

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

Tahap Implementasi dan Pembahasan merupakan tahapan penerapan dari perancangan yang telah di buat. Tujuannya untuk melakukan uji coba pada aplikasi dan sebagai sarana pengolahan data dan penyajian informasi yang akan digunakan dalam proses pengambilan keputusan.

4.1. Implementasi Sistem

Sistem pendukung keputusan untuk penerima calon beasiswa ppa menggunakan metode electre berbasis web, pada bab ini akan dibahas mengenai implementasi dari perancangan sistem yang dibuat dengan menyertakan kode program.

4.1.1 Koneksi Database

Dalam proses koneksi ke database ini merupakan program untuk terhubung ke database dan membuat sebuah database yang bernama *db_beasiswa_ppa*. Program koneksi ke database bisa dilihat pada gambar 4.1

```
$host = 'localhost';
$user = 'id15895105_endri';
$pass = '<zo9r3bv7*R\}wob';
$dbname = 'id15895105_beasiswa';
try {
    $link = new PDO( "mysql:host=$host;dbname=$dbname", $user, $pass );
} catch( PDOException );
```

Gambar 4.1 Koneksi Database

Skrip pada gambar 4.1 merupakan program melakukan koneksi database. Program tersebut mempunyai beberapa variable, seperti *\$host*, *\$user*, *\$pass* dan *\$dbname*. Selanjutnya variable *\$link* yang berisi skrip untuk membuat database pada *mysql*.

4.1.2 Login

```
include "../connect.php";
try {
    $username = $_POST ['username'];
    $pass = $_POST ['pass'];
    $sql = "SELECT * FROM user WHERE username='$username' AND password='$pass'";
    $result = $link->prepare($sql);
    $result->execute();
    $data = $result->fetchAll(PDO::FETCH_ASSOC);
    if (!empty($data)){
        $_SESSION['username'] = $data[0]['username'];
        $_SESSION['role'] = $data[0]['role'];
        header("location:https://beasiswappa09stmikakakom.000webhostapp.com/index.php?module=home")
    }
}
```

Gambar 4.2 login

Pada gambar 4.2 diatas merupakan skrip program untuk login. Pada baris *include "../connect.php"* adalah program berfungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Yang dimana pada file tersebut tidak perlu untuk mengulangi skrip program yang telah di buat sebelumnya. *\$username=\$_POST['username'];* dan *\$pass=\$_POST['pass'];* adalah variable yang berisi value tersebut untuk melakukan pengambilan data yang telah di buat dan tersimpan di database, yaitu database *user*. Jika benar maka akan menuju pada halaman menu home dan jika data user tersebut salah maka akan menampilkan sebuah pesan kesalahan dan tetap pada halaman login.

4.1.3 Program Perhitungan Electre

```
$sql = "SELECT kriteria1 / (SELECT sqrt(sum(kriteria1 * kriteria1)) FROM alternatif) as matriks1,
kriteria2 / (SELECT sqrt(sum(kriteria2 * kriteria2)) FROM alternatif) as matriks2,
kriteria3 / (SELECT sqrt(sum(kriteria3 * kriteria3)) FROM alternatif) as matriks3,
kriteria4 / (SELECT sqrt(sum(kriteria4 * kriteria4)) FROM alternatif) as matriks4,
kriteria5 / (SELECT sqrt(sum(kriteria5 * kriteria5)) FROM alternatif) as matriks5,
kriteria6 / (SELECT sqrt(sum(kriteria6 * kriteria6)) FROM alternatif) as matriks6 FROM alternatif";
```

Gambar 4.3 Program Perhitungan Normalisasi Matrix

Pada tabel 4.3 adalah skrip untuk menghitung pada setiap alternatif yang memiliki nilai bobot. *\$sql* adalah sebuah variabel yang berguna untuk mengeksekusi program yang di panggil, lalu pada query *SELECT kriteria1 / (SELECT sqrt(sum(kriteria1*kriteria1)) FROM alternatif) as matriks1* berfungsi menghitung setiap baris pada kolom kriteria. Rumus yang di pakai adalah sqrt yaitu merupakan rumus dari perhitungan metode elektre. Seperti nilai kriteria mahasiswa pada baris 1 dan kolom 1 di kuadratkan, lalu hitung semua nilai mahasiswa pada baris 1 dan kolom 1 yang sudah di kuadratkan juga dan hasil dari semua nilai yang sudah di kuadratkan tersebut di jumlahkan. Selanjutnya nilai kriteria pada baris 1 dan kolom 1 di kalikan dengan semua nilai kriteria yang sudah di jumlahkan tadi.

```

array_push($bobotmatriks[$i], $row['matriks1'] * $bobot_kriteria[0]['bobot_kriteria'
]);
array_push($bobotmatriks[$i], $row['matriks2'] * $bobot_kriteria[1]['bobot_kriteria'
]);
array_push($bobotmatriks[$i], $row['matriks3'] * $bobot_kriteria[2]['bobot_kriteria'
]);
array_push($bobotmatriks[$i], $row['matriks4'] * $bobot_kriteria[3]['bobot_kriteria'
]);
array_push($bobotmatriks[$i], $row['matriks5'] * $bobot_kriteria[4]['bobot_kriteria'
]);
array_push($bobotmatriks[$i], $row['matriks6'] * $bobot_kriteria[5]['bobot_kriteria'
]);
$i ++;}

```

Gambar 4.4 Pembobotan Matrix yang telah di Normalisasi

Pada gambar 4.4 adalah sebuah program untuk menghitung nilai normalisasi langkah 2 yaitu menghitung dengan pembobotan mahasiswa. Selanjutnya program (*\$bobotmatriks[\$i], \$row ['matriks1'] * \$bobot_kriteria [0] ['bobot_kriteria']*); merupakan proses untuk pengambilan data dari variabel yang di panggil lalu di kalikan dengan bobot kriteria. Dengan kata lain yaitu menghitung nilai mahasiswa pada langkah 1 lalu di kalikan dengan bobot kriteria. Nilai mahasiswa pada baris 1 di kali dengan bobot kriteria pada kolom 1.

```

for ( $z = 1; $z < $baris; $z++ ) {
    $c = 1;
    for ( $i = 1; $i < $baris; $i++ ) {
        if ( empty( $concordance[$c] ) ) {
            $concordance[$c] = array(); }
        if ( empty($stampilanconcordance[$c])) {
            $stampilanconcordance[$c] = array(); }
        if ( empty($stampilandiscordance[$c])) {
            $stampilandiscordance[$c] = array(); }
        if ( empty( $discordance[$c] ) ) {
            $discordance[$c] = array(); }
    }
}
$y = 0;

```

Gambar 4.5 Program Menentukan Himpunan Cocordance dan Discordance-1

```

foreach ( $bobotmatriks[$i] as $colom ) {
  if ( $bobotmatriks[$z][$y] >= $bobotmatriks[$i][$y] ) {
    $y++;
    if ( $y == 1 ) {
      array_push( $concordance[$c], $bobot_kriteria[0]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 1 );      }
    if ( $y == 2 ) {
      array_push( $concordance[$c], $bobot_kriteria[1]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 2 );      }
    if ( $y == 3 ) {
      array_push( $concordance[$c], $bobot_kriteria[2]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 3 );      }
    if ( $y == 4 ) {
      array_push( $concordance[$c], $bobot_kriteria[3]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 4 );      }
    if ( $y == 5 ) {
      array_push( $concordance[$c], $bobot_kriteria[4]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 5 );      }
    if ( $y == 6 ) {
      array_push( $concordance[$c], $bobot_kriteria[5]['bobot_kriteria']);
      array_push( $stampilanconcordance[$c], 6 );      }
    array_push( $stampilandiscordance[$c], $y );
  }
}

```

Gambar 4.6 Program Menentukan Himpunan Cocordance dan Discordance-2

Pada gambar 4.5 dan 4.6 adalah program satu array penuh untuk perulangan dalam menentukan sebuah nilai. Terbagi menjadi dua dimensi, dimensi pertama untuk menentukan baris dan dimensi kedua untuk menentukan kolom. Pertama-tama pada program *if (\$bobotmatriks[\$z][\$y]* untuk menentukan variabel dari baris dan kolom, yaitu membandingkan setiap baris dan kolom pada variabel tersebut. Program *array_push(\$stampilanconcordance[\$c],1);* merupakan proses perbandingan antara baris satu dan baris dua, jika nilai pada baris satu/utama lebih besar dari baris dua/tamu maka akan bernilai satu. Selanjutnya program *array_push(\$stampilandiscordance[\$c], \$y);* berfungsi untuk proses bagian discordance, yaitu jika variabel pada concordance sudah memiliki nilai, maka bagian discordance hanya menerima nilai yang tidak ada pada concordance. Lebih

tepatnya pada potongan program ini menjelaskan untuk menentukan nilai yang terbesar pada perbandingan setiap baris. Seperti pada baris 1 sebagai perbandingan utama dan di bandingkan dengan baris 2 sebagai perbandingan tamu. Pada baris 1 kolom 1 di bandingkan dengan baris 2 kolom 1, jika pada baris 1 tersebut memiliki nilai yang lebih besar dari baris 2 kolom 1 maka baris 1 akan mempunyai nilai 1 atau true. Jika baris 1 kolom 1 lebih kecil dari baris 2 kolom 1 maka alternatif pada baris 1 tidak memiliki nilai 1 atau false.

```

array_push( $discordance[$c], abs( $bobotmatriks[$z][$y]
$bobotmatriks[$i][$y] ) );
array_push( $pembagi[$c], abs( $bobotmatriks[$z][$y] -$bobotmatriks[$i][$y] ) );
array_push( $stampilandiscordance[$c], $y );    }    }
    array_push( $langkah3['concordance'], array_sum( $concordance[$c] ) );
    array_push( $stampilconcordance, $stampilconcordance[$c] );
    array_push( $stampildiscordance, $stampilandiscordance[$c] );
    if ( max( $discordance[$c] ) != 0 ) {
        array_push( $langkah3['discordance'], max( $pembagi[$c] ) / max( $discordanc
e[$c] ) );
    } else {
array_push( $langkah3['discordance'], 0 );    }
    $c++;}

```

Gambar 4.7 Program Matrix Concordance dan Discordance

Pada gambar 4.7 adalah perhitungan untuk langkah 4, terdapat potongan program array untuk mengambil data concordance dan discordance pada langkah 3. Program `array_push($concordance[$c], $bobot_kriteria[0]['bobot_kriteria']);` merupakan proses eksekusi nilai pada concordance yang melibatkan dengan nilai bobot kriteria. Tujuannya untuk proses pada program (`$langkah3['concordance'], array_sum($concordance[$c]);`); yaitu proses perhitungan yang sudah memiliki nilai pada langkah 3 kemudian nilai pada langkah 3 tersebut dijadikan sebagai pokok acuan

untuk menjumlahkan nilai pada nilai bobot kriteria. Kemudian untuk proses perhitungan discordance yaitu pada program `array_push($langkah3['discordance'],max($pembagi[$c])/max($discordance[$c]));` merupakan proses menjumlahkan nilai discordance, nilai pada variabel langkah3 merupakan sebagai bahan pokok tujuan proses perhitungan, kemudian mengambil data discordance dan di beri nama variabel pembagi lalu di bagi dengan variabel discordance.

```

$c = array_sum( $langkah3['concordance'] ) / ( $jumlah_peserta*( $jumlah_peserta - 1 ) );
$d = array_sum( $langkah3['discordance'] ) / ( $jumlah_peserta*( $jumlah_peserta - 1 ) );
$concordance = array();
$discordance = array();
foreach ( $langkah3['concordance'] as $value ) {
    if ( $value >= round( $c ) ) {
        array_push( $concordance, 1 );
    } else {
        array_push( $concordance, 0 ); } }
foreach ( $langkah3['discordance'] as $value ) {
    if ( $value >= $d ) {
        array_push( $discordance, 1 );
    } else {
        array_push( $discordance, 0 ); } }
$pembobotanconcordance = array_chunk( $concordance, $jumlah_peserta - 1 );
$pembobotandiscordance = array_chunk( $discordance, $jumlah_peserta - 1 );
$langkah6 = array();

```

Gambar 4.8 Program Matrix Dominan Concordance dan Discordance

Pada gambar 4.8 adalah potongan program untuk menentukan nilai concordance dan discordance. Program `($langkah3['concordance']) / ($jumlah_peserta * ($jumlah_peserta - 1));` program ini adalah untuk menjumlahkan semua nilai matrix concordance pada langkah 4, proses pembagian dengan jumlah mahasiswa yang ada atau mendaftar dan di kali dengan jumlah

semua mahasiswa lalu di kurangi -1 poin, maka akan mendapatkan hasil nilai dari variabel tersebut. Proses penilaian pada langkah lima ini jika nilai dari langkah empat lebih besar nilai dari variabel c yaitu akan bernilai satu, jika lebih kecil maka akan bernilai nol.

```
$langkah6 = array();
$i = 0 ;
foreach ( $concordance as $value ) {
    array_push($langkah6, $value * $discordance[$i] );
    $i++; }

```

Gambar 4.9 Program Eliminasi Alternatif Less Favourable

Pada gambar 4.9 adalah sebuah potongan program menghitung nilai dari langkah 5 concordance dan discordance. Nilai concordance di kali dengan nilai discordance.

4.1.4 Program Upload

```
$target_dir = '../uploads/';
$target_file = $target_dir . basename( $_SESSION['username'].'.zip' );
$uploadOk = 1;
$filename = $_SESSION['username'].'.zip';
$username = $_SESSION['username'];
$imageFileType = strtolower( pathinfo( $target_file, PATHINFO_EXTENSION ) );
if ( isset( $_POST['submit'] ) ) {
    $check = getimagesize( $_FILES['fileToUpload']['tmp_name'] );
    if ( $check !== false ) {
        $uploadOk = 1;
    } else {
        $uploadOk = 0; } }
if ( $_FILES['fileToUpload']['size'] > 500000 ) {
    $uploadOk = 0; }
if ( $imageFileType != 'jpg' && $imageFileType != 'png' && $imageFileType !=
'zip'
&& $imageFileType != 'rar' ) {
    $uploadOk = 0; }

```

Gambar 4.10 Program Upload

Gambar 4.10 adalah program untuk proses upload. Proses untuk penyimpanan data upload ke database, yang berarti dalam program tersebut menjelaskan bahwa untuk jenis data yang di upload sesuai dengan mengikuti prosedur yang telah di perintahkan pada program tersebut. Program *if(move_uploaded_file(\$_FILES["fileToUpload"] ["tmp_name"], \$target_file))* merupakan program untuk meyimpan data yang akan di upload.

4.1.5 Program Hasil

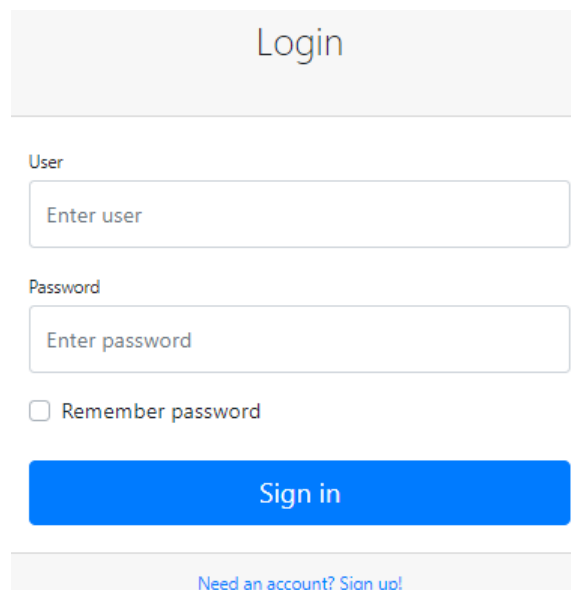
```
'nim' => $ kriteria ['nim'],
'nama' => $kriteria[$i]['nama_lengkap'],
'fakultas' => $ kriteria ['fakultas'],
'nilai' => array_sum($nilaiAkhir[$i]),
'kriteria1' => $kriteria[$i]['kriteria1'],
'kriteria2' => $kriteria[$i]['kriteria2'],
'kriteria3' => $kriteria[$i]['kriteria3'],
'kriteria4' => $kriteria[$i]['kriteria4'],
'kriteria5' => $kriteria[$i]['kriteria5'],
'kriteria6' => $kriteria[$i]['kriteria6']);
array_push($hasil, $array);
unset($array); }
function sortByOrder($a, $b){
if($a['nilai']==$b['nilai']) {
if($a['kriteria4']==$b['kriteria4']) {
if($a['kriteria3']==$b['kriteria4']) {
if($a['kriteria2']==$b['kriteria2']) {
if($a['kriteria1']==$b['kriteria1']) {
if($a['kriteria5']==$b['kriteria5']) {
if($a['kriteria6']==$b['kriteria6']) {
return 0;}
return $a['kriteria6'] < $b['kriteria6']?1:-1; }
return $a['kriteria5'] < $b['kriteria5']?1:-1; }
return $a['kriteria1'] < $b['kriteria1']?1:-1;}
return $a['kriteria2'] < $b['kriteria2']?1:-1;}
return $a['kriteria3'] < $b['kriteria3']?1:-1;}
return $a['kriteria4'] < $b['kriteria4']?1:-1;}
return $a['nilai'] < $b['nilai']?1:-1;}
usort($hasil, 'sortByOrder');
```

Gambar 4.11 Program Hasil

Pada gambar 4.11 adalah program untuk menampilkan data pada menu hasil, nilai yang di peroleh tersebut diambil dari nilai akhir perhitungan lalu di cari nilai yang terbesar sampai terkecil, kemudian jika terbatas kuota dan terdapat nilai yang sama besar pada batas kuota maka akan di lakukan perbandingan lagi pada data kriteria mahasiswa tersebut. Program tersebut untuk mencari nilai bobot yang terbesar.

4.2. Pembahasan

4.2.1 Halaman Login



The image shows a login interface with the following elements:

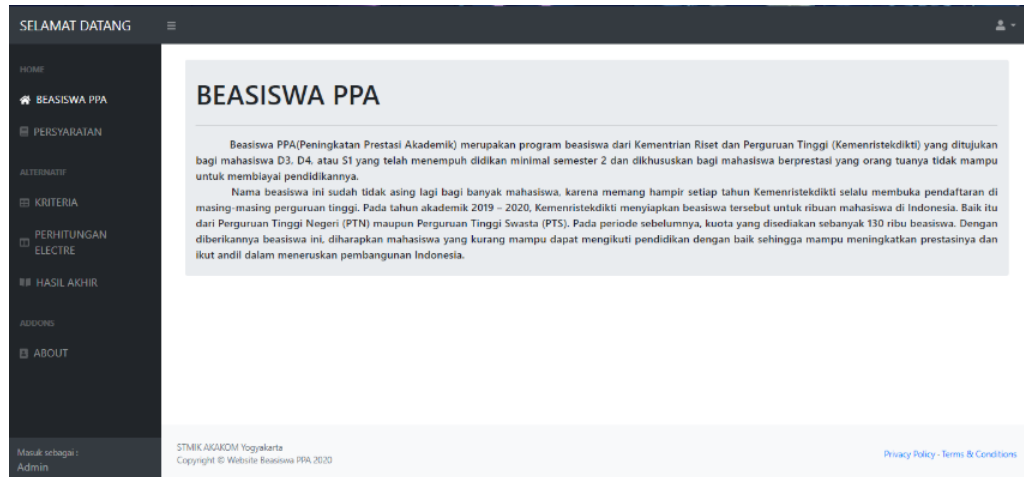
- A header box with the text "Login".
- A "User" label above a text input field containing the placeholder "Enter user".
- A "Password" label above a text input field containing the placeholder "Enter password".
- A checkbox labeled "Remember password".
- A prominent blue button labeled "Sign in".
- A footer box containing the text "Need an account? Sign up!".

Gambar 4.12 Login

Pada gambar 4.12 adalah sebuah tampilan login yang akan nantinya untuk bisa masuk ke website Beasiswa PPA.

4.2.2 Halaman Menu Utama / Home

Pada halaman menu utama / home merupakan tampilan awal setelah melakukan login. Hasilnya dapat dilihat pada gambar 4.13 :



Gambar 4.13 Tampilan Utama

Gambar 4.13 adalah tampilan untuk user admin. Jika sudah melakukan login, maka akan masuk ke tampilan home / utama. Yang dimana pada tampilan tersebut mempunyai beberapa menu dan setiap menu tersebut akan berbeda-beda pada user yang akan login.

4.2.3 Perhitungan Manual dan Hasil Akhir

no	Alternatif	Kriteria					
		IPK	KODE_PEKERJAA N	JML_TANGG UNGAN	PENGHASIL AN/TAHUN	Oragnisasi	Prestasi
1	ANITA	5	3	4	4	1	2
2	WAHYU	3	2	3	3	2	2
3	AMALIA RIZKI	4	4	4	4	2	1
4	NOHAN	3	4	4	3	2	1
5	DANA NUR	5	4	4	4	1	2
6	DONI	5	1	4	4	3	3
7	VITA	5	4	3	5	1	3
8	ULFA NUR	5	4	3	5	1	1
9	ROHMATUL	5	4	2	5	2	1
10	ANNISA	4	4	3	5	2	2
	W	5	4	5	5	4	4

Gambar 4.14 Data Bobot Mahasiswa

Pada gambar 4.14 merupakan data penilaian kriteria dari setiap alternatif.

	IPK	KODE_PEKER JAAN	JML_TANG GUNGAN	PENGHASILA N/TAHUN	Oragnisasi	Prestasi
R	0,35	0,27	0,37	0,30	0,17	0,32
	0,21	0,18	0,27	0,22	0,35	0,32
	0,28	0,36	0,37	0,30	0,35	0,16
	0,21	0,36	0,37	0,22	0,35	0,16
	0,35	0,36	0,37	0,30	0,17	0,32
	0,35	0,09	0,37	0,30	0,52	0,49
	0,35	0,36	0,27	0,37	0,17	0,49
	0,35	0,36	0,27	0,37	0,17	0,16
	0,35	0,36	0,18	0,37	0,35	0,16
	0,28	0,36	0,27	0,37	0,35	0,32

Gambar 4.15 Matriks Normalisasi

Pada gambar 4.15 adalah proses perhitungan langkah k-1, yang di mana pada proses ini merupakan hasil nilai kriteria dari setiap alternatif tersebut.

	IPK	KODE PEKERJAAN	JML TANGGUNGJAWABAN	PENGHASILAN/TAHUN	Oragnisasi	Prestasi
V	1,77	1,07	1,83	1,48	0,70	1,30
	1,06	0,71	1,37	1,11	1,39	1,30
	1,41	1,43	1,83	1,48	1,39	0,65
	1,06	1,43	1,83	1,11	1,39	0,65
	1,77	1,43	1,83	1,48	0,70	1,30
	1,77	0,36	1,83	1,48	2,09	1,95
	1,77	1,43	1,37	1,85	0,70	1,95
	1,77	1,43	1,37	1,85	0,70	0,65
	1,77	1,43	0,91	1,85	1,39	0,65
	1,41	1,43	1,37	1,85	1,39	1,30

Gambar 4.16 Pembobotan Yang Di Normalisasi

Gambar 4.16 merupakan proses perhitungan langkah ke-2, yaitu perhitungan dengan nilai bobot dari kriteria lalu di kali dengan nilai kriteria alternatif. Nilai bobot kriteria bisa di lihat pada gambar 4.14.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
CKL		{1, 2, 3, 4, 6}	{1, 3, 4, 6}	{1, 3, 4, 6}	{1, 3, 4, 5, 6}	{1, 2, 3, 4, }	{1, 3, 5, }	{1, 3, 5, 6}	{1, 3, 6}	{1, 3, 6}
	{5, 6}		{5, 6}	{1, 4, 5, 6}	{5, 6}	{2, }	{3, 5, }	{3, 5, 6}	{3, 5, 6}	{3, 5, 6}
	{2, 3, 4, 5, }	{1, 2, 3, 4, 5, }		{1, 2, 3, 4, 5, 6}	{2, 3, 4, 5, }	{2, 3, 4, }	{2, 3, 5, }	{2, 3, 5, 6}	{2, 3, 5, 6}	{1, 2, 3, 5, }
	{2, 3, 5, 6}	{1, 2, 3, 4, 5, }	{2, 3, 5, 6}		{2, 3, 5, }	{2, 3, }	{2, 3, 5, }	{2, 3, 5, 6}	{2, 3, 5, 6}	{2, 3, 5, }
	{1, 2, 3, 4, 5, 6}	{1, 2, 3, 4, 6}	{1, 2, 3, 4, 6}	{1, 2, 3, 4, 6}		{1, 2, 3, 4, }	{1, 2, 3, 5, }	{1, 2, 3, 5, 6}	{1, 2, 3, 6}	{1, 2, 3, 6}
	{1, 3, 4, 5, 6}	{1, 3, 4, 5, 6}	{1, 3, 4, 5, 6}	{1, 3, 4, 5, 6}	{1, 3, 4, 5, 6}		{1, 3, 5, 6}	{1, 3, 5, 6}	{1, 3, 5, 6}	{1, 3, 5, 6}
	{1, 2, 4, 5, 6}	{1, 2, 3, 4, 6}	{1, 2, 4, 6}	{1, 2, 4, 6}	{1, 2, 4, 5, 6}	{1, 2, 4, 6}		{1, 2, 3, 4, 5, 6}	{1, 2, 3, 4, 6}	{1, 2, 3, 4, 6}
	{1, 2, 4, 5, }	{1, 2, 3, 4, }	{1, 2, 4, 6}	{1, 2, 4, 6}	{1, 2, 4, 5, }	{1, 2, 4, }	{1, 2, 3, 4, 5, }		{1, 2, 3, 4, 6}	{1, 2, 3, 4, }
	{1, 2, 4, 5, }	{1, 2, 4, 5, }	{1, 2, 4, 5, 6}	{1, 2, 4, 5, 6}	{1, 2, 4, 5, }	{1, 2, 4, }	{1, 2, 4, 5, }	{1, 2, 4, 5, 6}		{1, 2, 4, 5, }
	{2, 4, 5, 6}	{1, 2, 3, 4, 5, 6}	{1, 2, 4, 5, 6}	{1, 2, 4, 5, 6}	{2, 4, 5, 6}	{2, 4, }	{2, 3, 4, 5, }	{2, 3, 4, 5, 6}	{2, 3, 4, 5, 6}	

Gambar 4.17 Matriks Himpunan Concordance

Gambar 4.17 yaitu perhitungan electre pada langkah ke-3 bagian concordance. Proses perhitungan ini membandingkan dari setiap alternatif. Lalu jika nilai pembanding lebih besar dari yang di banding maka akan bernilai, nilai tersebut di ambil dari perurutan nama kriteria, seperti kriteria IPK maka mempunyai nilai 1, kriteria kode pekerjaan orang tua maka bernilai dua dan seterusnya.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
DKL		{5, }	{2, 5, }	{2, 5, }	{2, }	{5, 6}	{2, 4, 6}	{2, 4, }	{2, 4, 5, }	{2, 4, 5, }
	{1, 2, 3, 4, }		{1, 2, 3, 4, }	{2, 3, }	{1, 2, 3, 4, }	{1, 3, 4, 5, 6}	{1, 2, 4, 6}	{1, 2, 4, }	{1, 2, 4, }	{1, 2, 4, }
	{1, 6}	{6}		{}	{1, 6}	{1, 5, 6}	{1, 4, 6}	{1, 4, }	{1, 4, }	{4, 6}
	{1, 4, 6}	{6}	{1, 4, }		{1, 4, 6}	{1, 4, 5, 6}	{1, 4, 6}	{1, 4, }	{1, 4, }	{1, 4, 6}
	{}	{5, }	{5, }	{5, }		{5, 6}	{4, 6}	{4, }	{4, 5, }	{4, 5, }
	{2, }	{2, }	{2, }	{2, }	{2, }		{2, 4, }	{2, 4, }	{2, 4, }	{2, 4, }
	{3, }	{5, }	{3, 5, }	{3, 5, }	{3, }	{3, 5, }		{}	{5, }	{5, }
	{3, 6}	{5, 6}	{3, 5, }	{3, 5, }	{3, 6}	{3, 5, 6}	{6}		{5, }	{5, 6}
	{3, 6}	{3, 6}	{3, }	{3, }	{3, 6}	{3, 5, 6}	{3, 6}	{3, }		{3, 6}
	{1, 3, }	{}	{3, }	{3, }	{1, 3, }	{1, 3, 5, 6}	{1, 6}	{1, }	{1, }	

Gambar 4.18 Matriks Himpunan Discordance

Pada gambar 4.18 merupakan proses perhitungan langkah ke-3 bagian discordance. Yang dimana pada proses ini adalah kebalikannya dari proses concordance pada gambar 4.17. Membandingkan nilai terkecil.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1		23	19	19	23	19	14	18	14	14
A2	8		8	18	8	4	9	13	13	13
A3	18	23		27	18	14	13	17	17	18
A4	17	23	17		13	9	13	17	17	13
A5	27	23	23	23		19	18	22	18	18
A6	23	23	23	23	23		18	18	18	18
A7	22	23	18	18	22	18		27	23	23
A8	18	19	18	18	18	14	23		23	19
A9	18	27	22	22	18	14	18	22		18
A10	17	27	22	22	17	9	18	22	22	

Gambar 4.19 Menghitung Matriks Concordance

Gambar 4.19 adalah proses perhitungan langkah ke-4 bagian concordance yaitu menjumlahkan nilai bobot kriteria.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1		0,98	1,00	0,98	1,00	1,00	0,57	0,76	1,00	1,00
A2	1,00		1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
A3	0,93	0,91		0,00	0,93	1,00	1,00	0,53	0,41	1,00
A4	1,00	0,91	1,00		1,00	1,00	1,00	1,00	0,81	1,00
A5	0,00	0,98	1	4,00		1,00	1,00	0,57	0,76	1,00
A6	0,51	0,50	0,82	0,82	0,77		0,77	0,77	0,82	1,00
A7	0,70	0,94	0,54	0,54	0,70	1,00		0,00	0,54	1,00
A8	1,00	0,94	1,00	0,94	1,00	1,00	1,00		1,00	1,00
A9	1,00	0,88	1,00	1,00	1,00	1,00	1,00	0,66		1,00
A10	0,66	0,00	0,70	0,62	0,66	0,65	0,93	0,51	0,54	

Gambar 4.20 Menghitung Matriks Discordance

Pada gambar 4.20 merupakan proses perhitungan langkah ke-4 bagian discordance yaitu menghitung nilai pada langkah ke-2 yang sebelumnya nilai tersebut di cocokkan dengan nilai perhitungan langkah ke-3 bagian discordance.

C	=	18								
	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1		1	1	1	1	1	0	1	0	0
A2	0		0	1	0	0	0	0	0	0
A3	1	1		1	1	0	0	0	0	1
A4	0	1	0		0	0	0	0	0	0
A5	1	1	1	1		1	1	1	1	1
A6	1	1	1	1	1		1	1	1	1
A7	1	1	1	1	1	1		1	1	1
A8	1	1	1	1	1	0	1		1	1
A9	1	1	1	1	1	0	1	1		1
A10	0	1	1	1	0	0	1	1	1	

Gambar 4.21 Matriks Dominan Concordance

Pada gambar 4.21 adalah proses perhitungan langkah ke-5 bagian concordance. Pada perhitungan ini yaitu menjumlahkan nilai pada matriks langkah ke-4 lalu di kali dengan nilai C yaitu menjumlahkan total pada matriks concordance

langkah ke-4 lalu di bagi dengan semua alternatif dan semua alternatif di kurangi 1 poin.

D	=	0,87									
	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	
A1		1	1	1	1	1	0	0	1	1	
A2	1		1	1	1	1	1	1	1	1	
A3	1	1		0	1	1	1	0	0	1	
A4	1	1	1		1	1	1	1	0	1	
A5	0	1	1	1		1	1	0	0	1	
A6	0	0	0	0	0		0	0	0	1	
A7	0	1	0	0	0	1		0	0	1	
A8	1	1	1	1	1	1	1		1	1	
A9	1	1	1	1	1	1	1	0		1	
A10	0	0	0	0	0	0	1	0	0		

Gambar 4.22 Matriks Dominan Discordance

Gambar 4.22 adalah proses perhitungan langkah ke-5 bagian discordance. Proses perhitungan ini sama halnya dengan proses perhitungan concordance pada langkah ke-5.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Total
A1		1	1	1	1	1	0	0	0	0	5
A2	0		0	1	0	0	0	0	0	0	1
A3	1	1		0	1	0	0	0	0	1	4
A4	0	1	0		0	0	0	0	0	0	1
A5	0	1	1	1		1	1	0	0	1	6
A6	0	0	0	0	0		0	0	0	1	1
A7	0	1	0	0	0	1		0	0	1	3
A8	1	1	1	1	1	0	1		1	1	8
A9	1	1	1	1	1	0	1	0		1	7
A10	0	0	0	0	0	0	1	0	0		1

Gambar 4.23 Matriks Eliminasi Alternatif Yang Less Favourable

Pada gambar 4.23 adalah matrik perhitungan langkah akhir yaitu eliminasi yang less favourable. Karena untuk batas kuota yang diterima pada beasiswa ppa ini ada sebanyak 8 orang, maka proses perhitungannya yaitu, hasil dimana pada

matriks agregat yang memiliki total nilai = 0 adalah kosong. Selanjutnya matriks agregat yang memiliki total nilai = 1 adalah A2, A4, A6 dan A10. Karena pada total nilai tersebut terdapat yang sama maka akan dilakukan perbandingan dengan nilai kriteria mahasiswa. Bisa di lihat pada gambar 4.13 data bobot mahasiswa. Pada alternatif A2, A4, A6 dan A10 mempunyai total nilai yang sama akan dibandingkan dengan nilai bobot kriteria mahasiswa, seperti kriteria pada penghasilan pertahun orangtua. Terdapat nilai bobot kriteria mahasiswa yang mempunyai nilai kriteria lebih kecil adalah alternatif A2 dan A4, maka di eliminasi. Bisa dilihat pada gambar 4.24 untuk proses perhitungan total nilai akhir di web.

	ANITA ANGGUNTARI	WAHYU JANUARI	AMALIA RIZKI PRADIKA	NOHAN ARYANG KAUTSAR	DANA NUR FIQI	DONI YANUAR KUSUMA PUTRA	VITA ARIYANA	ULFA NUR FAJRI MAHARANI	ROHMATUL UMMAH	ANNISA FAHMA AZIZA YEKTI	TOTAL
ANITA ANGGUNTARI		1	1	1	1	1	0	0	0	0	5
WAHYU JANUARI	0		0	1	0	0	0	0	0	0	1
AMALIA RIZKI PRADIKA	1	1		0	1	0	0	0	0	1	4
NOHAN ARYANG KAUTSAR	0	1	0		0	0	0	0	0	0	1
DANA NUR FIQI	0	1	1	1		1	1	0	0	1	6
DONI YANUAR KUSUMA PUTRA	0	0	0	0	0		0	0	0	1	1
VITA ARIYANA	0	1	0	0	0	1		0	0	1	3
ULFA NUR FAJRI MAHARANI	1	1	1	1	1	0	1		1	1	8
ROHMATUL UMMAH	1	1	1	1	1	0	1	0		1	7
ANNISA FAHMA AZIZA YEKTI	0	0	0	0	0	0	1	0	0		1

Gambar 4.24 Total Nilai Akhir

Dapat disimpulkan bahwa yang berhasil lolos pada perhitungan tersebut dan mendapat beasiswa ppa adalah mereka yang memiliki total nilai dimulai dari yang terbesar lalu terkecil dari hasil akhir perhitungan electre serta sudah termasuk kedalam batas kuota yang diterima. Bisa di lihat pada gambar 4.25 di bawah ini :

HASIL AKHIR

NO	NIM	NAMA MAHASISWA	FAKULTAS	KETERANGAN	NILAI AKHIR
1	8	ULFA NUR FAJRI MAHARANI	TI	Mendapatkan Beasiswa PPA	8
2	9	ROHMATUL UMMAH	MI	Mendapatkan Beasiswa PPA	7
3	5	DANA NUR FIQI	SI	Mendapatkan Beasiswa PPA	6
4	1	ANITA ANGGUNTARI	TI	Mendapatkan Beasiswa PPA	5
5	3	AMALIA RIZKI PRADIKA	TK	Mendapatkan Beasiswa PPA	4
6	7	VITA ARIYANA	SI	Mendapatkan Beasiswa PPA	3
7	10	ANNISA FAHMA AZIZA YEKTI	SI	Mendapatkan Beasiswa PPA	1
8	6	DONI YANUAR KUSUMA PUTRA	KA	Mendapatkan Beasiswa PPA	1

Pengumuman

Gambar 4.25 Menu Hasil Akhir

Pada gambar 4.25 adalah menu hasil akhir, yaitu tampilan yang berisikan nama-nama mahasiswa yang berhasil lolos pada tahap perhitungan electre dan akan mendapatkan beasiswa ppa.