satu sama lain.

## 1.2 Rumusan Masalah

Berdasarkan uraian permasalahan pada latar belakang, maka rumusan masalah dalam penelitian ini adalah bagaimana menggabungkan metode AHP dengan metode lain untuk mengatasi kelemahan metode AHP dalam masalah seleksi peserta pelatihan di BLK Bantul.

## 1.3 Batasan Masalah

Batasan dalam penelitian ini meliputi:

1. Tahapan metode *Modified Delphi* yang digunakan berdasarkan penelitian oleh Afshari dkk. (2012).
2. Dalam metode *Modified Delphi* putaran pertama, masing-masing responden/ahli memasukkan jawaban kriteria yang dipilih ke dalam sistem. Dan untuk putaran kedua dan ketiga, proses diskusi dilakukan di luar sistem dan hasil diskusi dimasukkan ke dalam sistem oleh admin.
3. Pengembangan sistem difokuskan pada seleksi peserta pelatihan BLK Bantul, mulai dari pendaftaran hingga diperoleh peringkat peserta seleksi. Sistem tidak berkaitan dengan pelaksanaan dan evaluasi pelatihan.
4. Masukan yang digunakan berupa data peserta seleksi yang diinputkan peserta melalui formulir yang sudah disediakan, hasil seleksi tertulis dan hasil seleksi wawancara.

5. Hasil akhir dari sistem ini berupa peringkat peserta seleksi yang menjadi pertimbangan bagi pembuat keputusan untuk memilih peserta pelatihan.
6. Data uji yang digunakan yaitu data seleksi peserta pelatihan subkejuruan Teknik Sepeda Motor II Institusional APBN tahun 2016.

#### **1.4 Tujuan Penelitian**

Tujuan yang ingin dicapai dalam penelitian ini adalah menggabungkan metode AHP dengan metode *Modified Delphi* dan PROMETHEE untuk mengatasi kelemahan metode AHP dalam masalah seleksi peserta pelatihan di BLK Bantul.

#### **1.5 Manfaat Penelitian**

Bagi BLK Bantul khususnya, penelitian ini dapat digunakan untuk membantu pembuat keputusan dalam memilih peserta pelatihan berdasarkan kriteria yang telah ditetapkan. Hasil dari perhitungan nilai dibuat peringkat dan dapat digunakan untuk mendukung keputusan seleksi.

#### **1.6 Keaslian Penelitian**

Dalam penelitian yang sudah ada, permasalahan seleksi personil ditangani dengan metode MADM, antara lain metode SAW (Afshari dkk., 2010), AHP (Gibney dan Shang, 2007), AHP-*Dynamic Programming* (Özdemir, 2013), AHP-ROC-SAW (Sudipa dan Hartati, 2017), AHP-TOPSIS (Efendi dkk., 2017), *fuzzy AHP-Yager's Weighted Goals* (Güngör dkk., 2009), *Geometric Mean-SAW*, WPM, AHP, TOPSIS (Kumar dkk., 2013), VIKOR-*Information Entropy Weight* (El-santawy, 2012), *modified fuzzy VIKOR-Worst Case* (Alguliyev dkk., 2015), *fuzzy TOPSIS* (Kelemenis dan Askounis, 2010), PROMETHEE (ÖZTÜRK, 2013), *fuzzy AHP-fuzzy ELECTRE* (Rouyendegh dan Erkan, 2013), *fuzzy AHP-fuzzy VIKOR* (Salehi, 2016) dan AHP-PROMETHEE (Lemantara dan Setiawan, 2013). Dari penelitian tersebut, penggabungan metode menunjukkan hasil yang lebih baik dalam pemilihan personil. Metode AHP tepat digunakan untuk menentukan bobot berdasarkan prioritas tertentu. Dan metode PROMETHEE tepat digunakan

untuk menentukan peringkat. Namun mayoritas penentuan kriteria belum berdasarkan suatu metode tertentu, sehingga dalam penelitian ini digunakan gabungan metode *Modified Delphi*, AHP dan PROMETHEE untuk masalah seleksi personil. Berdasarkan kajian literatur, memang sudah ada peneliti yang menggabungkan metode *Modified Delphi*, AHP dan PROMETHEE (Mousavi dkk., 2013). Namun dalam penelitian tersebut *Modified Delphi* yang digunakan belum mempertimbangkan kajian literatur dan murni menggunakan pendapat ahli atau pembuat keputusan dalam proses pemilihan kriteria. Hasil peringkat yang diperoleh dalam penelitian yang dikembangkan, dibandingkan dengan metode lain yaitu metode TOPSIS.

### **1.7 Metodologi Penelitian**

Tahap-tahap penelitian yang dilakukan dalam penelitian ini yaitu sebagai berikut:

1. Pengumpulan data

Tahap ini merupakan tahap pengumpulan informasi yang dibutuhkan untuk mengembangkan sistem. Tahap pengumpulan data mencakup studi literatur dan wawancara.

- a. Studi literatur

Tahap ini mempelajari teori dan penelitian relevan terkait seleksi personil, metode *Modified Delphi*, AHP dan PROMETHEE berdasarkan kajian dari buku referensi, jurnal dan sumber lainnya. Studi literatur juga digunakan dalam proses metode *Modified Delphi*, yaitu untuk memperoleh kriteria-kriteria dalam seleksi personil secara umum.

- b. Wawancara

Wawancara dilakukan dengan melakukan tanya jawab langsung kepada pihak yang memiliki kapasitas dan informasi yang dibutuhkan dalam pelaksanaan penelitian.

Pihak-pihak tersebut yaitu sebagai berikut:

- 1) Plt. kepala BLK Bantul, untuk memperoleh informasi tentang proses seleksi peserta pelatihan, penentuan peserta pelatihan yang diterima dan kriteria yang digunakan.
  - 2) Kepala sub-bagian tata usaha, untuk memperoleh informasi tentang proses pendaftaran peserta pelatihan dan data calon peserta pelatihan.
  - 3) Kepala kejuruan, untuk memperoleh informasi tentang proses tes tertulis dan wawancara serta penilaiannya.
2. Analisis dan perancangan sistem

Tahap analisis dan perancangan sistem meliputi analisis data, pertimbangan penggunaan metode *Modified Delphi-AHP-PROMETHEE*, analisis kebutuhan sistem, perancangan sistem dengan menggunakan *Unified Modelling Language (UML)*, perancangan basis data, perancangan antarmuka sistem dan perancangan pemodelan. Analisis dan perancangan sistem disesuaikan dengan tahapan dalam metode *Modified Delphi*, AHP dan PROMETHEE.

### 3. Implementasi

Tahap ini merupakan pembuatan sistem berdasarkan analisis dan perancangan yang telah dibuat. Sistem dibangun dengan bahasa pemrograman PHP dan MySQL untuk basis data.

### 4. Pengujian

Pengujian dilakukan untuk mengetahui tingkat kesesuaian (*accuracy*) gabungan metode *Modified Delphi-AHP-PROMETHEE* untuk masalah seleksi peserta pelatihan BLK Bantul. Tingkat kesesuaian diketahui dengan membandingkan hasil implementasi metode dengan hasil data masa lalu. Pengujian meliputi dua tahap. Tahap pertama untuk mengetahui kesesuaian penggunaan metode *Modified Delphi*. Dan tahap kedua untuk mengetahui kesesuaian penggunaan metode PROMETHEE.

Berikut ini langkah-langkah pengujian pada tahap pertama:

- 1) Menentukan kriteria berdasarkan pendapat dari pengambil keputusan.
- 2) Menghitung bobot kriteria menggunakan metode AHP.
- 3) Menentukan peringkat peserta seleksi pelatihan dengan menggunakan metode PROMETHEE.
- 4) Membandingkan hasil AHP-PROMETHEE dan hasil *Modified Delphi*-AHP-PROMETHEE dengan hasil data masa lalu (tidak menggunakan metode MADM).

Berikut ini langkah-langkah pengujian pada tahap kedua:

- 1) Menentukan kriteria dengan menggunakan metode *Modified Delphi*.
- 2) Menghitung bobot kriteria menggunakan metode AHP.
- 3) Setelah diperoleh bobot kriteria menggunakan AHP, hasil tersebut digunakan untuk menentukan peringkat peserta seleksi pelatihan dengan menggunakan metode TOPSIS. TOPSIS dipilih karena dalam menentukan preferensinya berbeda dengan PROMETHEE yaitu menggunakan solusi ideal.
- 4) Peringkat peserta seleksi pelatihan yang menggunakan metode PROMETHEE dan TOPSIS dibandingkan dengan hasil data masa lalu (tidak menggunakan metode MADM).
5. Penulisan laporan hasil penelitian

## 1.8 Sistematika Penulisan

Penelitian ini termasuk penelitian pengembangan sistem dan penyajian penulisan laporan terdiri dari tujuh bab yang mengandung pokok bahasan sebagai berikut:

### BAB I PENDAHULUAN

Bab ini menjelaskan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, keaslian penelitian, metodologi penelitian dan sistematika penulisan.

BAB II	TINJAUAN PUSTAKA
	Bab ini menjelaskan beberapa pustaka yang relevan untuk menunjukkan keaslian penelitian.
BAB III	LANDASAN TEORI
	Bab ini menjelaskan teori-teori yang digunakan dalam pembahasan untuk menjawab permasalahan, seperti metode <i>Modified Delphi</i> , AHP dan PROMETHEE.
BAB IV	ANALISIS DAN RANCANGAN SISTEM
	Bab ini menjelaskan analisis dan rancangan sistem untuk seleksi peserta pelatihan yang meliputi analisis data, pertimbangan penggunaan metode <i>Modified Delphi-AHP-PROMETHEE</i> , analisis kebutuhan sistem, perancangan sistem dengan menggunakan <i>Unified Modelling Language</i> (UML), perancangan basis data, perancangan antarmuka sistem dan perancangan pemodelan.
BAB V	IMPLEMENTASI
	Bab ini menjelaskan implementasi dari perancangan sistem yang berupa penjelasan program.
BAB VI	HASIL PENELITIAN DAN PEMBAHASAN
	Bab ini menjelaskan hasil akhir dari sistem yang dibangun dan hasil dari pengujian.
BAB VII	KESIMPULAN
	Bab ini berisi kesimpulan dari hasil penelitian.

## **BAB II**

### **TINJAUAN PUSTAKA**

Permasalahan seleksi personil merupakan salah satu hal yang penting bagi suatu organisasi. Seleksi personil bertujuan merekrut individu yang sesuai dengan kualifikasi yang dibutuhkan untuk melakukan suatu pekerjaan dengan cara yang terbaik. Penelitian untuk memecahkan permasalahan seleksi personil telah banyak dilakukan, yaitu dengan menggunakan metode MADM untuk menentukan alternatif yang tepat.

Gibney dan Shang (2007) menggunakan metode AHP dalam proses seleksi dekan. Metode AHP melibatkan tiga tahap pemecahan masalah, yaitu prinsip dekomposisi, perbandingan penilaian dan sintesis prioritas. Namun dalam penelitian ini belum ada kesepakatan dari semua pembuat keputusan dalam menentukan kriteria dan subkriteria, sehingga hasil diskusi kelompok dan penerapan AHP berbeda dengan keputusan rektor. Penelitian ini menyimpulkan bahwa AHP merupakan alat yang penting dan harus dimasukkan ke dalam proses seleksi personil karena lebih efektif. Berdasarkan hasil tersebut, Özdemir (2013) juga menggunakan pendekatan metode yang sama yaitu AHP yang digabungkan dengan metode *Dynamic Programming* (DP). AHP digunakan untuk menangani multikriteria proses seleksi personil dan proses terstruktur secara hirarkis. Dengan AHP diperoleh pembobotan masing-masing kriteria. DP untuk membagi proses seleksi personil menjadi langkah-langkah. Perumusan model DP menggunakan bobot AHP. Tujuan dari menggabungkan metode tersebut yaitu untuk menentukan alternatif keputusan yang optimal dalam setiap langkah dari proses seleksi personil. Penggunaan model DP tidak selalu cocok digunakan pada semua kasus. Hal ini karena dalam model DP, persoalan dibagi menjadi beberapa langkah, yang pada setiap langkah hanya diambil satu keputusan. Solusi pada setiap langkah dibangun dari hasil solusi langkah sebelumnya. Pendekatan metode AHP juga

digunakan Kumar dkk. (2013) dalam penelitiannya untuk mengidentifikasi staf pengajar yang tepat dan untuk mengevaluasi yang terbaik berdasarkan pengukuran kinerja yang tepat. Pemecahan masalah mencakup dua langkah, yaitu mengidentifikasi bobot yang cocok dan menerapkan metode MADM yang berbeda. Identifikasi bobot dengan menggunakan metode *Geometric Mean*. Hasil dari bobot digunakan metode SAW, WPM, AHP, TOPSIS untuk menentukan peringkat alternatif. Metode *Geometric Mean* digunakan dalam AHP sebagai metode perataan dalam matriks perbandingan berpasangan. Sudipa dan Hartati (2017) menggunakan metode AHP, ROC dan SAW untuk seleksi penerima beasiswa PPA dan BBP di STIKOM Indonesia (STIKI). AHP digunakan untuk menentukan bobot kriteria dan subkriteria, ROC digunakan untuk menentukan skoring data tingkat kejuaraan dan bidang kejuaraan dan SAW digunakan untuk mencari nilai akhir dan perankingan. Metode AHP yang digabungkan dengan metode lain yaitu TOPSIS juga diterapkan oleh Efendi dkk. (2017) dalam penelitiannya untuk menentukan penerima Bantuan Langsung Masyarakat (BLM) yang sesuai bagi kelurahan berdasarkan standar parameter yang telah ditetapkan oleh P2KP Kota Ternate. AHP untuk menghitung bobot prioritas dari tiap parameter, yang nantinya menjadi acuan perankingan yang dilakukan dengan metode TOPSIS.

Metode AHP dalam kasus tertentu memungkinkan ketidaktepatan dalam ekspresi masing-masing kriteria, sehingga dikembangkan dengan model *fuzzy*. Model *fuzzy* dapat menerjemahkan dari ungkapan verbal menjadi numerik. Penggunaan *fuzzy* untuk menangani sumber yang tidak tepat karena informasi yang tidak lengkap dan sulit diperoleh. Güngör dkk. (2009) menggunakan *fuzzy* AHP untuk memecahkan masalah seleksi personil. Hasil *fuzzy* AHP dibandingkan dengan metode *Yager's weighted goals*, menunjukkan bahwa ada konsistensi yang baik antara keduanya. Peringkat alternatif oleh masing-masing metode ini sangat dekat satu sama lain. Namun metode *Yager's weighted goals* membutuhkan perhitungan operasional yang lebih kompleks.

El-santawy (2012) mengkombinasikan metode VIKOR dan *Information Entropy Weight* (IEW) untuk memecahkan masalah seleksi personil dalam suatu perusahaan. VIKOR digunakan untuk menentukan peringkat kandidat. IEW digunakan untuk pembobotan kriteria. VIKOR dan IEW peka terhadap adanya perubahan, sehingga jika ada parameter atau alternatif yang ditambahkan atau dihapus, harus dirumuskan dan dipecahkan lagi. Alguliyev dkk. (2015) menggunakan model *fuzzy hybrid* MADM untuk memecahkan masalah evaluasi personil. Pembobotan kriteria dihitung dengan metode *Worst-Case*. Metode *modified fuzzy* VIKOR untuk menentukan peringkat alternatif. *Fuzzy hybrid* MADM juga diusulkan oleh Salehi (2016), yaitu dengan menggabungkan metode *fuzzy AHP* dan *fuzzy VIKOR* untuk memecahkan masalah seleksi personil. *Fuzzy AHP* untuk menentukan kriteria yang sesuai dan menghitung bobot kriteria. *Fuzzy VIKOR* untuk menentukan peringkat. Kombinasi metode MADM untuk pemecahan masalah seleksi personil merupakan sesuatu hal yang unik dan direkomendasikan untuk penelitian selanjutnya. Penelitian selanjutnya dapat mengkombinasikan metode selain AHP-VIKOR.

Afshari dkk. (2010) mengusulkan metode SAW untuk memecahkan masalah seleksi personil. Dalam penelitian ini perhitungan bobot kriteria dengan menggunakan *Comparison Matrix* dan penentuan peringkat personil dengan menggunakan metode SAW. Perhitungan bobot kriteria dan penentuan peringkat dalam SAW sangat sederhana, sehingga kurang cocok untuk menangani permasalahan yang kompleks.

Kelemenis dan Askounis (2010) menggunakan metode *fuzzy TOPSIS* untuk pemecahan masalah personil, karena merupakan pengukuran baru untuk peringkat alternatif yang berdasarkan konsep *veto*. Kriteria keputusan akhir diperoleh bukan dari kesamaan dengan solusi ideal, tetapi jarak alternatif dari *veto* yang ditetapkan oleh pengambil keputusan. Namun dalam *fuzzy TOPSIS* belum ada analisis sensitivitas pada hasil yang diperoleh.

Rouyendegh dan Erkan (2013) menggunakan metode *fuzzy* ELECTRE untuk pemilihan kandidat terbaik dari 5 kandidat staf pengajar. Penentuan kriteria ditetapkan oleh instansi, terdapat 10 kriteria yang dikelompokkan ke dalam 3 kriteria utama. Dalam penelitian ini juga membandingkan peringkat yang diperoleh menggunakan *fuzzy* AHP dan *fuzzy* ELECTRE. Metode *fuzzy* ELECTRE merupakan konsep outranking dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai, sehingga metode ini cocok untuk menentukan peringkat alternatif. Namun dalam metode *fuzzy* ELECTRE belum ada analisis sensitivitas pada hasil yang diperoleh, sehingga belum dapat ditentukan ketepatan hasil tersebut.

ÖZTÜRK (2013) menggunakan metode PROMETHEE untuk seleksi personil resepsionis di perusahaan akomodasi. Pemilihan PROMETHEE karena metode ini dapat membantu pemilihan personil yang tepat, cepat dan efisien. PROMETHEE dapat memberikan peringkat parsial setelah mengidentifikasi alternatif yang tidak dapat dibandingkan satu sama lain. PROMETHEE menganggap fungsi preferensi masing-masing kriteria yang diidentifikasi oleh para pembuat keputusan, sehingga setiap kriteria dapat dievaluasi secara berbeda. Dengan demikian, keputusan yang diperoleh lebih baik. Lemantara dan Setiawan (2013) juga menggunakan metode PROMETHEE yang digabungkan dengan metode AHP untuk menentukan mahasiswa berprestasi yang akan dikirim ke *event*. Tujuan penggabungan metode ini yaitu untuk meningkatkan kualitas saran pemilihan mahasiswa. AHP digunakan untuk memperoleh bobot kriteria dan PROMETHEE digunakan untuk menentukan urutan prioritas dari calon peserta *event*. Penggabungan metode AHP dan PROMETHEE juga dilakukan Bogdanovic dan Miletic (2014) dalam pemilihan personil yang sesuai. Metode AHP digunakan untuk menganalisis struktur masalah seleksi personil dan menentukan bobot kriteria. Metode PROMETHEE untuk menentukan peringkat. Metode PROMETHEE memungkinkan analisis sensitivitas pada hasil yang diperoleh dan menentukan kriteria yang paling efektif dalam proses pengambilan

keputusan. Hal ini tidak tersedia dalam metode lain, seperti AHP, *fuzzy* AHP, ELECTRE, TOPSIS, dll. AHP memberikan tingkat koherensi, korelasi, konsistensi dan akurasi bobot kriteria yang lebih tinggi daripada bobot yang ditentukan berdasarkan intuisi atau pengetahuan ahli. Mousavi dkk. (2013) menggabungkan metode *Modified* Delphi, AHP dan PROMETHEE untuk memecahkan masalah pemilihan lokasi pabrik yang tepat. *Modified* Delphi untuk memilih kriteria yang paling berpengaruh oleh ahli profesional atau pembuat keputusan. AHP digunakan untuk menganalisis struktur masalah pemilihan lokasi pabrik dan mendapatkan bobot dari kriteria yang dipilih. PROMETHEE untuk menilai dan menentukan peringkat alternatif lokasi. Dalam penelitian ini terdapat 3 pembuat keputusan, 5 kriteria dan 3 alternatif. Peneliti merekomendasikan penggunaan gabungan metode ini di bidang aplikasi yang berbeda untuk studi di masa depan, terutama di bidang aplikasi manufaktur lain di mana terdapat beberapa kriteria yang berkonflik. Gabungan metode *Modified* Delphi, AHP dan PROMETHEE dapat meningkatkan proses seleksi dalam masalah pemilihan lokasi pabrik dan proses penilaian dapat dimengerti oleh para pembuat keputusan. Namun, *Modified* Delphi dalam penelitian ini belum mempertimbangkan kajian literatur terkait. Daftar kriteria yang diperoleh berdasarkan hasil bertanya dengan ahli profesional atau pembuat keputusan. Untuk masalah seleksi personil, selain berdasarkan pengetahuan ahli atau pembuat keputusan, juga membutuhkan kajian literatur dalam penentuan kriteria karena sudah banyak literatur yang membahas terkait hal ini.

Pemecahan masalah seleksi personil dengan menggunakan metode MADM yang telah dijelaskan dalam penelitian-penelitian tersebut, sebagian besar belum menggunakan metode tertentu untuk menentukan kriteria. Pemilihan kriteria sangat penting dalam menentukan kesesuaian keputusan dengan tujuan yang ingin dicapai. Afshari dkk. (2012) membangun suatu model untuk mengidentifikasi kriteria dalam seleksi personil dengan menggunakan metode *Modified* Delphi. Teknik *Modified* Delphi digunakan untuk mencari ide-ide terbaik dari para

manajer dan ahli untuk pemilihan kriteria. Metode *Modified Delphi* menunjukkan hasil yang sangat baik dalam pemilihan kriteria dan dapat meningkatkan efisiensi dalam suatu proses pengambilan keputusan.

Berdasarkan beberapa penelitian tersebut, dapat dijelaskan perbedaan penelitian yang diusulkan, yaitu belum ada penelitian yang menggabungkan metode *Modified Delphi*, AHP dan PROMETHEE dalam pemecahan masalah seleksi personil. Metode *Modified Delphi* untuk memilih kriteria yang tepat. Metode AHP untuk menentukan bobot kriteria dan subkriteria. Dan metode PROMETHEE untuk menentukan peringkat alternatif. Studi kasus gabungan metode *Modified Delphi-AHP-PROMETHEE* yaitu digunakan untuk memecahkan masalah seleksi peserta pelatihan BLK Bantul. Perbandingan dengan penelitian terdahulu dapat ditunjukkan pada Tabel (2.1).

**Tabel 2.1 Tinjauan Pustaka**

No.	Nama	Metode Pendekatan	Hasil	Komentar
1	Gibney dan Shang (2007)	AHP	<p>Penelitian ini menjelaskan penggunaan metode AHP dalam proses seleksi dekan. AHP melibatkan tiga tahap pemecahan masalah, yaitu prinsip dekomposisi, perbandingan penilaian dan sintesis prioritas. Panitia seleksi dekan membuat peringkat kandidat melalui diskusi kelompok dan penerapan AHP. Namun hasil akhir dari keduanya berbeda dengan keputusan rektor. Hal ini disebabkan karena perbedaan pemilihan kriteria dan pembobotan masing-masing kriteria. Penelitian ini menyimpulkan bahwa AHP merupakan alat yang penting dan harus dimasukkan ke dalam proses seleksi personil karena lebih efektif.</p>	<p>Hasil dari AHP yaitu peringkat prioritas yang menunjukkan preferensi keseluruhan untuk setiap alternatif. AHP fokus pada pendekatan relatif/prioritas masing-masing kriteria, sehingga dalam penelitian yang akan dikembangkan menggunakan AHP sebagai teknik pembobotan kriteria. Namun dalam penelitian ini belum ada kesepakatan dari semua pembuat keputusan dalam menentukan kriteria dan subkriteria, sehingga hasil diskusi kelompok dan penerapan AHP berbeda dengan keputusan rektor. Selain itu tidak dijelaskan peringkat keseluruhan kandidat karena penelitian ini memilih kandidat yang terbaik, sehingga perlu menggabungkan dengan metode lain untuk menentukan peringkat keseluruhan kandidat.</p>
2	Özdemir (2013)	AHP, <i>Dynamic Programming</i> (DP)	<p>Penelitian ini menggabungkan dua metode, yaitu AHP dan DP. AHP digunakan untuk menangani multikriteria proses seleksi personil dan proses terstruktur secara hierarkis. AHP diperoleh pembobotan</p>	<p>Penggunaan model DP tidak selalu cocok digunakan pada semua kasus. Hal ini karena dalam model DP, persoalan dibagi menjadi beberapa langkah, yang pada setiap langkah hanya diambil satu</p>

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
			masing-masing kriteria. DP untuk membagi proses seleksi personil menjadi langkah-langkah. Perumusan model DP menggunakan bobot AHP. Tujuan dari menggabungkan metode tersebut yaitu untuk menentukan alternatif keputusan yang optimal dalam setiap langkah dari proses seleksi personil. Kriteria ditentukan perusahaan berdasarkan proses seleksi personil perusahaan.	keputusan. Solusi pada setiap langkah dibangun dari hasil solusi langkah sebelumnya. Penentuan kriteria spesifik pada proses bisnis perusahaan tertentu.
3	Sudipa dan Hartati (2017)	AHP, ROC, SAW	Metode AHP, ROC dan SAW digunakan untuk penilaian yang objektif dalam seleksi penerima beasiswa PPA dan BBP di STIKOM Indonesia (STIKI). AHP digunakan untuk menentukan bobot kriteria dan subkriteria, ROC digunakan untuk menentukan skoring data tingkat kejuaraan dan bidang kejuaraan dan SAW digunakan untuk mencari nilai akhir dan perankingan.	Penelitian ini belum dijelaskan metode yang digunakan dalam menentukan kriteria. Dan berdasarkan saran dari peneliti bahwa diperlukan penambahan kriteria guna menunjang penilaian akhir yang lebih detail dan komprehensif, sehingga dibutuhkan metode untuk menentukan kriteria.
4	Efendi dkk. (2017)	AHP, TOPSIS	Penggabungan metode AHP dan TOPSIS digunakan untuk menentukan penerima BLM yang sesuai bagi kelurahan berdasarkan standar parameter yang telah ditetapkan oleh P2KP Kota Ternate. AHP untuk menghitung bobot prioritas dari tiap parameter. TOPSIS digunakan sebagai acuan	Kriteria untuk penentuan penerima BLM sudah ditetapkan oleh P2KP Kota Ternate, sehingga dalam penelitian ini tidak menggunakan metode penentuan kriteria. Belum ada perbandingan hasil perankingan TOPSIS dengan metode lain. Validasi penggunaan metode AHP-

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
			perankingan. Hasil pengujian menunjukkan bahwa sistem dapat melakukan perhitungan yang valid dan bersifat fleksibel akan perubahan parameter dan alternatif.	TOPSIS masih ditentukan berdasarkan hasil keputusan sistem yang menunjukkan hasil yang sama dengan hasil keputusan Team Leader.
5	Kumar dkk. (2013)	Metode <i>Geometric Mean</i> , SAW, WPM, AHP, TOPSIS	Penerapan beberapa paradigma MADM untuk mengidentifikasi staf pengajar yang tepat dan untuk mengevaluasi yang terbaik berdasarkan pengukuran kinerja yang tepat. Kriteria yang digunakan ditentukan langsung oleh pembuat keputusan. Pemecahan masalah mencakup dua langkah, yaitu mengidentifikasi bobot yang cocok dan menerapkan metode MADM yang berbeda. Identifikasi bobot dengan menggunakan metode <i>Geometric Mean</i> . Hasil dari bobot digunakan metode SAW, WPM, AHP, TOPSIS untuk menentukan peringkat alternatif. Dari metode tersebut menghasilkan alternatif terbaik yang sama yaitu Rama. Namun dalam penentuan peringkat diperoleh hasil yang berbeda-beda.	Penentuan kriteria kurang objektif, perlu pertimbangan dari literatur. Metode <i>Geometric Mean</i> digunakan dalam AHP sebagai metode perataan dalam matriks perbandingan berpasangan. Perlu metode lain untuk menentukan peringkat alternatif sehingga memperoleh hasil yang lebih tepat.
6	Güngör dkk. (2009)	Fuzzy AHP, Metode <i>Yager's weighted goals</i>	Penelitian ini menggunakan FAHP untuk mengevaluasi personil terbaik berdasarkan peringkat kedua kriteria kualitatif dan kuantitatif dengan model <i>fuzzy</i> . Model <i>fuzzy</i> digunakan karena dapat menerjemahkan dari	Metode FAHP dengan <i>Yager's weighted goals</i> menunjukkan konsistensi yang baik. Peringkat alternatif oleh masing-masing metode ini sangat dekat satu sama lain. Namun metode <i>Yager's weighted goals</i>

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
			ungkapan verbal menjadi numerik, sehingga dapat menggambarkan ketidaktepatan dalam ekspresi masing-masing kriteria. Hasil FAHP dibandingkan dengan metode <i>Yager's weighted goals</i> , menunjukkan bahwa ada konsistensi yang baik antara keduanya. Kriteria ditentukan oleh ahli.	membutuhkan perhitungan operasional yang lebih kompleks. Berdasarkan pertimbangan tersebut, untuk melakukan seleksi personil dalam penelitian ini tidak menggunakan metode <i>Yager's weighted goals</i> . Penentuan kriteria selain dari ahli, bisa dipertimbangkan dengan literatur tentang seleksi personil.
7	Salehi (2016)	<i>Fuzzy AHP</i> , <i>Fuzzy VIKOR</i>	Penelitian ini mengembangkan pendekatan <i>hybrid fuzzy MADM</i> dengan mengkombinasikan AHP dan VIKOR untuk pemecahan masalah seleksi personil. Penggunaan <i>fuzzy</i> untuk menangani sumber yang tidak tepat karena informasi yang tidak lengkap dan sulit diperoleh. <i>Fuzzy AHP</i> untuk menentukan kriteria yang sesuai dan menghitung bobot kriteria. <i>Fuzzy VIKOR</i> untuk menentukan peringkat. VIKOR dipilih karena salah satu metode MADM yang sering digunakan peneliti dan sudah berhasil digunakan di berbagai macam masalah dan konteks.	Kombinasi metode MADM untuk pemecahan masalah seleksi personil merupakan sesuatu hal yang unik dan direkomendasikan untuk penelitian selanjutnya. Penelitian selanjutnya dapat mengkombinasikan metode selain AHP-VIKOR. Dalam penelitian ini tidak dijelaskan metode penentuan kriteria karena studi kasus menggunakan penelitian sebelumnya.
8	Afshari dkk. (2010)	SAW	Metode SAW digunakan untuk memecahkan masalah seleksi personil. Dalam penelitian ini perhitungan bobot kriteria dengan menggunakan <i>Comparison Matrix</i> dan penentuan peringkat personil dengan	Perhitungan bobot kriteria dan penentuan peringkat dalam SAW sangat sederhana. Penentuan bobot kriteria dalam SAW cenderung subjektif.

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
			menggunakan metode SAW. Penelitian menerapkan 7 kriteria untuk memilih personil terbaik dari 5 personil yang ada.	
9	Kelemenis dan Askounis (2010)	Fuzzy TOPSIS	<i>Fuzzy</i> TOPSIS digunakan untuk pemecahan masalah personil, karena merupakan pengukuran baru untuk peringkat alternatif yang berdasarkan konsep <i>veto</i> . Kriteria keputusan akhir diperoleh bukan dari kesamaan dengan solusi ideal, tetapi jarak alternatif dari <i>veto</i> yang ditetapkan oleh pengambil keputusan. Kriteria ditentukan oleh konsultan dan CEO, yang menghasilkan 11 kriteria untuk menentukan CIO terpilih dari 4 kandidat yang ada.	<i>Fuzzy</i> TOPSIS cocok untuk model penentuan peringkat dalam pengambilan keputusan. Namun dalam <i>Fuzzy</i> TOPSIS belum ada analisis sensitivitas pada hasil yang diperoleh. Penentuan kriteria kurang objektif, perlu pertimbangan dari literatur.
10	El-santawy (2012)	VIKOR, <i>Information Entropy Weight</i> (IEW)	Penelitian ini mengkombinasikan metode VIKOR dan IEW untuk memecahkan masalah seleksi personil dalam suatu perusahaan. VIKOR digunakan untuk menentukan peringkat kandidat. IEW digunakan untuk pembobotan kriteria. Departemen HR yang menentukan kriteria dan menilai masing-masing kandidat, yaitu terdapat 5 kriteria dan 6 kandidat. Kandidat yang diterima sebanyak 2 orang.	VIKOR dan IEW peka terhadap adanya perubahan, sehingga jika ada parameter atau alternatif yang ditambahkan atau dihapus, harus dirumuskan dan dipecahkan lagi. Penelitian ini belum dijelaskan metode yang digunakan dalam menentukan kriteria.

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
11	Alguliyev dkk. (2015)	<i>Modified Fuzzy VIKOR, Worst Case</i>	<p>Model <i>fuzzy hybrid</i> MADM untuk memecahkan masalah evaluasi personil. Pembobotan kriteria dihitung dengan metode <i>Worst-Case</i>. Metode <i>modified fuzzy</i> VIKOR untuk menentukan peringkat alternatif. Penelitian ini menggunakan 5 kriteria untuk evaluasi, sehingga menghasilkan peringkat dan terpilih alternatif terbaik dengan metode <i>fuzzy</i> VIKOR dan <i>modified fuzzy</i> VIKOR. Setelah dilakukan perbandingan antara hasil <i>fuzzy</i> VIKOR dan <i>modified fuzzy</i> VIKOR, menunjukkan bahwa metode <i>modified fuzzy</i> VIKOR mempunyai lebih banyak keuntungan dibandingkan metode <i>fuzzy</i> VIKOR. Dari segi kompleksitas komputasi, metode <i>modified fuzzy</i> VIKOR lebih efektif.</p>	Pemilihan kriteria berdasarkan kajian literatur tentang evaluasi personil. Hal ini perlu adanya pertimbangan dari ahli.
12	Rouyendegh dan Erkan (2013)	<i>Fuzzy ELECTRE, Fuzzy AHP</i>	<p>Metode <i>Fuzzy ELECTRE</i> untuk pemilihan kandidat terbaik dari 5 kandidat staf pengajar. Penentuan kriteria ditetapkan oleh instansi, terdapat 10 kriteria yang dikelompokkan ke dalam 3 kriteria utama. Dalam penelitian ini juga membandingkan peringkat yang diperoleh menggunakan <i>fuzzy AHP</i> dan <i>fuzzy ELECTRE</i>. Hasil kandidat terbaik dari kedua metode yang diperoleh sama, yaitu kandidat 2.</p>	Metode <i>fuzzy ELECTRE</i> merupakan konsep outranking dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai, sehingga metode ini cocok untuk menentukan peringkat alternatif. Namun dalam metode <i>fuzzy ELECTRE</i> belum ada analisis sensitivitas pada hasil yang diperoleh, sehingga belum dapat ditentukan ketepatan hasil tersebut.

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
13	Afshari dkk. (2012)	<i>Modified Delphi</i>	Penelitian ini membangun suatu model untuk mengidentifikasi kriteria dalam seleksi personil dengan menggunakan metode <i>Modified Delphi</i> . Teknik <i>Modified Delphi</i> digunakan untuk mencari ide-ide terbaik dari para manajer dan para ahli untuk pemilihan kriteria. Metode <i>Modified Delphi</i> menunjukkan hasil yang sangat baik dalam pemilihan kriteria dan dapat meningkatkan efisiensi dalam suatu proses pengambilan keputusan. Dalam penelitian ini, metode <i>Modified Delphi</i> mencakup tiga putaran survei dengan 9 orang ahli dan menghasilkan 14 kriteria yang disusun ke dalam 4 kelompok utama.	Dengan hasil yang ditunjukkan dalam paper, metode <i>Modified Delphi</i> cocok diimplementasikan dalam pemilihan kriteria suatu proses pengambilan keputusan. Pemilihan kriteria sangat penting dalam menentukan kesesuaian keputusan dengan tujuan yang ingin dicapai. Dalam paper ini fokus membahas model pemilihan kriteria, sehingga belum diimplementasikan dengan metode MCDM.
14	ÖZTÜRK (2013)	PROMETHEE	Metode PROMETHEE digunakan untuk seleksi personil resepsionis di perusahaan akomodasi. Pemilihan PROMETHEE karena metode ini dapat membantu pemilihan personil yang tepat, cepat dan efisien. Kriteria ditentukan melalui wawancara dengan manajer HRD dari perusahaan akomodasi dan kajian literatur tentang seleksi personil. Dalam penelitian ini terdapat 15 kriteria dan 7 kandidat. Hasil analisis diperoleh dengan menentukan	Keuntungan metode PROMETHEE yaitu merupakan metode penentuan peringkat yang <i>user-friendly</i> , sehingga cocok digunakan untuk menentukan peringkat alternatif. PROMETHEE dapat memberikan peringkat parsial setelah mengidentifikasi alternatif yang tidak dapat dibandingkan satu sama lain. PROMETHEE menganggap fungsi preferensi masing-masing kriteria yang diidentifikasi oleh para pembuat

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
			peringkat parsial dan lengkap.	keputusan, sehingga setiap kriteria dapat dievaluasi secara berbeda. Dengan demikian, keputusan yang diperoleh lebih baik.
15	Lemantara dan Setiawan (2013)	AHP, PROMETHEE	Metode AHP dan PROMETHEE digabungkan untuk menentukan mahasiswa berprestasi yang akan dikirim ke <i>event</i> . Tujuan penggabungan metode ini yaitu untuk meningkatkan kualitas saran pemilihan mahasiswa. AHP digunakan untuk memperoleh bobot kriteria dan PROMETHEE digunakan untuk menentukan urutan prioritas dari calon peserta <i>event</i> .	Penentuan kriteria kurang objektif, perlu pertimbangan dari literatur. Berdasarkan hasil penelitian, gabungan metode AHP-PROMETHEE terbukti lebih baik dan objektif daripada metode AHP atau PROMETHEE saja.
16	Bogdanovic dan Miletic (2014)	AHP, PROMETHEE	Penelitian ini mengimplementasikan gabungan metode AHP dan PROMETHEE dalam pemilihan personil yang sesuai. Perusahaan akan melakukan proses reorganisasi dan pengurangan tenaga kerja, sehingga perlu dipilih 3 tenaga kerja yang sesuai dari 5 tenaga kerja yang ada. Metode AHP digunakan untuk menganalisis struktur masalah seleksi personil dan menentukan bobot kriteria. Metode PROMETHEE untuk menentukan peringkat. Kriteria ditentukan dari tahap pengumpulan data, yaitu terdapat 9 kriteria.	Metode PROMETHEE memungkinkan analisis sensitivitas pada hasil yang diperoleh dan menentukan kriteria yang paling efektif dalam proses pengambilan keputusan. Hal ini tidak tersedia dalam metode lain, seperti AHP, <i>fuzzy</i> AHP, ELECTRE, TOPSIS, dll. AHP memberikan tingkat koherensi, korelasi, konsistensi dan akurasi bobot kriteria yang lebih tinggi daripada bobot yang ditentukan berdasarkan intuisi atau pengetahuan ahli. Dalam penelitian ini tidak dijelaskan cara menentukan kriteria.

**Tabel 2.1 (lanjutan)**

No.	Nama	Metode Pendekatan	Hasil	Komentar
17	Mousavi dkk. (2013)	<i>Modified</i> Delphi, AHP, PROMETHEE	Penelitian ini menggabungkan metode <i>Modified</i> Delphi, AHP dan PROMETHEE untuk memecahkan masalah pemilihan lokasi pabrik yang tepat. <i>Modified</i> Delphi untuk memilih kriteria yang paling berpengaruh oleh ahli profesional atau pembuat keputusan. AHP digunakan untuk menganalisis struktur masalah pemilihan lokasi pabrik dan mendapatkan bobot dari kriteria yang dipilih. PROMETHEE untuk menilai dan menentukan peringkat alternatif lokasi. Dalam penelitian ini terdapat 3 pembuat keputusan, 5 kriteria dan 3 alternatif. Peneliti merekomendasikan penggunaan gabungan metode ini di bidang aplikasi yang berbeda untuk studi di masa depan, terutama di bidang aplikasi manufaktur lain di mana terdapat beberapa kriteria yang berkonflik.	Gabungan metode <i>Modified</i> Delphi, AHP dan PROMETHEE dapat meningkatkan proses seleksi dalam masalah pemilihan lokasi pabrik dan proses penilaian dapat dimengerti oleh para pembuat keputusan. Namun, <i>Modified</i> Delphi dalam penelitian ini belum mempertimbangkan kajian literatur. Daftar kriteria yang diperoleh berdasarkan hasil bertanya dengan ahli profesional atau pembuat keputusan. Untuk masalah seleksi personil, selain berdasarkan pengetahuan ahli atau pembuat keputusan, juga membutuhkan kajian literatur dalam penentuan kriteria karena sudah banyak literatur yang membahas terkait hal ini.

## BAB III

### LANDASAN TEORI

#### 3.1 Metode MADM (*Multiple Attribute Decision-Making*)

MADM mengacu pada membuat keputusan berdasarkan beberapa kriteria yang saling bertentangan di antara alternatif yang ada (Hwang dan Yoon, 1981). MADM merupakan metode pembuatan keputusan berdasarkan preferensi tertentu dan menentukan bobot relatif yang benar untuk kriteria (Huang dan Tzeng, 2011). Dari penjelasan tersebut, MADM yaitu suatu metode pengambilan keputusan untuk menetapkan alternatif terbaik dari sejumlah alternatif berdasarkan beberapa kriteria yang saling bertentangan. Permasalahan dalam MADM sangat beragam, yaitu kejadian sehari-hari dalam kehidupan. Namun semua permasalahan mempunyai karakteristik umum.

Menurut Hwang dan Yoon (1981), karakteristik umum tersebut adalah sebagai berikut:

1. Setiap masalah mempunyai beberapa tujuan/atribut. Pembuat keputusan harus mendefinisikan tujuan/atribut yang relevan dalam setiap masalah.
2. Beberapa kriteria yang saling bertentangan.
3. Masing-masing tujuan/atribut mempunyai satuan ukuran yang berbeda-beda.
4. Solusi untuk setiap masalah yaitu untuk merancang atau memilih alternatif terbaik yang memenuhi semua kriteria.

Metode MADM mempunyai empat bagian utama, yaitu alternatif, atribut, bobot atau prioritas masing-masing atribut dan ukuran kinerja alternatif sehubungan dengan atribut (Rao, 2013). Hwang dan Yoon (1981) menyatakan metode MADM mempunyai dua jenis, yaitu *Non-compensatory Methods* dan *Compensatory Methods*. *Non-compensatory Methods* tidak mengizinkan timbal

balik antara atribut. Sebuah nilai yang tidak menguntungkan dalam satu atribut tidak dapat diimbangi dengan nilai yang menguntungkan dalam atribut lainnya. Setiap atribut harus berdiri sendiri, sehingga perbandingan yang dibuat yaitu secara *attribute-by-attribute*. Contoh dari metode ini antara lain *dominance method*, *maxmin method*, *maxmax method*, *conjunctive constraint method* dan *disjunctive constraint method*. *Compensatory Methods* mengizinkan timbal balik antara atribut. Sebuah penurunan dalam satu atribut diijinkan jika dikompensasi oleh beberapa peningkatan dalam satu atau lebih atribut lainnya. Metode *Compensatory Methods* dikelompokkan menjadi empat subkelompok, yaitu *Scoring Methods*, *Compromising Methods*, *Concordance Methods* dan *Evidential Reasoning Approach*.

Prosedur MADM dapat diringkas menjadi lima langkah utama sebagai berikut (Dubois dan Prade, 1980):

**Langkah 1** : Menentukan keaslian dari masalah.

**Langkah 2** : Membangun sistem hierarki untuk evaluasi. Sistem hierarki dalam MADM dapat dilihat pada Gambar (3.1).

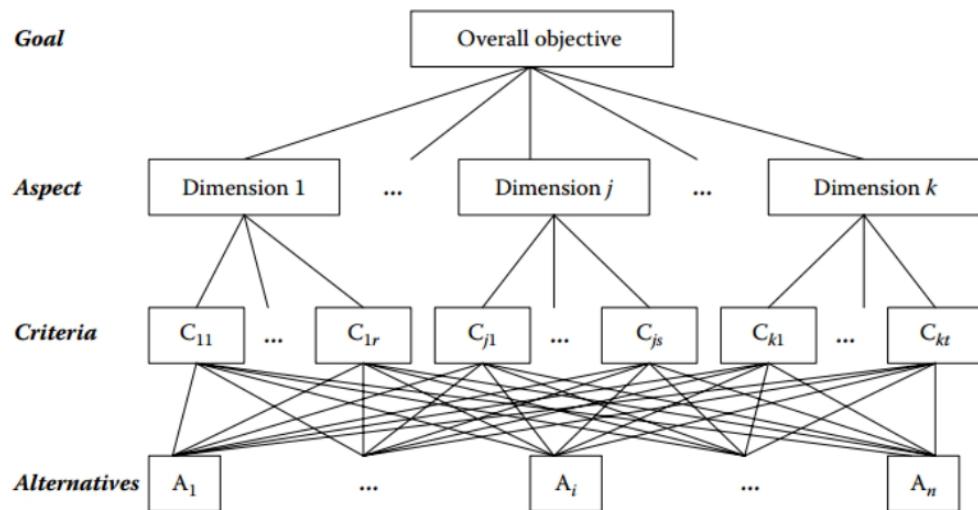
**Langkah 3** : Memilih model evaluasi yang tepat.

**Langkah 4** : Mendapatkan bobot relatif dan skor kinerja setiap atribut terhadap setiap alternatif.

**Langkah 5** : Menentukan alternatif terbaik sesuai dengan nilai-nilai utilitas sintetis, yang merupakan nilai agregasi bobot relatif dan skor kinerja sesuai dengan alternatif.

Jika skor keseluruhan alternatif kabur (*fuzzy*), dapat menambahkan Langkah 6 untuk menentukan peringkat alternatif dalam memilih yang terbaik.

**Langkah 6** : *Outranking* alternatif mengacu pada nilai-nilai utilitas sintetik di bawah lingkungan *fuzzy* dari Langkah 5.



Gambar 3.1 Sistem Hierarki untuk MADM (Dubois dan Prade, 1980)

### 3.2 Metode *Modified Delphi*

Metode Delphi merupakan proses terstruktur untuk mengumpulkan dan menyaring pengetahuan dari sekelompok ahli dengan melalui serangkaian kuisioner yang diselingi opini umpan balik (Adler dan Ziglio, 1996). Metode Delphi merupakan metode untuk penataan proses komunikasi sekelompok individu yang efektif dalam menangani masalah yang kompleks (Linstone dan Turoff, 2002). Metode Delphi merupakan perangkat komunikasi yang berguna di antara sekelompok ahli untuk memfasilitasi pembentukan keputusan kelompok (Helmer, 1977). Dari penjelasan tersebut, metode Delphi adalah proses komunikasi terstruktur dari sekelompok ahli yang efektif untuk mengumpulkan dan menyaring pengetahuan melalui serangkaian kuisioner yang diselingi opini umpan balik, sehingga menghasilkan keputusan kelompok yang dapat menangani masalah yang kompleks. Metode Delphi telah banyak digunakan untuk menghasilkan prakiraan dalam teknologi, pendidikan, dan bidang lainnya (Cornish, 1977).

Elemen kunci dalam proses Delphi adalah 1) penataan arus informasi; 2) umpan balik kepada peserta; dan 3) anonimitas bagi para peserta (Fowles, 1978). Interaksi kelompok dalam Delphi adalah anonim, dalam arti bahwa komentar,

prakiraan dan sejenisnya tidak diidentifikasi sebagai pencetusnya tetapi disajikan ke kelompok untuk dikendalikan dalam setiap identifikasi. Fowles (1978) menjelaskan sepuluh langkah dalam metode Delphi sebagai berikut:

1. Pembentukan tim untuk melakukan dan memantau Delphi pada subjek tertentu.
2. Pemilihan satu atau lebih panel untuk berpartisipasi dalam proses Delphi. Biasanya, para panelis ahli di bidang yang akan diteliti.
3. Pengembangan Delphi untuk kuesioner putaran pertama.
4. Pengujian kuesioner untuk kata-kata yang tepat (misalnya: ambiguitas dan ketidakjelasan).
5. Transmisi kuesioner putaran pertama ke panelis.
6. Analisis tanggapan putaran pertama.
7. Penyusunan kuesioner putaran kedua (dan dapat disertakan pengujian).
8. Transmisi kuesioner putaran kedua ke panelis.
9. Analisis tanggapan putaran kedua (langkah 7-9 dapat berulang sesuai yang diinginkan atau diperlukan untuk mencapai stabilitas hasil).
10. Penyusunan laporan oleh tim analis untuk mempresentasikan kesimpulan dari proses Delphi.

Masalah yang paling penting dalam proses Delphi adalah pemahaman tentang tujuan dari Delphi oleh semua peserta (Delbecq dkk., 1975). Jika tidak, panelis dapat menjawab tidak tepat. Pemilihan panelis yang tepat juga tidak kalah penting. Jumlah minimum peserta untuk memastikan kinerja kelompok yang baik tergantung pada desain studi.

Afshari dkk. (2012) dalam penelitiannya mengusulkan metode *Modified Delphi* sehingga prosesnya lebih cepat dan berkualitas. Langkah-langkah yang diusulkan yaitu sebagai berikut:

1. Prosedur untuk memilih ahli. Metode Delphi merupakan mekanisme keputusan kelompok yang membutuhkan ahli yang berkualitas dan

memiliki pemahaman yang mendalam dari masalah. Rekomendasi dari literatur Delphi, terdiri dari 5-18 orang setiap panel.

2. Studi literatur terkait untuk putaran pertama. Peneliti membuat daftar kriteria untuk masalah yang akan dipecahkan berdasarkan kajian literatur. Daftar kriteria beserta penjelasan kompetensi dikirim ke anggota panel. Anggota panel diperintahkan untuk menggunakan daftar kriteria dan kompetensi sebagai panduan dalam membuat daftar masing-masing.
3. Delphi putaran pertama untuk menghasilkan gagasan. Dari daftar kriteria berdasarkan studi literatur, para ahli diminta untuk mencentang setiap kriteria yang relevan untuk membuat keputusan dari masalah yang akan dipecahkan. Kemudian para ahli diminta untuk mengirimkan kriteria-kriteria tambahan yang belum tercantum dalam daftar kriteria beserta penjelasannya.
4. Delphi putaran kedua untuk umpan balik. Setelah Delphi putaran pertama, tanggapan dikelompokkan untuk dianalisis. Pada akhir putaran ini, kriteria yang kurang dari tujuh puluh persen dari kesepakatan dihilangkan dan menawarkan untuk penggabungan, menambahkan kriteria baru, pemisahan dan mengedit dari beberapa kriteria yang diterapkan.
5. Delphi putaran ketiga untuk mengembangkan hierarki. Tanggapan Delphi putaran kedua dianalisis, dikategorikan dan kembali ke ahli untuk mengembangkan struktur hierarki dari kriteria penilaian. Tujuan dari putaran ketiga adalah untuk memperoleh konsesus di antara para ahli untuk hierarki kriteria. Verifikasi hierarki dapat dilakukan melalui interaksi dengan pembuat keputusan. Struktur hierarki terdiri dari tingkat atas berupa tujuan pemilihan dalam suatu masalah, tingkat berikutnya berupa kriteria dan akhirnya berupa subkriteria.

### **3.3 Metode AHP (*Analytic Hierarchy Process*)**

AHP merupakan salah satu metode analisis untuk mengurai masalah MADM yang kompleks menjadi sistem hierarki. AHP digunakan untuk memilih

yang terbaik dari sejumlah alternatif yang dievaluasi menggunakan beberapa kriteria. Dalam proses ini, pembuat keputusan melakukan penilaian perbandingan berpasangan yang digunakan untuk memperoleh prioritas secara keseluruhan (Saaty dan Vargas, 2001). Prioritas alternatif dan kriteria digunakan untuk menilai alternatif. Karakteristik AHP yaitu mencakup kemampuan untuk menangani situasi keputusan yang melibatkan penilaian subjektif pembuat keputusan dan kemampuan untuk memberikan ukuran konsistensi preferensi (Triantaphyllou, 2000). AHP menyediakan sarana untuk meningkatkan konsistensi dalam pengambilan keputusan.

Proses AHP diusulkan untuk memperoleh bobot relatif sesuai dengan sistem hierarki yang tepat. Dalam memperoleh bobot relatif dapat digunakan *eigenvalue method*, *geometric mean method*, *linier programming method* dan *lambda-max method* (Huang dan Tzeng, 2011). Di antara empat metode tersebut, hanya metode *eigenvalue* yang digunakan untuk angka *crisp* dan metode lainnya digunakan untuk menangani AHP di bawah angka *fuzzy*.

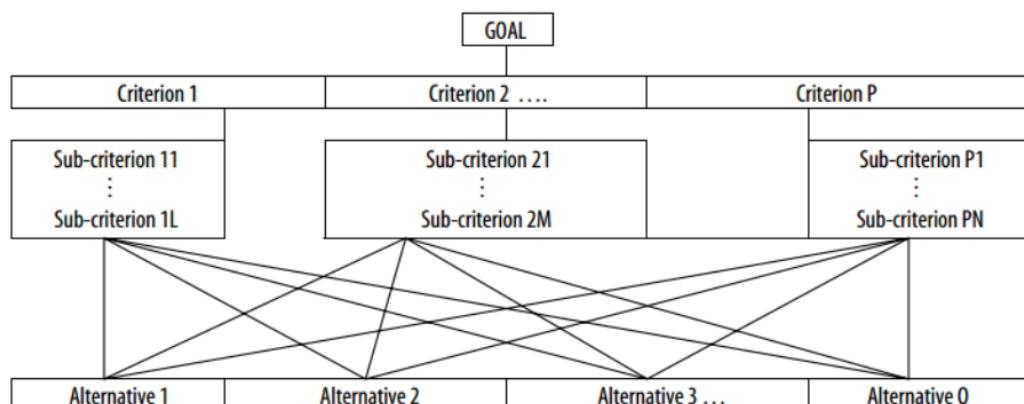
Dalam AHP semua masalah pengambilan keputusan dianggap sebagai struktur hierarki. Tingkat pertama menunjukkan tujuan untuk masalah pengambilan keputusan yang spesifik. Tingkat kedua dimaksudkan untuk menguraikan beberapa kriteria dan tingkat yang lebih rendah untuk membagi ke dalam subkriteria lainnya.

Saaty dan Vargas (2001) menjelaskan tujuh pilar dalam AHP yaitu 1) skala rasio, proporsionalitas dan skala rasio ternormalisasi; 2) perbandingan berpasangan; 3) sensitivitas dari *eigenvector* yang tepat; 4) *clustering* dan menggunakan *pivot* untuk memperpanjang skala; 5) sintesis untuk membuat rasio satu dimensi untuk mewakili hasil keseluruhan; 6) penentuan peringkat; 7) mengintegrasikan penilaian kelompok.

### 3.4 Tahapan Dalam AHP

Langkah-langkah dalam metode AHP yaitu sebagai berikut (Bhushan dan Rai, 2004):

**Langkah 1:** Masalah diurai menjadi sistem hierarki yang terdiri dari tujuan, kriteria, subkriteria dan alternatif. Langkah ini merupakan bagian yang paling penting dalam pengambilan keputusan dengan proses AHP. Hierarki menunjukkan hubungan antara elemen-elemen pada satu tingkat dengan elemen-elemen di tingkat bawahnya. Tingkat paling atas (*root*) dalam hierarki adalah sasaran atau tujuan dari masalah yang dianalisis. Tingkat paling bawah (*leaf*) merupakan alternatif untuk dibandingkan. Di antara *root* dan *leaf* merupakan kriteria dan subkriteria. Struktur umum hierarki AHP ditunjukkan pada Gambar (3.2).



Gambar 3.2 Struktur Umum Hierarki AHP (Bhushan dan Rai, 2004)

**Langkah 2:** Data dikumpulkan dari para ahli atau para pembuat keputusan sesuai dengan struktur hierarki. Para ahli/pembuat keputusan menilai perbandingan berpasangan dari alternatif dengan skala kualitatif (kuat, sangat kuat, sama kuat, sedikit kuat, dan lain-lain) seperti yang ditunjukkan Gambar (3.3).

A								X	B
Extremely strong	Very strong	Strong	Marginally strong	Equal	Marginally strong	Strong	Very strong	Extremely strong	

Gambar 3.3 Format untuk Perbandingan Berpasangan (Bhushan dan Rai, 2004)

Perbandingan yang dibuat untuk masing-masing kriteria dikonversi menjadi skala kuantitatif sesuai Tabel (3.1).

**Tabel 3.1 Skala untuk Perbandingan Kuantitatif Alternatif (Bhushan dan Rai, 2004)**

Pilihan	Nilai
Sama pentingnya ( <i>equal</i> )	1
Agak lebih penting yang satu atas lainnya ( <i>marginally strong</i> )	3
Cukup penting ( <i>strong</i> )	5
Sangat penting ( <i>very strong</i> )	7
Kepentingan yang ekstrim ( <i>extremely strong</i> )	9
Nilai tengah di antara dua nilai keputusan yang berdekatan	2, 4, 6, 8
Jika aktivitas i mempunyai nilai yang lebih tinggi dari aktivitas j, maka j mempunyai nilai berbalikan ketika dibandingkan dengan i.	Berbalikan

**Langkah 3:** Perbandingan berpasangan dari berbagai kriteria yang dihasilkan pada Langkah 2 akan disusun dalam matriks persegi. Elemen diagonal dalam matriks adalah 1.

**Langkah 4:** Mencari bobot relatif dari berbagai kriteria yang dibandingkan, dengan menghitung *eigenvalue* dan *eigenvector* dari matriks perbandingan berpasangan.

**Langkah 5:** Mengevaluasi konsistensi matriks. Perbandingan yang dibuat dalam metode AHP adalah subjektif. AHP memberikan toleransi inkonsistensi dalam jumlah tertentu. Jika indeks konsistensi pada tingkat yang diperlukan gagal untuk dicapai, maka perlu mengkaji ulang perbandingan. Indeks konsistensi/*consistency index* (CI) dapat dihitung melalui Persamaan (3.1).

$$CI = (\lambda_{\max} - n) / (n - 1) \quad (3.1)$$

dimana  $\lambda_{\text{I}}$  merupakan *eigenvalue* maksimum dari matriks penilaian dan  $n$  merupakan jumlah kriteria yang digunakan. Rasio konsistensi/*consistency ratio* (CR) dapat dihitung dengan Persamaan (3.2). Saaty menunjukkan nilai CR harus kurang dari 0,1 sehingga dapat dinyatakan konsisten.

$$CR = CI / RI \quad (3.2)$$

dimana CI merupakan *consistency index* dan RI merupakan *random consistency index* yang diperoleh dari matriks perbandingan berpasangan secara acak.

**Langkah 6:** Penilaian dari masing-masing alternatif dikalikan dengan bobot subkriteria untuk memperoleh peringkat lokal. Peringkat lokal kemudian dikalikan dengan bobot kriteria dan diperoleh peringkat global.

### 3.5 PROMETHEE (*Preference Ranking Organization Method For Enrichment Evaluation*)

Tujuan utama metode PROMETHEE yaitu metode *outranking* yang harus mudah dipahami oleh pembuat keputusan untuk memecahkan masalah MADM (Brans dan Vincke, 1985). Dalam penelitian yang dilakukan Bogdanovic dan Miletic (2014) menyatakan tahapan PROMETHEE mengikuti langkah-langkah sebagai berikut:

**Langkah 1:** Membentuk matriks untuk kriteria yang dipilih ( $j = 1 \dots n$ ) dan alternatif ( $i = 1 \dots m$ ) dengan menggunakan data kuantitatif dan kualitatif.

**Langkah 2:** Memilih dan menerapkan fungsi preferensi  $P(a, b)$ . Untuk setiap kriteria, fungsi preferensi yang dipilih  $P(a, b)$  diterapkan untuk memutuskan berapa banyak hasil  $a$  lebih disukai daripada  $b$ .

**Langkah 3:** Perhitungan indeks preferensi global  $\Pi(a, b)$  dengan Persamaan (3.3). Indeks preferensi global yang mewakili intensitas preferensi  $a$  terhadap  $b$ .

$$\pi(a, b) = \sum_{j=1}^n w_j \cdot P_j(a, b); (\sum_{j=1}^n w_j = 1) \quad (3.3)$$

dimana  $w_j$  merupakan bobot relatif dari kriteria dan  $P_j(a, b)$  merupakan fungsi preferensi.

**Langkah 4:** Perhitungan *outranking flow* untuk setiap alternatif  $a$ .  $A$  terdiri dari *positive outranking flow* dan *negative outranking flow*. *Positive outranking flow*

dapat dihitung dengan Persamaan (3.4) dan *negative outranking flow* dapat dihitung dengan Persamaan (3.5).

1. *Positive outranking flow (leaving flow)*:

$$\Phi^+(a) = 1/(m-1) \cdot \sum_{(x \neq a)} \pi(a, x) \quad (3.4)$$

2. *Negative outranking flow (entering flow)*:

$$\Phi^-(a) = 1/(m-1) \cdot \sum_{(x \neq a)} \pi(x, a) \quad (3.5)$$

dimana  $\pi(a, x)$  dan  $\pi(x, a)$  merupakan indeks preferensi global dan m merupakan jumlah alternatif yang digunakan.

#### **Langkah 5:** PROMETHEE I – peringkat parsial

Dari hasil perhitungan *leaving flow* dan *entering flow*, maka langkah selanjutnya adalah melakukan penyusunan peringkat parsial pada PROMETHEE I. Suatu alternatif dikatakan mempunyai peringkat (urutan) paling tinggi jika nilai *leaving flow*-nya lebih besar dibandingkan dengan alternatif lainnya dan nilai *entering flow*-nya lebih kecil dibandingkan dengan alternatif lainnya. Hasil perhitungan PROMETHEE I menghasilkan bentuk hubungan antara setiap alternatif. Cara mengetahui bentuk hubungan ini dengan membandingkan *outranking flow* seperti pada Persamaan (3.6). Jika terdapat urutan alternatif yang tidak dapat dibandingkan (*incomparable*), maka dilanjutkan dengan perhitungan PROMETHEE II.

$$\left\{ \begin{array}{ll} aP^I b & \text{if } \begin{cases} \Phi^+(a) > \Phi^+(b) & \text{and} \\ \Phi^-(a) < \Phi^-(b) & \end{cases} \\ aI^I b & \text{if } \Phi^+(a) = \Phi^+(b) \text{ and } \Phi^-(a) = \Phi^-(b) \\ aRb & \text{otherwise} \end{array} \right. \quad (3.6)$$

(P, I, R mewakili preferensi, pengabaian dan yang tidak dapat dibandingkan).

#### **Langkah 5:** Perhitungan *net flow* dengan Persamaan (3.7)

$$\Phi(a) = \Phi^+(a) - \Phi^-(a) \quad (3.7)$$

dimana  $\Phi^+(a)$  merupakan *leaving flow* dan  $\Phi^-(a)$  merupakan *entering flow*.

**Langkah 6:** PROMETHEE II – peringkat lengkap

Jika pada PROMETHEE I urutan alternatif tidak diperoleh, maka dilanjutkan pada PROMETHEE II. PROMETHEE II memberikan peringkat lengkap alternatif dengan mempertimbangkan nilai *net flow*.

## **BAB IV**

### **ANALISIS DAN RANCANGAN SISTEM**

#### **4.1 Analisis Data**

Berdasarkan pengumpulan data melalui wawancara, diperoleh informasi proses seleksi peserta pelatihan di BLK Bantul dan data seleksi peserta pelatihan. Tahapan dalam proses seleksi yaitu calon peserta mendaftar pelatihan yang ingin diikuti dengan mengisi identitas diri, mengikuti seleksi tertulis dan wawancara sesuai dengan jadwal yang sudah ditentukan, pengecekan referensi dan pembuatan keputusan seleksi. Data seleksi peserta yang digunakan sebagai data uji yaitu data subkejuruan Teknik Sepeda Motor II Institusional APBN tahun 2016, yang terdiri dari identitas pendaftar, hasil penilaian seleksi tertulis dan wawancara serta daftar peserta seleksi pelatihan yang diterima. Tabel 4.1 menunjukkan identitas pendaftar dan Tabel 4.2 menunjukkan hasil penilaian pendaftar. Penamaan US, PT, PP, IK, KT, PG, KB, KS, PO, PE, KO, PD, KK, PM, PK, ME, MO, SI, KJ, RK dan RS sudah disesuaikan dengan penamaan kriteria yang ditentukan pada proses *Modified-Delphi*.

**Tabel 4.1 Identitas pendaftar**

No.	Pendaftar	US (tahun)	PT	PP	IK
1	P1	21	-	-	Belum pernah
2	P2	21	SMP	-	Belum pernah
3	P3	20	SMK	Bengkel	Belum pernah
4	P4	20	SMK	Bengkel	Belum pernah
5	P5	18	SMP	-	Belum pernah
6	P6	22	SMA	-	Belum pernah
7	P7	24	SMK	Stir mobil, bengkel	Belum pernah
8	P8	19	SMK	-	Belum pernah
9	P9	21	SMK	-	Belum pernah

**Tabel 4.1 (lanjutan)**

No.	Pendaftar	US (tahun)	PT	PP	IK
10	P10	18	SD	Bengkel	Belum pernah
11	P11	18	SMP	-	Belum pernah
12	P12	18	SD	-	Belum pernah
13	P13	23	SMA	-	Belum pernah
14	P14	21	SMK	-	Belum pernah
15	P15	20	SMK	-	Belum pernah
16	P16	20	SMK	Desain grafis	Belum pernah
17	P17	26	SMK	-	Belum pernah
18	P18	21	SMK	-	Belum pernah
19	P19	18	SMP	-	Belum pernah
20	P20	24	SMK	-	Belum pernah
21	P21	19	SMK	-	Belum pernah
22	P22	20	SMK	-	Belum pernah
23	P23	20	SMK	-	Belum pernah
24	P24	20	SMK	-	Belum pernah
25	P25	20	SMK	-	Belum pernah
26	P26	20	SMK	-	Belum pernah
27	P27	20	SMK	-	Belum pernah
28	P28	17	SD	-	Belum pernah
29	P29	20	SMK	-	Belum pernah
30	P30	39	SMA	-	Belum pernah
31	P31	21	SMK	-	Belum pernah
32	P32	20	SMK	-	Belum pernah

Proses penilaian seleksi belum berdasarkan standar kriteria tertentu, karena BLK Bantul belum memiliki ketetapan standar kriteria yang digunakan dalam seleksi peserta pelatihan. Keputusan seleksi didominasi oleh hasil wawancara dan belum berdasarkan kalkulasi hasil dari seluruh proses seleksi. Hal ini membutuhkan perumusan kriteria. Berdasarkan pengumpulan data melalui studi literatur, diperoleh data kriteria seleksi personil secara umum, yang ditunjukkan pada Tabel 4.3. Setiap kriteria mempunyai tujuan masing-masing. Data kriteria ini digunakan dalam proses metode *Modified Delphi*.

## Tabel 4.2 Hasil penilaian pendaftar

**Tabel 4.2 (lanjutan)**

**Tabel 4.3 Kriteria seleksi personil dari studi literatur**

Situs Kriteria	(Noe dkk., 2012)	(Ivancevich dan Konopaske, 2013)	(Bogdanovic dan Miletic, 2014)	(Kumar dkk., 2013)	(Rouyendegh dan Erkan, 2013)	(El-santawy, 2012)	(Blume dkk., 2010)	(Robertson dan Smith, 2001)	(Güngör dkk., 2009)	(Fajar dan Heru, 2010)	(Safrizal dan Tanti, 2015)	(Ubaidi, 2015)	(Hough dkk., 2001)	(Hough dan Oswald, 2000)	(Kelemenis dan Askounis, 2010)	(Vinchur dkk., 1998)	(Chen dkk., 2013)	(Barrett dkk., 2003)
Pengalaman pelatihan	•	•	•	•	•	•	•	•										
Orientasi masa depan	•								•									
Rekomendasi	•	•																
Kesehatan	•								•		•	•						
Kepribadian	•	•							•		•							
Pengetahuan	•		•				•		•		•	•		•				
Keterampilan teknis	•	•		•	•						•							
Kejujuran	•	•							•									
Bebas obat-obatan terlarang	•																	
Sikap	•	•								•	•							
Pendidikan formal	•	•					•	•	•			•			•			
Pengalaman pekerjaan	•	•												•				
Jenis kelamin	•	•						•					•					
Usia	•	•				•	•	•	•		•		•		•			
Kefasihan bahasa asing	•		•								•			•		•		

**Tabel 4.3 (lanjutan)**

## 4.2 Pertimbangan Penggunaan Metode *Modified Delphi-AHP-PROMETHEE*

Pertimbangan pemilihan metode merupakan salah satu langkah penting dalam pembuatan model sistem pendukung keputusan untuk memecahkan suatu masalah. Berdasarkan hasil analisis data, pertimbangan metode yang digunakan untuk memecahkan masalah seleksi peserta pelatihan di BLK Bantul yaitu penggabungan metode *Modified Delphi-AHP-PROMETHEE*. Penilaian yang diharapkan untuk seleksi peserta pelatihan di BLK Bantul yaitu berdasarkan pembobotan kriteria karena setiap kriteria memiliki prioritas yang berbeda. Maka dari itu metode yang dapat digunakan dalam pembobotan ini yaitu AHP. Pertimbangan penggunaan metode AHP juga didasari karena AHP melakukan proses uji konsistensi, sehingga bobot yang dihasilkan konsisten.

Namun, dalam AHP belum dijelaskan cara menentukan kriteria, sehingga digunakan metode *Modified Delphi* untuk merumuskan kriteria yang berpengaruh. Pertimbangan penggunaan metode *Modified Delphi* didasari karena perumusan kriteria dilakukan melalui studi literatur dan pertimbangan ahli. Selain itu, hasil akhir dari metode *Modified Delphi* yaitu hierarki kriteria yang dapat digunakan dalam metode AHP.

Metode AHP membutuhkan gabungan dengan metode *outranking*, karena dalam AHP belum ada *Min/Max* atau *Benefit/Cost* sehingga digunakan metode PROMETHEE. Nilai *Min/Max* menunjukkan kecenderungan data, dimana jika kecenderungannya *Min* maka menunjukkan nilai yang semakin kecil adalah semakin baik, sedangkan jika *Max* maka nilai yang semakin tinggi adalah semakin baik. *Min/max* digunakan untuk menentukan fungsi preferensi. PROMETHEE menentukan fungsi preferensi untuk setiap kriteria yang paling cocok didasarkan pada data dan pertimbangan pembuat keputusan, sehingga diharapkan menghasilkan keputusan seleksi yang tepat. Penentuan peringkat alternatif dalam PROMETHEE terdiri dari dua tahap, yaitu peringkat parsial dan peringkat lengkap.

### 4.3 Analisis Kebutuhan Sistem

Sistem terdiri dari dua bagian, yaitu untuk pendaftaran seleksi dan untuk penilaian seleksi. Pendaftaran seleksi digunakan sebagai masukan sistem yang mana merupakan objek yang akan diseleksi. Penilaian seleksi merupakan implementasi pemodelan metode *Modified Delphi-AHP-PROMETHEE* yang digunakan untuk menilai objek yang diseleksi. Hasil dari penilaian digunakan sebagai bahan pertimbangan dalam mengambil keputusan seleksi peserta pelatihan. Pembahasan sistem lebih difokuskan pada bagian penilaian seleksi.

Sistem untuk pendaftaran seleksi digunakan oleh pendaftar. Sedangkan sistem untuk penilaian seleksi digunakan kepala BLK Bantul, kepala sub-bagian tata usaha, koordinator instruktur, kepala kejuruan dan staf. Semua pengguna dapat menggunakan sistem dengan melakukan login. Hal ini dimaksudkan untuk keamanan. Pendaftar yang belum mempunyai akun, wajib melakukan registrasi sistem. Sedangkan untuk kepala BLK Bantul, kepala sub-bagian tata usaha, koordinator instruktur, kepala kejuruan dan staf tidak perlu melakukan registrasi. Kebutuhan fungsional dari sistem yang dibangun digambarkan dalam *use case diagram* pada Gambar 4.1. *Use case diagram* menjelaskan interaksi antara aktor. Penjelasan *use case* ditunjukkan pada Tabel 4.4.

**Tabel 4.4 Deskripsi *use case***

No.	Use Case	Deskripsi
1	Mendaftar seleksi	Pendaftar mendaftar seleksi dengan memasukkan profil kemudian memilih pelatihan apa yang ingin diikuti. Pendaftar hanya diijinkan mendaftar jika belum pernah mengikuti pelatihan atau sudah pernah mengikuti pelatihan dan proses penilaian sudah selesai.
2	Mengelola data pengguna	Staf mengelola data pengguna yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data pengguna.
3	Mengelola data role	Staf mengelola data <i>role</i> yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data <i>role</i> .
4	Mengelola data kejuruan	Staf mengelola data kejuruan yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data kejuruan.
5	Mengelola data	Staf mengelola data subkejuruan yang meliputi lihat semua,

**Tabel 4.4 (lanjutan)**

No.	<i>Use Case</i>	Deskripsi
	subkejuruan	lihat detail, tambah, edit dan hapus data subkejuruan.
6	Mengelola data pendidikan	Staf mengelola data pendidikan yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data pendidikan.
7	Mengelola data kursus	Staf mengelola data kursus yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data kursus.
8	Mengelola jadwal seleksi	Staf mengelola data jadwal seleksi yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data jadwal seleksi.
9	Melihat jadwal seleksi	Kepala, kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat data jadwal seleksi.
10	Melihat data pendaftar	Staf, kepala, kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat data pendaftar sesuai yang dimasukkan oleh pendaftar.
11	Mengelola nilai seleksi	Kepala kejuruan mengelola data nilai seleksi yang meliputi lihat semua, lihat detail dan tambah data nilai seleksi.
12	Melihat nilai seleksi	Kepala dan koordinator Instruktur melihat data nilai seleksi.
13	Mengelola kriteria pustaka	Staf/peneliti mengelola data kriteria dari kajian pustaka yang meliputi lihat semua, lihat detail, tambah, edit dan hapus data kriteria.
14	Mengisi kuesioner kriteria	Kepala, kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur mengisi kuesioner kriteria. Daftar kriteria merupakan kriteria dari kajian pustaka. Aktor mengisi sesuai atau tidak sesuai, serta dapat memberikan usulan kriteria. Isian yang sudah di- <i>submit</i> tidak dapat diubah, hanya dapat dilihat.
15	Melihat hasil kriteria tahap 1	Kepala, kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat hasil kriteria tahap 1. Hasil kriteria tahap 1 merupakan kalkulasi hasil isian kuesioner kriteria dari semua aktor. Hasil kriteria tahap 1 hanya bisa dilihat jika sudah mengisi kuesioner kriteria.
16	Mengelola kriteria tahap 2	Kepala mengelola data kriteria tahap 2 yang meliputi lihat semua data kriteria tahap 1 dan 2, menggunakan kriteria pada tahap 1, tambah, edit dan hapus.
17	Melihat kriteria tahap 2	Kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat semua data kriteria tahap 1 dan 2.
18	Mengelola hierarki kriteria	Kriteria yang sudah dirumuskan pada tahap 2 kemudian dikelompokkan sehingga membentuk suatu hierarki kriteria. Kepala mengelola hierarki kriteria yang meliputi melihat hierarki kriteria, tambah kriteria ke kelompok, hapus kriteria

**Tabel 4.4 (lanjutan)**

No.	<i>Use Case</i>	Deskripsi
		dari kelompok, tambah, edit dan hapus kelompok.
19	Melihat hierarki kriteria	Kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat hierarki kriteria.
20	Mengelola bobot kriteria	Kepala mengelola bobot kriteria. Bobot kriteria dihitung dengan metode AHP. Kepala memasukkan nilai perbandingan berpasangan kriteria/subkriteria, kemudian dihitung bobot parsial dan global.
21	Mengelola tipe preferensi	Kepala mengelola tipe preferensi dan kaidah max/min untuk setiap kriteria.
22	Mengelola parameter kriteria	Kepala mengelola parameter kriteria. Pengelolaan parameter kriteria bersifat opsional tergantung tipe preferensi yang digunakan.
23	Mengelola nilai alternatif	Kepala mengelola data nilai alternatif sesuai dengan parameter yang sudah ditetapkan. Setelah semua alternatif dilakukan penilaian, sistem akan menentukan peringkat menggunakan metode PROMETHEE, yaitu dengan menghitung <i>leaving flow</i> , <i>entering flow</i> dan <i>net flow</i> .
24	Melihat hasil ranking	Kepala, kepala sub-bagian tata usaha, kepala kejuruan dan koordinator instruktur melihat hasil peringkat alternatif yang diperoleh dengan metode PROMETHEE.

#### 4.4 Perancangan Sistem

Perancangan sistem merupakan langkah selanjutnya setelah melakukan analisis kebutuhan. Perancangan sistem digunakan untuk memberikan gambaran tentang aliran aktivitas sistem dan menghasilkan rancangan bangun sistem. Aliran aktivitas sistem ini digambarkan dengan *activity diagram*. *Activity diagram* dibuat berdasarkan satu atau beberapa *use case* dalam *use case diagram*. Pembahasan perancangan sistem difokuskan pada bagian implementasi pemodelan *Modified Delphi-AHP-PROMETHEE*. Berikut ini *activity diagram* yang menggambarkan aliran aktivitas dalam sistem pada bagian implementasi pemodelan *Modified Delphi-AHP-PROMETHEE*.

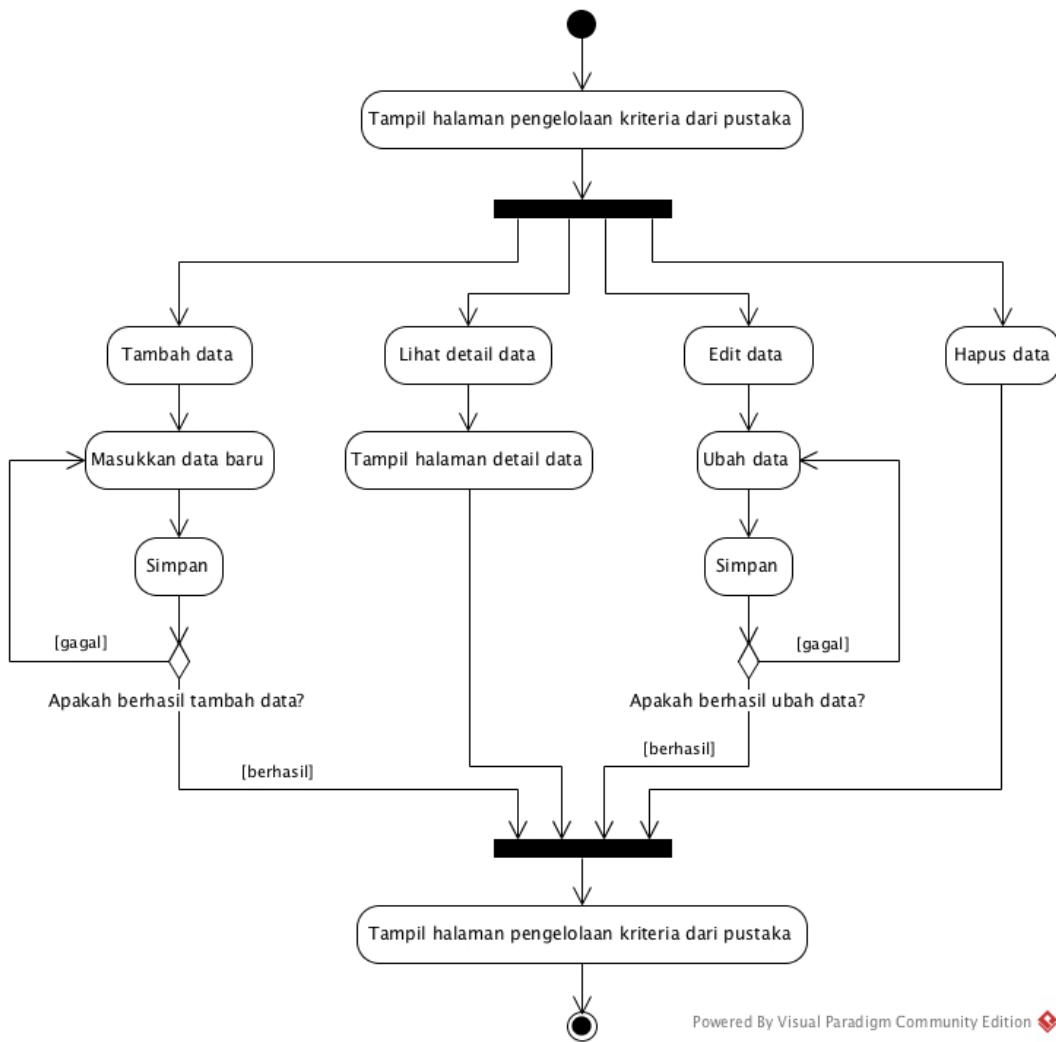
1. *Activity diagram* mengelola kriteria pustaka

*Activity diagram* mengelola kriteria pustaka menggambarkan aliran aktivitas untuk pengelolaan kriteria yang bersumber dari studi pustaka, yang

terdiri dari aktivitas paralel yaitu menambah data kriteria baru, melihat detail data kriteria, mengubah data kriteria dan menghapus data kriteria. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran pertama. *Activity diagram* mengelola kriteria pustaka ditunjukkan pada Gambar 4.2.



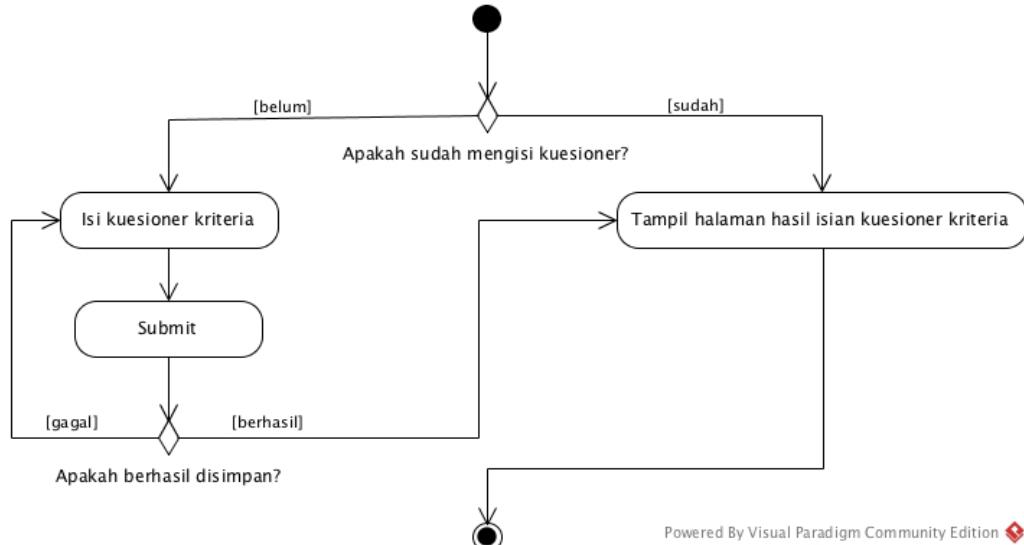
Gambar 4.1 *Use case diagram* sistem



**Gambar 4.2 Activity diagram mengelola kriteria pustaka**

## 2. Activity diagram mengisi kuesioner kriteria

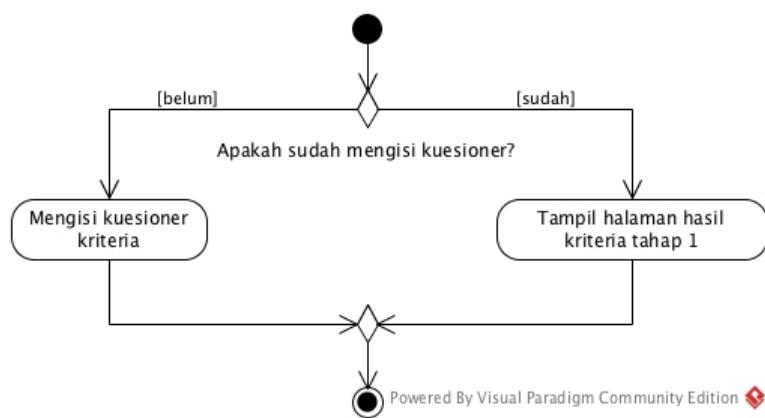
*Activity diagram* mengisi kuesioner kriteria menggambarkan aliran aktivitas untuk pengisian kuesioner kriteria. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menampilkan isian kuesioner. Jika sudah mengisi kuesioner kriteria, maka akan menampilkan hasil isian kuesioner kriteria. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran pertama. *Activity diagram* mengisi kuesioner kriteria ditunjukkan pada Gambar 4.3.



**Gambar 4.3 Activity diagram mengisi kuesioner kriteria**

### 3. *Activity diagram* melihat hasil kriteria tahap 1

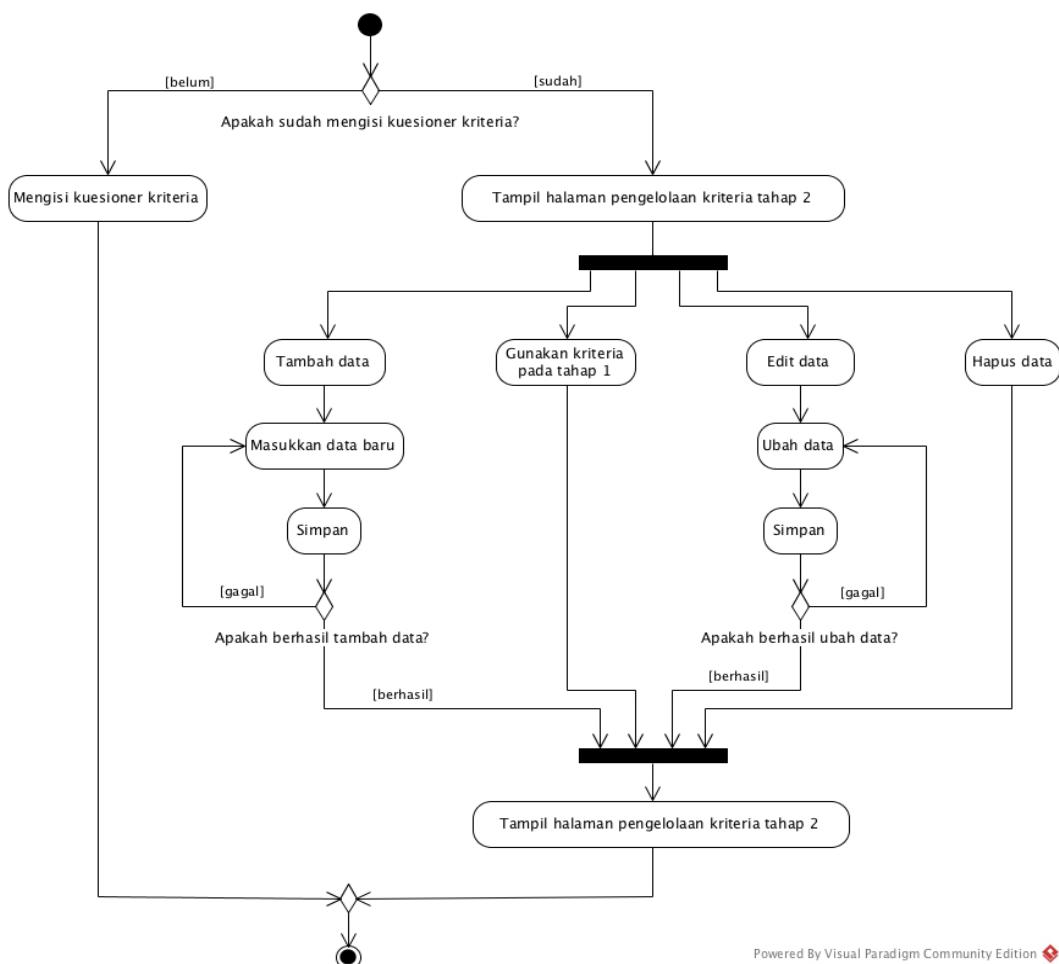
*Activity diagram* melihat hasil kriteria tahap 1 menggambarkan aliran aktivitas untuk melihat hasil kriteria tahap 1. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menjalankan aliran aktivitas untuk mengisi kuesioner kriteria. Jika sudah mengisi kuesioner kriteria, maka akan menampilkan hasil kriteria tahap 1. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran pertama. *Activity diagram* melihat hasil kriteria tahap 1 ditunjukkan pada Gambar 4.4.



**Gambar 4.4 Activity diagram melihat hasil kriteria tahap 1**

#### 4. *Activity diagram* mengelola kriteria tahap 2

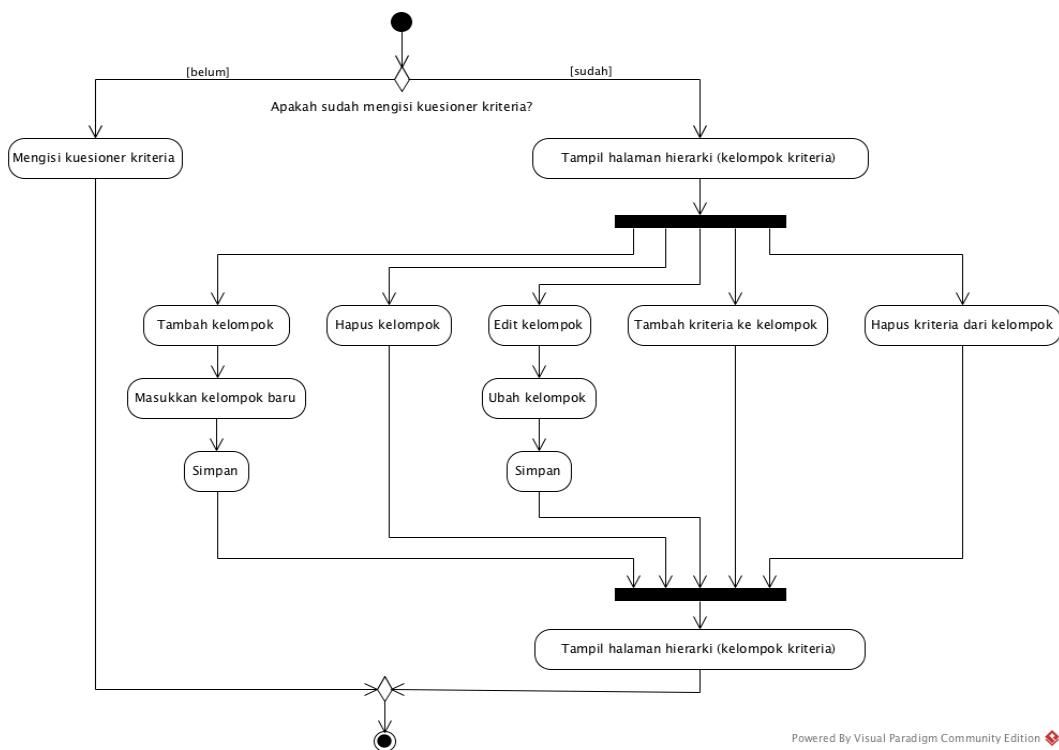
*Activity diagram* mengelola kriteria tahap 2 menggambarkan aliran aktivitas untuk pengelolaan kriteria tahap 2. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menjalankan aliran aktivitas untuk mengisi kuesioner kriteria. Jika sudah mengisi kuesioner kriteria, maka akan menampilkan halaman untuk mengelola kriteria tahap 2, yang terdiri dari aktivitas paralel yaitu menambah data kriteria baru pada tahap 2, mengubah data kriteria pada tahap 2, menggunakan kriteria pada tahap 1 dan menghapus data kriteria pada tahap 2. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran kedua. *Activity diagram* mengelola kriteria tahap 2 ditunjukkan pada Gambar 4.5.



Gambar 4.5 *Activity diagram* mengelola kriteria tahap 2

### 5. *Activity diagram* mengelola hierarki kriteria

*Activity diagram* mengelola hierarki kriteria menggambarkan aliran aktivitas untuk pengelolaan hierarki kriteria. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menjalankan aliran aktivitas untuk mengisi kuesioner kriteria. Jika sudah mengisi kuesioner kriteria, maka akan menampilkan halaman untuk mengelola hierarki (kelompok kriteria), yang terdiri dari aktivitas paralel yaitu menambah kelompok, mengubah kelompok, menghapus kelompok, menambah kriteria ke dalam kelompok dan menghapus kriteria dari kelompok. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran ketiga. *Activity diagram* mengelola hierarki kriteria ditunjukkan pada Gambar 4.6.

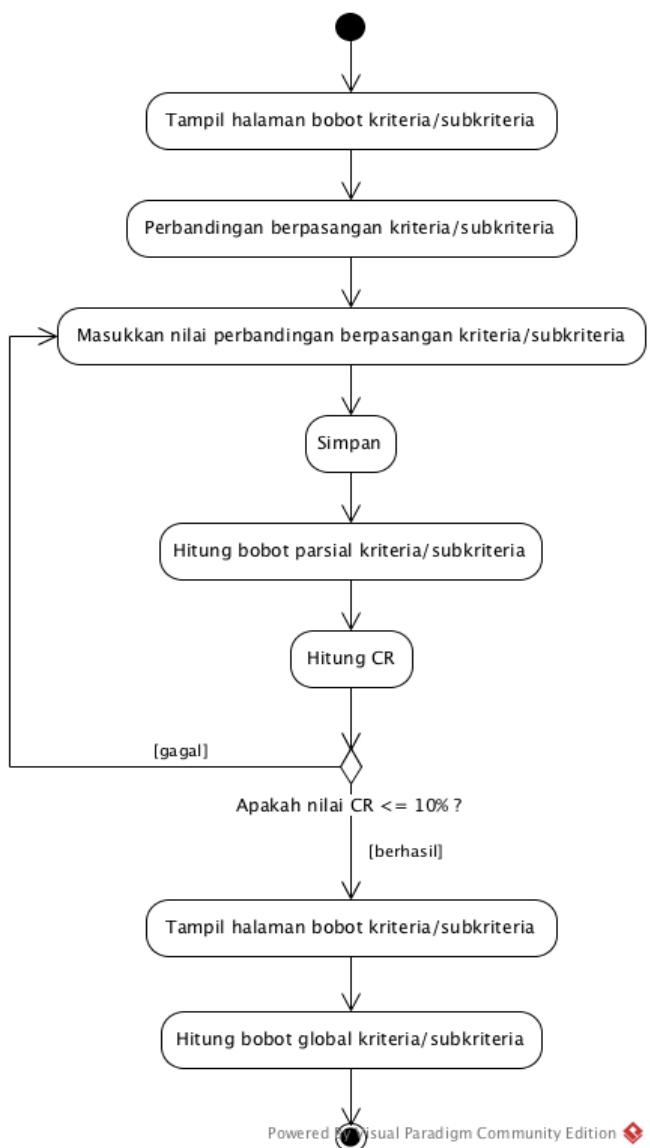


Gambar 4.6 *Activity diagram* mengelola hierarki kriteria

### 6. *Activity diagram* mengelola bobot kriteria

*Activity diagram* mengelola bobot kriteria menggambarkan aliran aktivitas untuk pengelolaan bobot kriteria. Aktivitas diawali dengan mengelola bobot

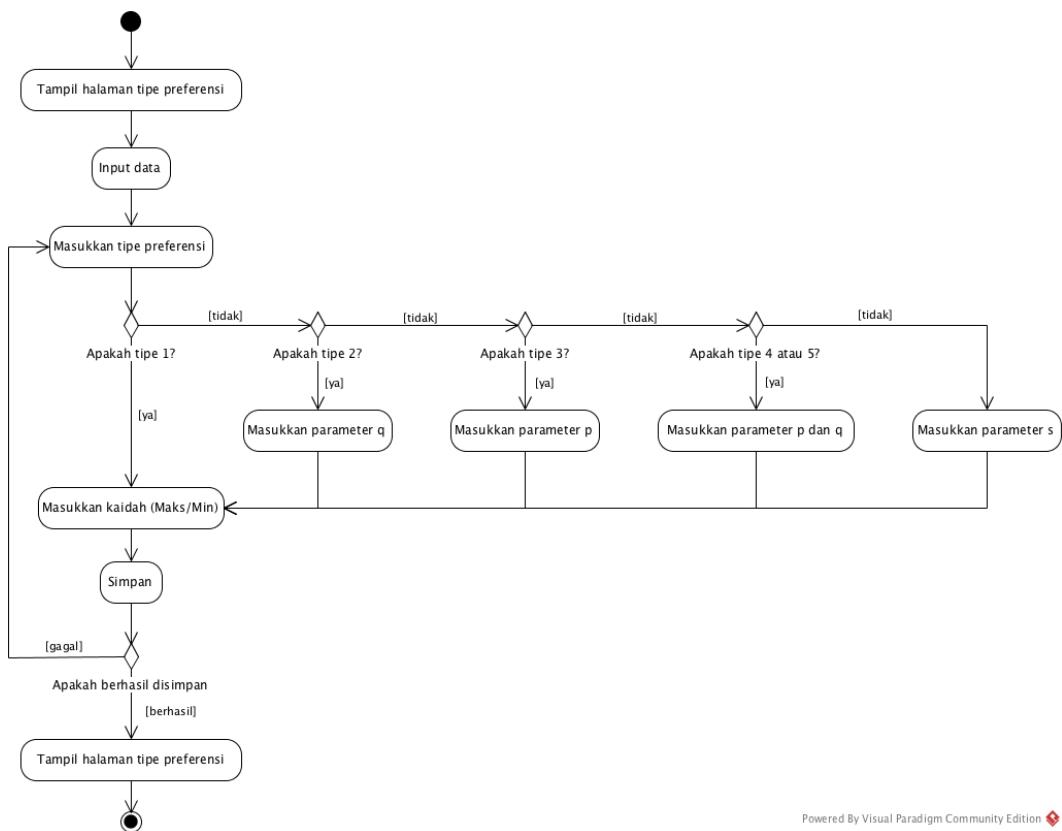
kriteria. Kemudian melakukan perbandingan berpasangan kriteria/subkriteria dengan memasukkan nilai perbandingan berpasangan. Dari perbandingan berpasangan dapat dihitung bobot parsial. Kemudian dihitung nilai CR. Jika nilai CR > 10%, maka harus melakukan perbandingan berpasangan ulang hingga nilai CR <=10%. Setelah diperoleh bobot parsial, maka dapat dihitung bobot global. Aliran aktivitas ini mengimplementasikan metode AHP. *Activity diagram* mengelola bobot kriteria ditunjukkan pada Gambar 4.7.



**Gambar 4.7 Activity diagram mengelola bobot kriteria**

### 7. Activity diagram mengelola tipe preferensi

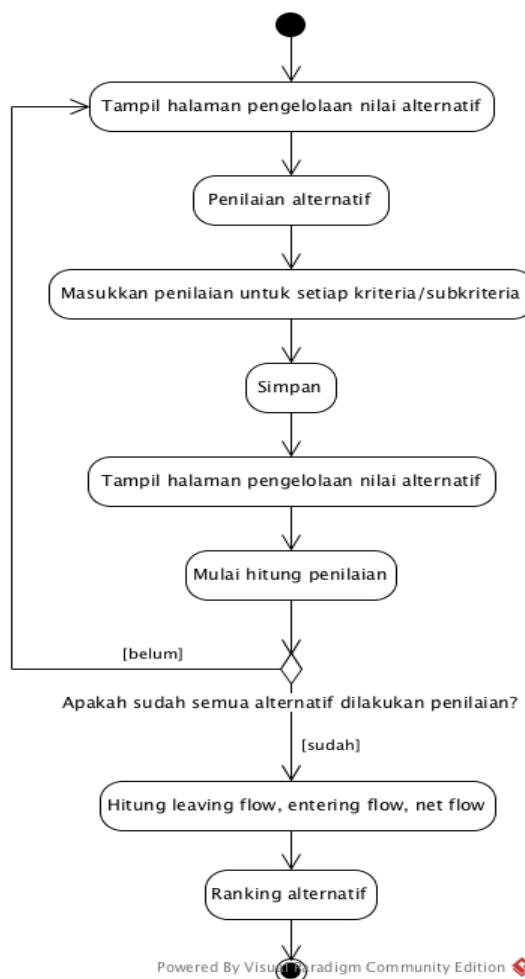
*Activity diagram* mengelola tipe preferensi menggambarkan aliran aktivitas untuk pengelolaan tipe preferensi yaitu dengan memasukkan data berupa tipe preferensi, parameter dan kaidah max/min. Namun parameter ini opsional tergantung tipe preferensi yang dipilih. Jika memasukkan tipe 1, maka tidak memiliki parameter, hanya memasukkan tipe preferensi dan kaidah max/min. Jika memasukkan tipe 2, maka perlu memasukkan nilai parameter q. Jika memasukkan tipe 3, maka perlu memasukkan nilai parameter p. Jika memasukkan tipe 4 dan 5, maka perlu memasukkan nilai parameter p dan q. Jika memasukkan tipe 6, maka perlu memasukkan nilai parameter s. Aliran aktivitas ini mengimplementasikan metode PROMETHEE. *Activity diagram* mengelola tipe preferensi ditunjukkan pada Gambar 4.8.



Gambar 4.8 *Activity diagram* mengelola tipe preferensi

### 8. *Activity diagram* mengelola nilai alternatif

*Activity diagram* mengelola nilai alternatif menggambarkan aliran aktivitas untuk pengelolaan nilai alternatif. Aktivitas penilaian alternatif dilakukan dengan memasukkan penilaian untuk setiap kriteria/subkriteria. Setelah nilai alternatif dimasukkan, dilakukan hitung penilaian. Sebelum dihitung, akan dicek apakah sudah semua alternatif dilakukan penilaian untuk semua kriteria/subkriteria atau belum. Jika belum, maka harus memasukkan nilai alternatif. Jika sudah semua dinilai, maka dapat dihitung untuk nilai *leaving flow*, *entering flow* dan *net flow*. Dari perhitungan dapat ditentukan peringkat alternatif. Aliran aktivitas ini mengimplementasikan metode PROMETHEE. *Activity diagram* mengelola nilai alternatif ditunjukkan pada Gambar 4.9.



Gambar 4.9 *Activity diagram* mengelola nilai alternatif

#### 9. *Activity diagram* melihat hasil ranking

*Activity diagram* melihat hasil ranking menggambarkan aliran aktivitas untuk melihat hasil peringkat. Hasil peringkat ini merupakan keluaran dari aliran aktivitas mengelola nilai alternatif. Aliran aktivitas ini merupakan hasil akhir dari implementasi metode *Modified Delphi-AHP-PROMETHEE*. *Activity diagram* mengelola nilai alternatif ditunjukkan pada Gambar 4.10.



**Gambar 4.10 *Activity diagram* melihat hasil ranking**

### 4.5 Perancangan Basis Data

Perancangan basis data merupakan proses untuk membangun sebuah basis data yang sesuai aturan bisnis sistem. Perancangan basis data bertujuan untuk memenuhi kebutuhan data, mengatasi redundansi data, menghindari terjadinya inkonsistensi data, mengatasi kesulitan dalam mengakses data, menyusun format yang standar dari sebuah data dan menyusun independensi data.

Data-data yang digunakan dalam sistem pendukung keputusan seleksi peserta pelatihan di BLK Bantul disimpan dalam sebuah basis data. Perancangan basis data terdiri dari pembuatan ERD dan implementasi basis data.

#### 4.5.1 ERD (*Entity Relationship Diagram*)

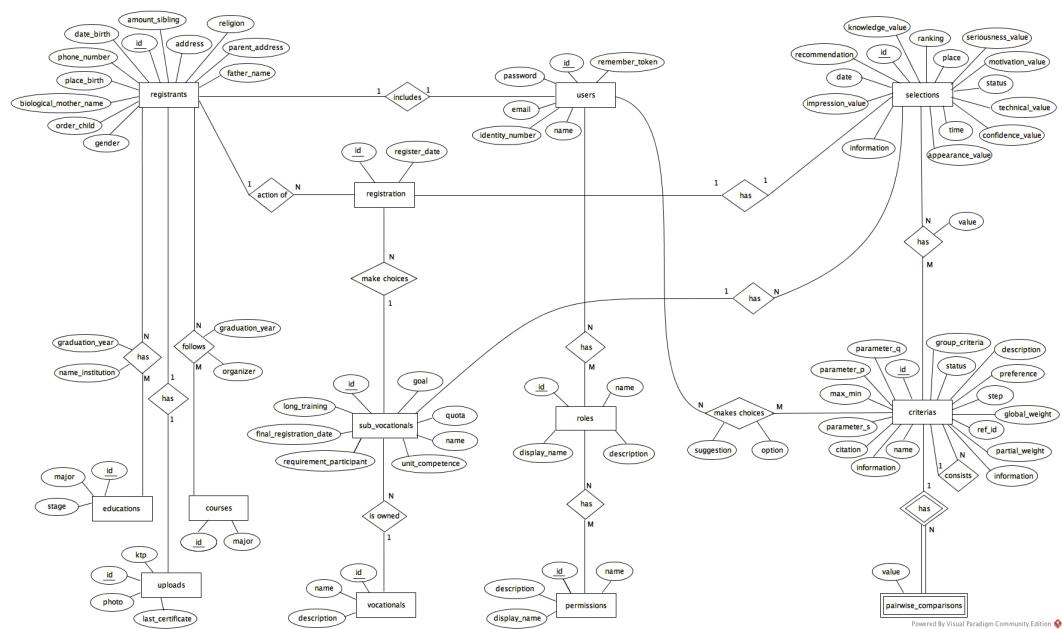
ERD digunakan untuk memodelkan kebutuhan data dari suatu sistem yang menggambarkan hubungan antara entitas dan relasi di antara entitas tersebut. Dalam membangun ERD dibutuhkan aturan bisnis sistem. Berikut ini aturan bisnis sistem pendukung keputusan seleksi peserta pelatihan:

1. Satu pendaftar bisa memiliki satu atau banyak jenjang pendidikan.

2. Satu jenjang pendidikan bisa dimiliki oleh satu atau banyak pendaftar.
3. Satu pendaftar bisa memiliki satu atau banyak bidang pelatihan.
4. Satu bidang pelatihan bisa dimiliki oleh satu atau banyak pendaftar.
5. Satu pendaftar hanya bisa memiliki satu data upload.
6. Satu data upload hanya bisa dimiliki oleh satu pendaftar.
7. Satu pendaftar hanya bisa memiliki satu akun pengguna.
8. Satu akun pengguna hanya bisa dimiliki satu pendaftar.
9. Satu akun pengguna bisa memiliki satu atau banyak *role*.
10. Satu *role* bisa dimiliki oleh satu atau banyak akun.
11. Satu *role* bisa memiliki satu atau banyak *permission*.
12. Satu *permission* bisa dimiliki oleh satu atau banyak *role*.
13. Satu pendaftar bisa melakukan satu atau beberapa pendaftaran.
14. Satu pendaftaran hanya bisa dilakukan oleh satu pendaftar.
15. Satu pendaftaran hanya bisa memilih satu subkejuruan.
16. Satu subkejuruan bisa dipilih oleh satu atau beberapa pendaftaran.
17. Satu subkejuruan hanya bisa masuk dalam satu kejuruan.
18. Satu kejuruan bisa mencakup satu atau beberapa subkejuruan.
19. Satu subkejuruan bisa memiliki satu atau banyak seleksi.
20. Satu seleksi hanya bisa dimiliki oleh satu subkejuruan.
21. Satu akun pengguna bisa memilih satu atau banyak kriteria.
22. Satu kriteria bisa dipilih oleh satu atau banyak akun pengguna.
23. Satu seleksi bisa memiliki satu atau banyak kriteria.
24. Satu kriteria bisa dimiliki oleh satu atau banyak seleksi.
25. Satu kriteria bisa memiliki satu atau banyak perbandingan berpasangan.
26. Satu perbandingan berpasangan hanya bisa dimiliki oleh satu kriteria.
27. Satu kriteria hanya bisa dicakup oleh satu kelompok kriteria.

28. Satu kelompok kriteria bisa mencakup satu atau banyak kriteria.

Berdasarkan entitas dan hubungan antara entitas yang telah dijelaskan dalam aturan bisnis, dapat dibangun ERD yang ditunjukkan pada Gambar 4.11. Entitas yang berhubungan langsung dengan implementasi metode *Modified Delphi-AHP-PROMETHEE* yaitu criterias, pairwise\_comparisons, selections dan users.



**Gambar 4.11 Entity Relationship Diagram (ERD)**

#### **4.5.2 Implementasi basis data**

Berdasarkan ERD pada Gambar 4.11, dapat diimplementasi ke dalam tabel basis data yang ditunjukkan pada Gambar 4.12. Tabel-tabel dari basis data tersebut saling berelasi antara tabel. Gambar 4.12 merupakan implementasi basis data yang sudah dinormalisasi. Hubungan antara tabel sub\_vocationals dan selections memiliki ketergantungan parsial, sehingga menerapkan normalisasi 2NF. Atribut date, time, place dan information pada tabel selections hanya tergantung kepada atribut sub\_vocational\_id dan tidak tergantung pada atribut registrant\_id, sehingga menggunakan tabel bantuan yaitu tabel selection\_schedules. Adapun struktur tabel dari basis data pada sistem ini yaitu

sebagai berikut:

### 1. Tabel users

Tabel users digunakan untuk menyimpan data akun pengguna. Struktur tabel users ditunjukkan pada Tabel 4.5.

**Tabel 4.5 Struktur tabel users**

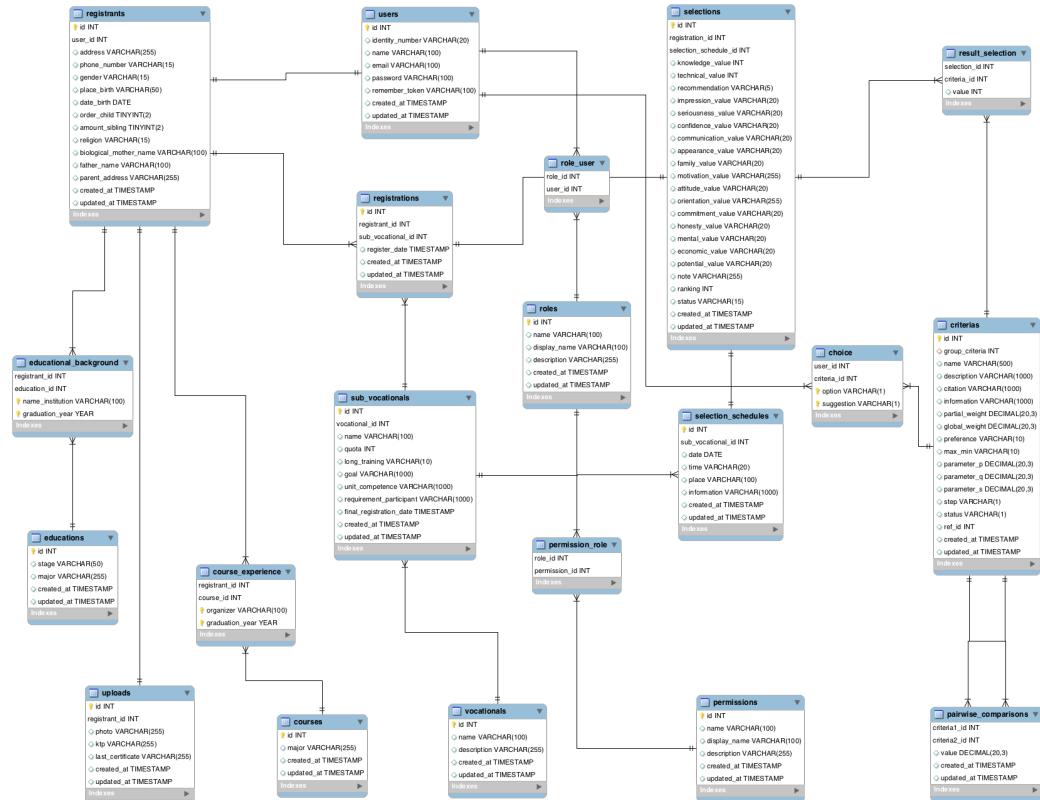
Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id pengguna dan merupakan <i>primary key</i>
identity_number	varchar(20)	Menyimpan nomor identitas pengguna dan merupakan <i>unique</i>
name	varchar(100)	Menyimpan nama user
email	varchar(100)	Menyimpan email pengguna dan merupakan <i>unique</i>
password	varchar(100)	Menyimpan password pengguna
remember_token	varchar(100)	Menyimpan sebuah <i>token</i> pada fungsi “remember me” yang dikelola oleh aplikasi
created_at	timestamp	Menyimpan tanggal saat data pengguna dibuat
updated_at	timestamp	Menyimpan tanggal saat data pengguna diubah

### 2. Tabel uploads

Tabel uploads digunakan untuk menyimpan data upload berupa file pas foto, ktp dan ijazah terakhir. Struktur tabel uploads ditunjukkan pada Tabel 4.6.

**Tabel 4.6 Struktur tabel uploads**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id data yang diunggah dan merupakan <i>primary key</i>
registrant_id	int	Menyimpan id pendaftar dan merupakan <i>foreign key</i>
photo	varchar(255)	Menyimpan pasfoto
ktp	varchar(255)	Menyimpan ktp
last_certificate	varchar(255)	Menyimpan ijazah terakhir
created_at	timestamp	Menyimpan tanggal saat data dibuat
updated_at	timestamp	Menyimpan tanggal saat data diubah



Gambar 4.12 Implementasi basis data

### 3. Tabel educations

Tabel educations digunakan untuk menyimpan data tingkat pendidikan dan jurusan pendidikan. Struktur tabel educations ditunjukkan pada Tabel 4.7.

Tabel 4.7 Struktur tabel educations

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id pendidikan dan merupakan <i>primary key</i>
stage	varchar(50)	Menyimpan tingkat pendidikan
major	varchar(255)	Menyimpan jurusan pendidikan
created_at	timestamp	Menyimpan tanggal saat data pendidikan dibuat
updated_at	timestamp	Menyimpan tanggal saat data pendidikan diubah

#### 4. Tabel educational\_background

Tabel educational\_background digunakan untuk menyimpan data riwayat pendidikan pendaftar. Struktur tabel educational\_background ditunjukkan pada Tabel 4.8.

**Tabel 4.8 Struktur tabel educational\_background**

Nama Kolom	Tipe Data	Keterangan
registrant_id	int	Menyimpan id pendaftar dan merupakan <i>foreign key</i>
educational_id	int	Menyimpan id pendidikan dan merupakan <i>foreign key</i>
name_institution	varchar(100)	Menyimpan nama institusi pendidikan
graduation_year	year	Menyimpan tahun lulus

#### 5. Tabel courses

Tabel courses digunakan untuk menyimpan data jurusan kursus. Struktur tabel courses ditunjukkan pada Tabel 4.9.

**Tabel 4.9 Struktur tabel courses**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id kursus dan merupakan <i>primary key</i>
major	varchar(255)	Menyimpan jurusan kursus
created_at	timestamp	Menyimpan tanggal saat data kursus dibuat
updated_at	timestamp	Menyimpan tanggal saat data kursus diubah

#### 6. Tabel course\_experience

Tabel course\_experience digunakan untuk menyimpan data pengalaman kursus pendaftar. Struktur tabel course ditunjukkan pada Tabel 4.10.

**Tabel 4.10 Struktur tabel course\_experience**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
registrant_id	int	Menyimpan id pendaftar dan merupakan <i>foreign key</i>
course_id	int	Menyimpan id kursus dan merupakan <i>foreign key</i>
organizer	varchar(100)	Menyimpan nama penyelenggara kursus
graduation_year	year	Menyimpan tahun lulus

### 7. Tabel registrants

Tabel registrants digunakan untuk menyimpan detail data pendaftar. Struktur tabel registrants ditunjukkan pada Tabel 4.11.

**Tabel 4.11 Struktur tabel registrants**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id pendaftar dan merupakan <i>primary key</i>
user_id	int	Menyimpan id pengguna dan merupakan <i>foreign key</i>
address	varchar(255)	Menyimpan alamat pendaftar
phone_number	varchar(15)	Menyimpan nomor telepon/hp pendaftar
gender	varchar(15)	Menyimpan jenis kelamin pendaftar
place_birth	varchar(50)	Menyimpan tempat lahir pendaftar
date_birth	date	Menyimpan tanggal lahir pendaftar
order_child	tinyint(2)	Menyimpan urutan anak ke- pendaftar
amount_sibling	tinyint(2)	Menyimpan jumlah saudara pendaftar
religion	varchar(15)	Menyimpan agama pendaftar
biological_mother_name	varchar(100)	Menyimpan nama ibu biologis pendaftar
father_name	varchar(100)	Menyimpan nama ayah pendaftar
parent_address	varchar(255)	Menyimpan alamat orangtua pendaftar
created_at	timestamp	Menyimpan tanggal saat data pendaftar dibuat
updated_at	timestamp	Menyimpan tanggal saat data pendaftar diubah

### 8. Tabel roles

Tabel roles digunakan untuk menyimpan data *role*. Struktur tabel roles ditunjukkan pada Tabel 4.12.

**Tabel 4.12 Struktur tabel roles**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id <i>role</i> dan merupakan <i>primary key</i>
name	varchar(100)	Menyimpan nama <i>role</i>
display_name	varchar(100)	Menyimpan nama <i>role</i> yang ditampilkan
description	varchar(255)	Menyimpan deskripsi <i>role</i>
created_at	timestamp	Menyimpan tanggal saat data <i>role</i> dibuat
updated_at	timestamp	Menyimpan tanggal saat data <i>role</i> diubah

### 9. Tabel role\_user

Tabel role\_user digunakan untuk menyimpan data *role* untuk setiap pengguna. Struktur tabel role\_user ditunjukkan pada Tabel 4.13.

**Tabel 4.13 Struktur tabel role\_user**

Nama Kolom	Tipe Data	Keterangan
role_id	int	Menyimpan id <i>role</i> dan merupakan <i>foreign key</i>
user_id	int	Menyimpan id pengguna dan merupakan <i>foreign key</i>

### 10. Tabel permissions

Tabel permissions digunakan untuk menyimpan data permissions. Struktur permissions ditunjukkan pada Tabel 4.14.

**Tabel 4.14 Struktur tabel permissions**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id <i>permission</i> dan merupakan <i>primary key</i>

**Tabel 4.14 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
name	varchar(100)	Menyimpan nama <i>permission</i>
display_name	varchar(100)	Menyimpan nama <i>permission</i> yang ditampilkan
description	varchar(255)	Menyimpan deskripsi <i>permission</i>
created_at	timestamp	Menyimpan tanggal saat data <i>permission</i> dibuat
updated_at	timestamp	Menyimpan tanggal saat data <i>permission</i> diubah

### 11. Tabel permission\_role

Tabel permission\_role digunakan untuk menyimpan data *permission* untuk setiap *role*. Struktur tabel permission\_role ditunjukkan pada Tabel 4.15.

**Tabel 4.15 Struktur tabel permission\_role**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
role_id	int	Menyimpan id <i>role</i> dan merupakan <i>foreign key</i>
permission_id	int	Menyimpan id <i>permission</i> dan merupakan <i>foreign key</i>

### 12. Tabel vocational

Tabel vocational digunakan untuk menyimpan data kejuruan. Struktur tabel vocational ditunjukkan pada Tabel 4.16.

**Tabel 4.16 Struktur tabel vocational**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id kejuruan dan merupakan <i>primary key</i>
name	varchar(100)	Menyimpan nama kejuruan
description	varchar(255)	Menyimpan deskripsi kejuruan
created_at	timestamp	Menyimpan tanggal saat data kejuruan dibuat
updated_at	timestamp	Menyimpan tanggal saat data kejuruan diubah

### 13. Tabel sub\_vocationals

Tabel sub\_vocationals digunakan untuk menyimpan data subkejuruan. Struktur tabel sub\_vocationals ditunjukkan pada Tabel 4.17.

**Tabel 4.17 Struktur tabel sub\_vocationals**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id subkejuruan dan merupakan <i>primary key</i>
vocational_id	int	Menyimpan id kejuruan dan merupakan <i>foreign key</i>
name	varchar(100)	Menyimpan nama subkejuruan
quota	int	Menyimpan kuota peserta pelatihan dalam subkejuruan
long_training	varchar(10)	Menyimpan lama waktu pelatihan
goal	varchar(1000)	Menyimpan tujuan pelatihan
unit_competence	varchar(1000)	Menyimpan unit kompetensi pelatihan
requirement_participant	varchar(1000)	Menyimpan persyaratan peserta pelatihan
final_registration_date	timestamp	Menyimpan tanggal akhir pendaftaran pelatihan
created_at	timestamp	Menyimpan tanggal saat data subkejuruan dibuat
updated_at	timestamp	Menyimpan tanggal saat data subkejuruan diubah

### 14. Tabel registrations

Tabel registrations digunakan untuk menyimpan data pendaftaran. Struktur tabel registrations ditunjukkan pada Tabel 4.18.

**Tabel 4.18 Struktur tabel registrations**

Nama Kolom	Tipe Data	Keterangan
id	int	Menyimpan id pendaftaran dan merupakan <i>primary key</i>
registrant_id	int	Menyimpan id pendaftar dan merupakan <i>foreign key</i>

**Tabel 4.18 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
sub_vocational_id	int	Menyimpan id subkejuruan dan merupakan <i>foreign key</i>
register_date	timestamp	Menyimpan tanggal pendaftaran
created_at	timestamp	Menyimpan tanggal saat data pendaftaran dibuat
updated_at	timestamp	Menyimpan tanggal saat data pendaftaran diubah

#### 15. Tabel selection\_schedules

Tabel selection\_schedules digunakan untuk menyimpan data jadwal seleksi. Struktur tabel selection\_schedules ditunjukkan pada Tabel 4.19.

**Tabel 4.19 Struktur tabel selection\_schedules**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id jadwal seleksi dan merupakan <i>primary key</i>
sub_vocational_id	int	Menyimpan id subkejuruan dan merupakan <i>foreign key</i>
date	date	Menyimpan tanggal seleksi
time	varchar(20)	Menyimpan waktu seleksi
place	varchar(100)	Menyimpan tempat seleksi
information	varchar(1000)	Menyimpan informasi seleksi
created_at	timestamp	Menyimpan tanggal saat data jadwal seleksi dibuat
updated_at	timestamp	Menyimpan tanggal saat data jadwal seleksi diubah

#### 16. Tabel selections

Tabel selections digunakan untuk menyimpan data seleksi berupa penilaian dan hasil peringkat peserta seleksi . Struktur tabel selections ditunjukkan pada Tabel 4.20.

**Tabel 4.20 Struktur tabel selections**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id seleksi dan merupakan <i>primary key</i>
registration_id	int	Menyimpan id pendaftaran dan merupakan <i>foreign key</i>
selection_schedule_id	int	Menyimpan id jadwal seleksi dan merupakan <i>foreign key</i>
knowledge_value	int	Menyimpan nilai pengetahuan
technical_value	int	Menyimpan nilai keterampilan teknis
recommendation	varchar(5)	Menyimpan apakah merupakan rekomendasi atau tidak
impression_value	varchar(20)	Menyimpan penilaian kesan baik
seriousness_value	varchar(20)	Menyimpan penilaian kesungguhan
confidence_value	varchar(20)	Menyimpan penilaian percaya diri
communication_value	varchar(20)	Menyimpan penilaian keterampilan komunikasi
appearance_value	varchar(20)	Menyimpan penilaian penampilan
family_value	varchar(20)	Menyimpan penilaian pertimbangan keluarga
motivation_value	varchar(255)	Menyimpan penilaian motivasi
attitude_value	varchar(20)	Menyimpan penilaian sikap
orientation_value	varchar(255)	Menyimpan penilaian rencana setelah selesai pelatihan
commitment_value	varchar(20)	Menyimpan penilaian komitmen
honesty_value	varchar(20)	Menyimpan penilaian kejujuran
mental_value	varchar(20)	Menyimpan penilaian mental
economic_value	varchar(20)	Menyimpan penilaian pertimbangan ekonomi
potential_value	varchar(20)	Menyimpan penilaian potensi
note	varchar(255)	Menyimpan catatan jika ada hal-hal selain dalam penilaian yang perlu ditambahkan saat wawancara.
ranking	int	Menyimpan peringkat berdasarkan hasil penilaian
status	varchar(15)	Menyimpan status penilaian. Berikut ini penjelasan isian status: Kosong: sedang proses penilaian Selesai: penilaian selesai dilakukan

**Tabel 4.20 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
		Diterima: hasil seleksi lampau diterima Ditolak: hasil seleksi lampau tidak diterima
created_at	timestamp	Menyimpan tanggal saat data seleksi dibuat
updated_at	timestamp	Menyimpan tanggal saat data seleksi diubah

### 17. Tabel criterias

Tabel criterias digunakan untuk menyimpan data kriteria. Struktur tabel criterias ditunjukkan pada Tabel 4.21.

**Tabel 4.21 Struktur tabel criterias**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id kriteria dan merupakan <i>primary key</i>
group_criteria	int	Menyimpan id kelompok kriteria dan merupakan <i>foreign key</i> dengan <i>default null</i>
name	varchar(500)	Menyimpan nama kriteria
description	varchar(1000)	Menyimpan penjelasan kriteria
citation	varchar(1000)	Menyimpan sitasi kriteria dari kajian pustaka
information	varchar(1000)	Menyimpan parameter kriteria
partial_weight	decimal(20,3)	Menyimpan bobot parsial
global_weight	decimal(20,3)	Menyimpan bobot global
preference	varchar(10)	Menyimpan tipe preferensi
max_min	varchar(10)	Menyimpan kaidah max/min
parameter_p	decimal(20,3)	Menyimpan parameter p
parameter_q	decimal(20,3)	Menyimpan parameter q
parameter_s	decimal(20,3)	Menyimpan parameter s
step	varchar(1)	Menyimpan tahap dalam perumusan kriteria, supaya tahap sebelumnya tetap dapat dilihat. Berikut ini penjelasan isian <i>step</i> : 1: tahap 1 2: tahap 2
status	varchar(1)	Menyimpan status kriteria. Berikut ini penjelasan isian status: 0: deleted

**Tabel 4.21 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
		1: active
ref_id	int	Menyimpan id referensi kriteria, sehingga kriteria yang sudah diubah dalam suatu tahap, dapat diketahui sumber aslinya
created_at	timestamp	Menyimpan tanggal saat data kriteria dibuat
updated_at	timestamp	Menyimpan tanggal saat data kriteria diubah

#### 18. Tabel choice

Tabel choice digunakan untuk menyimpan data pilihan kesesuaian dan usulan kriteria oleh pengguna. Struktur tabel choice ditunjukkan pada Tabel 4.22.

**Tabel 4.22 Struktur tabel choice**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
user_id	int	Menyimpan id pengguna dan merupakan <i>foreign key</i>
criteria_id	int	Menyimpan id kriteria dan merupakan <i>foreign key</i>
option	varchar(1)	Menyimpan apakah pilihan pengguna terhadap suatu kriteria. Berikut ini penjelasan isian <i>option</i> : 0: tidak sesuai 1: sesuai
suggestion	varchar(1)	Menyimpan apakah kriteria merupakan kriteria baku yang bersumber dari kajian pustaka atau usulan dari pengguna. Berikut ini penjelasan isian <i>suggestion</i> : 0: baku 1: usulan

#### 19. Tabel result\_selection

Tabel result\_selection digunakan untuk menyimpan data hasil seleksi sesuai dengan kriteria yang sudah ditentukan dan ketetapan parameter setiap kriteria. Struktur tabel result\_selection ditunjukkan pada Tabel 4.23.

**Tabel 4.23 Struktur tabel result\_selection**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
selection_id	int	Menyimpan id seleksi dan merupakan <i>foreign key</i>
criteria_id	int	Menyimpan id kriteria dan merupakan <i>foreign key</i>
value	int	Menyimpan nilai alternatif terhadap suatu kriteria

#### 20. Tabel pairwise\_comparisons

Tabel pairwise\_comparisons digunakan untuk menyimpan data hasil perbandingan berpasangan kriteria/subkriteria. Struktur tabel pairwise\_comparisons ditunjukkan pada Tabel 4.24.

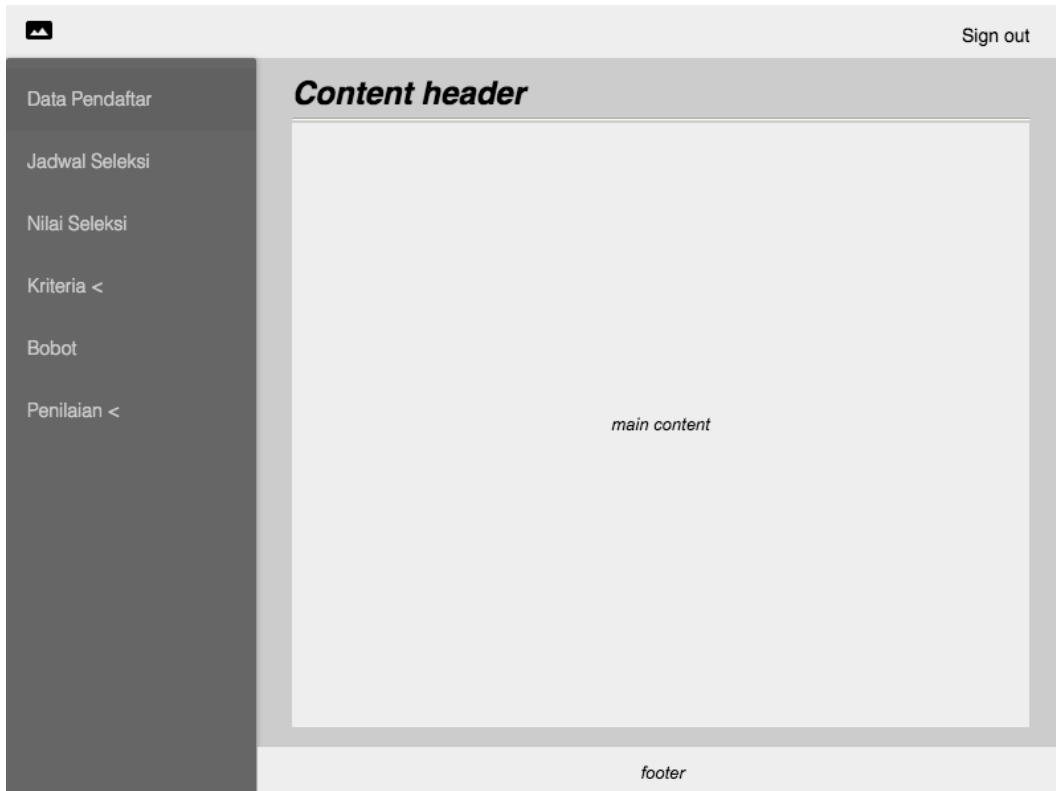
**Tabel 4.24 Struktur tabel pairwise\_comparisons**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
criteria1_id	int	Menyimpan id kriteria 1 dan merupakan <i>foreign key</i>
criteria2_id	int	Menyimpan id kriteria 2 dan merupakan <i>foreign key</i>
value	decimal(20,3)	Menyimpan nilai perbandingan berpasangan kriteria
created_at	timestamp	Menyimpan tanggal saat data perbandingan berpasangan dibuat
updated_at	timestamp	Menyimpan tanggal saat data perbandingan berpasangan diubah

#### 4.6 Perancangan Antarmuka

Tujuan perancangan antarmuka yaitu untuk merancang antarmuka sistem yang sesuai dengan kebutuhan pengguna, sehingga dapat mempermudah interaksi antara pengguna dan sistem. Dalam sistem pendukung keputusan untuk seleksi peserta pelatihan terdapat enam aktor yang memiliki *permission* yang berbeda. Dengan demikian, keenam aktor juga mempunyai halaman utama dengan menu

yang berbeda-beda yaitu disesuaikan dengan kebutuhan masing-masing aktor. Halaman utama merupakan halaman yang berisi fungsionalitas sistem dan ditampilkan setelah pengguna berhasil melakukan login. Rancangan antarmuka halaman utama sistem untuk aktor kepala ditunjukkan pada Gambar 4.13.



**Gambar 4.13 Rancangan antarmuka halaman utama**

Rancangan antarmuka halaman utama untuk keseluruhan aktor terdiri dari empat bagian utama, yaitu *header*, *menu bar*, *content* dan *footer*. Rancangan antarmuka halaman utama untuk aktor yang lain hampir sama, hanya saja bagian menu yang ditampilkan yang berbeda. Berikut ini penjelasan menu untuk masing-masing aktor:

1. Menu untuk aktor kepala yaitu Data Pendaftar, Jadwal Seleksi, Nilai Seleksi, Kriteria, Bobot dan Penilaian. Menu Kriteria mempunyai submenu Kuesioner Kriteria, Hasil Kuesioner Tahap 1, Kriteria Tahap 2 dan Hierarki Kriteria. Menu Penilaian mempunyai submenu Tipe Preferensi, Data Alternatif dan Hasil.

2. Menu untuk aktor staf yaitu Data Pengguna, Data Pendaftar, Data Role, Program, Pendidikan, Kursus, Jadwal Seleksi, Kriteria dan Penilaian. Menu Program mempunyai submenu Kejuruan dan Sub-Kejuruan. Menu Kriteria berisikan fungsionalitas untuk pengelolaan kriteria dari kajian pustaka. Menu Penilaian berisikan hasil dari penilaian yaitu berupa peringkat peserta seleksi pelatihan.
3. Menu untuk aktor kepala sub-bagian tata usaha yaitu Data Pendaftar, Jadwal Seleksi, Kriteria dan Penilaian. Menu Kriteria mempunyai submenu Kuesioner Kriteria, Hasil Kuesioner Tahap 1, Kriteria Tahap 2 dan Hierarki Kriteria. Menu Penilaian berisikan hasil dari penilaian yaitu berupa peringkat peserta seleksi pelatihan.
4. Menu untuk aktor koordinator instruktur dan kepala kejuruan hampir sama yaitu Data Pendaftar, Jadwal Seleksi, Nilai Seleksi, Kriteria dan Penilaian. Hanya saja untuk menu Nilai Seleksi, kepala kejuruan dapat memasukkan nilai seleksi pendaftar sedangkan koordinator instruktur hanya dapat melihat. Menu Kriteria mempunyai submenu Kuesioner Kriteria, Hasil Kuesioner Tahap 1, Kriteria Tahap 2 dan Hierarki Kriteria. Menu Penilaian berisikan hasil dari penilaian yaitu berupa peringkat peserta seleksi pelatihan.
5. Menu untuk aktor pendaftar yaitu Profil dan Daftar. Menu Profil mempunyai submenu Data Diri, Riwayat Pendidikan dan Pengalaman Kursus.

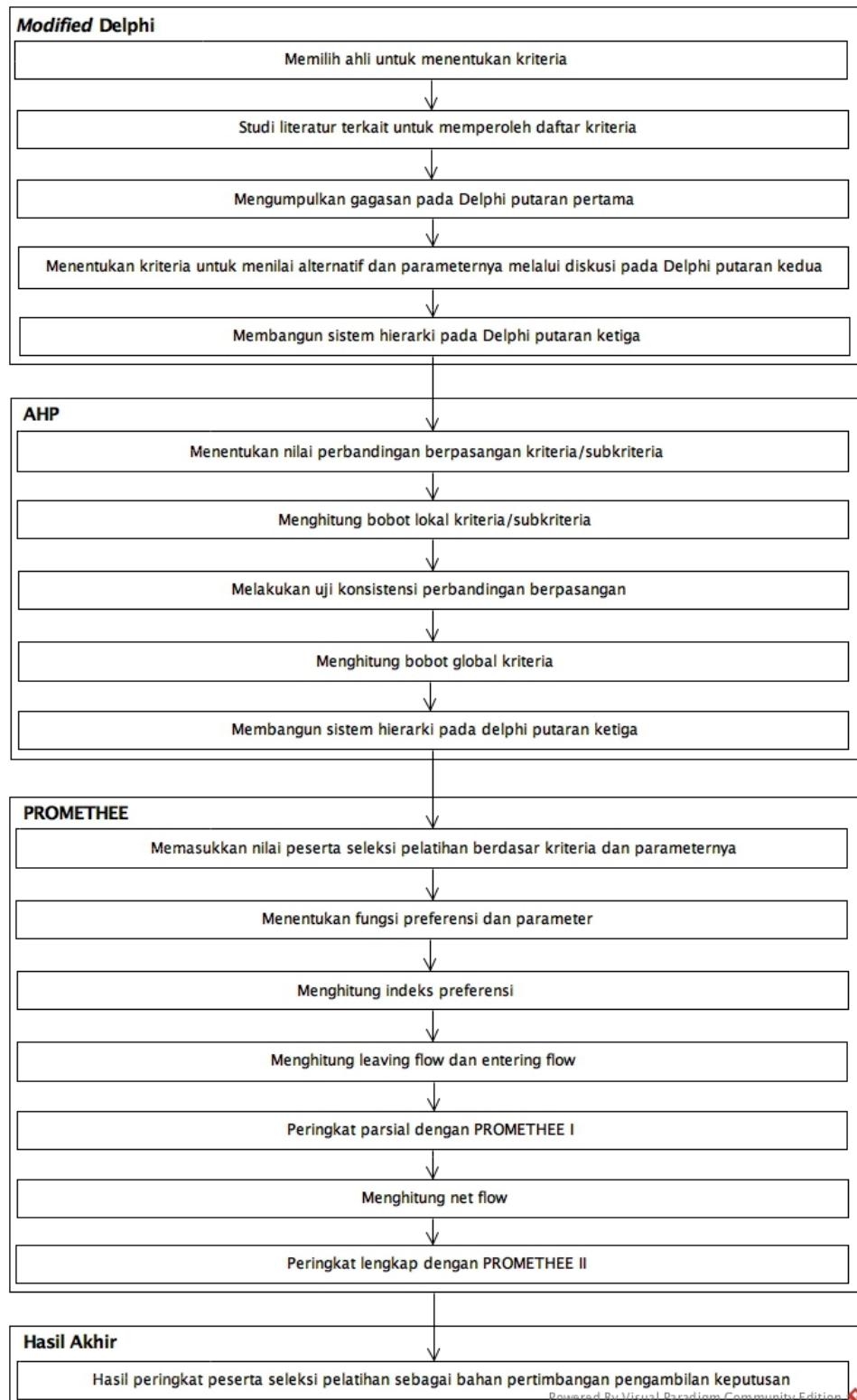
#### **4.7 Perancangan Pemodelan**

Perancangan pemodelan bertujuan untuk merancang model sistem pendukung keputusan seleksi peserta pelatihan. Perancangan pemodelan disesuaikan dengan kebutuhan di BLK Bantul. Rancangan pemodelan sistem ini merupakan gabungan model *Modified Delphi-AHP- PROMETHEE*.

#### 4.7.1 Proses model keputusan seleksi peserta pelatihan

Model *Modified-Delphi* digunakan untuk penentuan kriteria yang akan digunakan dalam penilaian seleksi peserta pelatihan. Model AHP digunakan untuk memperoleh bobot kriteria. Dan model PROMETHEE digunakan untuk memperoleh peringkat alternatif yang digunakan sebagai pertimbangan pembuat keputusan untuk menentukan peserta seleksi yang diterima. Model keputusan secara keseluruhan ditunjukkan pada Gambar 4.14. Rancangan pemodelan sistem ini yaitu sebagai berikut:

1. Metode *Modified Delphi*, terdiri dari proses-proses berikut ini:
  - a) Memilih ahli yang memahami tentang seleksi peserta pelatihan BLK untuk menentukan kriteria, yaitu terdiri dari plt. kepala BLK, koordinator instruktur, kepala sub-bagian tata usaha, kepala kejuruan masing-masing kejuruan (otomotif, furniture, listrik, teknik mekanik, tik, pengolahan hasil pertanian dan aneka kejuruan). Koordinator instruktur dan kepala kejuruan furniture dijabat oleh orang yang sama, sehingga total ahli yang tergabung dalam proses *Modified Delphi* adalah 9 orang.
  - b) Peneliti melakukan studi literatur yang berkaitan dengan kriteria untuk masalah seleksi personil dari jurnal ilmiah dan buku referensi. Daftar kriteria dan penjelasan kompetensi digunakan untuk Delphi putaran pertama. Daftar kriteria berdasarkan studi literatur yang dilakukan oleh peneliti dapat dilihat pada Tabel 4.3.
  - c) Mengumpulkan gagasan pada Delphi putaran pertama. Pada Delphi putaran pertama, responden/ahli diminta untuk mencentang setiap kriteria yang relevan dari daftar kriteria yang telah disiapkan peneliti. Kemudian responden diminta untuk mengirimkan sebanyak-banyaknya kriteria tambahan beserta penjelasannya. Setelah Delphi putaran pertama, tanggapan dikelompokkan untuk dianalisis pada Delphi putaran kedua.



Gambar 4.14 Proses model keputusan seleksi peserta pelatihan

- d) Menentukan kriteria untuk menilai alternatif dan parameternya melalui diskusi pada Delphi putaran kedua. Masing-masing ahli diperbolehkan untuk meninjau pendapat ahli yang lain dan berkomentar ataupun memodifikasi tanggapannya hingga tercapai konsensus. Setiap kriteria yang memiliki banyak pendukung akan diterapkan. Kriteria yang kurang dari tujuh puluh persen dari kesepakatan akan dihilangkan atau ditawarkan untuk penggabungan, menambahkan kriteria baru, pemisahan dan mengedit dari beberapa kriteria yang diterapkan.
- e) Hasil konsensus Delphi putaran kedua dikategorikan dan dikembangkan ke dalam sistem hierarki kriteria penilaian melalui konsensus para ahli dalam Delphi putaran ketiga. Sistem hierarki terdiri dari tingkat atas berupa tujuan pemilihan dalam suatu masalah, tingkat berikutnya berupa kriteria dan akhirnya berupa subkriteria. Verifikasi hierarki kriteria dilakukan oleh pembuat keputusan.
2. Perhitungan AHP menghasilkan bobot kriteria terdiri dari proses-proses berikut ini:
- a) Pembuat keputusan menentukan nilai perbandingan berpasangan kriteria/subkriteria. Skala nilai yaitu 1-9 dengan ketentuan seperti pada Tabel 3.1. Perbandingan berpasangan kriteria/subkriteria akan membentuk matriks seperti pada Persamaan (4.1) .
- $$A = [a_{ij}] = \begin{bmatrix} 1 & a_{12} & \dots & a_{1n} \\ 1/a_{12} & 1 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ 1/a_{1n} & 1/a_{2n} & \dots & 1 \end{bmatrix} \quad (4.1)$$
- dimana matriks  $A = [a_{ij}]$  menggambarkan nilai preferensi pembuat keputusan di antara pasangan kriteria/subkriteria, yaitu  $a_i$  dibanding  $a_j$  untuk  $i, j = 1, 2, 3, \dots, n$ .
- b) Menghitung bobot lokal kriteria/subkriteria dengan Persamaan (4.3). Namun sebelum menghitung bobot, lakukan normalisasi matriks  $A = [a_{ij}]$  terlebih dahulu yaitu dengan Persamaan (4.2).

$$a'_{ij} = a_{ij} / \sum_{(i=1)}^n a_{ij} \quad (4.2)$$

dimana  $a_{ij}$  merupakan hasil perbandingan berpasangan,  $n$  merupakan jumlah kriteria/subkriteria yang dibandingkan dan untuk semua  $j = 1, 2, \dots, n$ .

$$w_i = \sum_{(i=1)}^n a'_{ij} / n \quad (4.3)$$

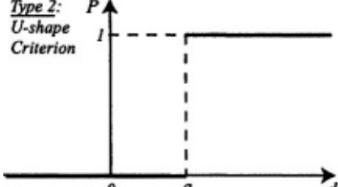
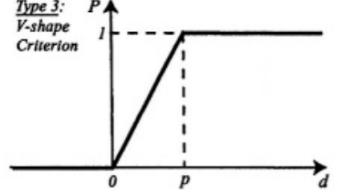
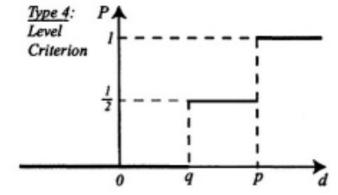
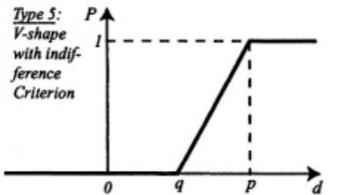
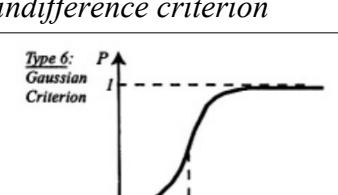
dimana  $a'_{ij}$  merupakan hasil perbandingan berpasangan yang sudah dinormalisasi,  $n$  merupakan jumlah kriteria/subkriteria yang dibandingkan dan untuk semua  $j = 1, 2, \dots, n$ .

- c) Melakukan uji konsistensi perbandingan berpasangan. Pertama, menghitung CI dengan Persamaan (3.1). Kedua, menghitung CR dengan Persamaan (3.2). Nilai CR harus kurang dari 0,1 sehingga dapat dinyatakan konsisten.
  - d) Menghitung bobot global.
3. Perhitungan PROMETHEE untuk memperoleh peringkat peserta pelatihan yang terdiri dari proses-proses berikut ini:
- a) Memasukkan nilai peserta seleksi pelatihan berdasar kriteria dan parameter yang sudah ditentukan. Penilaian ini digunakan sebagai objek yang akan diseleksi.
  - b) Menentukan fungsi preferensi  $P(a,b)$  dan parameter untuk kriteria. Terdapat enam tipe preferensi seperti pada Tabel 4.25.

**Tabel 4.25 Fungsi Preferensi (Brans dan Mareschal, 2005)**

Tipe preferensi	Definisi	Parameter
<p>Type I: Usual Criterion</p> <p>Tipe 1: Usual criterion</p>	$P(d) = \begin{cases} 0 & d \leq 0 \\ 1 & d > 0 \end{cases}$	-

Tabel 4.25 (lanjutan)

Tipe preferensi	Definisi	Parameter
 Tipe 2: <i>U-shape criterion</i>	$P(d) = \begin{cases} 0 & d \leq q \\ 1 & d > q \end{cases}$	q
 Tipe 3: <i>V-shape criterion</i>	$P(d) = \begin{cases} 0 & d \leq 0 \\ d/p & 0 \leq d \leq p \\ 1 & d > p \end{cases}$	p
 Tipe 4: <i>Level criterion</i>	$P(d) = \begin{cases} 0 & d \leq q \\ 1/2 & q < d \leq p \\ 1 & d > p \end{cases}$	p, q
 Tipe 5: <i>V-shape with indifference criterion</i>	$P(d) = \begin{cases} 0 & d \leq q \\ d - q/p - q & q < d \leq p \\ 1 & d > p \end{cases}$	p, q
 Tipe 6: <i>Gaussian criterion</i>	$P(d) = \begin{cases} 0 & d \leq 0 \\ 1 - e^{-(d^2/2s^2)} & d > 0 \end{cases}$	s

- c) Menghitung indeks preferensi  $\Pi(a,b)$  dengan Persamaan (3.3).
- d) Menghitung *leaving flow* berdasarkan nilai dari indeks preferensi dengan Persamaan (3.4) dan menghitung *entering flow* berdasarkan nilai dari indeks preferensi dengan Persamaan (3.5).

- e) Dari hasil perhitungan *leaving flow* dan *entering flow*, kemudian melakukan penyusunan peringkat pada PROMETHEE I. Peserta seleksi dikatakan mempunyai urutan paling tinggi jika nilai *leaving flow*-nya lebih besar dibandingkan dengan peserta seleksi lainnya dan nilai *entering flow*-nya lebih kecil dibandingkan dengan peserta seleksi lainnya. Hal ini menghasilkan bentuk hubungan antara setiap peserta seleksi. Hubungan antarpeserta seleksi dapat diketahui dengan Persamaan (3.6). Jika terdapat urutan peserta seleksi yang tidak dapat dibandingkan (*incomparable*), maka dilanjutkan dengan perhitungan PROMETHEE II.
- f) Menghitung *net flow* dengan Persamaan (3.7).
- g) Jika pada PROMETHEE I urutan peserta seleksi tidak diperoleh, maka dilanjutkan pada PROMETHEE II. PROMETHEE II memberikan peringkat lengkap peserta seleksi dengan mempertimbangkan nilai *net flow*.

#### **4.7.2 Flowchart Pemodelan**

*Flowchart* pemodelan bertujuan untuk menggambarkan diagram alir masing-masing pemodelan yang digunakan. Berikut ini *flowchart* untuk Modified Delphi, AHP dan PROMETHEE.

1. *Flowchart* model Modified Delphi

Gambar 4.15 menunjukkan diagram alir dalam pemodelan Modified Delphi sehingga menghasilkan kriteria yang siap digunakan untuk pemodelan AHP.

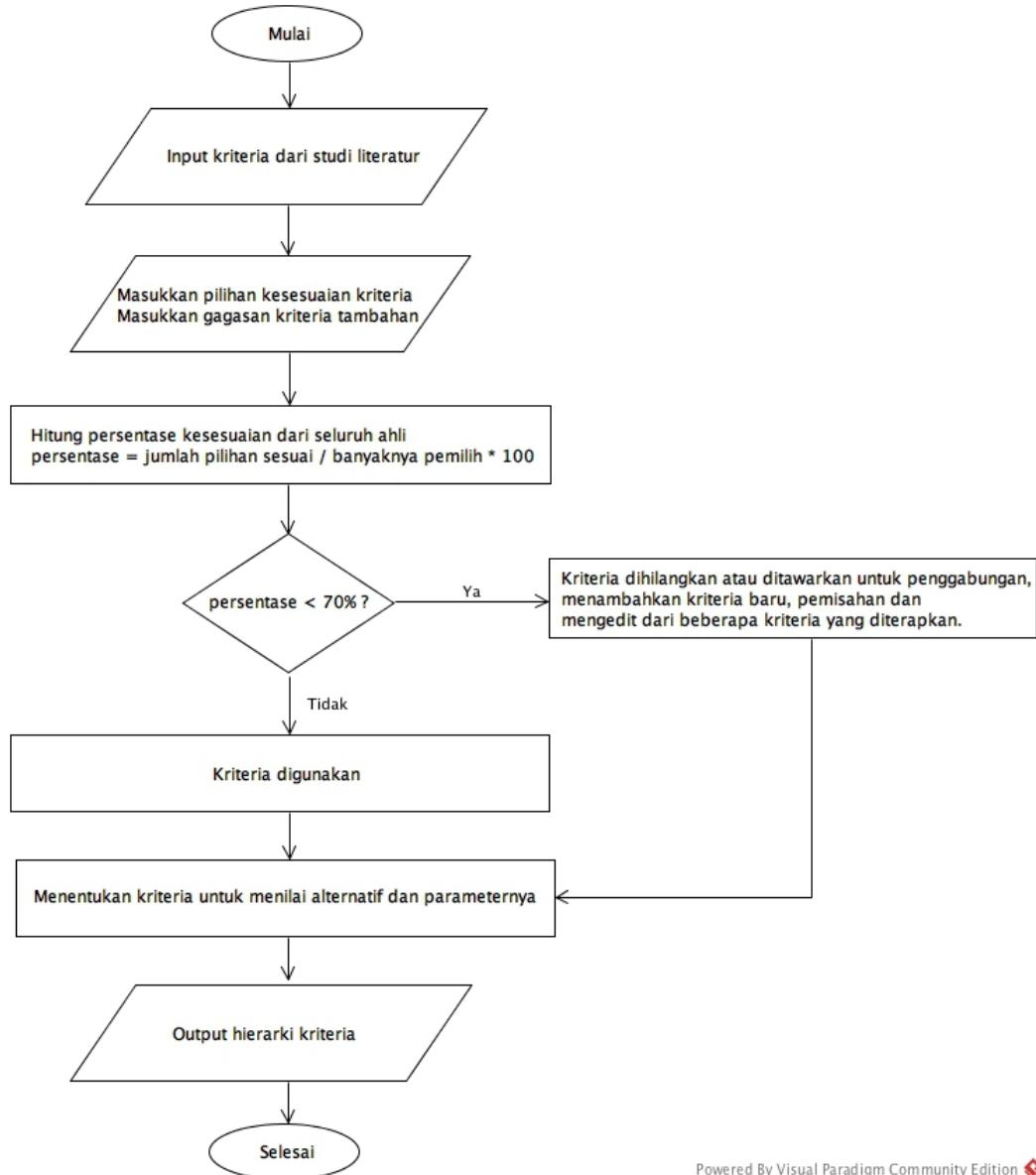
2. *Flowchart* model AHP

Gambar 4.16 menunjukkan diagram alir dalam pemodelan AHP sehingga menghasilkan bobot kriteria/subkriteria yang siap digunakan untuk penilaian peserta seleksi pelatihan dalam pemodelan PROMETHEE.

3. *Flowchart* model PROMETHEE

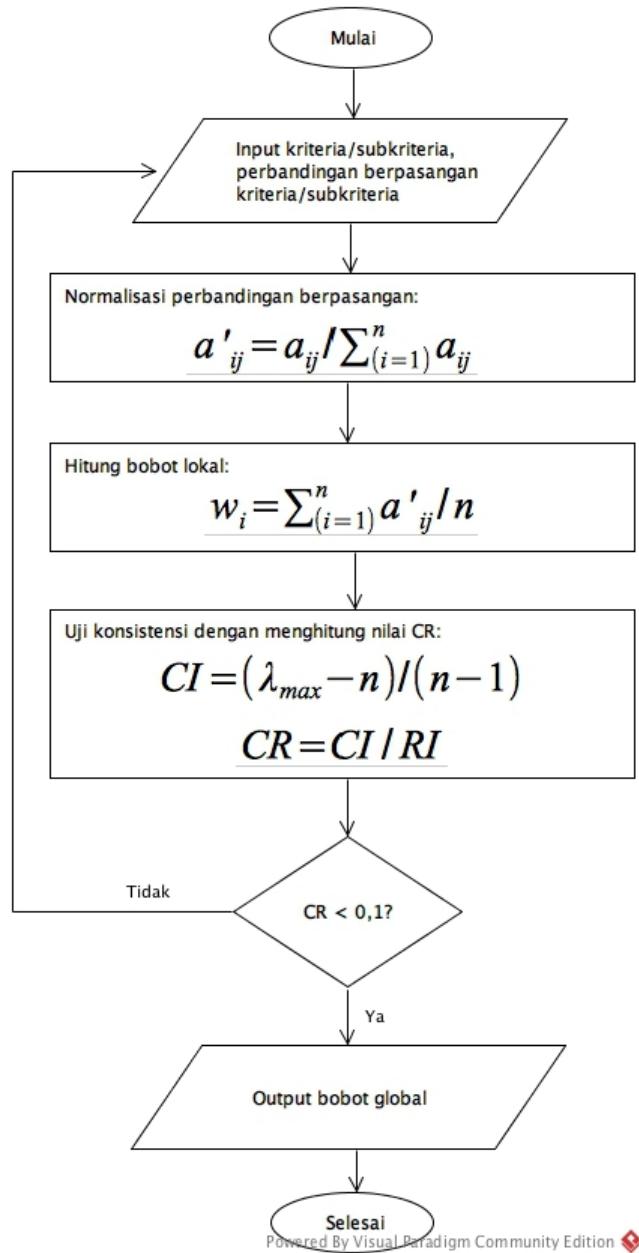
Gambar 4.17 menunjukkan diagram alir dalam pemodelan PROMETHEE sehingga menghasilkan peringkat peserta seleksi pelatihan yang dapat digunakan

pengambil keputusan sebagai bahan pertimbangan memilih peserta seleksi.



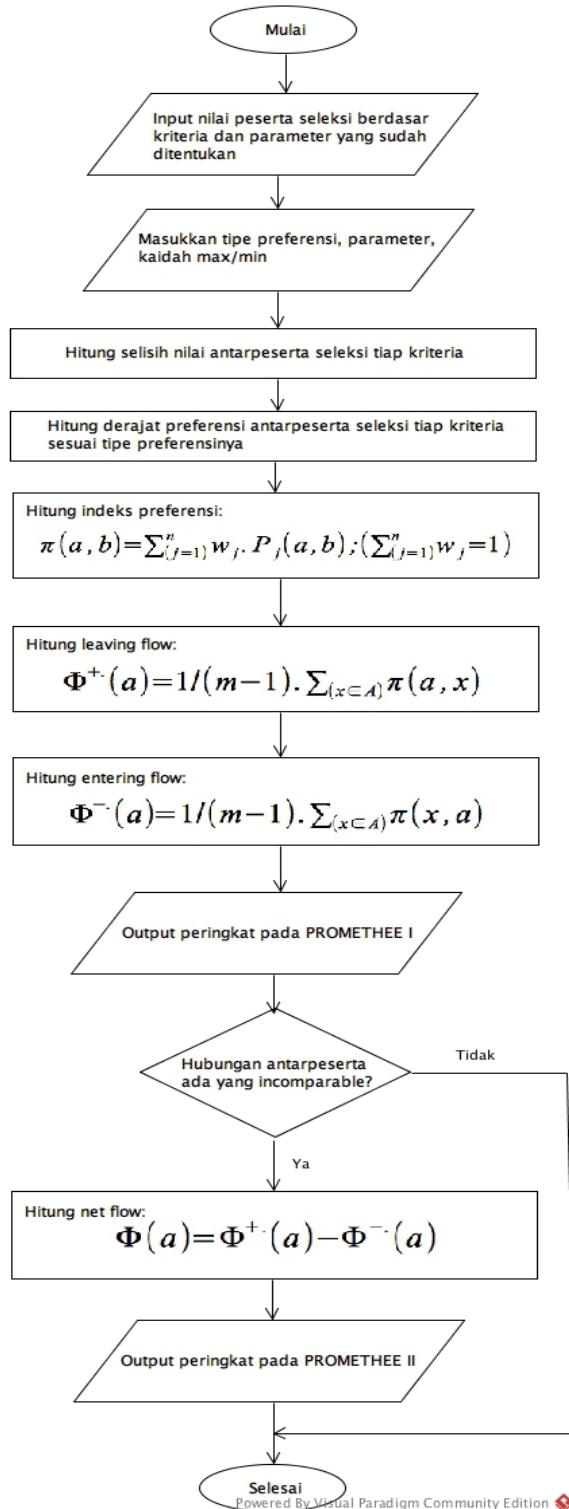
Powered By Visual Paradigm Community Edition

**Gambar 4.15 Flowchart model *Modified Delphi***



Gambar 4.16 Flowchart model AHP

Powered By Visual Paradigm Community Edition ♦

Gambar 4.17 *Flowchart* model PROMETHEE

## 4.8 Rangkaian Penggunaan Pemodelan

Tahap ini merupakan simulasi proses penggunaan model *Modified Delphi-AHP-PROMETHEE*. Saat perancangan sistem ini, data yang digunakan adalah data *dummy*, baik dari kriteria maupun alternatif. Hal ini karena belum ada ketetapan kriteria. Namun dalam penjelasan rangkaian penggunaan pemodelan ini menggunakan data uji. Data uji ini merupakan data sesungguhnya yang diperoleh dari BLK Bantul. Namun ada data yang dialiaskan untuk kepentingan keamanan, seperti nama peserta seleksi pelatihan.

### 4.8.1 Rangkaian penggunaan model *Modified Delphi*

Penggunaan model *Modified Delphi* bertujuan untuk menentukan kriteria yang digunakan dalam menilai peserta seleksi pelatihan. Berikut ini rangkaian penggunaan model *Modified Delphi*:

1. Memasukkan data kriteria dari studi literatur ke dalam sistem. Data kriteria antara lain seperti pada Tabel 4.3.
2. Tahap Delphi putaran pertama, sembilan orang ahli yang sudah ditentukan mengisi kuesioner kriteria. Kuesioner berisi daftar kriteria dari studi literatur. Para ahli juga diijinkan untuk memberi gagasan masukan kriteria yang belum tercakup dalam studi literatur beserta penjelasannya.
3. Dari Delphi putaran pertama, menghasilkan kumpulan gagasan dan kesesuaian kriteria studi literatur dengan kondisi di BLK Bantul. Terdapat satu masukan kriteria yaitu intensitas keikutsertaan pelatihan di BLK Bantul. Hasil kesesuaian kriteria ditunjukkan pada Tabel 4.26.

**Tabel 4.26 Hasil kesesuaian kriteria pada Delphi putaran pertama**

No.	Kriteria	Kesesuaian (%)
1	Kesan baik	100
2	Kesungguhan	100
3	Potensi	77,8
4	Keuangan	77,8

**Tabel 4.26 (lanjutan)**

No.	Kriteria	Kesesuaian (%)
5	Komitmen	100
6	Kemantapan emosional	11,1
7	Percaya diri	88,9
8	Keterampilan komunikasi	88,9
9	Penampilan	100
10	Keterampilan komputer	22,2
11	Kemampuan kognitif	0
12	Karakteristik fisik	0
13	Status perkawinan	22,2
14	Ras dan etnis	22,2
15	Pertimbangan keluarga	77,8
16	Mental	88,9
17	Motivasi	100
18	Kefasihan bahasa asing	11,1
19	Usia	77,8
20	Jenis kelamin	22,2
21	Pengalaman pekerjaan	66,7
22	Pendidikan formal	77,8
23	Sikap	100
24	Bebas obat-obatan terlarang	44,4
25	Kejujuran	88,9
26	Keterampilan teknis	88,9
27	Pengetahuan	100
28	Kepribadian	66,7
29	Kesehatan	44,4
30	Rekomendasi	77,8
31	Orientasi masa depan	100
32	Pengalaman pelatihan	88,9

4. Pada Delphi putaran kedua, ditentukan kriteria dan parameter yang digunakan untuk menilai peserta seleksi pelatihan. Kriteria yang kurang dari 70% dari kesepakatan dihilangkan. Terdapat 21 kriteria yang

dihilangkan yaitu kemantapan emosional, keterampilan komputer, kemampuan kognitif, karakteristik fisik, status perkawinan, ras dan etnis, kefasihan bahasa asing, jenis kelamin, pengalaman pekerjaan, bebas obat-obatan terlarang, kepribadian dan kesehatan. Dalam Delphi putaran kedua ini, para ahli juga diijinkan untuk mengubah nama disesuaikan dengan BLK Bantul, seperti kriteria orientasi masa depan diubah menjadi rencana setelah selesai pelatihan. Kriteria dan parameter yang digunakan untuk menilai peserta seleksi ditunjukkan pada Tabel 4.27.

**Tabel 4.27 Hasil kriteria pada Delphi putaran kedua**

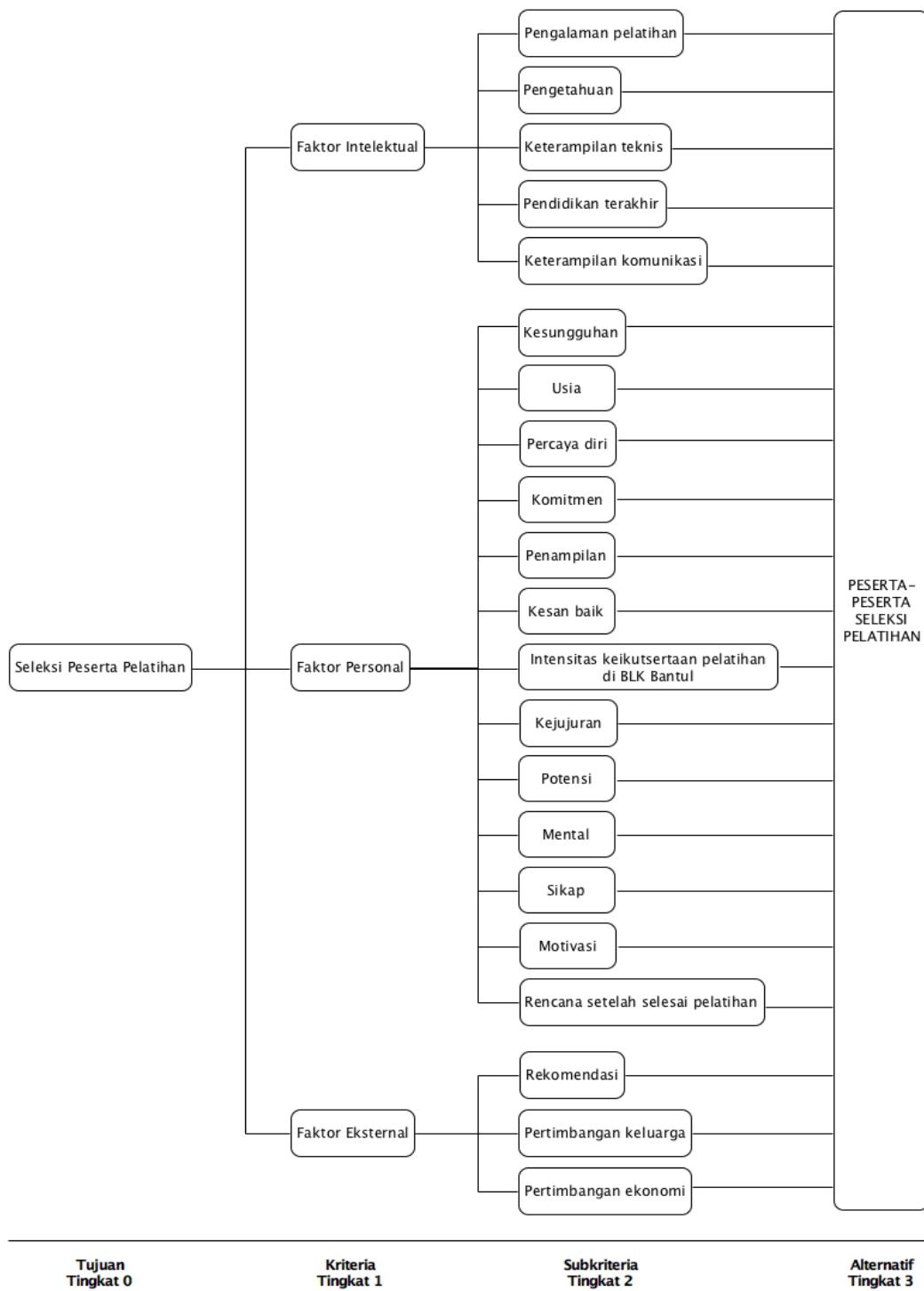
No.	Kriteria	Kode	Parameter Penilaian
1	Intensitas keikutsertaan pelatihan di BLK Bantul	IK	Belum pernah: 4 Pernah ikut 1 kali: 3 Pernah ikut 2 kali: 2 Pernah ikut $\geq 3$ kali: 1
2	Pengalaman pelatihan	PP	Pernah ikut pelatihan di luar BLK dan sesuai bidang yang diminati: 3 Pernah ikut pelatihan di luar BLK dan tidak sesuai bidang yang diminati: 2 Belum pernah: 1
3	Rencana setelah selesai pelatihan	RS	Membuka usaha: 3 Melamar pekerjaan: 2 Menambah ilmu/keterampilan: 1 Tidak ada: 0
4	Rekomendasi	RK	Ada: 2 Tidak: 1
5	Pengetahuan	PG	0-100
6	Keterampilan teknis	KT	0-100
7	Kejujuran	KJ	Sesuai: 1 Tidak Sesuai: 0
8	Sikap	SI	Baik: 3 Cukup: 2 Kurang: 1
9	Pendidikan teakhir	PT	SMA/SMK: 4 SMP: 3 SD ke bawah: 2 Diploma ke atas: 1

**Tabel 4.27 (lanjutan)**

No.	Kriteria	Kode	Parameter Penilaian
10	Usia	US	17 - 25 tahun: 4 26 - 34 tahun: 3 35 - 40 tahun: 2 41 tahun ke atas: 1 di bawah 17 tahun: 0
11	Motivasi	MO	Kemauan sendiri: 2 Dorongan orang lain: 1 Tidak ada: 0
12	Mental	ME	Baik: 3 Cukup: 2 Kurang: 1
13	Pertimbangan keluarga	PK	Dijijinkan: 2 Tidak diijinkan: 1
14	Penampilan	PM	Baik: 3 Cukup: 2 Kurang: 1
15	Keterampilan komunikasi	KK	Baik: 3 Cukup: 2 Kurang: 1
16	Percaya diri	PD	Baik: 3 Cukup: 2 Kurang: 1
17	Komitmen	KO	Sanggup: 2 Ragu-ragu: 1 Tidak sanggup: 0
18	Pertimbangan ekonomi	PE	Kurang: 3 Cukup: 2 Mapan: 1
19	Potensi	PO	Berpotensi: 2 Kurang berpotensi: 1
20	Kesungguhan	KS	Baik: 3 Cukup: 2 Kurang: 1
21	Kesan baik	KB	Baik: 3 Cukup: 2 Kurang: 1

5. Delphi putaran ketiga mengkategorikan kriteria menjadi tiga kelompok yaitu faktor intelektual, faktor personal dan faktor eksternal. Kemudian

dikembangkan ke dalam sistem hierarki kriteria yang ditunjukkan pada Gambar 4.18.



Gambar 4.18 Hierarki kriteria seleksi peserta pelatihan

#### 4.8.2 Rangkaian penggunaan model AHP

Penggunaan model AHP bertujuan untuk memperoleh bobot kriteria yang digunakan dalam menilai peserta seleksi pelatihan. Berikut ini rangkaian penggunaan model AHP:

1. Melakukan perbandingan berpasangan kriteria. Hasil perbandingan berpasangan ditunjukkan Tabel 4.28.

**Tabel 4.28 Matriks perbandingan berpasangan kriteria**

	<b>FI</b>	<b>FP</b>	<b>FE</b>
<b>FI</b>	1.000	0.500	2.000
<b>FP</b>	2.000	1.000	3.000
<b>FE</b>	0.500	0.333	1.000
<b>Jumlah</b>	3.500	1.833	6.000

2. Normalisasi matriks perbandingan berpasangan kriteria yang ditunjukkan pada Tabel 4.29.

**Tabel 4.29 Normalisasi matriks perbandingan berpasangan kriteria**

	<b>FI</b>	<b>FP</b>	<b>FE</b>	<b>Jumlah'</b>
<b>FI</b>	0.286	0.273	0.333	0.892
<b>FP</b>	0.571	0.545	0.500	1.617
<b>FE</b>	0.143	0.182	0.167	0.491

3. Menghitung bobot kriteria yaitu dengan melakukan pembagian jumlah per baris matriks yang sudah dinormalisasi dengan jumlah kriteria yang dibandingkan. Hasil bobot ditunjukkan pada Tabel 4.30.

**Tabel 4.30 Hasil bobot kriteria**

<b>Kriteria</b>	<b>Bobot = Jumlah'/3</b>
FI	0.297
FP	0.539
FE	0.164

4. Uji konsistensi dengan menghitung CR.

$$\lambda_{max} = \sum bobot * jumlah = (0.297 * 3.500) + (0.539 * 1.833) + (0.164 * 6.000) = 3.011$$

$$CI = (\lambda_{max} - n) / (n - 1) = (3.011 - 3) / (3 - 1) = 0.006$$

$$CR = CI / RI = 0.006 / 0.580 = 0.01$$

Nilai CR <= 0.1, sehingga perbandingan berpasangan konsisten.

5. Melakukan perbandingan berpasangan subkriteria untuk kriteria FI. Hasil perbandingan berpasangan ditunjukkan Tabel 4.31.

**Tabel 4.31 Matriks perbandingan berpasangan subkriteria untuk kriteria FI**

	PP	PG	KT	PT	KK
PP	1.000	1.000	0.500	0.250	0.500
PG	1.000	1.000	0.500	0.250	0.500
KT	2.000	2.000	1.000	0.500	1.000
PT	4.000	4.000	2.000	1.000	2.000
KK	2.000	2.000	1.000	0.500	1.000
Jumlah	10.000	10.000	5.000	2.500	5.000

6. Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FI yang ditunjukkan pada Tabel 4.32.

**Tabel 4.32 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FI**

	PP	PG	KT	PT	KK	Jumlah'
PP	0.100	0.100	0.100	0.100	0.100	0.500
PG	0.100	0.100	0.100	0.100	0.100	0.500
KT	0.200	0.200	0.200	0.200	0.200	1.000
PT	0.400	0.400	0.400	0.400	0.400	2.000
KK	0.200	0.200	0.200	0.200	0.200	1.000

7. Menghitung bobot subkriteria untuk kriteria FI. Hasil bobot ditunjukkan pada Tabel 4.33.

**Tabel 4.33 Hasil bobot subkriteria untuk kriteria FI**

<b>Subkriteria</b>	<b>Bobot = Jumlah'/5</b>
PP	0.100
PG	0.100
KT	0.200
PT	0.400
KK	0.200

8. Uji konsistensi dengan menghitung CR.

$$\lambda_{max} = \sum \text{bobot} * \text{jumlah} = 5.000$$

$$CI = (\lambda_{max} - n) / (n - 1) = 0.000$$

Nilai RI untuk n=5 yaitu 1.120

$$CR = CI / RI = 0.000$$

Nilai CR <= 0.1, sehingga perbandingan berpasangan konsisten.

9. Melakukan perbandingan berpasangan subkriteria untuk kriteria FP. Hasil perbandingan berpasangan ditunjukkan Tabel 4.34.

**Tabel 4.34 Matriks perbandingan berpasangan subkriteria untuk kriteria FP**

	<b>IK</b>	<b>RS</b>	<b>KJ</b>	<b>SI</b>	<b>US</b>	<b>MO</b>	<b>ME</b>	<b>PM</b>	<b>PD</b>	<b>KO</b>	<b>PO</b>	<b>KS</b>	<b>KB</b>
<b>IK</b>	1.000	1.000	0.500	1.000	1.000	0.500	0.500	2.000	0.500	0.500	0.500	0.500	2.000
<b>RS</b>	1.000	1.000	0.500	1.000	1.000	0.500	0.500	2.000	1.000	1.000	1.000	1.000	2.000
<b>KJ</b>	2.000	2.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>SI</b>	1.000	1.000	0.500	1.000	1.000	0.500	0.500	2.000	0.500	0.500	0.500	0.500	2.000
<b>US</b>	1.000	1.000	0.500	1.000	1.000	0.500	0.500	2.000	0.500	0.500	0.500	0.500	2.000
<b>MO</b>	2.000	2.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>ME</b>	2.000	2.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>PM</b>	0.500	0.500	0.333	0.500	0.500	0.333	0.333	1.000	0.333	0.333	0.333	0.333	2.000
<b>PD</b>	2.000	1.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>KO</b>	2.000	1.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>PO</b>	2.000	1.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>KS</b>	2.000	1.000	1.000	2.000	2.000	1.000	1.000	3.000	1.000	1.000	1.000	1.000	5.000
<b>KB</b>	0.500	0.500	0.200	0.500	0.500	0.200	0.200	0.500	0.200	0.200	0.200	0.200	1.000
<b>Jml</b>	19.000	15.000	9.533	19.000	19.000	9.533	9.533	30.500	10.033	10.033	10.033	10.033	46.000

10. Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FP yang ditunjukkan pada Tabel 4.35.

**Tabel 4.35 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FP**

	<b>IK</b>	<b>RS</b>	<b>KJ</b>	<b>SI</b>	<b>US</b>	<b>MO</b>	<b>ME</b>	<b>PM</b>	<b>PD</b>	<b>KO</b>	<b>PO</b>	<b>KS</b>	<b>KB</b>	<b>Jml</b>
<b>IK</b>	0.053	0.067	0.052	0.053	0.053	0.052	0.052	0.066	0.050	0.050	0.050	0.050	0.043	0.690
<b>RS</b>	0.053	0.067	0.052	0.053	0.053	0.052	0.052	0.066	0.100	0.100	0.100	0.100	0.043	0.890
<b>KJ</b>	0.105	0.133	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.370
<b>SI</b>	0.053	0.067	0.052	0.053	0.053	0.052	0.052	0.066	0.050	0.050	0.050	0.050	0.043	0.690
<b>US</b>	0.053	0.067	0.052	0.053	0.053	0.052	0.052	0.066	0.050	0.050	0.050	0.050	0.043	0.690
<b>MO</b>	0.105	0.133	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.370
<b>ME</b>	0.105	0.133	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.370
<b>PM</b>	0.026	0.033	0.035	0.026	0.026	0.035	0.035	0.033	0.033	0.033	0.033	0.033	0.043	0.426
<b>PD</b>	0.105	0.067	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.303
<b>KO</b>	0.105	0.067	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.303
<b>PO</b>	0.105	0.067	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.303
<b>KS</b>	0.105	0.067	0.105	0.105	0.105	0.105	0.105	0.098	0.100	0.100	0.100	0.100	0.109	1.303
<b>KB</b>	0.026	0.033	0.021	0.026	0.026	0.021	0.021	0.016	0.200	0.200	0.200	0.200	0.022	0.293

11. Menghitung bobot subkriteria untuk kriteria FP. Hasil bobot ditunjukkan pada Tabel 4.36.

**Tabel 4.36 Hasil bobot subkriteria untuk kriteria FP**

<b>Subkriteria</b>		<b>Bobot = Jumlah'/13</b>
IK		0.053
RS		0.068
KJ		0.105
SI		0.053
US		0.053
MO		0.105
ME		0.105
PM		0.033
PD		0.100

**Tabel 4.36 (lanjutan)**

<b>Subkriteria</b>	<b>Bobot = Jumlah'/13</b>
KO	0.100
PO	0.100
KS	0.100
KB	0.023

12. Uji konsistensi dengan menghitung CR.

$$\lambda_{max} = \sum bobot * jumlah = 13.125$$

$$CI = (\lambda_{max} - n) / (n - 1) = 0.010$$

Nilai RI untuk n=13 yaitu 1.560

$$CR = CI / RI = 0.007$$

Nilai CR <= 0.1, sehingga perbandingan berpasangan konsisten.

13. Melakukan perbandingan berpasangan subkriteria untuk kriteria FE. Hasil perbandingan berpasangan ditunjukkan Tabel 4.37.

**Tabel 4.37 Matriks perbandingan berpasangan subkriteria untuk kriteria FE**

	<b>RK</b>	<b>PK</b>	<b>PE</b>
<b>RK</b>	1.000	3.000	2.000
<b>PK</b>	0.333	1.000	0.500
<b>PE</b>	0.500	2.000	1.000
<b>Jumlah</b>	1.833	6.000	3.500

14. Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FE yang ditunjukkan pada Tabel 4.38.

**Tabel 4.38 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FE**

	<b>RK</b>	<b>PK</b>	<b>PE</b>	<b>Jumlah'</b>
<b>RK</b>	0.546	0.500	0.571	1.617
<b>PK</b>	0.182	0.167	0.143	0.491
<b>PE</b>	0.273	0.333	0.286	0.892

15. Menghitung bobot subkriteria untuk kriteria FE. Hasil bobot ditunjukkan pada Tabel 4.39.

**Tabel 4.39 Hasil bobot subkriteria untuk kriteria FE**

Subkriteria	Bobot = Jumlah'/3
RK	0.539
PK	0.164
PE	0.297

16. Uji konsistensi dengan menghitung CR.

$$\lambda_{max} = \sum \text{bobot} * \text{jumlah} = 3.011$$

$$CI = (\lambda_{max} - n) / (n - 1) = 0.005$$

Nilai RI untuk n=3 yaitu 0.580

$$CR = CI / RI = 0.009$$

Nilai CR <= 0.1, sehingga perbandingan berpasangan konsisten.

17. Menghitung bobot global dengan mengalikan bobot subkriteria dengan bobot kriteria dalam kelompok kriteria yang bersangkutan. Hasil bobot global ditunjukkan pada Tabel 4.40.

**Tabel 4.40 Bobot global kriteria**

Kriteria	Subkriteria	Bobot global
FI	PP	0.030
	PG	0.030
	KT	0.059
	PT	0.119
	KK	0.059
FP	IK	0.029
	RS	0.037
	KJ	0.057
	SI	0.029

**Tabel 4.40 (lanjutan)**

<b>Kriteria</b>	<b>Subkriteria</b>	<b>Bobot global</b>
	US	0.029
	MO	0.057
	ME	0.057
	PM	0.018
	PD	0.054
	KO	0.054
	PO	0.054
	KS	0.054
	KB	0.012
FE	RK	0.088
	PK	0.027
	PE	0.049

Contoh untuk menghitung bobot global PE:

$$\begin{aligned}
 \text{Bobot global} &= \text{bobot PE} * \text{bobot FE} \\
 &= 0.297 * 0164 \\
 &= 0.049
 \end{aligned}$$

#### **4.8.3 Rangkaian penggunaan model PROMETHEE**

Penggunaan model PROMETHEE bertujuan untuk memperoleh peringkat peserta seleksi pelatihan yang dapat digunakan pembuat keputusan sebagai bahan pertimbangan peserta pelatihan yang diterima. Berikut ini rangkaian penggunaan model PROMETHEE:

1. Memasukkan nilai peserta seleksi pelatihan berdasarkan kriteria dan parameter yang sudah ditentukan pada Tabel 4.27. Jumlah peserta seleksi yang digunakan dalam penilaian yaitu 32 orang. Data peserta seleksi dapat dilihat pada Tabel 4.1 dan Tabel 4.2. Nilai peserta seleksi ditunjukkan pada Tabel 4.41.
2. Menentukan fungsi preferensi  $P(a,b)$  dan parameter untuk masing-masing kriteria. Fungsi preferensi ditunjukkan pada Tabel 4.43.

**Tabel 4.41 Nilai peserta seleksi pelatihan**

No.	Pendaftar	KB	KS	PO	PE	KO	PD	KK	PM	PK	ME	MO	US	PT	SI	KJ	KT	PG	RK	RS	PP	IK	
1	P1	2	1	1	3	2	2	2	2	2	2	0	4	2	2	0	85	50	1	1	1	4	
2	P2	3	3	2	3	2	3	3	2	3	2	2	4	3	3	1	55	50	1	3	1	4	
3	P3	3	3	2	3	2	3	3	2	3	2	4	4	3	1	90	70	2	3	3	4		
4	P4	3	3	2	3	2	3	3	2	3	2	4	4	3	1	90	80	1	2	3	4		
5	P5	2	3	2	3	2	2	2	2	2	2	1	4	3	2	1	90	80	2	3	1	4	
6	P6	3	3	2	3	2	3	3	2	3	2	2	4	4	3	1	85	70	1	2	1	4	
7	P7	2	1	2	1	2	3	3	2	3	2	2	4	1	3	1	100	80	1	3	3	4	
8	P8	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	0	1	4
9	P9	3	3	2	3	2	3	3	2	3	2	4	4	3	1	30	80	1	2	1	4		
10	P10	3	3	2	3	2	3	3	2	3	2	4	2	3	1	70	80	2	2	3	4		
11	P11	2	2	2	3	2	2	2	2	2	3	0	4	3	3	1	85	40	2	1	1	4	
12	P12	2	3	2	3	2	3	3	2	2	2	2	4	2	2	0	50	70	2	3	1	4	
13	P13	3	3	2	3	2	2	2	3	2	1	0	4	4	3	1	95	60	1	0	1	4	
14	P14	3	3	2	2	2	3	3	3	2	3	2	4	4	3	1	65	70	1	2	1	4	
15	P15	1	1	1	3	2	2	1	1	2	2	0	4	4	2	0	85	60	1	1	1	4	
16	P16	3	3	2	2	2	3	3	2	3	0	4	4	3	1	80	50	1	0	2	4		
17	P17	3	3	2	2	2	3	3	2	3	2	3	4	3	1	55	90	1	3	1	4		
18	P18	3	3	2	3	2	3	3	2	3	2	4	4	3	1	95	70	2	2	1	4		

**Tabel 4.41 (lanjutan)**

No.	Pendaftar	KB	KS	PO	PE	KO	PD	KK	PM	PK	ME	MO	US	PT	SI	KJ	KT	PG	RK	RS	PP	IK
19	P19	3	3	2	3	2	3	3	3	2	3	2	4	3	3	1	95	70	1	2	1	4
20	P20	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
21	P21	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
22	P22	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
23	P23	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
24	P24	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
25	P25	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
26	P26	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
27	P27	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
28	P28	0	0	0	0	0	0	0	0	0	0	0	4	2	0	0	0	0	0	0	1	4
29	P29	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
30	P30	0	0	0	0	0	0	0	0	0	0	0	2	4	0	0	0	0	0	0	1	4
31	P31	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4
32	P32	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	0	0	0	1	4

**Tabel 4.42 Selisih (1,2) untuk setiap kriteria**

f <sub>j</sub> (..., ...)	fKB(.)	fKS(.)	fPO(.)	fPE(.)	fKO(.)	fPD(.)	fKK(.)	fPM(.)	fPK(.)	fME(.)	fMO(.)	fUS(.)	fPT(.)	fSI(.)	fKJ(.)	fKT(.)	fPG(.)	fRK(.)	fRS(.)	fPP(.)	fIK(.)
	Tipe 4	Tipe 5	Tipe 5	Tipe 4	Tipe 4	Tipe 4	Tipe 4														
(1,2)	-1	-2	-1	0	0	-1	-1	-1	0	-1	-2	0	-1	-1	-1	30	0	0	-2	0	0

**Tabel 4.43 Fungsi preferensi**

No.	Kriteria	Tipe Kriteria	Kaidah Maks/Min	Parameter		
				q	p	s
1	KB	4	Maksimasi	0.5	1	-
2	KS	4	Maksimasi	0.5	1	-
3	PO	4	Maksimasi	0.5	1	-
4	PE	4	Maksimasi	0.5	1	-
5	KO	4	Maksimasi	0.5	1	-
6	PD	4	Maksimasi	0.5	1	-
7	KK	4	Maksimasi	0.5	1	-
8	PM	4	Maksimasi	0.5	1	-
9	PK	4	Maksimasi	0.5	1	-
10	ME	4	Maksimasi	0.5	1	-
11	MO	4	Maksimasi	0.5	1	-
12	US	4	Maksimasi	0.5	1	-
13	PT	4	Maksimasi	0.5	1	-
14	SI	4	Maksimasi	0.5	1	-
15	KJ	4	Maksimasi	0.5	1	-
16	KT	5	Maksimasi	5	10	-
17	PG	5	Maksimasi	5	10	-
18	RK	4	Maksimasi	0.5	1	-
19	RS	4	Maksimasi	0.5	1	-
20	PP	4	Maksimasi	0.5	1	-
21	IK	4	Maksimasi	0.5	1	-

3. Menghitung indeks preferensi  $\Pi(a,b)$ . Index preferensi ditunjukkan pada Tabel 4.44. Dalam Tabel 4.44, inisial P maksudnya adalah Pendaftar.

**Tabel 4.44 Indeks preferensi**

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	
	0	6	0	0	0	0	1	6	6	6	3	6	5	8	4	4	0	0	6	6	6	6	6	6	6	6	6	9	6	6		
2	0.3	0.0	0.0	0.0	0.1	0.0	0.2	0.7	0.0	0.0	0.2	0.1	0.2	0.0	0.3	0.1	0.0	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	
	8	0	0	2	4	2	3	2	8	8	2	5	1	4	6	2	4	2	2	2	2	2	2	2	2	2	8	2	5	2	2	
3	0.5	0.2	0.0	0.0	0.2	0.0	0.2	0.8	0.1	0.2	0.3	0.2	0.3	0.1	0.4	0.2	0.1	0.0	0.1	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.9	0.8	0.8	0.8	0.8	
	4	2	0	6	3	9	7	0	5	0	1	9	1	8	7	7	7	5	5	0	0	0	0	0	0	0	1	0	2	0	0	
4	0.4	0.1	0.0	0.0	0.2	0.0	0.2	0.7	0.0	0.1	0.2	0.3	0.2	0.1	0.4	0.2	0.1	0.0	0.1	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	
	8	8	3	0	3	6	3	5	9	8	9	2	7	4	0	2	3	6	2	5	5	5	5	5	5	5	7	5	8	5	5	
5	0.3	0.1	0.0	0.0	0.0	0.0	0.2	0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.2	0.2	0.1	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	
	1	3	3	6	0	9	7	4	2	4	2	8	7	8	9	2	4	5	9	4	4	4	4	4	4	4	0	4	7	4	4	
6	0.4	0.1	0.0	0.0	0.2	0.0	0.2	0.7	0.0	0.1	0.2	0.2	0.2	0.0	0.3	0.1	0.1	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	
	5	5	0	0	0	0	3	2	6	8	6	6	4	8	7	5	0	0	6	2	2	2	2	2	2	2	4	2	5	2	2	
7	0.3	0.1	0.0	0.0	0.2	0.1	0.0	0.7	0.1	0.0	0.2	0.2	0.2	0.1	0.4	0.2	0.1	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	
	8	8	2	9	8	3	4	0	0	1	8	8	0	7	4	2	0	0	8	8	0	0	0	0	0	0	0	3	0	0		
8	0.1	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.1	0.0	0.1	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	
	2	6	0	0	6	0	2	0	0	2	6	2	0	0	0	0	0	1	0	6	0	0	0	0	0	0	2	0	3	0	0	
9	0.4	0.0	0.0	0.0	0.2	0.0	0.2	0.7	0.0	0.1	0.2	0.2	0.2	0.0	0.3	0.1	0.0	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7	
	5	9	3	0	0	3	3	2	0	2	6	3	4	5	7	5	4	3	9	2	2	2	2	2	2	2	4	2	5	2	2	
10	0.4	0.1	0.0	0.0	0.1	0.1	0.2	0.8	0.1	0.0	0.2	0.2	0.2	0.3	0.1	0.4	0.2	0.1	0.0	0.1	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	
	0	6	3	4	7	0	1	0	3	0	3	0	1	3	5	1	7	6	0	0	0	0	0	0	0	0	0	2	0	0		
11	0.2	0.1	0.0	0.0	0.0	0.0	0.2	0.6	0.1	0.1	0.0	0.1	0.1	0.2	0.0	0.1	0.0	0.0	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.7	0.6	0.7	0.6		
	3	0	0	4	4	4	4	9	0	2	0	9	2	3	1	9	4	0	4	9	9	9	9	9	9	9	5	9	2	9	9	
12	0.3	0.0	0.0	0.0	0.0	0.0	0.2	0.7	0.1	0.0	0.2	0.0	0.2	0.0	0.3	0.1	0.0	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	
	1	7	0	6	9	6	1	4	2	2	1	0	5	9	5	9	8	2	6	4	4	4	4	4	4	4	7	4	4	4		
13	0.3	0.1	0.0	0.0	0.0	0.0	0.2	0.6	0.0	0.1	0.1	0.2	0.0	0.0	0.2	0.1	0.1	0.0	0.0	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.7	0.6	0.6	0.6		
	5	5	0	0	9	6	3	0	6	8	9	4	0	8	4	1	0	0	6	0	0	0	0	0	0	0	2	0	3	0	0	
14	0.4	0.1	0.0	0.0	0.2	0.0	0.2	0.7	0.0	0.1	0.2	0.2	0.2	0.0	0.3	0.1	0.0	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7	
	5	5	0	0	0	0	2	6	2	6	4	0	7	2	7	0	6	2	2	2	2	2	2	2	2	4	2	5	2	2		
15	0.1	0.1	0.0	0.0	0.0	0.0	0.1	0.5	0.0	0.1	0.0	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	
	5	5	0	0	6	0	7	2	6	8	9	8	5	8	0	7	0	0	6	2	2	2	2	2	2	2	4	2	5	2	2	
16	0.3	0.1	0.0	0.0	0.1	0.0	0.2	0.6	0.0	0.1	0.2	0.2	0.1	0.0	0.2	0.0	0.0	0.0	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.7	0.6	0.6	0.6		
	6	3	0	0	9	1	0	4	7	8	0	8	3	7	8	0	9	1	7	4	4	4	4	4	4	4	6	4	7	4	4	
17	0.4	0.0	0.0	0.0	0.2	0.0	0.2	0.7	0.1	0.1	0.2	0.2	0.2	0.0	0.3	0.1	0.0	0.0	0.1	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7	
	7	9	3	5	3	5	3	2	1	7	8	3	4	5	9	2	0	5	1	2	2	2	2	2	2	4	2	4	2	2		
18	0.5	0.1	0.0	0.0	0.2	0.1	0.2	0.7	0.1	0.1	0.3	0.2	0.2	0.1	0.4	0.2	0.1	0.0	0.1	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7	
	5	9	0	4	0	0	7	7	0	8	2	6	8	3	8	5	4	0	0	7	7	7	7	7	7	8	7	9	7	7		
19	0.4	0.0	0.0	0.0	0.1	0.0	0.0	0.2	0.7	0.0	0.1	0.2	0.2	0.2	0.0	0.4	0.2	0.1	0.0	0.0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7
	5	9	0	0	4	6	3	2	6	2	6	0	4	8	3	1	0	0	0	2	2	2	2	2	2	2	8	2	5	2	2	

**Tabel 4.44 (lanjutan)**

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
20	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
21	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
22	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
23	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
24	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
25	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
26	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
27	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
28	0.0 0																															
29	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
30	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
31	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									
32	0.1 2	0.0 6	0.0 0	0.0 6	0.0 0	0.1 2	0.0 0	0.0 2	0.1 6	0.0 2	0.1 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 3	0.0 0	0.0 0									

**Tabel 4.45 Derajat preferensi d(1,2) untuk setiap kriteria**

fj (...)	fKB(.)	fKS(.)	fPO(.)	fPE(.)	fKO(.)	fPD(.)	fKK(.)	fPM(.)	fPK(.)	fME(.)	fMO(.)	fUS(.)	fPT(.)	fSI(.)	fKJ(.)	fKT(.)	fPG(.)	fRK(.)	fRS(.)	fPP(.)	fIK(.)
	Tipe 4																				
d(1,2)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	

Contoh perhitungan indeks preferensi  $\Pi(1,2)$ :

- Mencari selisih (1,2) untuk setiap kriteria. Hasil selisih (1,2) ditunjukkan pada Tabel 4.42.
- Mencari derajat preferensi  $d(1,2)$  untuk setiap kriteria. Hasil derajat preferensi ditunjukkan pada Tabel 4.45.
- Menghitung indeks preferensi  $\Pi(1,2) = (0*0.012) + (0*0.054) + (0*0.054) + (0*0.049) + (0*0.054) + (0*0.054) + (0*0.059) + (0*0.018) + (0*0.027) + (0*0.057) + (0*0.057) + (0*0.029) + (0*0.119) + (0*0.029) + (0*0.057) + (1*0.059) + (0*0.030) + (0*0.088) + (0*0.037) + (0*0.030) + (0*0.029) = 0.059$
- Menghitung *leaving flow* dan *entering flow*. Hasil *leaving flow* dan *entering flow* ditunjukkan pada Tabel 4.46.

**Tabel 4.46 Hasil *Leaving flow* dan *entering flow***

No.	Pendaftar	Leaving flow	Entering flow
1	P1	0.278	0.265
2	P2	0.397	0.097
3	P3	0.492	0.008
4	P4	0.455	0.015
5	P5	0.419	0.111
6	P6	0.421	0.030
7	P7	0.413	0.173
8	P8	0.028	0.407
9	P9	0.415	0.050
10	P10	0.461	0.123
11	P11	0.374	0.149
12	P12	0.405	0.171
13	P13	0.344	0.116
14	P14	0.414	0.056
15	P15	0.284	0.192
16	P16	0.369	0.088
17	P17	0.424	0.065
18	P18	0.467	0.014
19	P19	0.415	0.066
20	P20	0.028	0.407

**Tabel 4.46 (lanjutan)**

No.	Pendaftar	<i>Leaving flow</i>	<i>Entering flow</i>
21	P21	0.028	0.407
22	P22	0.028	0.407
23	P23	0.028	0.407
24	P24	0.028	0.407
25	P25	0.028	0.407
26	P26	0.028	0.407
27	P27	0.028	0.407
28	P28	0.003	0.503
29	P29	0.028	0.407
30	P30	0.027	0.436
31	P31	0.028	0.407
32	P32	0.028	0.407

Contoh perhitungan *leaving flow* untuk P1:

$$\Phi^+(P1) = 1/(32-1) \cdot (0.00 + 0.06 + 0.00 + 0.00 + \dots + 0.56) = 0.278$$

Contoh perhitungan *entering flow* untuk P1:

$$\Phi^-(P1) = 1/(32-1) \cdot (0.00 + 0.38 + 0.54 + 0.48 + 0.31 + \dots + 0.12) = 0.265$$

5. PROMETHEE I: peringkat parsial berdasarkan hasil *leaving flow* dan *entering flow*, sehingga menghasilkan bentuk hubungan antara setiap peserta seleksi. Hubungan antarpeserta seleksi diketahui dengan Persamaan (3.6), terdapat urutan peserta seleksi yang tidak dapat dibandingkan (*incomparable*), maka dilanjutkan dengan perhitungan PROMETHEE II.

Contoh hubungan yang *incomparable* yaitu hubungan antara P2 dan P5:

P2 S+ P5 =  $\phi^+(P2) > \phi^+(P5) \parallel \phi^+(P2) = \phi^+(P5)$ , bernilai *false*.

P2 S- P5 =  $\phi^-(P2) > \phi^-(P5) \parallel \phi^-(P2) = \phi^-(P5)$ , bernilai *true*.

P2 S+ P5 bernilai *false* dan P2 S- P5 bernilai *true*, maka P2 Pi P5 bernilai *false*.

P2 Ii P5 =  $\phi^+(P2) = \phi^+(P5) \&& \phi^-(P2) = \phi^-(P5)$ , bernilai *false*.

Karena tidak memenuhi kondisi P2 Pi P5 atau P2 Ii P5, maka hubungan antara P2 dan P5 adalah *incomparable* (P2 R P5 bernilai *true*).

6. Menghitung *net flow* yaitu dengan pengurangan antara *leaving flow* dan *entering flow*. Hasil *net flow* ditunjukkan pada Tabel 4.47.

**Tabel 4.47 Hasil *net flow***

No.	Pendaftar	Net flow
1	P1	0.013
2	P2	0.300
3	P3	0.484
4	P4	0.440
5	P5	0.309
6	P6	0.391
7	P7	0.240
8	P8	-0.379
9	P9	0.365
10	P10	0.338
11	P11	0.226
12	P12	0.234
13	P13	0.229
14	P14	0.358
15	P15	0.092
16	P16	0.281
17	P17	0.359
18	P18	0.454
19	P19	0.349
20	P20	-0.379
21	P21	-0.379
22	P22	-0.379
23	P23	-0.379
24	P24	-0.379
25	P25	-0.379
26	P26	-0.379
27	P27	-0.379
28	P28	-0.500
29	P29	-0.379
30	P30	-0.409
31	P31	-0.379
32	P32	-0.379

Contoh perhitungan *net flow* untuk P1:

$$\phi(P1) = \phi+(P1) - \phi-(P1) = 0.278 - 0.265 = 0.013$$

7. PROMETHEE II: memberikan peringkat lengkap peserta seleksi dengan mempertimbangkan nilai *net flow*. Peringkat lengkap peserta seleksi ditunjukkan pada Tabel 4.48.

**Tabel 4.48 Peringkat lengkap PROMETHEE II**

Pendaftar	Net flow	Peringkat
P3	0.484	1
P18	0.454	2
P4	0.440	3
P6	0.391	4
P9	0.365	5
P17	0.359	6
P14	0.358	7
P19	0.349	8
P10	0.338	9
P5	0.309	10
P2	0.300	11
P16	0.281	12
P7	0.240	13
P12	0.234	14
P13	0.229	15
P11	0.226	16
P15	0.092	17
P1	0.013	18
P8	-0.379	19
P20	-0.379	20
P21	-0.379	21
P22	-0.379	22
P23	-0.379	23
P24	-0.379	24
P25	-0.379	25
P26	-0.379	26
P27	-0.379	27
P29	-0.379	28
P31	-0.379	29
P32	-0.379	30
P30	-0.409	31
P28	-0.500	32

## BAB V

# IMPLEMENTASI

### 5.1 Implementasi Metode *Modified Delphi*

Implementasi metode *Modified Delphi* menjelaskan implementasi metode ke dalam sistem sesuai dengan rancangan yang sudah dibuat pada BAB IV. Berikut ini implementasinya:

1. Implementasi proses memasukkan kriteria dari studi literatur.

Implementasi ini digunakan untuk mengelola kriteria dari studi literatur oleh peneliti, yang terdiri dari fasilitas menambah, mengubah, melihat detail dan menghapus kriteria. Implementasi halaman pengelolaan kriteria dari studi literatur dapat dilihat pada Gambar 5.1.

No	Kriteria	Penjelasan Kriteria	Aksi
1	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]
2	Kesungguhan	Kriteria yang digunakan untuk mengukur tingkat kesungguhan pendaftar. Kesungguhan dapat tercermin dari apakah pendaftar mengikuti semua proses seleksi dengan baik, datang tepat waktu, sesuai persyaratan, dll. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]
3	Potensi	Kriteria yang digunakan untuk mengukur potensi. Potensi yaitu kemampuan dan kualitas yang dimiliki oleh seseorang namun belum dipergunakan secara maksimal. Data diperoleh dari pengukuran potensi diri yang dilakukan melalui diri sendiri (self assessment), melalui feedback dari orang lain dan tes-tes psikologis seperti tes kecerdasan, tes kepribadian, tes kepemimpinan, tes kreativitas dll.	[Detail] [Edit] [Delete]
4	Keuangan	Kriteria yang digunakan untuk mengetahui konsisi ekonomi pendaftar. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]
5	Komitmen	Kriteria yang digunakan untuk mengetahui komitmen pendaftar apakah setelah dinyatakan lulus, pendaftar tidak akan mundurkan diri dan serius mengikuti pelatihan. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]
6	Kemampuan emosional	Kriteria yang digunakan untuk mengetahui kemampuan emosional. Data diperoleh dari tes EQ (Emotional Questions).	[Detail] [Edit] [Delete]
7	Percaya diri	Kriteria yang digunakan untuk mengetahui tingkat kepercayaan diri pendaftar. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]
8	Keterampilan komunikasi	Kriteria yang digunakan untuk mengetahui kemampuan berkomunikasi. Komunikasi yang baik akan mempermudah untuk menangkap ilmu dalam pelatihan. Data diperoleh dari wawancara.	[Detail] [Edit] [Delete]

**Gambar 5.1 Implementasi halaman pengelolaan kriteria dari studi literatur**

Potongan *source code* implementasi pengelolaan kriteria dari studi literatur dapat dilihat pada Tabel 5.1. Baris 3-7 digunakan untuk meng-*import* kelas-kelas yang digunakan pada CriteriaController. Baris 11-29 merupakan fungsi untuk menampilkan daftar kriteria dari studi literatur. Baris 31-39 digunakan untuk menampilkan halaman tambah kriteria baru. Baris 41-54 digunakan untuk

memproses masukan dari halaman tambah kriteria baru dan menyimpannya ke dalam basis data. Baris 56-65 digunakan untuk menampilkan detail suatu kriteria. Baris 67-76 digunakan untuk menampilkan halaman edit kriteria dan akan diproses serta disimpan ke dalam basis data dengan fungsi *update* pada baris 78-88. Baris 90-102 digunakan untuk menghapus kriteria.

**Tabel 5.1 Potongan *source code* implementasi pengelolaan kriteria dari studi literatur**

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\Criteria;
6 use App\Http\Models\Choice;
7 use Auth;
8
9 class CriteriaController extends Controller
{
10 public function index(Request $request)
{
11     $role_id = Auth::user()->roleId();
12     if ($role_id == 1) {
13         $criterias = Criteria::select('*')
14             ->where('description', '!=', 'null')
15             ->where('step', '=', '1')
16             ->where('status', '=', '1')
17             ->whereNotIn('id', function($query){
18                 $query->select('criteria_id')
19                 ->from(with(new Choice)->getTableName())
20                 ->where('suggestion', 1);
21             })->orderBy('id', 'DESC')->paginate(10);
22         return view('criterias.index', compact('criterias'))
23             ->with('i', ($request->input('page', 1) - 1) * 10);
24     } else {
25         return redirect()->route('profile_users.show');
26     }
27 }
28
29
30 public function create()
{
31     $role_id = Auth::user()->roleId();
32     if ($role_id == 1) {
33         return view('criterias.create');
34     } else {
35         return redirect()->route('profile_users.show');
36     }
37 }
38
39
40 public function store(Request $request)
{
41     $this->validate($request, [
42         'name' => 'required',
43

```

```

45 'description' => 'required',
46 'citation' => 'required',
47 ]);
48 $input = $request->all();
49 $input['step'] = '1';
50 $input['status'] = '1';
51 Criteria::create($input);
52 return redirect()->route('criterias.index')
53 ->with('success','Kriteria berhasil dibuat');
54 }
55
56 public function show($id)
57 {
58 $role_id = Auth::user()->roleId();
59 if ($role_id == 1) {
60 $criteria = Criteria::find($id);
61 return view('criterias.show',compact('criteria'));
62 } else {
63 return redirect()->route('profile_users.show');
64 }
65 }
66
67 public function edit($id)
68 {
69 $role_id = Auth::user()->roleId();
70 if ($role_id == 1) {
71 $criteria = Criteria::find($id);
72 return view('criterias.edit',compact('criteria'));
73 } else {
74 return redirect()->route('profile_users.show');
75 }
76 }
77
78 public function update(Request $request, $id)
79 {
80 $this->validate($request, [
81 'name' => 'required',
82 'description' => 'required',
83 'citation' => 'required',
84 ]);
85 Criteria::find($id)->update($request->all());
86 return redirect()->route('criterias.index')
87 ->with('success','Kriteria berhasil diedit');
88 }
89
90 public function destroy($id)
91 {
92 $choice = Choice::where('criteria_id', '=', $id)->first();
93 if ($choice == null) {
94 Criteria::find($id)->delete();
95 return redirect()->route('criterias.index')
96 ->with('success','Kriteria berhasil dihapus');
97 } else {
98 return redirect()->route('criterias.index')
99 ->with('failed','Kriteria tidak bisa dihapus karena sudah ada penilaian kesesuaian
100 kriteria oleh tim penilai');
101 }
102 }
103 }

```

## 2. Implementasi proses mengisi kuesioner kriteria

Implementasi ini digunakan untuk mengisi kuesioner dari daftar kriteria berdasarkan studi literatur oleh ahli yang sudah ditentukan. Para ahli juga diijinkan memberi gagasan masukan kriteria tambahan. Implementasi halaman isian kuesioner kriteria dapat dilihat pada Gambar 5.2.

No	Kriteria	Penjelasan Kriteria	Sumber Pustaka	Pilihan
1	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Data diperoleh dari wawancara.	(Barrett, dkk., 2003)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai
2	Kesungguhan	Kriteria yang digunakan untuk mengukur tingkat kesungguhan pendaftar. Kesungguhan dapat tercermin dari apakah pendaftar mengikuti semua proses seleksi dengan baik, datang tepat waktu, sesuai persyaratan, dll. Data diperoleh dari wawancara.	(Blume, dkk., 2010) (Vinchur, dkk., 1998) (Barrett, dkk., 2003) (Hough dan Oswald, 2000) (Robertson dan Smith, 2001)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai
3	Potensi	Kriteria yang digunakan untuk mengukur potensi. Potensi yaitu kemampuan dan kualitas yang dimiliki oleh seseorang namun belum dipergunakan secara maksimal. Data diperoleh dari pengukuran potensi diri yang dilakukan melalui diri sendiri (self assessment), melalui feedback dari orang lain dan tes-tes psikologis seperti tes kecerdasan, tes kepribadian, tes kepemimpinan, tes kreativitas dll.	(Vinchur, dkk., 1998) (Robertson dan Smith, 2001)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai
4	Keuangan	Kriteria yang digunakan untuk mengetahui konsisi ekonomi pendaftar. Data diperoleh dari wawancara.	(Noe, dkk., 2012)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai
5	Komitmen	Kriteria yang digunakan untuk mengetahui komitmen pendaftar apakah setelah dinyatakan lulus, pendaftar tidak akan mengundurkan diri dan serius mengikuti pelatihan. Data diperoleh dari wawancara.	(Ubaldi, 2015)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai
32	Pengalaman pelatihan	Kriteria yang digunakan untuk mengetahui keterampilan yang sudah dimilikinya. Data diperoleh dari pengalaman pelatihan yang dilisikan pendaftar dan wawancara.	(Noe, dkk., 2012) (Ivancevich dan Konopaske, 2013) (Bogdanovic dan Miletic, 2014) (Kumar dkk., 2013) (Rouyendegh dan Erkan, 2013) (El-santawy, 2012) (Blume, dkk., 2010) (Robertson dan Smith, 2001)	<input type="radio"/> Sesuai <input type="radio"/> Tidak Sesuai

**Masukan**  
Isikan kriteria tambahan beserta keterangannya yang sesuai dengan proses seleksi peserta pelatihan BLK Bantul.

Kriteria	Penjelasan Kriteria	+ Tambah
----------	---------------------	----------

Submit

**Gambar 5.2 Implementasi halaman kuesioner kriteria**

Para ahli hanya diijinkan mengisi satu kali saja dan tidak boleh mengubah isian yang sudah di-submit. Setelah mengisi kuesioner, para ahli dapat melihat hasil isiannya, seperti yang ditunjukkan pada Gambar 5.3.

Potongan *source code* implementasi kuesioner kriteria pada Delphi putaran pertama dapat dilihat pada Tabel 5.2. Baris 50-74 digunakan untuk menampilkan halaman isian kuesioner kriteria dan diproses dengan fungsi *store* pada baris 76-149. Kuesioner yang sudah di-submit akan menampilkan halaman hasil isian kuesioner kriteria yang diimplementasi pada baris 13-48.

No	Kriteria	Penjelasan Kriteria	Sumber Pustaka	Pilihan
1	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Data diperoleh dari wawancara.	(Barrett, dkk., 2003)	Sesuai
2	Kesungguhan	Kriteria yang digunakan untuk mengukur tingkat kesungguhan pendaftar. Kesungguhan dapat tercermin dari apakah pendaftar mengikuti semua proses seleksi dengan baik, datang tepat waktu, sesuai persyaratan, dll. Data diperoleh dari wawancara.	(Blume, dkk., 2010) (Vinchur, dkk., 1998) (Barrett, dkk., 2003) (Hough dan Oswald, 2000) (Robertson dan Smith, 2001)	Sesuai
3	Potensi	Kriteria yang digunakan untuk mengukur potensi. Potensi yaitu kemampuan dan kualitas yang dimiliki oleh seseorang namun belum diungkapkan secara maksimal. Data diperoleh dari pengukuran potensi yang dilakukan melalui diri sendiri (self assessment), melalui feedback dari orang lain dan tes-psikologi seperti tes kecerdasan, tes kepribadian, tes kepemimpinan, tes kreativitas dll.	(Vinchur, dkk., 1998) (Robertson dan Smith, 2001)	Tidak Sesuai
4	Keuangan	Kriteria yang digunakan untuk mengetahui konsisi ekonomi pendaftar. Data diperoleh dari wawancara.	(Noe, dkk., 2012)	Tidak Sesuai
5	Komitmen	Kriteria yang digunakan untuk mengetahui komitmen pendaftar apakah setelah dinyatakan lulus, pendaftar tidak akan mengundurkan diri dan serius mengikuti pelatihan. Data diperoleh dari wawancara.	(Ubaidi, 2015)	Sesuai
6	Kemantapan emosional	Kriteria yang digunakan untuk mengetahui kemantapan emosional. Data diperoleh dari tes EO (Emotional Outcomes).	(Chen, dkk., 2013) (Vinchur, dkk., 1998) (Barrett, dkk., 2003)	Tidak Sesuai
32	Pengalaman pelatihan	Kriteria yang digunakan untuk mengetahui keterampilan yang sudah dimilikinya. Data diperoleh dari pengalaman pelatihan yang diisikan pendaftar dan wawancara.	(Noe, dkk., 2012) (Ivanecovich dan Konopaske, 2013) (Bogdanovic dan Miletic, 2014) (Kumar dkk., 2013) (Rouyendegh dan Erkan, 2013) (El-santawy, 2012) (Blume, dkk., 2010) (Robertson dan Smith, 2001)	Sesuai

**Masukan**

No	Kriteria	Penjelasan Kriteria
1	Intensitas keikutsertaan pelatihan di BLK Bantul	Untuk mengetahui seberapa banyak pendaftar pernah mengikuti pelatihan di BLK Bantul

Gambar 5.3 Implementasi hasil isian kuesioner kriteria

Tabel 5.2 Potongan *source code* implementasi kuesioner kriteria

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Models\User;
6 use App\Models\Choice;
7 use App\Models\Criteria;
8 use Auth;
9 use Carbon;
10
11 class QuestionnaireController extends Controller
12 {
13 public function index()
14 {
15 $role_id = Auth::user()->roleId();
16 $user = User::find(Auth::user()->id);
17 $data = Choice::where('user_id', '=', $user->id)->first();
18 if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
19 if ($user->id == 1) {
20 return redirect()->route('questionnaire.create')
21 ->with('failed','Maaf, peneliti tidak perlu mengisi kuesioner ini.
22 Apapun yang disubmit tidak akan tersimpan dalam database');
23 }
24 if ($data == null) {
25 return redirect()->route('questionnaire.create');
26 } else {

```

```

27 $i = 0;
28 $j = 0;
29 $data_standart = Choice::select('choice.*', 'criterias.*')
30 ->join('criterias','criterias.id','=',$choice.criteria_id')
31 ->where('choice.suggestion', '=', '0')
32 ->where('choice.user_id', '=', $user->id)
33 ->where('criterias.step', '=', '1')
34 ->where('criterias.status', '=', '1')
35 ->orderBy('criterias.id','DESC')->get();
36 $data_suggestion = Choice::select('choice.*', 'criterias.*')
37 ->join('criterias','criterias.id','=',$choice.criteria_id')
38 ->where('choice.suggestion', '=', '1')
39 ->where('choice.user_id', '=', $user->id)
40 ->where('criterias.step', '=', '2')
41 ->where('criterias.status', '=', '1')
42 ->orderBy('criterias.id','ASC')->get();
43 return view('questionnaire.index', compact('data_standart', 'data_suggestion', 'i', 'j'));
44 }
45 } else {
46 return redirect()->route('profile_users.show');
47 }
48 }
49
50 public function create(Request $request)
51 {
52 $role_id = Auth::user()->roleId();
53 $user = User::find(Auth::user()->id);
54 $data = Choice::where('user_id', '=', $user->id)->first();
55 if ($role_id == 3 || $role_id == 4 || $role_id == 5 ||$role_id == 6) {
56 if ($data != null) {
57 return redirect()->route('questionnaire.index');
58 } else {
59 $i = 0;
60 $criteria = Criteria::where('description','<>','null')
61 ->where('step','=', '1')
62 ->where('status','=', '1')
63 ->whereNotIn('id', function($query){
64 $query->select('criteria_id')
65 ->from(with(new Choice)->getTable())
66 ->where('suggestion', 1);
67 })
68 ->orderBy('id','DESC')->get();
69 return view('questionnaire.create',compact('criteria', 'i'));
70 }
71 } else {
72 return redirect()->route('profile_users.show');
73 }
74 }
75
76 public function store(Request $request)
77 {
78 $user = User::find(Auth::user()->id);
79 if ($user->id == 1) {
80 return redirect()->route('questionnaire.create')
81 ->with('failed','Maaf, peneliti tidak perlu mengisi kuesioner ini.
82 Apapun yang disubmit tidak akan tersimpan dalam database');
83 }
84 $input = $request->all();
85 $valid = true;
86 $optional = array();

```

```

87 $i = 0;
88 foreach ($input["criteriamore"] as $crit) {
89 if (($crit != "") && ($input["descriptionmore"][$i] != "")) {
90 $optional[$i]["criteria"] = $crit;
91 $optional[$i]["description"] = $input["descriptionmore"][$i];
92 } else if (($crit == "") && ($input["descriptionmore"][$i] == "")) {
93 } else {
94 $valid = false;
95 break;
96 }
97 $i = $i + 1;
98 }
99 $choices = array();
100 if ($valid) {
101 $criteria = Criteria::where('description','<>','null')
->where('step','=',1)
->where('status','=',1)
->whereNotIn('id', function($query){
105 $query->select('criteria_id')
->from(with(new Choice)->getTableName())
->where('suggestion', 1);
})
->orderBy('id','DESC')->lists('id');
110 foreach ($criteria as $value) {
111 if (!array_key_exists($value,$input)) {
112 $valid = false;
113 break;
114 } else {
115 $choices[$value] = $input[$value];
116 }
117 }
118 }
119 if ($valid) {
120 foreach ($choices as $criteriaid=>$option) {
121 $data["user_id"] = $user->id;
122 $data["criteria_id"] = $criteriaid;
123 $data["option"] = $option;
124 $data["suggestion"] = 0;
125 Choice::create($data);
126 }
127 foreach ($optional as $optionalCriteria) {
128 $dataoptional["name"] = $optionalCriteria["criteria"];
129 $dataoptional["description"] = $optionalCriteria["description"];
130 $dataoptional["step"] = 2;
131 $dataoptional["status"] = 1;
132 $dataoptional["created_at"] = Carbon\Carbon::now(7)->toDateTimeString();
133 $dataoptional["updated_at"] = $dataoptional["created_at"];
134 $suggest = Criteria::create($dataoptional);
135 $optionalChoice["user_id"] = $user->id;
136 $optionalChoice["criteria_id"] = $suggest->id;
137 $optionalChoice["option"] = 1;
138 $optionalChoice["suggestion"] = 1;
139 Choice::create($optionalChoice);
140 }
141 return redirect()->route('questionnaire.index')
->with('success','Selamat Anda berhasil mengisi kuesioner kriteria.
Data yang sudah diisikan tidak dapat diubah.');
144 } else {
145 return redirect()->route('questionnaire.create')
->with('failed','Maaf! Semua pilihan kriteria harus diisi.');
146 }

```

147	}
148	}
149	}

Implementasi saat ahli mengisi kuesioner kriteria dapat dilihat pada Gambar 5.6.



**Gambar 5.4 Implementasi ahli mengisi kuesioner kriteria**

### 3. Implementasi proses menampilkan hasil Delphi putaran pertama

Implementasi ini digunakan untuk menampilkan hasil Delphi putaran pertama yang berisi kalkulasi pilihan kesesuaian kriteria dan gagasan dari para ahli. Implementasi halaman hasil Delphi putaran pertama dapat dilihat pada Gambar 5.5.

No	Kriteria	Penjelasan Kriteria	Sumber Pustaka	Kesesuaian (%)	Jumlah Sesuai dari Keseluruhan
1	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Data diperoleh dari wawancara.	(Barrett, dkk., 2003)	100	9 dari 9
2	Kesungguhan	Kriteria yang digunakan untuk mengukur tingkat kesungguhan pendaftar. Kesungguhan dapat tercerminkan dari apakah pendaftar mengikuti semua proses seleksi dengan baik, datang tepat waktu, sesuai persyaratan, dll. Data diperoleh dari wawancara.	(Blume, dkk., 2010) (Vincze, dkk., 1998) (Barrett, dkk., 2003) (Hough dan Osswald, 2000) (Robertson dan Smith, 2001)	100	9 dari 9
3	Potensi	Kriteria yang digunakan untuk mengukur potensi. Potensi yaitu kemampuan dan keadaan yang dimiliki oleh seseorang untuk melanjutkan pengembangan secara maksimal. Data diperoleh dari pengeksploran potensi diri yang dilakukan melalui diri sendiri (self assessment), melalui feedback dari orang lain dan tes-tes psikologis seperti tes kecerdasan, tes kepribadian, tes kepemimpinan, tes kreativitas dll.	(Vinchur, dkk., 1998) (Robertson dan Smith, 2001)	77.777777777778	7 dari 9
4	Keuangan	Kriteria yang digunakan untuk mengetahui konsisi ekonomi pendaftar. Data diperoleh dari wawancara.	(Noe, dkk., 2012)	77.777777777778	7 dari 9
5	Komitmen	Kriteria yang digunakan untuk mengetahui komitmen pendaftar apakah setelah dinyatakan lulus, pendaftar tidak akan mengundurkan diri dan serius mengikuti pelatihan. Data diperoleh dari wawancara.	(Ubaidi, 2015)	100	9 dari 9
6	Kemampuan emosional	Kriteria yang digunakan untuk mengetahui kemampuan emosional. Data diperoleh dari tes EQ (Emotional Questions).	(Chen, dkk., 2013) (Vinchur, dkk., 1998) (Barrett, dkk., 2003)	11.111111111111	1 dari 9
32	Pengalaman pelatihan	Kriteria yang digunakan untuk mengetahui keterampilan yang sudah dimilikinya. Data diperoleh dari pengalaman pelatihan yang diikuti pendaftar dan wawancara.	(Noe, dkk., 2012) (Vancevich dan Konopacke, 2013) (Bogdanovic dan Miletic, 2014) (Kumar dkk., 2013) (Rouyenegeh dan Erkan, 2013) (El-santawy, 2012) (Blume, dkk., 2010) (Robertson dan Smith, 2001)	88.888888888889	8 dari 9

**Masukan**

No	Kriteria	Penjelasan Kriteria	Pengusul
1	Intensitas keikutsertaan pelatihan di BLK Bantul	Untuk mengetahui seberapa banyak pendaftar pernah mengikuti pelatihan di BLK Bantul	Siti Astuti S.E.

**Gambar 5.5 Implementasi halaman hasil Delphi putaran pertama**

Potongan *source code* implementasi hasil Delphi putaran pertama dapat dilihat pada Tabel 5.3. Baris 13-73 digunakan untuk menampilkan hasil kalkulasi pilihan kesesuaian kriteria dan gagasan dari para ahli.

**Tabel 5.3 Potongan *source code* implementasi hasil Delphi putaran pertama**

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\User;
6 use App\Http\Models\Choice;
7 use App\Http\Models\Criteria;
8 use Auth;
9 use DB;
10
11 class ResultStep1Controller extends Controller
12 {
13     public function index(Request $request)
14     {
15         $role_id = Auth::user()->roleId();
16         $user = User::find(Auth::user()->id);
17         $data = Choice::where('user_id', '=', $user->id)->first();
18         if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
19             if ($user->id == 1) {
20                 $i = 0;
21                 $j = 0;
22                 $percentages = Choice::select('criterias.name', 'criterias.description',
23                     'criterias.citation',
24                     DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
25                     DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
26                     ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
27                     ->where('choice.suggestion', '=', '0')
28                     ->where('criterias.step', '=', '1')
29                     ->where('criterias.status', '=', '1')
30                     ->groupBy('criterias.id')
31                     ->orderBy('criterias.id', 'DESC')
32                     ->get();
33                 $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
34                     ->join('choice', 'choice.user_id', '=', 'users.id')
35                     ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
36                     ->where('choice.suggestion', '=', '1')
37                     ->where('criterias.step', '=', '2')
38                     ->where('criterias.status', '=', '1')
39                     ->orderBy('criterias.id', 'ASC')
40                     ->get();
41                 return view('result_step1.index', compact('percentages', 'data_suggestion', 'i', 'j'));
42             }
43             if ($data == null) {
44                 return redirect()->route('questionnaire.create')
45                 ->with('failed', 'Maaf, silahkan isi kuesioner kriteria dahulu.');
46             } else {
47                 $i = 0;
48                 $j = 0;
49                 $percentages = Choice::select('criterias.name', 'criterias.description',
50                     'criterias.citation',

```

```

51 DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
52 DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
53 ->join('criterias','criterias.id','=',$choice->criteria_id')
54 ->where('choice.suggestion', '=', '0')
55 ->where('criterias.step', '=', '1')
56 ->where('criterias.status', '=', '1')
57 ->groupBy('criterias.id')
58 ->orderBy('criterias.id','DESC')
59 ->get();
60 $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
61 ->join('choice', 'choice.user_id', '=', 'users.id')
62 ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
63 ->where('choice.suggestion', '=', '1')
64 ->where('criterias.step', '=', '2')
65 ->where('criterias.status', '=', '1')
66 ->orderBy('criterias.id','ASC')
67 ->get();
68 return view('result_step1.index', compact('percentages', 'data_suggestion', 'i', 'j'));
69 }
70 } else {
71 return redirect()->route('profile_users.show');
72 }
73 }
74 }

```

#### 4. Implementasi proses Delphi putaran kedua

Implementasi ini digunakan untuk menentukan kriteria dalam penilaian seleksi peserta pelatihan. Kriteria ditentukan berdasarkan kesepakatan melalui diskusi yang dilakukan di luar sistem. Sistem digunakan untuk menyimpan hasil diskusi tersebut. Implementasi halaman pengelolaan kriteria dalam Delphi putaran kedua dapat dilihat pada Gambar 5.6.

Potongan *source code* implementasi Delphi putaran kedua dapat dilihat pada Tabel 5.4. Baris 15-115 digunakan untuk menampilkan halaman utama pengelolaan kriteria dalam Delphi putaran kedua. Baris 117-120 digunakan untuk menampilkan halaman tambah kriteria baru dan masukan akan diproses dengan fungsi *store* pada baris 122-140. Baris 142-146 digunakan untuk menampilkan halaman ubah data kriteria dan akan diproses dengan fungsi *update* pada baris 148-157. Baris 159-174 digunakan untuk menghapus kriteria. Baris 176-195 digunakan untuk menggunakan kriteria pada Delphi putaran pertama.

**Kriteria Tahap 2**

No	Kriteria	Penjelasan Kriteria	Informasi	Aksi
1	Intensitas keikutsertaan pelatihan di BLK Bantul	Untuk mengetahui seberapa banyak pendaftar pernah mengikuti pelatihan di BLK Bantul	Belum pernah: 4 Pernah ikut 1 kali: 3 Pernah ikut 2 kali: 2 Pernah ikut > 3 kali: 1	<a href="#">Edit</a> <a href="#">Delete</a>
2	Pengalaman pelatihan	Kriteria yang digunakan untuk mengetahui keterampilan yang sudah dimilikinya. Data diperoleh dari pengalaman pelatihan yang dilakukan pendaftar dan wawancara.	Pernah ikut pelatihan di luar BLK dan sesuai bidang yang diminati: 3 Pernah ikut pelatihan di luar BLK dan tidak sesuai bidang yang diminati: 2 Belum pernah: 1	<a href="#">Edit</a> <a href="#">Delete</a>
3	Rencana setelah selesai pelatihan	Kriteria yang digunakan untuk mengetahui rencana yang dilakukan pendaftar setelah mengikuti pelatihan. Data diperoleh dari wawancara.	Membuka usaha: 3 Melamar pekerjaan: 2 Menambah ilmu/keterampilan: 1 Tidak ada: 0	<a href="#">Edit</a> <a href="#">Delete</a>
4	Rekomendasi	Kriteria yang digunakan untuk mengetahui apakah pendaftar merupakan rekomendasi dari pegawai atau pihak tertentu. Rekomendasi ini disusulkan oleh suatu pihak yang menyatakan bahwa seseorang memiliki potensi yang baik. Data diperoleh dari wawancara dan pihak yang memberikan rekomendasi.	Ada: 2 Tidak: 1	<a href="#">Edit</a> <a href="#">Delete</a>
5	Pengetahuan	Kriteria yang digunakan untuk mengukur tingkat pemahaman seseorang terhadap suatu bidang keilmuan. Data diperoleh dari ujian tertulis.		<a href="#">Edit</a> <a href="#">Delete</a>
21	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Data diperoleh dari wawancara.	Baik: 3 Cukup: 2 Kurang: 1	<a href="#">Edit</a> <a href="#">Delete</a>

**Hasil Kriteria Tahap 1**

No	Kriteria	Penjelasan Kriteria	Sumber Pustaka	Kesesuaian (%)	Jumlah Sesuai dari Keseluruhan	Aksi
1	Kemampuan emosional	Kriteria yang digunakan untuk mengetahui kemampuan emosional. Data diperoleh dari tes EQ (Emotional Questions).	(Chen, dkk., 2013) (Vinchur, dkk., 1998)	11.1111111111	1 dari 9	
2	Keterampilan komputer	Kriteria yang digunakan untuk mengukur kemampuan pendaftar dalam hal keterampilan komputer. Data diperoleh dari ujian praktik dengan mengoperasikan komputer.	(Gungör dkk., 2009) (Bogdanovic dan Miletic, 2014)	22.2222222222	2 dari 9	
3	Kemampuan kognitif	Kriteria yang digunakan untuk mengetahui kemampuan kognitif, yang dapat diukur dengan kemampuan verbal dan matematika. Data diperoleh dari tes psikolog.	(Noe, dkk., 2012) (Vancevich dan Konopaske, 2013)	0	0 dari 9	
12	Kesehatan	Kriteria yang digunakan untuk mengetahui kondisi kesehatan jasmani pendaftar yang meliputi tekanan darah, apakah mengidap penyakit tertentu, dll. Data diperoleh dari tes kesehatan, yang ditunjukkan dengan surat keterangan dokter.	(Noe, dkk., 2012) (Fajar dan Heru, 2010) (Safrizal dan Tanti, 2015) (Robertson dan Smith, 2001)	44.4444444444	4 dari 9	

**Masukan**

No	Kriteria	Penjelasan Kriteria	Pengusul	Aksi
----	----------	---------------------	----------	------

Gambar 5.6 Halaman implementasi Delphi putaran kedua

Tabel 5.4 Potongan *source code* implementasi Delphi putaran kedua

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\User;
6 use App\Http\Models\Choice;
7 use App\Http\Models\Criteria;
8 use App\Http\Models\ResultSelection;
9 use App\Http\Models\PairwiseComparison;
10 use Auth;
11 use DB;
12
13 class CriteriaStep2Controller extends Controller
14 {
15     public function index(Request $request)
16     {

```

```

17 $role_id = Auth::user()->roleId();
18 $user = User::find(Auth::user()->id);
19 $data = Choice::where('user_id', '=', $user->id)->first();
20 if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
21 if ($user->id == 1) {
22 $i = 0;
23 $j = 0;
24 $k = 0;
25 $data_fix = Choice::select('choice.*', 'criterias.*')
26 ->join('criterias','criterias.id','=','choice.criteria_id')
27 ->where('choice.suggestion', '=', '0')
28 ->where('criterias.step', '=', '2')
29 ->where('criterias.status', '=', '1')
30 ->orderBy('criterias.id','DESC')
31 ->get();
32 $criteriaCheck = Criteria::select('criterias.ref_id')
33 ->where('criterias.ref_id','<>',null)
34 ->where('criterias.step','=', '2')
35 ->get();
36 $usedCriteria = array();
37 foreach ($criteriaCheck as $used){
38 if ($used["ref_id"] != null) {
39 $usedCriteria[] = $used["ref_id"];
40 }
41 }
42 $data_standart = Choice::select('choice.*', 'criterias.*',
43 DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
44 DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
45 ->join('criterias','criterias.id','=','choice.criteria_id')
46 ->where('choice.suggestion', '=', '0')
47 ->where('criterias.step', '=', '1')
48 ->where('criterias.status', '=', '1')
49 ->whereNotIn('criterias.id', $usedCriteria)
50 ->groupBy('criterias.id')
51 ->orderBy('criterias.id','DESC')
52 ->get();
53 $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
54 ->join('choice','choice.user_id','=','users.id')
55 ->join('criterias','criterias.id','=','choice.criteria_id')
56 ->where('choice.suggestion', '=', '1')
57 ->where('criterias.step', '=', '2')
58 ->where('criterias.status', '=', '1')
59 ->whereNotIn('criterias.id', $usedCriteria)
60 ->orderBy('criterias.id','ASC')
61 ->get();
62 return view('criteria_step2.index', compact('data_standart', 'data_suggestion',
63 'data_fix', 'i', 'j', 'k'));
64 }
65 if ($data == null) {
66 return redirect()->route('questionnaire.create')
67 ->with('failed','Maaf, silahkan isi kuesioner krriteria dahulu.');
68 } else {
69 $i = 0;
70 $j = 0;
71 $k = 0;
72 $data_fix = Choice::select('choice.*', 'criterias.*')
73 ->join('criterias','criterias.id','=','choice.criteria_id')
74 ->where('choice.suggestion', '=', '0')
75 ->where('criterias.step', '=', '2')
76 ->where('criterias.status', '=', '1')

```

```

77 $criterias->orderBy('criterias.id','DESC')
78 $criterias->get();
79 $criteriaCheck = Criteria::select('criterias.ref_id')
80 ->where('criterias.ref_id','<>',null)
81 ->where('criterias.step','=', '2')
82 $criterias->get();
83 $usedCriteria = array();
84 foreach ($criteriaCheck as $used){
85 if ($used["ref_id"] != null) {
86 $usedCriteria[] = $used["ref_id"];
87 }
88 }
89 $data_standart = Choice::select('choice.*', 'criterias.*',
90 DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
91 DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
92 ->join('criterias','criterias.id','=','choice.criteria_id')
93 ->where('choice.suggestion', '=', '0')
94 ->where('criterias.step', '=', '1')
95 ->where('criterias.status', '=', '1')
96 ->whereNotIn('criterias.id', $usedCriteria)
97 ->groupBy('criterias.id')
98 ->orderBy('criterias.id','DESC')
99 $criterias->get();
100 $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
101 ->join('choice','choice.user_id','=','users.id')
102 ->join('criterias','criterias.id','=','choice.criteria_id')
103 ->where('choice.suggestion', '=', '1')
104 ->where('criterias.step', '=', '2')
105 ->where('criterias.status', '=', '1')
106 ->whereNotIn('criterias.id', $usedCriteria)
107 ->orderBy('criterias.id','ASC')
108 $criterias->get();
109 return view('criteria_step2.index', compact('data_standart', 'data_suggestion',
110 'data_fix', 'i', 'j', 'k'));
111 }
112 } else {
113 return redirect()->route('profile_users.show');
114 }
115 }
116
117 public function create()
118 {
119 return view('criteria_step2.create');
120 }
121
122 public function store(Request $request)
123 {
124 $this->validate($request, [
125 'name' => 'required',
126 'description' => 'required',
127 ]);
128 $input = $request->all();
129 $input["step"] = 2;
130 $input["status"] = 1;
131 $criteria = Criteria::create($input);
132 $user = User::find(Auth::user()->id);
133 $choice["user_id"] = $user->id;
134 $choice["criteria_id"] = $criteria->id;
135 $choice["option"] = 1;
136 $choice["suggestion"] = 0;

```

```

137 Choice::create($choice);
138 return redirect()->route('criteriastep2.index')
->with('success','Kriteria berhasil dibuat');
139 }
140 }
141 public function edit($id)
142 {
143 $criteria = Criteria::find($id);
144 return view('criteria_step2.edit',compact('criteria'));
145 }
146 }
147
148 public function update(Request $request, $id)
149 {
150 $this->validate($request, [
151 'name' => 'required',
152 'description' => 'required',
153 ]);
154 Criteria::find($id)->update($request->all());
155 return redirect()->route('criteriastep2.index')
->with('success','Kriteria berhasil diedit');
156 }
157 }
158
159 public function destroy($id)
160 {
161 $result = ResultSelection::where('criteria_id', '=', $id)->first();
162 $pairwise = PairwiseComparison::where('criteria1_id', '=', $id)->first();
163 if ($pairwise == null && $result == null) {
164 Choice::join('criterias','criterias.id','=','choice.criteria_id')
->where('choice.criteria_id', '=', $id)
->delete();
165 Criteria::find($id)->delete();
166 return redirect()->route('criteriastep2.index')
->with('success','Kriteria berhasil dihapus');
167 } else {
168 return redirect()->route('criteriastep2.index')
->with('failed','Kriteria tidak bisa dihapus karena sudah ada penilaian');
169 }
170 }
171 }
172
173
174
175
176 public function use(Request $request, $id)
177 {
178 $data = Choice::join('criterias','criterias.id','=','choice.criteria_id')
->where('choice.criteria_id', '=', $id)
->first();
179 $input['name'] = $data['name'];
180 $input['description'] = $data['description'];
181 $input["step"] = 2;
182 $input["status"] = 1;
183 $input["ref_id"] = $data["id"];
184 $criteria = Criteria::create($input);
185 $user = User::find(Auth::user()->id);
186 $choice["user_id"] = $user->id;
187 $choice["criteria_id"] = $criteria->id;
188 $choice["option"] = 1;
189 $choice["suggestion"] = 0;
190 Choice::create($choice);
191 return redirect()->route('criteriastep2.index')
->with('success','Kriteria berhasil digunakan');
192 }
193 }
194 }
```

Implementasi saat proses diskusi untuk menentukan kriteria yang digunakan dalam penilaian seleksi peserta pelatihan oleh para ahli dapat dilihat pada Gambar 5.7.



**Gambar 5.7 Implementasi saat proses diskusi menentukan kriteria**

##### 5. Implementasi proses Delphi putaran ketiga

Implementasi ini digunakan untuk mengkategorikan kriteria sehingga membentuk hierarki kriteria. Implementasi halaman hierarki kriteria dapat dilihat pada Gambar 5.8.

No	Kriteria/Sub-Kriteria	Aksi
1	Faktor intelektual <ul style="list-style-type: none"> <li>Pengalaman pelatihan</li> <li>Pengetahuan</li> <li>Keterampilan teknis</li> <li>Pendidikan terakhir</li> <li>Keterampilan komunikasi</li> </ul>	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">Remove from Group</a> <a href="#">Remove from Group</a> <a href="#">Remove from Group</a> <a href="#">Remove from Group</a> <a href="#">Remove from Group</a>
2	Faktor personal <ul style="list-style-type: none"> <li>Intensitas keikutsertaan pelatihan di BLK Bantul</li> </ul>	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">Remove from Group</a>

**Gambar 5.8 Implementasi halaman hierarki kriteria**

Potongan *source code* implementasi Delphi putaran kedua dapat dilihat pada Tabel 5.5. Baris 13-93 digunakan untuk menampilkan halaman hierarki kriteria. Baris 95-108 digunakan untuk menampilkan halaman tambah kelompok dan diproses dengan fungsi *store* pada baris 110-118. Baris 120-135 digunakan untuk menampilkan halaman edit kelompok dan diproses dengan fungsi *update*

pada baris 137-142. Baris 144-150 digunakan untuk menghapus kelompok. Baris 152-157 digunakan untuk menambah kriteria ke dalam kelompok, sehingga kriteria ini tingkatannya adalah sebagai subkriteria. Baris 159-164 digunakan untuk menghapus kriteria dari kelompok, sehingga kriteria ini tidak mempunyai kelompok dan tingkatannya adalah sebagai kriteria.

**Tabel 5.5 Potongan implementasi hierarki kriteria**

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\User;
6 use App\Http\Models\Choice;
7 use App\Http\Models\Criteria;
8 use Auth;
9 use DB;
10
11 class CriteriaGroupController extends Controller
12 {
13 public function index(Request $request)
14 {
15 $role_id = Auth::user()->roleId();
16 $user = User::find(Auth::user()->id);
17 $data = Choice::where('user_id', '=', $user->id)->first();
18 if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
19 if ($user->id == 1) {
20 $i = 0;
21 $j = 0;
22 $criterias_fix = Choice::select('choice.*', 'criterias.*')
23 ->join('criterias','criterias.id','=','choice.criteria_id')
24 ->where('choice.suggestion', '=', '0')
25 ->where('criterias.step', '=', '2')
26 ->where('criterias.status', '=', '1')
27 ->where('criterias.group_criteria', '=', null)
28 ->orderBy('criterias.id','DESC')
29 ->get();
30 $list_group = Criteria::where('group_criteria','=',null)
31 ->where('description','=',null)
32 ->lists('name','id')
33 ->all();
34 $criterias_group = array();
35 foreach ($list_group as $key=>$name){
36 $criterias_group[$key]["name"] = $name;
37 $criterias_group[$key]["data"] = array();
38 $criterias = Choice::select('choice.*', 'criterias.*')
39 ->join('criterias','criterias.id','=','choice.criteria_id')
40 ->where('choice.suggestion', '=', '0')
41 ->where('criterias.step', '=', '2')
42 ->where('criterias.status', '=', '1')
43 ->where('criterias.group_criteria', '=', $key)
44 ->orderBy('criterias.id','DESC')
45 ->get();
46 foreach ($criterias as $criteria){
```

```

47 $criterias_group[$key]["data"][] = $criteria;
48 }
49 }
50 return view('criteria_group.index',compact('criterias', 'criterias_fix', 'list_group',
51 'criterias_group', 'i', 'j'));
52 }
53 if ($data == null) {
54 return redirect()->route('questionnaire.create')
55 ->with('failed','Maaf, silahkan isi kuesioner kriteria dahulu.');
56 } else {
57 $i = 0;
58 $j = 0;
59 $criterias_fix = Choice::select('choice.*', 'criterias.*')
60 ->join('criterias','criterias.id','=',$choice.criteria_id')
61 ->where('choice.suggestion', '=', '0')
62 ->where('criterias.step', '=', '2')
63 ->where('criterias.status', '=', '1')
64 ->where('criterias.group_criteria', '=', null)
65 ->orderBy('criterias.id','DESC')
66 ->get();
67 $list_group = Criteria::where('group_criteria','=',null)
68 ->where('description','=',null)
69 ->lists('name','id')
70 ->all();
71 $criterias_group = array();
72 foreach ($list_group as $key=>$name){
73 $criterias_group[$key]["name"] = $name;
74 $criterias_group[$key]["data"] = array();
75 $criterias = Choice::select('choice.*', 'criterias.*')
76 ->join('criterias','criterias.id','=',$choice.criteria_id')
77 ->where('choice.suggestion', '=', '0')
78 ->where('criterias.step', '=', '2')
79 ->where('criterias.status', '=', '1')
80 ->where('criterias.group_criteria', '=', $key)
81 ->orderBy('criterias.id','DESC')
82 ->get();
83 foreach ($criterias as $criteria){
84 $criterias_group[$key]["data"][] = $criteria;
85 }
86 }
87 return view('criteria_group.index',compact('criterias', 'criterias_fix', 'list_group',
88 'criterias_group', 'i', 'j'));
89 }
90 } else {
91 return redirect()->route('profile_users.show');
92 }
93 }
94
95 public function create()
96 {
97 $user = User::find(Auth::user()->id);
98 $data = Choice::where('user_id', '=', $user->id)->first();
99 if ($user->id == 1) {
100 return view('criteria_group.create');
101 }
102 if ($data == null) {
103 return redirect()->route('questionnaire.create')
104 ->with('failed','Maaf, silahkan isi kuesioner kriteria dahulu.');
105 } else {
106 return view('criteria_group.create');
}

```

```

107 }
108 }
109
110 public function store(Request $request)
111 {
112 $input = $request->all();
113 $input['step'] = '2';
114 $input['status'] = '1';
115 Criteria::create($input);
116 return redirect()->route('criteriagroup.index')
117 ->with('success', 'Kelompok kriteria berhasil dibuat');
118 }
119
120 public function edit($id)
121 {
122 $user = User::find(Auth::user()->id);
123 $data = Choice::where('user_id', '=', $user->id)->first();
124 if ($user->id == 1) {
125 $criteria_group = Criteria::find($id);
126 return view('criteria_group.edit', compact('criteria_group'));
127 }
128 if ($data == null) {
129 return redirect()->route('questionnaire.create')
130 ->with('failed', 'Maaf, silahkan isi kuesioner krriteria dahulu.');
131 } else {
132 $criteria_group = Criteria::find($id);
133 return view('criteria_group.edit', compact('criteria_group'));
134 }
135 }
136
137 public function update(Request $request, $id)
138 {
139 Criteria::find($id)->update($request->all());
140 return redirect()->route('criteriagroup.index')
141 ->with('success', 'Kelompok kriteria berhasil diedit');
142 }
143
144 public function destroy($id)
145 {
146 Criteria::where('group_criteria', '=', $id)->update(['group_criteria' => null]);
147 Criteria::find($id)->delete();
148 return redirect()->route('criteriagroup.index')
149 ->with('success', 'Kelompok kriteria berhasil dihapus');
150 }
151
152 public function add(Request $request)
153 {
154 Criteria::find($request['id'])->update($request->all());
155 return redirect()->route('criteriagroup.index')
156 ->with('success', 'Kriteria berhasil dikelompokkan');
157 }
158
159 public function out(Request $request)
160 {
161 Criteria::find($request['id'])->update(['group_criteria' => null]);
162 return redirect()->route('criteriagroup.index')
163 ->with('success', 'Kriteria berhasil dikeluarkan dari kelompok');
164 }
165 }
```

## 5.2 Implementasi Metode AHP

Implementasi metode AHP digunakan untuk memperoleh bobot kriteria. Proses pembobotan dengan memasukkan nilai kepentingan antarkriteria. Implementasi halaman perbandingan berpasangan antarkriteria ditunjukkan pada Gambar 5.9.

No	Lebih penting A atau B?		Sama	Berapa banyak?
1	<input type="radio"/> Faktor eksternal	atau	<input checked="" type="radio"/> Faktor personal	<input type="radio"/> 1 <input type="radio"/> 2 <input checked="" type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input type="radio"/> 8 <input type="radio"/> 9
2	<input type="radio"/> Faktor eksternal	atau	<input checked="" type="radio"/> Faktor intelektual	<input type="radio"/> 1 <input checked="" type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input type="radio"/> 8 <input type="radio"/> 9
3	<input checked="" type="radio"/> Faktor personal	atau	<input type="radio"/> Faktor intelektual	<input type="radio"/> 1 <input checked="" type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input type="radio"/> 8 <input type="radio"/> 9

**Gambar 5.9 Implementasi halaman perbandingan berpasangan**

Setelah pembuat keputusan menyimpan nilai kepentingan antarkriteria, terjadi beberapa aksi dalam sistem yaitu melakukan normalisasi, menghitung bobot lokal dan CR. Perbandingan berpasangan tidak langsung tersimpan, sistem akan mengecek apakah nilai  $CR \leq 10\%$ . Jika nilai  $CR \leq 10\%$  maka perbandingan berpasangan konsisten dan berhasil disimpan. Namun jika nilai  $CR > 10\%$ , maka gagal disimpan dan harus melakukan perbandingan berpasangan ulang. Jika sudah berhasil disimpan, akan menuju ke halaman bobot untuk melihat bobot lokal dan bobot global. Bobot global akan dihitung dari hasil perkalian bobot lokal. Gambar 5.10 menunjukkan implementasi halaman bobot.

No	Kriteria/Sub-Kriteria	Pairwise Comparison	Bobot Parsial	Bobot Global
<b>Tujuan: SPK Seleksi Peserta Pelatihan</b>				
1	Faktor intelektual	AHP	0.297	
	• Pengalaman pelatihan		0.100	0.030
	• Pengetahuan		0.100	0.030
	• Keterampilan teknis		0.200	0.059
	• Pendidikan terakhir		0.400	0.119
	• Keterampilan komunikasi		0.200	0.059
2	Faktor personal	AHP	0.539	
	• Intensitas keikutsertaan pelatihan di BLK Bantul		0.053	0.029
	• Rencana setelah selesai pelatihan		0.068	0.037
	• Kejujuran		0.105	0.057
	• Sikap		0.053	0.029

**Gambar 5.10 Implementasi halaman bobot**

Potongan *source code* implementasi metode AHP dapat dilihat pada Tabel 5.6. Baris 12-57 digunakan untuk menampilkan halaman bobot. Baris 42-51 digunakan untuk menghitung bobot global. Baris 59-117 digunakan untuk menampilkan halaman perbandingan berpasangan. Baris 119-269 digunakan untuk memproses masukan perbandingan berpasangan. Baris 200 digunakan untuk menghitung bobot lokal. Baris 208 digunakan untuk menghitung CI. Baris 210-238 digunakan untuk menentukan nilai RI berdasarkan jumlah kriteria yang dibandingkan. Baris 240 digunakan untuk menghitung CR.

**Tabel 5.6 Potongan *source code* implementasi metode AHP**

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\Criteria;
6 use App\Http\Models\Choice;
7 use App\Http\Models\PairwiseComparison;
8 use Auth;
9
10 class WeightController extends Controller
11 {
12     public function index()
13     {
14         $role_id = Auth::user()->roleId();
15         if ($role_id == 3) {
16             $i = 0;
17         }
18     }
19 }
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
369
370
371
372
373
374
375
376
377
378
378
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
478
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
678
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
778
779
779
780
781
782
783
784
785
786
787
787
788
789
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
817
818
819
819
819
820
821
822
823
824
825
826
827
827
828
829
829
829
830
831
832
833
834
835
836
837
837
838
839
839
839
840
841
842
843
844
845
846
847
847
848
849
849
849
850
851
852
853
854
855
856
857
857
858
859
859
859
860
861
862
863
864
865
866
866
867
868
868
868
869
870
871
872
873
874
875
876
876
877
878
878
878
879
880
881
882
883
884
885
886
886
887
888
888
888
889
889
889
890
891
892
893
894
895
895
895
896
896
896
897
897
897
898
898
898
899
899
899
900
900
900
901
901
901
902
902
902
903
903
903
904
904
904
905
905
905
906
906
906
907
907
907
908
908
908
909
909
909
910
910
910
911
911
911
912
912
912
913
913
913
914
914
914
915
915
915
916
916
916
917
917
917
918
918
918
919
919
919
920
920
920
921
921
921
922
922
922
923
923
923
924
924
924
925
925
925
926
926
926
927
927
927
928
928
928
929
929
929
930
930
930
931
931
931
932
932
932
933
933
933
934
934
934
935
935
935
936
936
936
937
937
937
938
938
938
939
939
939
940
940
940
941
941
941
942
942
942
943
943
943
944
944
944
945
945
945
946
946
946
947
947
947
948
948
948
949
949
949
950
950
950
951
951
951
952
952
952
953
953
953
954
954
954
955
955
955
956
956
956
957
957
957
958
958
958
959
959
959
960
960
960
961
961
961
962
962
962
963
963
963
964
964
964
965
965
965
966
966
966
967
967
967
968
968
968
969
969
969
970
970
970
971
971
971
972
972
972
973
973
973
974
974
974
975
975
975
976
976
976
977
977
977
978
978
978
979
979
979
980
980
980
981
981
981
982
982
982
983
983
983
984
984
984
985
985
985
986
986
986
987
987
987
988
988
988
989
989
989
990
990
990
991
991
991
992
992
992
993
993
993
994
994
994
995
995
995
996
996
996
997
997
997
998
998
998
999
999
999
1000
1000
1000
1001
1001
1001
1002
1002
1002
1003
1003
1003
1004
1004
1004
1005
1005
1005
1006
1006
1006
1007
1007
1007
1008
1008
1008
1009
1009
1009
1010
1010
1010
1011
1011
1011
1012
1012
1012
1013
1013
1013
1014
1014
1014
1015
1015
1015
1016
1016
1016
1017
1017
1017
1018
1018
1018
1019
1019
1019
1020
1020
1020
1021
1021
1021
1022
1022
1022
1023
1023
1023
1024
1024
1024
1025
1025
1025
1026
1026
1026
1027
1027
1027
1028
1028
1028
1029
1029
1029
1030
1030
1030
1031
1031
1031
1032
1032
1032
1033
1033
1033
1034
1034
1034
1035
1035
1035
1036
1036
1036
1037
1037
1037
1038
1038
1038
1039
1039
1039
1040
1040
1040
1041
1041
1041
1042
1042
1042
1043
1043
1043
1044
1044
1044
1045
1045
1045
1046
1046
1046
1047
1047
1047
1048
1048
1048
1049
1049
1049
1050
1050
1050
1051
1051
1051
1052
1052
1052
1053
1053
1053
1054
1054
1054
1055
1055
1055
1056
1056
1056
1057
1057
1057
1058
1058
1058
1059
1059
1059
1060
1060
1060
1061
1061
1061
1062
1062
1062
1063
1063
1063
1064
1064
1064
1065
1065
1065
1066
1066
1066
1067
1067
1067
1068
1068
1068
1069
1069
1069
1070
1070
1070
1071
1071
1071
1072
1072
1072
1073
1073
1073
1074
1074
1074
1075
1075
1075
1076
1076
1076
1077
1077
1077
1078
1078
1078
1079
1079
1079
1080
1080
1080
1081
1081
1081
1082
1082
1082
1083
1083
1083
1084
1084
1084
1085
1085
1085
1086
1086
1086
1087
1087
1087
1088
1088
1088
1089
1089
1089
1090
1090
1090
1091
1091
1091
1092
1092
1092
1093
1093
1093
1094
1094
1094
1095
1095
1095
1096
1096
1096
1097
1097
1097
1098
1098
1098
1099
1099
1099
1100
1100
1100
1101
1101
1101
1102
1102
1102
1103
1103
1103
1104
1104
1104
1105
1105
1105
1106
1106
1106
1107
1107
1107
1108
1108
1108
1109
1109
1109
1110
1110
1110
1111
1111
1111
1112
1112
1112
1113
1113
1113
1114
1114
1114
1115
1115
1115
1116
1116
1116
1117
1117
1117
1118
1118
1118
1119
1119
1119
1120
1120
1120
1121
1121
1121
1122
1122
1122
1123
1123
1123
1124
1124
1124
1125
1125
1125
1126
1126
1126
1127
1127
1127
1128
1128
1128
1129
1129
1129
1130
1130
1130
1131
1131
1131
1132
1132
1132
1133
1133
1133
1134
1134
1134
1135
1135
1135
1136
1136
1136
1137
1137
1137
1138
1138
1138
1139
1139
1139
1140
1140
1140
1141
1141
1141
1142
1142
1142
1143
1143
1143
1144
1144
1144
1145
1145
1145
1146
1146
1146
1147
1147
1147
1148
1148
1148
1149
1149
1149
1150
1150
1150
1151
1151
1151
1152
1152
1152
1153
1153
1153
1154
1154
1154
1155
1155
1155
1156
1156
1156
1157
1157
1157
1158
1158
1158
1159
1159
1159
1160
1160
1160
1161
1161
1161
1162
1162
1162
1163
1163
1163
1164
1164
1164
1165
1165
1165
1166
1166
1166
1167
1167
1167
1168
1168
1168
1169
1169
1169
1170
1170
1170
1171
1171
1171
1172
1172
1172
1173
1173
1173
1174
1174
1174
1175
1175
1175
1176
1176
1176
1177
1177
1177
1178
1178
1178
1179
1179
1179
1180
1180
1180
1181
1181
1181
1182
1182
1182
1183
1183
1183
1184
1184
1184
1185
1185
1185
1186
1186
1186
1187
1187
1187
1188
1188
1188
1189
1189
1189
1190
1190
1190
1191
1191
1191
1192
1192
1192
1193
1193
1193
1194
1194
1194
1195
1195
1195
1196
1196
1196
1197
1197
1197
1198
1198
1198
1199
1199
1199
1200
1200
1200
1201
1201
1201
1202
1202
1202
1203
1203
1203
1204
1204
1204
1205
1205
1205
1206
1206
1206
1207
1207
1207
1208
1208
1208
1209
1209
1209
1210
1210
1210
1211
1211
1211
1212
1212
1212
1213
1213
1213
1214
1214
1214
1215
1215
1215
1216
1216
1216
1217
1217
1217
1218
1218
1218
1219
1219
1219
1220
1220
1220
1221
1221
1221
1222
1222
1222
1223
1223
1223
1224
1224
1224
1225
1225
1225
1226
1226
1226
1227
1227
1227
1228
1228
1228
1229
1229
1229
1230
1230
1230
1231
1231
1231
1232
1232
1232
1233
1233
1233
1234
1234
1234
1235
1235
1235
1236
1236
1236
1237
1237
1237
1238
1238
1238
1239
1239
1239
1240
1240
1240
1241
1241
1241
1242
1242
1242
1243
1243
1243
1244
1244
1244
1245
1245
1245
1246
1246
1246
1247
1247
1247
1248
1248
1248
1249
1249
1249
1250
1250
1250
1251
1251
1251
1252
1252
1252
1253
1253
1253
1254
1254
1254
1255
1255
1255
1256
1256
1256
1257
1257
1257
1258
1258
1258
1259
1259
1259
1260
1260
1260
1261
1261
1261
1262
1262
1262
1263
1263
1263
1264
1264
1264
1265
1265
1265
1266
1266
1266
1267
1267
1267
1268
1268
1268
1269
1269
1269
1270
1270
1270
1271
1271
1271
1272
1272
1272
1273
1273
1273
1274
1274
1274
1275
1275
1275
1276
1276
1276
1277
1277
1277
1278
1278
1278
1279
1279
1279
1280
1280
1280
1281
1281
1281
1282
1282
1282
1283
1283
1283
1284
1284
1284
1285
1285
1285
1286
1286
1286
1287
1287
1287
1288
1288
1288
1289
1289
1289
1290
1290
1290
1291
1291
1291
1292
1292
1292
1293
1293
1293
1294
1294
1294
1295
1295
1295
1296
1296
1296
1297
1297
1297
1298
1298
1298
1299
1299
1299
1300
1300
1300
1301
1301
1301
1302
1302
1302
1303
1303
1303
1304
1304
1304
1305
1305
1305
1306
1306
1306
1307
1307
1307
1308
1308
1308
1309
1309
1309
1310
1310
1310
131
```

```

18 $criterias = Criteria::where('step', '=', '2')
19 ->where('status', '=', '1')
20 ->where('group_criteria', '=', null)
21 ->whereNotIn('id', function($query){
22 $query->select('criteria_id')
23 ->from(with(new Choice)->getTable())
24 ->where('suggestion', 1);
25 })
26 ->get();
27 $criterias_group = array();
28 $total_criterias = 0;
29 foreach ($criterias as $criteria) {
30 $criterias_group[$criteria->id]["group"] = $criteria;
31 $criterias_group[$criteria->id]["data"] = array();
32 $subcriterias = Criteria::where('step', '=', '2')
33 ->where('status', '=', '1')
34 ->where('group_criteria', '=', $criteria->id)
35 ->whereNotIn('id', function($query){
36 $query->select('criteria_id')
37 ->from(with(new Choice)->getTable())
38 ->where('suggestion', 1);
39 })
40 ->orderBy('id','DESC')
41 ->get();
42 if (count($subcriterias) == 0) {
43 $criteria->global_weight = $criteria->partial_weight;
44 $criteria->save();
45 }
46 foreach ($subcriterias as $subcriteria) {
47 $subcriteria->global_weight = number_format($criteria->partial_weight * $subcriteria-
48 >partial_weight, 3);
49 $subcriteria->save();
50 $criterias_group[$criteria->id]["data"][] = $subcriteria;
51 }
52 }
53 return view('weights.index', compact('criterias_group', 'i'));
54 } else {
55 return redirect()->route('profile_users.show');
56 }
57 }

59 public function create($id)
60 {
61 $role_id = Auth::user()->roleId();
62 if ($role_id == 3) {
63 $i = 0;
64 $requestedId = null;
65 if ($id > 0) {
66 $requestedId = $id;
67 }
68 $criterias = Criteria::where('step', '=', '2')
69 ->where('status', '=', '1')
70 ->where('group_criteria', '=', $requestedId)
71 ->whereNotIn('id', function($query){
72 $query->select('criteria_id')
73 ->from(with(new Choice)->getTable())
74 ->where('suggestion', 1);
75 })
76 ->orderBy('id','DESC')
77 ->get();

```

```

78 $compares = array();
79 foreach ($criterias as $criteria1) {
80 foreach ($criterias as $criteria2) {
81 if (!array_key_exists($criteria1->id.":". $criteria2->id, $compares) &&
82 array_key_exists($criteria2->id.":". $criteria1->id, $compares) && $criteria1->id !=
83 $criteria2->id) {
84 $data = array(); $data[] = $criteria1;
85 $data[] = $criteria2;
86 $compares[$criteria1->id.":". $criteria2->id] = array();
87 $compares[$criteria1->id.":". $criteria2->id]['criteria'] = $data;
88 $pairwises = PairwiseComparison::where(function($q) use ($criteria1, $criteria2) {
89 $q->where(function($query) use ($criteria1, $criteria2) {
90 $query->where('criteria1_id', '=', $criteria1->id)
91 ->where('criteria2_id', $criteria2->id);
92 })
93 ->orWhere(function($query) use ($criteria1, $criteria2) {
94 $query->where('criteria1_id', $criteria2->id)
95 ->where('criteria2_id', $criteria1->id);
96 });
97 })
98 ->get();
99 $value = 0;
100 $selected_criteria = $criteria1->id;
101 foreach ($pairwises as $pairwise) {
102 if ($pairwise->value > $value) {
103 $value = $pairwise->value;
104 $selected_criteria = $pairwise->criteria1_id;
105 }
106 }
107 $compares[$criteria1->id.":". $criteria2->id]['value'] = $value;
108 $compares[$criteria1->id.":". $criteria2->id]['selected_criteria'] = $selected_criteria;
109 $compares[$criteria1->id.":". $criteria2->id]['pairwise'] = $pairwises;
110 }
111 }
112 }
113 return view('weights.pairwise', compact('compares', 'id', 'i'));
114 } else {
115 return redirect()->route('profile_users.show');
116 }
117 }

118
119 public function store(Request $request, $id)
120 {
121 $i = 0;
122 $requestedId = null;
123 if ($id > 0) {
124 $requestedId = $id;
125 }
126 $input = $request->all();
127 $criterias = Criteria::where('step', '=', '2')
128 ->where('status', '=', '1')
129 ->where('group_criteria', '=', $requestedId)
130 ->whereNotIn('id', function($query){
131 $query->select('criteria_id')
132 ->from(with(new Choice)->getTable())
133 ->where('suggestion', 1);
134 })
135 ->orderBy('id', 'DESC')
136 ->get();
137 $pairwises = array();

```

```

138 $count = 0;
139 foreach ($criterias as $criteria1) {
140 foreach ($criterias as $criteria2) {
141 $data = array();
142 $data["criteriad_1"] = $criteria1->id;
143 $data["criteriad_2"] = $criteria2->id;
144 if ($criteria1->id == $criteria2->id) {
145 $data["value"] = 1;
146 } elseif (array_key_exists($criteria1->id.":".$criteria2->id,$input["criteria_id"])) {
147 $selected_id = $input["criteria_id"][$criteria1->id.":".$criteria2->id];
148 if ($selected_id == $criteria1->id) {
149 $data["value"] = $input["value"][$criteria1->id.":".$criteria2->id];
150 } else {
151 $data["value"] = 1 / $input["value"][$criteria1->id.":".$criteria2->id];
152 }
153 } elseif (array_key_exists($criteria2->id.":".$criteria1->id,$input["criteria_id"])) {
154 $selected_id = $input["criteria_id"][$criteria2->id.":".$criteria1->id];
155 if ($selected_id == $criteria1->id) {
156 $data["value"] = $input["value"][$criteria2->id.":".$criteria1->id];
157 } else {
158 $data["value"] = 1 / $input["value"][$criteria2->id.":".$criteria1->id];
159 }
160 }
161 $pairwises[] = $data;
162 }
163 }
164 $base_table = array();
165 foreach ($pairwises as $pairwise){
166 if (!array_key_exists($pairwise["criteriad_1"],$base_table)) {
167 $base_table[$pairwise["criteriad_1"]] = array();
168 }
169 $base_table[$pairwise["criteriad_1"]][$pairwise["criteriad_2"]] = $pairwise["value"];
170 }
171 $normalized_table = array();
172 $total_basecol_val = array();
173 foreach ($base_table as $row=>$row_val) {
174 if (!array_key_exists($row,$normalized_table)) {
175 $normalized_table[$row] = array();
176 }
177 foreach ($row_val as $col=>$col_val) {
178 $total_row_val = 0;
179 $total_basecol_val[$col] = 0;
180 foreach ($base_table as $r=>$each_row) {
181 foreach ($each_row as $c=>$c_val) {
182 if ($c == $col) {
183 $total_row_val += $c_val;
184 $total_basecol_val[$col] += $c_val;
185 break;
186 }
187 }
188 }
189 $normalized_table[$row][$col] = $col_val / $total_row_val;
190 }
191 }
192 $partial_weight = array();
193 foreach ($normalized_table as $row=>$each_row) {
194 // hitung total jumlah di baris
195 $total_row_val = 0;
196 foreach ($each_row as $col=>$c_val){
197 $total_row_val += $c_val;

```

```

198 }
199 // hitung bobot partial
200 $partial_weight[$row] = $total_row_val / count($each_row);
201 }
202 $lamda_max = 0;
203 foreach ($partial_weight as $r=>$value) {
204 $criteria_factor = $value * $total_basecol_val[$r];
205 $lamda_max += $criteria_factor;
206 }
207 //hitung ci
208 $ci = ($lamda_max - count($partial_weight))/(count($partial_weight) - 1);
209 //random index
210 if (count($partial_weight) == 3) {
211 $ri = 0.58;
212 } elseif (count($partial_weight) == 4) {
213 $ri = 0.90;
214 } elseif (count($partial_weight) == 5) {
215 $ri = 1.12;
216 } elseif (count($partial_weight) == 6) {
217 $ri = 1.24;
218 } elseif (count($partial_weight) == 7) {
219 $ri = 1.32;
220 } elseif (count($partial_weight) == 8) {
221 $ri = 1.41;
222 } elseif (count($partial_weight) == 9) {
223 $ri = 1.45;
224 } elseif (count($partial_weight) == 10) {
225 $ri = 1.49;
226 } elseif (count($partial_weight) == 11) {
227 $ri = 1.51;
228 } elseif (count($partial_weight) == 12) {
229 $ri = 1.48;
230 } elseif (count($partial_weight) == 13) {
231 $ri = 1.56;
232 } elseif (count($partial_weight) == 14) {
233 $ri = 1.57;
234 } elseif (count($partial_weight) == 15) {
235 $ri = 1.59;
236 } else {
237 $ri = 0;
238 }
239 //hitung cr
240 $cr = number_format($ci / $ri * 100, 2);
241 if ($cr <= 10) {
242 foreach ($partial_weight as $key=>$val_partial_weight) {
243 $data_criteria["partial_weight"] = $val_partial_weight;
244 Criteria::find($key)->update($data_criteria);
245 }
246 foreach ($pairwises as $pairwise) {
247 $data_pairwise["criteriad1_id"] = $pairwise["criteriad1_id"];
248 $data_pairwise["criteriad2_id"] = $pairwise["criteriad2_id"];
249 $data_pairwise["value"] = $pairwise["value"];
250 $check_pairwise = PairwiseComparison::where('criteriad1_id', '=', 
251 $pairwise["criteriad1_id"])
252 ->where('criteriad2_id', '=', $pairwise["criteriad2_id"])
253 ->first();
254 if ($check_pairwise == null){
255 PairwiseComparison::create($data_pairwise);
256 } else {
257 PairwiseComparison::where('criteriad1_id', '=', $pairwise["criteriad1_id"])

```

```

258->where('criteria2_id', '=', $pairwise["criteriaid_2"])
259->update($data_pairwise);
260}
261}
262return redirect()->route('weights.index')
263->with('success','Penilaian bobot kriteria berhasil disimpan');
264} else {
265return redirect()->route('weights.pairwise', $id)
266->with('failed', 'Nilai Consistency Ratio (CR) = ' . $cr . '%. Nilai CR harus <= 10%.
267 Silahkan reevaluasi penilaian perbandingan berpasangan kriteria.');
268}
269}
270}

```

### 5.3 Implementasi Metode PROMETHEE

Implementasi metode PROMETHEE digunakan untuk memperoleh peringkat peserta pelatihan yang dapat digunakan pembuat keputusan sebagai pertimbangan menentukan peserta yang diterima. Berikut ini implementasinya:

1. Implementasi proses memasukkan nilai peserta seleksi pelatihan

Implementasi ini digunakan untuk memasukkan nilai peserta seleksi pelatihan berdasarkan kriteria dan parameter yang sudah ditentukan. Implementasi halaman pengelolaan nilai peserta seleksi pelatihan ditunjukkan pada Gambar 5.11. Referensi penilaian merupakan data peserta seleksi yang digunakan sebagai acuan penilaian, terdiri dari profil peserta, riwayat pelatihan di BLK Bantul yang pernah diikuti dan hasil nilai seleksi tertulis maupun wawancara.

Potongan *source code* implementasi tambah nilai peserta seleksi pelatihan dapat dilihat pada Tabel 5.7. Baris 1-70 digunakan untuk menampilkan halaman tambah nilai peserta seleksi pelatihan. Baris 72-108 digunakan untuk memproses masukan nilai dan menyimpannya ke dalam basis data. Baris 15-40 digunakan untuk menampilkan data peserta seleksi pelatihan yang digunakan sebagai referensi penilaian.

No	Kriteria/Sub-Kriteria	Nilai	Informasi Konversi
1	Intensitas keikutsertaan pelatihan di BLK Bantul	4	Belum pernah: 4 Pernah ikut 1 kali: 3 Pernah ikut 2 kali: 2 Pernah ikut >= 3 kali: 1
2	Pengalaman pelatihan	1	Pernah ikut pelatihan di luar BLK dan sesuai bidang yang diminati: 3 Pernah ikut pelatihan di luar BLK dan tidak sesuai bidang yang diminati: 2 Belum pernah: 1
3	Rencana setelah selesai pelatihan	3	Membuka usaha: 3 Melamar pekerjaan: 2 Menambah ilmu/keterampilan: 1 Tidak ada: 0
4	Rekomendasi	2	Ada: 2 Tidak: 1
5	Pengetahuan	70	
6	Keterampilan teknis	50	
7	Kejujuran	0	Sesuai: 1 Tidak Sesuai: 0
21	Kesan baik	2	Baik: 3 Cukup: 2 Kurang: 1

**Submit**

**Referensi Penilaian**

No. Identitas	[REDACTED]
Nama Pendaftar	Rahmat [REDACTED]
email	[REDACTED]
Alamat	[REDACTED]
No. Telepon/HP	[REDACTED]
Jenis Kelamin	Laki-laki
Tempat, Tanggal Lahir	Bantul, 2000-03-[REDACTED]
Usia	18
Anak ke-	2
Jumlah Saudara	3
Agama	Islam
Nama Ibu Kandung	[REDACTED]
Nama Ayah	[REDACTED]
Alamat Orangtua	[REDACTED]

**Gambar 5.11 Implementasi halaman tambah nilai peserta seleksi pelatihan**

**Tabel 5.7 Potongan *source code* implementasi tambah nilai peserta seleksi pelatihan**

```

1 public function assessment($id)
2 {
3     $role_id = Auth::user()->roleId();
4     if ($role_id == 3) {
5         $i = 0;
6         $j = 0;
7         $data = Selection::select('selections.id AS ID_SELECTION', 'users.id AS ID_USER',
8             'registrants.id AS ID_REGISTRANT',
9             'users.*', 'registrants.*', 'selections.*')
10        ->join('registrants', 'registrants.id', '=', 'selections.registration_id')
11        ->join('users', 'users.id', '=', 'registrants.user_id')
12        ->where('selections.id', '=', $id)
13        ->first();
14         $date_birth = Carbon\Carbon::parse($data->date_birth);
15         $age = Carbon\Carbon::createFromDate($date_birth->year, $date_birth->month, $date_birth->day)->age;
16
17

```

```

18 $check_result = ResultSelection::with('selection')
19 ->where('selection_id', '=', $data->ID_SELECTION)
20 ->first();
21 $educational_background = EducationalBackground::with('education')
22 ->where('registrant_id', '=', $data->ID_REGISTRANT)
23 ->orderBy('education_id','DESC')
24 ->get();
25 $course_experience = CourseExperience::with('course')
26 ->where('registrant_id', '=', $data->ID_REGISTRANT)
27 ->get();
28 $upload = Upload::where('registrant_id', '=', $data->ID_REGISTRANT)->first();
29 $registration = Registration::select('registrations.id AS ID_REGISTRATION', 'selections.id
30 AS ID_SELECTION',
31 'selection_schedules.id AS ID_SSCHEDULE', 'sub_vocationals.id AS ID_SUBVOC',
32 'registrations.*', 'selections.*', 'selection_schedules.*', 'sub_vocationals.*')
33 ->join('selections', 'selections.registration_id', '=', 'registrations.id')
34 ->join('selection_schedules', 'selection_schedules.id', '=',
35 'selections.selection_schedule_id')
36 ->join('sub_vocationals', 'sub_vocationals.id', '=',
37 'selection_schedules.sub_vocational_id')
38 ->where('registrant_id', '=', $data->ID_REGISTRANT)
39 ->orderBy('ID_SELECTION','DESC')
40 ->get();
41 $criterias = Criteria::where('step', '=', '2')
42 ->where('status', '=', '1')
43 ->where('description', '<>', null)
44 ->whereNotIn('id', function($query){
45 $query->select('criteria_id')
46 ->from(with(new Choice)->getTable())
47 ->where('suggestion', 1);
48 })
49 ->orderBy('id','DESC')->get();
50 $return_data = array();
51 foreach ($criterias as $criteria) {
52 $single_data = array();
53 $single_data["criteria"] = $criteria;
54 $result_selection = ResultSelection::where('selection_id', '=', $data->ID_SELECTION)
55 ->where('criteria_id', '=', $criteria->id)
56 ->first();
57 if ($result_selection == null) {
58 $single_data["value"] = null;
59 } else {
60 $single_data["value"] = $result_selection;
61 }
62 $return_data[] = $single_data;
63 }
64 return view('result_selection.assessment',compact('data', 'age',
65 'check_result','educational_background','course_experience','upload','registration','retur
66 n_data','i','j'));
67 } else {
68 return redirect()->route('profile_users.show');
69 }
70 }
71
72 public function store(Request $request, $id)
73 {
74 $input = $request->all();
75 $criterias = Criteria::where('step', '=', '2')
76 ->where('status', '=', '1')
77 ->where('description', '<>', null)

```

```

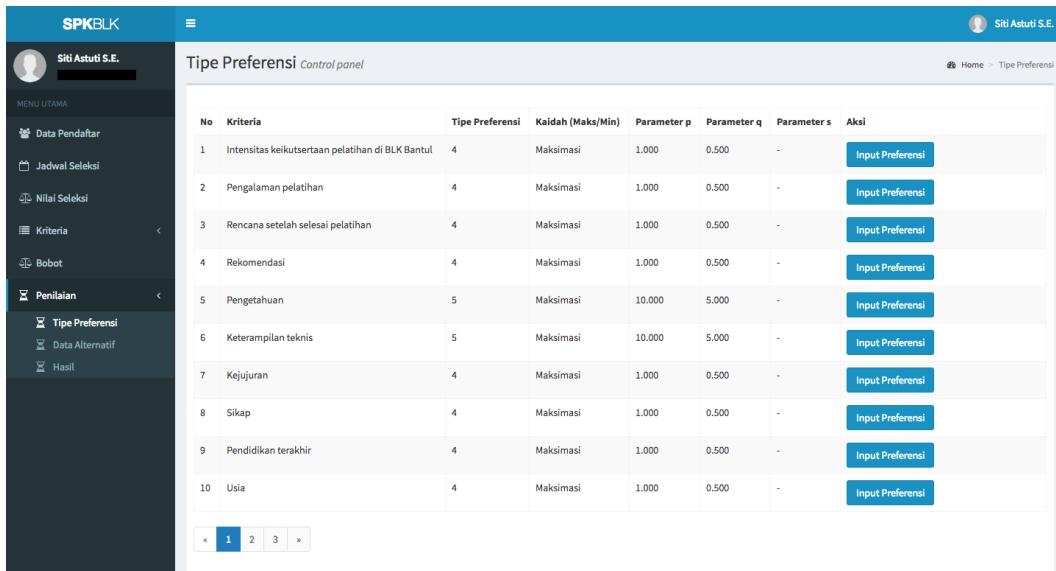
78 ->whereNotIn('id', function($query){
79 $query->select('criteria_id')
80 ->from(with(new Choice)->getTable())
81 ->where('suggestion', 1);
82 })
83 ->orderBy('id','DESC')
84 ->lists('id');
85 foreach ($criterias as $criteria) {
86 $result_selection = ResultSelection::where('selection_id', '=', $id)
87 ->where('criteria_id', '=', $criteria)
88 ->first();
89 if ($input[$criteria] != "") {
90 $data["selection_id"] = $id;
91 $data["criteria_id"] = $criteria;
92 $data["value"] = $input[$criteria];
93 if ($result_selection == null) {
94 ResultSelection::create($data);
95 } else {
96 ResultSelection::where('selection_id', '=', $id)
97 ->where('criteria_id', '=', $criteria)
98 ->update($data);
99 }
100 } else {
101 ResultSelection::where('selection_id', '=', $id)
102 ->where('criteria_id', '=', $criteria)
103 ->delete();
104 }
105 }
106 return redirect()->route('result_selection.index')
107 ->with('success','Penilaian berhasil disimpan');
108 }

```

## 2. Implementasi proses menentukan fungsi preferensi

Implementasi ini digunakan untuk menentukan fungsi preferensi setiap kriteria yang terdiri dari tipe preferensi, *Min/Max* dan parameternya. Setiap tipe preferensi mempunyai ketentuan parameter yang berbeda seperti yang dijelaskan pada Tabel 4.25. Nilai *Min/Max* menunjukkan kecenderungan data, dimana jika kecenderungannya *Min* maka menunjukkan nilai yang semakin kecil adalah semakin baik, sedangkan jika *Max* maka nilainya semakin baik jika nilainya semakin tinggi. Implementasi halaman pengelolaan fungsi preferensi dapat dilihat pada Gambar 5.12.

Potongan *source code* implementasi pengelolaan fungsi preferensi dapat dilihat pada Tabel 5.8. Baris 11-30 digunakan untuk menampilkan halaman daftar fungsi preferensi dari keseluruhan kriteria. Baris 32-41 menampilkan halaman untuk memasukkan fungsi preferensi dan masukan akan diproses dengan fungsi *update* pada baris 43-69.



Gambar 5.12 Implementasi halaman pengelolaan fungsi preferensi

Tabel 5.8 Impelemensi pengelolaan fungsi preferensi

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\Criteria;
6 use App\Http\Models\Choice;
7 use Auth;
8
9 class PreferenceController extends Controller
{
10 public function index(Request $request)
{
11 $role_id = Auth::user()->roleId();
12 if ($role_id == 3) {
13 $preferences = Criteria::where('step', '=', '2')
14 ->where('status', '=', '1')
15 ->where('description', '<>', null)
16 ->whereNotIn('id', function($query){
17 $query->select('criteria_id')
18 ->from(with(new Choice)->getTable())
19 ->where('suggestion', 1);
20 })
21 ->orderBy('id', 'DESC')
22 ->paginate(10);
23 return view('preferences.index', compact('preferences'))
24 ->with('i', ($request->input('page', 1) - 1) * 10);
25 } else {
26 return redirect()->route('profile_users.show');
27 }
28 }
29 }
30
31 public function edit($id)
32 {
33

```

```

34 $role_id = Auth::user()->roleId();
35 if ($role_id == 3) {
36 $preference = Criteria::find($id);
37 return view('preferences.edit',compact('preference'));
38 } else {
39 return redirect()->route('profile_users.show');
40 }
41 }
42
43 public function update(Request $request, $id)
44 {
45 $this->validate($request, [
46 'preference' => 'required',
47 'max_min' => 'required',
48 ]);
49 $input = $request->all();
50 if ($input['preference'] == 1) {
51 $input['parameter_p'] = null;
52 $input['parameter_q'] = null;
53 $input['parameter_s'] = null;
54 } elseif ($input['preference'] == 2) {
55 $input['parameter_p'] = null;
56 $input['parameter_s'] = null;
57 } elseif ($input['preference'] == 3) {
58 $input['parameter_q'] = null;
59 $input['parameter_s'] = null;
60 } elseif ($input['preference'] == 4 || $input['preference'] == 5) {
61 $input['parameter_s'] = null;
62 } else {
63 $input['parameter_p'] = null;
64 $input['parameter_q'] = null;
65 }
66 Criteria::find($id)->update($input);
67 return redirect()->route('preferences.index')
68 ->with('success','Tipe preferensi berhasil disimpan');
69 }
70 }

```

### 3. Implementasi proses penilaian dengan metode PROMETHEE

Implementasi ini digunakan untuk memperoleh peringkat berdasarkan hasil dari penilaian dengan metode PROMETHEE. Implementasi halaman penilaian peserta seleksi pelatihan ditunjukkan pada Gambar 5.13. Untuk memulai proses penilaian dilakukan dengan aksi Mulai Hitung Penilaian. Sebelum diproses, sistem melakukan pengecekan apakah sudah semua data nilai peserta seleksi pelatihan lengkap. Jika sudah lengkap, maka sistem akan memproses perhitungan penilaian. Namun jika ada data nilai yang belum lengkap, maka harus melengkapi terlebih dahulu nilai yang masih kosong tersebut.

No	Nama Pendaftar	Sub-Kejuruan	Tanggal Seleksi	Waktu Seleksi	Aksi
1	Safitri Yahya	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
2	Ridwan Nur Y	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
3	Muklas Abror Aulawi	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
4	Muh Ambar Wijayanto	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
5	Maendra Yoga Pratama	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
6	Joko Tri Suwahlan	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
7	Indra Godama	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
8	Hendra Prasetya	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
9	Endro Widayat	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>
10	Berly Gunawan	Teknik Sepeda Motor	2016-04-25	08:00	<button>Penilaian</button>

« 1 2 3 4 »

[Mulai Hitung Penilaian](#)

**Gambar 5.13 Implementasi halaman penilaian peserta seleksi pelatihan**

Potongan *source code* implementasi penilaian peserta seleksi pelatihan ditunjukkan pada Tabel 5.9. Baris 19-44 digunakan untuk menampilkan daftar peserta seleksi pelatihan dan aksi untuk menghitung penilaian. Baris 46-280 digunakan untuk proses menghitung penilaian dengan metode PROMETHEE. Perhitungan penilaian akan menghasilkan peringkat peserta seleksi pelatihan. Baris 98-108 digunakan untuk menghitung selisih nilai antarpeserta untuk setiap kriteria. Baris 109-181 digunakan untuk menghitung derajat preferensi berdasarkan fungsi preferensi yang sudah ditentukan. Baris 182-196 digunakan untuk menghitung indeks preferensi. Baris 197-211 digunakan untuk menghitung *leaving flow* dan *entering flow*. Baris 212-249 digunakan untuk mengetahui hubungan antarpeserta apakah semua *comparable* atau ada yang *incomparable*. Jika semua *comparable*, maka dapat diperoleh peringkat parsial dengan baris 252-261. Namun jika ada yang *incomparable*, maka perlu menghitung *net flow* yaitu dengan baris 263-266. Berdasarkan hasil *net flow* dapat diperoleh peringkat lengkap dengan baris 266-274.

**Tabel 5.9 Potongan *source code* implementasi penilaian peserta seleksi pelatihan**

<pre> 1 &lt;?php 2 namespace App\Http\Controllers; 3 use Illuminate\Http\Request; 4 use App\Http\Controllers\Controller; 5 use App\Http\Models\ResultSelection; 6 use App\Http\Models\Selection; 7 use App\Http\Models\Criteria; 8 use App\Http\Models\Choice; 9 use App\Http\Models\EducationalBackground; 10 use App\Http\Models\CourseExperience; 11 use App\Http\Models\Upload; 12 use App\Http\Models\Registration; 13 use App\Http\Models\Subvocational; 14 use Auth; 15 use Carbon; 16 17 class ResultSelectionController extends Controller 18 { 19     public function index(Request \$request) 20     { 21         \$role_id = Auth::user()-&gt;roleId(); 22         if (\$role_id == 3) { 23             \$data = Selection::select('selections.*', 'users.name AS name_registrant', 24                                     'selection_schedules.date', 25                                     'selection_schedules.time', 'sub_vocationals.name AS name_sub_vocational') 26             -&gt;join('registrations', 'registrations.id', '=', 'selections.registration_id') 27             -&gt;join('registrants', 'registrants.id', '=', 'selections.registrant_id') 28             -&gt;join('users', 'users.id', '=', 'selections.user_id') 29             -&gt;join('selection_schedules', 'selection_schedules.id', '=', 30                     'selections.selection_schedule_id') 31             -&gt;join('sub_vocationals', 'sub_vocationals.id', '=', 32                     'selection_schedules.sub_vocational_id') 33             -&gt;where(function(\$query){ 34                 \$query-&gt;whereNull('selections.status') 35                 -&gt;orWhere('selections.status', '=', ''); 36             }) 37             -&gt;orderBy('selections.id', 'DESC') 38             -&gt;paginate(10); 39             return view('result_selection.index', compact('data', 'result_selection')) 40             -&gt;with('i', (\$request-&gt;input('page', 1) - 1) * 10); 41         } else { 42             return redirect()-&gt;route('profile_users.show'); 43         } 44     } 45 46     public function count() 47     { 48         \$selections = Selection::select('selections.*', 'users.name AS name_registrant', 49                                     'selection_schedules.date', 50                                     'selection_schedules.time', 'sub_vocationals.name AS name_sub_vocational') 51             -&gt;join('registrations', 'registrations.id', '=', 'selections.registration_id') 52             -&gt;join('registrants', 'registrants.id', '=', 'selections.registrant_id') 53             -&gt;join('users', 'users.id', '=', 'selections.user_id') 54             -&gt;join('selection_schedules', 'selection_schedules.id', '=', 55                     'selections.selection_schedule_id') 56             -&gt;join('sub_vocationals', 'sub_vocationals.id', '=', </pre>
---

```

57 'selection_schedules.sub_vocational_id')
58 ->where(function($query){
59 $query->whereNull('selections.status')
60 ->orWhere('selections.status', '=', '');
61 })
62 ->orderBy('selections.id', 'DESC')
63 ->get();
64 $selectionsId = array();
65 foreach ($selections as $selection) {
66 $selectionsId[] = $selection->id;
67 $selectionsData[$selection->id] = $selection;
68 }
69 $tabel_alternative = array();
70 $criterias = Criteria::where('step', '=', '2')
71 ->where('status', '=', '1')
72 ->where('description', '!=', null)
73 ->whereNotIn('id', function($query){
74 $query->select('criteria_id')
75 ->from(with(new Choice)->getTable())
76 ->where('suggestion', 1);
77 })
78 ->orderBy('id', 'DESC')
79 ->get();
80 $criteriasData = array();
81 foreach ($criterias as $criteria) {
82 $criteriasData[$criteria->id] = $criteria;
83 }
84 foreach ($selections as $selection) {
85 $tabel_alternative[$selection->id] = array();
86 foreach ($criterias as $criteria) {
87 $result_selection = ResultSelection::where('selection_id', '=', $selection->id)
88 ->where('criteria_id', '=', $criteria->id)
89 ->first();
90 if ($result_selection == null) {
91 return redirect()->route('result_selection.index')
92 ->with('failed', 'Hitung penilaian GAGAL! ' . $selection->name_registrant . ' belum dinilai.
93 Silahkan lakukan penilaian');
94 }
95 $tabel_alternative[$selection->id][$criteria->id] = $result_selection->value;
96 }
97 }
98 $tabel_selisih = array();
99 foreach ($tabel_alternative as $key1=>$data_selisih1) {
100 foreach ($tabel_alternative as $key2=>$data_selisih2) {
101 if ($key1 != $key2) {
102 $tabel_selisih[$key1.".". $key2] = array();
103 foreach ($data_selisih1 as $criteria=>$value) {
104 $tabel_selisih[$key1.".". $key2][$criteria] = $value - $data_selisih2[$criteria];
105 }
106 }
107 }
108 }
109 $tabel_derajat = array();
110 foreach ($tabel_selisih as $alternatives=>$crt_data) {
111 $tabel_derajat[$alternatives] = array();
112 foreach ($crt_data as $criteriaId=>$value) {
113 $criteria = $criteriasData[$criteriaId];
114 $type = $criteria->preference;
115 switch ($type) {
116 case "1":

```

```

117 if ($value <= 0) {
118 $tabel_derajat[$alternatives][$criteriaId] = 0;
119 } else {
120 $tabel_derajat[$alternatives][$criteriaId] = 1;
121 }
122 break;
123 case "2":
124 $q = $criteria->parameter_q;
125 if ($value <= $q) {
126 $tabel_derajat[$alternatives][$criteriaId] = 0;
127 } else {
128 $tabel_derajat[$alternatives][$criteriaId] = 1;
129 }
130 break;
131 case "3":
132 $p = $criteria->parameter_p;
133 if ($value <= 0) {
134 $tabel_derajat[$alternatives][$criteriaId] = 0;
135 } else if($value > $p) {
136 $tabel_derajat[$alternatives][$criteriaId] = 1;
137 } else {
138 $tabel_derajat[$alternatives][$criteriaId] = $value / $p;
139 }
140 break;
141 case "4":
142 $p = $criteria->parameter_p;
143 $q = $criteria->parameter_q;
144 if ($value <= $q) {
145 $tabel_derajat[$alternatives][$criteriaId] = 0;
146 } else if ($value > $p) {
147 $tabel_derajat[$alternatives][$criteriaId] = 1;
148 } else {
149 $tabel_derajat[$alternatives][$criteriaId] = 0.5;
150 }
151 break;
152 case "5":
153 $p = $criteria->parameter_p;
154 $q = $criteria->parameter_q;
155 if ($value <= $q) {
156 $tabel_derajat[$alternatives][$criteriaId] = 0;
157 } else if ($value > $p) {
158 $tabel_derajat[$alternatives][$criteriaId] = 1;
159 } else {
160 $tabel_derajat[$alternatives][$criteriaId] = ($value - $q)/($p - $q);
161 }
162 break;
163 case "6":
164 $e = 2.71828;
165 $s = $criteria->parameter_s;
166 if ($value <= 0) {
167 $tabel_derajat[$alternatives][$criteriaId] = 0;
168 } else {
169 $pow_d = pow($value, 2);
170 $pow_s = pow($s, 2);
171 $pow_val = -($pow_d / (2 * $pow_s));
172 $pow_e = pow($e, $pow_val);
173 $tabel_derajat[$alternatives][$criteriaId] = 1 - $pow_e;
174 }
175 break;
176 default:

```

```

177 $tabel_derajat[$alternatives][$criteriaId] = "-";
178 break;
179 }
180 }
181 }
182 $tabel_index = array();
183 foreach ($tabel_derajat as $alternatives=>$data_index) {
184 $alternativesId = explode(",",$alternatives);
185 $id1 = $alternativesId[0];
186 $id2 = $alternativesId[1];
187 $tabel_index[$id1][$id2] = 0;
188 $tabel_index[$id1][$id1] = 0;
189 $tabel_index[$id2][$id2] = 0;
190 foreach ($data_index as $criteriaId=>$value) {
191 $criteria = $criteriasData[$criteriaId];
192 $bobot = $criteria->global_weight;
193 $mlt = $bobot * $value;
194 $tabel_index[$id1][$id2] += number_format($mlt,5);
195 }
196 }
197 $tabel_leaving = array();
198 $tabel_entering = array();
199 $n = count($selectionsId);
200 foreach($selectionsId as $selectionId1) {
201 $sum_row = 0;
202 $sum_col = 0;
203 foreach($tabel_index[$selectionId1] as $value) {
204 $sum_row += $value;
205 }
206 foreach($selectionsId as $selectionId2) {
207 $sum_col += $tabel_index[$selectionId2][$selectionId1];
208 }
209 $tabel_leaving[$selectionId1] = number_format((1 / ($n-1)) * $sum_row, 5);
210 $tabel_entering[$selectionId1] = number_format((1 / ($n-1)) * $sum_col, 5);
211 }
212 $isComparable = true;
213 $condition =array();
214 foreach($selectionsId as $selectionIdA) {
215 foreach($selectionsId as $selectionIdB) {
216 if ($selectionIdA < $selectionIdB) {
217 $condition[$selectionIdA.".".$selectionIdB] = array();
218 $condition[$selectionIdA.".".$selectionIdB]["A_Ii_B"] = false;
219 $condition[$selectionIdA.".".$selectionIdB]["A_S+ B"] = false;
220 $condition[$selectionIdA.".".$selectionIdB]["A_S- B"] = false;
221 $condition[$selectionIdA.".".$selectionIdB]["A_Pi_B"] = false;
222 $condition[$selectionIdA.".".$selectionIdB]["A_R_B"] = true;
223 $condition1 = ($tabel_leaving[$selectionIdA] == $tabel_leaving[$selectionIdB]) &&
224 ($tabel_entering[$selectionIdA] == $tabel_entering[$selectionIdB]);
225 $condition2a = ($tabel_leaving[$selectionIdA] > $tabel_leaving[$selectionIdB]) ||
226 ($tabel_leaving[$selectionIdA] == $tabel_leaving[$selectionIdB]);
227 $condition2b = ($tabel_entering[$selectionIdA] < $tabel_entering[$selectionIdB]) ||
228 ($tabel_entering[$selectionIdA] == $tabel_entering[$selectionIdB]);
229 if ($condition1) {
230 $condition[$selectionIdA.".".$selectionIdB]["A_Ii_B"] = true;
231 }
232 if ($condition2a) {
233 $condition[$selectionIdA.".".$selectionIdB]["A_S+ B"] = true;
234 }
235 if ($condition2b) {
236 $condition[$selectionIdA.".".$selectionIdB]["A_S- B"] = true;

```

```

237 }
238 if ($condition2a == $condition2b) {
239 $condition[$selectionIdA.".".$selectionIdB]["A Pi B"] = true;
240 }
241 if ($condition[$selectionIdA.".".$selectionIdB]["A Pi B"] || $condition[$selectionIdA."."
242 $selectionIdB]["A Ii B"]) {
243 $condition[$selectionIdA.".".$selectionIdB]["A R B"] = false;
244 } else {
245 $isComparable = false;
246 }
247 }
248 }
249 }
250 $sortedSelection = array();
251 $rank = 1;
252 if ($isComparable) {
253 arsort($tabel_leaving);
254 foreach ($tabel_leaving as $key=>$value) {
255 $sortedSelection[] = $key;
256 $selection = $selectionsData[$key];
257 $selection->ranking = $rank;
258 $selection->status = "Selesai";
259 $selection->save();
260 $rank++;
261 }
262 } else {
263 $netflow = array();
264 foreach ($tabel_leaving as $key=>$value) {
265 $netflow[$key] = number_format($value - $tabel_entering[$key], 5);
266 }
267 arsort($netflow);
268 foreach ($netflow as $key=>$value) {
269 $sortedSelection[] = $key;
270 $selection = $selectionsData[$key];
271 $selection->ranking = $rank;
272 $selection->status = "Selesai";
273 $selection->save();
274 $rank++;
275 }
276 }
277 return redirect()->route('result_selection.index')
278 ->with('success','Hitung penilaian berhasil. Lihat hasil di menu "Hasil"');
279 }
280 }

```

#### 4. Implementasi proses menampilkan hasil peringkat peserta seleksi

Implementasi ini digunakan untuk menampilkan hasil peringkat berdasarkan proses perhitungan penilaian dengan PROMETHEE. Implementasi halaman hasil peringkat peserta seleksi pelatihan ditunjukkan pada Gambar 5.14. Potongan *source code* implementasi hasil peringkat peserta seleksi dapat dilihat pada Tabel 5.10.

No	No. Identitas	Nama Pendaftar	Sub-Kejuruan	Tanggal dan Waktu Seleksi	Ranking
1	610901200897003	Agus Widodo	Teknik Sepeda Motor	2016-04-25 dan 08:00	1
2	1808120912960003	Trio Adi Saputra	Teknik Sepeda Motor	2016-04-25 dan 08:00	2
3	3402031003980003	Anang Ma'ruf Martanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	3
4	3402102904950001	Edi Sudarman	Teknik Sepeda Motor	2016-04-25 dan 08:00	4
5	3402051601970002	Moko Ginto	Teknik Sepeda Motor	2016-04-25 dan 08:00	5
6	3402041811910002	Susanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	6
7	3402050411960002	Ridwan Susanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	7
8	3402101005990002	Zupriyanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	8
9	3402021604990001	Mustakim	Teknik Sepeda Motor	2016-04-25 dan 08:00	9
10	3402052005990001	Bimas Wasono	Teknik Sepeda Motor	2016-04-25 dan 08:00	10
11	340280408960002	Agus Hidayanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	11
12	3402081509970001	Ryan Aji Pamungkas	Teknik Sepeda Motor	2016-04-25 dan 08:00	12
13	10000007	Eko Samsudin	Teknik Sepeda Motor	2016-04-25 dan 08:00	13
14	3402052903010001	Rahmat Nur Islam	Teknik Sepeda Motor	2016-04-25 dan 08:00	14
15	34021102808940001	Rigunawan	Teknik Sepeda Motor	2016-04-25 dan 08:00	15
16	100000011	Rajif Waluyo	Teknik Sepeda Motor	2016-04-25 dan 08:00	16

Gambar 5.14 Implementasi halaman hasil peringkat peserta seleksi

Tabel 5.10 Potongan *source code* implementasi hasil peringkat peserta seleksi

```

1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Http\Models\User;
6 use App\Http\Models\Selection;
7 use Auth;
8
9 class ResultController extends Controller
{
10 public function index()
11 {
12 $role_id = Auth::user()->roleId();
13 if ($role_id == 1 || $role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
14 $i = 0;
15 $result = Selection::select('selections.*', 'users.identity_number', 'users.name AS
16 name_registrant', 'selection_schedules.date',
17 'selection_schedules.time', 'sub_vocationals.name AS name_sub_vocational')
18 ->join('registrations', 'registrations.id', '=', 'selections.registration_id')
19 ->join('registrants', 'registrants.id', '=', 'selections.selection_id')
20 ->join('users', 'users.id', '=', 'selections.user_id')
21 ->join('selection_schedules', 'selection_schedules.id', '=',
22 'selections.selection_schedule_id')
23 ->join('sub_vocationals', 'sub_vocationals.id', '=',
24 'selection_schedules.sub_vocational_id')
25 ->where('selections.status', '=', 'Selesai')
26 ->orderBy('selections.rank', 'ASC')
27 ->get();
28 return view('result.index', compact('result', 'i'));
29 } else {
30 return redirect()->route('profile_users.show');
31 }
32 }
33 }
34 }
```

## BAB VI

### HASIL PENELITIAN DAN PEMBAHASAN

#### **6.1 Perbandingan Metode AHP-PROMETHEE Dan *Modified* Delphi-AHP-PROMETHEE**

Pengujian ini dilakukan untuk mengetahui tingkat kesesuaian penggunaan metode *Modified* Delphi, yaitu dengan membandingkan hasil AHP-PROMETHEE dan hasil *Modified* Delphi-AHP-PROMETHEE dengan data masa lalu (belum menggunakan metode MADM). Dalam kasus ini belum ada ketetapan kriteria/subkriteria yang digunakan, sehingga untuk implementasi metode AHP-PROMETHEE menggunakan kriteria/subkriteria yang ditentukan oleh pembuat keputusan. Pembuat keputusan menentukan kriteria/subkriteria sesuai dengan gagasan dan kompetensinya dalam seleksi peserta pelatihan. Pembuat keputusan merupakan plt. kepala BLK yang sudah bekerja di BLK selama 23 tahun. Peneliti memberikan lembar kertas kepada pembuat keputusan untuk menuliskan kriteria/subkriteria apa saja yang digunakan dalam seleksi peserta pelatihan. Proses ini dilakukan sebelum proses *Modified* Delphi, sehingga gagasan merupakan gagasan murni pembuat keputusan, tanpa adanya interfensi dari peneliti, pustaka atau pihak lain. Perbandingan kriteria/subkriteria yang ditentukan melalui proses *Modified* Delphi dan pembuat keputusan ditunjukkan pada Tabel 6.1.

**Tabel 6.1 Perbandingan kriteria/subkriteria berdasarkan metode *Modified* Delphi dan pembuat keputusan**

<b>Pembuat keputusan</b>		<b><i>Modified</i> Delphi</b>	
<b>Kriteria</b>	<b>Subkriteria</b>	<b>Kriteria</b>	<b>Subkriteria</b>
Umur	-	Faktor intelektual	Pengalaman pelatihan
Pendidikan terakhir	-		Pengetahuan

**Tabel 6.1 (lanjutan)**

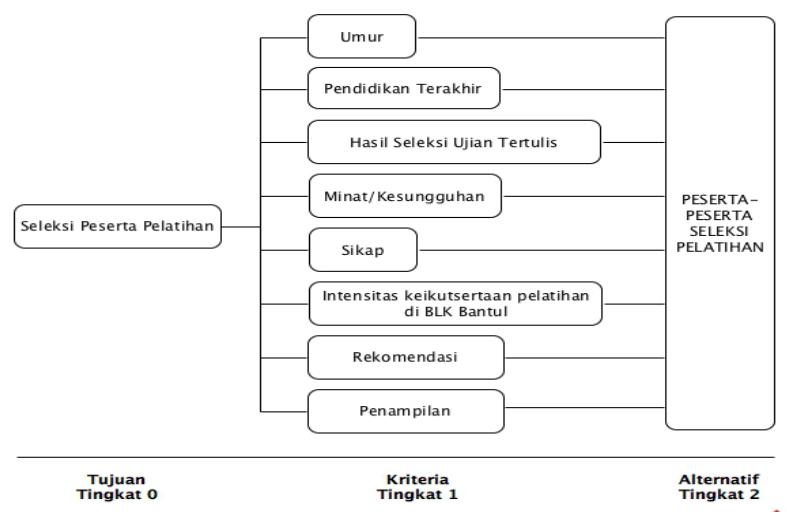
<b>Pembuat keputusan</b>		<b><i>Modified Delphi</i></b>	
<b>Kriteria</b>	<b>Subkriteria</b>	<b>Kriteria</b>	<b>Subkriteria</b>
Hasil seleksi ujian tertulis	-		Keterampilan teknis
Minat/kesungguhan	-		Pendidikan terakhir
Sikap	-		Keterampilan komunikasi
Intensitas keikutsertaan pelatihan di BLK Bantul	-	Faktor personal	Intensitas keikutsertaan pelatihan di BLK Bantul
Rekomendasi	-		Rencana setelah selesai pelatihan
Penampilan	-		Kejujuran
			Sikap
			Usia
			Motivasi
			Mental
			Penampilan
			Percaya diri
			Komitmen
			Potensi
			Kesungguhan
			Kesan baik
		Faktor eksternal	Rekomendasi
			Pertimbangan keluarga
			Pertimbangan ekonomi

Berdasarkan Tabel 6.1 dapat dianalisis bahwa kriteria/subkriteria yang ditentukan pembuat keputusan sudah dapat tercakup oleh kriteria/subkriteria yang ditentukan dalam proses *Modified Delphi*. Kriteria/subkriteria yang ditentukan oleh pembuat keputusan yaitu umur, pendidikan terakhir, minat/kesungguhan, sikap, intensitas keikutsertaan pelatihan di BLK Bantul, rekomendasi dan penampilan, sama persis seperti yang ditentukan melalui proses *Modified Delphi*.

Namun untuk kriteria hasil seleksi tertulis yang ditentukan pembuat keputusan, sedikit berbeda dengan proses *Modified Delphi*. Karena dalam proses *Modified Delphi* kriteria tersebut sudah dipecah menjadi lebih spesifik yaitu pengetahuan dan keterampilan teknis.

Penentuan kriteria/subkriteria melalui *Modified Delphi* menghasilkan kriteria yang lebih banyak dibanding penentuan kriteria/subkriteria oleh pembuat keputusan. Hal ini disebabkan karena proses *Modified Delphi* juga mengacu studi literatur, sehingga dapat menjadi referensi bagi para ahli untuk menentukan kriteria/subkriteria. Selain itu dalam *Modified Delphi* juga dilakukan diskusi oleh para ahli untuk menentukan kriteria, sehingga dapat menghasilkan umpan-balik untuk kesepakatan bersama.

Dalam *Modified Delphi* menghasilkan hierarki tiga tingkat yang terdiri dari tujuan, kriteria, subkriteria dan alternatif, seperti pada Gambar 4.18. Kriteria/subkriteria yang sudah ditentukan oleh pembuat keputusan dibuat ke dalam bentuk hierarki 2 tingkat, yang terdiri dari tujuan, kriteria dan alternatif. Hierarki berdasarkan pembuat keputusan ditunjukkan pada Gambar 6.1. Hierarki ini digunakan pada proses AHP.



**Gambar 6.1 Hierarki kriteria berdasarkan pembuat keputusan**

Setelah ditentukan kriteria/subkriteria yang digunakan dalam penilaian seleksi peserta pelatihan dan menghasilkan hierarki, kemudian dilakukan

pembobotan untuk masing-masing kriteria/subkriteria dengan metode AHP. Proses ini dilakukan oleh pembuat keputusan. Hasil bobot prioritas dengan AHP untuk kriteria yang ditentukan oleh pembuat keputusan ditunjukkan pada Tabel 6.2. Dan hasil bobot prioritas dengan AHP untuk kriteria/subkriteria yang ditentukan melalui *Modified Delphi* ditunjukkan pada Tabel 6.3.

**Tabel 6.2 Hasil perhitungan bobot dengan AHP untuk kriteria berdasarkan pembuat keputusan**

Kriteria	Bobot global
Umur	0.100
Pendidikan terakhir	0.200
Hasil seleksi ujian tertulis	0.100
Minat/kesungguhan	0.200
Sikap	0.200
Intensitas keikutsertaan pelatihan di BLK Bantul	0.050
Rekomendasi	0.050
Penampilan	0.100

**Tabel 6.3 Hasil perhitungan bobot dengan AHP untuk kriteria/subkriteria berdasarkan *Modified Delphi***

Kriteria	Bobot	Subkriteria	Bobot	Bobot global
Faktor intelektual	0.297	Pengalaman pelatihan	0.100	0.030
		Pengetahuan	0.100	0.030
		Keterampilan teknis	0.200	0.059
		Pendidikan terakhir	0.400	0.119
		Keterampilan komunikasi	0.200	0.059
Faktor personal	0.539	Intensitas keikutsertaan pelatihan di BLK Bantul	0.053	0.029
		Rencana setelah selesai pelatihan	0.068	0.037

**Tabel 6.3 (lanjutan)**

<b>Kriteria</b>	<b>Bobot</b>	<b>Subkriteria</b>	<b>Bobot</b>	<b>Bobot global</b>
		Kejujuran	0.105	0.057
		Sikap	0.053	0.029
		Usia	0.053	0.029
		Motivasi	0.105	0.057
		Mental	0.105	0.057
		Penampilan	0.033	0.018
		Percaya diri	0.100	0.054
		Komitmen	0.100	0.054
		Potensi	0.100	0.054
		Kesungguhan	0.100	0.054
Faktor eksternal	0.164	Kesan baik	0.023	0.012
		Rekomendasi	0.539	0.088
		Pertimbangan keluarga	0.164	0.027
		Pertimbangan ekonomi	0.297	0.049

Berdasarkan Tabel 6.2 dan Tabel 6.3 mempunyai urutan prioritas yang berbeda. Urutan prioritas untuk kriteria yang ditentukan oleh pembuat keputusan yaitu pendidikan terakhir, minat/kesungguhan, sikap, umur, hasil seleksi ujian tertulis, penampilan, intensitas keikutsertaan pelatihan di BLK Bantul dan rekomendasi. Sedangkan urutan prioritas untuk kriteria/subkriteria yang ditentukan melalui *Modified Delphi* yaitu pendidikan terakhir, rekomendasi, keterampilan teknis, keterampilan komunikasi, kejujuran, motivasi, mental, percaya diri, komitmen, potensi, kesungguhan, pertimbangan ekonomi, rencana setelah selesai pelatihan, pengalaman pelatihan, pengetahuan, intensitas keikutsertaan pelatihan di BLK Bantul, sikap, usia, pertimbangan keluarga, penampilan dan kesan baik.

Bobot yang sudah diperoleh kemudian digunakan untuk penilaian menggunakan PROMETHEE. Penilaian akan menghasilkan peringkat yang digunakan pembuat keputusan sebagai pertimbangan untuk menentukan peserta seleksi yang diterima. Hasil peringkat peserta seleksi dengan PROMETHEE

untuk kriteria/subkriteria yang ditentukan oleh pembuat keputusan dan melalui *Modified Delphi* ditunjukkan pada Tabel 6.4.

**Tabel 6.4 Perbandingan hasil AHP-PROMETHEE dan *Modified Delphi*-AHP-PROMETHEE**

Peringkat	AHP-PROMETHEE	<i>Modified Delphi</i> -AHP-PROMETHEE
1	P18	P3
2	P3	P18
3	P4	P4
4	P6	P6
5	P13	P9
6	P14	P17
7	P16	P14
8	P9	P19
9	P17	P10
10	P19	P5
11	P5	P2
12	P2	P16
13	P10	P7
14	P11	P12
15	P12	P13
16	P15	P11
17	P7	P15
18	P1	P1
19	P8	P8
20	P20	P20
21	P21	P21
22	P22	P22
23	P23	P23
24	P24	P24
25	P25	P25
26	P26	P26
27	P27	P27

**Tabel 6.4 (lanjutan)**

Peringkat	AHP-PROMETHEE	<i>Modified Delphi-AHP-PROMETHEE</i>
28	P29	P29
29	P31	P31
30	P32	P32
31	P30	P30
32	P28	P28

Berdasarkan data masa lalu, peserta seleksi yang diterima yaitu P2, P3, P4, P5, P6, P8, P9, P10, P11, P12, P13, P14, P16, P17, P18 dan P19. Dalam hasil masa lalu tidak ada peringkat, sehingga hasil akhir hanya peserta seleksi yang diterima. Kuota peserta seleksi yang diterima yaitu 16 orang, sehingga peringkat 1-16 adalah peserta seleksi yang direkomendasikan untuk diterima. Berdasarkan Tabel 6.4, terdapat perbedaan antara hasil AHP-PROMETHEE, *Modified Delphi-AHP-PROMETHEE* dan data masa lalu. Hasil AHP-PROMETHEE menunjukkan peserta yang direkomendasikan untuk diterima adalah P18, P3, P4, P6, P13, P14, P16, P9, P17, P19, P5, P2, P10, P11, P12 dan P15. Hasil *Modified Delphi-AHP-PROMETHEE* menunjukkan peserta yang direkomendasikan untuk diterima adalah P3, P18, P4, P6, P9, P17, P14, P19, P10, P5, P2, P16, P7, P12, P13 dan P11.

Jika hasil AHP-PROMETHEE dibandingkan dengan hasil masa lalu, data yang berbeda yaitu P15 dalam hasil masa lalu tidak diterima sedangkan yang diterima adalah P8. Jika hasil *Modified Delphi-AHP-PROMETHEE* dibandingkan dengan hasil masa lalu, data yang berbeda yaitu P7 dalam hasil masa lalu tidak diterima sedangkan yang diterima adalah P8. Setelah data dianalisis, diketahui terdapat penyimpangan data pada P8. Berdasarkan data peserta seleksi pada Tabel 4.1 dan 4.2 diketahui bahwa P8 tidak mengikuti seleksi tertulis dan wawancara. Setelah dikonfirmasi kepada pihak BLK Bantul, hal ini memang memungkinkan terjadi, yang disebabkan oleh beberapa faktor. Berikut ini beberapa kemungkinan faktor yang menyebabkan adanya penyimpangan hasil:

1. Peserta yang seharusnya diterima, mengundurkan diri sebelum adanya pengumuman.

2. Peserta yang memiliki urutan di bawah peserta yang diterima, tidak dapat dihubungi atau tidak bersedia untuk mengikuti pelatihan karena sudah ada aktivitas lain, misalnya sudah diterima bekerja.
3. Adanya rekomendasi dari pihak tertentu.

Menurut pihak BLK Bantul, hasil dengan metode *Modified Delphi-AHP-PROMETHEE* lebih sesuai dibandingkan AHP-PROMETHEE. Karena dengan metode *Modified Delphi*, kriteria yang diperoleh merupakan kesepakatan bersama antara pihak manajemen di BLK Bantul, sehingga kriteria lebih sesuai dengan kondisi lapangan dan lebih detail.

## **6.2 Perbandingan Metode *Modified Delphi-AHP-PROMETHEE* Dan *Modified Delphi-AHP-TOPSIS***

Pengujian ini dilakukan untuk mengetahui tingkat kesesuaian penggunaan metode PROMETHEE, yaitu dengan membandingkan hasil *Modified Delphi-AHP-PROMETHEE* dan hasil *Modified Delphi-AHP-TOPSIS* dengan data masa lalu (belum menggunakan metode MADM). TOPSIS dipilih karena dalam menentukan preferensinya berbeda dengan PROMETHEE yaitu menggunakan solusi ideal. Setelah kriteria ditentukan dengan *Modified Delphi* dan diperoleh bobot masing-masing kriteria dengan AHP, kemudian dicari peringkat peserta seleksi dengan metode PROMETHEE dan TOPSIS. Hasil peringkat dengan menggunakan *Modified Delphi-AHP-PROMETHEE* dan *Modified Delphi-AHP-TOPSIS* ditunjukkan pada Tabel 6.5.

**Tabel 6.5 Perbandingan hasil *Modified Delphi-AHP-PROMETHEE* dan *Modified Delphi-AHP-TOPSIS***

Peringkat	<i>Modified Delphi-AHP-PROMETHEE</i>	<i>Modified Delphi-AHP-TOPSIS</i>
1	P3	P3
2	P18	P18
3	P4	P10
4	P6	P4
5	P9	P5

**Tabel 6.5 (lanjutan)**

<b>Peringkat</b>	<b>Modified Delphi-AHP-PROMETHEE</b>	<b>Modified Delphi-AHP-TOPSIS</b>
6	P17	P6
7	P14	P19
8	P19	P17
9	P10	P14
10	P5	P2
11	P2	P9
12	P16	P12
13	P7	P11
14	P12	P7
15	P13	P16
16	P11	P13
17	P15	P15
18	P1	P1
19	P8	P8
20	P20	P20
21	P21	P21
22	P22	P22
23	P23	P23
24	P24	P24
25	P25	P25
26	P26	P26
27	P27	P27
28	P29	P29
29	P31	P31
30	P32	P32
31	P30	P30
32	P28	P28

Berdasarkan data masa lalu, peserta seleksi yang diterima yaitu P2, P3, P4, P5, P6, P8, P9, P10, P11, P12, P13, P14, P16, P17, P18 dan P19. Pada hasil masa lalu terjadi penyimpangan hasil untuk P8, sehingga dalam hasil *Modified Delphi-AHP-PROMETHEE* dan *Modified Delphi-AHP-TOPSIS* direkomendasikan untuk

tidak diterima. Berdasarkan Tabel 6.5, hasil *Modified Delphi-AHP-PROMETHEE* dan *Modified Delphi-AHP-TOPSIS* mempunyai kemiripan, hanya berbeda urutan saja, sehingga perankingan baik berdasarkan jarak solusi ideal ataupun tipe preferensi dengan PROMETHEE dapat digunakan. Keduanya sesuai dengan hasil data riil jika mengabaikan penyimpangan hasil. Hasil *Modified Delphi-AHP-PROMETHEE* dan *Modified Delphi-AHP-TOPSIS* menunjukkan peserta yang direkomendasikan untuk diterima adalah P2, P3, P4, P5, P6, P7, P9, P10, P11, P12, P13, P14, P16, P17, P18 dan P19. Dengan demikian hasil *Modified Delphi-AHP-PROMETHEE* sesuai untuk digunakan dalam penilaian seleksi peserta pelatihan di BLK Bantul.

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1 Kesimpulan**

Kesimpulan yang dapat diberikan berdasarkan hasil penelitian dan pembahasan sesuai dengan tujuan penelitian yaitu menggabungkan metode AHP dengan metode *Modified Delphi* dan PROMETHEE dapat mengatasi kelemahan AHP dalam penentuan kriteria dan penilaian peserta seleksi. Dengan menggabungkan metode AHP dan *Modified Delphi* menghasilkan 21 subkriteria yang dikelompokkan ke dalam 3 kriteria. Penggabungan metode AHP dan PROMETHEE menghasilkan rekomendasi peserta seleksi yang diterima sesuai dengan data masa lalu, dengan syarat mengabaikan adanya penyimpangan hasil.

#### **7.2 Saran**

Penelitian menggabungkan metode *Modified Delphi-AHP-PROMETHEE* untuk seleksi peserta pelatihan yang dibangun ini masih membutuhkan pengembangan untuk penelitian lanjut guna perbaikan dari kekurangan penelitian. Berikut ini saran yang diberikan untuk penelitian lanjut:

1. Melakukan perbandingan penggabungan metode *Modified Delphi-AHP-PROMETHEE* dengan metode MADM lainnya, karena dalam penelitian ini hanya membandingkan dengan *Modified Delphi-AHP-TOPSIS*, sehingga dapat diketahui dan dianalisis apakah dari beberapa metode lainnya yang dibandingkan ada perbedaan hasil atau tidak dengan yang dibandingkan dalam penelitian ini.
2. Sistem ini belum menangani lebih dari satu pembuat keputusan karena dalam studi kasus hanya satu pembuat keputusan, sehingga bisa dikembangkan sistem *Group Decision Support System* (GDSS) untuk kasus yang mempunyai lebih dari satu pembuat keputusan.

3. Penggunaan gabungan metode *Modified Delphi-AHP-PROMETHEE* pada kasus selain seleksi peserta pelatihan.

## DAFTAR PUSTAKA

- Adler, M. dan Ziglio, E., 1996, *Gazing Into the Oracle*, Jessica Kingsley Publishers, London dan Philadelphia
- Afshari, A., Mojahed, M. dan Yusuff, R., 2010, Simple additive weighting approach to personnel selection problem, *International Journal of Innovation, Management and Technology*, 5, 1, 511-515
- Afshari, A.R., Yusuff, R.M. dan Derayatifar, A.R., 2012, An application of Delphi method for eliciting criteria in personnel selection problem, *Scientific Research and Essays*, 33, 7, 2927–2935
- Alguliyev, R.M.O., Aliguliyev, R.M. dan Mahmudova, R.S., 2015, Multicriteria Personnel Selection by the Modified Fuzzy VIKOR Method, *Scientific World Journal*, 612767, 2015, 1-16
- Barrett, G.V., Miguel, R.F., Hurd, J.M., Lueke, S.B. dan Tan, J.A., 2003, Practical issues in the use of personality tests in police selection, *Public Personnel Management*, 4, 32, 497-517
- Bhushan, N. dan Rai, K., 2004, *Strategic Decision Making Applying the Analytic Hierarchy Process*, Springer-Verlag, London
- Blume, B.D., Ford, J.K., Baldwin, T.T. dan Huang, J.L., 2010, Transfer of Training: A Meta-Analytic Review, *Journal of Management*, 4, 36, 1065-1105
- Bogdanovic, D. dan Miletic, S., 2014, Personnel evaluation and selection by multicriteria decision making method, *Economic Computation and Economic Cybernetics Studies and Research*, 3, 8, 179–196
- Brans, J. dan Mareschal, B., 2005, *PROMETHEE Methods*, Springer, New York, NY
- Brans, J.P. dan Vincke, P., 1985, A Preference Ranking Organisation Method (The PROMETHEE Method for Multiple Criteria Decision Making), *Management Science*, 6, 31, 647–657
- Chen, C.T., Pai, P.F. dan Hung, W.Z., 2013, A New Decision-Making Process for Selecting Project Leader Based on Social Network and Knowledge Map, *International Journal of Fuzzy Systems*, 1, 15, 36-46
- Cornish, E., 1977, *The Study of the Future: An Introduction to the Art and Science of Understanding and Shaping Tomorrow's World*, World Future Society, Washington, D.C.
- Delbecq, A.L., Van de Ven, A.H. dan Gustafson, D.H., 1975, *Group Techniques for Program Planning: A Guide to Nominal Group and Delphi Processes*, Scott Foreman, Glenview, IL
- Dubois, D. dan Prade, H., 1980, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York
- Efendi, S. R., Wardoyo, R. dan Sari, A. K., 2017, Sistem Pendukung Keputusan Penentuan Penerima Bantuan Langsung Masyarakat (BLM) Program

- Peningkatan Kualitas Pemukiman (P2KP) dengan Metode Gabungan dari AHP dan TOPSIS (Studi Kasus: Kota Ternate), , , Universitas Gadjah Mada, Yogyakarta
- El-santawy, M.F., 2012, A VIKOR Method for Solving Personnel Training, *International Journal of Computing Science*, 2, 1, 9–12
- Fajar, S.A. dan Heru, T., 2010, *Manajemen Sumberdaya Manusia Sebagai Dasar Meraih Keunggulan Bersaing*, 1Sekolah Tinggi Ilmu Manajemen YKPN, Yogyakarta
- Fowles, R.B., 1978, *Handbook of Future Research*, Greenwood Press, Connecticut
- Gibney, R. dan Shang, J., 2007, Decision making in academia: A case of the dean selection process, *Mathematical and Computer Modelling*, 7–8, 46, 1030–10400
- Güngör, Z., Serhadlioğlu, G. & Keser, S.E., 2009, A fuzzy AHP approach to personnel selection problem, *Applied Soft Computing*, 2, 9, 641–646
- Helmer, O., 1977, Problems in Future Research: Delphi and Casual Cross-Impact Analysis, *Futures*, 1, 9, 17–31
- Hough, L.M. dan Oswald, F.L., 2000, PERSONNEL SELECTION: Looking Toward the Future—Remembering the Past, *Annual Review of Psychology*, , 51, 631–664
- Hough, L.M., Oswald, F.L. dan Ployhart, R.E., 2001, Determinants, Detection and Amelioration of Adverse Impact in Personnel Selection Procedures: Issues, Evidence and Lessons Learned, *INTERNATIONAL JOURNAL OF SELECTION AND ASSESSMENT*, 1/2, 9, 152–194
- Huang, J.-J. dan Tzeng, G.-H., 2011, *Multiple attribute decision making: methods and applications*, CRC Press, Boca Raton FL
- Hwang, C.-L. dan Yoon, K., 1981, *Multiple attribute decision making: methods and applications*, Springer-Verlag, London
- Ivancevich, J.M. dan Konopaske, R., 2013, *Human Resource Management*, 12McGraw-Hill/Irwin, New York
- Kelemenis, A. dan Askounis, D., 2010, A new TOPSIS-based multi-criteria approach to personnel selection, *Expert Systems with Applications*, 7, 37, 4999–5008
- Kumar, D.S., Radhika, S. dan Suman, K.N.S., 2013, MADM Methods for Finding The Right Personnel in Academic Institutions, *International Journal of u-and e- Service, Science and Technology*, 5, 6, 133–144
- Lemantara, J. dan Setiawan, N. A., 2013, Rancang Bangun Sistem Pendukung Keputusan Pemilihan Mahasiswa Berprestasi Menggunakan Metode AHP dan PROMETHEE, , , Universitas Gadjah Mada, Yogyakarta
- Linstone, H.A. dan Turoff, M., 2002, *The Delphi Method - Techniques and Applications*, Addison-Wesley, New York
- Mousavi, S.M., Tavakkoli-Moghaddam, R., Heydar, M. dan Ebrahimnejad, S., 2013, Multi-Criteria Decision Making for Plant Location Selection: An Integrated Delphi-AHP-PROMETHEE Methodology, *Arabian Journal for*

- Science and Engineering*, 5, 38, 1255–1268
- Noe, R.A., Hollenbeck, J.R., Gerhart, B. dan Wright, P.M., 2012, *Human Resource Management: Gaining a Competitive Advantage 8th ed.*, McGraw-Hill, New York
- Özdemir, A., 2013, A two-phase multi criteria dynamic programing approach for personnel selection process, *Problems and Perspectives in Management*, 2, 11, 98-108
- ÖZTÜRK, A., 2013, Personnel Selection In An Accomodation Enterprise By Promethee Methods, *International Journal of Business and Commerce*, 3, 5, 1-19
- Rao, R.V., 2013, *Decision Making in the Manufacturing Environment Using Graph Theory and Fuzzy Multiple Attribute Decision Making Methods*, Springer-Verlag, London
- Rivai, V. dan Sagala, E.J., 2014, *Manajemen Sumber Daya Manusia untuk Perusahaan: Dari Teori ke Praktik*, 3Raja Grafindo Persada, Jakarta
- Robertson, I.T. dan Smith, M., 2001, Personnel selection, *Journal of Occupational and Organizational Psychology*, 4, 74, 441–472
- Rouyendegh, B.D. dan Erkan, T.E., 2013, An Application of the Fuzzy ELECTRE Method for Academic Staff Selection, *Human Factors and Ergonomics in Manufacturing*, 2, 23, 07–115
- Saaty, T.L. dan Vargas, L.G., 2001, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*, Springer Science & Business Media, New York
- Safrizal dan Tanti, 2015: Safrizal dan Tanti, L., Penerapan Metode Promethee Dalam Penyeleksian Siswa Baru (Airlines Staff) pada LPP Penerbangan, 2015
- Salehi, K., 2016, An Integrated Approach of Fuzzy AHP and Fuzzy VIKOR for Personnel Selection Problem, *Global Journal of Management Studies and Researches*, 3, 3, 89–95
- Sudipa, I. G. I dan Hartati, S., 2017, Pembuatan Keputusan Seleksi Pemberian Beasiswa Menggunakan Metode AHP, ROC dan SAW (Studi Kasus: STIKI BALI), , Universitas Gadjah Mada, Yogyakarta
- Triantaphyllou, E., 2000, *Multi-Criteria Decision Making Methods: A Comparative Study*, Springer Science & Business Media, Dordrecht
- Ubaidi, 2015, Sistem Pendukung Keputusan Penerimaan Siswa Baru dengan Metode Promethee (Studi Kasus SD Plus Nurul Hikmah Pamekasan), , Surabaya, 19 Maret 2015
- Vinchur, A. J., Schippmann, J. S., Switzer III, F. S. dan Roth, P. L. , 1998, A meta-analytic review of predictors of job performance for salespeople, *Journal of applied psychology*, 4, 83, 586

**LAMPIRAN A**