

**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  
DEPARTEMEN 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  
DEPARTEMEN ILMU KOMPUTER DAN ELEKTRONIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS GADJAH MADA  
YOGYAKARTA**

**2018**

**HALAMAN PERSETUJUAN**

**PRA 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 disetujui  
pada tanggal <masukan tanggal ujian>

Pembimbing

Dr. Yohanes Suyanto, M.I.Kom.  
Pembimbing I

Dr. Sigit Priyanta, S.Si., M.Kom.  
Pembimbing II

## **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. Triyono, S.U. 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>INTISARI.....</b>	xii
<b>ABSTRACT.....</b>	xiii
<b>BAB I</b>	
<b>PENDAHULUAN.....</b>	14
1.1 Latar Belakang .....	14
1.2 Rumusan Masalah.....	16
1.3 Batasan Masalah.....	16
1.4 Tujuan Penelitian.....	17
1.5 Manfaat Penelitian.....	17
1.6 Keaslian Penelitian.....	17
1.7 Metodologi Penelitian.....	18
1.8 Sistematika Penulisan.....	20
<b>BAB II</b>	
<b>TINJAUAN PUSTAKA.....</b>	22
<b>BAB III</b>	
<b>LANDASAN TEORI.....</b>	29
3.1 Metode MADM (Multiple Attribute Decision-Making) .....	29
3.2 Metode Modified Delphi .....	31
3.3 Metode AHP (Analytic Hierarchy Process) .....	33
3.4 Tahapan Dalam AHP .....	34
3.5 PROMETHEE (Preference Ranking Organization Method For Enrichment Evaluation).....	37
<b>BAB IV</b>	
<b>ANALISIS DAN RANCANGAN SISTEM.....</b>	39
4.1 Analisis Data .....	39
4.2 Pertimbangan Penggunaan Metode Modified Delphi-AHP-PROMETHEE.....	51
4.3 Analisis Kebutuhan Sistem .....	52
4.4 Perancangan Sistem.....	55
4.5 Perancangan Basis Data.....	64
4.6 Perancangan Antarmuka.....	78
4.7 Perancangan Pemodelan.....	80
4.8 Rangkaian Penggunaan Pemodelan.....	90
<b>BAB V</b>	
<b>IMPLEMENTASI.....</b>	112
5.1 Implementasi Metode Modified Delphi.....	112
5.2 Implementasi Metode AHP.....	129
5.3 Implementasi Metode PROMETHEE.....	135

<b>BAB VI</b>	
<b>HASIL PENELITIAN DAN PEMBAHASAN.....</b>	<b>148</b>
6.1 Perbandingan Metode AHP-PROMETHEE Dan Modified Delphi-AHP-PROMETHEE.....	148
6.2 Perbandingan Metode Modified Delphi-AHP-PROMETHEE Dan Modified Delphi-AHP-TOPSIS.....	155
<b>BAB VII</b>	
<b>KESIMPULAN DAN SARAN.....</b>	<b>158</b>
7.1 Kesimpulan.....	158
7.2 Saran.....	158
<b>LAMPIRAN.....</b>	<b>162</b>

## DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka.....	27
Tabel 3.1 Skala untuk Perbandingan Kuantitatif Alternatif (Bhushan dan Rai, 2004).....	36
Tabel 4.1 Identitas pendaftar.....	40
Tabel 4.2 Hasil penilaian pendaftar.....	42
Tabel 4.3 Kriteria seleksi personil dari studi literatur.....	49
Tabel 4.4 Deskripsi use case .....	53
Tabel 4.5 Struktur tabel users.....	67
Tabel 4.6 Struktur tabel uploads.....	67
Tabel 4.7 Struktur tabel educations.....	68
Tabel 4.8 Struktur tabel educational_background.....	68
Tabel 4.9 Struktur tabel courses.....	70
Tabel 4.10 Struktur tabel course_experience.....	70
Tabel 4.11 Struktur tabel registrants.....	70
Tabel 4.12 Struktur tabel roles.....	71
Tabel 4.13 Struktur tabel role_user.....	72
Tabel 4.14 Struktur tabel permissions.....	72
Tabel 4.15 Struktur tabel permission_role .....	72
Tabel 4.16 Struktur tabel vocational.....	73
Tabel 4.17 Struktur tabel sub_vocational.....	73
Tabel 4.18 Struktur tabel registrations.....	74
Tabel 4.19 Struktur tabel selection_schedules.....	74
Tabel 4.20 Struktur tabel selections.....	75
Tabel 4.21 Struktur tabel criterias.....	76
Tabel 4.22 Struktur tabel choice.....	77
Tabel 4.23 Struktur tabel result_selection.....	77
Tabel 4.24 Struktur tabel pairwise_comparisons.....	78
Tabel 4.25 Fungsi Preferensi (Brans dan Mareschal, 2005).....	84
Tabel 4.26 Hasil kesesuaian kriteria pada Delphi putaran pertama.....	90
Tabel 4.27 Hasil kriteria pada Delphi putaran kedua.....	92
Tabel 4.28 Matriks perbandingan berpasangan kriteria.....	95
Tabel 4.29 Normalisasi matriks perbandingan berpasangan kriteria.....	95
Tabel 4.30 Hasil bobot kriteria.....	96
Tabel 4.31 Matriks perbandingan berpasangan subkriteria untuk kriteria FI.....	96
Tabel 4.32 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FI.....	97
Tabel 4.33 Hasil bobot subkriteria untuk kriteria FI.....	97
Tabel 4.34 Matriks perbandingan berpasangan subkriteria untuk kriteria FP.....	98
Tabel 4.35 Normalisasi matriks perbandingan berpasangan subkriteria untuk kriteria FP.....	98
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 6.1 Perbandingan kriteria/subkriteria berdasarkan metode Modified Delphi dan pembuat keputusan.....	148
Tabel 6.2 Hasil perhitungan bobot dengan AHP untuk kriteria berdasarkan pembuat keputusan.....	151
Tabel 6.3 Hasil perhitungan bobot dengan AHP untuk kriteria/subkriteria berdasarkan Modified Delphi.....	151
Tabel 6.4 Perbandingan hasil AHP-PROMETHEE dan Modified Delphi-AHP-PROMETHEE.....	153
Tabel 6.5 Perbandingan hasil Modified Delphi-AHP-PROMETHEE dan Modified Delphi-AHP-TOPSIS.....	155

## DAFTAR GAMBAR

Gambar 3.1 Sistem Hierarki untuk MADM (Dubois dan Prade, 1980).....	31
Gambar 3.2 Struktur Umum Hierarki AHP (Bhushan dan Rai, 2004).....	35
Gambar 3.3 Format untuk Perbandingan Berpasangan (Bhushan dan Rai, 2004).....	35
Gambar 4.1 Use case diagram sistem.....	53
Gambar 4.2 Activity diagram mendaftar seleksi.....	56
Gambar 4.3 Activity diagram mengelola kriteria pustaka.....	57
Gambar 4.4 Activity diagram mengisi kuesioner kriteria.....	58
Gambar 4.5 Activity diagram melihat hasil kriteria tahap 1 .....	59
Gambar 4.6 Activity diagram mengelola kriteria tahap 2 .....	59
Gambar 4.7 Activity diagram mengelola hierarki kriteria.....	60
Gambar 4.8 Activity diagram mengelola bobot kriteria .....	61
Gambar 4.9 Activity diagram mengelola tipe preferensi.....	62
Gambar 4.10 Activity diagram mengelola nilai alternatif.....	63
Gambar 4.11 Activity diagram melihat hasil ranking.....	64
Gambar 4.12 Entity Relationship Diagram (ERD).....	66
Gambar 4.13 Implementasi basis data.....	69
Gambar 4.14 Rancangan antarmuka halaman utama .....	79
Gambar 4.15 Proses model keputusan seleksi peserta pelatihan.....	81
Gambar 4.16 Flowchart model keseluruhan.....	86
Gambar 4.17 Flowchart model Modified Delphi.....	87
Gambar 4.18 Flowchart model AHP.....	88
Gambar 4.19 Flowchart model PROMETHEE.....	89
Gambar 4.20 Hierarki kriteria seleksi peserta pelatihan.....	94
Gambar 5.1 Implementasi halaman pengelolaan kriteria dari studi literatur.....	112
Gambar 5.2 Potongan source code untuk menampilkan daftar kriteria dari studi literatur.....	113
Gambar 5.3 Potongan source code untuk memproses masukan dari halaman tambah kriteria studi literatur.....	113
Gambar 5.4 Potongan source code untuk menampilkan detail kriteria studi literatur.....	114
Gambar 5.5 Potongan source code untuk memproses perubahan dari halaman edit kriteria studi literatur.....	114
Gambar 5.6 Potongan source code untuk menghapus kriteria studi literatur.....	115
Gambar 5.7 Implementasi halaman kuesioner kriteria .....	115
Gambar 5.8 Implementasi hasil isian kuesioner kriteria.....	116
Gambar 5.9 Potongan source code untuk menampilkan hasil isian kuesioner....	117
Gambar 5.10 Potongan source code untuk memproses isian kuesioner.....	118
Gambar 5.11 Implementasi ahli mengisi kuesioner kriteria.....	118
Gambar 5.12 Implementasi halaman hasil Delphi putaran pertama.....	119
Gambar 5.13 Potongan source code untuk menampilkan hasil Delphi putaran pertama.....	120

Gambar 5.14 Halaman implementasi Delphi putaran kedua.....	121
Gambar 5.15 Potongan source code untuk menampilkan halaman utama pengelolaan kriteria dalam Delphi putaran kedua.....	122
Gambar 5.16 Potongan source code untuk memproses masukan dari halaman tambah kriteria dalam Delphi putaran kedua.....	123
Gambar 5.17 Potongan source code untuk memproses perubahan dari halaman edit kriteria dalam Delphi putaran kedua.....	124
Gambar 5.18 Potongan source code untuk menghapus suatu kriteria dalam Delphi putaran kedua.....	124
Gambar 5.19 Potongan source code untuk menggunakan kriteria pada Delphi putaran pertama.....	125
Gambar 5.20 Implementasi saat proses diskusi menentukan kriteria.....	125
Gambar 5.21 Implementasi halaman hierarki kriteria.....	126
Gambar 5.22 Potongan source code untuk menampilkan halaman hierarki kriteria .....	127
Gambar 5.23 Potongan source code untuk memproses masukan dari halaman tambah kelompok kriteria.....	128
Gambar 5.24 Potongan source code untuk memproses perubahan dari halaman edit kelompok kriteria.....	128
Gambar 5.25 Potongan source code untuk menghapus kelompok kriteria.....	128
Gambar 5.26 Potongan source code untuk menambahkan kriteria ke dalam kelompok kriteria.....	129
Gambar 5.27 Potongan source code untuk menghapus kriteria dari kelompok kriteria.....	129
Gambar 5.28 Implementasi halaman perbandingan berpasangan.....	130
Gambar 5.29 Implementasi halaman bobot.....	130
Gambar 5.30 Potongan source code untuk menampilkan halaman bobot.....	131
Gambar 5.31 Potongan source code untuk menghitung bobot global.....	131
Gambar 5.32 Potongan source code untuk memproses masukan perbandingan berpasangan.....	132
Gambar 5.33 Potongan source code untuk membentuk matriks hasil perbandingan berpasangan.....	132
Gambar 5.34 Potongan source code untuk membentuk matriks hasil normalisasi perbandingan berpasangan.....	133
Gambar 5.35 Potongan source code untuk menghitung bobot lokal.....	133
Gambar 5.36 Potongan source code untuk menghitung lamda max.....	134
Gambar 5.37 Potongan source code untuk menghitung CI.....	134
Gambar 5.38 Potongan source code untuk menentukan nilai RI.....	134
Gambar 5.39 Potongan source code untuk menghitung CR.....	135
Gambar 5.40 Potongan source code untuk mengecek apakah perbandingan berpasangan konsisten.....	135
Gambar 5.41 Implementasi halaman tambah nilai peserta seleksi pelatihan.....	136
Gambar 5.42 Potongan source code untuk memproses masukan dari halaman tambah nilai peserta seleksi pelatihan.....	137

Gambar 5.43 Implementasi halaman pengelolaan fungsi preferensi.....	138
Gambar 5.44 potongan source code untuk menampilkan halaman pengelolaan fungsi preferensi.....	138
Gambar 5.45 Potongan source code untuk menampilkan halaman pengelolaan fungsi preferensi.....	139
Gambar 5.46 Implementasi halaman penilaian peserta seleksi pelatihan.....	140
Gambar 5.47 Potongan source code untuk menampilkan halaman penilaian peserta seleksi pelatihan.....	141
Gambar 5.48 Potongan source code untuk membentuk matriks perbandingan berpasangan antara alternatif dan kriteria.....	142
Gambar 5.49 Potongan source code untuk menghitung selisih antar-alternatif pada setiap kriteria.....	142
Gambar 5.50 Potongan source code untuk menghitung derajat preferensi antar-alternatif pada setiap kriteria.....	143
Gambar 5.51 Potongan source code untuk menghitung indeks preferensi.....	144
Gambar 5.52 Potongan source code untuk menghitung leaving flow dan entering flow.....	144
Gambar 5.53 Potongan source code untuk mengetahui hubungan antar-alternatif .....	145
Gambar 5.54 Potongan source code untuk meranking alternatif.....	146
Gambar 5.55 Implementasi halaman hasil peringkat peserta seleksi .....	147
Gambar 5.56 Potongan source code untuk menampilkan hasil peringkat peserta seleksi pelatihan.....	147
Gambar 6.1 Hierarki kriteria berdasarkan pembuat keputusan.....	150

## **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. Seleksi peserta pelatihan juga dilakukan oleh Balai Latihan Kerja (BLK) Bantul. BLK Bantul merupakan lembaga pelatihan kerja milik pemerintah di Kabupaten Bantul yang mempunyai maksud dan tujuan dalam upaya mengurangi pengangguran.

Tahapan proses seleksi di BLK Bantul yaitu pengisian identitas diri, seleksi tertulis, seleksi wawancara dan pembuatan keputusan seleksi. Penilaian seleksi belum berdasarkan standar kriteria tertentu, karena BLK Bantul belum memiliki ketetapan standar kriteria yang digunakan dalam seleksi peserta pelatihan. Belum adanya ketetapan standar kriteria dapat menyebabkan kekeliruan dalam penentuan peserta seleksi yang diterima. Keputusan seleksi didominasi oleh hasil seleksi wawancara dan belum berdasarkan perhitungan penilaian dari seluruh proses seleksi.

Belum adanya perhitungan penilaian seleksi peserta pelatihan, mengakibatkan belum adanya transparansi dalam penentuan hasil seleksi. Hasil seleksi diumumkan tidak berdasarkan peringkat peserta seleksi, namun hanya diumumkan peserta seleksi yang diterima saja. Hal ini karena belum ada peringkat peserta seleksi pelatihan berdasarkan hasil penilaian seleksi. Maka dari itu, BLK

Bantul memerlukan sistem yang mampu membantu membuat keputusan dalam pemilihan peserta pelatihan berdasarkan kriteria-kriteria seleksi dan perhitungan penilaian yang dapat dipertanggungjawabkan.

Dalam beberapa literatur untuk memecahkan masalah seleksi personil digunakan metode *Multiple Attribute 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), PROMETHEE (ÖZTÜRK, 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 perankingan. Hal ini membutuhkan metode untuk merumuskan kriteria, pembobotan kriteria dan menentukan peringkat peserta seleksi pelatihan. Metode AHP digunakan untuk menentukan pembobotan terhadap kriteria, 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. Salehi (2016), Lemantara dan Setiawan (2013) dan Mousavi dkk. (2013) merekomendasikan penggabungan metode AHP dengan metode MADM lainnya, karena penggabungan metode menghasilkan kualitas pemeringkatan yang baik. Metode yang digunakan yaitu PROMETHEE untuk menentukan peringkat terhadap alternatif. PROMETHEE dipilih karena setiap kriteria ditentukan fungsi

preferensinya, sehingga setiap kriteria dapat dievaluasi secara berbeda dan memperoleh keputusan yang lebih baik (ÖZTÜRK, 2013). 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 membangun model penilaian seleksi peserta pelatihan di BLK Bantul yang berdasarkan standar kriteria tertentu.

## 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 membangun model penilaian seleksi peserta pelatihan di BLK Bantul dengan menggabungkan metode *Modified Delphi*, AHP dan PROMETHEE.

#### **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), PROMETHEE (ÖZTÜRK, 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 penilaianya.
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 menggunakan bahasa pemrograman PHP dengan framework Laravel 5.2 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 *Modified Delphi*, 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 *Modified Delphi-AHP-PROMETHEE*,

analisis kebutuhan sistem, perancangan sistem dengan menggunakan *Unified Modelling Language* (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 beberapa tahap. Perumusan model DP menggunakan bobot AHP. Tujuan dari menggabungkan metode tersebut yaitu untuk menentukan alternatif keputusan yang optimal dalam setiap tahap dari proses seleksi personil. Penggunaan model DP tidak selalu cocok digunakan pada semua kasus. Hal ini karena dalam model DP, persoalan dibagi menjadi beberapa tahap, yang pada setiap tahap hanya diambil satu keputusan. Solusi pada setiap tahap dibangun dari hasil solusi tahap sebelumnya. Kriteria ditentukan perusahaan berdasarkan proses

seleksi personil perusahaan. Pendekatan metode AHP juga digunakan Sudipa dan Hartati (2017) yaitu dengan menggabungkan metode AHP, ROC dan SAW untuk seleksi penerima beasiswa PPA dan BBP di STIKOM Indonesia (STIKI). Dalam penelitian ini terdapat 3 kriteria yang masing-masing kriteria memiliki 3 subkriteria dan 22 calon penerima beasiswa. 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. Sistem berjalan dengan benar. Namun ada perbedaan hasil ranking, perhitungan STIKI dan sistem karena terletak pada jumlah kriteria penilaian yang digunakan. Efendi dkk. (2017) menggabungkan metode AHP dan TOPSIS untuk menentukan penerima Bantuan Langsung Masyarakat (BLM) yang sesuai bagi kelurahan berdasarkan standar parameter yang telah ditetapkan oleh P2KP Kota Ternate. Dalam penelitian ini menerapkan 13 kriteria untuk menentukan penerima BLM yang sesuai dari 7 kandidat kelurahan yang ada. AHP untuk menghitung bobot prioritas dari tiap parameter, yang nantinya menjadi acuan perankingan yang dilakukan dengan metode TOPSIS. Dari keseluruhan pengujian yang dilakukan menunjukkan hasil yang sesuai. Pengujian yang dilakukan antara lain pengujian input sistem, pengujian kesesuaian perhitungan sistem, pengujian fleksibilitas sistem, pengujian input paramater dan alternatif dicocokkan keputusan *team leader* P2KP Kota Ternate. Namun belum ada pengujian yang membandingkan hasil dengan metode lain.

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. Kriteria ditentukan oleh ahli.

Salehi (2016) menggunakan model *hybrid fuzzy* MADM, 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. *Comparison Matrix* yang dimaksud merupakan tahapan dalam metode AHP. Penentuan peringkat dalam SAW sangat sederhana. Namun belum dapat menangani beberapa kriteria yang memiliki struktur kualitatif, sehingga kurang cocok untuk menangani permasalahan yang kompleks. Penelitian menerapkan 7 kriteria untuk memilih personil terbaik dari 5 personil yang ada.

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 dan peringkat lengkap 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. 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.

Lemantara dan Setiawan (2013) 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. 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. 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. Metode *Modified Delphi* mencakup tiga putaran survei dengan 9 orang ahli dan menghasilkan 14 kriteria yang disusun ke dalam 4 kelompok utama. Hasil akhir metode *Modified Delphi* yaitu berupa hierarki kriteria.

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	Hasil	Komentar
1	Gibney dan Shang (2007)	AHP	AHP berhasil memecahkan masalah seleksi personil di dunia akademis. Namun, belum ada kesepakatan dari semua pembuat keputusan dalam menentukan kriteria, sehingga hasil diskusi kelompok dan penerapan AHP berbeda dengan keputusan rektor.	Membutuhkan model untuk menentukan kriteria yang digunakan dalam proses AHP dari semua pembuat keputusan.
2	Özdemir (2013)	- AHP - DP	AHP untuk menghitung kepentingan relatif semua elemen dalam hierarki. DP untuk membagi proses seleksi personil menjadi beberapa tahap, yang pada setiap tahap hanya dapat diambil satu keputusan.	Pengambilan keputusan di BLK bukan berdasarkan setiap tahap, namun merupakan hasil kalkulasi dari seluruh tahap.
3	Sudipa dan Hartati (2017)	- AHP - ROC - SAW	Adanya perbedaan hasil ranking, perhitungan STIKI dan sistem karena terletak pada jumlah kriteria penilaian yang digunakan.	Membutuhkan model untuk menentukan kriteria penilaian yang digunakan.
4	Efendi dkk. (2017)	- AHP - TOPSIS	Keseluruhan pengujian yang dilakukan menunjukkan hasil yang sesuai. Penentuan kriteria berdasarkan pedoman P2KP Kota Ternate.	- Belum ada pengujian yang membandingkan hasil dengan metode lain. - Penentuan kriteria sudah ada pedomannya.
5	Güngör dkk. (2009)	- <i>Fuzzy AHP</i> - <i>Yager's weighted goals</i>	- Hasil FAHP dibandingkan dengan metode <i>Yager's weighted goals</i> , menunjukkan bahwa ada konsistensi yang baik antara keduanya. - Kriteria ditentukan oleh ahli.	- Perhitungan dalam metode <i>Yager's weighted goals</i> membutuhkan perhitungan operasional yang lebih kompleks dibanding AHP. - Penentuan kriteria selain dari ahli, bisa dipertimbangkan berdasarkan literatur.
6	Salehi (2016)	- <i>Fuzzy AHP</i> - <i>Fuzzy VIKOR</i>	Kombinasi metode MADM untuk pemecahan masalah seleksi personil direkomendasikan untuk penelitian selanjutnya. Penelitian selanjutnya dapat mengkombinasikan metode selain AHP-VIKOR.	Metode penentuan kriteria tidak dijelaskan karena studi kasus menggunakan penelitian sebelumnya.

**Tabel 2.1 (lanjutan)**

No	Nama	Metode	Hasil	Komentar
7	Afshari dkk. (2010)	- <i>Comparison Matrix</i> - SAW	Penentuan peringkat dalam SAW sangat sederhana. Namun belum dapat menangani beberapa kriteria yang memiliki struktur kualitatif, sehingga kurang cocok untuk menangani permasalahan yang kompleks	Menggunakan metode lain untuk penentuan peringkat, yang dapat menangani beberapa kriteria yang memiliki struktur kualitatif.
8	ÖZTÜRK (2013)	PROMETHEE	- Dapat memberikan peringkat parsial dan peringkat lengkap - Setiap kriteria ditentukan fungsi preferensinya, sehingga setiap kriteria dapat dievaluasi secara berbeda dan memperoleh keputusan yang lebih baik.	- Tidak adanya pengecekan konsistensi dalam pembobotan kriteria. - Kriteria ditentukan melalui wawancara dengan manajer HRD dari perusahaan akomodasi dan kajian literatur tentang seleksi personil.
9	Lemantara dan Setiawan (2013)	- AHP - PROMETHEE	Kombinasi AHP-PROMETHEE menghasilkan kualitas pemeringkatan yang baik, karena AHP menjamin konsistensi untuk pembobotan kriteria dan PROMETHEE memiliki <i>adjustment</i> .	- Kombinasi AHP-PROMETHEE dapat digunakan. - Sudah ada ketetapan kriteria yang digunakan dalam penilaian.
10	Bogdanovic dan Miletic (2014)	- AHP - PROMETHEE	Integrasi metode yang digunakan dapat berhasil memecahkan masalah pemilihan personil.	Tidak dijelaskan cara menentukan kriteria.
11	Afshari dkk. (2012)	<i>Modified Delphi</i>	- Digunakan untuk mencari ide-ide terbaik dari para manajer dan para ahli untuk pemilihan kriteria. - Menunjukkan hasil yang sangat baik dalam pemilihan kriteria dan dapat meningkatkan efisiensi dalam suatu proses pengambilan keputusan. - Hasil akhir berupa hierarki kriteria.	- Metode <i>Modified Delphi</i> cocok diimplementasikan dalam pemilihan kriteria suatu proses pengambilan keputusan. - Belum diimplementasikan dengan metode MADM. - Hasil akhir berupa hierarki kriteria dapat digunakan dalam proses AHP.
12	Mousavi dkk. (2013)	- <i>Modified Delphi</i> - AHP - PROMETHEE	Gabungan ketiga metode dapat memecahkan masalah pemilihan lokasi pabrik yang tepat. Peneliti merekomendasikan penggunaan gabungan metode ini di bidang lain.	<i>Modified Delphi</i> dalam penelitian ini belum mempertimbangkan kajian literatur. Daftar kriteria yang diperoleh berdasarkan hasil bertanya dengan ahli profesional atau membuat keputusan.

## 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.

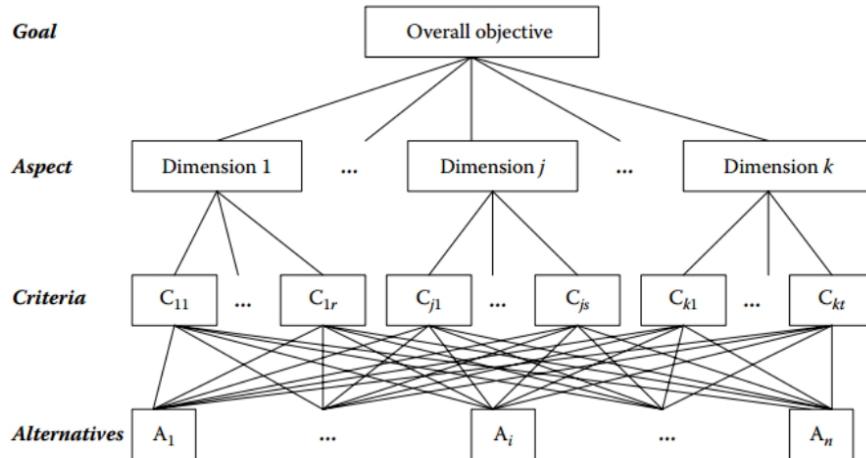
**Langkah 3** : Memilih model evaluasi yang tepat.

**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.

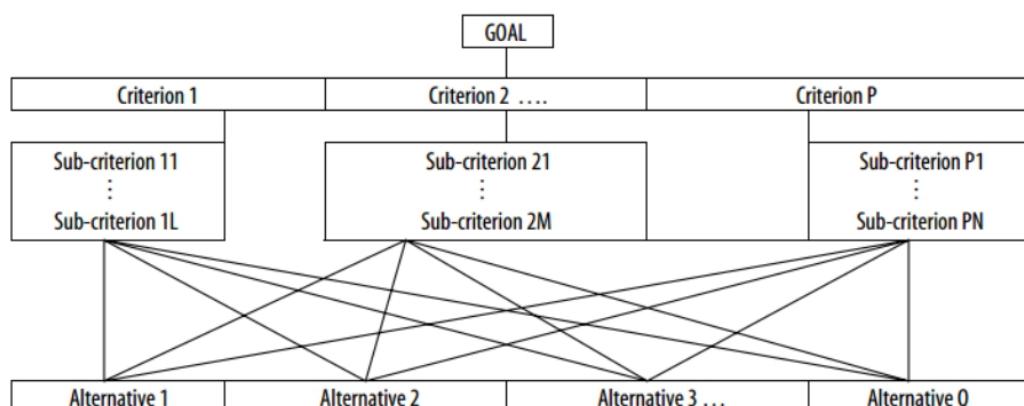
Saaty dan Vargas (2001) menjelaskan tujuh pilar dalam AHP yaitu 1) skala rasio, proporsionalitas dan skala rasio ternormalisasi; 2) perbandingan berpasangan; 3) sensitivitas dari *eigenvector* yang tepat; 4) *clustering* dan menggunakan *pivot* untuk memperpanjang skala; 5) sintesis untuk membuat rasio 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  $\lambda_{\max}$  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.

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

(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. 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 (usia), PT (pendidikan terakhir), PP (pengalaman pelatihan), IK (intensitas keikutsertaan pelatihan di BLK Bantul), KT (keterampilan teknis), PG (pengetahuan), 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), SI (sikap), KJ (kejujuran), RK (rekomendasi) dan RS (rencana setelah selesai pelatihan) sudah disesuaikan dengan penamaan kriteria/subkriteria yang ditentukan pada proses *Modified-Delphi*. Peserta seleksi pelatihan yang diterima yaitu P2, P3, P4, P5, P6, P8, P9, P10, P11, P12, P13, P14, P16, P17, P18 dan P19.

Dalam Tabel 4.1 terlihat bahwa terdapat 32 pendaftar yang semuanya belum pernah mengikuti pelatihan di BLK Bantul. Mayoritas pendaftar belum memiliki pengalaman pelatihan. Latar belakang pendidikan terakhir pendaftar didominasi oleh SMK. Dan usia pendaftar didominasi oleh usia 17-25 tahun. Nama pendaftar diinsalkan karena untuk alasan keamanan.

**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
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

Tabel 4.2 menyajikan hasil penilaian seleksi tertulis dan wawancara pendaftar. Dalam Tabel 4.2 terdapat 14 pendaftar yang nilainya kosong. Hal ini disebabkan karena pendaftar tidak mengikuti seleksi tertulis dan wawancara. Pendaftar yang tidak mengikuti seleksi tertulis dan wawancara yaitu P8, P20, P21, P22, P23, P24, P25, P26, P27, P28, P29, P30, P31 dan P32. Hasil seleksi tertulis terdiri dari keterampilan teknis dan pengetahuan. Hasil seleksi tertulis merepresentasikan kemampuan pendaftar dalam bidang pelatihan yang ingin diikuti. Hasil seleksi wawancara terdiri dari hasil penilaian terhadap kesan baik, kesungguhan, potensi, pertimbangan ekonomi, komitmen, percaya diri, keterampilan komunikasi, penampilan, pertimbangan keluarga, mental, motivasi, sikap, kejujuran, rekomendasi dan rencana setelah selesai pelatihan. Hasil seleksi wawancara merepresentasikan kemampuan personal pendaftar dalam aspek kepribadian.

Dari keseluruhan pendaftar yang mengikuti seleksi wawancara, hasil penilaian terhadap kesan baik, kesungguhan, percaya diri, keterampilan komunikasi, penampilan, mental dan sikap didominasi oleh hasil baik. Sedangkan mayoritas pendaftar yang mengikuti seleksi wawancara bukan merupakan rekomendasi dari pihak tertentu. Semua pendaftar yang mengikuti seleksi wawancara diijinkan oleh pihak keluarga dan sanggup berkomitmen untuk mengikuti pelatihan. Dari segi ekonomi, mayoritas pendaftar yang mengikuti seleksi wawancara berasal dari ekonomi yang kurang. Meskipun dari ekonomi yang kurang, mayoritas memiliki potensi terhadap bidang pelatihan yang ingin diikuti. Motivasi pendaftar untuk mengikuti pelatihan didominasi oleh kemauan sendiri. Rencana setelah selesai mengikuti pelatihan pendaftar beragam yaitu antara lain mencari ilmu, pengalaman, membuka usaha dan melamar pekerjaan. Kejujuran mayoritas pendaftar sesuai. Kejujuran diperoleh dari kesesuaian jawaban yang diisikan pendaftar dalam form pendaftaran dengan wawancara, kesesuaian apakah merupakan rekomendasi atau tidak dan mengerjakan dengan jujur saat seleksi tertulis.

## Tabel 4.2 Hasil penilaian pendaftar

Tabel 4.2 (lanjutan)

Proses penilaian seleksi belum berdasarkan standar kriteria tertentu, karena BLK Bantul belum memiliki ketetapan standar kriteria yang digunakan dalam seleksi peserta pelatihan. Hal ini membutuhkan perumusan kriteria. Berdasarkan pengumpulan data melalui studi literatur, diperoleh data kriteria seleksi personil secara umum, yang ditunjukkan pada Tabel 4.3. Data kriteria ini digunakan dalam proses metode *Modified Delphi*.

Berdasarkan Tabel 4.3, kriteria yang dapat digunakan dalam seleksi personil yaitu pengalaman pelatihan, orientasi masa depan, rekomendasi, kesehatan, kepribadian, pengetahuan, keterampilan teknis, kejujuran, bebas obat-obatan terlarang, sikap, pendidikan formal, pengalaman kerja, jenis kelamin, usia, kefasihan bahasa asing, motivasi, mental, pertimbangan keluarga, ras dan etnis, status perkawinan, karakteristik fisik, kemampuan kognitif, keterampilan komputer, penampilan, keterampilan komunikasi, percaya diri, kemampuan emosional, komitmen, keuangan, potensi, kesungguhan dan kesan baik. Setiap kriteria mempunyai tujuan masing-masing.

Kriteria pengalaman pelatihan digunakan untuk mengetahui keterampilan yang sudah dimilikinya. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Bogdanovic dan Miletic (2014), Kumar dkk. (2013), Rouyendegh dan Erkan (2013), El-santawy dkk. (2012), Blume dkk. (2010) dan Robertson dan Smith (2001).

Kriteria orientasi masa depan digunakan untuk mengetahui rencana yang dilakukan seseorang setelah mengikuti pelatihan. Kriteria ini mereferensi dari Noe dkk. (2012) dan Güngör dkk. (2009).

Kriteria rekomendasi digunakan untuk mengetahui apakah seseorang merupakan rekomendasi dari pegawai atau pihak tertentu. Rekomendasi ini diusulkan oleh suatu pihak yang menyatakan bahwa seseorang memiliki potensi yang baik. Kriteria ini mereferensi dari Noe dkk. (2012) dan Ivancevich dan Konopaske (2013).

Kriteria kesehatan digunakan untuk mengetahui kondisi kesehatan jasmani seseorang yang meliputi tekanan darah, apakah mengidap penyakit tertentu, dll. Kriteria ini mereferensi dari Noe dkk. (2012), Robertson dan Smith (2001), Fajar dan Heru (2010) dan Safrizal dan Tanti (2015).

Kriteria kepribadian digunakan untuk mengkategorikan individu seperti apa. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Robertson dan Smith (2001), dan Fajar dan Heru (2010).

Kriteria pengetahuan digunakan untuk mengukur tingkat pemahaman seseorang terhadap suatu bidang keilmuan. Kriteria ini mereferensi dari Noe dkk. (2012), Bogdanovic dan Miletic (2014), El-santawy dkk. (2012), Robertson dan Smith (2001), Fajar dan Heru (2010), Ubaidi (2015) dan Hough dan Oswald (2000).

Kriteria keterampilan teknis digunakan untuk mengukur kemampuan seseorang dalam hal keterampilan teknis suatu bidang keilmuan. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Kumar dkk. (2013), Rouyendegh dan Erkan (2013) dan Fajar dan Heru (2010).

Kriteria kejujuran digunakan untuk mengetahui tingkat kejujuran seseorang. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013) dan Robertson dan Smith (2001).

Kriteria bebas obat-obatan terlarang digunakan untuk mengetahui apakah seseorang tidak menggunakan obat-obatan terlarang. Kriteria ini mereferensi dari Noe dkk. (2012).

Kriteria sikap digunakan untuk mengetahui apakah seseorang memiliki sikap, sopan dan santun yang baik atau tidak. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Güngör dkk. (2009) dan Fajar dan Heru (2010).

Kriteria pendidikan formal digunakan untuk mengetahui profil seseorang apakah profilnya sesuai dengan yang diharapkan atau tidak. Kriteria pendidikan formal diperoleh dari data riwayat pendidikan yang diisikan dalam formulir

pendaftaran. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), El-santawy dkk. (2012), Blume dkk. (2010), Robertson dan Smith (2001), Ubaidi (2015) dan Kelemenis dan Askounis (2010).

Kriteria pengalaman kerja digunakan untuk mengetahui keterampilan yang dimiliki seseorang. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013) dan Kelemenis dan Askounis (2010).

Kriteria jenis kelamin digunakan untuk mengetahui profil seseorang apakah profilnya sesuai dengan yang diharapkan atau tidak. Kriteria jenis kelamin diperoleh dari data jenis kelamin yang diisikan dalam formulir pendaftaran. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Blume dkk. (2010) dan Hough dkk. (2001).

Kriteria usia digunakan untuk mengetahui profil seseorang apakah profilnya sesuai dengan yang diharapkan atau tidak. Kriteria usia diperoleh dari data tanggal lahir yang diisikan dalam formulir pendaftaran. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Rouyendegh dan Erkan (2013), El-santawy dkk. (2012), Blume dkk. (2010), Robertson dan Smith (2001), Güngör dkk. (2009), Safrizal dan Tanti (2015), Hough dkk. (2001) dan Vinchur dkk. (1998).

Kriteria kefasihan bahasa asing digunakan untuk mengetahui kemampuan bahasa asing seseorang. Jika pelatihan menggunakan bahasa asing, dapat membantu dalam berkomunikasi. Kriteria ini mereferensi dari Noe dkk. (2012), Bogdanovic dan Miletic (2014), Safrizal dan Tanti (2015), Hough dan Oswald (2000) dan Chen dkk. (2013).

Kriteria motivasi digunakan untuk mengetahui alasan seseorang mengikuti pelatihan. Motivasi yang baik akan mempengaruhi usaha untuk mencapai tujuannya. Kriteria ini mereferensi dari Noe dkk. (2012), Ivancevich dan Konopaske (2013), Blume dkk. (2010), Güngör dkk. (2009) dan Ubaidi (2015).

Kriteria mental digunakan untuk mengetahui kesehatan mental seseorang. Kriteria ini mereferensi Noe dkk. (2012) dan Ivancevich dan Konopaske (2013).

Kriteria pertimbangan keluarga digunakan untuk mengetahui latar belakang keluarga seseorang. Kriteria ini mereferensi dari Noe dkk. (2012).

Kriteria ras dan etnis digunakan untuk mengetahui asal ras dan etnis seseorang. Kriteria ini mereferensi dari Ivancevich dan Konopaske (2013), Hough dkk. (2001) dan Hough dan Oswald (2000).

Kriteria status perkawinan digunakan untuk mengetahui status perkawinan seseorang. Status perkawinan seseorang pada bidang keilmuan tertentu dapat menjadi pertimbangan. Kriteria ini mereferensi dari Ivancevich dan Konopaske (2013).

Kriteria karakteristik fisik digunakan untuk mengetahui karakteristik fisik seseorang, misal: tinggi badan, berat badan, rambut, dll. Kriteria ini mereferensi dari Ivancevich dan Konopaske (2013) dan Safrizal dan Tanti (2015).

Kriteria kemampuan kognitif digunakan untuk mengetahui kemampuan kognitif seseorang, yang dapat diukur dengan kemampuan verbal dan matematika. Kriteria ini mereferensi dari Noe dkk. (2012) dan Ivancevich dan Konopaske (2013).

Kriteria keterampilan komputer digunakan untuk mengukur kemampuan seseorang dalam hal keterampilan komputer. Kriteria ini mereferensi dari Bogdanovic dan Miletic (2014) dan Güngör dkk. (2009).

Kriteria penampilan digunakan untuk menilai penampilan seseorang, misal: kerapian berpakaian, penampilan yang sopan dan tidak berlebihan, dll. Kriteria ini mereferensi dari Güngör dkk. (2009) dan Safrizal dan Tanti (2015).

Kriteria keterampilan komunikasi digunakan untuk mengetahui kemampuan berkomunikasi seseorang. Komunikasi yang baik akan mempermudah untuk menangkap ilmu dalam pelatihan. Kriteria ini mereferensi dari Bogdanovic dan Miletic (2014), Kumar dkk. (2013), Kelemenis dan Askounis (2010) dan Chen dkk. (2013).

Kriteria percaya diri digunakan untuk mengetahui tingkat kepercayaan diri seseorang. Kriteria ini mereferensi dari Rouyendegh dan Erkan (2013).

Kriteria kemantapan emosional digunakan untuk mengetahui kemantapan emosional seseorang. Kriteria ini mereferensi dari Vinchur dkk. (1998), Chen dkk. (2013) dan Barrett dkk. (2003).

Kriteria komitmen digunakan untuk mengetahui komitmen seseorang apakah setelah dinyatakan lulus, seseorang tidak akan mengundurkan diri dan serius mengikuti pelatihan. Kriteria ini mereferensi dari Ubaidi (2015).

Kriteria keuangan digunakan untuk mengetahui kondisi ekonomi seseorang. Kriteria ini mereferensi dari Noe dkk. (2012).

Kriteria potensi digunakan untuk mengukur potensi. Potensi merupakan kemampuan dan kualitas yang dimiliki oleh seseorang namun belum dipergunakan secara maksimal. Kriteria ini mereferensi dari Robertson dan Smith (2001) dan Vinchur dkk. (1998).

Kriteria kesungguhan digunakan untuk mengukur tingkat kesungguhan seseorang. Kesungguhan dapat tercermin dari apakah seseorang mengikuti semua proses seleksi dengan baik, datang tepat waktu, sesuai persyaratan, dll. Kriteria ini mereferensi dari Blume dkk. (2010), Robertson dan Smith (2001), Hough dan Oswald (2000), Vinchur dkk. (1998) dan Barrett dkk. (2003).

Kriteria kesan baik digunakan untuk mengetahui apakah seseorang memberikan kesan yang baik atau tidak. Kesan baik dapat diperoleh dari sikap, penampilan, sopan, santun yang baik. Kriteria ini mereferensi dari Barrett dkk. (2003).

**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. Menurut Bogdanovic dan Miletic (2014), AHP memberikan tingkat koherensi, korelasi, konsistensi dan akurasi bobot kriteria yang lebih tinggi daripada bobot yang ditentukan berdasarkan intuisi atau pengetahuan ahli. Hal ini karena AHP melakukan proses uji konsistensi, sehingga bobot yang dihasilkan konsisten.

Namun, dalam AHP belum dijelaskan cara menentukan kriteria, sehingga digunakan metode *Modified Delphi* yang dibangun oleh Afshari dkk. (2012) 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.

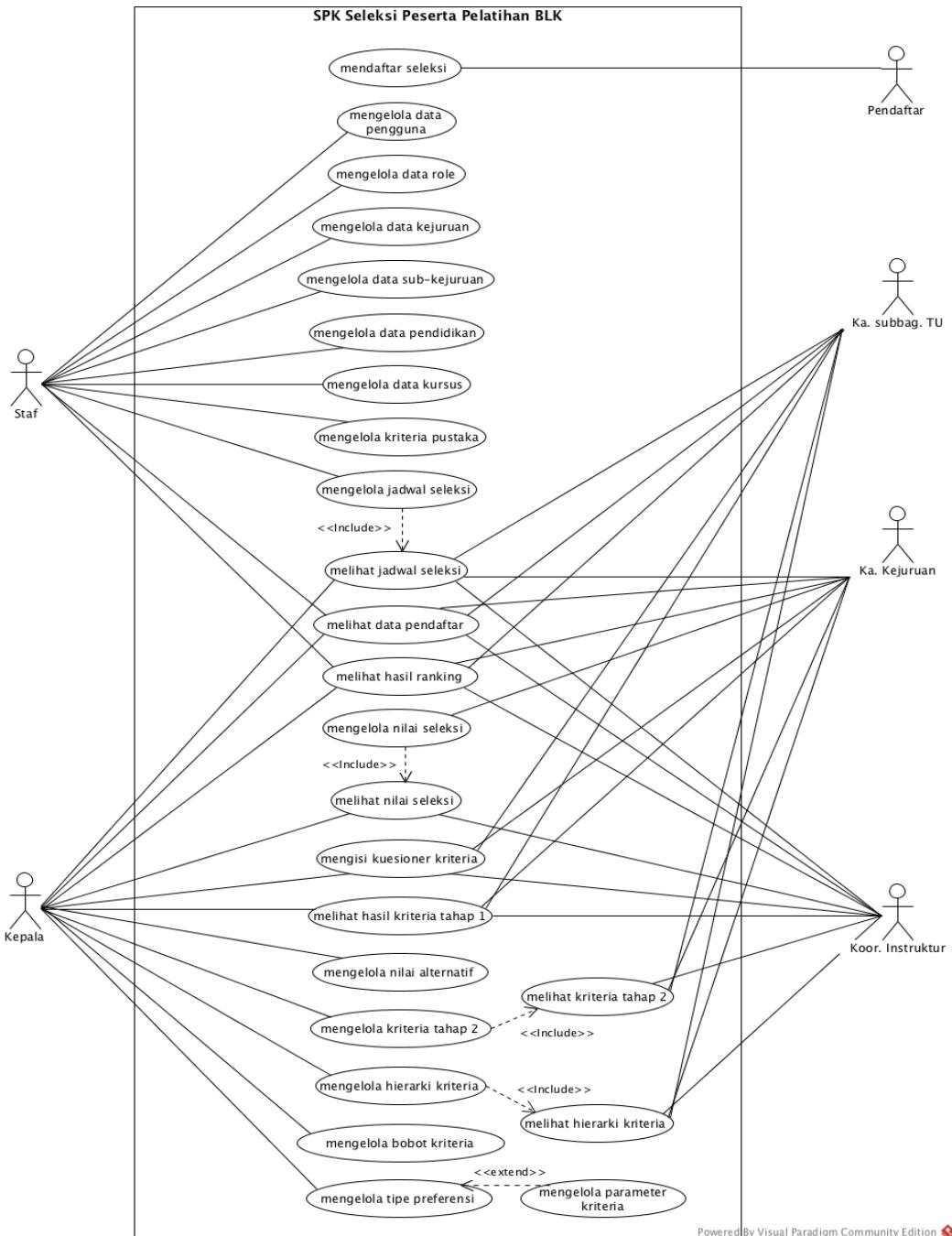
Salehi (2016), Lemantara dan Setiawan (2013) dan Mousavi dkk. (2013) merekomendasikan penggabungan metode AHP dengan metode MADM lainnya, karena penggabungan metode menghasilkan kualitas pemeringkatan yang baik. Metode yang digunakan yaitu PROMETHEE untuk menentukan peringkat terhadap alternatif. PROMETHEE dipilih karena setiap kriteria ditentukan fungsi preferensinya, sehingga setiap kriteria dapat dievaluasi secara berbeda dan memperoleh keputusan yang lebih baik (ÖZTÜRK, 2013). 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* menggambarkan interaksi antara aktor dengan sistem yang dibangun. Penjelasan *use case* ditunjukkan pada Tabel 4.4. Penjelasan *use case* mencakup deskripsi dari setiap *use case* yang terdiri dari aktor dan interaksi yang dilakukannya.

Gambar 4.1 *Use case diagram sistem*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.

**Tabel 4.4 (lanjutan)**

No.	<i>Use Case</i>	Deskripsi
		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 <i>role</i>	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 subkejuruan	Staf mengelola data subkejuruan yang meliputi lihat semua, 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

**Tabel 4.4 (lanjutan)**

No.	<i>Use Case</i>	Deskripsi
		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 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.

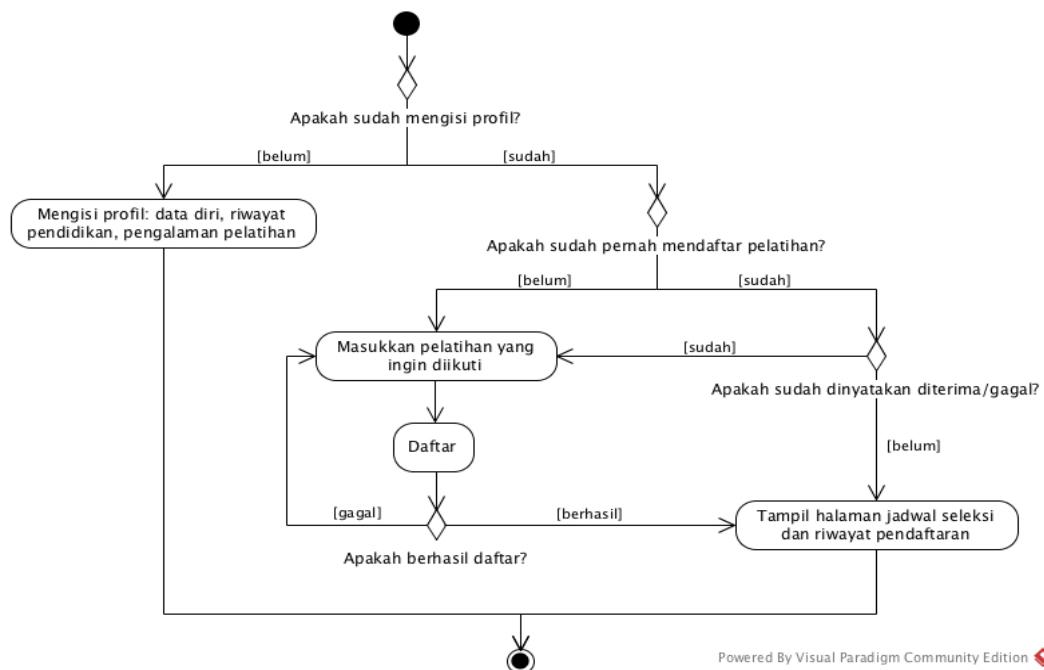
#### 4.4 Perancangan Sistem

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, sehingga yang memiliki kemiripan aliran aktivitas hanya digambarkan dalam satu *activity diagram* saja. Berikut ini *activity diagram* yang menggambarkan aliran aktivitas dalam sistem pada bagian implementasi pemodelan *Modified Delphi-AHP-PROMETHEE*.

### 1. *Activity diagram* mendaftar seleksi

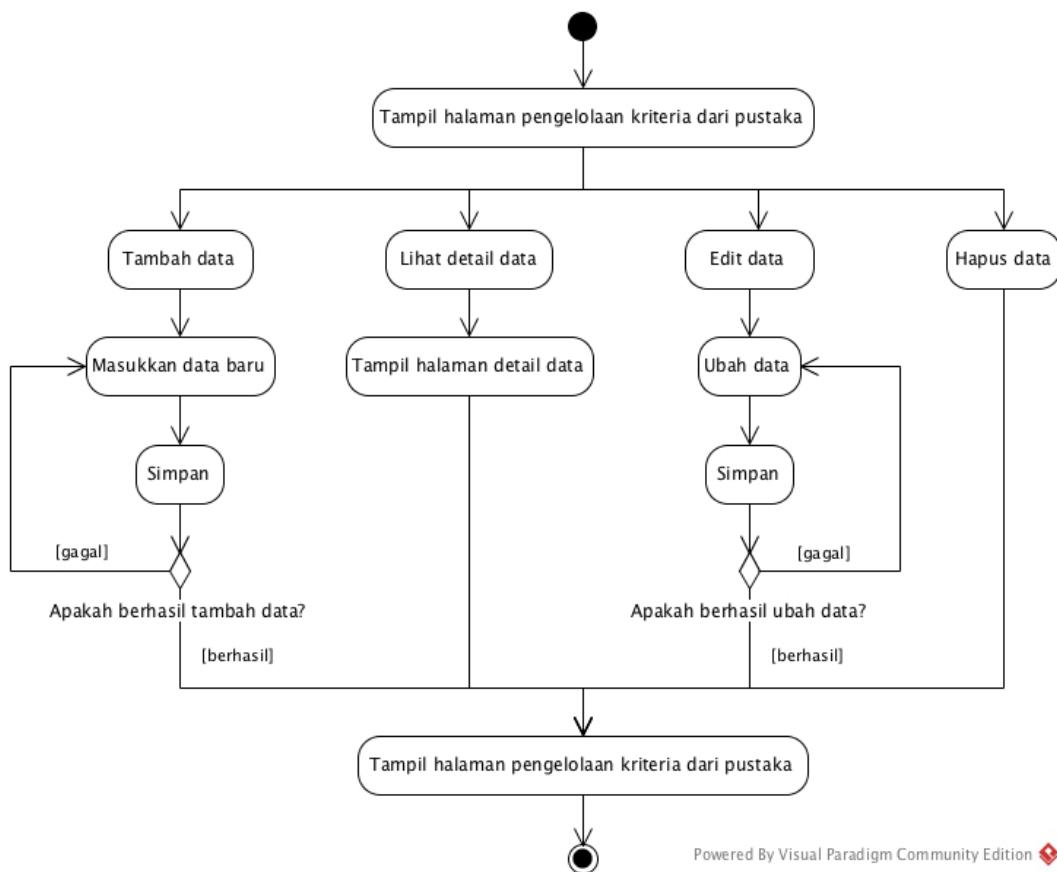
*Activity diagram* mendaftar seleksi menggambarkan aliran aktivitas untuk mendaftar pelatihan yang ingin diikuti. Aktivitas diawali dengan percabangan, apakah sudah mengisi profil. Jika belum mengisi profil, maka diwajibkan untuk mengisi profil yang terdiri dari data diri, riwayat pendidikan dan pengalaman pelatihan. Jika sudah mengisi profil, maka dilakukan pengecekan lagi, apakah sudah pernah mendaftar pelatihan. Jika belum pernah mendaftar pelatihan, maka diijinkan untuk mendaftar pelatihan. Jika sudah pernah mendaftar pelatihan, maka dilakukan pengecekan apakah sudah dinyatakan gagal/diterima. Jika sudah dinyatakan gagal/diterima, maka diijinkan untuk mendaftar pelatihan lagi. Namun jika belum dinyatakan gagal/diterima, maka tidak diijinkan untuk mendaftar pelatihan. *Activity diagram* mendaftar seleksi ditunjukkan pada Gambar 4.2.



Gambar 4.2 *Activity diagram* mendaftar seleksi

## 2. *Activity diagram* mengelola kriteria pustaka

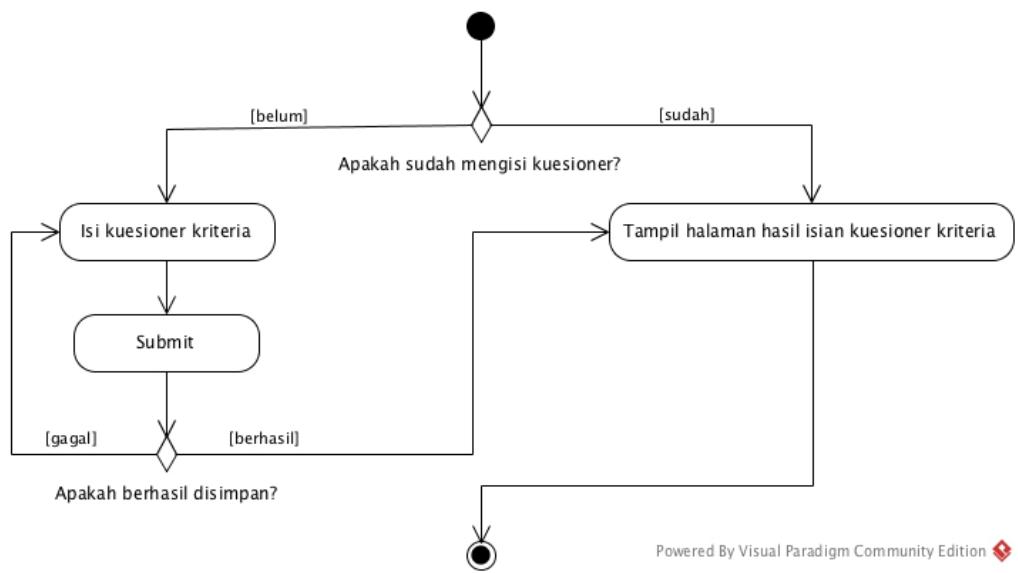
*Activity diagram* mengelola kriteria pustaka menggambarkan aliran aktivitas untuk pengelolaan kriteria yang bersumber dari studi pustaka, yang terdiri dari aktivitas paralel yaitu menambah data kriteria baru, melihat detail data kriteria, mengubah data kriteria dan menghapus data kriteria. Aliran aktivitas ini mengimplementasikan metode *Modified Delphi* putaran pertama. *Activity diagram* mengelola kriteria pustaka ditunjukkan pada Gambar 4.3. *Activity diagram* yang memiliki kemiripan aliran aktivitas dengan *activity diagram* mengelola kriteria pustaka yaitu mengelola data pengguna, mengelola data *role*, mengelola data kejuruan, mengelola data subkejuruan, mengelola data pendidikan, mengelola data kursus, mengelola jadwal seleksi dan mengelola nilai seleksi.



Gambar 4.3 *Activity diagram* mengelola kriteria pustaka

### 3. *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 4.4.



**Gambar 4.4** *Activity diagram* mengisi kuesioner kriteria

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

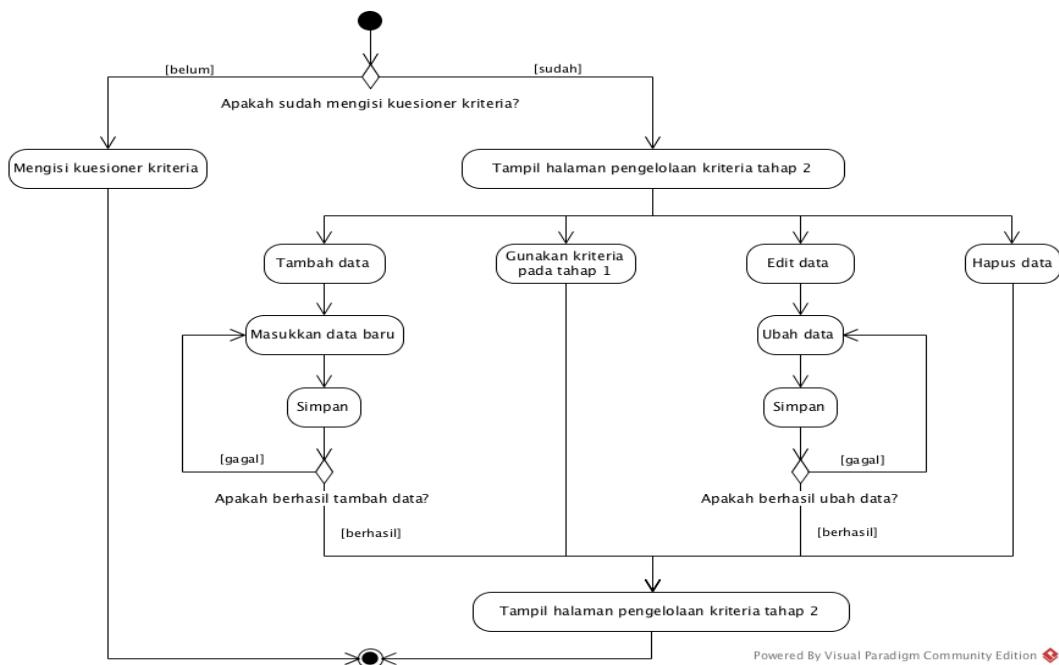
*Activity diagram* melihat hasil kriteria tahap 1 menggambarkan aliran aktivitas untuk melihat hasil kriteria tahap 1. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menjalankan aliran aktivitas untuk mengisi 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 4.5.



**Gambar 4.5 Activity diagram melihat hasil kriteria tahap 1**

##### 5. Activity diagram mengelola kriteria tahap 2

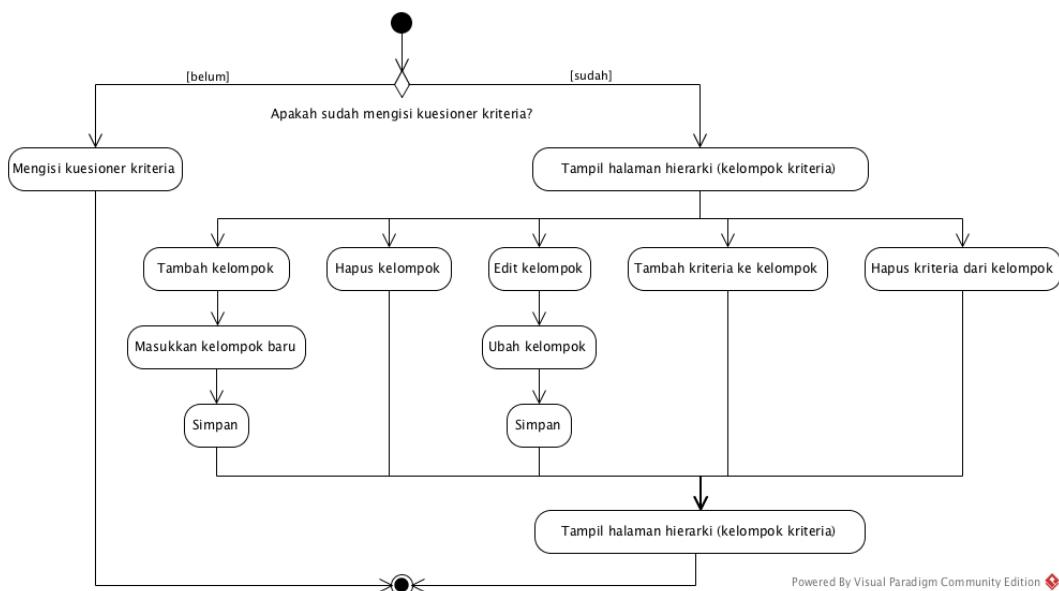
*Activity diagram* mengelola kriteria tahap 2 menggambarkan aliran aktivitas untuk pengelolaan kriteria tahap 2. Aktivitas diawali dengan percabangan, apakah sudah mengisi kuesioner kriteria atau belum. Jika belum mengisi kuesioner kriteria, maka akan menjalankan aliran aktivitas untuk mengisi kuesioner 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 4.6.



**Gambar 4.6 Activity diagram mengelola kriteria tahap 2**

### 6. *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 4.7.

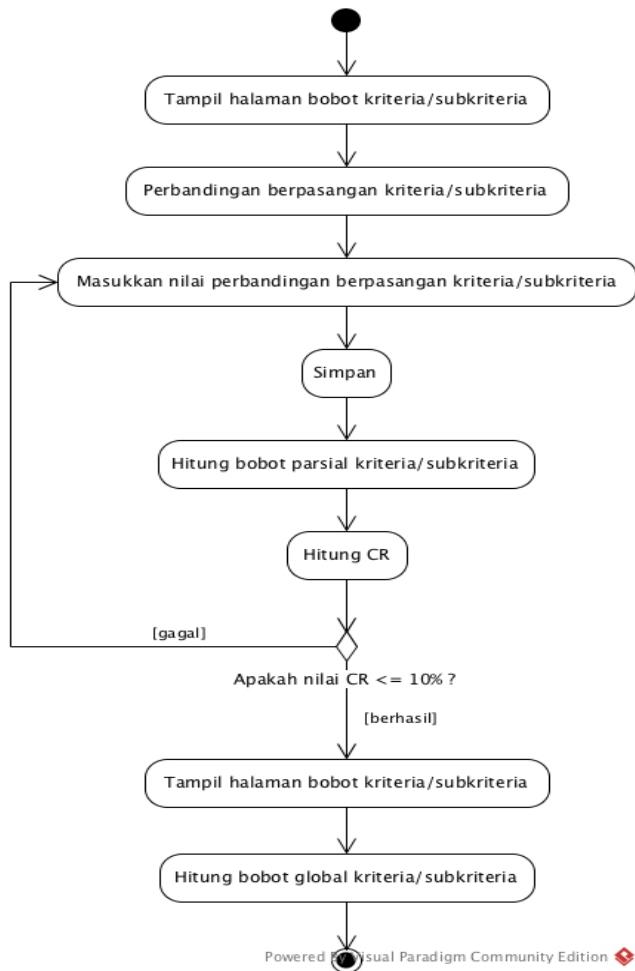


**Gambar 4.7** *Activity diagram* mengelola hierarki kriteria

### 7. *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

$CR \leq 10\%$ . Setelah diperoleh bobot parsial, maka dapat dihitung bobot global. Aliran aktivitas ini mengimplementasikan metode AHP. *Activity diagram* mengelola bobot kriteria ditunjukkan pada Gambar 4.8.

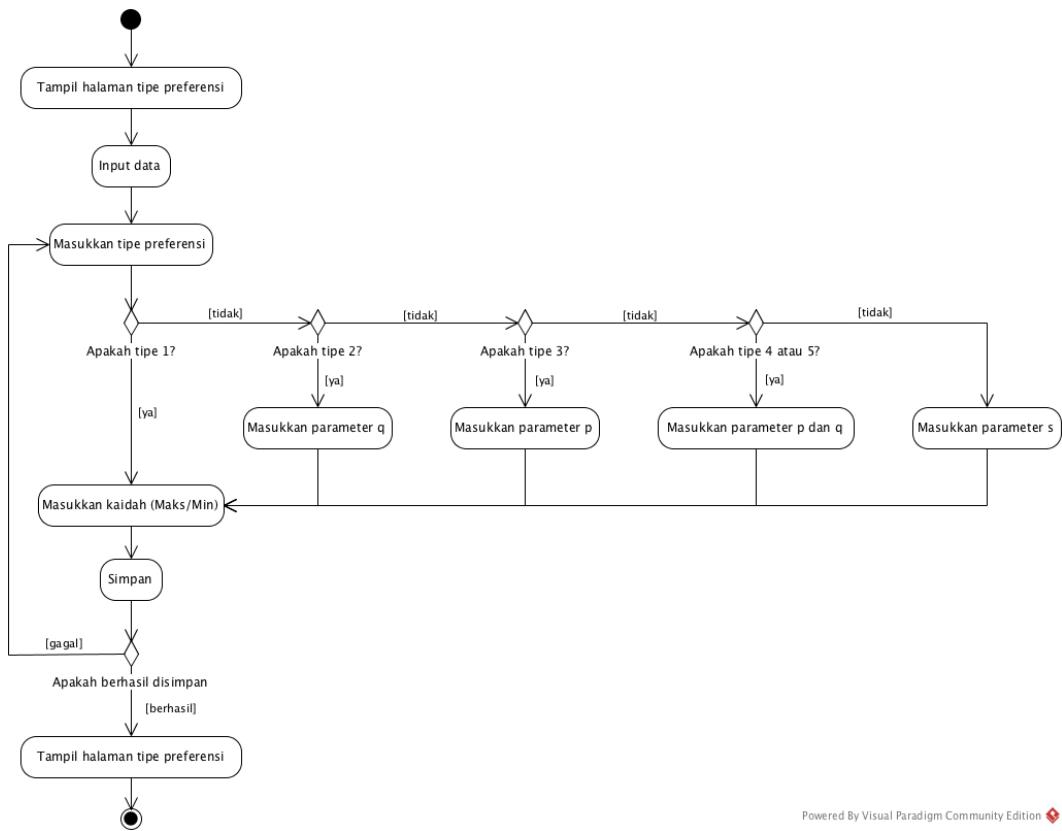


**Gambar 4.8 Activity diagram mengelola bobot kriteria**

#### 8. *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 4.9.

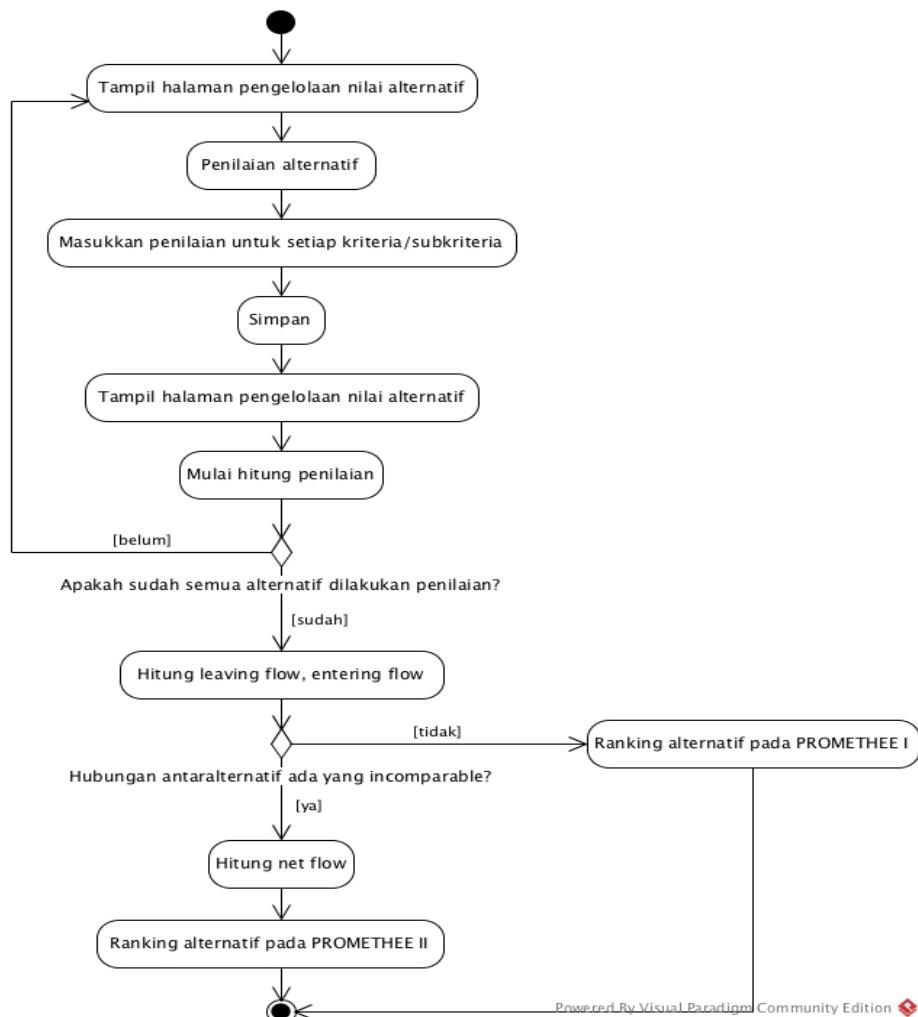


**Gambar 4.9 Activity diagram mengelola tipe preferensi**

#### 9. *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

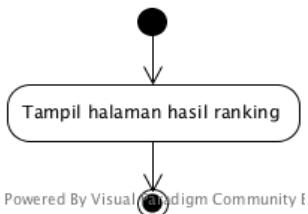
mengimplementasikan metode PROMETHEE. *Activity diagram* mengelola nilai alternatif ditunjukkan pada Gambar 4.10.



**Gambar 4.10** *Activity diagram* mengelola nilai alternatif

#### 10. *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 4.11. *Activity diagram* yang memiliki kemiripan aliran aktivitas dengan *activity diagram* melihat hasil ranking yaitu melihat data pendaftar.



**Gambar 4.11 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.

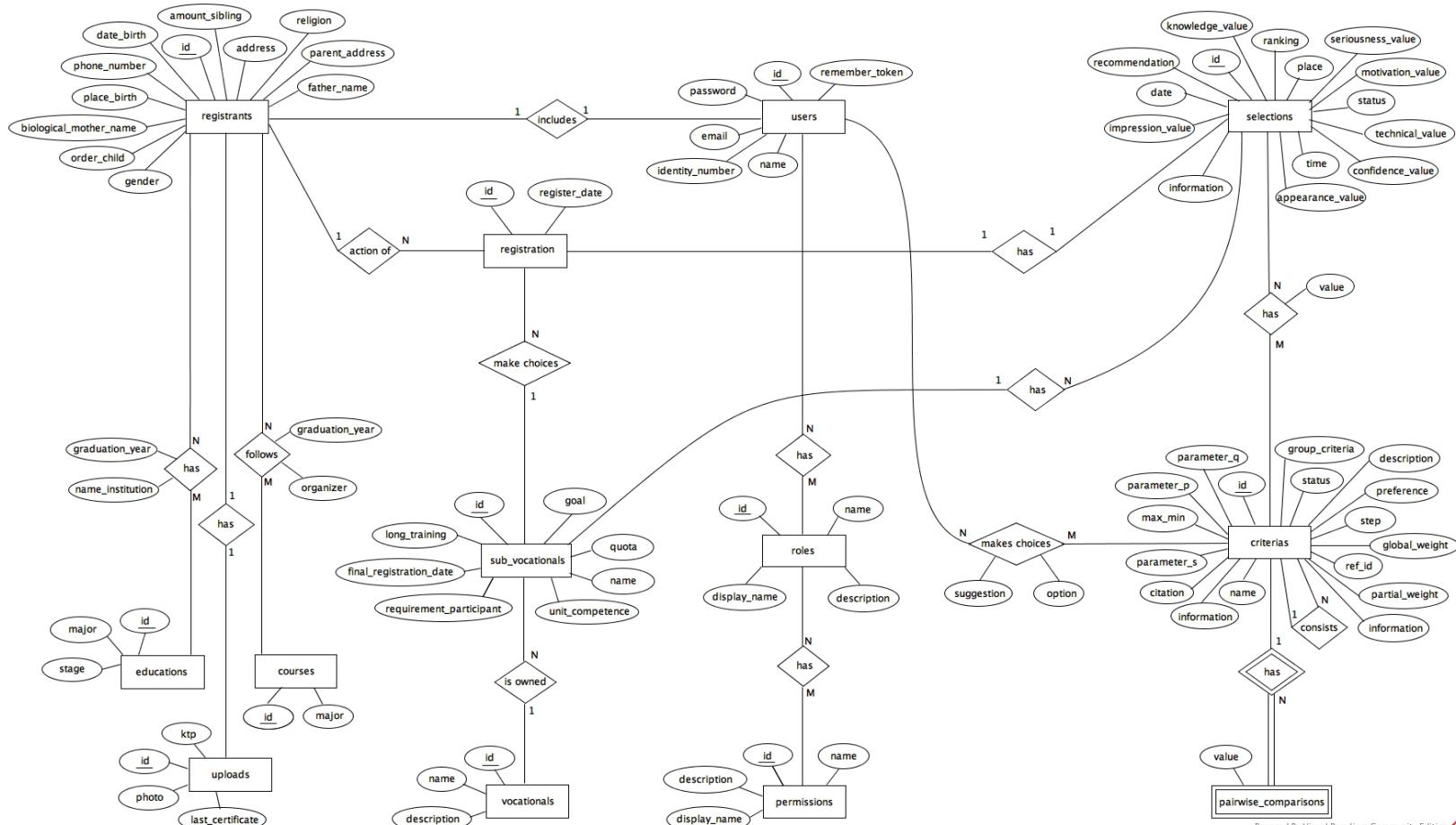
##### 4.5.1 ERD (*Entity Relationship Diagram*)

ERD digunakan untuk memodelkan kebutuhan data dari suatu sistem yang menggambarkan hubungan antara entitas dan relasi di antara entitas tersebut. Entitas dalam ERD ini yaitu users, registrants, educations, uploads, courses, registration, sub\_vocationals, vocationals, roles, permissions, selections, criterias dan pairwise\_comparisons. Entitas yang berhubungan langsung dengan implementasi metode *Modified Delphi-AHP-PROMETHEE* yaitu criterias, pairwise\_comparisons, selections dan users. 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 4.12.



Gambar 4.12 Entity Relationship Diagram (ERD)

Powered By Visual Paradigm Community Edition

#### 4.5.2 Implementasi basis data

Berdasarkan ERD pada Gambar 4.12, dapat diimplementasi ke dalam tabel basis data yang ditunjukkan pada Gambar 4.13. 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

**Tabel 4.6 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
last_certificate	varchar(255)	Menyimpan ijazah terakhir
created_at	timestamp	Menyimpan tanggal saat data dibuat
updated_at	timestamp	Menyimpan tanggal saat data diubah

### 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**

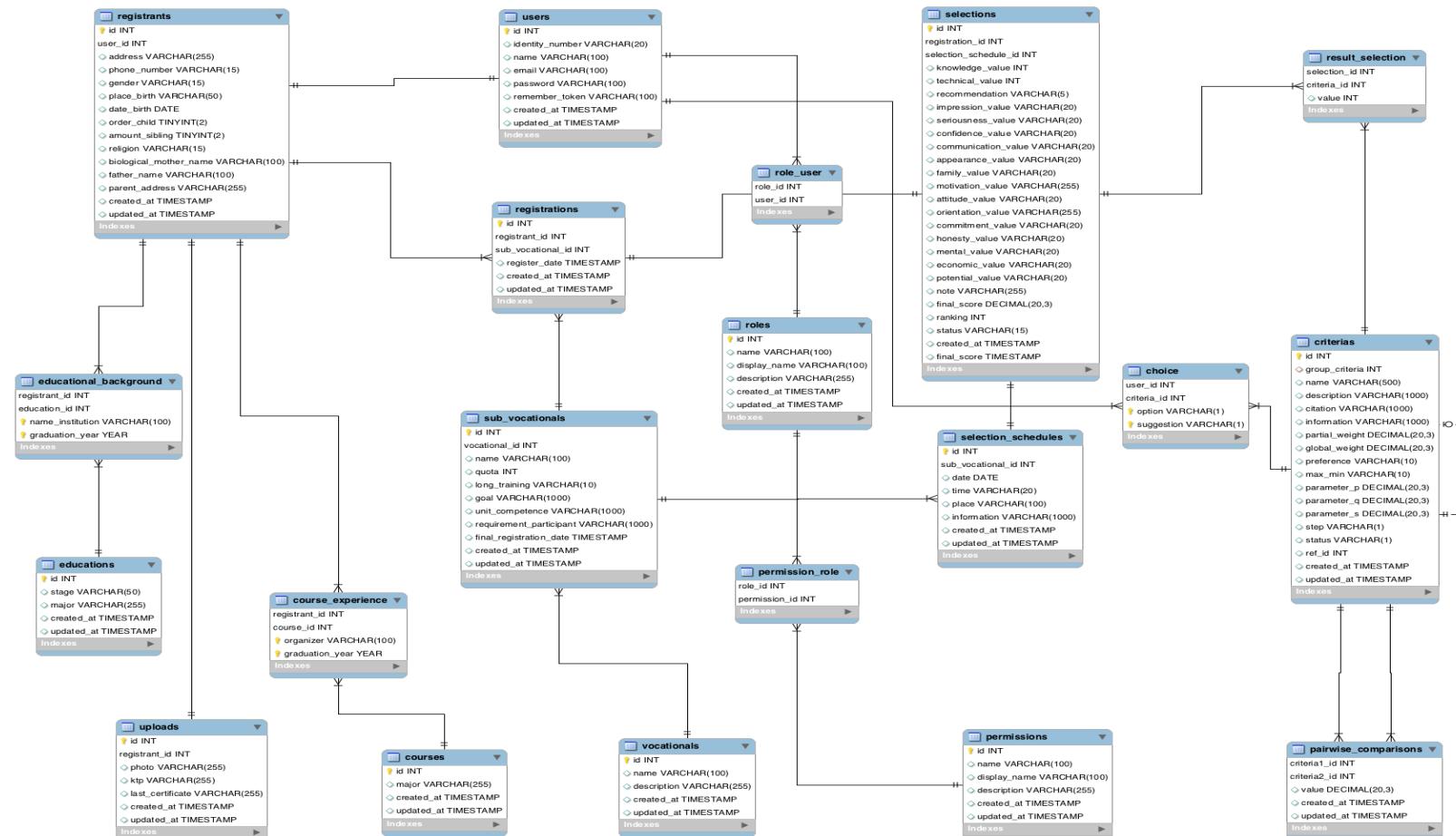
<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id pendidikan dan merupakan <i>primary key</i>
stage	varchar(50)	Menyimpan tingkat pendidikan
major	varchar(255)	Menyimpan jurusan pendidikan
created_at	timestamp	Menyimpan tanggal saat data pendidikan dibuat
updated_at	timestamp	Menyimpan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

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
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



Gambar 4.13 Implementasi basis data

### 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**

Nama Kolom	Tipe Data	Keterangan
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**

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

**Tabel 4.11 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
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**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
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>
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**

Nama Kolom	Tipe Data	Keterangan
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**

Nama Kolom	Tipe Data	Keterangan
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**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
id	int	Menyimpan id pendaftaran dan merupakan <i>primary key</i>
registrant_id	int	Menyimpan id pendaftar dan merupakan <i>foreign key</i>
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.
final_score	decimal(20,3)	Menyimpan hasil penilaian seleksi
ranking	int	Menyimpan peringkat berdasarkan hasil penilaian seleksi
status	varchar(15)	Menyimpan status penilaian. Berikut ini penjelasan isian status: Kosong: sedang proses penilaian

**Tabel 4.20 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
		Selesai: penilaian selesai dilakukan 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 1: active

**Tabel 4.21 (lanjutan)**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
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>

**Tabel 4.23 (lanjutan)**

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

#### 20. Tabel pairwise\_comparisons

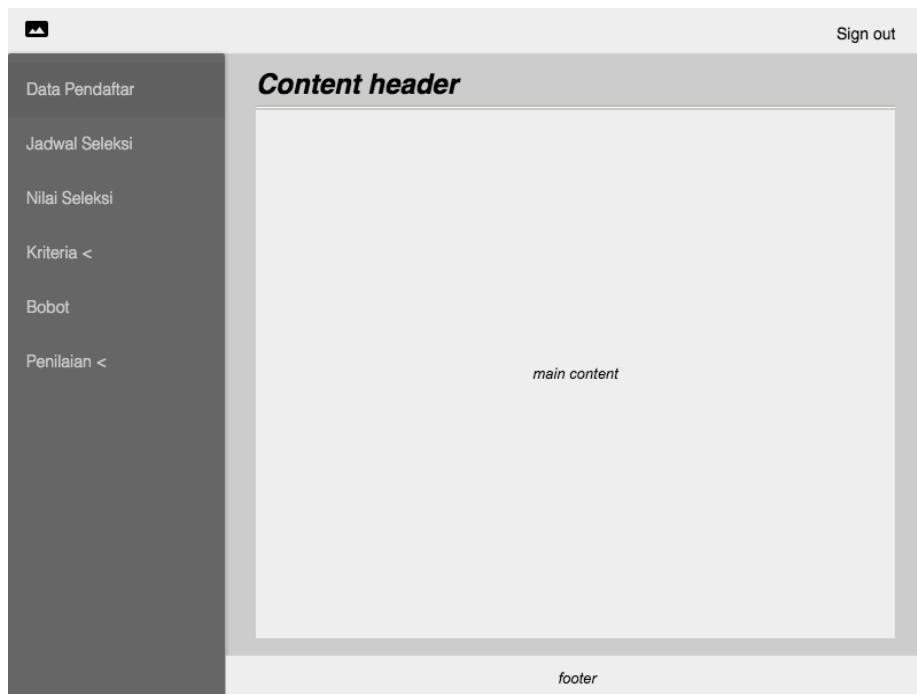
Tabel pairwise\_comparisons digunakan untuk menyimpan data hasil perbandingan berpasangan kriteria/subkriteria. Struktur tabel pairwise\_comparisons ditunjukkan pada Tabel 4.24.

**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 4.14.



**Gambar 4.14 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.

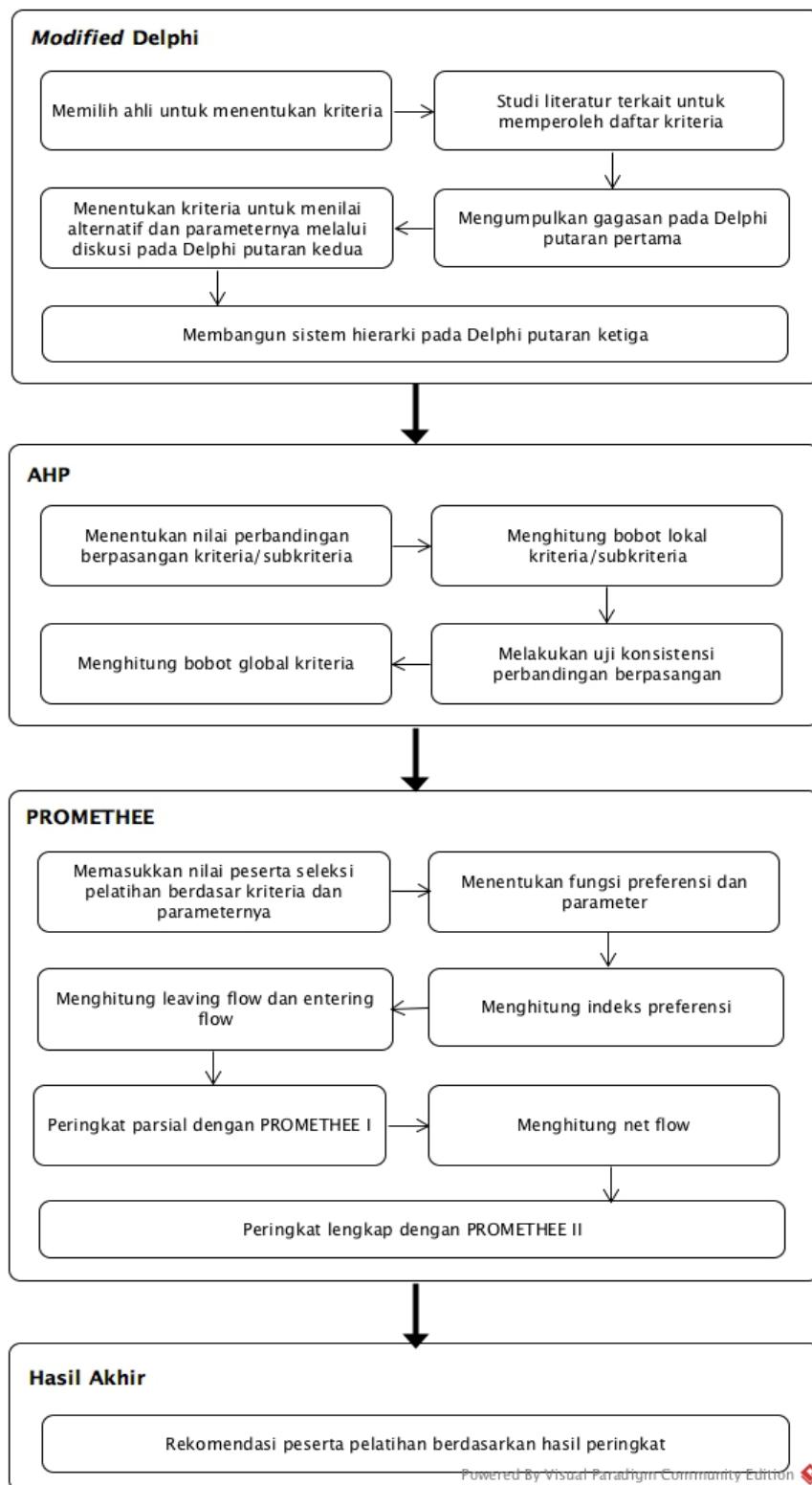
3. Menu untuk aktor kepala sub-bagian tata usaha yaitu Data Pendaftar, Jadwal Seleksi, Kriteria dan Penilaian. Menu Kriteria mempunyai submenu Kuesioner Kriteria, Hasil Kuesioner Tahap 1, Kriteria Tahap 2 dan Hierarki Kriteria. Menu Penilaian berisikan hasil dari penilaian yaitu berupa peringkat peserta 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 membuat keputusan untuk menentukan peserta seleksi yang diterima. Model keputusan secara keseluruhan ditunjukkan pada Gambar 4.15.



Gambar 4.15 Proses model keputusan seleksi peserta pelatihan

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.
  - 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  $j = 1, 2, \dots, n$ .

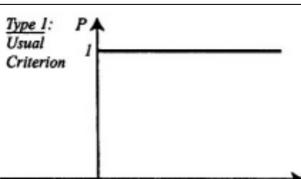
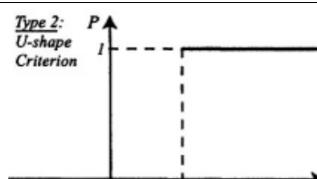
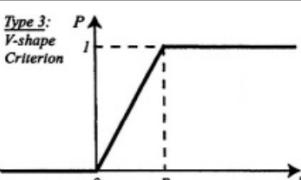
$$w_i = \sum_{(i=1)}^n a'_{ij} / n \quad (4.3)$$

dimana  $w_i$  merupakan hasil perbandingan berpasangan yang sudah dinormalisasi,  $n$  merupakan jumlah kriteria/subkriteria yang dibandingkan

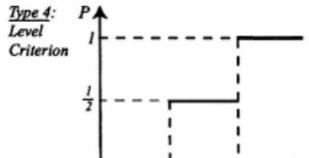
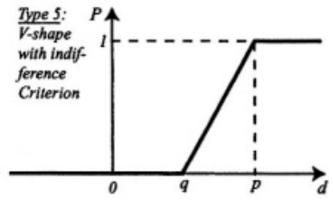
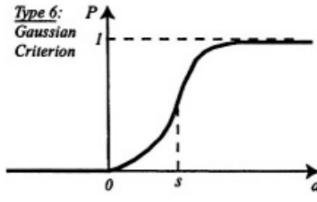
dan untuk semua  $j = 1, 2, \dots, n$ .

- 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
 Tipe 1: <i>Usual criterion</i>	$P(d) = \begin{cases} 0 & d \leq 0 \\ 1 & d > 0 \end{cases}$	-
 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

Tabel 4.25 (lanjutan)

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

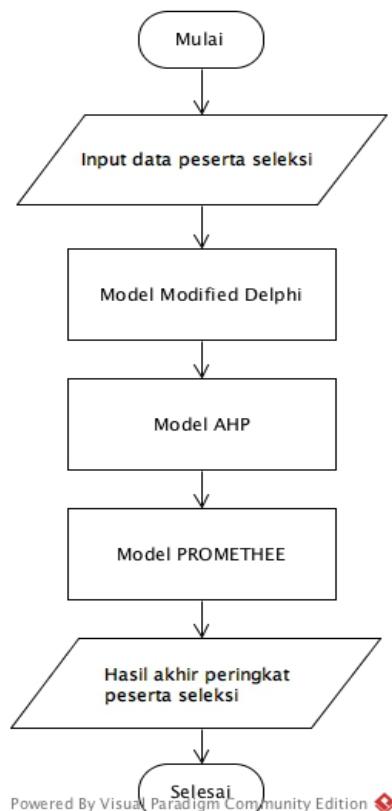
- g) Jika pada PROMETHEE I urutan peserta seleksi tidak diperoleh, maka dilanjutkan pada PROMETHEE II. PROMETHEE II memberikan peringkat lengkap peserta seleksi dengan mempertimbangkan nilai *net flow*.

#### 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 keseluruhan

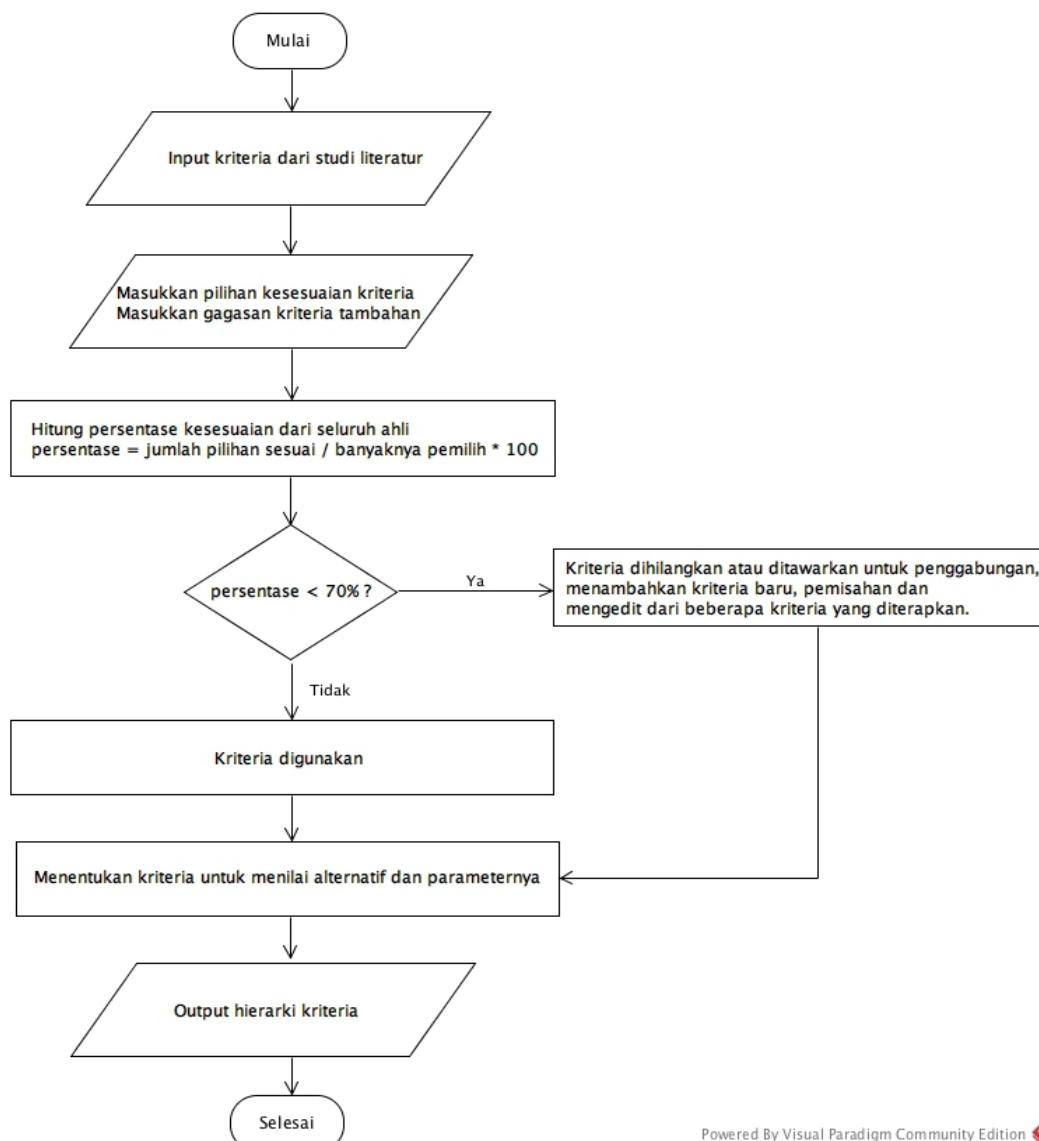
Gambar 4.16 menunjukkan diagram alir pemodelan secara keseluruhan dalam sistem. Masukan dalam pemodelan adalah data peserta seleksi yang digunakan dalam proses model *Modified Delphi*, AHP dan PROMETHEE. Keluaran sistem berupa hasil akhir peringkat peserta seleksi.



**Gambar 4.16 *Flowchart* model keseluruhan**

## 2. Flowchart model *Modified Delphi*

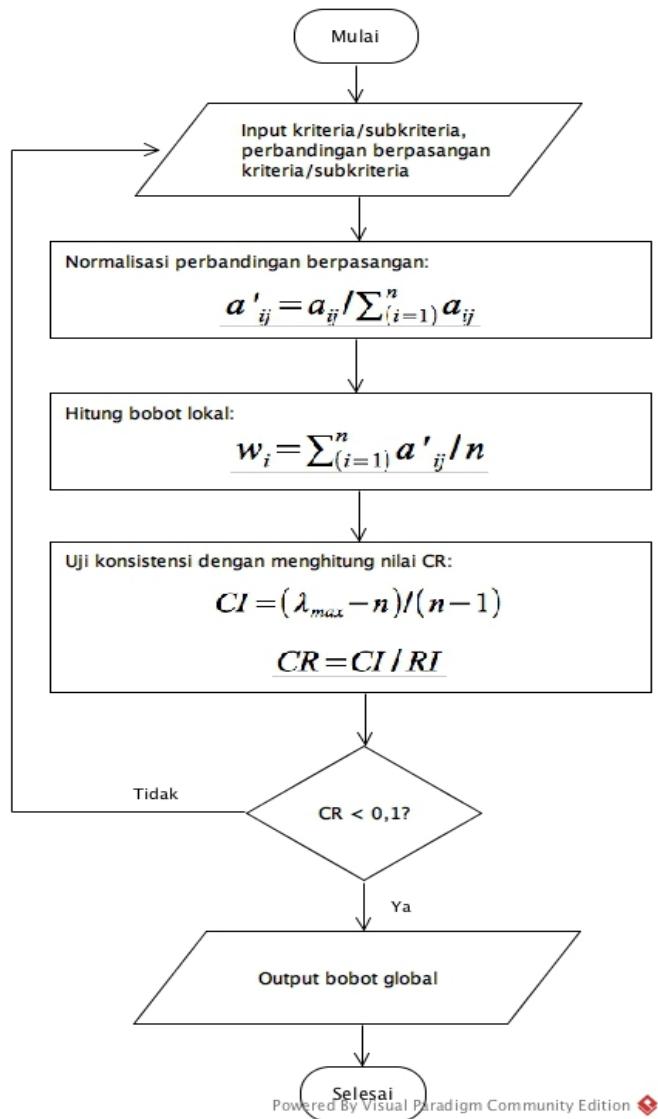
Gambar 4.17 menunjukkan diagram alir dalam pemodelan *Modified Delphi* sehingga menghasilkan kriteria/subkriteria yang siap digunakan untuk pemodelan AHP. Kriteria yang digunakan untuk pemodelan AHP yaitu faktor intelektual (FI), faktor personal (FP) dan faktor eksternal (FE). Kriteria FI terdiri dari subkriteria PP, PG, KT, PT dan KK. Kriteria FP terdiri dari subkriteria KS, US, PD, KO, PM, KB, IK, KJ, PO, ME, SI, MO dan RS. Kriteria FE terdiri dari subkriteria RK, PK dan PE.



**Gambar 4.17 Flowchart model *Modified Delphi***

### 3. Flowchart model AHP

Gambar 4.18 menunjukkan diagram alir dalam pemodelan AHP sehingga menghasilkan bobot kriteria/subkriteria yang siap digunakan untuk penilaian peserta seleksi pelatihan dalam pemodelan PROMETHEE. Bobot yang digunakan dalam pemodelan PROMETHEE dapat dilihat pada Tabel 4.40.

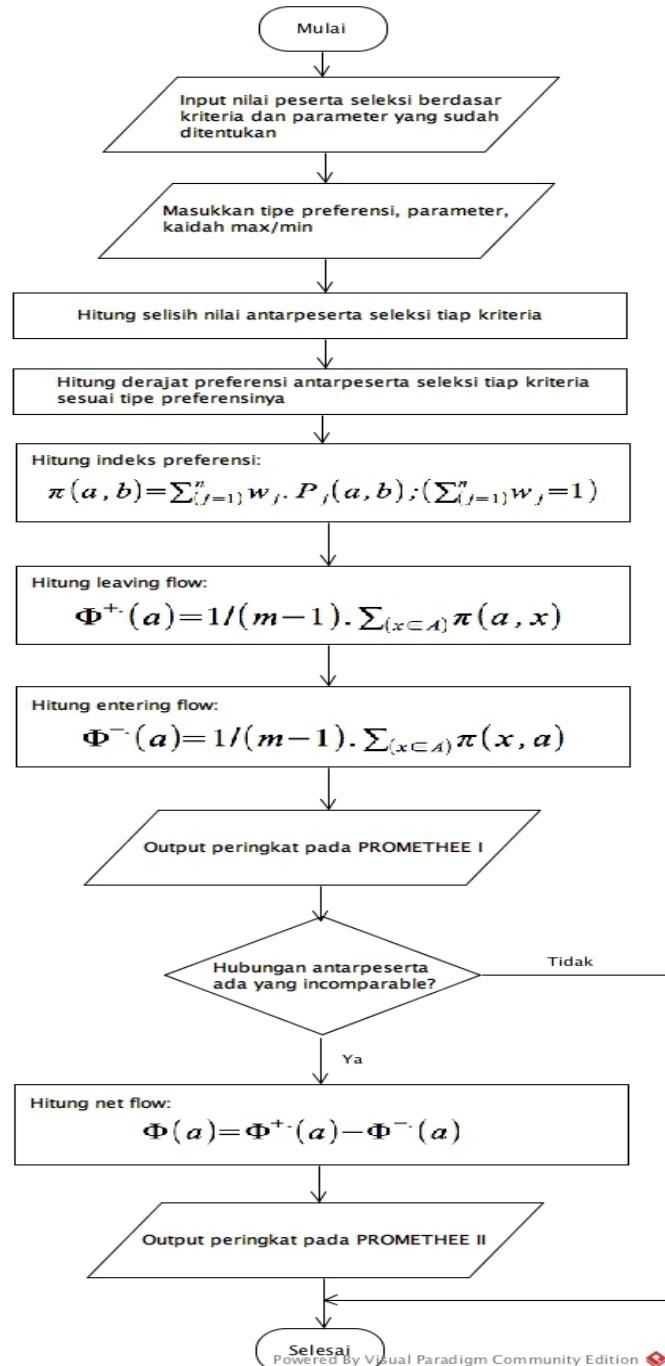


**Gambar 4.18 Flowchart model AHP**

### 4. Flowchart model PROMETHEE

Gambar 4.19 menunjukkan diagram alir dalam pemodelan PROMETHEE sehingga menghasilkan peringkat peserta seleksi pelatihan yang dapat digunakan

pengambil keputusan sebagai bahan pertimbangan memilih peserta seleksi. Berdasarkan hasil peringkat, peserta yang direkomendasikan untuk diterima adalah P2, P3, P4, P5, P6, P7, P9, P10, P11, P12, P13, P14, P16, P17, P18 dan P19.



Gambar 4.19 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 12 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.

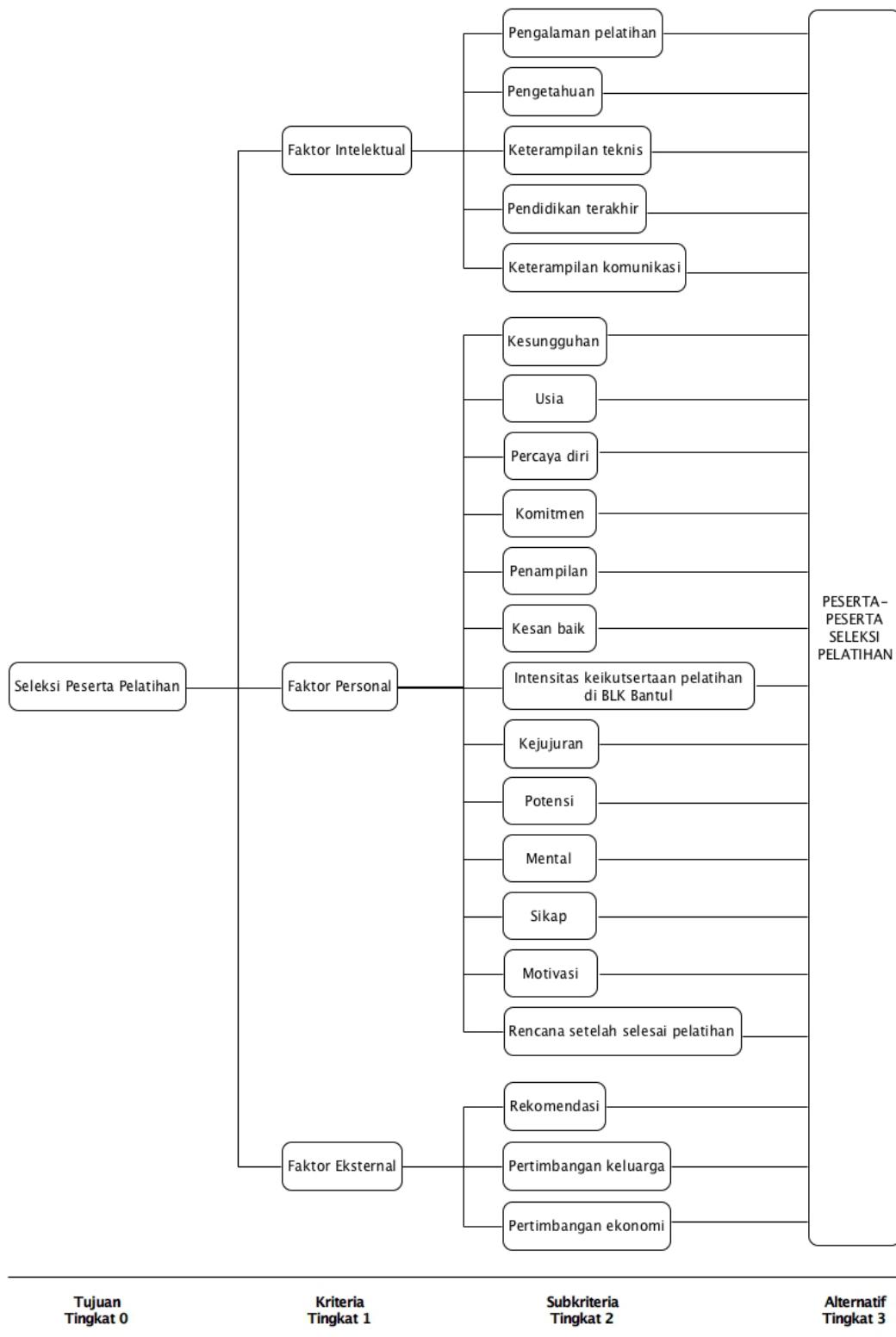
**Tabel 4.27 Hasil kriteria pada Delphi putaran kedua**

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
10	Usia	US	17 - 25 tahun: 4 26 - 34 tahun: 3

**Tabel 4.27 (lanjutan)**

No.	Kriteria	Kode	Parameter Penilaian
			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 4.20.



Gambar 4.20 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

Contoh perhitungan untuk jumlah kolom pada kriteria FI:

$$Jumlah = 1.000 + 2.000 + 0.500 = 3.500$$

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

Contoh perhitungan normalisasi perbandingan berpasangan kriteria FI:

$$FI' = FI / Jumlah = 1.000 / 3.500 = 0.286$$

Contoh perhitungan untuk jumlah baris pada kriteria FI:

$$Jumlah' = 0.286 + 0.273 + 0.333 = 0.892$$

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</b>
FI	0.297
FP	0.539
FE	0.164

Contoh perhitungan untuk bobot kriteria FI:

$$Bobot_{FI} = Jumlah' / n = 0.892 / 3 = 0.297$$

4. Uji konsistensi dengan menghitung CR.

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

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

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

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**

	<b>PP</b>	<b>PG</b>	<b>KT</b>	<b>PT</b>	<b>KK</b>
<b>PP</b>	1.000	1.000	0.500	0.250	0.500
<b>PG</b>	1.000	1.000	0.500	0.250	0.500
<b>KT</b>	2.000	2.000	1.000	0.500	1.000
<b>PT</b>	4.000	4.000	2.000	1.000	2.000
<b>KK</b>	2.000	2.000	1.000	0.500	1.000
<b>Jumlah</b>	10.000	10.000	5.000	2.500	5.000

Contoh perhitungan untuk jumlah kolom pada subkriteria PP:

$$Jumlah = 1.000 + 1.000 + 2.000 + 4.000 + 2.000 = 10.000$$

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**

	<b>PP</b>	<b>PG</b>	<b>KT</b>	<b>PT</b>	<b>KK</b>	<b>Jumlah'</b>
<b>PP</b>	0.100	0.100	0.100	0.100	0.100	0.500
<b>PG</b>	0.100	0.100	0.100	0.100	0.100	0.500
<b>KT</b>	0.200	0.200	0.200	0.200	0.200	1.000
<b>PT</b>	0.400	0.400	0.400	0.400	0.400	2.000
<b>KK</b>	0.200	0.200	0.200	0.200	0.200	1.000

Contoh perhitungan normalisasi perbandingan berpasangan subkriteria PP:

$$PP' = PP / Jumlah' = 1.000 / 10.000 = 0.100$$

Contoh perhitungan untuk jumlah baris pada subkriteria PP:

$$Jumlah' = 0.100 + 0.100 + 0.100 + 0.100 + 0.100 = 0.500$$

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</b>
PP	0.100
PG	0.100
KT	0.200
PT	0.400
KK	0.200

Contoh perhitungan untuk bobot subkriteria PP:

$$Bobot_{PP} = Jumlah' / n = 0.500 / 5 = 0.100$$

8. Uji konsistensi dengan menghitung CR.

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

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

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

Contoh perhitungan untuk jumlah kolom pada subkriteria IK:

$$\text{Jumlah} = 1+1+2+1+1+2+2+0.5+2+2+2+2+0.5 = 19$$

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

**Tabel 4.35 (lanjutan)**

	<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>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

Contoh perhitungan normalisasi perbandingan berpasangan subkriteria IK:

$$IK' = IK / Jumlah = 1.000 / 19.000 = 0.053$$

Contoh perhitungan untuk jumlah baris pada subkriteria IK:

$$Jumlah' = 0.053 + 0.067 + 0.052 + 0.053 + 0.052 + 0.052 + 0.066 + 0.050$$

$$+ 0.050 + 0.050 + 0.050 + 0.043 = 0.690$$

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</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
KO	0.100
PO	0.100
KS	0.100
KB	0.023

Contoh perhitungan untuk bobot subkriteria IK:

$$Bobot_{IK} = Jumlah' / n = 0.690 / 13 = 0.053$$

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**

	RK	PK	PE
RK	1.000	3.000	2.000
PK	0.333	1.000	0.500
PE	0.500	2.000	1.000
Jumlah	1.833	6.000	3.500

Contoh perhitungan untuk jumlah kolom pada subkriteria RK:

$$Jumlah = 1.000 + 0.333 + 0.500 = 1.833$$

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**

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

Contoh perhitungan normalisasi perbandingan berpasangan subkriteria RK:

$$RK' = RK / Jumlah = 1.000 / 1.833 = 0.546$$

Contoh perhitungan untuk jumlah baris pada subkriteria RK:

$$Jumlah' = 0.546 + 0.500 + 0.571 = 1.617$$

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
RK	0.539
PK	0.164
PE	0.297

Contoh perhitungan untuk bobot subkriteria RK:

$$Bobot_{IK} = Jumlah' / n = 1.617 / 3 = 0.539$$

16. Uji konsistensi dengan menghitung CR.

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

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

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

**Tabel 4.40 (lanjutan)**

<b>Kriteria</b>	<b>Subkriteria</b>	<b>Bobot global</b>
	KJ	0.057
	SI	0.029
	US	0.029
	MO	0.057
	ME	0.057
	PM	0.018
	PD	0.054
	KO	0.054
	PO	0.054
	KS	0.054
FE	KB	0.012
	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.

**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	0	1	4
22	P22	0	0	0	0	0	0	0	0	0	0	0	4	4	0	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	0	1	4
24	P24	0	0	0	0	0	0	0	0	0	0	0	4	4	0	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	0	1	4
26	P26	0	0	0	0	0	0	0	0	0	0	0	4	4	0	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	0	1	4
28	P28	0	0	0	0	0	0	0	0	0	0	0	4	2	0	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	0	1	4
30	P30	0	0	0	0	0	0	0	0	0	0	0	2	4	0	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	0	1	4
32	P32	0	0	0	0	0	0	0	0	0	0	0	4	4	0	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	-1	30	0	0	-2	0	0

2. Menentukan fungsi preferensi  $P(a,b)$  dan parameter untuk masing-masing kriteria. Fungsi preferensi ditunjukkan pada Tabel 4.43.

**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**

**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 2	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 2	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 2	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 2	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 2	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 2	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 2	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 2	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 2	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 0	0.0 0	0.0 6	0.0 0	0.1 0	0.0 2	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 0	0.0 1	0.0 0	0.0 6	0.0 0	0.1 0	0.0 2	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 2	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	Tipe 5	Tipe 5	Tipe 4	Tipe 4	Tipe 4	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	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 digunakan 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 mengundurkan 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**

Gambar 5.2 menunjukkan potongan *source code* untuk menampilkan daftar kriteria dari studi literatur. Pengelolaan kriteria hanya boleh diakses oleh role\_id = 1 yaitu staf. Jika role\_id selain 1 mengakses, maka diarahkan ke halaman profil pengguna. Variabel \$criteria pada baris 5-13 merupakan *query* yang digunakan untuk menampilkan kriteria dari model Criteria yang mempunyai

description-nya tidak sama dengan null, pada step = 1 dan satus = 1 serta bukan merupakan usulan. Daftar kriteria dari studi literatur disajikan dalam bentuk tabel yang mempunyai paginasi 10 pada setiap halamannya.

```

1 public function index(Request $request)
2 {
3     $role_id = Auth::user()->roleId();
4     if ($role_id == 1) {
5         $criterias = Criteria::select('*')
6             ->where('description', '!=', 'null')
7             ->where('step', '=', '1')
8             ->where('status', '=', '1')
9             ->whereNotIn('id', function($query){
10                 $query->select('criteria_id')
11                 ->from(with(new Choice)->getTable())
12                 ->where('suggestion', 1);
13             })->orderBy('id', 'DESC')->paginate(10);
14         return view('criterias.index', compact('criterias'))
15             ->with('i', ($request->input('page', 1) - 1) * 10);
16     } else {
17         return redirect()->route('profile_users.show');
18     }
19 }
```

**Gambar 5.2 Potongan *source code* untuk menampilkan daftar kriteria dari studi literatur**

Gambar 5.3 menunjukkan potongan *source code* untuk memproses masukan dari halaman tambah kriteria baru. Fungsi validate pada baris 3 digunakan untuk validasi masukan yaitu name, description dan citation harus diisi. Semua masukan disimpan dengan model Criteria. Masukan step dan status masing-masing diberikan nilai 1.

```

1 public function store(Request $request)
2 {
3     $this->validate($request, [
4         'name' => 'required',
5         'description' => 'required',
6         'citation' => 'required',
7     ]);
8     $input = $request->all();
9     $input['step'] = '1';
10    $input['status'] = '1';
11    Criteria::create($input);
12    return redirect()->route('criterias.index')
13        ->with('success', 'Kriteria berhasil dibuat');
14 }
```

**Gambar 5.3 Potongan *source code* untuk memproses masukan dari halaman tambah kriteria studi literatur**

Gambar 5.4 menunjukkan potongan *source code* untuk menampilkan detail suatu kriteria. Varibel \$criteria pada baris 5 merupakan *query* untuk menampilkan data kriteria dari model Criteria dengan \$id sesuai masukan.

```

1 public function show($id)
2 {
3     $role_id = Auth::user()->roleId();
4     if ($role_id == 1) {
5         $criteria = Criteria::find($id);
6         return view('criterias.show', compact('criteria'));
7     } else {
8         return redirect()->route('profile_users.show');
9     }
10 }
```

**Gambar 5.4 Potongan *source code* untuk menampilkan detail kriteria studi literatur**

Gambar 5.5 menunjukkan potongan *source code* untuk memproses perubahan dari halaman edit kriteria. Fungsi validate pada baris 3 digunakan untuk validasi data yaitu name, description dan citation harus disi. Semua perubahan untuk kriteria dengan \$id sesuai masukan di-*update* ke dalam model Criteria.

```

1 public function update(Request $request, $id)
2 {
3     $this->validate($request, [
4         'name' => 'required',
5         'description' => 'required',
6         'citation' => 'required',
7     ]);
8     Criteria::find($id)->update($request->all());
9     return redirect()->route('criterias.index')
10    ->with('success', 'Kriteria berhasil diedit');
11 }
```

**Gambar 5.5 Potongan *source code* untuk memproses perubahan dari halaman edit kriteria studi literatur**

Gambar 5.6 menunjukkan potongan *source code* untuk menghapus suatu kriteria. Sebelum menghapus, dilakukan pengecekan dahulu, apakah kriteria sudah dilakukan evaluasi penilaian kesesuaian kriteria oleh tim penilai. Jika belum dilakukan evaluasi, maka kriteria dengan \$id sesuai yang dipilih dapat langsung dihapus menggunakan fungsi delete. Namun jika sudah dilakukan evaluasi, maka kriteria tidak bisa dihapus dan memunculkan pesan kesalahan.

```

1 public function destroy($id)
2 {
3 $choice = Choice::where('criteria_id', '=', $id)->first();
4 if ($choice == null) {
5 Criteria::find($id)->delete();
6 return redirect()->route('criterias.index')
7 ->with('success','Kriteria berhasil dihapus');
8 } else {
9 return redirect()->route('criterias.index')
10 ->with('failed','Kriteria tidak bisa dihapus karena sudah ada penilaian kesesuaian
11 kriteria oleh tim penilai');
12 }
13 }

```

**Gambar 5.6 Potongan *source code* untuk menghapus kriteria studi literatur**

## 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 5.7.

No	Kriteria	Penjelasan Kriteria	Sumber Pustaka	Pilihan
1	Kesan baik	Kriteria yang digunakan untuk mengetahui apakah pendaftar memberikan kesan 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 yang tergambar dari sejak pendidik 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 keperimaninan, 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 dilakukan 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 5.7 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 5.8.

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 tergambar 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)	Sesuai
3	Potensi	Kriteria yang digunakan untuk mengukur potensi. Potensi itu kemampuan dan kualitas yang dimiliki oleh seseorang namun belum dipungkarkan secara maksimal. Data diperoleh dari pengukuran potensi diri yang dilakukan melalui diri sendiri ( <i>self assessment</i> ), 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)	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 Questions).	(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 disikapi 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)	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 5.8 Implementasi hasil isian kuesioner kriteria

Gambar 5.9 menunjukkan potongan *source code* untuk menampilkan hasil isian kuesioner. Variabel \$role\_id digunakan untuk memperoleh role\_id pengguna. Halaman ini hanya boleh diakses oleh kepala, kepala sub-bagian tata usaha, koordinator instruktur dan kepala kejuruan. Variabel \$user digunakan untuk mengetahui id pengguna. Variabel \$data digunakan untuk mengecek apakah pengguna yang bersangkutan sudah pernah mengisi kuesioner. Halaman ini hanya boleh diakses jika pengguna sudah mengisi kuesioner. Variabel \$data\_standart pada baris 17-23 merupakan *query* yang digunakan untuk menampilkan hasil penilaian kesesuaian oleh pengguna yang bersangkutan untuk kriteria yang berasal dari studi literatur. Variabel \$data\_suggestion pada baris 24-30 merupakan *query* yang digunakan untuk menampilkan kriteria tambahan oleh pengguna yang bersangkutan.

```

1 public function index()
2 {
3     $role_id = Auth::user()->roleId();
4     $user = User::find(Auth::user()->id);
5     $data = Choice::where('user_id', '=', $user->id)->first();
6     if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
7         if ($user->id == 1) {
8             return redirect()->route('questionnaire.create')
9             ->with(['failed', 'Maaf, peneliti tidak perlu mengisi kuesioner ini.
10            Apapun yang disubmit tidak akan tersimpan dalam database']);
11        }
12        if ($data == null) {
13            return redirect()->route('questionnaire.create');
14        } else {
15            $i = 0;
16            $j = 0;
17            $data_standart = Choice::select('choice.*', 'criterias.*')
18            ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
19            ->where('choice.suggestion', '=', '0')
20            ->where('choice.user_id', '=', $user->id)
21            ->where('criterias.step', '=', '1')
22            ->where('criterias.status', '=', '1')
23            ->orderBy('criterias.id', 'DESC')->get();
24            $data_suggestion = Choice::select('choice.*', 'criterias.*')
25            ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
26            ->where('choice.suggestion', '=', '1')
27            ->where('choice.user_id', '=', $user->id)
28            ->where('criterias.step', '=', '2')
29            ->where('criterias.status', '=', '1')
30            ->orderBy('criterias.id', 'ASC')->get();
31            return view('questionnaire.index', compact('data_standart', 'data_suggestion', 'i', 'j'));
32        }
33    } else {
34        return redirect()->route('profile_users.show');
35    }
36}

```

**Gambar 5.9 Potongan *source code* untuk menampilkan hasil isian kuesioner**

Gambar 5.10 menunjukkan potongan *source code* untuk memproses isian kuesioner. Isian kuesioner terdapat dua jenis, yaitu isian kesesuaian kriteria dari studi literatur dan isian kriteria tambahan sebagai masukan/gagasan dari pengguna. Isian kesesuaian kriteria dari studi literatur harus diisi semuanya yaitu dengan memilih opsi sesuai atau tidak sesuai. Isian kesesuaian kriteria dari studi literatur disimpan ke dalam model Choice, dapat dilihat pada baris 4-8. Isian kriteria tambahan bersifat opsional. Isian kriteria tambahan disimpan ke dalam dua model Criteria dan Choice. Pertama, disimpan ke model Criteria terlebih dahulu, yaitu terdiri dari name dan description sesuai dengan masukan pengguna, step = 2 dan status = 1. Kedua, disimpan ke model Choice, yaitu terdiri dari

user\_id merupakan id pengguna, criteria\_id merupakan id criteria pada langkah pertama, option = 1 dan suggestion = 1.

```

1 $input = $request->all();
2 if ($valid) {
3     foreach ($choices as $criterioid=>$option) {
4         $data["user_id"] = $user->id;
5         $data["criteria_id"] = $criterioid;
6         $data["option"] = $option;
7         $data["suggestion"] = 0;
8         Choice::create($data);
9     }
10    foreach ($optional as $optionalCriteria) {
11        $dataoptional["name"] = $optionalCriteria["criteria"];
12        $dataoptional["description"] = $optionalCriteria["description"];
13        $dataoptional["step"] = 2;
14        $dataoptional["status"] = 1;
15        $dataoptional["created_at"] = Carbon\Carbon::now(7)->toDateTimeString();
16        $dataoptional["updated_at"] = $dataoptional["created_at"];
17        $suggest = Criteria::create($dataoptional);
18        $optionalChoice["user_id"] = $user->id;
19        $optionalChoice["criteria_id"] = $suggest->id;
20        $optionalChoice["option"] = 1;
21        $optionalChoice["suggestion"] = 1;
22        Choice::create($optionalChoice);
23    }
24    return redirect()->route('questionnaire.index')
25    ->with('success', 'Selamat Anda berhasil mengisi kuesioner kriteria.
26 Data yang sudah diisikan tidak dapat diubah.');
27 } else {
28     return redirect()->route('questionnaire.create')
29     ->with('failed', 'Maaf! Semua pilihan kriteria harus diisi.');
30 }
```

**Gambar 5.10 Potongan source code untuk memproses isian kuesioner**

Implementasi saat ahli mengisikan kuesioner kriteria dapat dilihat pada Gambar 5.14. Proses pengisian kuesioner oleh ahli dilaksanakan pada tanggal 24 Maret 2018 di BLK Bantul.



**Gambar 5.11 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 5.12.

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 tergambar 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)	100	9 dari 9
3	Potensi	Kriteria yang digunakan untuk mengukur tingkat kesungguhan pendaftar. Data diperoleh dari wawancara.	(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 dilakukan pendaftar dan wawancara.	(Noe, dkk., 2012) (Ivancevich dan Konopaske, 2013) (Bogdanovic dan Miletic, 2014) (Kumar dkk., 2013) (Rouyeneh 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 5.12 Implementasi halaman hasil Delphi putaran pertama**

Gambar 5.13 menunjukkan potongan *source code* untuk menampilkan hasil Delphi putaran pertama. Halaman ini dapat diakses oleh kepala, kepala sub-bagian tata usaha, koordinator instruktur dan kepala kejuruan. Jika belum mengisi kuesioner kriteria, maka tidak bisa mengakses halaman ini. Variabel \$percentages pada baris 13-23 merupakan *query* yang digunakan untuk menampilkan kalkulasi pilihan kesesuaian kriteria dari para ahli. Dan variabel \$suggestion pada baris 24-31 merupakan *query* yang digunakan untuk menampilkan kriteria tambahan sebagai gagasan dari para ahli.

```

1 public function index(Request $request)
2 {
3     $role_id = Auth::user()->roleId();
4     $user = User::find(Auth::user()->id);
5     $data = Choice::where('user_id', '=', $user->id)->first();
6     if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
7         if ($data == null) {
8             return redirect()->route('questionnaire.create')
9             ->with('failed', 'Maaf, silahkan isi kuesioner kriteria dahulu.');
10        } else {
11            $i = 0;
12            $j = 0;
13            $percentages = Choice::select('criterias.name', 'criterias.description',
14                'criterias.citation',
15                DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
16                DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
17                ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
18                ->where('choice.suggestion', '=', '0')
19                ->where('criterias.step', '=', '1')
20                ->where('criterias.status', '=', '1')
21                ->groupBy('criterias.id')
22                ->orderBy('criterias.id', 'DESC')
23                ->get();
24            $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
25                ->join('choice', 'choice.user_id', '=', 'users.id')
26                ->join('criterias', 'criterias.id', '=', 'choice.criteria_id')
27                ->where('choice.suggestion', '=', '1')
28                ->where('criterias.step', '=', '2')
29                ->where('criterias.status', '=', '1')
30                ->orderBy('criterias.id', 'ASC')
31                ->get();
32            return view('result_step1.index', compact('percentages', 'data_suggestion', 'i', 'j'));
33        }
34    } else {
35        return redirect()->route('profile_users.show');
36    }
37 }

```

**Gambar 5.13 Potongan source code untuk menampilkan hasil Delphi putaran pertama**

#### 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 5.14.

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 merupakn rekomenadasi dari pegawai atau pihak tertentu. Rekomendasi ini diusulkan 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 kelilmuan. 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>

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) (Barrett, dkk., 2003)	11.11111111111111	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.	(Güngör dkk., 2009) (Bogdanovic dan Miletic, 2014)	22.22222222222222	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) (Ivancevich dan Konopaske, 2013)	0	0 dari 9	
12	Kesehatan	Kriteria yang digunakan untuk mengetahui kondisi kesehatan jasman pendaftar yang meliputi takaran darah, apakah mengidap penyakit tertentu, dll. Data diperoleh dari tes kesehatan, yang ditunjukkan dengan surat keterangan dokter.	(Noe, dkk., 2012) (Fajar dan Heru, 2010) (Safrial dan Tanti, 2015) (Robertson dan Smith, 2001)	44.44444444444444	4 dari 9	

No	Kriteria	Penjelasan Kriteria	Pengusul	Aksi
Masukan				

Copyright © 2017-2018 Dewi Anisa Istiqomah. Magister Ilmu Komputer Universitas Gadja Mada

Gambar 5.14 Halaman implementasi Delphi putaran kedua

Gambar 5.15 menunjukkan potongan *source code* untuk menampilkan halaman utama pengelolaan kriteria dalam Delphi putaran kedua. Halaman ini dapat diakses oleh kepala, kepala sub-bagian tata usaha, koordinator instruktur dan kepala kejuruan. Jika belum mengisi kuesioner kriteria, maka tidak bisa mengakses halaman ini. Variabel \$data\_fix pada baris 12-18 merupakan *query* yang digunakan untuk menampilkan semua kriteria yang digunakan pada tahap selanjutnya. Pada baris 19-28 digunakan untuk memberikan id referensi yang berfungsi supaya dapat dilacak asal mula kriteria jika kriteria mengalami perubahan nama atau detail yang lain. Variabel \$data\_standart pada baris 29-39 merupakan *query* yang digunakan untuk menampilkan kalkulasi pilihan kesesuaian kriteria dari para ahli. Dan variabel \$data\_suggestion pada baris 40-48 merupakan *query* yang digunakan untuk menampilkan kriteria tambahan sebagai gagasan dari para ahli.

```

1 $role_id = Auth::user()->roleId();
2 $user = User::find(Auth::user()->id);
3 $data = Choice::where('user_id', '=', $user->id)->first();
4 if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
5 if ($data == null) {
6 return redirect()->route('questionnaire.create')
7 ->with('failed','Maaf, silahkan isi kuesioner kriteria dahulu.');
8 } else {
9 $i = 0;
10 $j = 0;
11 $k = 0;
12 $data_fix = Choice::select('choice.*', 'criterias.*')
13 ->join('criterias','criterias.id','=','choice.criteria_id')
14 ->where('choice.suggestion', '=', '0')
15 ->where('criterias.step', '=', '2')
16 ->where('criterias.status', '=', '1')
17 ->orderBy('criterias.id','DESC')
18 ->get();
19 $criteriaCheck = Criteria::select('criterias.ref_id')
20 ->where('criterias.ref_id','<>',null)
21 ->where('criterias.step','=', '2')
22 ->get();
23 $usedCriteria = array();
24 foreach ($criteriaCheck as $used){
25 if ($used["ref_id"] != null) {
26 $usedCriteria[] = $used["ref_id"];
27 }
28 }
29 $data_standart = Choice::select('choice.*', 'criterias.*',
30 DB::raw('sum(choice.option) as sum'), DB::raw('count(choice.option) as count'),
31 DB::raw('sum(choice.option)/count(choice.option)*100 as result'))
32 ->join('criterias','criterias.id','=','choice.criteria_id')
33 ->where('choice.suggestion', '=', '0')
34 ->where('criterias.step', '=', '1')
35 ->where('criterias.status', '=', '1')
36 ->whereNotIn('criterias.id', $usedCriteria)
37 ->groupBy('criterias.id')
38 ->orderBy('criterias.id','DESC')
39 ->get();
40 $data_suggestion = User::select('choice.*', 'criterias.*', 'users.name AS user_name')
41 ->join('choice','choice.user_id','=','users.id')
42 ->join('criterias','criterias.id','=','choice.criteria_id')
43 ->where('choice.suggestion', '=', '1')
44 ->where('criterias.step', '=', '2')
45 ->where('criterias.status', '=', '1')
46 ->whereNotIn('criterias.id', $usedCriteria)
47 ->orderBy('criterias.id','ASC')
48 ->get();
49 return view('criteria_step2.index', compact('data_standart', 'data_suggestion',
50 'data_fix', 'i', 'j', 'k'));
51 }
52 } else {
53 return redirect()->route('profile_users.show');
54 }

```

**Gambar 5.15 Potongan source code untuk menampilkan halaman utama pengelolaan kriteria dalam Delphi putaran kedua**

Gambar 5.16 menunjukkan potongan *source code* untuk memproses masukan dari halaman tambah kriteria baru dalam Delphi putaran kedua. Fungsi validate pada baris 3 digunakan untuk validasi masukan yaitu name dan description harus disi. Semua masukan disimpan ke dalam dua model yaitu Criteria dan Choice. Pertama, disimpan ke model Criteria terlebih dahulu, yaitu terdiri dari name dan description sesuai dengan masukan pengguna, step = 2 dan status = 1. Kedua, disimpan ke model Choice, yaitu terdiri dari user\_id merupakan id pengguna, criteria\_id merupakan id criteria pada langkah pertama, option = 1 dan suggestion = 0.

```

1 public function store(Request $request)
2 {
3     $this->validate($request, [
4         'name' => 'required',
5         'description' => 'required',
6     ]);
7     $input = $request->all();
8     $input["step"] = 2;
9     $input["status"] = 1;
10    $criteria = Criteria::create($input);
11    $user = User::find(Auth::user()->id);
12    $choice["user_id"] = $user->id;
13    $choice["criteria_id"] = $criteria->id;
14    $choice["option"] = 1;
15    $choice["suggestion"] = 0;
16    Choice::create($choice);
17    return redirect()->route('criteriastep2.index')
18        ->with('success', 'Kriteria berhasil dibuat');
19 }

```

**Gambar 5.16 Potongan *source code* untuk memproses masukan dari halaman tambah kriteria dalam Delphi putaran kedua**

Gambar 5.17 menunjukkan potongan *source code* untuk memproses perubahan dari halaman edit kriteria dalam Delphi putaran kedua. Fungsi validate pada baris 3 digunakan untuk validasi data yaitu name dan description harus disi. Semua perubahan untuk kriteria dengan \$id sesuai masukan di-*update* ke dalam model Criteria.

```

1 public function update(Request $request, $id)
2 {
3     $this->validate($request, [
4         'name' => 'required',
5         'description' => 'required',
6     ]);
7     Criteria::find($id)->update($request->all());
8     return redirect()->route('criteriastep2.index')
9     ->with(['success', 'Kriteria berhasil diedit']);
10 }

```

**Gambar 5.17 Potongan *source code* untuk memproses perubahan dari halaman edit kriteria dalam Delphi putaran kedua**

Gambar 5.18 menunjukkan potongan *source code* untuk menghapus suatu kriteria dalam Delphi putaran kedua. Sebelum menghapus, dilakukan pengecekan dahulu, apakah kriteria sudah ada penilaian. Jika belum ada penilaian, maka kriteria dengan \$id sesuai yang dipilih dapat langsung dihapus menggunakan fungsi delete pada model Criteria dan Choice. Namun jika sudah ada penilaian, maka kriteria tidak bisa dihapus dan memunculkan pesan kesalahan.

```

1 public function destroy($id)
2 {
3     $result = ResultSelection::where('criteria_id', '=', $id)->first();
4     $pairwise = PairwiseComparison::where('criterial_id', '=', $id)->first();
5     if ($pairwise == null && $result == null) {
6         Choice::join('criterias', 'criterias.id', '=', 'choice.criteria_id')
7         ->where('choice.criteria_id', '=', $id)
8         ->delete();
9         Criteria::find($id)->delete();
10        return redirect()->route('criteriastep2.index')
11        ->with(['success', 'Kriteria berhasil dihapus']);
12    } else {
13        return redirect()->route('criteriastep2.index')
14        ->with(['failed', 'Kriteria tidak bisa dihapus karena sudah ada penilaian']);
15    }
16 }

```

**Gambar 5.18 Potongan *source code* untuk menghapus suatu kriteria dalam Delphi putaran kedua**

Gambar 5.19 menunjukkan potongan *source code* untuk menggunakan kriteria pada Delphi putaran pertama. Prinsip dalam menggunakan kriteria pada tahap sebelumnya adalah membuat kriteria baru dengan nama dan deskripsi yang sama. Semua data disimpan ke dalam dua model yaitu Criteria dan Choice. Pertama, disimpan ke model Criteria terlebih dahulu, yaitu terdiri dari name dan

description sesuai kriteria yang digunakan, step = 2, status = 1 dan ref\_id sesuai dengan id kriteria yang digunakan. Id referensi digunakan untuk menandai asal mula kriteria yang baru. Kedua, disimpan ke model Choice, yaitu terdiri dari user\_id merupakan id pengguna, criteria\_id merupakan id criteria pada langkah pertama, option = 1 dan suggestion = 0.

```

1 public function use(Request $request, $id)
2 {
3     $data = Choice::join('criterias','criterias.id','=','choice.criteria_id')
4     ->where('choice.criteria_id', '=', $id)
5     ->first();
6     $input['name'] = $data['name'];
7     $input['description'] = $data['description'];
8     $input["step"] = 2;
9     $input["status"] = 1;
10    $input["ref_id"] = $data["id"];
11    $criteria = Criteria::create($input);
12    $user = User::find(Auth::user()->id);
13    $choice["user_id"] = $user->id;
14    $choice["criteria_id"] = $criteria->id;
15    $choice["option"] = 1;
16    $choice["suggestion"] = 0;
17    Choice::create($choice);
18    return redirect()->route('criteriastep2.index')
19    ->with('success','Kriteria berhasil digunakan');
20 }
21 }
```

**Gambar 5.19 Potongan *source code* untuk menggunakan kriteria pada Delphi putaran pertama**

Implementasi saat proses diskusi untuk menentukan kriteria yang digunakan dalam penilaian seleksi peserta pelatihan oleh para ahli dapat dilihat pada Gambar 5.20.



**Gambar 5.20 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 5.21.

No	Kriteria/Sub-Kriteria	Aksi
1	Faktor intelektual	<a href="#">Edit</a> <a href="#">Delete</a>
	• Pengalaman pelatihan	<a href="#">Remove from Group</a>
	• Pengetahuan	<a href="#">Remove from Group</a>
	• Keterampilan teknis	<a href="#">Remove from Group</a>
	• Pendidikan terakhir	<a href="#">Remove from Group</a>
	• Keterampilan komunikasi	<a href="#">Remove from Group</a>
2	Faktor personal	<a href="#">Edit</a> <a href="#">Delete</a>
	• Intensitas keikutsertaan pelatihan di BLK Bantul	<a href="#">Remove from Group</a>

**Gambar 5.21 Implementasi halaman hierarki kriteria**

Gambar 5.22 menunjukkan potongan *source code* untuk menampilkan halaman hierarki kriteria. Halaman ini dapat diakses oleh kepala, kepala sub-bagian tata usaha, koordinator instruktur dan kepala kejuruan. Jika belum mengisi kuesioner kriteria, maka tidak bisa mengakses halaman ini. Variabel \$criteria\_fix pada baris 11-18 merupakan *query* yang digunakan untuk menampilkan semua kriteria yang sudah ditentukan pada Delphi putaran kedua. Variabel \$list\_group pada baris 19-22 digunakan untuk mengambil data kelompok kriteria. Pada baris 24-38 digunakan untuk menampilkan semua data kriteria yang sudah dikelompokkan berdasarkan kelompok kriteria. Inisialisasi variabel \$i dan \$j pada baris 9-10 digunakan untuk penomoran pada halaman hierarki kriteria. Kriteria yang tidak mempunyai kelompok kriteria, tingkatannya adalah sebagai kriteria. Dan kriteria yang mempunyai kelompok kriteria, tingkatannya adalah sebagai subkriteria.

```

1 $role_id = Auth::user()->roleId();
2 $user = User::find(Auth::user()->id);
3 $data = Choice::where('user_id', '=', $user->id)->first();
4 if ($role_id == 3 || $role_id == 4 || $role_id == 5 || $role_id == 6) {
5 if ($data == null) {
6 return redirect()->route('questionnaire.create')
7 ->with('failed','Maaf, silahkan isi kuesioner kriteria dahulu.');
8 } else {
9 $i = 0;
10 $j = 0;
11 $criterias_fix = Choice::select('choice.*', 'criterias.*')
12 ->join('criterias','criterias.id','=','choice.criteria_id')
13 ->where('choice.suggestion', '=', '0')
14 ->where('criterias.step', '=', '2')
15 ->where('criterias.status', '=', '1')
16 ->where('criterias.group_criteria', '=', null)
17 ->orderBy('criterias.id','DESC')
18 ->get();
19 $list_group = Criteria::where('group_criteria','=',null)
20 ->where('description','=',null)
21 ->lists('name','id')
22 ->all();
23 $criterias_group = array();
24 foreach ($list_group as $key=>$name){
25 $criterias_group[$key]["name"] = $name;
26 $criterias_group[$key]["data"] = array();
27 $criterias = Choice::select('choice.*', 'criterias.*')
28 ->join('criterias','criterias.id','=','choice.criteria_id')
29 ->where('choice.suggestion', '=', '0')
30 ->where('criterias.step', '=', '2')
31 ->where('criterias.status', '=', '1')
32 ->where('criterias.group_criteria', '=', $key)
33 ->orderBy('criterias.id','DESC')
34 ->get();
35 foreach ($criterias as $criteria){
36 $criterias_group[$key]["data"][] = $criteria;
37 }
38 }
39 return view('criteria_group.index',compact('criterias', 'criterias_fix', 'list_group',
40 'criterias_group', 'i', 'j'));
41 }
42 } else {
43 return redirect()->route('profile_users.show');
44 }

```

**Gambar 5.22 Potongan *source code* untuk menampilkan halaman hierarki kriteria**

Gambar 5.23 menunjukkan potongan *source code* untuk memproses masukan dari halaman tambah kelompok kriteria. Masukan step diberikan nilai 2 dan status diberikan nilai 1. Semua masukan disimpan ke dalam model Criteria. Jika proses menyimpan kelompok kriteria ke *database* berhasil, maka menampilkan pesan kelompok kriteria berhasil dibuat.

```

1 public function store(Request $request)
2 {
3     $input = $request->all();
4     $input['step'] = '2';
5     $input['status'] = '1';
6     Criteria::create($input);
7     return redirect()->route('criteriagroup.index')
8     ->with(['success', 'Kelompok kriteria berhasil dibuat']);
9 }

```

**Gambar 5.23 Potongan *source code* untuk memproses masukan dari halaman tambah kelompok kriteria**

Gambar 5.24 menunjukkan potongan *source code* untuk memproses perubahan dari halaman edit kelompok kriteria. Semua perubahan untuk kelompok kriteria dengan \$id sesuai masukan di-*update* ke dalam model Criteria.

```

1 public function update(Request $request, $id)
2 {
3     Criteria::find($id)->update($request->all());
4     return redirect()->route('criteriagroup.index')
5     ->with(['success', 'Kelompok kriteria berhasil diedit']);
6 }

```

**Gambar 5.24 Potongan *source code* untuk memproses perubahan dari halaman edit kelompok kriteria**

Gambar 5.25 menunjukkan potongan *source code* untuk menghapus kelompok kriteria. Sebelum menghapus kelompok kriteria, harus memberikan nilai untuk group\_criteria = null pada anggotanya, sehingga saat kelompok kriteria dihapus maka anggotanya tersebut otomatis sudah tidak mempunyai kelompok. Setelah itu, kelompok kriteria dengan \$id sesuai yang dipilih dihapus menggunakan fungsi delete pada model Criteria.

```

1 public function destroy($id)
2 {
3     Criteria::where('group_criteria', '=', $id)->update(['group_criteria' => null]);
4     Criteria::find($id)->delete();
5     return redirect()->route('criteriagroup.index')
6     ->with(['success', 'Kelompok kriteria berhasil dihapus']);
7 }

```

**Gambar 5.25 Potongan *source code* untuk menghapus kelompok kriteria**

Gambar 5.26 menunjukkan potongan *source code* untuk menambahkan kriteria ke dalam kelompok kriteria. Kriteria dengan \$id yang dipilih diberikan

nilai untuk group\_criteria sesuai dengan masukan. Kriteria ini tingkatannya adalah sebagai subkriteria.

```

1 public function add(Request $request)
2 {
3 Criteria::find($request['id'])->update($request->all());
4 return redirect()->route('criteriagroup.index')
5 ->with('success','Kriteria berhasil dikelompokkan');
6 }

```

**Gambar 5.26 Potongan *source code* untuk menambahkan kriteria ke dalam kelompok kriteria**

Gambar 5.27 menunjukkan potongan *source code* untuk menghapus kriteria dari kelompok kriteria. Kriteria dengan \$id yang dipilih diberikan nilai untuk group\_criteria = null. Kriteria ini tidak mempunyai kelompok dan tingkatannya adalah sebagai kriteria.

```

1 public function out(Request $request)
2 {
3 Criteria::find($request['id'])->update(['group_criteria' => null]);
4 return redirect()->route('criteriagroup.index')
5 ->with('success','Kriteria berhasil dikeluarkan dari kelompok');
6 }

```

**Gambar 5.27 Potongan *source code* untuk menghapus kriteria dari kelompok kriteria**

## 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 5.28.

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 5.29 menunjukkan implementasi halaman bobot.

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 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
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 5.28 Implementasi halaman perbandingan berpasangan

No	Kriteria/Sub-Kriteria	Pairwise Comparison	Bobot Parsial	Bobot Global
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 5.29 Implementasi halaman bobot

Gambar 5.30 menunjukkan potongan *source code* untuk menampilkan halaman bobot. Halaman bobot hanya bisa diakses oleh kepala. Variabel \$criterias pada baris 4-12 digunakan untuk mengambil data kriteria yang group\_criteria = null, artinya adalah mengambil data pada tingkatan kriteria. Pada baris 13-27 digunakan untuk mengambil data subkriteria berdasarkan kriterianya.

```

1 $role_id = Auth::user()->roleId();
2 if ($role_id == 3) {
3     $i = 0;
4     $criterias = Criteria::where('step', '=', '2')
5         ->where('status', '=', '1')
6         ->where('group_criteria', '=', null)
7         ->whereNotIn('id', function($query){
8             $query->select('criteria_id')
9             ->from(with(new Choice)->getTable())
10            ->where('suggestion', 1);
11        })
12        ->get();
13     $criterias_group = array();
14     $total_criterias = 0;
15     foreach ($criterias as $criteria) {
16         $criterias_group[$criteria->id]["group"] = $criteria;
17         $criterias_group[$criteria->id]["data"] = array();
18         $subcriterias = Criteria::where('step', '=', '2')
19             ->where('status', '=', '1')
20             ->where('group_criteria', '=', $criteria->id)
21             ->whereNotIn('id', function($query){
22                 $query->select('criteria_id')
23                 ->from(with(new Choice)->getTable())
24                 ->where('suggestion', 1);
25             })
26             ->orderBy('id', 'DESC')
27             ->get();
28         return view('weights.index', compact('criterias_group', 'i'));
29     } else {
30         return redirect()->route('profile_users.show');
31     }

```

**Gambar 5.30 Potongan *source code* untuk menampilkan halaman bobot**

Gambar 5.31 menunjukkan potongan *source code* untuk menghitung bobot global. Jika tidak mempunyai subkriteria, maka bobot globalnya adalah bobot lokal dari kriteria tersebut. Jika terdapat subkriteria, maka bobot global dari subkriteria tersebut adalah perkalian antara bobot lokal kriteria dan bobot lokal subkriteria. Bobot global yang sudah dihitung disimpan ke dalam *database*.

```

1 if (count($subcriterias) == 0) {
2     $criteria->global_weight = $criteria->partial_weight;
3     $criteria->save();
4 }
5 foreach ($subcriterias as $subcriteria) {
6     $subcriteria->global_weight = number_format($criteria->partial_weight * $subcriteria-
7         >partial_weight, 3);
8     $subcriteria->save();
9     $criterias_group[$criteria->id]["data"][] = $subcriteria;
10    }

```

**Gambar 5.31 Potongan *source code* untuk menghitung bobot global**

Gambar 5.32 menunjukkan potongan *source code* untuk memproses masukan perbandingan berpasangan. Variable \$pairwises digunakan untuk menangkap nilai perbandingan berpasangan yang dimasukkan. Pengguna hanya membandingkan dengan nilai bilangan bulat, sehingga untuk kebalikan dari perbandingan diberikan nilai 1/nilai masukan tersebut, seperti pada baris 12-16. Perbandingan antara kriteria yang sama diberikan nilai = 1, seperti pada baris 9.

```

1 $pairwises = array();
2 $count = 0;
3 foreach ($criterias as $criteria1) {
4 foreach ($criterias as $criteria2) {
5 $data = array();
6 $data["criteriad_1"] = $criteria1->id;
7 $data["criteriad_2"] = $criteria2->id;
8 if ($criteria1->id == $criteria2->id) {
9 $data["value"] = 1;
10 } elseif (array_key_exists($criteria1->id . ":" . $criteria2->id, $input["criteria_id"])) {
11 $selected_id = $input["criteria_id"][$criteria1->id . ":" . $criteria2->id];
12 if ($selected_id == $criteria1->id) {
13 $data["value"] = $input["value"][$criteria1->id . ":" . $criteria2->id];
14 } else {
15 $data["value"] = 1 / $input["value"][$criteria1->id . ":" . $criteria2->id];
16 }
17 } elseif (array_key_exists($criteria2->id . ":" . $criteria1->id, $input["criteria_id"])) {
18 $selected_id = $input["criteria_id"][$criteria2->id . ":" . $criteria1->id];
19 if ($selected_id == $criteria2->id) {
20 $data["value"] = $input["value"][$criteria2->id . ":" . $criteria1->id];
21 } else {
22 $data["value"] = 1 / $input["value"][$criteria2->id . ":" . $criteria1->id];
23 }
24 }
25 $pairwises[] = $data;
26 }}
```

**Gambar 5.32 Potongan *source code* untuk memproses masukan perbandingan berpasangan**

Gambar 5.33 menunjukkan potongan *source code* untuk membentuk matriks hasil perbandingan berpasangan.

```

1 $base_table = array();
2 foreach ($pairwises as $pairwise){
3 if (!array_key_exists($pairwise["criteriad_1"], $base_table)) {
4 $base_table[$pairwise["criteriad_1"]] = array();
5 }
6 $base_table[$pairwise["criteriad_1"]][$pairwise["criteriad_2"]] = $pairwise["value"];
7 }
```

**Gambar 5.33 Potongan *source code* untuk membentuk matriks hasil perbandingan berpasangan**

Gambar 5.34 menunjukkan potongan *source code* untuk membentuk matriks hasil normalisasi perbandingan berpasangan. Perhitungan normalisasi yaitu dengan membagi antara nilai perbandingan kriteria pada matriks perbandingan berpasangan dengan jumlah kolom kriteria tersebut.

```

1 $normalized_table = array();
2 $total_basecol_val = array();
3 foreach ($base_table as $row=>$row_val) {
4 if (!array_key_exists($row,$normalized_table)) {
5 $normalized_table[$row] = array();
6 }
7 foreach ($row_val as $col=>$col_val) {
8 $total_row_val = 0;
9 $total_basecol_val[$col] = 0;
10 foreach ($base_table as $r=>$each_row) {
11 foreach ($each_row as $c=>$c_val) {
12 if ($c == $col) {
13 $total_row_val += $c_val;
14 $total_basecol_val[$col] += $c_val;
15 break;
16 }
17 }
18 }
19 $normalized_table[$row][$col] = $col_val / $total_row_val;
20 }
21 }

```

**Gambar 5.34 Potongan *source code* untuk membentuk matriks hasil normalisasi perbandingan berpasangan.**

Gambar 5.35 menunjukkan potongan *source code* untuk menghitung bobot lokal kriteria. Bobot lokal kriteria dihitung dengan membagi antara total jumlah baris setiap kriteria pada hasil normalisasi perbandingan berpasangan dengan banyaknya data yang dibandingkan.

```

1 $partial_weight = array();
2 foreach ($normalized_table as $row=>$each_row) {
3 // hitung total jumlah di baris
4 $total_row_val = 0;
5 foreach ($each_row as $col=>$c_val){
6 $total_row_val += $c_val;
7 }
8 // hitung bobot partial
9 $partial_weight[$row] = $total_row_val / count($each_row);
10 }

```

**Gambar 5.35 Potongan *source code* untuk menghitung bobot lokal**

Gambar 5.36 menunjukkan potongan *source code* untuk menghitung lamda max. Lamda max dapat dihitung dengan menjumlahkan seluruh hasil perkalian antara bobot dan jumlah kolom setiap kriteria pada matriks perbandingan berpasangan.

```

1 $lamda_max = 0;
2 foreach ($partial_weight as $r=>$value) {
3 $criteria_factor = $value * $total_basecol_val[$r];
4 $lamda_max += $criteria_factor; }
```

**Gambar 5.36 Potongan *source code* untuk menghitung lamda max**

Gambar 5.37 menunjukkan potongan *source code* untuk menghitung CI. CI dihitung dengan Persamaan (3.1).

```
1 $ci = ($lamda_max - count($partial_weight))/(count($partial_weight) - 1);
```

**Gambar 5.37 Potongan *source code* untuk menghitung CI**

Gambar 5.38 menunjukkan potongan *source code* untuk menentukan nilai RI. Nilai RI ditentukan berdasarkan jumlah kriteria yang dibandingkan.

```

1 if (count($partial_weight) == 3) {
2 $ri = 0.58;
3 } elseif (count($partial_weight) == 4) {
4 $ri = 0.90;
5 } elseif (count($partial_weight) == 5) {
6 $ri = 1.12;
7 } elseif (count($partial_weight) == 6) {
8 $ri = 1.24;
9 } elseif (count($partial_weight) == 7) {
10 $ri = 1.32;
11 } elseif (count($partial_weight) == 8) {
12 $ri = 1.41;
13 } elseif (count($partial_weight) == 9) {
14 $ri = 1.45;
15 } elseif (count($partial_weight) == 10) {
16 $ri = 1.49;
17 } elseif (count($partial_weight) == 11) {
18 $ri = 1.51;
19 } elseif (count($partial_weight) == 12) {
20 $ri = 1.48;
21 } elseif (count($partial_weight) == 13) {
22 $ri = 1.56;
23 } elseif (count($partial_weight) == 14) {
24 $ri = 1.57;
25 } elseif (count($partial_weight) == 15) {
26 $ri = 1.59;
27 } else { $ri = 0; }
```

**Gambar 5.38 Potongan *source code* untuk menentukan nilai RI**

Gambar 5.39 menunjukkan potongan *source code* untuk menghitung CR. CR dihitung dengan Persamaan (3.2).

```
1 $cr = number_format($ci / $ri * 100, 2);
```

### **Gambar 5.39 Potongan *source code* untuk menghitung CR**

Gambar 5.40 menunjukkan potongan *source code* untuk mengecek apakah perbandingan berpasangan konsisten. Jika  $CR \leq 10\%$ , maka perbandingan berpasangan dinyatakan konsisten dan berhasil untuk disimpan. Namun jika  $CR > 10\%$ , maka perbandingan berpasangan dinyatakan tidak konsisten dan harus dilakukan reevaluasi penilaian perbandingan berpasangan.

```
1 if ($cr <= 10) {
2   foreach ($partial_weight as $key=>$val_partial_weight) {
3     $data_criteria["partial_weight"] = $val_partial_weight;
4     Criteria::find($key)->update($data_criteria);
5   }
6   foreach ($pairwises as $pairwise) {
7     $data_pairwise["criteria1_id"] = $pairwise["criteriad_1"];
8     $data_pairwise["criteria2_id"] = $pairwise["criteriad_2"];
9     $data_pairwise["value"] = $pairwise["value"];
10    $check_pairwise = PairwiseComparison::where('criteria1_id', '=', 
11      $pairwise["criteriad_1"])
12      ->where('criteria2_id', '=', $pairwise["criteriad_2"])
13      ->first();
14    if ($check_pairwise == null){
15      PairwiseComparison::create($data_pairwise);
16    } else {
17      PairwiseComparison::where('criteria1_id', '=', $pairwise["criteriad_1"])
18        ->where('criteria2_id', '=', $pairwise["criteriad_2"])
19        ->update($data_pairwise);
20    }
21  }
22  return redirect()->route('weights.index')
23  ->with('success', 'Penilaian bobot kriteria berhasil disimpan');
24 } else {
25   return redirect()->route('weights.pairwise', $id)
26   ->with('failed', 'Nilai Consistency Ratio (CR) = ' . $cr . '%. Nilai CR harus <= 10%.
27   Silahkan reevaluasi penilaian perbandingan berpasangan kriteria.');
28 }
```

### **Gambar 5.40 Potongan *source code* untuk mengecek apakah perbandingan berpasangan konsisten**

#### **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 5.41. 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.

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 5.41 Implementasi halaman tambah nilai peserta seleksi pelatihan**

Gambar 5.42 menunjukkan potongan *source code* untuk memproses masukan dari halaman tambah nilai peserta seleksi pelatihan. Variabel \$criterias pada baris 4-13 digunakan untuk mengambil data kriteria yang sudah ditentukan

pada tahap metode *Modified Delphi*. Pemberian nilai peserta seleksi pelatihan ini berdasarkan kriteria.

```

1 public function store(Request $request, $id)
2 {
3     $input = $request->all();
4     $criterias = Criteria::where('step', '=', '2')
5         ->where('status', '=', '1')
6         ->where('description', '<>', null)
7         ->whereNotIn('id', function($query){
8             $query->select('criteria_id')
9             ->from('with(new Choice)->getTable()')
10            ->where('suggestion', 1);
11        })
12        ->orderBy('id', 'DESC')
13        ->lists('id');
14    foreach ($criterias as $criteria) {
15        $result_selection = ResultSelection::where('selection_id', '=', $id)
16            ->where('criteria_id', '=', $criteria)
17            ->first();
18        if ($input[$criteria] != "") {
19            $data["selection_id"] = $id;
20            $data["criteria_id"] = $criteria;
21            $data["value"] = $input[$criteria];
22            if ($result_selection == null) {
23                ResultSelection::create($data);
24            } else {
25                ResultSelection::where('selection_id', '=', $id)
26                    ->where('criteria_id', '=', $criteria)
27                    ->update($data);
28            }
29        } else {
30            ResultSelection::where('selection_id', '=', $id)
31                ->where('criteria_id', '=', $criteria)
32                ->delete();
33        }
34    }
35    return redirect()->route('result_selection.index')
36        ->with('success', 'Penilaian berhasil disimpan');
37 }
```

**Gambar 5.42 Potongan source code untuk memproses masukan dari halaman tambah nilai peserta seleksi pelatihan**

## 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 5.43.

No	Kriteria	Tipe Preferensi	Kaidah (Maks/Min)	Parameter p	Parameter q	Parameter s	Aksi
1	Intensitas keikutsertaan pelatihan di BLK Bantul	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
2	Pengalaman pelatihan	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
3	Rencana setelah selesai pelatihan	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
4	Rekomendasi	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
5	Pengetahuan	5	Maksimasi	10.000	5.000	-	<button>Input Preferensi</button>
6	Keterampilan teknis	5	Maksimasi	10.000	5.000	-	<button>Input Preferensi</button>
7	Kejujuran	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
8	Sikap	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
9	Pendidikan terakhir	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>
10	Usia	4	Maksimasi	1.000	0.500	-	<button>Input Preferensi</button>

**Gambar 5.43 Implementasi halaman pengelolaan fungsi preferensi**

Gambar 5.44 menunjukkan potongan *source code* untuk menampilkan halaman pengelolaan fungsi preferensi. Halaman ini hanya dapat diakses oleh kepala. Variabel \$preferences pada baris 5-14 digunakan untuk menampilkan daftar kriteria dengan paginasi 10 untuk setiap halamannya.

```

1 public function index(Request $request)
2 {
3     $role_id = Auth::user()->roleId();
4     if ($role_id == 3) {
5         $preferences = Criteria::where('step', '=', '2')
6             ->where('status', '=', '1')
7             ->where('description', '<>', null)
8             ->whereNotIn('id', function($query){
9                 $query->select('criteria_id')
10                ->from(with(new Choice)->getTable())
11                ->where('suggestion', 1);
12            })
13            ->orderBy('id', 'DESC')
14            ->paginate(10);
15        return view('preferences.index', compact('preferences'))
16        ->with('i', ($request->input('page', 1) - 1) * 10);
17    } else {
18        return redirect()->route('profile_users.show');
19    }
20 }
```

**Gambar 5.44 potongan *source code* untuk menampilkan halaman pengelolaan fungsi preferensi**

Gambar 5.45 menunjukkan potongan *source code* untuk memproses masukan dari halaman input fungsi preferensi. Fungsi validate pada baris 3 digunakan untuk validasi yaitu tipe preferensi dan kaidah max/min harus diisi. Parameter bersifat opsional tergantung tipe preferensinya. Jika tipe preferensi 1, maka tidak mempunyai parameter. Jika tipe preferensi 2, maka mempunyai parameter q. Jika tipe preferensi 3, maka mempunyai parameter p. Jika tipe preferensi 4 dan 5, maka mempunyai parameter p dan q. Jika selain itu atau tipe preferensi 6, maka mempunyai parameter s.

```

1 public function update(Request $request, $id)
2 {
3     $this->validate($request, [
4         'preference' => 'required',
5         'max_min' => 'required',
6     ]);
7     $input = $request->all();
8     if ($input['preference'] == 1) {
9         $input['parameter_p'] = null;
10        $input['parameter_q'] = null;
11        $input['parameter_s'] = null;
12    } elseif ($input['preference'] == 2) {
13        $input['parameter_p'] = null;
14        $input['parameter_s'] = null;
15    } elseif ($input['preference'] == 3) {
16        $input['parameter_q'] = null;
17        $input['parameter_s'] = null;
18    } elseif ($input['preference'] == 4 || $input['preference'] == 5) {
19        $input['parameter_s'] = null;
20    } else {
21        $input['parameter_p'] = null;
22        $input['parameter_q'] = null;
23    }
24     Criteria::find($id)->update($input);
25     return redirect()->route('preferences.index')
26     ->with('success', 'Tipe preferensi berhasil disimpan');
27 }
```

**Gambar 5.45 Potongan *source code* untuk menampilkan halaman pengelolaan fungsi preferensi**

### 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 5.46. 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>

Gambar 5.46 Implementasi halaman penilaian peserta seleksi pelatihan

Gambar 5.47 menunjukkan potongan *source code* untuk menampilkan halaman penilaian peserta seleksi pelatihan. Halaman ini hanya dapat diakses oleh kepala. Variabel \$data pada baris 5-20 digunakan untuk menampilkan data yang berhubungan dengan peserta seleksi antara lain data seleksi, jadwal seleksi, identitas peserta dan subkejuruan yang daftar. Data peserta seleksi pelatihan ditampilkan dalam paginasi. Setiap paginasi akan ditampilkan sepuluh data peserta seleksi pelatihan. Peserta seleksi pelatihan yang ditampilkan hanya peserta yang masih dalam tahap proses penilaian. Jika peserta sudah dinyatakan diterima/gagal, tidak akan ditampilkan dalam halaman ini.

```

1 public function index(Request $request)
2 {
3     $role_id = Auth::user()->roleId();
4     if ($role_id == 3) {
5         $data = Selection::select('selections.*', 'users.name AS name_registrant',
6             'selection_schedules.date',
7             'selection_schedules.time', 'sub_vocationals.name AS name_sub_vocational')
8             ->join('registrations', 'registrations.id', '=', 'selections.registration_id')
9             ->join('registrants', 'registrants.id', '=', 'selections.registrant_id')
10            ->join('users', 'users.id', '=', 'selections.user_id')
11            ->join('selection_schedules', 'selection_schedules.id', '=',
12                'selections.selection_schedule_id')
13            ->join('sub_vocationals', 'sub_vocationals.id', '=',
14                'selection_schedules.sub_vocational_id')
15            ->where(function($query){
16                $query->whereNull('selections.status')
17                ->orWhere('selections.status', '=', '');
18            })
19            ->orderBy('selections.id', 'DESC')
20            ->paginate(10);
21        return view('result_selection.index', compact('data', 'result_selection'))
22        ->with('i', ($request->input('page', 1) - 1) * 10);
23    } else {
24        return redirect()->route('profile_users.show');
25    }
26}

```

**Gambar 5.47 Potongan *source code* untuk menampilkan halaman penilaian peserta seleksi pelatihan**

Gambar 5.48 menunjukkan potongan *source code* untuk membentuk matriks perbandingan berpasangan antara alternatif dan kriteria. Variabel \$criterias pada baris 2-11 digunakan untuk mengambil data kriteria yang sudah ditentukan berdasarkan metode *Modified Delphi*. Peserta seleksi pelatihan dinilai berdasarkan kriteria sehingga membentuk matriks perbandingan berpasangan antara alternatif dan kriteria, seperti terlihat pada baris 27. Jika ada peserta seleksi pelatihan yang nilainya masih kosong, maka tidak dapat dilakukan penilaian. Pengguna harus memasukkan semua nilai peserta seleksi pelatihan terlebih dahulu.

```

1 $tabel_alternative = array();
2 $criterias = Criteria::where('step', '=', '2')
3 ->where('status', '=', '1')
4 ->where('description', '<>', null)
5 ->whereNotIn('id', function($query){
6 $query->select('criteria_id')
7 ->from(with(new Choice)->getTable())
8 ->where('suggestion', 1);
9 })
10 ->orderBy('id','DESC')
11 ->get();
12 $criteriasData = array();
13 foreach ($criterias as $criteria) {
14 $criteriasData[$criteria->id] = $criteria;
15 }
16 foreach ($selections as $selection) {
17 $tabel_alternative[$selection->id] = array();
18 foreach ($criterias as $criteria) {
19 $result_selection = ResultSelection::where('selection_id', '=', $selection->id)
20 ->where('criteria_id', '=', $criteria->id)
21 ->first();
22 if ($result_selection == null) {
23 return redirect()->route('result_selection.index')
24 ->with('failed','Hitung penilaian GAGAL! '. $selection->name_registrant . ' belum dinilai.
25 Silahkan lakukan penilaian');
26 }
27 $tabel_alternative[$selection->id][$criteria->id] = $result_selection->value;
28 }
29 }

```

**Gambar 5.48 Potongan *source code* untuk membentuk matriks perbandingan berpasangan antara alternatif dan kriteria**

Gambar 5.49 menunjukkan potongan *source code* untuk menghitung selisih antar-alternatif pada setiap kriteria. Selisih ini diperoleh dengan mengurangkan nilai alternatif pertama dengan kedua atau sebaliknya pada setiap kriteria.

```

1 $tabel_selisih = array();
2 foreach ($tabel_alternative as $key1=>$data_selisih1) {
3 foreach ($tabel_alternative as $key2=>$data_selisih2) {
4 if ($key1 != $key2) {
5 $tabel_selisih[$key1.".".$key2] = array();
6 foreach ($data_selisih1 as $criteria=>$value) {
7 $tabel_selisih[$key1.".".$key2][$criteria] = $value - $data_selisih2[$criteria];
8 }
9 }
10 }
11 }

```

**Gambar 5.49 Potongan *source code* untuk menghitung selisih antar-alternatif pada setiap kriteria**

Gambar 5.50 menunjukkan potongan *source code* untuk menghitung derajat preferensi antar-alternatif pada setiap kriteria. Perhitungan derajat preferensi tergantung pada tipe preferensi masing-masing kriteria. Perhitungan derajat preferensi dapat dilihat pada Tabel 4.25.

```

1 $tabel_derajat = array();
2 foreach ($tabel_selisih as $alternatives=>$crt_data) {
3   $tabel_derajat[$alternatives] = array();
4   foreach ($crt_data as $criteriaId=>$value) {
5     $criteria = $criteriasData[$criteriaId];
6     $type = $criteria->preference;
7     switch ($type) {
8       case "1":
9         if ($value <= 0) {
10           $tabel_derajat[$alternatives][$criteriaId] = 0;
11         } else {
12           $tabel_derajat[$alternatives][$criteriaId] = 1;
13         }
14         break;
15       case "2":
16         $q = $criteria->parameter_q;
17         if ($value <= $q) {
18           $tabel_derajat[$alternatives][$criteriaId] = 0;
19         } else {
20           $tabel_derajat[$alternatives][$criteriaId] = 1;
21         }
22         break;
23       ...
24       case "6":
25         $e = 2.71828;
26         $s = $criteria->parameter_s;
27         if ($value <= 0) {
28           $tabel_derajat[$alternatives][$criteriaId] = 0;
29         } else {
30           $pow_d = pow($value, 2);
31           $pow_s = pow($s, 2);
32           $pow_val = -($pow_d / (2 * $pow_s));
33           $pow_e = pow($e, $pow_val);
34           $tabel_derajat[$alternatives][$criteriaId] = 1 - $pow_e;
35         }
36         break;
37       default:
38         $tabel_derajat[$alternatives][$criteriaId] = "-";
39         break;
40     }
41   }
42 }
```

**Gambar 5.50 Potongan *source code* untuk menghitung derajat preferensi antar-alternatif pada setiap kriteria**

Gambar 5.51 menunjukkan potongan *source code* untuk menghitung indeks preferensi. Indeks preferensi dapat dihitung dengan menjumlahkan semua

hasil perkalian antara bobot dan derajat preferensi, seperti pada baris 11-13.

```

1 $tabel_index = array();
2 foreach ($tabel_derajat as $alternatives=>$data_index) {
3 $alternativesId = explode(",",$alternatives);
4 $id1 = $alternativesId[0];
5 $id2 = $alternativesId[1];
6 $tabel_index[$id1][$id2] = 0;
7 $tabel_index[$id1][$id1] = 0;
8 $tabel_index[$id2][$id2] = 0;
9 foreach ($data_index as $criterionId=>$value) {
10 $criterion = $criteriasData[$criterionId];
11 $bobot = $criterion->global_weight;
12 $mlt = $bobot * $value;
13 $tabel_index[$id1][$id2] += number_format($mlt,5);
14 }
15 }
```

**Gambar 5.51 Potongan *source code* untuk menghitung indeks preferensi**

Gambar 5.52 menunjukkan potongan *source code* untuk menghitung *leaving flow* dan *entering flow*. *Leaving flow* dapat dihitung dengan Persamaan (3.4), seperti pada baris 13. *Entering flow* dapat dihitung dengan Persamaan (3.5), seperti pada baris 14.

```

1 $tabel_leaving = array();
2 $tabel_entering = array();
3 $n = count($selectionsId);
4 foreach($selectionsId as $selectionId1) {
5 $sum_row = 0;
6 $sum_col = 0;
7 foreach($tabel_index[$selectionId1] as $value) {
8 $sum_row += $value;
9 }
10 foreach($selectionsId as $selectionId2) {
11 $sum_col += $tabel_index[$selectionId2][$selectionId1];
12 }
13 $tabel_leaving[$selectionId1] = number_format((1 / ($n-1)) * $sum_row, 5);
14 $tabel_entering[$selectionId1] = number_format((1 / ($n-1)) * $sum_col, 5);
15 }
```

**Gambar 5.52 Potongan *source code* untuk menghitung *leaving flow* dan *entering flow***

Gambar 5.53 menunjukkan potongan *source code* untuk mengetahui hubungan antar-alternatif apakah *comparable* atau *incomparable*. Kondisi untuk menentukan *comparable* atau *incomparable* sesuai dengan Persamaan (3.6).

```

1 $isComparable = true;
2 $condition =array();
3 foreach($selectionsId as $selectionIdA) {
4 foreach($selectionsId as $selectionIdB) {
5 if ($selectionIdA < $selectionIdB) {
6 $condition[$selectionIdA.".$selectionIdB] = array();
7 $condition[$selectionIdA.".$selectionIdB]["A Ii B"] = false;
8 $condition[$selectionIdA.".$selectionIdB]["A S+ B"] = false;
9 $condition[$selectionIdA.".$selectionIdB]["A S- B"] = false;
10 $condition[$selectionIdA.".$selectionIdB]["A Pi B"] = false;
11 $condition[$selectionIdA.".$selectionIdB]["A R B"] = true;
12 $condition1 = ($tabel_leaving[$selectionIdA] == $tabel_leaving[$selectionIdB]) &&
13 ($tabel_entering[$selectionIdA] == $tabel_entering[$selectionIdB]);
14 $condition2a = ($tabel_leaving[$selectionIdA] > $tabel_leaving[$selectionIdB]) ||
15 ($tabel_leaving[$selectionIdA] == $tabel_leaving[$selectionIdB]);
16 $condition2b = ($tabel_entering[$selectionIdA] < $tabel_entering[$selectionIdB]) ||
17 ($tabel_entering[$selectionIdA] == $tabel_entering[$selectionIdB]);
18 if ($condition1) {
19 $condition[$selectionIdA.".$selectionIdB]["A Ii B"] = true;
20 }
21 if ($condition2a) {
22 $condition[$selectionIdA.".$selectionIdB]["A S+ B"] = true;
23 }
24 if ($condition2b) {
25 $condition[$selectionIdA.".$selectionIdB]["A S- B"] = true;
26 }
27 if ($condition2a == $condition2b) {
28 $condition[$selectionIdA.".$selectionIdB]["A Pi B"] = true;
29 }
30 if ($condition[$selectionIdA.".$selectionIdB]["A Pi B"] || $condition[$selectionIdA.".
31 $selectionIdB]["A Ii B"]) {
32 $condition[$selectionIdA.".$selectionIdB]["A R B"] = false;
33 } else {
34 $isComparable = false;
35 }
36 }
37 }
38 }

```

**Gambar 5.53 Potongan *source code* untuk mengetahui hubungan antar-alternatif**

Gambar 5.54 menunjukkan potongan *source code* untuk meranking alternatif. Jika hubungan antar-alternatif semuanya *comparable* maka penentuan peringkat berdasarkan nilai *leaving flow*. Penentuan peringkat dengan mengurutkan nilai *leaving flow* dari besar ke kecil. Namun jika ada hubungan antar-alternatif yang *incomparable* maka perlu menghitung *net flow* dengan mengurangkan nilai *leaving flow* dan *entering flow*, seperti pada baris 17. Setelah diperoleh *net flow*, kemudian dilakukan penentuan peringkat dengan mengurutkan nilai *net flow* dari besar ke kecil.

```

1 $sortedSelection = array();
2 $rank = 1;
3 if ($isComparable) {
4 arsort($tabel_leaving);
5 foreach ($tabel_leaving as $key=>$value) {
6 $sortedSelection[] = $key;
7 $selection = $selectionsData[$key];
8 $selection->final_score = $value;
9 $selection->ranking = $rank;
10 $selection->status = "Selesai";
11 $selection->save();
12 $rank++;
13 }
14 } else {
15 $netflow = array();
16 foreach ($tabel_leaving as $key=>$value) {
17 $netflow[$key] = number_format($value - $tabel_entering[$key], 5);
18 }
19 arsort($netflow);
20 foreach ($netflow as $key=>$value) {
21 $sortedSelection[] = $key;
22 $selection = $selectionsData[$key];
23 $selection->final_score = $value;
24 $selection->ranking = $rank;
25 $selection->status = "Selesai";
26 $selection->save();
27 $rank++;
28 }
29 }

```

**Gambar 5.54 Potongan *source code* untuk meranking alternatif**

4. Implementasi proses menampilkan hasil peringkat peserta seleksi

Implementasi ini digunakan untuk menampilkan hasil peringkat berdasarkan proses perhitungan penilaian dengan PROMETHEE. Hasil peringkat ini dapat digunakan sebagai pertimbangan dalam menentukan peserta seleksi yang diterima. Implementasi halaman hasil peringkat peserta seleksi pelatihan ditunjukkan pada Gambar 5.55.

Gambar 5.56 menunjukkan potongan *source code* untuk menampilkan hasil peringkat peserta seleksi pelatihan. Halaman ini hanya dapat diakses oleh staf, kepala, kepala sun-bagian tata usaha, koordinator instruktur dan kepala kejuruan. Variabel \$result pada baris 6-18 merupakan *query* yang digunakan untuk menampilkan data peserta seleksi pelatihan dan diurutkan berdasarkan urutan peringkat seleksi.

Hasil Seleksi Peserta Pelatihan					
No	Nama Pendaftar	Sub-Kejuruan	Tanggal dan Waktu Seleksi	Nilai Seleksi	Ranking
1	Agus Widodo	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.485	1
2	Trio Adi Saputra	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.454	2
3	Anang Ma'ruf Martanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.441	3
4	Edi Sudarman	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.391	4
5	Moko Ginto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.366	5
6	Susanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.360	6
7	Ridwan Susanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.358	7
8	Zupriyanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.349	8
9	Mustakim	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.338	9
10	Bimas Wasono	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.309	10
11	Agus Hidayanto	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.300	11
12	Ryan Aji Pamungkas	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.281	12
13	Eko Samsudin	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.241	13
14	Rahmat Nur Islam	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.234	14
15	Rigunawan	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.228	15
16	Rajif Waluyo	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.226	16
17	Rino Anung K	Teknik Sepeda Motor	2016-04-25 dan 08:00	0.092	17

Gambar 5.55 Implementasi halaman hasil peringkat peserta seleksi

```

1 public function index()
2 {
3 $role_id = Auth::user()->roleId();
4 if ($role_id == 1 || $role_id == 3 || $role_id == 4 || $role_id == 5 ||$role_id == 6) {
5 $i = 0;
6 $result = Selection::select('selections.*', 'users.identity_number','users.name AS
7 name_registrant', 'selection_schedules.date',
8 'selection_schedules.time', 'sub_vocationals.name AS name_sub_vocational')
9 ->join('registrations', 'registrations.id', '=', 'selections.registration_id')
10 ->join('registrants', 'registrants.id', '=', 'selections.selection_id')
11 ->join('users', 'users.id', '=', 'registrants.user_id')
12 ->join('selection_schedules', 'selection_schedules.id', '=',
13 'selections.selection_schedule_id')
14 ->join('sub_vocationals', 'sub_vocationals.id', '=',
15 'selection_schedules.sub_vocational_id')
16 ->where('selections.status', '=', 'Selesai')
17 ->orderBy('selections.ranking', 'ASC')
18 ->get();
19 return view('result.index',compact('result','i'));
20 } else {
21 return redirect()->route('profile_users.show');
22 }
23 }

```

Gambar 5.56 Potongan *source code* untuk menampilkan hasil peringkat peserta seleksi pelatihan

## 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)**

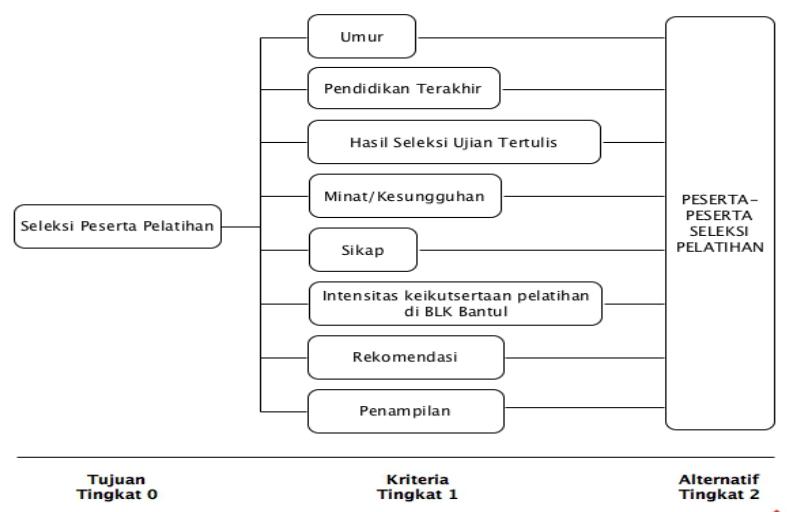
<b>Pembuat keputusan</b>		<b><i>Modified Delphi</i></b>	
<b>Kriteria</b>	<b>Subkriteria</b>	<b>Kriteria</b>	<b>Subkriteria</b>
Hasil seleksi ujian tertulis	-		Keterampilan teknis
Minat/kesungguhan	-		Pendidikan terakhir
Sikap	-		Keterampilan komunikasi
Intensitas keikutsertaan pelatihan di BLK Bantul	-	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 4.20. 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
		Kesan baik	0.023	0.012
Faktor eksternal	0.164	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)**

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

Berdasarkan data masa lalu (belum menggunakan metode MADM), 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. Perbedaan hasil disebabkan karena perbedaan kriteria yang digunakan dalam penilaian. 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.

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

**Tabel 6.5 (lanjutan)**

<b>Peringkat</b>	<b><i>Modified Delphi-AHP-PROMETHEE</i></b>	<b><i>Modified Delphi-AHP-TOPSIS</i></b>
5	P9	P5
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 (belum menggunakan metode MADM), 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.

Setelah hasil penelitian dengan berbagai pengujian dipaparkan dan dianalisis bersama pihak BLK Bantul, metode yang tepat dan dapat diterapkan di BLK Bantul untuk seleksi peserta pelatihan adalah metode *Modified Delphi-AHP-PROMETHEE*.

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1 Kesimpulan**

Kesimpulan yang dapat diberikan berdasarkan hasil penelitian dan pembahasan sesuai dengan tujuan penelitian yaitu menggabungkan metode *Modified Delphi-AHP-PROMETHEE* menghasilkan penilaian seleksi peserta pelatihan yang tepat dan dapat diterapkan di BLK Bantul. Dengan *Modified Delphi* menghasilkan 3 kriteria yaitu faktor intelektual, faktor personal dan faktor eksternal. Faktor intelektual mencakup 5 subkriteria, faktor personal mencakup 13 subkriteria dan faktor eksternal mencakup 3 subkriteria. Penggunaan metode AHP berhasil memperoleh bobot kriteria/subkriteria berdasarkan prioritasnya. Dan PROMETHEE dapat menghasilkan peringkat peserta seleksi yang tepat sebagai pertimbangan pembuat keputusan menentukan peserta seleksi yang diterima.

#### **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. 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
- 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 Imu 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, 1030–1040
- 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 programming approach for personnel selection process, *Problems and Perspectives in Management*, 2, 11, 98-108
- ÖZTÜRK, A., 2013, Personnel Selection In An Accommodation 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**