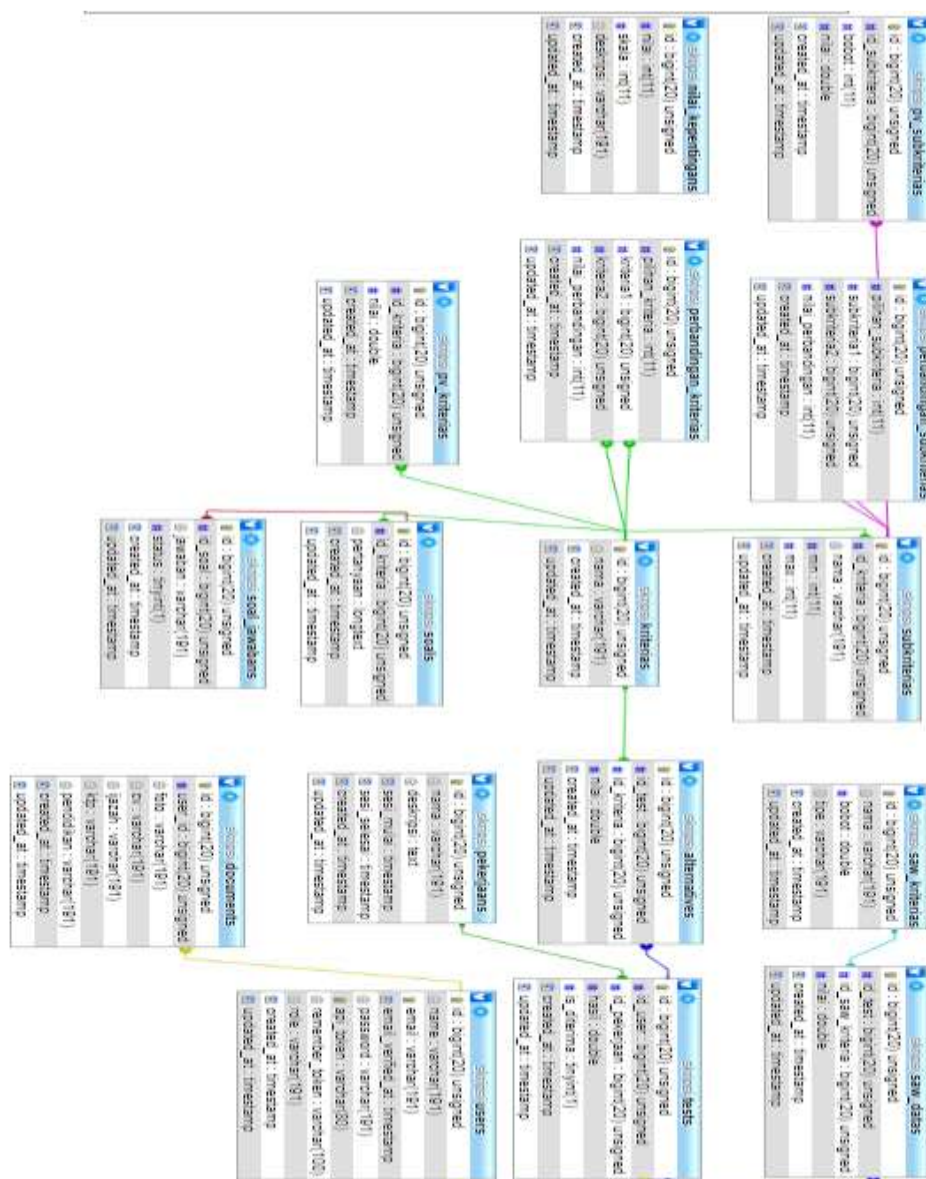


## BAB V. IMPLEMENTASI DAN PENGUJIAN

Pada bab ini membahas mengenai implementasi sistem dan pengujian dari perancangan sistem prediksi penjualan *spare part* yang telah dibahas pada bab 4.

### 5.1 Implementasi Database

Implementasi database dilakukan sesuai dengan perancangan yang telah dilakukan pada bab 4. Berikut adalah gambar sistem penyeleksi calon guru baru menggunakan metode saw dan dikombinasikan dengan ahp studi kasus di sekolah al – uswah bangil yang terdiri dari 16 tabel



Gambar 5.1 Impelentasi Databases

Terdapat 16 tabel yaitu *alternative*, *document*, *kriterias*, *nilai\_kepentingans*, *pekerjaans*, *perbandingan\_kriterias*, *perbandingan\_subkriterias*, *pv\_kriterias*, *pv\_subkriterias*, *saw\_datas*, *saw\_kriterias*, *soals*, *soal\_jawabans*, *subkriterias*, *test*, *users*.

### 1. Tabel Alternatif

Berikut adalah gambar implementasi database tabel *alternative*. terdiri dari *id*, *id\_test*, *id\_kriteria*, *nilai* tabel ini digunakan untuk menyimpan nilai dari setiap *alternative*.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
<input type="checkbox"/>	1 <b>id</b>	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
<input type="checkbox"/>	2 <b>id_test</b>	bigint(20)		UNSIGNED	Tidak	Tidak ada		
<input type="checkbox"/>	3 <b>id_kriteria</b>	bigint(20)		UNSIGNED	Tidak	Tidak ada		
<input type="checkbox"/>	4 <b>nilai</b>	double			Tidak	Tidak ada		
<input type="checkbox"/>	5 <b>created_at</b>	timestamp			Ya	NULL		
<input type="checkbox"/>	6 <b>updated_at</b>	timestamp			Ya	NULL		

Gambar 5.2 alternatif

### 2. Tabel document

Berikut adalah gambar implementasi database tabel *document*. Terdiri dari *id*, *user\_id*, *foto*, *cv*, *ijazah*, *ktp* tabel ini digunakan untuk menyimpan data *document*

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
<input type="checkbox"/>	1 <b>id</b>	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
<input type="checkbox"/>	2 <b>user_id</b>	bigint(20)		UNSIGNED	Tidak	Tidak ada		
<input type="checkbox"/>	3 <b>foto</b>	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
<input type="checkbox"/>	4 <b>cv</b>	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
<input type="checkbox"/>	5 <b>ijazah</b>	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
<input type="checkbox"/>	6 <b>ktp</b>	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
<input type="checkbox"/>	7 <b>pendidikan</b>	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
<input type="checkbox"/>	8 <b>created_at</b>	timestamp			Ya	NULL		
<input type="checkbox"/>	9 <b>updated_at</b>	timestamp			Ya	NULL		

Gambar 5.3 Document

### 3. Tabel Kriterias

Berikut adalah gambar implementasi database tabel kriterias. Terdiri dari id, nama tabel ini digunakan untuk menyimpan data kriteria.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nama	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	created_at	timestamp			Ya	NULL		
4	updated_at	timestamp			Ya	NULL		

Gambar 5.4 Kriteria

### 4. Tabel nilai\_kepentingan

Berikut adalah gambar implementasi database tabel nilai kepentingan. Terdapat id, nilai, skala, deskripsi table ini digunakan untuk menyimpan nilai kepentingan.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nilai	int(11)			Tidak	Tidak ada		
3	skala	int(11)			Tidak	Tidak ada		
4	deskripsi	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		

Gambar 5.5 Nilai Kepentingan

### 5. Tabel pekerjaans

Berikut adalah gambar implementasi database tabel pekerjaan. Terdapat id, id\_pekerjaan, nama\_pekerjaan tabel ini digunakan untuk menyimpan jenis pekerjaan.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_pekerjaan 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	nama_pekerjaan	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	created_at	timestamp			Ya	NULL		
5	updated_at	timestamp			Ya	NULL		

Gambar 5.6 Pekerjaan

## 6. Tabel perbandingan\_kriterias

Berikut adalah gambar implementasi database tabel perbandingan\_kriterias. Terdapat id, pilihan\_kriteria, kriterias1, kriteria2, nilai, nilai\_perbandingan, konversi tabel ini digunakan untuk menyimpan data perbandingan kriterias.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	pilihan_kriteria	int(11)			Tidak	Tidak ada		
3	kriteria1 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
4	kriteria2 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
5	nilai	int(11)			Tidak	Tidak ada		
6	nilai_perbandingan	int(11)			Tidak	Tidak ada		
7	konversi	int(11)			Tidak	Tidak ada		
8	created_at	timestamp			Ya	NULL		
9	updated_at	timestamp			Ya	NULL		

Gambar 5.7 Perbandingan Kriteria

## 7. Tabel perbandingan\_subkriterias

Berikut adalah gambar implementasi database tabel perbandingan subkriterias. Terdapat id, pilihan\_subkriteria, subkriteria1, subkriteria2, nilai\_perbandingan tabel ini digunakan untuk menyimpan data subkriteria.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	pilihan_subkriteria	int(11)			Tidak	Tidak ada		
3	subkriteria1 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
4	subkriteria2 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		
5	nilai_perbandingan	int(11)			Tidak	Tidak ada		
6	created_at	timestamp			Ya	NULL		
7	updated_at	timestamp			Ya	NULL		

Gambar 5.8 Perbandingan Subkriteria

## 8. Tabel pv\_kriterias

Berikut adalah gambar implementasi database tabel pv\_kroterias. Terdapat id\_kriteria, nilai tabel ini digunakan untuk menyimpan data pv\_kriterias.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id_kriteria	bigint(20)		UNSIGNED	Tidak	Tidak ada		
2	nilai	int(11)			Tidak	Tidak ada		
3	created_at	timestamp			Ya	NULL		
4	updated_at	timestamp			Ya	NULL		

Gambar 5.9 Pv Kriteria

#### 9. Tabel pv\_subkriterias

Berikut adalah gambar implementasi database tabel pv\_subkriterias. Terdapat id, id\_subkriterias, bobot, nilai tabel ini digunakan untuk menyimpan data pv\_subkriterias.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_subkriteria	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	bobot	int(11)			Tidak	Tidak ada		
4	nilai	double			Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		

Gambar 5.10 Pv Subkriteria

#### 10. Tabel saw\_datas

Berikut adalah gambar implementasi database tabel saw\_datas. Terdapat id, id\_test, id\_saw\_kriterias, nilai tabel ini digunakan untuk menyimpan data saw.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_test	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	id_saw_kriteria	bigint(20)		UNSIGNED	Tidak	Tidak ada		
4	nilai	double			Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		

Gambar 5.11 Saw Data

#### 11. Tabel saw\_kriterias

Berikut adalah gambar implementasi database tabel saw\_kriterias. Terdapat id, nama, bobot, tipe tabel ini digunakan untuk menyimpan data saw kriteria.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nama	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	bobot	double			Tidak	Tidak ada		
4	tipe	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		

Gambar 5.12 Saw Kriteria

## 12. Tabel soals

Berikut adalah gambar implementasi database tabel soals. Terdapat id, id\_kriteria, pertanyaan tabel ini digunakan untuk menyimpan data soal.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_kriteria 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	pertanyaan	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	created_at	timestamp			Ya	NULL		
5	updated_at	timestamp			Ya	NULL		

Gambar 5.13 Soal

## 13. Tabel soal\_jawabans

Berikut adalah gambar implementasi database tabel soal\_jawabans. Terdapat id, id\_soal, jawaban, status tabel ini digunakan untuk menyimpan data soal\_jawabans.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_soal 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	jawaban	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	status	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
5	created_at	timestamp			Ya	NULL		
6	updated_at	timestamp			Ya	NULL		

Gambar 5.14 Jawaban

## 14. Tabel subkriterias

Berikut adalah gambar implementasi database tabel subkriterias. Terdapat id, id\_kriteria, nama, min, max tabel ini digunakan untuk menyimpan nilai subkriterias.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_kriteria 🗨️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	nama	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	min	int(11)			Tidak	Tidak ada		
5	max	int(11)			Tidak	Tidak ada		
6	created_at	timestamp			Ya	NULL		
7	updated_at	timestamp			Ya	NULL		

Gambar 5.15 Subkriteria

## 15. Tabel test

Berikut adalah gambar implementasi database tabel test. Terdapat id, id\_user, hasil tabel ini digunakan untuk menyimpan data test.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	id_user 🗨️	bigint(20)		UNSIGNED	Tidak	Tidak ada		
3	hasil	double			Ya	NULL		
4	created_at	timestamp			Ya	NULL		
5	updated_at	timestamp			Ya	NULL		

Gambar 5.16 Test

## 16. Tabel users

Berikut adalah gambar implementasi database tabel users. Terdapat id, name, email, email\_verified\_at, password, api\_token, remember\_token, role tabel ini digunakan untuk menyimpan data user.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🔑	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	name	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	email 🗨️	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	email_verified_at	timestamp			Ya	NULL		
5	password	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
6	api_token 🗨️	varchar(80)	utf8mb4_unicode_ci		Ya	NULL		
7	remember_token	varchar(100)	utf8mb4_unicode_ci		Ya	NULL		
8	role	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
9	created_at	timestamp			Ya	NULL		
10	updated_at	timestamp			Ya	NULL		

Gambar 5.17 Users

## 5.2 Implementasi Sistem

Pada bagian ini, implementasi sistem berisikan aplikasi dan potongan kode program. Berikut adalah implementasi sistem penyeleksi calon guru baru menggunakan metode saw dan dikombinasikan dengan ahp studi kasus di sekolah al – uswah bangil.

### 5.2.1 *Source Code* Sistem

- ***Users***

1. **Pilih Lowongan Pekerjaan**

Pada menu lowongan pekerjaan berisikan *Source code* untuk memilih lowongan pekerjaan.

```

public function lowonganPekerjaan()
{
    // return view("users.index");
    return view('admin.lowongan');
}
public function Pekerjaan()
{
    // return view("users.index");
    $lowongan = Pekerjaan::all();
    $sudah_memilih = Test::where('id_user', Auth::user()->id)->first();
    return view('users.pilihPekerjaan', compact(
        [
            'lowongan',
            'sudah_memilih',
        ]
    ));
}

public function simpanLowongan(Request $request) {
    $this->validate($request,[
        'nama' => 'required',
        'deskripsi' => 'required',
    ]);

    $lowongan = Pekerjaan::create(
        [
            'nama' => $request->nama,
            'deskripsi' => $request->deskripsi,
        ]
    );

    return redirect('/lowongan');
}

```



- -----Tampilan-----

```

<div class="card card-primary">
  <div class="card-header">
    <h3 class="card-title">Pilih Lowongan Pekerjaan</h3>
  </div>
  <!-- /.card-header -->
  <!-- form start -->
  <form action="{{ url('/pilihPekerjaan/save') }}" method="post
">
    @csrf
    <div class="card-body">
      <div class="form-group">
        <label for="exampleInputEmail1">Lowongan Pekerjaaa
n</label>
        <select class="form-
control select2" name="id_pekerjaan">
          @foreach($lowongan as $low)
            <option value="{{ $low->id }}"><strong>{{ $low-
>nama }}</strong> - {{ $low->deskripsi }}</option>
          @endforeach
        </select>
      </div>
      <!-- /.card-body -->
      <div class="card-footer">
        <button type="submit" class="btn btn-
primary" {{ $sudah_memilih ? 'disabled' : " }}">Ikuti Tes</button>
        @if($sudah_memilih)
          <p class="text-
danger"><i>Anda sudah pernah memilih lowongan pekerjaan</i></p>
        @endif
      </div>
    </form>
  </div>

```

## 2. Document

Pada menu *document* berisikan *Source code* untuk memilih *documnet*.

```

public function store(Request $request)
{
    $this->validate($request, [
        'foto' => 'required|file|image|mimes:jpeg,png,jpg|max:2048',
        'cv' => 'required|file|image|mimes:jpeg,png,jpg|max:2048',

```

```

    'ijazah' => 'required|file|image|mimes:jpeg,png,jpg|max:2048',
    'ktp' => 'required|file|image|mimes:jpeg,png,jpg|max:2048',
    'pendidikan' => 'required',

]);
// menyimpan data file yang diupload ke variabel $file
$foto = $request->file('foto');
$cv = $request->file('cv');
$ijazah = $request->file('ijazah');
$ktp = $request->file('ktp');

// isi dengan nama folder tempat kemana file diupload
$tujuan_upload = 'data_file';

// upload file
$foto->move($tujuan_upload, $foto->getClientOriginalName());
$cv->move($tujuan_upload, $cv->getClientOriginalName());
$ijazah->move($tujuan_upload, $ijazah->getClientOriginalName());
$ktp->move($tujuan_upload, $ktp->getClientOriginalName());

$document = Document::updateOrCreate(
    [
        'user_id' => auth()->id(),
    ],
    [
        'foto' => $foto->getClientOriginalName(),
        'cv' => $cv->getClientOriginalName(),
        'ijazah' => $ijazah->getClientOriginalName(),
        'ktp' => $ktp->getClientOriginalName(),
        'pendidikan' => $request->pendidikan,
    ]
);
// return new DocumentResource($document);
return redirect('/index');
}

public function uploadGambar()
{
    // $dataGambar = uploadGambar::latest()->get();
    $document= Document::where('user_id', Auth::user()->id)->get();
    return view('users.index', compact(
        [
            'document',
        ]
    ));
}
-----tampilan-----

```

```

<form action="{ {url('/index/save') }}" method="POST" enctype="multipart/fo
rm-data">
    @csrf

    <div class="form-group">
        <label for="">Pendidikan Terakhir</label>
        <input type="text" class="form-
control" id="" name="pendidikan" placeholder="Masukan Pendidikan" require
d>
    </div>
    <div class="form-group">
        <label for="exampleInputFile">Foto</label>
        <div class="input-group">
            <div class="custom-file">
                <input type="file" name="foto" class="custom-file-
input" id="exampleInputFile" required>
                <label class="custom-file-
label" for="exampleInputFile">Choose file</label>
            </div>
            <div class="input-group-append">
                <span class="input-group-text">Upload</span>
            </div>
        </div>
    </div>
    <div class="form-group">
        <label for="exampleInputFile">Cv</label>
        <div class="input-group">
            <div class="custom-file">
                <input type="file" name="cv" class="custom-file-
input" id="exampleInputFile" required>
                <label class="custom-file-
label" for="exampleInputFile">Choose file</label>
            </div>
            <div class="input-group-append">
                <span class="input-group-text">Upload</span>
            </div>
        </div>
    </div>
    <div class="form-group">
        <label for="exampleInputFile">Ijazah</label>
        <div class="input-group">
            <div class="custom-file">
                <input type="file" name="ijazah" class="custom-file-
input" id="exampleInputFile" required>
                <label class="custom-file-
label" for="exampleInputFile">Choose file</label>
            </div>
            <div class="input-group-append">

```

```

        <span class="input-group-text">Upload</span>
    </div>
</div>
</div>
<div class="form-group">
    <label for="exampleInputFile">Ktp</label>
    <div class="input-group">
        <div class="custom-file">
            <input type="file" name="ktp" class="custom-file-
input" id="exampleInputFile" required>
            <label class="custom-file-
label" for="exampleInputFile">Choose file</label>
        </div>
        <div class="input-group-append">
            <span class="input-group-text">Upload</span>
        </div>
    </div>
</div>
<div class="card-footer">
    <button type="submit" value="simpan" class="btn btn-
primary">Submit</button>
</div>
</form>
<!-- ----- -->
<h4 class="my-5">Data</h4>

<table class="table table-bordered table-striped">
    <thead>
        <tr>
            <th width="1%">foto</th>
            <th width="1%">cv</th>
            <th width="1%">ijazah</th>
            <th width="1%">ktp</th>

            <th>pendidikan</th>
            <!-- <th width="1%">OPSI</th -->
        </tr>
    </thead>
    <tbody>
        @foreach($document as $g)
        <tr>
            <td></td>
            <td></td>
            <td></td>

```

```

        <td></td>
        <td>{{ $g->pendidikan }}</td>

        <!-- <td><a class="btn btn-
danger" href="index/upload/hapus/{{ $g->id }}">HAPUS</a></td> -->
    </tr>
    @endforeach
</tbody>
</table>

</div>
</div>

```

### 3. Test Tpa

Pada menu *test tpa* berisikan *Source code* untuk memilih *test tpa*.

```

public function store(Request $request){
    $this->validate($request,[
        'id_user'=>'required',
        'hasil'=>'required',

    ]);
    $test = Test::create([
        'id_user'=> $request->id_user,
        'hasil'=> $request->hasil,

    ]);
    return new TestResource($test);
}

public function hitungHasil() {
    $kriteria = app('App\Http\Controllers\Api\SawKriteriaController')->getInPercentage();

    $data = app('App\Http\Controllers\Api\SawDataController')->getNormalisasi();

    $tests = Test::with('user')->get();

    foreach ($tests as $key_test => $test) {
        $test->referensi = 0;
        $test->referensi_hitung = "";
        foreach ($data as $key_data => $d) {

```

```

        if ($d->id_test == $stest->id) {
            foreach ($kriteria as $key_kriteria => $krit) {
                if ($d->id_saw_kriteria == $krit->id) {
                    $stest->referensi_hitung = $stest-
>referensi_hitung . ' + ( ' . $krit->bobot_persen . ' * ' . $d-
>nilai_normalisasi . ')';
                    $stest->referensi += ($krit->bobot_persen * $d-
>nilai_normalisasi);
                }
            }
        }
    }
}

foreach ($tests as $test) {
    $test = Test::updateOrCreate(
        [
            'id_user'=> $test->id_user,
        ],
        [
            'hasil'=> $test->referensi,
        ]
    );
}

return $this->sortingTests($tests);
}

-----tampilan-----

<!-- Default box -->
@if($boleh_mengerjakan)
<form action="{{ url('/testTpa/selesai') }}" method="POST">
    @csrf
    <input hidden type="number" name="id_test" value="{{ $test->id }}">
    @foreach($kriteria as $krit)
    <div class="card">
        <div class="card-header">
            <h3 class="card-title"><strong>{{ $krit->nama }}</strong></h3>
            <div class="card-tools">
                <button type="button" class="btn btn-tool" data-card-
widget="collapse" title="Collapse">
                    <i class="fas fa-minus"></i>
                </button>
            </div>
        </div>
        <div class="card-body">
            <!-- -----Test Tpa----- -->

```

```

@php
$index = 0;
@endphp
@foreach($krit->soal as $soal)
@php
$index++;
@endphp
<div class="row mb-4">
  <div class="col">
    <strong>{{ $index }}. </strong>
    {{ $soal->pertanyaan }}
    @foreach($soal->soal_jawaban as $jawaban)
    <div class="form-check pl-5">
      <input type="radio" class="form-check-
input" name="user_jawaban{{ $soal->id }}" value="{{ $jawaban->id }}">
      <label class="form-check-
label" for="user_jawaban">{{ $jawaban->jawaban }}</label>
    </div>
    @endforeach
  </div>
</div>
@endforeach
<!-- /.card-body -->
<div class="card-footer">
</div>
<!-- /.card-footer-->
</div>
<!-- /.card -->
</div>
@foreach
<div class="card">
  <div class="card-body">
    <button type="button" class="btn btn-primary" data-
toggle="modal" data-target="#selesaiConfirm">
      Selesai
    </button>
  </div>
</div>

  <div class="modal fade" id="selesaiConfirm" tabindex="-
1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog" role="document">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-
title" id="exampleModalLabel">Selesai test</h5>

```

```

        <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
            <span aria-hidden="true">&times;</span>
        </button>
    </div>
    <div class="modal-body">
        Apakah anda yakin untuk menyelesaikan?
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-
dismiss="modal">Tidak</button>
        <button type="submit" class="btn btn-primary">Iya</button>
    </div>
</div>
</div>
</form>
@else
<div class="row">
    <div class="col">
        <p class="text-
danger"><i>Anda belum/tidak bisa mengerjakan tes TPA</i></p>
    </div>
</div>
@endif
</section>

```



- **Admin**

1. *Input* Pekerjaan

Pada menu *input* pekerjaan berisikan *Source code* untuk memasukan lowongan pekerjaan didalam sistem.

```

public function simpanPilihan(Request $request) {
    $this->validate($request,[
        'id_pekerjaan' => 'required',
    ]);

    $test = Test::create(
        [
            'id_user' => Auth::user()->id,
            'id_pekerjaan' => $request->id_pekerjaan,
        ]
    );

    $saw_kriteria = Saw_kriteria::all();

    foreach ($saw_kriteria as $krit) {
        $saw_data = Saw_data::updateOrCreate(
            [
                'id_test'=> $test->id,
                'id_saw_kriteria'=> $krit->id,
            ],
            [
                'nilai'=> null,
            ]
        );
    }

    return redirect('/testTpa');
}

-----tampilan-----

<div class="card card-primary">
    <div class="card-header">
        <h3 class="card-title">Input Lowongan Pekerjaan</h3>
    </div>
    <!-- /.card-header -->
    <!-- form start -->
    <form action="{{ url('/lowongan/save') }}" method="post">
        @csrf

```

```

<div class="card-body">
  <div class="form-group">
    <label for="exampleInputEmail1">Nama</label>
    <input type="name" name="nama" class="form-
control" id="exampleInputEmail1" placeholder="">
  </div>
  <div class="form-group">
    <label for="exampleInputEmail1">Deskripsi</label>
    <textarea name="deskripsi" class="form-
control" rows="5"></textarea>
  </div>
  <!-- /.card-body -->
  <div class="card-footer">
    <button type="submit" class="btn btn-
primary">Simpan</button>
  </div>
</form>
</div>

```

## 2. *Input* Soal

Pada menu *input* soal berisikan Source code untuk memasukan soal didalam sistem.

```

public function inputSoal()
{
  $kriteria = Kriteria::all();
  return view('admin.inputSoal', compact(
    [
      'kriteria',
    ]
  ));
}

public function saveSoal(Request $request) {
  $this->validate($request,[
    'id_kriteria'=> 'required',
    'pertanyaan'=> 'required',
    'jawaban_status'=> 'required',
    'jawaban'=> 'required',
  ]);
}

```

```

$soal = Soal::create([
    'id_kriteria'=> $request->id_kriteria,
    'pertanyaan'=> $request->pertanyaan,
]);

// jika jawaban yang dipilih adalah kosong, maka otomatis memilih jawaban pertama
if ($request->jawaban[$request->jawaban_status - 1] == "") {
    $request->jawaban_status = 1;
}

for ($i=0; $i < count($request->jawaban); $i++) {
    // store ke db hanya jawaban yang ada isinya
    if ($request->jawaban[$i] != "") {
        $jawaban = soal_jawaban::create([
            'id_soal'=> $soal->id,
            'jawaban'=> $request->jawaban[$i],
            'status'=> $i == ($request->jawaban_status - 1) ? 1 : 0,
        ]);
    }
}

return redirect('/inputSoal');
}

```

```

-----tampilan-----
<form action="{ url('/inputSoal/save') }" method="POST">
    @csrf
    <div class="card">
        <div class="card-header">
            <h3 class="card-title">Input Soal</h3>
            <div class="card-tools">
                <button type="button" class="btn btn-tool" data-card-
widget="collapse" title="Collapse">
                    <i class="fas fa-minus"></i>
                </button>
                <button type="button" class="btn btn-tool" data-card-
widget="remove" title="Remove">
                    <i class="fas fa-times"></i>
                </button>
            </div>
        </div>
        <div class="card-body">
            <!-- -----input soal----- -->
            <div class="row">

```

```

<div class="col-md">
  <div class="form-group">
    <label>Mata Pelajaran</label>
    <select class="form-control select2" name="id_kriteria">
      @foreach($kriteria as $krit)
        <option value="{{ $krit->id }}">{{ $krit-
>nama }}</option>
      @endforeach
    </select>
  </div>
  <div class="form-group">
    <label>Masukan Pertanyaan</label>
    <textarea name="pertanyaan" rows="5" class="form-
control" required></textarea>
  </div>
  @for ($i = 1; $i <= 5; $i++)
    <div class="form-row mb-2">
      <div class="col-md-1" style="padding-left: 30px;">
        <input class="form-check-
input" type="radio" name="jawaban_status" value="{{ $i }}" required>
        <label class="form-check-
label" for="jawaban_status"><strong>Pilihan {{ $i }}</strong></label>
      </div>
      <div class="col-md-11">
        <input type="text" class="form-
control" name="jawaban[]" placeholder="{{ $i < 3 ? 'Masukkan Pilihan ' . $i : '
Kosong jika tidak ada pilihan ' . $i }}" {{ $i < 3 ? 'required' : " }}">
      </div>
    </div>
  @endfor
</div>
</div>
<!-- /.card-body -->
<div class="card-footer">
  <button type="submit" class="btn btn-primary">Simpan</button>
</div>

</div>

```

### 3. *Input* Nilai

Pada menu *input* nilai berisikan *Source code* untuk memasukan nilai didalam sistem.

```
public function inputNilai()
{
    $saw_kriteria = Saw_kriteria::all();
    $tests = Test::with(['saw_data','saw_data.saw_kriteria','user'])->get();
    $tidak_lengkap = Saw_data::whereNull('nilai')->first();
    $hasil = Test::with(['user'])->orderBy('hasil', 'desc')->get();
    return view('admin.inputNilai', compact([
        'saw_kriteria',
        'tests',
        'tidak_lengkap',
        'hasil',
    ]));
}

public function updateNilai(Request $request) {
    $this->validate($request,[
        'id'=> 'required',
        'nilai'=> 'required',
    ]);

    for ($i=0; $i < count($request->id); $i++) {
        $saw_data = Saw_data::find($request->id[$i]);

        $saw_data->nilai = $request->nilai[$i];

        $saw_data->save();
    }

    return redirect('/inputNilai');
}

public function hitungNilai() {
    $tests = app('App\Http\Controllers\Api\TestController')->hitungHasil();

    return redirect('/inputNilai');
}
```

-----tampilan-----

<section class="content">

```

<!-- Default box -->
<div class="card">
  <div class="card-header">
    <h3 class="card-title">Tampilan Input Data Nilai</h3>

    <div class="card-tools">
      <button type="button" class="btn btn-tool" data-card-
widget="collapse" title="Collapse">
        <i class="fas fa-minus"></i>
      </button>
      <button type="button" class="btn btn-tool" data-card-
widget="remove" title="Remove">
        <i class="fas fa-times"></i>
      </button>
    </div>
  </div>
  <div class="card-body">
    <!-- -----input nilai----- -->

    <section class="content">
      <div class="container-fluid">
        <div class="row">
          <div class="col-12">
            <div class="card">
              <div class="card-header">
                <h3 class="card-title">Memasukan Data Nilai </h3>
              </div>
              <!-- /.card-header -->
              <div class="card-body">
                <table id="example2" class="table table-bordered table-
hover">
                  <thead>
                    <tr>
                      <th>Nama</th>
                      @foreach($saw_kriteria as $krit)
                      <th>Nilai {{ $krit->nama }}</th>
                      @endforeach
                      <th>Edit Nilai</th>
                    </tr>
                  </thead>
                  <tbody>
                    @foreach($tests as $test)
                    <tr>
                      <td>{{ $test->user->name }}</td>
                      @foreach($test->saw_data as $data)
                      <td>{{ $data->nilai ? round($data->nilai, 2) : '-
' }}</td>

```

```

        @endforeach
        <td>
            <button type="button" class="btn btn-
default" data-toggle="modal" data-target="#editModal{{ $test->user->id }}">
                <i class="fas fa-pencil-alt"></i>
                Edit
            </button>
        </td>
    </tr>

    <!-- Modal -->
    <div class="modal fade" id="editModal{{ $test-
>user->id }}" tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-hidden="true">
        <div class="modal-dialog" role="document">
            <form action="{{ url('/inputNilai/edit') }}" me
thod="POST">

                @csrf
                <div class="modal-content">
                    <div class="modal-header">
                        <h5 class="modal-
title" id="exampleModalLabel">Nilai {{ $test->user->name }}</h5>
                        <button type="button" class="close" da
ta-dismiss="modal" aria-label="Close">
                            <span aria-
hidden="true">&times;</span>
                        </button>
                    </div>
                    <div class="modal-body">
                        @foreach($test->saw_data as $data)
                            <input hidden type="number" name="i
d[]" value="{{ $data->id }}">

                            <div class="form-row mb-2">
                                <div class="col-md-6">
                                    <label class="form-check-
label" for="saw_kriteria"><strong>{{ $data->saw_kriteria-
>nama }}</strong></label>

                                    </div>
                                    <div class="col-md-6">
                                        <input type="number" step="0.01
" class="form-control" name="nilai[]" value="{{ $data->nilai }}" {{ $data-
>id_saw_kriteria == 1 ? 'readonly' : '' }}>
                                        </div>
                                    </div>
                                </div>
                            @endforeach
                        </div>
                    <div class="modal-footer">

```

```

                <button type="button" class="btn btn-
secondary" data-dismiss="modal">Kembali</button>
                <button type="submit" class="btn btn-
primary">Simpan</button>
            </div>
        </div>
    </form>
</div>
</div>
<!-- /.modal -->
@endforeach
</tbody>
</table>
</div>
<!-- /.card-body -->
</div>
<!-- /.card -->

</div>
<!-- /.col -->
</div>
<!-- /.row -->
</div>
<!-- /.container-fluid -->
</section>
<div class="card-footer">
    @if($tidak_lengkap)
        <button class="btn btn-primary" disabled>Hitung</button>
        <p class="text-
danger"><i>Lengkapi semua nilai terlebih dahulu</i></p>
    @else
        <a class="btn btn-
primary" href="{{ url('/inputNilai/hitung') }}">Hitung</a>
    @endif
</div>
<!-- /.card-footer-->
</div>
</div>
<!-- /.card -->

</section>
<!-- /.content -->

<!-- Main content -->
<section class="content">

    <!-- Default box -->
    <div class="card">

```



```

<div class="card-header">
  <h3 class="card-title">Tampilan Hasil Perangkingan</h3>

  <div class="card-tools">
    <button type="button" class="btn btn-tool" data-card-
widget="collapse" title="Collapse">
      <i class="fas fa-minus"></i>
    </button>
    <button type="button" class="btn btn-tool" data-card-
widget="remove" title="Remove">
      <i class="fas fa-times"></i>
    </button>
  </div>
</div>
<div class="card-body">
  <!-- -----input nilai----- -->

  <table id="example2" class="table table-bordered table-hover">
    <thead>
      <tr>
        <th>Peringkat</th>
        <th>Nama</th>
        <th>Nilai</th>
      </tr>
    </thead>
    <tbody>
      @foreach($hasil as $data)
      <tr>
        <td>{{ $loop->index + 1 }}</td>
        <td>{{ $data->user->name }}</td>
        <td>{{ $data->hasil ? $data->hasil : '-' }}</td>
      </tr>
      @endforeach
    </tbody>
  </table>
</div>
<!-- /.row -->
</div>
<!-- /.container-fluid -->
</section>

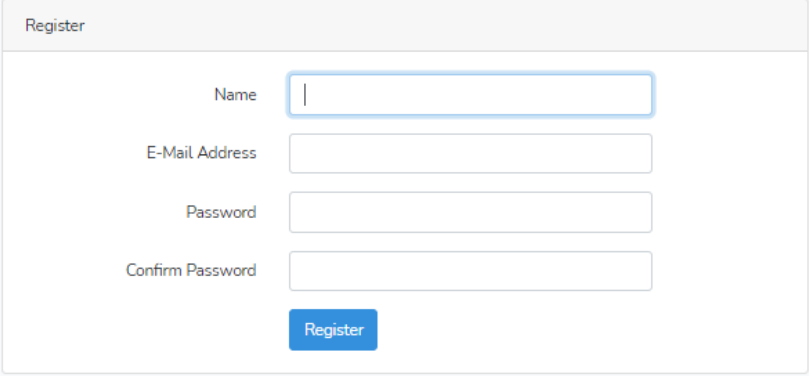
```

### 5.2.2 Tampilan Antarmuka Sistem

Tampilan antarmuka sistem adalah tampilan dari aplikasi sistem penyeleksi calon guru baru menggunakan metode saw dan dikombinasikan dengan ahp studi kasus di sekolah al – uswah bangil. Berikut adalah tampilan antamuka sistem:

- **Peserta**

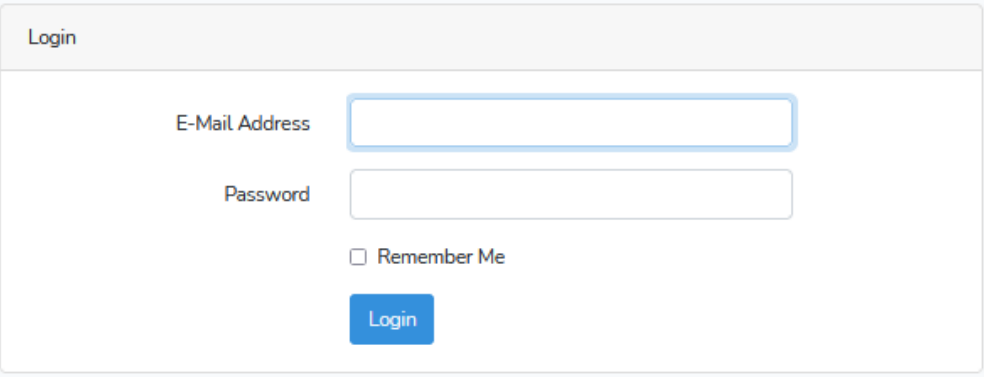
1. *Register*



The image shows a web form titled "Register". It contains four input fields: "Name", "E-Mail Address", "Password", and "Confirm Password". Below the fields is a blue "Register" button. The "Name" field is currently active, indicated by a blue border and a cursor.

Gambar 5.18 Tampilan Rigister Peserta

2. *Login*



The image shows a web form titled "Login". It contains two input fields: "E-Mail Address" and "Password". Below the fields is a checkbox labeled "Remember Me" and a blue "Login" button. The "E-Mail Address" field is currently active, indicated by a blue border.

Gambar 5.19 Tampilan Login Peserta

### 3. *Input data Diri*

The screenshot shows a web interface titled 'Document' with a 'Home / Blank Page' link. Below the title is an 'Input Document' section. A blue bar at the top of this section says 'Upload Data diri'. The form contains several fields: 'Pendidikan Terakhir' with a text input and a 'Browse' button; 'Foto' with a 'Choose file' button, a 'Browse' button, and an 'Upload' button; 'cv' with a 'Choose file' button, a 'Browse' button, and an 'Upload' button; 'Ijazah' with a 'Choose file' button, a 'Browse' button, and an 'Upload' button; and 'NIP' with a 'Choose file' button, a 'Browse' button, and an 'Upload' button. A 'Submit' button is located at the bottom left. A black box with white text 'Tidak ada file yang dipilih' is overlaid on the 'Foto' field.

Gambar 5.20 Tampilan Dashboard

The screenshot shows a table titled 'Data' with five columns: 'foto', 'cv', 'ijazah', 'nip', and 'pendidikan'. The 'foto' column displays a profile picture of a woman wearing a white hijab. The 'cv' column shows a resume document with a red header. The 'ijazah' column shows a diploma document with a green header. The 'nip' column shows a blue document. The 'pendidikan' column contains the text 'SMA'.

Gambar 5.21 Tampilan Data Document

### 4. *Pilih Lowongan Pekerjaan*

The screenshot shows a web interface titled 'Lowongan Pekerjaan' with a 'Home / Blank Page' link. Below the title is a 'Lowongan Pekerjaan' section. A blue bar at the top of this section says 'Pilih Lowongan Pekerjaan'. Below this is a search bar containing the text 'Guru - Bahasa Indonesia - pendidikan terakhir minimal sma (2021-09-26 11:12:00 - 2021-09-28 06:50:00)'. Below the search bar is a blue 'Search' button. Below the search bar is a red message that says 'Anda sudah pernah memilih lowongan pekerjaan'.

Gambar 5.22 Tampilan Lowongan Pekerjaan

## 5. Test Tpa

**Bahasa Indonesia**

1. Pada upacara upacara resmi seperti menyambut hari ulang tahun kemerdekaan, Perdana upacara berpidato dengan menggunakan bahasa Indonesia. [

Bahasa resmi keragaman

Bahasa pengantar dalam dunia pendidikan

Alat lempingan pembangun nasional

Alat pengembangan kebudayaan

Alat pertukangan

2. Penyajian buku pelajaran, penilaian buku atau pemerjemahan, dilakukan dalam bahasa Indonesia. Pernyataan di atas menunjukkan fungsi bahasa Indonesia sebagai ?

Bahasa resmi keragaman

bahasa pengantar dalam dunia pendidikan

Alat lempingan pembangun nasional

Alat pengembangan kebudayaan

Buletin tertulis yang mempunyai kekuatan hukum

Gambar 5.23 Tampilan *Test Tpa*

## 6. Hasil Test

Home Contact Login

**Hasil Test** Home / Hasil Page

Selamat, Anda dinyatakan lolos!!

Gambar 5.24 Hasil *Test*

- **Admin**

1. *Login*

Login

E-Mail Address

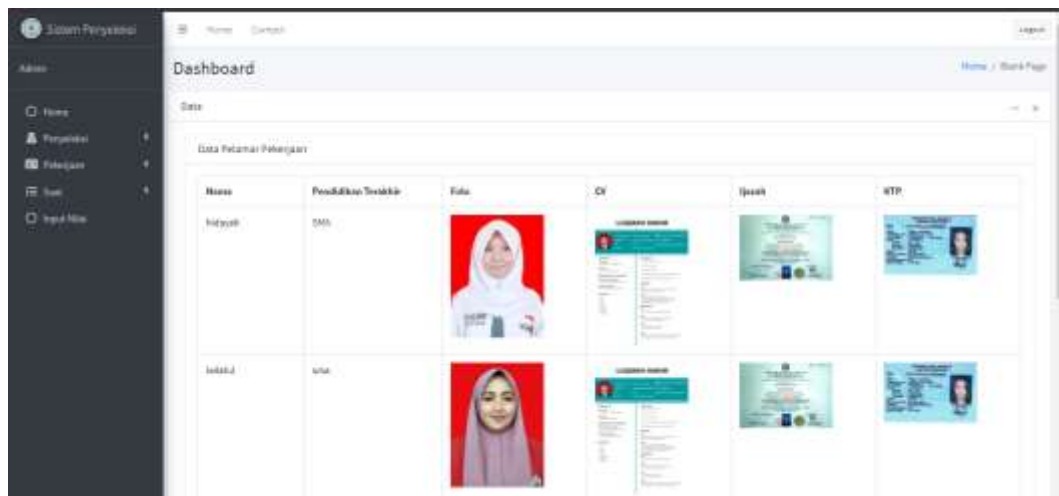
Password

Remember Me

Login

Gambar 5.25 Tampilan Login admin

## 2. Dashboard



Gambar 5.26 Tampilan Dashboard Admin

## 3. Input Pekerjaan

The screenshot shows the 'Input Pekerjaan' form. It includes a title bar 'Input Pekerjaan' and a sub-header 'Lowongan Pekerjaan'. The form contains the following fields:

- Name:** A text input field.
- Deskripsi:** A large text area for job description.
- Seal Mulai:** A date-time picker field with a calendar icon.
- Seal Selesai:** A date-time picker field with a calendar icon.
- Simpan:** A blue button to save the job listing.

Gambar 5.27 Tampilan *Input* Pekerjaan

## 4. List Pekerjaan

The screenshot shows the 'List Pekerjaan' table. It has columns for Name, Description, Start Date, End Date, and Action. The table contains three rows of job listings.

Nama	Deskripsi	Seal Mulai	Seal Selesai	
Rura - Bahasa Indonesia	penelitian terapan minimal 2000	2022-08-20 12:00:00	2022-08-28 08:30:00	
guru agama Islam	mengetahui agama Islam	2022-08-20 08:00:00	2022-08-22 08:00:00	
guru - bahasa	Mengetahui bahasa Inggris - Jawa	2022-08-21 08:00:00	2022-08-22 08:00:00	

Gambar 5.28 *List* Pekerjaan

## 5. *Input Soal Tpa*

Gambar 5.29 Tampilan *Input Soal*

## 6. *List Soal*

Kategori	Pertanyaan	
Bahasa Inggris	What are the main and the water table?	<a href="#">Edit</a> <a href="#">Hapus</a>
Bahasa Inggris	What does the ocean consist?	<a href="#">Edit</a> <a href="#">Hapus</a>
PK	Berikut ini merupakan jenis-jenis yang dimiliki oleh kelompok Protista adalah...	<a href="#">Edit</a> <a href="#">Hapus</a>
PK	Suatu yang hasil pelaksanaan organisasi kehidupan dari yang sederhana menuju kompleks adalah...	<a href="#">Edit</a> <a href="#">Hapus</a>
PK	Bahwa satu faktor dalam proses interaksi sosial adalah kesetaraan sesama yang sama dengan yang lain. Itu adalah faktor...	<a href="#">Edit</a> <a href="#">Hapus</a>
PK	Interaksi sosial dapat terjadi jika memenuhi dua persyaratan utama, yaitu...	<a href="#">Edit</a> <a href="#">Hapus</a>
Matematika	$0,23 + 0,05 =$	<a href="#">Edit</a> <a href="#">Hapus</a>
Matematika	$80 \times 0,25 = 1/4 =$	<a href="#">Edit</a> <a href="#">Hapus</a>
Bahasa Indonesia	Pada upacara upacara resmi seperti pernikahan hari ulang tahun kemerdekaan, pertama upacara berpidato dengan menggunakan bahasa Indonesia?	<a href="#">Edit</a> <a href="#">Hapus</a>
Bahasa Indonesia	Perjanjian bulat bulat perjanjian perburuan buaya atau perampokan, dilakukan dalam bahasa Indonesia. Perjanjian di atas menunjukkan fungsi bahasa Indonesia sebagai?	<a href="#">Edit</a> <a href="#">Hapus</a>

Gambar 5.30 *List Soal*

## 7. *Input Nilai*

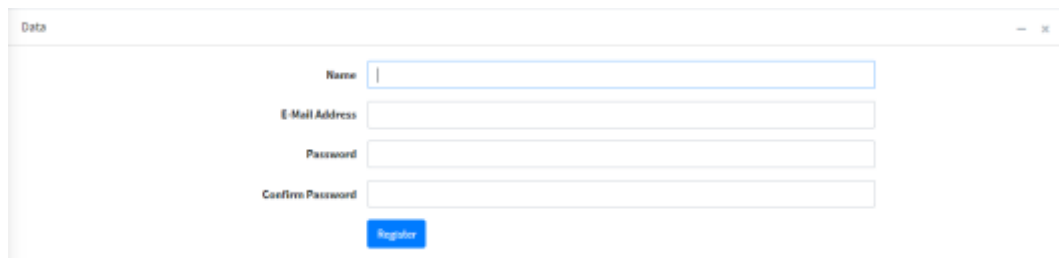
Informasi Data Nilai					
Nama	Nilai TPA	Nilai RTG	Nilai PUK	Nilai Absensi	Nilai Nilai
Hikmah	7,50	80	90	80	<a href="#">Edit</a>
Ahli	11,50	80	80	80	<a href="#">Edit</a>
Ahli	11,50	80	80	80	<a href="#">Edit</a>
ah	11,50	80	80	80	<a href="#">Edit</a>

Tampilan Hasil Penyelesaian			
Peringkat	Nama	Nilai	Keterangan
1	ah	8	100%
2	ahli	8307500748288	100%

Gambar 5.31 Tampilan *Input Nilai*

## 8. Register Penyeleksi



A screenshot of a web form titled "Data" for registering a reviewer. The form contains four input fields: "Name", "E-Mail Address", "Password", and "Confirm Password". Below the fields is a blue "Register" button.

Gambar 5.32 Registrasi Penyeleksi

## 9. List Penyeleksi



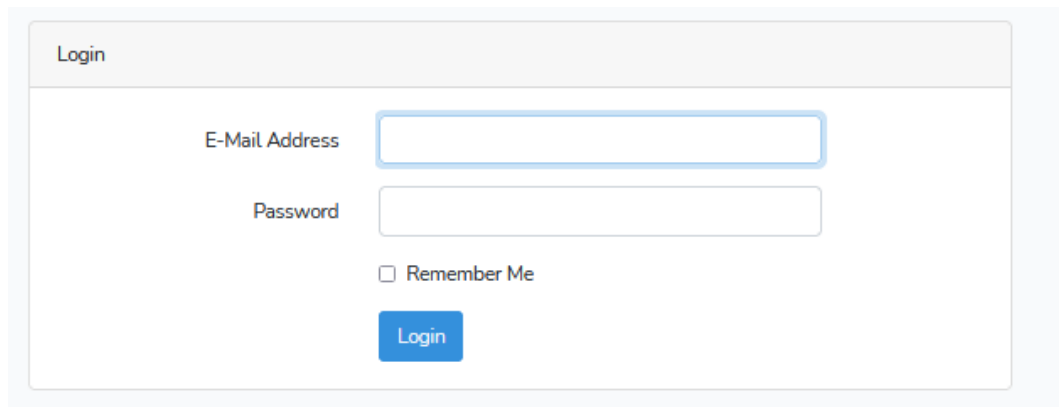
A screenshot of a web page titled "Penyeleksi" showing a list of reviewers. The page has a header "Penyeleksi" and a sub-header "Data Penyeleksi". Below is a table with two columns: "Name" and "Email". The first row shows "Arka" and "arka@gmail.com". There are "Edit" and "Delete" buttons next to the row.

Name	Email
Arka	arka@gmail.com

Gambar 5.33 List Penyeleksi

- **Penyeleksi**

### 1. Login



A screenshot of a web form titled "Login" for a reviewer. The form contains two input fields: "E-Mail Address" and "Password". Below the fields is a checkbox labeled "Remember Me" and a blue "Login" button.

Gambar 5.34 Login Penyeleksi

## 2. *Input Pekerjaan*

Gambar 5.35 *Input Pekerjaan*

## 3. *List Pekerjaan*

Gambar 5.36 *List Pekerjaan*

## 4. *Input Soal Tpa*

Gambar 5.37 *Input Soal Tpa*



## 5. List Soal Tpa

Kategori	Pertanyaan	✓	✗
Bahasa Inggris	What are the man and the woman talking?	✓	✗
Bahasa Inggris	What does the woman express?	✓	✗
PK	Berikut ini merupakan jenjang peng-ilmian salah satu jenjang Profesi adalah...	✓	✗
PK	Kawalan yang hasil pelaksanaan organisasi berdasarkan dari yang sudah ada maka itu kompleks adalah...	✓	✗
PK	Salah satu faktor dalam proses interaksi sosial adalah kesetaraan kesetaraan untuk sama dengan yang lain. Itu adalah faktor...	✓	✗
PK	Interaksi sosial dapat terjadi jika memenuhi dua persyaratan utama, yaitu...	✓	✗
Matematika	$0,225 + 0,005x$	✓	✗
Matematika	$40, 3,875, 1, 1/2$	✓	✗
Bahasa Indonesia	Pada upacara upacara resmi seperti pernikahan hari ulang tahun kemerdekaan, Peristiwa upacara berpidato dengan menggunakan bahasa Indonesia?	✓	✗
Bahasa Indonesia	Perjaya buku pelajaran penulisan buku atau penempatan, dilakukan dalam bahasa Indonesia. Perjaya di akan menuliskan luring bahasa Indonesia sebagai?	✓	✗

Gambar 5.38 List Soal Tpa

## 6. Input Nilai

Pencatatan Penilaian					
Tampilkan Seluruh Data Penilaian Penilaian					
Informasi Data Nilai					
Nama	Nilai TPA	Nilai RTG	Nilai Multikel	Nilai Manajemen	Nilai Nilai
Hidayati	7,37	30	30	30	✓
Adi	11,50	30	30	30	✓
Adina	11,50	30	40	30	✓
ad	13,80	30	40	30	✓

Tampilkan Hasil Penyeleksi			
Peringkat	Nama	Nilai	Keputusan
1	ad	1	✓
2	Adi	0,34052007448218	✗

Gambar 5.39 Input Nilai Penyeleksi

### 5.3 Pengujian

Pengujian merupakan proses untuk menentukan apakah hasil dari tugas akhir sudah sesuai dengan kebutuhan sistem dan berjalan sesuai lingkungan yang diinginkan. Pengujian dapat berupa pengujian fungsional, penerimaan pengguna, ataupun performa sistem. Pada pengujian dipaparkan secara detail mengenai metode pengujian, tujuan pengujian, proses pengujian serta analisa hasil pengujian.

#### 5.3.1 Pengujian Fungsional Sistem

Pengujian fungsionalitas sistem dilakukan dengan menggunakan *Black Box Testing*. Berikut adalah tabel pengujian fungsionalitas sistem dari sistem pendukung keputusan.

- **Admin**

Tabel 5.1 Pengujian Fungsional Admin

No	Fungsional	Scenario Pengujian	Hasil yang Diharapkan	Hasil yang Didapat
1	<i>Login</i>	Memasukkan <i>email</i> dan <i>password</i> kemudian pilih <i>button login</i> .	Sistem akan menampilkan halaman utama.	sesuai
2	<i>Input Lowongan Pekerjaan</i>	Memilih jenis pekerjaan dan memasukan detail pekerjaan kemudian klik <i>button submit</i> .	Sistem akan menyimpan lowongan pekerjaan.	sesuai
3	<i>Input Soal Tpa</i>	Memasukan soal dan jawaban, menentukan jawaban yang benar kemudian klik <i>button submit</i> .	Sistem akan menyimpan soal tpa yang akan di ujikan kepada peserta.	sesuai
4	<i>Input Nilai</i>	Memasukan nilai btq,wawancara,	Sistem akan menampilkan	sesuai

		psikotest dan menekan button hitung untuk melakukan perhitungan dan mengetahui siapa guru terbaik	hasil nilai dari setiap data pelamar pekerjaan.	
5	<i>Register</i> Penyeleksi	Memasukan nama, <i>email, password</i>	Sistem akan menyimpan data akun penyeleksi	sesuai

- **Penyeleksi**

Tabel 5.2 Pengujian Fungsional Penyeleksi

No	Fungsional	Scenario Pengujian	Hasil yang Diharapkan	Hasil yang Didapat
1	<i>Login</i>	Memasukkan <i>email</i> dan <i>password</i> kemudian pilih <i>button login</i> .	Sistem akan menampilkan halaman utama.	sesuai
2	<i>Input</i> Lowongan Pekerjaan	Memilih jenis pekerjaan dan memasukan detail pekerjaan kemudian klik <i>button submit</i> .	Sistem akan menyimpan lowongan pekerjaan.	sesuai
3	<i>Input</i> Soal Tpa	Memasukan soal dan jawaban, menentukan jawaban yang benar kemudian klik <i>button submit</i> .	Sistem akan menyimpan soal tpa yang akan di ujikan kepada Peserta.	sesuai

- **Peserta**

Tabel 5.3 Pengujian Fungsional Peserta

No	Fungsional	Scenario Pengujian	Hasil yang Diharapkan	Hasil yang Didapat
1	<i>Login</i>	Memasukkan <i>email</i> dan <i>password</i> kemudian pilih <i>button login</i> .	Sistem akan menampilkan halaman utama.	sesuai
2	<i>Register</i>	Memasukan <i>username</i> <i>email</i> dan <i>password</i> kemudian pilih <i>button submit</i>	Sistem akan menyimpan data peserta dan melakukan login untuk bisa mengakses halaman <i>dashboard</i>	sesuai
3	<i>Input Document</i>	Memasukan data yang terdapat di halaman <i>document</i> kemudian pilih <i>button submit</i>	Sistem akan menyimpan data <i>document</i> .	sesuai
4	Lowongan Pekerjaan	Memilih lowongan pekerjaan kemudian pilih <i>button submit</i>	Sistem menyimpan data pekerjaan yang dilamar oleh peserta	sesuai
5	<i>Test Potensial Akademik</i>	Melakukan <i>test</i> tpa dan memilih jawan yang benar kemudian menekan tombol <i>next</i> untuk berlanjut ke soal berikutnya dan <i>back</i> untuk kembali ke soal sebelumnya.	Sistem menampilkan soal tpa dan memilih jawaban mulai dari A hingga E kemudian melanjutkan ke soal berikutnya hingga selesai.	sesuai

## 5.3.2 Hasil Angket Penilaian Sistem Penyeleksi Guru

## • Admin

Tabel 5.4 Hasil Angket Admin

No.	PERNYATAAN	SKOR				
		SS	S	CS	TS	STS
1.	apakah fungsi <i>login</i> berjalan dengan normal	1				
2.	apakah fungsi <i>register</i> untuk penyeleksi berjalan dengan normal	1				
3.	apakah fungsi <i>input</i> pekerjaan berjalan dengan normal		1			
4.	apakah fungsi <i>input</i> soal tpa berjalan dengan normal		1			
5	apakah fungsi hitung nilai berjalan dengan normal	1				
6	apakah fungsi <i>logout</i> berjalan dengan normal	1				
7	apakah aplikasi sudah memenuhi kebutuhan anda?	1				
<b>Jumlah</b>		5	2	0	0	0
<b>Jumlah Skor</b>		25	8	0	0	0
<b>∑Skor</b>		33				
<b>Presentase</b>		91,25				

Jumlah skor observasi merupakan hasil dari penjumlahan skor masing-masing pernyataan hasil observasi yang dikalikan bobot skor menurut skala Likert. Skor maksimal merupakan skor maksimal pada skala likert yang dikalikan dengan jumlah soal, sehingga  $5 \times 8 = 40$ . Jumlah Skor yang diharapkan adalah jumlah skor observasi yang dikalikan dengan jumlah responden, sehingga  $40 \times 1 = 40$ . Perhitungan persentase kelayakan dari data admin (tabel 5.4) menggunakan rumus berikut:

$$\sum \text{skor observasi} = (\text{jumlah} \times \text{skor SS}) + (\text{jumlah} \times \text{skor S}) + (\text{jumlah} \times \text{skor CS}) + (\text{jumlah} \times \text{skor TS}) + (\text{jumlah} \times \text{skor STS})$$

$$\sum \text{skor observasi} = (5 \times 5) + (2 \times 4) + (0 \times 3) + (0 \times 2) + (0 \times 1)$$

$$\sum \text{skor observasi} = 33$$

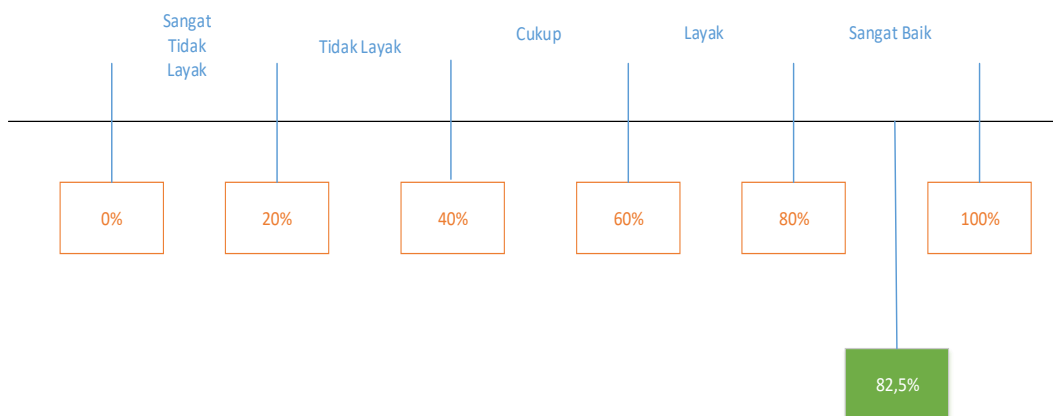
Sedangkan persentase kelayakan dari admin adalah sebagai berikut:

$$\text{Persentase kelayakan} = \frac{\sum \text{skor observasi}}{\text{skor yang diharapkan}} \times 100 \%$$

$$\text{Persentase kelayakan} = 33 / 40 = 0,825 \times 100 \% = 82,5\%$$

$$\text{Persentase kelayakan} = 82,5 \%$$

Total skor observasi dari data admin dengan jumlah 33 (82,5 %) dari skor yang diharapkan yaitu 40 (100%). Berdasarkan kriteria pada tabel kelayakan menurut Arikunto, Suharsimi & Safruddin A.J, Persentase total skor tersebut termasuk dalam kategori Sangat Layak. Penyajian skala sesuai persentase total skor menurut (Arikunto, Suharsimi & Safruddin A.J, Cepi, 2009) secara detail dapat digambarkan seperti gambar 5.28 berikut:



Gambar 5.40 Gambar Presentase Hasil Admin

Berikut ini adalah data saran perbaikan yang diberikan oleh admin terangkum dalam Tabel 5.5

Tabel 5.5 Saran Admin

No	Respon	Saran
1	Admin 1	sebaiknya sistem dapat juga menyeleksi calon murid baru

- **Penyeleksi**

Tabel 5.6 Hasil Angket Penyeleksi

No.	PERNYATAAN	SKOR				
		SS	S	CS	TS	STS
1.	apakah fungsi <i>login</i> berjalan dengan normal	1				
2.	apakah fungsi <i>input</i> pekerjaan berjalan dengan normal	1				
4.	apakah fungsi <i>input</i> soal tpa berjalan dengan normal		1			
6	apakah fungsi <i>logout</i> berjalan dengan normal		1			
7	apakah aplikasi sudah memenuhi kebutuhan anda?	1				
<b>Jumlah</b>		3	2	0	0	0
<b>Jumlah Skor</b>		15	8	0	0	0
<b><math>\Sigma</math>Skor</b>		23				
<b>Presentase</b>		92				

Jumlah skor observasi merupakan hasil dari penjumlahan skor masing-masing pernyataan hasil observasi yang dikalikan bobot skor menurut skala Likert. Skor maksimal merupakan skor maksimal pada skala likert yang dikalikan dengan jumlah soal, sehingga  $5 \times 5 = 25$ . Jumlah Skor yang diharapkan adalah jumlah skor observasi yang dikalikan dengan jumlah responden, sehingga  $25 \times 1 = 25$ . Perhitungan persentase kelayakan dari data admin (tabel 5.6) menggunakan rumus berikut:

$$\Sigma \text{skor observasi} = (\text{jumlah} \times \text{skor SS}) + (\text{jumlah} \times \text{skor S}) + (\text{jumlah} \times \text{skor CS}) + (\text{jumlah} \times \text{skor TS}) + (\text{jumlah} \times \text{skor STS})$$

$$\Sigma \text{skor observasi} = (3 \times 5) + (2 \times 4) + (0 \times 3) + (0 \times 2) + (0 \times 1)$$

$$\Sigma \text{skor observasi} = 23$$

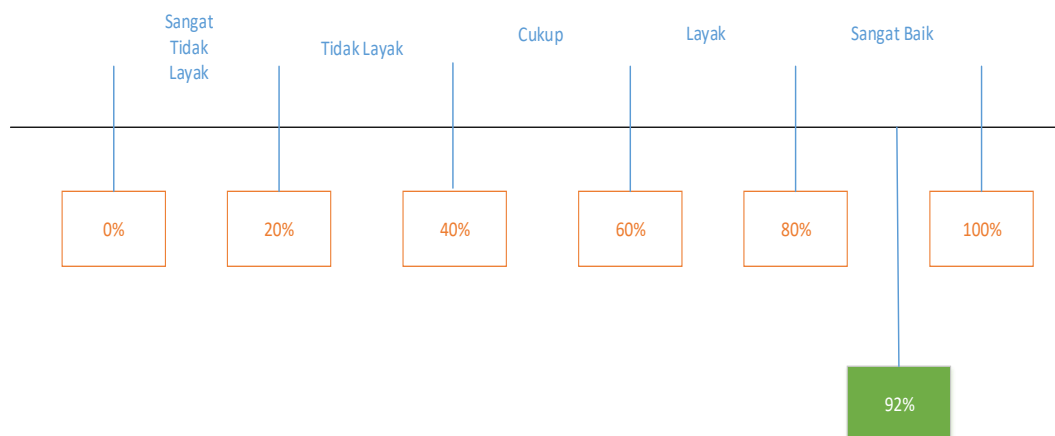
Sedangkan persentase kelayakan dari admin adalah sebagai berikut:

$$\text{Persentase kelayakan} = \frac{\Sigma \text{skor observasi}}{\text{skor yang diharapkan}} \times 100 \%$$

$$\text{Persentase kelayakan} = \frac{23}{25} = 0,92 \times 100 \% = 92\%$$

$$\text{Persentase kelayakan} = 92 \%$$

Total skor observasi dari data admin dengan jumlah 23 (92%) dari skor yang diharapkan yaitu 25 (100%). Berdasarkan kriteria pada tabel kelayakan menurut Arikunto, Suharsimi & Safruddin A.J, Persentase total skor tersebut termasuk dalam kategori Sangat Layak. Penyajian skala sesuai persentase total skor menurut (Arikunto, Suharsimi & Safruddin A.J, Cepi, 2009) secara detail dapat digambarkan seperti gambar 5.29 berikut:



Gambar 5.41 Hasil Presentase Penyeleksi

Berikut ini adalah data saran perbaikan yang diberikan oleh penyeleksi terangkum dalam Tabel 5.7

Tabel 5.7 Saran Penyeleksi

No	Respon	Saran
1	Ovi Yulita	sebaiknya sistem dapat juga menyeleksi calon murid baru

- **Peserta**

Tabel 5.8 Hasil Angket Peserta

No.	PERNYATAAN	SKOR				
		SS	S	CS	TS	STS
1.	apakah fungsi <i>login</i> berjalan dengan normal	4	1			
2.	apakah fungsi <i>register</i> untuk pengguna berjalan dengan normal	5				
3.	apakah fungsi <i>input document</i> berjalan dengan normal	4	1			



4	apakah fungsi <i>input</i> pekerjaan berjalan dengan normal	4	1			
5	apakah fungsi <i>test</i> tpa berjalan dengan normal	3	2			
6	apakah fungsi logout berjalan dengan normal	4	1			
7	apakah aplikasi sudah memenuhi kebutuhan anda?	5				
<b>Jumlah</b>		29	6	0	0	0
<b>Jumlah Skor</b>		145	24	0	0	0
<b>∑Skor</b>		169				
<b>Presentase</b>		96				

Jumlah skor observasi merupakan hasil dari penjumlahan skor masing-masing pernyataan hasil observasi yang dikalikan bobot skor menurut skala Likert. Skor maksimal merupakan skor maksimal pada skala likert yang dikalikan dengan jumlah soal, sehingga  $7 \times 5 = 35$ . Jumlah Skor yang diharapkan adalah jumlah skor observasi yang dikalikan dengan jumlah responden, sehingga  $35 \times 5 = 175$ . Perhitungan persentase kelayakan dari data Penyeleksi (tabel 5.8) menggunakan rumus berikut:

$$\sum \text{skor observasi} = (\text{jumlah} \times \text{skor SS}) + (\text{jumlah} \times \text{skor S}) + (\text{jumlah} \times \text{skor CS}) + (\text{jumlah} \times \text{skor TS}) + (\text{jumlah} \times \text{skor STS})$$

$$\sum \text{skor observasi} = (29 \times 5) + (6 \times 4) + (0 \times 3) + (0 \times 2) + (0 \times 1)$$

$$\sum \text{skor observasi} = 169$$

Sedangkan persentase kelayakan dari admin adalah sebagai berikut:

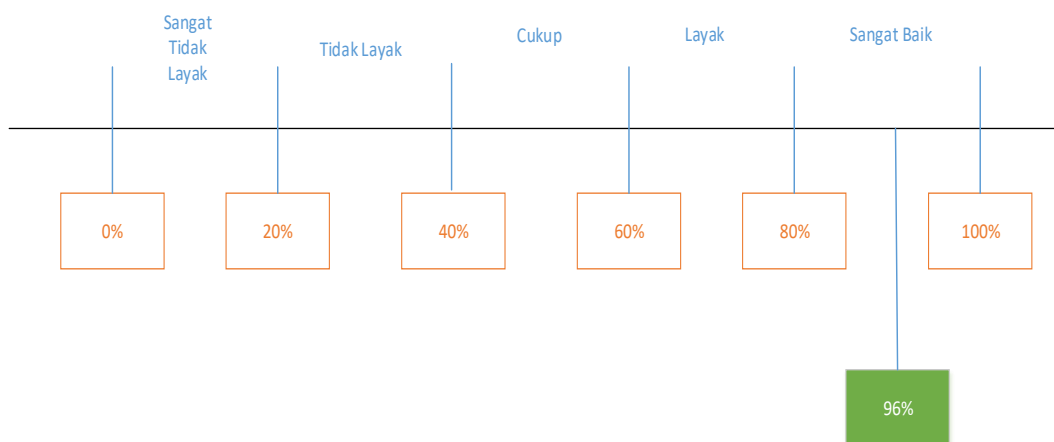
$$\text{Persentase kelayakan} = \frac{\sum \text{skor observasi}}{\text{skor yang diharapkan}} \times 100 \%$$

$$\text{Persentase kelayakan} = \frac{169}{175} = 0,96 \times 100 \% = 96\%$$

$$\text{Persentase kelayakan} = 96 \%$$

Total skor observasi dari data peserta dengan jumlah 169 (92%) dari skor yang diharapkan yaitu 175 (100%). Berdasarkan kriteria pada tabel kelayakan menurut Arikunto, Suharsimi & Safruddin A.J, Persentase total skor tersebut termasuk dalam kategori Sangat Layak. Penyajian skala sesuai persentase total skor

menurut (Arikunto, Suharsimi & Safruddin A.J, Cipi, 2009) secara detail dapat digambarkan seperti gambar 5.30 berikut:



Gambar 5.42 Hasil Presentase Peserta

Berikut ini adalah data saran perbaikan yang diberikan oleh peserta terangkum dalam Tabel 5.9

Tabel 5.9 Saran Peserta

No	Respon	Saran
1	Achmad Ma'mun	Sudah baik,jika bisa ditingkatkan akan lebih bagus
2	Elvina Fitria Irawan	eumzz sdh bagus dan ga ad saran.
3	Qori fadli	Sudah bagus
4	Nabil	Sudah bagus
5	Alvin	Tidak ada