

BAB III

PROGRAM SELEKSI MASUK SISWA BARU MENGUNAKAN METODE SMARTER

3.1 Metode Smarter (*Simple Multi-Attribute Rating Technique Exploiting Rank*)

Metode SMARTER (*Simple Multi-Attribute Rating Technique Exploiting Rank*) merupakan pengembangan dari metode sebelumnya, yaitu metode SMART (*Simple Multi-Attribute Rating Technique*). Metode SMART pertama kali diperkenalkan oleh Edward pada tahun 1971 dan baru dinamai sebagai metode SMART pada tahun 1977. Semenjak awal kemunculannya, metode SMART telah dikembangkan menjadi metode SMARTS (*Simple Multi-Attribute Rating Technique Swing*) lalu setelah dimodifikasi dan diperbaiki oleh Edward dan Baron pada tahun 1994 menjadi metode SMARTER (*Simple Multi-Attribute Rating Technique Exploiting Rank*).

Perbedaan antara metode SMARTER dengan metode SMART dan SMARTS terletak pada cara pembobotannya. Pembobotan kriteria pada ketiga metode tersebut tergantung pada urutan prioritas atribut dimana pada urutan pertama ditempati oleh atribut yang dianggap paling penting. Pada metode SMART dan SMARTER pembobotan diberikan langsung oleh pengambil keputusan. Tetapi prosedur pembobotan tersebut dianggap tidak proporsional dimana setiap bobot yang diberikan harus mencerminkan jarak dan prioritas setiap kriteria dengan tepat. Untuk mengatasi hal tersebut, pada metode SMARTER digunakan rumus pembobotan *Rank Order Centroid* (ROC).

3.2 Pembobotan Roc (*Rank Order Centroid*)

ROC didasarkan pada tingkat kepentingan atau prioritas dari kriteria. Menurut Jeffreys dan Cockfield (2008), teknik ROC memberikan bobot pada setiap kriteria sesuai dengan ranking yang dinilai berdasarkan tingkat prioritas. Biasanya dibentuk dengan pernyataan “Kriteria 1 lebih penting dari kriteria 2, yang lebih penting dari kriteria 3” dan

seterusnya hingga kriteria ke n , ditulis $Cr_1 \geq Cr_2 \geq Cr_3 \geq \dots \geq Cr_n$. Untuk menentukan bobotnya, diberikan aturan yang sama yaitu $W_1 \geq W_2 \geq W_3 \geq \dots \geq W_n$ dimana W_1 merupakan bobot untuk kriteria C_1 . Atau dapat dijelaskan sebagai berikut

Jika

$$Cr_1 \geq Cr_2 \geq Cr_3 \geq \dots \geq Cr_n \quad (3.1)$$

Maka

$$W_1 \geq W_2 \geq W_3 \geq \dots \geq W_n \quad (3.2)$$

Selanjutnya, jika k merupakan banyaknya kriteria, maka

$$W_1 = \frac{(1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k})}{k} \quad (3.3)$$

$$W_2 = \frac{(0 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k})}{k} \quad (3.4)$$

$$W_3 = \frac{(0 + 0 + \frac{1}{3} + \dots + \frac{1}{k})}{k} \quad (3.5)$$

$$W_k = \frac{(0 + \dots + 0 + \frac{1}{k})}{k} \quad (3.6)$$

Secara umum pembobotan ROC dapat dirumuskan sebagai berikut,

$$W_k = \frac{1}{k} \sum_{i=1}^k \left(\frac{1}{i}\right) \quad (3.7)$$

Untuk sebanyak 2 sampai 9 kriteria, nilai dari persamaan diatas dapat dilihat pada tabel sebagai berikut:

Tabel 3.1 Pembobotan ROC

| Ranking | Kriteria | | | | | | | |
|---------|----------|-------|-------|-------|-------|-------|-------|-------|
| | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | 0.75 | 0.611 | 0.521 | 0.457 | 0.408 | 0.370 | 0.340 | 0.314 |
| 2 | 0.25 | 0.278 | 0.271 | 0.257 | 0.242 | 0.228 | 0.215 | 0.203 |
| 3 | | 0.111 | 0.146 | 0.157 | 0.158 | 0.156 | 0.152 | 0.148 |
| 4 | | | 0.063 | 0.090 | 0.103 | 0.109 | 0.111 | 0.111 |
| 5 | | | | 0.040 | 0.061 | 0.073 | 0.079 | 0.083 |
| 6 | | | | | 0.028 | 0.044 | 0.054 | 0.061 |
| 7 | | | | | | 0.020 | 0.034 | 0.042 |
| 8 | | | | | | | 0.016 | 0.026 |
| 9 | | | | | | | | 0.012 |

Jika dua atau lebih kriteria dianggap sama penting, maka bobot yang diberikan bagi masing-masing kriteria merupakan rata-rata dari gabungan peringkatnya. Sebagai contoh, jika $Cr_1 \geq Cr_2 = Cr_3$, berarti kriteria 1 lebih penting daripada kriteria 2 yang sama pentingnya dengan kriteria 3. Maka bobot yang diberikan adalah sebagai berikut:

$$Cr_1 = W_1 = 0,611$$

$$Cr_2 = \left(\frac{W_2 + W_3}{2} \right) = \left(\frac{0.278 + 0.111}{2} \right) = 0.1945$$

$$Cr_3 = \left(\frac{W_2 + W_3}{2} \right) = \left(\frac{0.278 + 0.111}{2} \right) = 0.1945$$

Istara,dkk. (2013) menuuliskan rumus metode SMARTER secara umum sebagai berikut:

$$\text{maksimumkan } \sum_{j=1}^k W_j U_{ij} \quad , \forall i = 1 \text{ sampai } n \quad (3.8)$$

Dimana W_j adalah nilai skala pembobotan kriteria ke-j dari k kriteria, U_{ij} adalah nilai utilitas i pada kriteria j, dan n adalah banyak data.

3.3 Langkah-langkah metode SMARTER

Langkah-langkah metode SMARTER adalah sebagai berikut:

1. Identifikasi permasalahan, agar keputusan yang akan diambil lebih terarah dan tidak menyimpang dari tujuan yang akan dicapai.
2. Tentukan alternatif, kriteria dan sub kriteria yang digunakan dalam membuat keputusan.
3. Memberikan peringkat untuk setiap kriteria dan sub kriteria.
4. Menghitung bobot menggunakan pembobotan ROC untuk setiap kriteria, hal ini bergantung pada peringkat yang telah diberikan pada langkah 3.
5. Menghitung bobot menggunakan pembobotan ROC untuk setiap sub kriteria, hal ini bergantung pada peringkat yang telah diberikan pada langkah 3.
6. Menghitung bobot akhir setiap kriteria, dengan mengalikan hasil langkah 4 dengan hasil langkah 5
7. Memberikan penilaian pada semua kriteria untuk setiap alternatif. Nilai diberikan dalam skala 0-100 dimana 0 sebagai nilai minimum dan 100 sebagai nilai maksimum.
8. Menghitung utilitas terhadap setiap alternatif dengan menggunakan persamaan $\text{maksimumkan } \sum_{j=1}^k W_j U_{ij} \quad , \forall i = 1 \text{ sampai } n$
9. Memutuskan. jika hanya satu alternatif yang akan dipilih, maka akan dipilih alternatif dengan nilai utilitas paling besar.

3.4 Struktur Basis Data

Adapun struktur basis data pada program sistem pendukung keputusan seleksi masuk siswa menggunakan metode SMARTER diuraikan dengan tabel sebagai berikut:

Tabel 3.2 Identitas Peserta Ujian

| No. | Nama | Tipe | Null | Keterangan |
|-----|------------------|---------------|------|---------------------------------|
| 1 | k_identitas | Double | No | Kode peserta |
| 2 | nomor_peserta | Varchar (3) | No | Nomor urut |
| 3 | nama | Varchar (60) | Yes | Nama |
| 4 | tmpt_lahir | Varchar (50) | Yes | Tempat lahir |
| 5 | tgl_lahir | Date | Yes | Tanggal lahir |
| 6 | jenis_kelamin | Varchar (15) | Yes | Jenis kelamin |
| 7 | asal_sekolah | Varchar (150) | Yes | Asal sekolah |
| 8 | tahun_ujian | Varchar (4) | Yes | Tahun masuk ujian |
| 9 | Kategori_sekolah | Tinyint (4) | Yes | Kategori sekolah berbasis islam |
| 10 | hasil_test_lisan | Double | Yes | Skor ujian lisan |
| 11 | hasil_test_tulis | Double | Yes | Skor ujian tulis |

| | | | | |
|----|-------------|--------------|-----|----------------------------|
| 12 | masuk_kelas | Varchar (1) | Yes | Kelas A/B/C/D/Tajhiziyyah |
| 13 | i_entry | Varchar (50) | Yes | Admin yang meng-entry data |

Tabel 3.3 Jenis Ujian

| No. | Nama | Tipe | Null | Keterangan |
|-----|-----------------|--------------|------|-------------|
| 1 | kode_jenis_test | Varchar (3) | No | Kode ujian |
| 2 | nama_jenis_test | Varchar (50) | yes | jenis ujian |

Tabel 3.4 Kategori Soal Lisan

| No. | Nama | Tipe | Null | Keterangan |
|-----|---------------|---------------|------|---------------|
| 1 | kode_kategori | Varchar (4) | No | Kode kategori |
| 2 | prioritas | Tinyint (1) | yes | Prioritas |
| 3 | nama_kategori | Varchar (100) | Yes | Bidang studi |
| 4 | i_entry | Varcahr (50) | Yes | admin |

Tabel 3.5 Kategori Soal Tulis

| No. | Nama | Tipe | Null | Keterangan |
|-----|---------------|-------------|------|---------------|
| 1 | kode_kategori | Varchar (4) | No | Kode kategori |
| 2 | prioritas | Tinyint (1) | yes | Prioritas |

| | | | | |
|---|---------------|---------------|-----|--------------|
| 3 | nama_kategori | Varchar (100) | Yes | Bidang studi |
| 4 | i_entry | Varcahr (50) | Yes | admin |

Tabel 3.6 Maksimum Siswa Perkelas

| No. | Nama | Tipe | Null | Keterangan |
|-----|--------------|---------------|------|-------------------------|
| 1 | tahun | Mediumint (9) | No | Tahun ajaran |
| 2 | maks_siswa | Tinyint(4) | Yes | Maksimum siswa perkelas |
| 3 | jumlah_kelas | Tinyint(4) | Yes | Banyak kelas |
| 4 | i_entry | Varchar (50) | Yes | admin |

Tabel 3.7 Penguji

| No. | Nama | Tipe | Null | Keterangan |
|-----|---------------|---------------|------|---------------|
| 1 | kode_penguji | Varchar (30) | No | Kode |
| 2 | nama_penguji | Varchar (100) | Yes | Nama |
| 3 | alamat | Varchar (200) | Yes | Alamat |
| 4 | jenis_kelamin | Varchar (15) | Yes | Jenis kelamin |
| 5 | i_entry | Varchar (50) | Yes | Admin |

Tabel 3.8 Gelombang Ujian

| No. | Nama | Tipe | Null | Keterangan |
|-----|-----------------|--------------|------|-------------------------|
| 1 | kode_gelombang | Varchar (3) | No | Kode |
| 2 | kode_jenis_test | Varchar (3) | No | Jenis ujian |
| 3 | nama_gelombang | Varchar (15) | Yes | Nama gelombang I/II/III |
| 4 | jam_mulai | Time | Yes | Waktu mulai ujian |
| 5 | jam_selesai | Time | Yes | Waktu selesai ujian |
| 6 | i_entry | Varchar (50) | Yes | Admin |

Tabel 3.9 Jadwal Ujian

| No. | Nama | Tipe | Null | Keterangan |
|-----|-----------------|-------------|------|----------------------|
| 1 | kode_jadwal | Double | No | Kode jadwal |
| 2 | kode_jenis_test | Varchar (3) | No | Kode Jenis ujian |
| 3 | nomor_peserta | Varchar (3) | No | Nomor peserta |
| 4 | kode_gelombang | Varchar (3) | No | Kode gelombang ujian |
| 5 | kode_penguji | Varchar (3) | No | Kode penguji |
| 6 | tanggal_jadwal | Date | Yes | Tanggal ujian |

| | | | | |
|---|---------------|--------------|-----|---------------|
| 7 | Tahun | Varchar (4) | Yes | Tahun ujian |
| 8 | Status_jadwal | Tinyint (1) | yes | Status jadwal |
| 9 | i_entry | Varchar (50) | yes | Admin |

Tabel 3.10 Soal

| No. | Nama | Tipe | Null | Keterangan |
|-----|-------------------|---------------|------|--------------------------|
| 1 | kode_soal | Varchar (8) | No | Kode soal |
| 2 | kode_kategori | Varchar (4) | No | Kode kategori |
| 3 | pertanyaan | Text | Yes | Pertanyaan |
| 4 | pertanyaan_gambar | Mediumblob | Yes | Pertanyaan berupa gambar |
| 5 | pilihan_A | Varchar (200) | Yes | Jawaban a |
| 6 | pilihan_B | Varchar (200) | Yes | Jawaban b |
| 7 | pilihan_C | Varchar (200) | Yes | Jawaban c |
| 8 | jawaban | Varchar (1) | Yes | Jawaban |
| 9 | Prioritas | Tinyint(1) | Yes | Prioritas soal |
| 10 | i_entry | Varchar (50) | Yes | Admin |

Tabel 3.11 Ujian Lisan

| No. | Nama | Tipe | Null | Keterangan |
|-----|-----------------|--------------|------|-------------------|
| 1 | kode_test_lisan | Double | No | Kode ujian |
| 2 | kode_kategori | Varchar (4) | No | Kode bidang studi |
| 3 | kode_penguji | Varchar (3) | No | Kode penguji |
| 4 | nomor_peserta | Varchar (3) | No | Nomor peserta |
| 5 | nilai | Double | Yes | Nilai awal |
| 6 | tahun | Varchar (4) | Yes | Tahun ujian |
| 7 | nilai_akhir | Double | Yes | Nilai akhir |
| 8 | i_entry | Varchar (50) | yes | admin |

Tabel 3.12 Kode Jawaban

| No. | Nama | Tipe | Null | Keterangan |
|-----|---------------|-------------|------|---------------|
| 1 | kode_jawaban | Double | No | Kode jawaban |
| 2 | nomor_peserta | Varchar (3) | No | Nomor peserta |
| 3 | kode_soal | Varchar (8) | No | Kode soal |
| 4 | jawaban | Varchar (1) | Yes | Jawaban |

| | | | | |
|----|-----------------|--------------|-----|------------------|
| 5 | no_soal | Int (11) | Yes | Nomor soal |
| 6 | tahun | Varchar (4) | Yes | Tahun |
| 7 | stts_jawaban | Tinyint (4) | Yes | Status jawaban |
| 8 | nilai_prioritas | Double | yes | bobot pertanyaan |
| 9 | nilai_akhir | Double | Yes | nilai akhir |
| 10 | i_entry | Varchar (50) | yes | Admin |

Tabel 3.13 Panitia

| No. | Nama | Tipe | Null | Keterangan |
|-----|-----------|---------------|------|------------|
| 1 | username | Varchar (50) | no | username |
| 2 | password | Varchar (100) | Yes | Password |
| 3 | tahun | Varchar (4) | No | Tahun |
| 4 | jns_login | Tinyint (4) | Yes | otoritas |

3.5 Perancangan Sistem

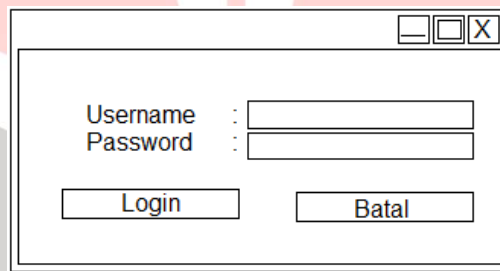
Perancangan sistem berikut ditujukan untuk memberikan gambaran yang jelas kepada pengguna tentang program yang akan dibuat. Hasil dari perancangan akan digunakan dalam tahap implementasi, tahap ini merupakan tahap penerapan sistem

agar dapat diketahui kelemahan-kelemahan sistem yang nantinya akan dilakukan perbaikan sampai sistem sesuai dengan apa yang diharapkan

3.5.1 Rancangan Antarmuka

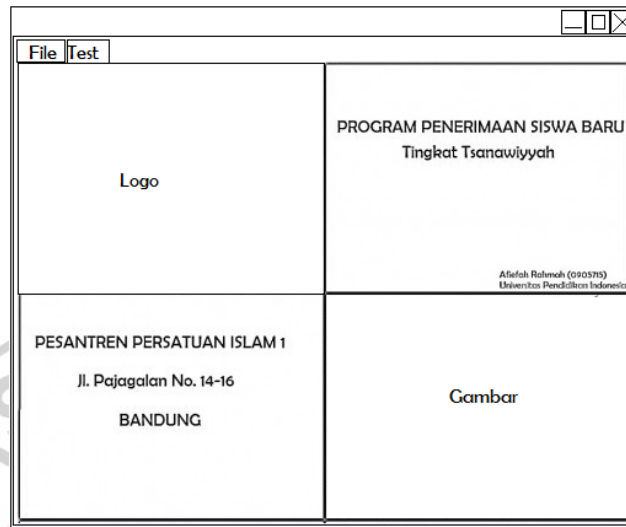
Penyajian program menggunakan tampilan antarmuka (*user interface*) akan membuat program menjadi lebih menarik bagi pengguna juga membuat pengoperasian program lebih mudah digunakan. Penyajian program tampilan antarmuka yang akan dirancang dibuat sesuai dengan kebutuhan *input-output* sistem sehingga dapat membantu pengguna untuk mengetahui langkah-langkah yang harus dilakukan saat menggunakan aplikasi tersebut.

Gambar-gambar berikut merupakan gambar rancangan antarmuka program seleksi penerimaan siswa baru menggunakan metode SMARTER.



Gambar 3.1 Desain Form Login

Pada Gambar 3. Terdapat *textbox username* dan *password* yang wajib diisi oleh pengguna. Selain login ditujukan untuk menampilkan menu utama, desain menu *login* juga ditujukan untuk menjaga keamanan data.



Gambar 3.2 Desain Halaman Utama

Dari **Gambar 3.2** Dapat dilihat bahwa dalam perancangan menu utama terdapat tampilan menu dan sub menu yang terdiri dari:

1. File

- Penguji
- Santri
 - Input Data Peserta
 - Laporan Data Peserta
- Soal
 - Input Kategori Soal Lisan
 - Input Pertanyaan Tulis
- Input Gelombang
- Input Maks Siswa /Kelas
- Jadwal Ujian
 - Input Data
 - Lihat Data

2. Test

- Lisan

- Tulisan
- Hasil Ujian

3. Login

- Logout
- Add User

| No. | Kode | Nama | Alamat |
|-----|------|------|--------|
| | | | |

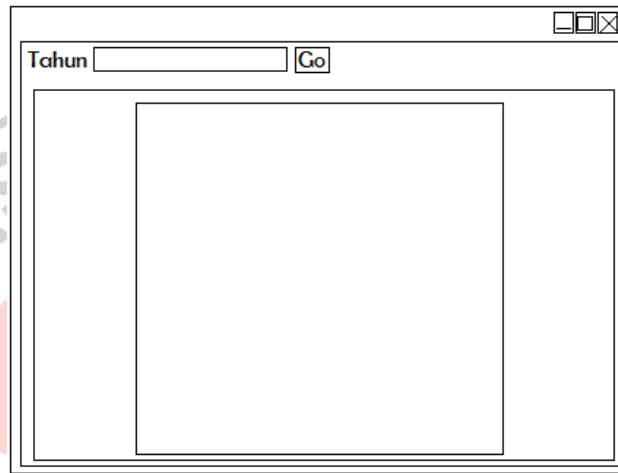
Gambar 3.3 Desain Data Penguji

Desain pada **Gambar 3.3** berfungsi untuk menambah data penguji untuk ujian lisan. Setelah diinput di form ini, data penguji pada form yang lain akan berupa *combo box* sehingga tidak perlu diinput ulang.

| Nomor Peserta | Nama Peserta | Tempat dan Tanggal Lahir | Jenis Kelamin | Asal Sekolah |
|---------------|--------------|--------------------------|---------------|--------------|
| | | | | |

Gambar 3.4 Desain Entri Data Peserta

Gambar 3.4 menunjukkan desain form entri data peserta. Format data yang harus diisi terdiri dari Tahun Ujian, Nama Peserta, Tempat dan Tanggal Lahir, Jenis Kelamin, dan Asal Sekolah. Sedangkan **Gambar 3.5** menunjukkan form untuk laporan data peserta yang dapat *diprint-out* berdasarkan tahun ujian.

The image shows a browser window with a title bar containing the text 'Tahun' and a 'Go' button. Below the title bar is a large, empty rectangular area, likely intended for displaying a list of participant data that can be printed.

Gambar 3.5 Laporan Data Peserta

Gambar 3.6 dan **Gambar 3.7** adalah desain form untuk entri kategori soal berdasarkan kriteria ujian tulis dan ujian lisan. Form ini berfungsi untuk menentukan banyaknya kriteria yang akan digunakan untuk ujian. Dimana banyak kategori dibuat fleksibel agar mudah digunakan saat ada penambahan atau pengurangan kriteria.

| Kode Kategori | Prioritas | Bidang Studi |
|---------------|-----------|--------------|
| | | |

Gambar 3.6 Desain Entri Kategori Soal Tulis

| Kode Kategori | Prioritas | Bidang Studi |
|---------------|-----------|--------------|
| | | |

Gambar 3.7 Desain Entri Kategori Soal Lisan

| Kode Kategori | Prioritas | Pertanyaan | Pilihan A | Pilihan B | Pilihan C | Jawaban |
|---------------|-----------|------------|-----------|-----------|-----------|---------|
| | | | | | | |

Gambar 3.8 Desain Entri Pertanyaan Tulis

Pertanyaan yang akan diujikan dapat langsung diinput kedalam program dengan desain form seperti pada **Gambar 3.8**. Sedangkan untuk memunculkan pertanyaan dapat dilihat pada form pertanyaan untuk peserta. Pertanyaan yang muncul akan diacak sehingga tiap peserta akan mendapatkan pertanyaan yang berbeda dengan bobot yang sama.

Gambar 3.9 Desain Entri Gelombang Ujian

Ujian akan dilakukan menggunakan komputer bagi setiap masing-masing peserta sedangkan jumlah komputer di sekolah kurang dari banyak siswa. Untuk mengatasi hal tersebut, maka setiap siswa akan dikelompokkan kedalam beberapa kelompok yang akan mendapat jadwal waktu ujian yang berbeda-beda.

A screenshot of a web form with a close button (X) in the top right corner. The form contains two input fields: the first is labeled 'Tahun Ujian' and the second is labeled 'Maksimum'. To the right of these fields is a button labeled 'Submit'.

Gambar 3.10 Desain Entri Maksimum Siswa Perkelas

Gambar 3.10 ditujukan untuk menentukan banyaknya siswa perkelas.

A screenshot of a web form for scheduling an exam. At the top, there is a dropdown menu for 'Jenis Tes'. Below it are input fields for 'No Peserta', 'Tgl Tes', 'Gelombang', and 'Penguji', followed by a 'Set' button. Below these fields is a table with the following columns: 'No. Nomor Peserta', 'Nama Peserta', 'Tanggal Ujian', 'Gelombang', and 'Penguji'. The table body is currently empty.

Gambar 3.11 Desain Jadwal Ujian

A screenshot of a web form for exam reports. It features a dropdown menu for 'Jenis Tes' and a 'Go' button. Below these is a large, empty rectangular area, likely intended for displaying a list of reports or data.

Gambar 3.12 Desain Laporan Ujian

Pada **Gambar 3.11** desain memiliki fungsi yang sama seperti form laporan data siswa.

| No. | Kode | Bidang Studi | Nilai |
|-----|------|--------------|-------|
| | | | |

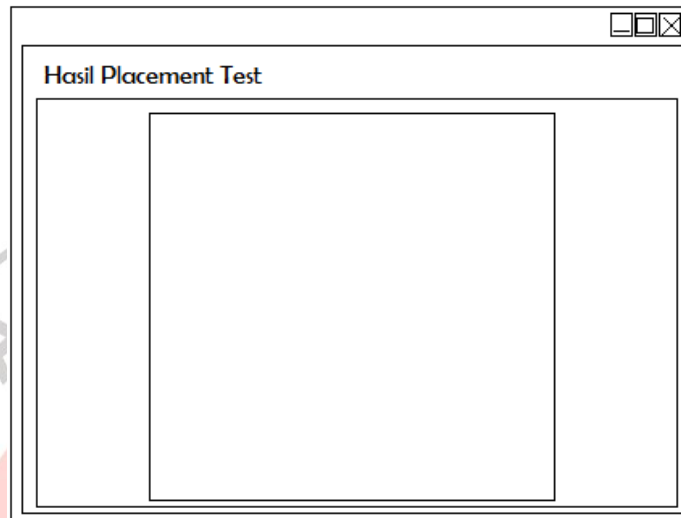
Gambar 3.13 Desain Ujian Lisan

Pada **Gambar 3.12**, data yang harus diinputkan adalah tahun, penguji dan nomor peserta. Sedangkan data yang lain akan muncul setelah nomor peserta diinputkan.

Gambar 3.14 Desain Form Ujian Peserta

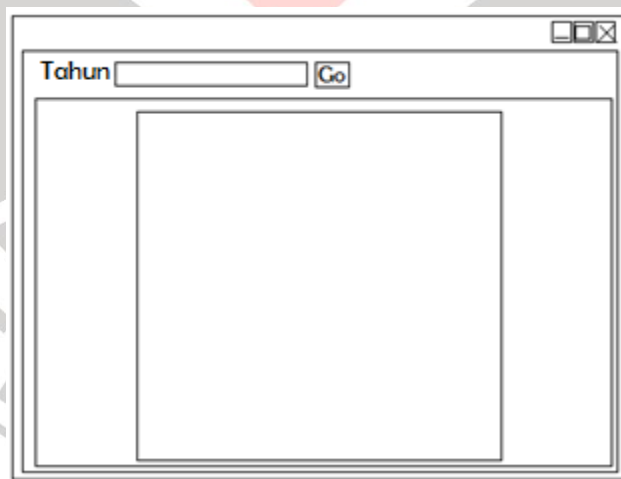
Desain **Gambar 3.14** Ditujukan untuk peserta yang akan muncul sesuai dengan waktu yang telah ditentukan. Pada form ini peserta dapat mengisi soal. Pada form ini terdapat *textbox* waktu ujian disebelah kanan *command button* selesai. Ini berfungsi untuk membatasi waktu yang dimiliki peserta untuk mengerjakan soal. Jika

waktu telah habis, maka form akan keluar otomatis. Apabila peserta selesai mengerjakan sebelum waktu habis, maka peserta harus menekan tombol Selesai.



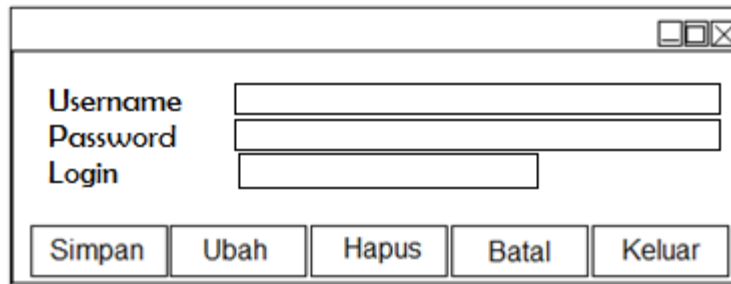
Gambar 3.15 Desain Gambar Hasil Ujian

Gambar diatas menunjukkan hasil ujian tes setelah dihitung menggunakan metode SMARTER.



Gambar 3.16 Desain Form Laporan Hasil Ujian

Gambar 3.16 Akan menunjukkan hasil data yang telah diolah menjadi urutan ranking yang telah dibagi kedalam beberapa kelas. Dalam form ini, terdapat menu print sehingga data hasil ujian bisa langsung di *Print-Out*.



| | |
|----------|--------------------------|
| Username | <input type="text"/> |
| Password | <input type="password"/> |
| Login | <input type="text"/> |
| Simpan | |
| Ubah | |
| Hapus | |
| Batal | |
| Keluar | |

Gambar 3.17 Desain Form Add User

Desain ini berfungsi untuk menambahkan *user* yaitu panitia yang memiliki izin untuk melakukan akses pada program ini.