**Rekrutmen Pegawai**

**A. Algoritma WP :**

1.  Menentukan kriteria, sub kriteria dan alternatif keputusan

2.  Pengambil keputusan memberikan bobot preferensi atau tingkat kepentingan setiap kriteria

3.  Menyusun Rating Kecocokan setiap alternatif atas setiap Kriteria dalam tabel rating kecocokan

4.  Menyusun tabel rating kecocokan menjadi Matriks Keputusan

5. Menghitung Vektor S

6. Proses Perankingan dilakukan dengan cara menentukan nilai vektor V

7. Proses Perankingan kedua dilakukan dengan cara mengurutkan alternatif berdasarkan nilai V secara urut Turun mulai V yang paling besar.

8. Selesai

**B. Kriteria Benefit**

–      Pengalaman kerja (saya simbolkan C1)

–      Pendidikan (C2)

–      Usia (C3)

**C. Kriteria Cost**

–      Status perkawinan (C4)

–      Alamat (C5)

**D. Pra-Proses**

a. Kriteria Pembobotan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Calon Pegawai | kriteria | | | | |
| C1 | C2 | C3 | C4 | C5 |
| A1 | 0,5 | 1 | 0,7 | 0,7 | 0,8 |
| A2 | 0,8 | 0,7 | 1 | 0,5 | 1 |
| A3 | 1 | 0,3 | 0,4 | 0,7 | 1 |
| A4 | 0,2 | 1 | 0,5 | 0,9 | 0,7 |
| A5 | 1 | 0,7 | 0,4 | 0,7 | 1 |

b. Pembobotan

|  |  |  |
| --- | --- | --- |
| Kriteria | Bobot | Attribut |
| C1 | 0,3 | Benefit |
| C2 | 0,2 | Benefit |
| C3 | 0,2 | Benefit |
| C4 | 0,15 | Cost |
| C5 | 0,15 | Cost |
| Total | 1 |  |

**E. Normalisasi**

Rumus :

wij=wi∑wj

a. W1

w1 = 0,3/0,3+0,2+0,2+0,15+0,15 = 0,3

b. W2

w2 = 0,2/0,3+0,2+0,2+0,15+0,15 = 0,2

c. W3

w3 = 0,2/0,3+0,2+0,2+0,15+0,15 = 0,2

d. W4

w4 = 0,15/0,3+0,2+0,2+0,15+0,15 = 0,15

e. W5

w5 = 0,15/0,3+0,2+0,2+0,15+0,15 = 0,15

d. Menghitung Nilai Vektor S

1) S1

S1 = (0,5^0.14814814814815)(1^0.14814814814815)(0,7^  
0.37037037037037)(0,7^−0.18518518518519)(0,8^−0.14814814814815)

= 0.7162417485\*1\*0.8762517609\*1.068281108\*1.033610801 = 0.7

2) S2

S2 = (0,8^0.14814814814815)(0,7^0.14814814814815)(1^  
0.37037037037037)(0,5^−0.18518518518519)(1^−0.14814814814815)= 1

3) S3

S3 = (1^0.14814814814815)(0,3^0.14814814814815)(0,4^  
0.37037037037037)(0,7^−0.18518518518519)(1^−0.14814814814815)= 0.6

4) S4

S4 = (0,2^0.14814814814815)(1^0.14814814814815)(0,5^  
0.37037037037037)(0,9^−0.18518518518519)(0,7^−0.14814814814815)= 0.4

5) S5

S5 = (1^0.14814814814815)(0,7^0.14814814814815)(0,4^  
0.37037037037037)(0,7^−0.18518518518519)(1^−0.14814814814815)= 0.7

e. Menghitung Nilai Vektor V

1) V1 = 0.7/0,7+1+0,6+0,4+0,7 = 0.2058823529

2) V2 = 1/0,7+1+0,6+0,4+0,7 = 0.2941176471

3) V3 = 0,6/0,7+1+0,6+0,4+0,7 = 0.1764705882

4) V4 = 0,4/0,7+1+0,6+0,4+0,7 = 0.1176470588

5) V5 = 0,7/0,7+1+0,6+0,4+0,7 = 0.2058823529

**F. Perankingan Nilai Vektor V**

|  |  |  |
| --- | --- | --- |
| No | Calon Pegawai | Nilai Akhir |
| 1 | A2 | 0.2941176471 |
| 2 | A1 | 0.2058823529 |
| 3 | A5 | 0.2058823529 |
| 4 | A3 | 0.1764705882 |
| 5 | A4 | 0.1176470588 |

**G. Kesimpulan**

Maka calon pegawai yang memiliki nilai tertinggi dan bisa dipilih adalah calon pegawai A2 dengan nilai 0.2941176471.

**WP Using PHP**

a. Proses Fetch Data dari Database

<?php  
//--- query mengambil nilai data-data kriteria  
$sql="SELECT id\_criteria,criteria,weight  
      FROM wp\_criterias  
      ORDER BY id\_criteria";  
$result=$db->query($sql);  
//--- inisialisasi array kriteria 'C'  
$C=array();  
//--- inisialisasi array weight/bobot 'W'  
$W=array();  
//--- inisialisasi jumlah kriteria 'n'  
$n=0;  
while($criteria=$result->fetch\_assoc()){  
  $W[$id\_criteria]=$criteria['weight'];  
  $C[]=$criteria;  
  ++$n;  
}  
$result->free();  
  
//--- query mengambil nilai data-data hasil evaluasi  
$sql="SELECT id\_criteria,criteria,weight  
      FROM wp\_evaluations  
      ORDER BY id\_alternative,id\_criteria";  
$result=$db->query($sql);  
//--- inisialisasi array X  
$X=array();  
$alternative='';  
//--- inisialisasi jumlah alternative 'm'  
$m=0;  
while($row=$result->fetch\_assoc()){  
  if($row['id\_alternative']!=$alternative){  
    $X[$row['id\_alternative']]=array();  
    $alternative=$row['id\_alternative'];  
    ++$m;  
  }  
  $X[$row['id\_alternative']][$row['id\_criteria']]=$row['value'];  
}  
$result->free();  
?>

b. Normalisasi Bobot

<?php  
//--- menghitung total jumlah bobot  
$sigma\_w=array\_sum($W);  
//--- membagi masing-masing bobot dengan total jumlah bobot  
foreach($W as $j=>$w){  
  $W[$j]=$w/$sigma\_w;  
}  
?>

c. Menghitung Nilai Vektor S

<?php  
//--- inisialisasi array 'S'  
$S=array();  
//--- menghitung nilai preferensi S untuk tiap-tiap alternatif  
foreach($X as $i=>$x){  
  //--- inisialisasi nilai S untuk alternatif ke-i  
  $S[$i]=1;  
  //--- lakukan iterasi untuk tiap-tiap data hasil evaluasi X  
  foreach($x as $j=>$value){  
    //--- kalikan dengan pangkat negatif dari nilai untuk kriteria ke j  
    //--- jika merupakan kriteria biaya/cost  
    if($criterias[$j]['attribute']=='cost')  
      $S[$i]\*=pow($value,-$W[$j]);  
    //--- kalikan dengan pangkat positif dari nilai untuk kriteria ke j  
    //--- jika merupakan kriteria keuntungan/benefit  
    else  
      $S[$i]\*=pow($value,$W[$j]);  
  }  
}  
?>

d. Menghitung Nilai Vektor V

<?php  
//--- inisialisasi vektor V  
$V=array();  
//--- menghitung total jumlah nilai preferensi S  
$sigma\_s=array\_sum($S);  
//--- membagi masing-masing bobot dengan total jumlah nilai preferensi S  
foreach($S as $j=>$s){  
  $V[$j]=$s/$sigma\_s;  
}  
?>

e.Perankingan Nilai Vektor V

<?php  
//--- mengurutkan data nilai vektor V secara descending  
arsort($V);  
//--- menampilkan data secara terurut dari yang terbesar  
$rank=0;  
foreach($V as $i=>$v){  
echo ++$rank." V{$i} = {$v} <br>";  
}  
?>

Terimakasih