

# **SKRIPSI**

## **APLIKASI TES POTENSI AKADEMIK DISERTAI TES BUTA WARNA MENGGUNAKAN DELPHI 7.0 & MYSQL**



**Disusun Oleh :**  
**RIZAL SUSANTO**  
**NIM 05.12.646**



**JURUSAN TEKNIK ELEKTRO S-1**  
**KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**2009**

LAWRENCE MICHIGAN POLICE AND MARSHAL  
MANAGERSHIP AUSTIN AND CO.  
JULY 24 1913

**THE  
GOLDEN  
CROWN JEWEL**

## LEMBAR PERSETUJUAN

### APLIKASI TES POTENSI AKADEMIK DISERTAI TES BUTA WARNA MENGGUNAKAN DELPHI 7.0 & MYSQL

#### SKRIPSI

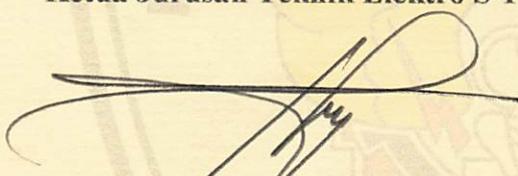
*Disusun dan Diajukan sebagai Salah Satu Syarat Untuk Memperoleh  
Gelar Sarjana Teknik Komputer Dan Informatika Strata Satu (S-1)*

Disusun Oleh :

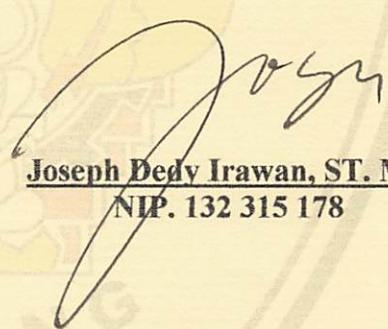
RIZAL SUSANTO

NIM : 05.12.646

Mengetahui  
Ketua Jurusan Teknik Elektro S-1

  
Ir. F. Yudi Limpraptono, MT  
NIP Y. 1039500274

Diperiksa dan Disetujui  
Dosen Pembimbing

  
Joseph Dedy Irawan, ST. MT  
NIP. 132 315 178

JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2009

## **ABSTRAK**

*Tes potensi akademik merupakan salah satu dari sekian jenis tes yang digunakan untuk mengukur dan memberi penilaian beberapa hal terkait dengan kemampuan dan potensi akademik seseorang. Sedangkan tes buta warna merupakan tes untuk mengetahui seseorang menderita buta warna atau tidak. aplikasi ini menggunakan database server MySQL dan perancangan scripting progamming menggunakan bahasa Delphi 7.0 yang berjalan di sistem.*

*Implementasi yang dilakukan pada aplikasi ini antara lain menyediakan fasilitas masukkan data peserta masukkan soal tes, lihat data peserta, hapus data peserta, edit data peserta yang berada di fasilitas admin. Sedangkan untuk peserta tes disediakan soal tes potensi akademik disertai tes buta warna. Soal tes disajikan secara random. Jawaban peserta yang telah disimpan di database dapat langsung dikoreksi secara otomatis oleh komputer dengan admin sebagai operator.*

**Kata Kunci :** *Tes Potensi Akademik, Tes Buta Warna, Delphi,, MySQL.*

## **KATA PENGANTAR**

Dengan mengucap puji syukur kehadirat Allah SWT yang dengan segala Kasih dan Anugerah – Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul :

### **APLIKASI TES POTENSI AKADEMIK DISERTAI TES BUTA WARNA MENGGUNAKAN DELPHI 7.0 & MYSQL**

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata-1 di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE., selaku Rektor ITN Malang.
2. Bapak Ir. Sidik Noertjahyono, MT., selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. F. Yudi Limpraptono, MT., selaku Ketua Jurusan Teknik Elektro S – 1 ITN Malang
4. Bapak Joseph Dedy Irawan, ST. MT., selaku Dosen Pembimbing.
5. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis juga berharap agar laporan tugas akhir ini dapat dikembangkan lebih lanjut sehingga dapat benar-benar digunakan sebaik-baiknya untuk mendukung perkembangan ilmu pengetahuan.. Oleh sebab itu kritik dan saran yang bersifat membangun sangat penulis harapkan.

Akhir kata, penulis mohon maaf kepada semua pihak apabila selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat baik bagi pembaca maupun penulis sendiri.  
Amin.

Malang, September 2009

Penulis

## **DAFTAR ISI**

<b>LEMBAR PERSETUJUAN .....</b>	<b>i</b>
<b>ABSTRAK.....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR .....</b>	<b>ix</b>
<b>DAFTAR TABEL.....</b>	<b>xii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan.....	3
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian .....	3
1.6. Sistematika Penulisan.....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>6</b>
2.1. Inteligensi dan Bakat.....	6
2.2. Tes Potensi Akademik.....	7
2.3. Sistem Penilaian .....	10
2.3.1 Tes Potensi Akademi.....	10
2.3.2 Tes Buta Warna.....	11

2.4. Delphi 7.0 .....	11
2.4.1. Apache.....	20
2.4.2. MySQL.....	20
2.4.3. PHP.....	23
2.5. MySQL.....	17
 <b>BAB III PERANCANGAN DAN DESAIN SISTEM.....</b>	<b>20</b>
3.1 Analisa Sistem.....	20
3.2 Deskripsi Sistem.....	20
3.3 Perancangan Sistem .....	21
3.3.1 <i>Flow Chart</i> Sistem .....	21
3.3.1.1 <i>Flow Chart Admin</i> .....	21
3.3.1.2 <i>Flow Chart Peserta</i> .....	25
3.3.2 Alur Proses .....	26
3.3.3 Perancangan Database.....	27
3.3.3.1 Tabel Pada Aplikasi <i>Admin</i> .....	27
3.3.3.2 Tabel Pada Aplikasi Peserta.....	28
3.3.4 Perancangan Program.....	27
3.3.4.1 <i>Admin</i> .....	42
3.3.4.2 Peserta .....	43
3.3.5 Desain Antarmuka Aplikasi .....	44
3.3.5.1 Aplikasi <i>Admin</i> .....	44
3.3.5.2 Aplikasi Peserta.....	48

<b>BAB IV IMPLEMENTASI DAN PENGUJIAN .....</b>	<b>50</b>
<b>4.1 Pengujian Sistem .....</b>	<b>50</b>
<b>4.1.1 Metode Pengoperasian .....</b>	<b>50</b>
<b>4.2 Pengujian Aplikasi Admin .....</b>	<b>50</b>
<b>4.2.1 Pengujian Login Admin .....</b>	<b>50</b>
<b>4.2.2 Pengujian Masukkan Data Peserta .....</b>	<b>53</b>
<b>4.2.3 Pengujian Lihat Data Peserta .....</b>	<b>54</b>
<b>4.2.4 Pengujian Hapus Data Peserta.....</b>	<b>55</b>
<b>4.2.5 Pengujian Edit Data Peserta .....</b>	<b>56</b>
<b>4.2.6 Pengujian Masukkan Soal .....</b>	<b>57</b>
<b>4.2.7 Pengujian Lihat Soal .....</b>	<b>58</b>
<b>4.2.8 Pengujian Hapus Soal .....</b>	<b>59</b>
<b>4.2.9 Pengujian Edit Soal .....</b>	<b>60</b>
<b>4.2.10 Pengujian Lihat Hasil Tes .....</b>	<b>61</b>
<b>4.3 Pengujian Aplikasi Peserta.....</b>	<b>61</b>
<b>4.3.1. Pengujian Login Peserta.....</b>	<b>61</b>
<b>4.3.2. Pengujian Soal Tes Potensi Akademik.....</b>	<b>63</b>
<b>4.3.3. Pengujian Soal Tes Buta Warna.....</b>	<b>69</b>
<b>4.4 Analisa Hasil Pengujian Sistem .....</b>	<b>69</b>
<b>4.5 Implementasi Sistem .....</b>	<b>69</b>
<b>4.5.1 Spesifikasi Sarana Perangkat Keras .....</b>	<b>70</b>
<b>4.5.2 Spesifikasi Sarana Piranti Lunak.....</b>	<b>70</b>

<b>BAB V PENUTUP.....</b>	<b>74</b>
<b>5.1 Kesimpulan.....</b>	<b>74</b>
<b>5.2 Saran.....</b>	<b>74</b>

**DAFTAR PUSTAKA**

**LAMPIRAN – LAMPIRAN**

## **DAFTAR GAMBAR**

### **BAB II LANDASAN TEORI**

Gambar 2.1 Menu Bar dan Tool Bar .....	13
Gambar 2.2 Component Pallete .....	13
Gambar 2.3 Form Designer .....	14
Gambar 2.4 Code Editor.....	15
Gambar 2.5 Object Inspector.....	16
Gambar 2.6 Object Tree View .....	16

### **BAB III PERANCANGAN DAN DESAIN SISTEM**

Gambar 3.1 <i>Flowchart Admin</i> .....	23
Gambar 3.2 <i>Flowchart Peserta</i> .....	25
Gambar 3.3 Desain Alur Proses .....	26
Gambar 3.4 <i>Login Admin</i> .....	44
Gambar 3.5 Menu Utama.....	45
Gambar 3.6 Masukkan Data Peserta .....	45
Gambar 3.7 Lihat Data Peserta .....	46
Gambar 3.8 Hapus Data Peserta.....	46
Gambar 3.9 Edit Data Peserta .....	47
Gambar 3.10 Masukkan Soal, Lihat Soal, Edit Soal, Hapus Soal.....	47
Gambar 3.11 Lihat Hasil Tes .....	48
Gambar 3.12 <i>Login Peserta</i> .....	48
Gambar 3.13 Soal .....	49

## **BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM**

Gambar 4.1 Halaman <i>Login Admin</i> .....	51
Gambar 4.2 Error.....	52
Gambar 4.3 Menu Utama .....	52
Gambar 4.4 Masukkan Data Peserta .....	53
Gambar 4.5 Lihat Data Peserta .....	54
Gambar 4.6 Hapus Data Peserta.....	55
Gambar 4.7 Edit Data Peserta .....	56
Gambar 4.8 Masukkan soal no 1 – 5 .....	57
Gambar 4.9 Lihat soal no 1 – 5 .....	58
Gambar 4.10 Hapus soal no 1 - 5 .....	59
Gambar 4.11 <i>Edit</i> soal no 1 – 5 .....	60
Gambar 4.12 Lihat Hasil Tes .....	61
Gambar 4.13 <i>Login</i> Peserta .....	62
Gambar 4.14 <i>Error</i> .....	62
Gambar 4.15 Petunjuk Pengerjaan Soal .....	63
Gambar 4.16 Tampilan Soal Tes Potensi Akademik no 1-5 .....	64
Gambar 4.17 Tampilan Soal Tes Potensi Akademik no 6-7 .....	65
Gambar 4.18 Tampilan Soal Tes Potensi Akademik no 8-10 .....	65
Gambar 4.19 Tampilan Soal Tes Potensi Akademik no 11-25 .....	66
Gambar 4.20 Tampilan Soal Tes Potensi Akademik no 26 -35 .....	66
Gambar 4.21 Tampilan Soal Tes Potensi Akademik no 36 -45 .....	67
Gambar 4.22 Tampilan Soal Tes Potensi Akademik no 46 - 48 .....	67

Gambar 4.23 Tampilan Soal Tes Potensi Akademik no 49 – 52 .....	68
Gambar 4.24 Tampilan Soal Tes Potensi Akademik no 53 – 55 .....	68
Gambar 4.25 Tampilan Soal Tes Buta Warna.....	69

## **DAFTAR TABEL**

### **BAB III PERANCANGAN DAN DESAIN SISTEM**

Tabel 3.1. Struktur Tabel <i>Admin</i> .....	27
Tabel 3.2. Struktur Tabel Peserta .....	28
Tabel 3.3. Struktur Tabel Soal_1_5 .....	29
Tabel 3.4. Struktur Tabel Soal_6_7 .....	29
Tabel 3.5. Struktur Tabel Soal_8_10 .....	30
Tabel 3.6. Struktur Tabel Soal_11_25 .....	30
Tabel 3.7. Struktur Tabel Soal_26_35 .....	31
Tabel 3.8. Struktur Tabel Soal_36_45 .....	32
Tabel 3.9. Struktur Tabel Soal_46_48 .....	33
Tabel 3.10 Struktur Tabel Soal_49_52 .....	34
Tabel 3.11 Struktur Tabel Soal_53_55 .....	35
Tabel 3.12 Struktur Tabel Buta .....	36
Tabel 3.13 Struktur Tabel J_1_5 .....	36
Tabel 3.14 Struktur Tabel J_6_7 .....	37
Tabel 3.15 Struktur Tabel J_8_10 .....	38
Tabel 3.16 Struktur Tabel J_11_25 .....	38
Tabel 3.17 Struktur Tabel J_26_35 .....	39
Tabel 3.18 Struktur Tabel J_36_45 .....	39
Tabel 3.19 Struktur Tabel J_46_48 .....	40
Tabel 3.20 Struktur Tabel J_49_52 .....	40

Tabel 3.12 Struktur Tabel J_53_55 .....	41
Tabel 3.12 Struktur Tabel J_butu.....	41

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Seiring dengan perkembangan ilmu pengetahuan dan teknologi saat ini, maka kebutuhan akan informasi yang tepat guna, tepat waktu dan akurat sangat dibutuhkan oleh banyak instansi. Komputer sebagai alat bantu dalam pengolahan data merupakan salah satu solusi dari pemecahan masalah tersebut. Kehadiran komputer saat ini hampir secara keseluruhan digunakan pada berbagai bidang, salah satunya adalah pengolahan informasi. Pada saat ini komputer terus dikembangkan terutama dalam hal informasi guna mendukung kinerja dari suatu instansi.

Dukungan kerja terhadap suatu instansi berkaitan dengan hasil kerja, untuk mendapatkan hasil yang baik maka diperlukan tes potensi akademik. Tes potensi akademik merupakan salah satu dari sekian jenis tes yang digunakan untuk mengukur dan memberi penilaian beberapa hal terkait dengan kemampuan dan potensi akademik seseorang, sejauh ini penggunaannya sudah semakin luas. Misalnya sebagai salah satu pertimbangan dalam menentukan kapasitas dan kapabilitas seseorang yang biasanya disyaratkan dalam kategori tertentu seperti penerimaan siswa atau mahasiswa baru, penerimaan karyawan, seleksi pegawai negeri hingga pengukur kecenderungan pola pikir dan kemampuan intelektual anggota dewan. Bahkan akhir-akhir ini, tes potensi akademik menjadi bagian dari

syarat kenaikan jabatan atau posisi-posisi tertentu pada perusahaan dan mulai merambah ke area tes kelayakan (proper test) untuk tugas-tugas tertentu.

Disamping tes potensi akademik, pada instansi yang dalam bidangnya membutuhkan persepsi yang berkaitan dengan warna, misalnya : bidang kesehatan, bidang tekstil, bidang kimia, dll. Maka dilakukan juga tes buta warna. Buta warna adalah suatu kelainan yang disebabkan ketidakmampuan sel-sel kerucut mata untuk menangkap suatu spektrum warna tertentu akibat faktor genetis.

Pada pelaksanaannya tes potensi akademik dan tes buta warna masih dilakukan secara manual, dimana tingkat kesalahan manual dalam segi koreksi kemungkinan bisa terjadi duplikasi hasil, selain itu membutuhkan waktu yang relatif lama untuk melakukan koreksi tes potensi akademik dengan tes buta warna. Dengan demikian diperlukan suatu usaha untuk mengetahui hasil yang lebih akurat dan efisiensi waktu, salah satunya dengan menggunakan media komputer. Oleh sebab itu penulis menggunakan gagasan untuk mengembangkan suatu aplikasi yang dapat dijalankan di komputer.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang maka timbul suatu permasalahan bagaimana membangun dan mengembangkan suatu aplikasi tes potensi

akademik disertai tes buta warna untuk mengetahui hasil yang lebih akurat dan efisiensi waktu.

### **1.3 Tujuan Penelitian**

Tujuan tugas akhir ini adalah membuat dan merancang aplikasi tes potensi akademik disertai tes buta warna guna meningkatkan efisiensi waktu dan kinerja suatu instansi yang bidangnya berkaitan dengan warna.

### **1.4 Batasan Masalah.**

Agar permasalahan mengarah sesuai dengan tujuan maka pembahasan dibatasi pada hal-hal sebagai berikut :

1. Aplikasi ini hanya dipergunakan untuk tes masuk pada suatu instansi.
2. Tidak membahas lebih jauh masalah sistem jaringan yang digunakan.
3. Tidak membahas lebih jauh masalah sistem keamanan yang digunakan.
4. Program dibuat dengan menggunakan bahasa pemrograman Delphi 7.0. dan serta penggunaan MYSQL sebagai databasenya.

### **1.5 Metodologi Penelitian**

Skripsi ini disusun dengan menggunakan metode:

1. Studi Literatur

Mempelajari dan mengumpulkan literatur baik dari buku-buku, maupun internet yang berkaitan dengan judul sehingga dapat mendukung penyusunan skripsi ini.

## 2. Perancangan

Merancang dan mengembangkan desain aplikasi tes potensi akademik disertai tes buta warna.

## 3. Pembuatan Program

Pembuatan Aplikasi dengan menggunakan bahasa DELPHI.

## 4. Implementasi & Pengujian

Merancang dan mengembangkan aplikasi tes potensi akademik disertai tes buta warna. berdasarkan data-data yang diperoleh.

## 1.6 Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan skripsi ini, maka peneliti menyajikan secara sistematis sebagai berikut :

### Bab I : Pendahuluan

Berisi Latar Belakang, Perumusan Masalah, Tujuan dan Manfaat Penelitian, Pembatasan Permasalahan, Metodologi Penelitian dan Sistematika Penulisan.

### Bab II : Tinjauan Pustaka

Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan pembahasan yang dilakukan, yang di dalamnya memuat teori-teori tentang aplikasi tes potensi akademik disertai tes buta warna.

**Bab III : Desain & Analisis System**

**Bab ini membahas Desain dan analisis dari aplikasi tes potensi akademik disertai tes buta warna.**

**Bab IV : Implementasi dan Pengujian Sistem**

**Berisi tentang Implementasi dari Hasil Desain Aplikasi.**

**Bab V : Kesimpulan dan saran**

**Berisi tentang kesimpulan yang dapat diambil berdasarkan hasil uraian pada bab-bab sebelumnya dan saran mengenai hasil yang telah diperoleh.**

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Inteligensi dan Bakat**

Inteligensi merupakan suatu konsep mengenai kemampuan umum individu dalam menyesuaikan diri dengan lingkungannya. Dalam kemampuan yang umum ini, terdapat kemampuan-kemampuan yang amat spesifik. Kemampuan-kemampuan yang spesifik ini memberikan pada individu suatu kondisi yang memungkinkan tercapainya pengetahuan, kecakapan, atau ketrampilan tertentu setelah melalui suatu latihan/pendidikan formal/informal. Inilah yang disebut bakat atau *Aptitude*. Karena tes inteligensi (tes IQ) tidak dirancang untuk menyingkap kemampuan-kemampuan khusus ini, maka bakat tidak dapat segera diketahui lewat tes inteligensi.

Alat yang digunakan untuk menyingkap kemampuan khusus ini disebut TES BAKAT atau *APTITUDE TEST*. Tes bakat yang dirancang untuk mengungkap prestasi belajar pada bidang tertentu dinamakan *Scholastic Aptitude Test/ Tes Bakat Skolastik (TBS)* . Contoh dari *Scholastic Aptitude Test* adalah Tes Potensi Akademik (TPA) .

## 2.2 Tes Potensi Akademik

Tes potensi akademik merupakan salah satu dari sekian jenis tes yang digunakan untuk mengukur dan memberi penilaian beberapa hal terkait dengan kemampuan dan potensi akademik seseorang dan sejauh ini penggunaannya sudah semakin luas. Misalnya sebagai salah satu pertimbangan dalam menentukan kapasitas dan kapabelitas seseorang yang biasanya disyaratkan dalam kategori tertentu seperti penerimaan siswa atau mahasiswa baru, penerimaan karyawan, seleksi pegawai negeri hingga pengukur kecenderungan pola pikir dan kemampuan intelektual anggota dewan. Bahkan akhir-akhir ini, tes potensi akademik menjadi bagian dari syarat kenaikan jabatan atau posisi-posisi tertentu pada perusahaan dan mulai merambah ke area tes kelayakan (proper test) untuk tugas-tugas tertentu.

Tes potensi akademik berisi bahan-bahan yang pada umumnya berisi :

a. Tes Verbal

Tes verbal terdiri dari :

1) Tes Diksi

Tes diksi adalah tes untuk memahami keluasan pengetahuan seseorang dalam kaitannya dengan varian pemilihan kata yang paling tepat dalam mengungkapkan gagasan dan pemikiran. .

2) Tes Lawan Pengetian

Tes ini bertujuan untuk mengetahui kemampuan mengartikan kata dari satu kata, dan kemudian mampu mencari lawan kata atau lawan arti dari kata tersebut.

**3) Tes Padan Kata**

Tes padan kata bertujuan untuk mengetahui pemahaman seseorang tentang pengindonesiaan kata-kata serapan dari bahasa asing. Pengetahuan tersebut sangat mendukung untuk diplomasi, negosiasi, presentasi, memahami bacaan, diskusi, dsb yang diperlukan dalam kerja suatu instansi atau perusahaan.

**4) Tes Analogi**

Tes analogi ini merupakan perbandingan kasus sejenis. Tujuan tes ini adalah untuk mengetahui kemampuan seseorang dalam kaitannya dengan pengambilan keputusan.

**b. Tes Numerik**

Tes Numerik terdiri dari :

**1) Tes Berhitung Cepat**

Tes ini tidak ada hal yang potensial menjebak jawaban, karena hampir tergantung sepenuhnya kepada kemampuan dan kecepatan numerik seseorang.

**2) Tes Deret Matematika**

Tes ini bertujuan untuk menggali kecerdasan numerik seseorang. Tes ini sepiantas akan terlihat sangat sulit, akan tetapi apabila sudah memahami trik serinya, maka akan sangat mudah mengerjakannya.

### 3) Tes Deret Huruf

Tes ini hampir sama dengan tes deret matematika, yang membedakan adalah tes deret huruf menggunakan deret didasarkan pada posisi huruf.

#### c. Tes Penalaran (Logika)

Tes logika berfungsi mengukur kemampuan seseorang dalam penalaran dan pemecahan persoalan secara logis atau masuk akal. Tes logika ini meliputi tes logika umum, tes analisa pernyataan dan kesimpulan (silogisme), tes logika cerita dan tes logika diagram.

#### d. Tes Ruang (Spatial)

Tes ini bertujuan untuk mengetahui kecerdasan akademis seseorang dalam kaitannya dengan ruang geografis. Tes ini menggunakan berbagai trik, namun semuanya berkaitan dengan pemahaman ruang.

## 2.2 Tes Buta Warna

Tes buta warna merupakan tes yang dipergunakan untuk mengetahui seseorang mengalami kelainan mata (buta warna) atau tidak. Tes buta warna diperlukan pada instansi yang bidang pekerjaannya berhubungan dengan persepsi warna. Sebagai contoh : instansi kesehatan, instansi tekstil, instansi kimia, dll.

Buta warna merupakan kelainan genetik / bawaan yang diturunkan dari orang tua kepada anaknya, kelainan ini sering juga disebut *sex linked*, karena kelainan ini dibawa oleh kromosom X, artinya kromosom Y tidak membawa

faktor buta warna. Hal inilah yang membedakan antara penderita buta warna pada laki-laki dan wanita. Seorang wanita terdapat istilah 'pembawa sifat' hal ini menunjukkan ada satu kromosom X yang membawa sifat buta warna. Wanita dengan pembawa sifat, secara fisik tidak mengalami kelainan buta warna sebagaimana wanita normal pada umumnya. Tetapi wanita dengan pembawa sifat berpotensi menurunkan faktor buta warna kepada anaknya kelak. Apabila pada kedua kromosom X mengandung faktor buta warna maka seorang wanita tersebut menderita buta warna.

Tes buta warna pada aplikasi ini menggunakan tes *ishihara*. Tes *ishihara* berupa lingkaran - lingkaran berwarna yang beberapa diantaranya dirancang agar ada tulisan tertentu yang hanya dapat dilihat atau tidak dapat dilihat oleh penderita buta warna..

## 2.3 Sistem Penilaian

### 2.3.1 Tes Potensi Akademik

Soal tes potensi akademik berjumlah 55 soal

Untuk jawaban benar diberi nilai 1.

Untuk jawaban salah diberi nilai 0.

$$\frac{(\sum \text{jawaban benar} - \sum \text{jawaban salah})}{\sum \text{soal}} \times 100\%$$

$$\sum \text{soal}$$

Kriteria Penilaian :

- Baik bila  $\geq 76\%$

- Cukup bila 56 % - 75 %
- Kurang bila  $\leq$  55 %

### 2.3.2 Tes Buta Warna

Soal tes buta warna berjumlah 12 soal.

Untuk jawaban benar diberi nilai 1.

Untuk jawaban salah diberi nilai 0.

- Jawaban benar sebanyak lebih dari sama dengan 8 maka peserta tidak mengalami buta warna / normal.
- Jawaban benar sebanyak kurang dari sama dengan 7 maka peserta mengalami buta warna sebagian.
- Jawaban benar kurang dari sama dengan 2 maka peserta mengalami buta warna total.

## 2.4 Delphi 7.0

Delphi 7 adalah paket bahasa pemrograman yang bekerja dalam sistem operasi Windows, yang mempunyai cakupan kemampuan yang luas dan sangat canggih. Beberapa jenis aplikasi yang dapat kita buat dengan Delphi, termasuk aplikasi untuk mengolah teks, grafik, angka, database dan aplikasi web.

Secara umum kemampuan Delphi adalah menyediakan komponen – komponen dan bahasa pemrograman yang andal, sehingga memungkinkan

kita untuk membuat program aplikasi sesuai dengan keinginan, dengan tampilan dan kemampuan yang canggih.

Untuk mempermudah pemrograman dalam membuat program aplikasi, Delphi menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas pemrograman tersebut dibagi dalam dua kelompok, yaitu object dan bahasa pemrograman. Secara ringkas, object adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat diliat (visual). Object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan – batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu.

Khusus untuk pemrograman database, Delphi menyediakan object yang sangat kuat, canggih dan lengkap, sehingga memudahkan pemrograman dalam merancang, membuat dan menyelesaikan aplikasi database yang diinginkan. Selain itu juga Delphi dapat menangani data dalam berbagai format database, misalnya format MySql, MS-Acces, Sybase, Oracle, FoxPro, Informix, DB2 dan lain – lain. Format database yang dianggap asli dari Delphi adalah Paradox dan dBase.

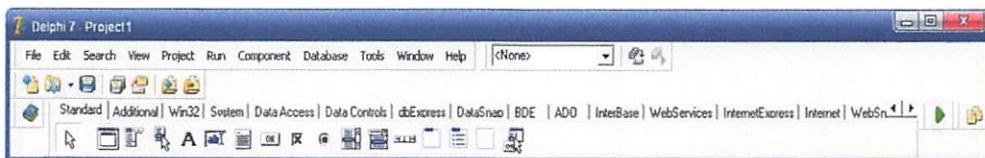
Bagian – bagian Delphi :

- **Menu Bar & Tool Bar**

Menu bar merupakan tempat menggulung (pull-down) menu – menu perintah. sebuah menu terdiri dari beberapa daftar perintah (menu

command). Disebelah kiri masing – masing perintah terdapat sebuah ikon yang menggambarkan fungsinya.

Tool Bar adalah sekumpulan tombol yang dapat digunakan untuk melakukan suatu perintah. Fungsi Tool Bar sama dengan Menu Bar, namun keunggulannya adalah cepat diakses untuk menjalankan perintah tertentu.



Gambar 2.1 Menu Bar dan Tool Bar

- Component Pallette

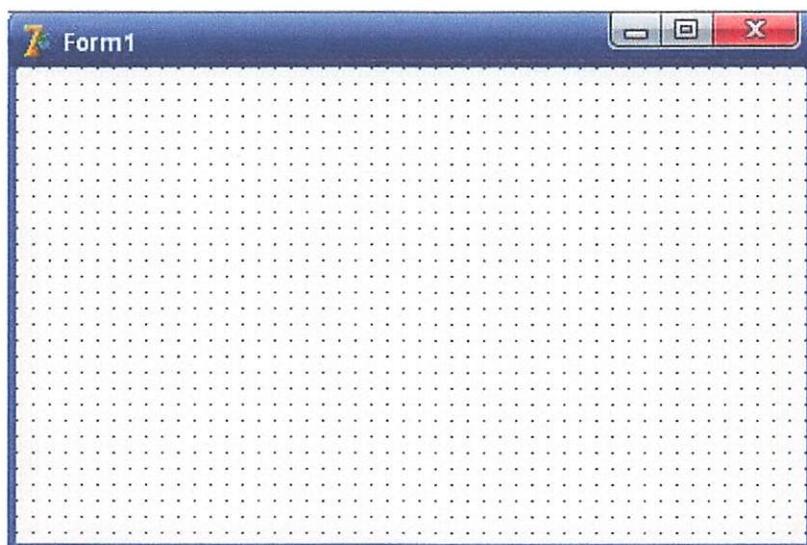
Component Palette digunakan untuk mengumpulkan komponen Delphi yang dibagi menjadi beberapa katagori. Secara garis besar komponen Delphi dapat dibagi menjadi dua jenis, yaitu komponen visual dan komponen non visual. Komponen visual adalah komponen yang terlihat pada saat proses mendesain aplikasi maupun setelah aplikasi dijalankan. komponen non visual adalah komponen yang hanya terlihat pada saat proses mendesain aplikasi, namun tidak dapat terlihat pada saat aplikasi dijalankan.



Gambar 2.2 Component Pallette

- Form Designer

Form Designer atau Form adalah tempat meletakkan komponen – komponen yang diambil dari komponent Palette saat membuat desain antarmuka aplikasi. Ketika menjalankan Delphi secara default maka akan tampil form secara otomatis.



Gambar 2.3 Form Designer

- Code Editor

Code Editor adalah tempat untuk menuliskan kode – kode program.

Didalam Code Editor telah ada beberapa baris yang dibuat Delphi secara otomatis. Ditempat inilah menyisipkan atau menuliskan kode – kode program secara lengkap.

```
Unit1.pas
Unit1

unit Unit1;

interface

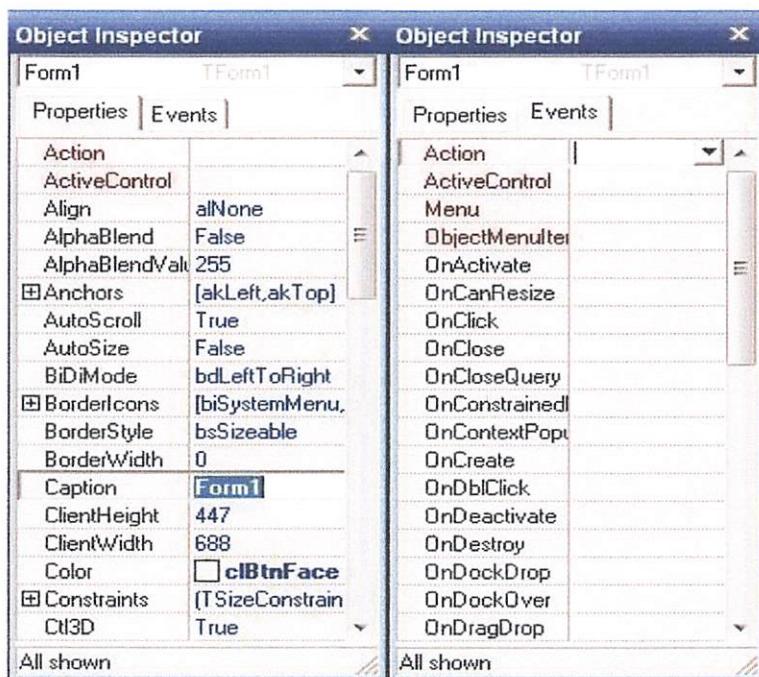
uses
  Windows, Messages, SysUtils, Variants, Classes,
  Dialogs;

type
  TForm1 = class(TForm)
private
  { Private declarations }
public
  { Public declarations }
```

Gambar 2.4 Code Editor

- Object Inspector

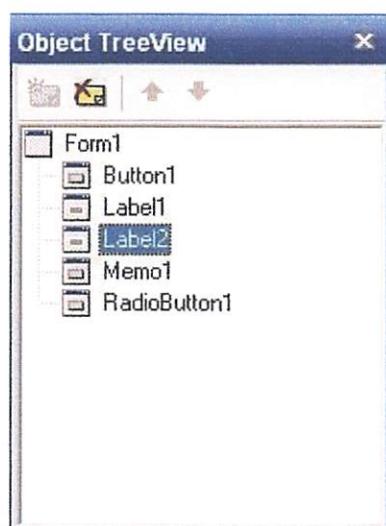
Object Inspector digunakan untuk mengubah properti komponen terpilih yang berada dalam di Form. Ada dua bagian utama pada Object Inspector, yaitu properties dan event. Pada setiap bagian tersebut terdapat dua buah kolom. Kolom disebelah kiri merupakan nama – nama properti atau event. sedangkan kolom disebelah kanan adalah nilai – nilai yang diisikan pada properti atau event yang bersangkutan.



Gambar 2.5 Object Inspector

- Object Tree View

Object TreeView berfungsi untuk menampilkan berbagai komponen yang digunakan pada saat membuat aplikasi. Komponen – komponen ditampilkan dengan struktur pohon.



Gambar 2.6 Object Tree View

## 2.5 MySQL

MySQL merupakan Relational Database Management Sistem (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structure Query Language).

SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan/seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu system database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan dengan database server yang lainnya dalam query data.

Sebagai database yang memiliki konsep database modern, MySQL memiliki banyak sekali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL :

- Portability. MySQL dapat berjalan stabil pada berbagai sistem operasi di antaranya adalah seperti Windows, Linux, FreeBSD, Mac OS X server, Solaris, Amiga, HP-UX dan masih banyak lagi.
- Open Source. MySQL didistribusikan secara open source (gratis), di bawah lisensi GPL.

- Multiuser. MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah database server MySQL dapat diakses client secara bersamaan.
- Performance Tuning. MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
- Column Types. MySQL memiliki tipe kolom yang sangat kompleks, seperti signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, year, set serta enum.
- Command dan Function. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.
- Security. MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan user dengan system perizinan yang mendetail serta password terencripsi.
- Stability dan Limits. MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu table serta 5 miliar baris. Selain itu, batas indeks yang dapat di tumpung mencapai 32 indeks pada tiap tabelnya.
- Connectivity. MySQL dapat melakukan koneksi dengan client menggunakan protocol TCP/IP, Unix soket (Unix), atau Named Pipes (NT).
- 10. Localisation MySQL dapat mendeteksi pesan kesalahan (error code) pada client dengan menggunakan lebih dari dua puluh

bahasa. Meski demikian, bahasa Indonesia belum termasuk di dalamnya.

- Localisation. MySQL dapat mendeteksi pesan kesalahan (error code) pada client dengan menggunakan lebih dari dua puluh bahasa. Meski demikian, bahasa Indonesia belum termasuk di dalamnya.
- Interface. MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
- Client dan Tools. MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertai petunjuk online.
- Struktur Tabel. MySQL memiliki struktur table yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan database lainnya semacam PostgreSQL ataupun Oracle.

## **BAB III**

### **PERANCANGAN DAN DESAIN SISTEM**

#### **3.1 Analisa Sistem**

Bab ini berisi penjelasan mengenai perancangan dan pembuatan sistem aplikasi Tes Potensi Akademik disertai Tes Buta Warna Menggunakan Delphi & MYSQL. Perancangan desain sistem digunakan untuk memberikan gambaran secara umum terhadap aplikasi.

Dalam merancang aplikasi pada proyek akhir ini terlebih dahulu dilakukan pembuatan desain antar muka aplikasi dan menentukan fasilitas yang dimasukan ke dalam aplikasi. Desain antarmuka dan fasilitas berguna untuk mempermudah pengguna aplikasi ini untuk mengoperasikannya.

#### **3.2 Deskripsi Sistem**

Aplikasi tes potensi akademik disertai tes buta warna merupakan sebuah tes yang bertujuan untuk mengukur dan memberi penilaian beberapa hal terkait dengan kemampuan dan potensi akademik seseorang, sedangkan tes buta warna dipergunakan untuk mengetahui kondisi mata seseorang normal atau mengalami kelainan pada matanya.

Dalam pengoperasian aplikasi ini admin terlebih dahulu meng-*entry* data peserta tes, data tersebut dimasukkan dan disimpan ke dalam database. Setelah data peserta tes disimpan di database, pada aplikasi peserta, peserta wajib mengisi nomor tes dan nama sesuai dengan kartu peserta. Peserta yang

telah memasukkan nama dan nomor tes secara benar akan mendapatkan soal-soal untuk dikerjakan. Soal peserta tes ditampilkan dengan cara *random*. Hal ini bertujuan untuk menghindari kerja sama antar peserta tes, sehingga tidak perlu pengawas untuk mengawasi peserta tes pada saat tes dilaksanakan. Untuk penilaian tes, setiap jawaban peserta akan dikirim ke dalam database yang kemudian dicocokkan dengan jawaban yang tersimpan di database.

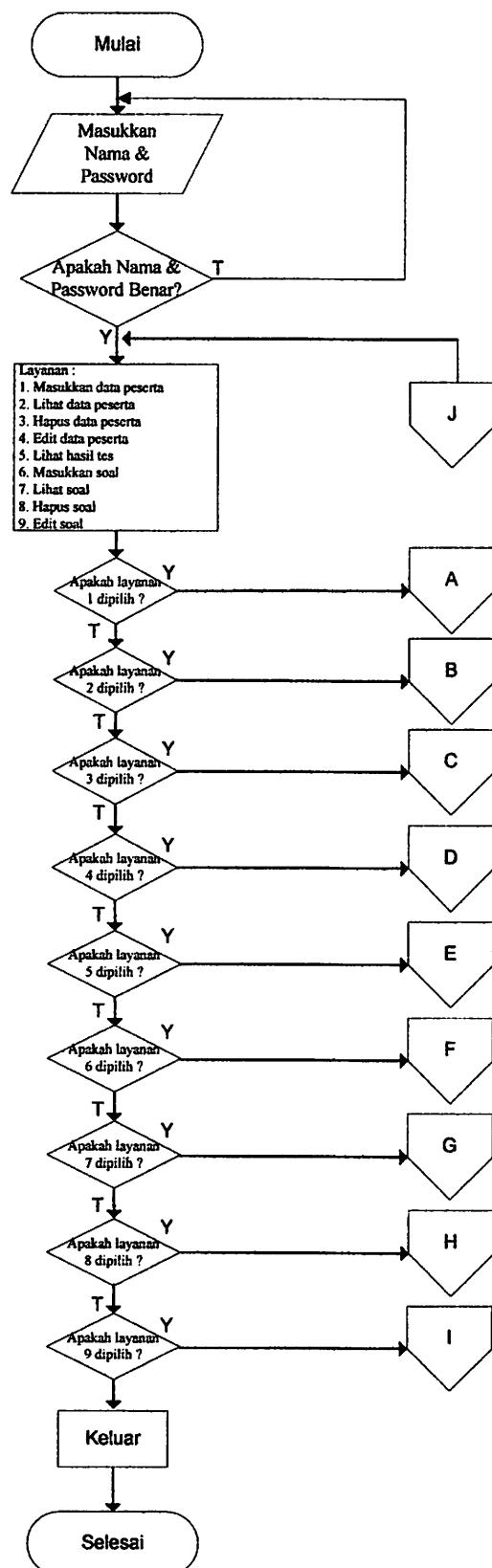
### **3.3 Perancangan Sistem**

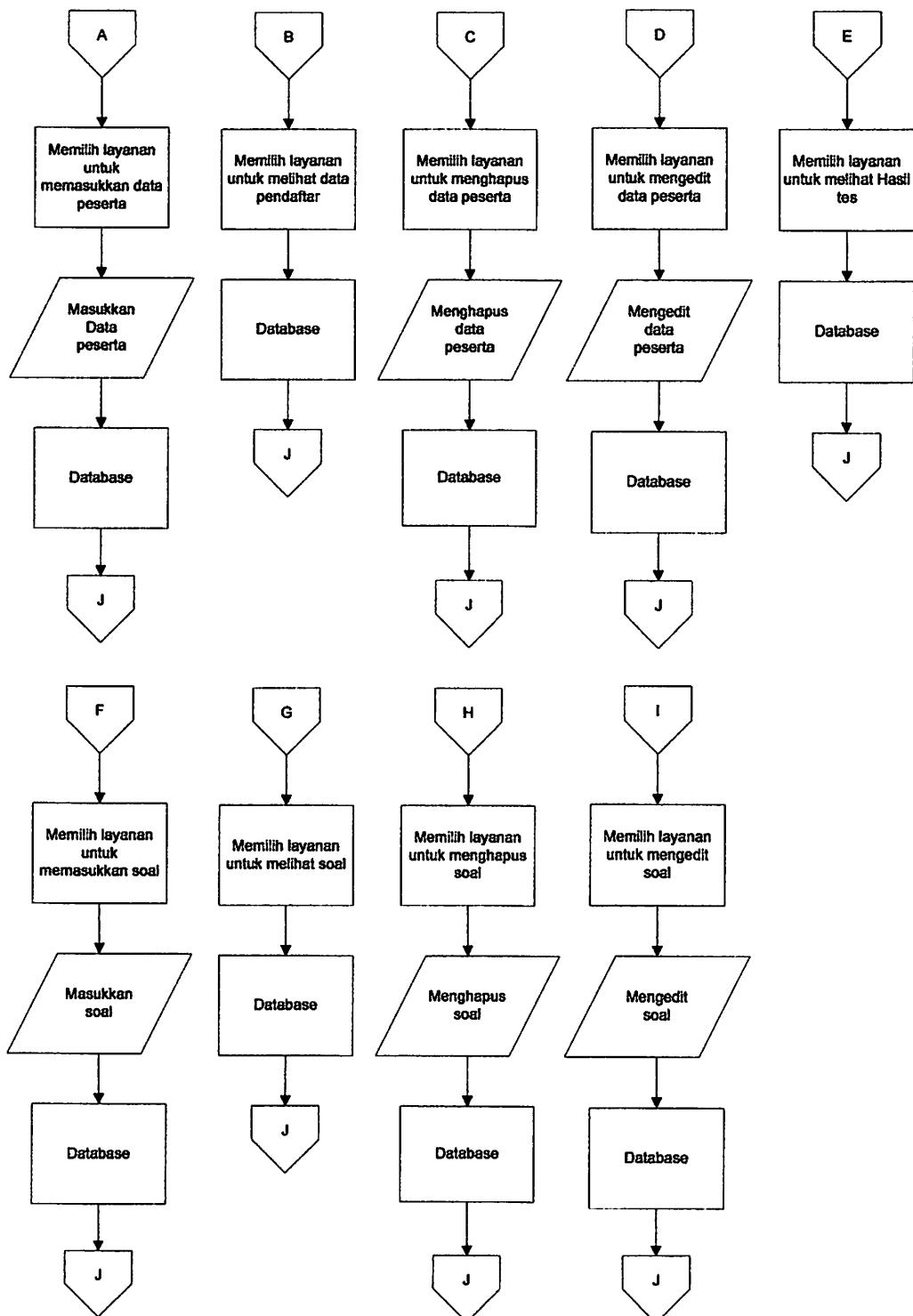
Dalam merancang suatu aplikasi, yakni aplikasi tes potensi akademik disertai tes buta warna diperlukan suatu desain sistem yang dibuat dengan pemodelan proses. Model proses dari aplikasi ini dibuat dalam bentuk *flowchart* dan alur proses.

Pada perancangan sistem ini akan dijelaskan bagaimana flowchart dan alur proses dari aplikasi ini:

#### **3.3.1 *Flow Chart* Sistem**

##### **3.3.1.1 *Admin***



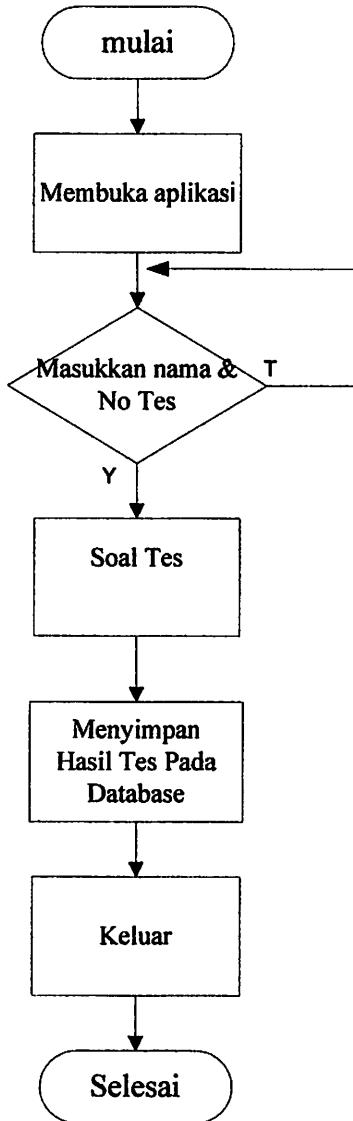


Gambar 3.1 Flowchart Admin

**Keterangan:**

- Masukkan nama & password secara benar, jika nama / password salah maka halaman admin tidak akan bisa dioperasikan.
- Jika nama & password benar, aplikasi admin akan terbuka.
- Dalam admin terdapat layanan :
  - Masukkan data peserta, digunakan untuk memasukkan data peserta.
  - Lihat data peserta, digunakan untuk melihat data peserta yang telah disimpan di database.
  - Hapus data peserta, digunakan untuk menghapus data peserta.
  - Edit data peserta, digunakan untuk mengedit data peserta.
  - Masukkan soal, digunakan untuk memasukkan tes potensi akademik.
  - Melihat soal, digunakan untuk melihat soal yang telah disimpan di database.
  - Menghapus soal, digunakan untuk menghapus soal yang telah disimpan di database.
  - Mengedit soal , digunakan untuk merubah soal yang ada pada database.
  - Lihat hasil tes, digunakan untuk melihat hasil tes .

### 3.3.1.2 Flow Chart Peserta



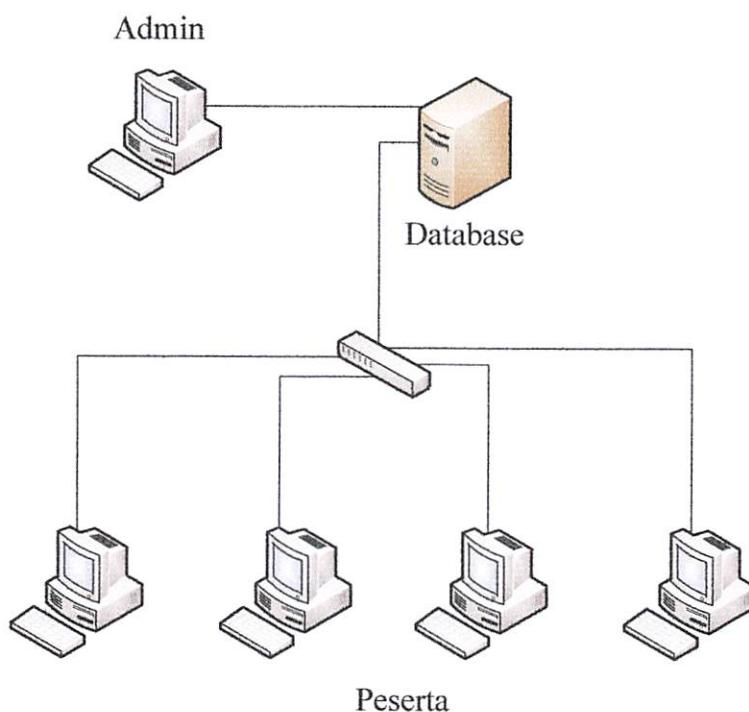
Gambar 3.2 *Flowchart* Peserta

Keterangan:

- Membuka aplikasi Peserta
- Masukkan nama dan no tes sesuai dengan data masing-masing peserta.
- Jika nama dan no tes telah sesuai, peserta akan mendapatkan soal, yang selanjutnya dikerjakan oleh para peserta.

- Setelah peserta selesai mengerjakan soal, hasil dari pekerjaan di simpan di database.

### 3.3.2 Alur Proses



Gambar 3.3 Desain Alur Proses

#### Keterangan

- Admin memasukkan data peserta tes potensi akademik disertai tes buta warna ke dalam database.
- Database dipergunakan untuk menyimpan soal, data peserta, data admin, dan hasil tes para peserta.
- User / peserta tes potensi akademik disertai tes buta warna mengerjakan soal, dimana soal yang ditampilkan di aplikasi user diambil dari database.
- Jawaban dari peserta disimpan ke dalam database.

### 3.3.3 Perancangan Database

Dalam aplikasi tes potensi akademik disertai tes buta warna diperlukan suatu database. Database ini berfungsi untuk menyimpan data yang ada di dalam suatu aplikasi. Dalam aplikasi tes potensi akademik disertai tes buta warna ini menggunakan database MySQL.

#### 3.3.3.1 Tabel Pada Aplikasi *Admin*

- Tabel *Admin*

Tabel 3.1. Struktur Tabel *Admin*

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Char	2	*	Not Null
Nama	Char	30		Not Null
Alamat	Char	30		Not Null
No_tlpn	Char	15		Not Null
Password	Char	10		Not Null
Jabatan	Char	15		Not Null

Fungsi : Untuk menyimpan data *admin*.

Primary Key : no.

- Tabel peserta

Tabel 3.2. Struktur Tabel Peserta

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	5	*	Not Null
Nama	Char	30		Not Null
Alamat	Char	30		Not Null
Ttl	Char	30		Not Null
Kelamin	Char	10		Not Null
Telpun	Char	15		Not Null

Fungsi : Untuk menyimpan data peserta.

Primary Key : No\_test.

### 3.3.3.2 Tabel Pada Aplikasi Peserta

- Tabel Soal
  - Soal\_1\_5

Tabel 3.3. Struktur Tabel Soal\_1\_5

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 1-5.

Primary Key : no.

- Soal\_6\_7

Tabel 3.4. Struktur Tabel Soal\_6\_7

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 6-7.

Primary Key : no.

- Soal\_8\_10

Tabel 3.5. Struktur Tabel Soal\_8\_10

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 8-10.

Primary Key : no.

- Soal\_11\_25

Tabel 3.6. Struktur Tabel Soal\_11\_25

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 11-25.

Primary Key : no.

- Soal\_26\_35

Tabel 3.7. Struktur Tabel Soal\_26\_35

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 26 – 35.

Primary Key : no.

- Soal\_36\_45

Tabel 3.8. Struktur Tabel Soal\_36\_45

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 36 – 45.

Primary Key : no.

- Soal\_46\_48

Tabel 3.9. Struktur Tabel Soal\_46\_48

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 46 – 48.

Primary Key : no.

- Soal\_49\_52

Tabel 3.10. Struktur Tabel Soal\_49\_52

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 53 – 55.

Primary Key : no.

- Soal\_53\_55

Tabel 3.11. Struktur Tabel Soal\_53\_55

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Varchar	1000		Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal no 53 – 55.

Primary Key : no.

- Buta

Tabel 3.12. Struktur Tabel Buta

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No	Int	2	*	Not Null
Soal	Longblob			Not Null
a	Varchar	1000		Not Null
b	Varchar	1000		Not Null
c	Varchar	1000		Not Null
d	Varchar	1000		Not Null
e	Varchar	1000		Not Null
Jawaban	Char	2		Not Null

Fungsi : Untuk menyimpan soal tes buta warna.

Primary Key : no.

- Tabel Jawaban

- J\_1\_5

Tabel 3.13. Struktur Tabel J\_1\_5

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 1 – 5.

Primary Key : no.

- J\_6\_7

Tabel 3.14. Struktur Tabel J\_6\_7

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 6 – 7.

Primary Key : no.

- J\_8\_10

Tabel 3.15. Struktur Tabel J\_8\_10

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 8 – 10.

Primary Key : no.

- J\_11\_25

Tabel 3.16. Struktur Tabel J\_11\_25

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 11 – 25.

Primary Key : no.

- J\_26\_35

Tabel 3.17. Struktur Tabel J\_26\_35

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 26 – 35.

Primary Key : no.

- J\_36\_45

Tabel 3.18. Struktur Tabel J\_36\_45

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 36 – 45.

Primary Key : no.

- J\_46\_48

Tabel 3.19. Struktur Tabel J\_46\_48

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 46 – 48.

Primary Key : no.

- J\_49\_52

Tabel 3.20. Struktur Tabel J\_49\_52

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 49 – 52.

Primary Key : no.

- J\_53\_55

Tabel 3.21. Struktur Tabel J\_53\_55

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban no 53 – 55.

Primary Key : no.

- J\_butu

Tabel 3.21. Struktur Tabel J\_butu

Nama Field	Tipe Data	Panjang	Primary Key	Keterangan
No_test	Int	8	*	Not Null
No	Int	8	*	Not Null
No_soal	Int	8		Not Null
Jawaban	Char	1		Not Null
Status	Int	1		Not Null

Fungsi : Untuk menyimpan jawaban tes buta warna.

Primary Key : no.

### 3.3.4 Perancangan Program

Sebelum memasuki menu yang dimiliki oleh aplikasi tes potensi akademik disertai tes buta warna, *admin / user* terlebih dahulu melakukan proses *login* untuk membuktikan bahwa ia berstatus *admin / user*. Jika tidak memiliki *account* maka tidak akan bisa mengoperasikan aplikasi ini.

Setelah melakukan *login*, *admin / peserta* memasuki menu-menu yang ditampilkan.

#### 3.3.4.1 Admin

Dalam *Admin* memungkinkan untuk menggunakan fasilitas-fasilitas yang dimiliki *adminr*. Menu *admin* antara lain :

- Masukkan Data Peserta

Fasilitas masukkan data peserta merupakan suatu layanan yang dipergunakan oleh admin untuk memasukkan data peserta tes yang akan mengikuti tes potensi akademik disertai tes buta warna. Data peserta yang dimasukkan oleh *admin* disimpan pada tabel peserta.

- Lihat Data Peserta

Fasilitas lihat data peserta dipergunakan oleh *admin* untuk melihat data peserta yang telah tersimpan di database.

- Hapus Data Peserta

Fasilitas hapus data peserta merupakan suatu layanan admin yang berfungsi untuk menghapus data peserta yang telah tersimpan pada database.

- ***Edit Data Peserta***

Fasilitas edit data peserta merupakan suatu layanan admin yang berfungsi untuk merubah data peserta yang telah tersimpan pada database.

- **Masukkan Soal**

Fasilitas masukkan soal merupakan suatu layanan admin yang berfungsi untuk memasukkan soal ke dalam database.

- **Lihat Soal**

Fasilitas lihat soal merupakan suatu layanan admin yang berfungsi untuk melihat soal yang telah tersimpan pada database.

- **Hapus Soal**

Fasilitas hapus soal merupakan suatu layanan admin yang berfungsi untuk menghapus soal yang telah tersimpan pada database.

- ***Edit Soal***

Fasilitas lihat soal merupakan suatu layanan admin yang berfungsi untuk merubah soal yang telah tersimpan pada database.

- **Lihat Hasil Tes**

Fasilitas lihat hasil tes merupakan layanan admin yang berfungsi untuk melihat hasil tes peserta.

### **3.3.4.2 Peserta**

- **Soal Tes**

Fasilitas soal tes merupakan layanan yang dipergunakan oleh para peserta / user, untuk mengerjakan tes, dimana tes tersebut terdiri dari

tes potensi akademik yang berjumlah 55 soal yang, dan tes buta warna yang terdiri dari 12 soal. Soal yang ditampilkan pada aplikasi user diacak sehingga setiap peserta mendapatkan soal yang berbeda-beda.

### 3.3.5 Desain Antarmuka Aplikasi

Desain antarmuka aplikasi dibagi dalam dua bagian, yaitu aplikasi *admin* dan aplikasi peserta.

#### 3.3.5.1 Aplikasi *Admin*

- Halaman *Login*

The diagram shows a wireframe of a login interface. At the top, there is a horizontal bar labeled "Admin". Below this, there is a large rectangular area. Inside this area, there are two smaller rectangles stacked vertically. The top rectangle is labeled "Username" and the bottom one is labeled "Password". All elements are represented by simple black outlines on a white background.

Gambar 3.4 *Login Admin*

- **Menu Utama**

Admin		
Menu	Soal	Lihat Hasil Tes
Masukkan Data Peserta		
Lihat Data Peserta		
Hapus Data Peserta		
Edit Data Peserta		
Exit		

Gambar 3.5 Menu Utama

- **Halaman Masukkan Data Peserta**

Admin	
<b>Form Isian</b>	

Gambar 3.6 Masukkan Data Peserta

- #### • Halaman Lihat Data Peserta

Tabel Peserta						

**Gambar 3.7 Lihat Data Peserta**

- #### • Halaman Hapus Data Peserta

<b>Tabel Peserta</b>					

**Form  
Isian**

Gambar 3.8 Hapus Data Peserta

- #### • Halaman Edit Data Peserta

Gambar 3.9 Edit Data Peserta

- Halaman masukkan soal, lihat soal, edit soal, hapus soal

Gambar 3.10 Masukkan Soal, Lihat Soal, Edit Soal, Hapus Soal

- Lihat Hasil Tes

Admin					
<b>Tabel Hasil Tes</b>					

Gambar 3.11 Lihat Hasil Tes

- Halaman Masukkan Data Peserta

Admin					
<input type="text"/>					
<b>Form Isian</b>					

Gambar 3.12 Masukkan Soal Gambar

### 3.3.5.2 Aplikasi Peserta

- Halaman Login

The screenshot shows a simple login interface. At the top, there is a header bar with the word "Admin". Below this, there are two input fields: one labeled "Nama" (Name) and another labeled "Nomer Tes" (Test Number). Both fields are enclosed in rectangular boxes.

Gambar 3.13 *Login* Peserta

- Halaman Soal

The screenshot displays a test question page. At the top, it says "Tes Potensi Akademik disertai Tes Buta Warna". Below this, there are several sections: "Petunjuk penggerjaan soal" (Instructions for solving the question) and "Jawaban peserta" (Participant's answer), both in their own boxes; a large central box labeled "Soal" (Question); and five smaller boxes labeled "Pilihan jawaban A", "Pilihan jawaban B", "Pilihan jawaban C", "Pilihan jawaban D", and "Pilihan jawaban E".

Gambar 3.14 Soal

## **DAB IV**

### **IMPLEMENTASI DAN PENGUJIAN SISTEM**

#### **4.1 Pengujian Sistem**

Pengujian ini bertujuan untuk mengetahui sejauh mana keberhasilan aplikasi tes bila diterapkan dalam sistem. Pengujian ini juga untuk mencari letak *error* atau kesalahan program dalam aplikasi. Pengujian ini akan dicoba pada 2 komputer yang saling terhubung / terkoneksi.

##### **4.1.1 Metode Pengoperasian**

Pada pengoperasian aplikasi ini diharapkan pemakai memenuhi prosedur yang ditetapkan. Prosedur yang ditetapkan adalah sebagai berikut :

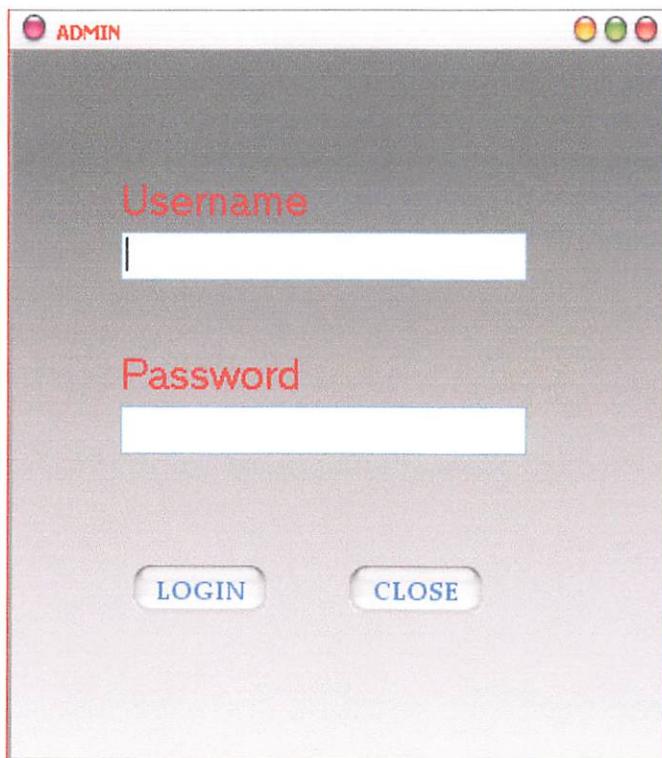
1. Koneksikan 2 komputer atau lebih .
  2. Untuk komputer peserta *instal* mysql-connector-odbc-5.1.5-win32.msi.
  3. Setting ODBC, tentukan alamat database server.
  4. Jalankan aplikasi peserta untuk mengerjakan tes potensi akademik disertai tes buta warna.
  5. Jalankan aplikasi admin untuk mengolah data peserta.
- .

#### **4.2 Pengujian Aplikasi *Admin***

##### **4.2.1 Pengujian *Login Admin***

Aplikasi *admin* adalah aplikasi yang disediakan untuk mengolah data peserta, mengolah soal tes, dan mengetahui hasil tes. Untuk akses ke dalam

menu utama, *admin* harus *login* terlebih dahulu dengan cara memasukkan *username* dan *password*.



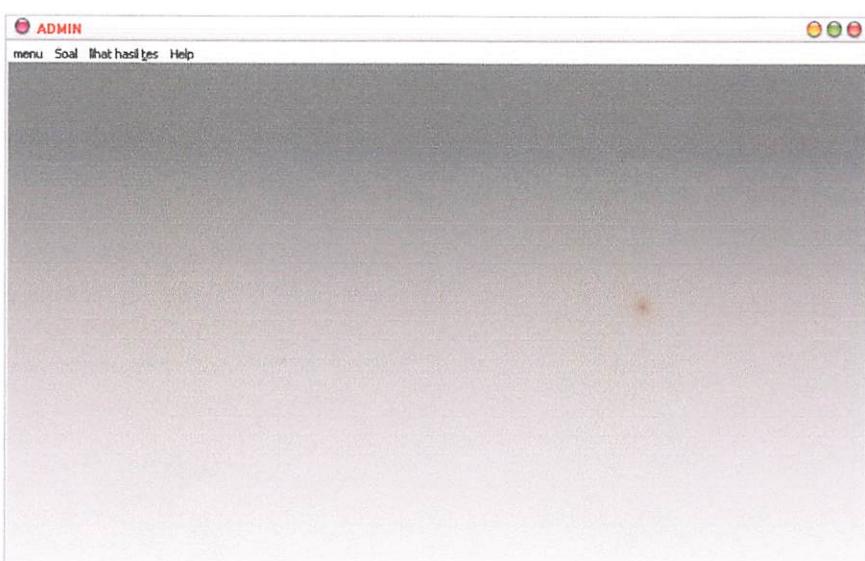
Gambar 4.1 Halaman *Login Admin*

Apabila username dan password yang diinputkan oleh admin salah maka akan keluar peringatan, admin harus mengulang memasukkan username dan password agar bisa masuk ke menu utama.



Gambar 4.2 Error

Jika username dan pasword diinputkan secara benar, maka menu utama akan terbuka.

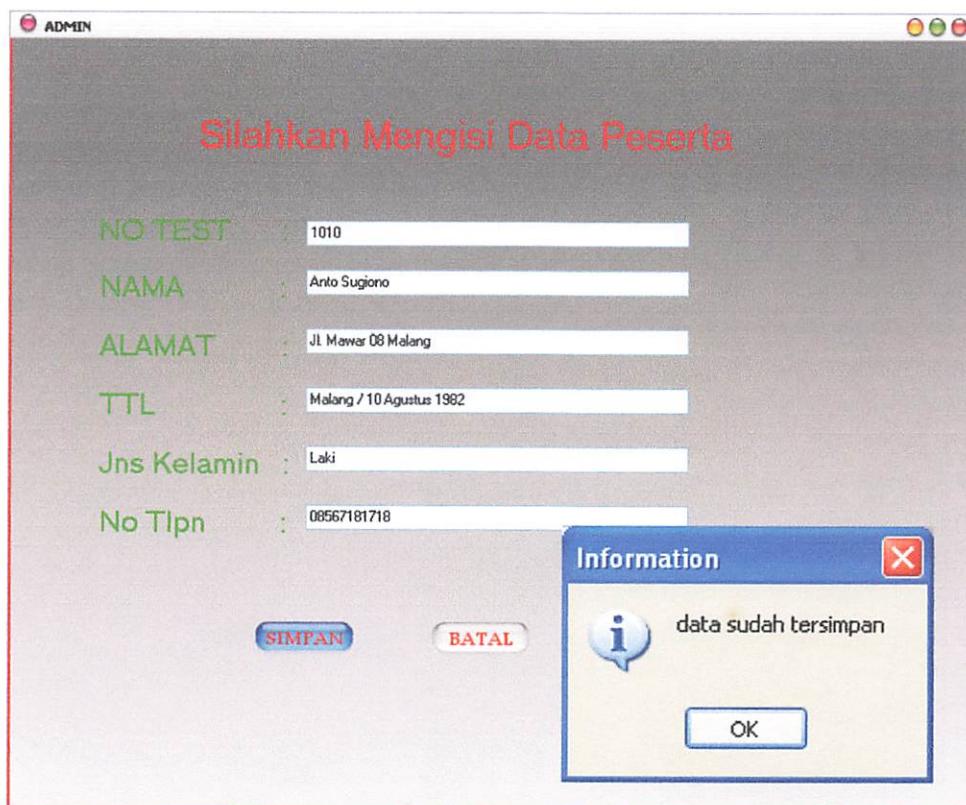


Gambar 4.3 Menu Utama

Pada menu utama terdapat beberapa layanan yang dipergunakan oleh admin untuk mengolah data peserta, mengolah soal, dan melihat hasil tes peserta tes potensi akademik disertai tes buta warna.

#### 4.2.2 Pengujian Masukkan Data Peserta

Untuk masuk ke dalam layanan masukkan data peserta, admin terlebih dahulu menekan tombol menu yang berada di sebelah kiri atas. Setelah menu ditekan, pilih layanan masukkan data peserta. Layanan masukkan data peserta digunakan untuk memasukkan data peserta yang terdiri dari no tes, nama, alamat, tempat tanggal lahir, dan no telpn.



Gambar 4.4 Masukkan Data Peserta

Pada pengujian layanan ini no test diisi '1010' , nama : 'Anto Suprapto' , alamat : 'Jl. Mawar 08 Malang', ttl : 'Malang / 10 Agustus 1982', jenis kelamin : 'laki-laki', dan no telpon : '08567181718'. Setelah data diisi kemudian tekan tombol simpan untuk menyimpan data peserta ke dalam database.

#### 4.2.3 Pengujian Lihat Data Peserta

Untuk masuk ke dalam layanan lihat data peserta, admin terlebih dahulu menekan tombol menu yang berada di sebelah kiri atas. Setelah menu ditekan, pilih layanan lihat data peserta. Layanan lihat data peserta dipergunakan untuk melihat data peserta yang telah disimpan dalam database.

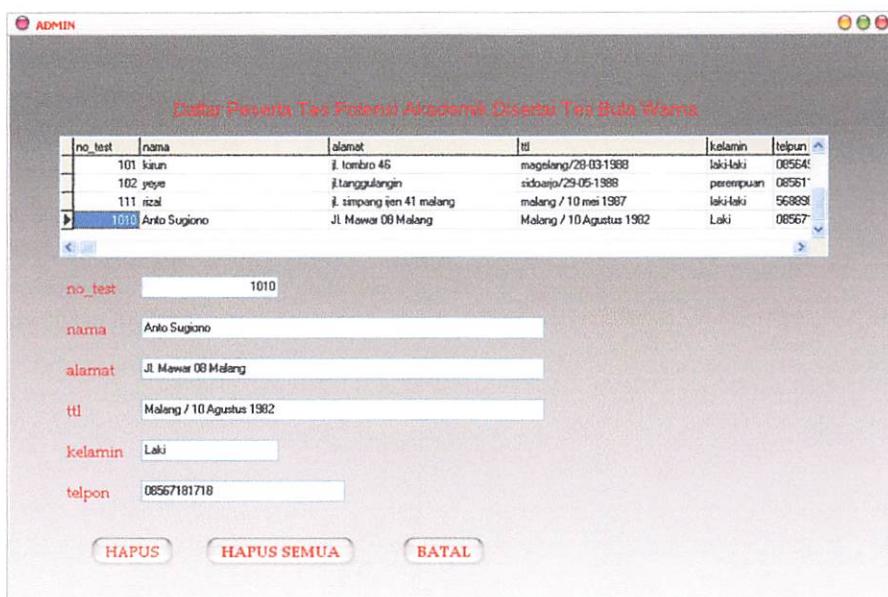
Data peserta yang telah tersimpan atas nama Anto Sugiono dengan nomer tes 1010 dapat ditampilkan pada layanan lihat data peserta.

no_test	nama	alamat	ttl	kelamin	telpon
100 rizal		Jl. simpang ijen	malang/10-05-1987	laki-laki	568998
101 kirun		Jl. tombo 46	magelang/28-03-1988	laki-laki	085649965064
102 peye		Jl.tanggulangin	sidoarjo/29-05-1988	perempuan	085611450459
111 rizal		Jl. simpang ien 41 malang	malang / 10 mei 1987	laki-laki	568998
1010 Anto Sugiono	Anto Sugiono	Jl. Mawar 08 Malang	Malang / 10 Agustus 1982	Laki	08567181718

Gambar 4.5 Lihat Data Peserta

#### 4.2.4 Pengujian Hapus Data Peserta

Untuk masuk ke dalam layanan hapus data peserta, admin terlebih dahulu menekan tombol menu yang berada di sebelah kiri atas. Setelah menu ditekan, pilih layanan hapus data peserta. Layanan hapus data peserta dipergunakan untuk menghapus data peserta yang telah tersimpan pada database.



Gambar 4.6 Hapus Data Peserta

Untuk pengujian layanan menghapus data peserta, terlebih dahulu pilih no test yang akan dihapus pada tabel, klik no test pada tabel. Disini yang dipilih yaitu no test : 1010, nama : Anto Sugiono, alamat :jl. Mawar 08 Malang, TTL : malang / 10 agustus 1982, jenis kelamin : laki-laki, telpon : 08567181718. Selanjutnya tekan tombol Hapus untuk menghapus data peserta dengan no Test : 1010. Sedangkan untuk menghapus semua data yang tersimpan pada database, tekan tombol hapus semua.

#### 4.2.5 Pengujian *Edit* Data Peserta

Untuk masuk ke dalam layanan *edit* data peserta, admin terlebih dahulu menekan tombol menu yang berada di sebelah kiri atas. Setelah menu ditekan, pilih layanan *edit* data peserta. Layanan *edit* data peserta dipergunakan untuk mengubah data peserta yang telah tersimpan pada database.

The screenshot shows a Windows application window titled "ADMIN". Inside, there is a table titled "Daftar Peserta Tes Potensi Akademik disertai Tes Buta Warna" with the following data:

no_test	nama	alamat	ttl	kel
100	rizal	jl. simpang ijen	malang/10-05-1987	laki
101	kirun	jl. tombo 46	magelang/28-03-1988	laki
102	yege	jl.tanggulangin	sidoarjo/29-05-1998	per

Below the table, there is an edit form for the student with no\_test 100:

no_test	100
nama	rizal
alamat	jl. simpang ijen
ttl	malang/10-05-1987
kelamin	laki-laki
telpon	568898

At the bottom are two buttons: "UBAH" (Change) and "BATAL" (Cancel).

Gambar 4.7 Edit Data Peserta

Untuk meng-*edit* data peserta tekan no test pada tabel yang berada pada jendela edit data peserta, kemudian ubah data yang telah tersimpan pada database, selanjutnya tekan tombol ubah untuk mengubah data peserta.

#### 4.2.6 Pengujian Masukkan Soal

Untuk masuk ke dalam layanan masukkan soal, admin terlebih dahulu menekan tombol soal yang berada di sebelah menu. Layanan masukkan soal dipergunakan untuk memasukkan soal pada database.

- Masukkan soal nomer 1 – 5

no	soal
1	BURUH : PABRIK
2	KERING : LEMBAB
3	PAUS : HIU
4	Mbenen : Dermawan
5	KEHAUSAN : AIR

no	6	A	bantal : guling
	KERIS : PUSAKA	B	Api : Neraka
		C	Jimat : Sakti
		D	Anak : Ibu
		E	Kamu : Dia
		Jawaban	C

Tambah    Ubah    Hapus    Hapus Semua    Kembali

Gambar 4.8 Masukkan Soal No 1 - 5

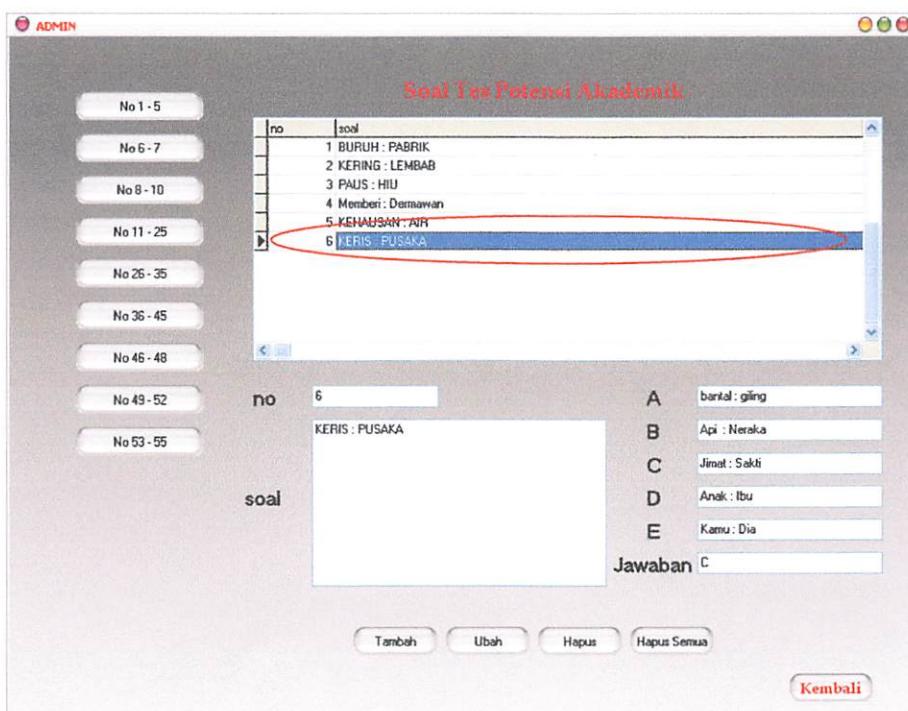
Pengujian layanan ini adalah memasukkan soal nomor 6 ke database, tetapi sebelum menginputkan data soal, admin terlebih dahulu harus menekan tombol No 1-5, setelah tombol No 1-5 ditekan, admin menginputkan no :’6’, soal :’KERIS : PUSAKA’, pilihan A : ’bantal:guling’, pilihan B : ’Api : Neraka’, pilihan C : ’Jimat : Sakti’, pilihan D : ’Anak : Ibu’, pilihan E : ’Kamu : Dia’, Jawaban : ’c’. Sesudah

memasukkan data soal, admin menekan tombol TAMBAH untuk menyimpan data soal pada database.

#### 4.2.7 Pengujian Lihat Soal

Untuk masuk ke dalam layanan lihat soal, admin terlebih dahulu menekan tombol soal yang berada di sebelah menu. Layanan lihat soal merupakan suatu layanan yang dipergunakan untuk melihat soal yang telah tersimpan pada database.

Data soal yang telah dimasukkan dalam database, yaitu no :’6’, soal :’KERIS : PUSAKA’, pilihan A : ’bantal:guling’, pilihan B : ’Api : Neraka’, pilihan C : ’Jimat : Sakti’, pilihan D : ’Anak : Ibu’, pilihan E: ’Kamu : Dia’, Jawaban : ’c’, dapat ditampilkan pada aplikasi ini. Untuk Kembali Ke menu utama tekan tombol KEMBALI.



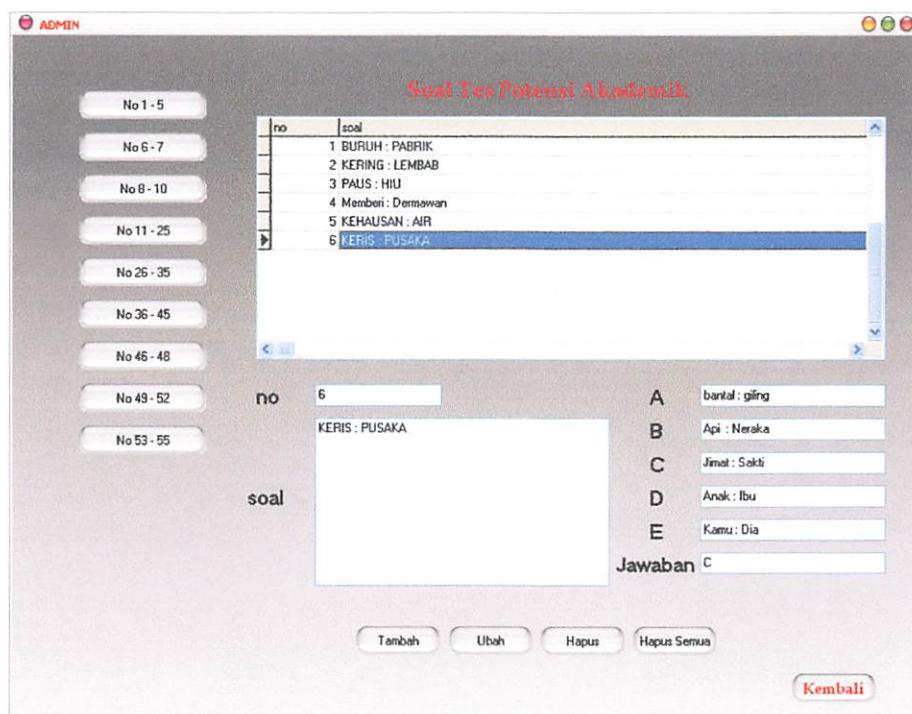
Gambar 4.9 Lihat Soal No 1 - 5

#### 4.2.8 Pengujian Hapus Soal

Untuk masuk ke dalam layanan hapus soal, admin terlebih dahulu menekan tombol soal yang berada di sebelah menu. Layanan hapus soal merupakan suatu layanan yang dipergunakan untuk menghapus sebagian atau semua soal yang telah tersimpan pada database.

Pada pengujian layanan hapus soal ini admin menghapus no : '6', setelah admin mengetik no : 6, langkah selanjutnya adalah menekan tombol hapus untuk menghapus soal nomor 6..

Untuk menghapus semua soal, admin diperintahkan untuk menekan tombol HAPUS SEMUA.

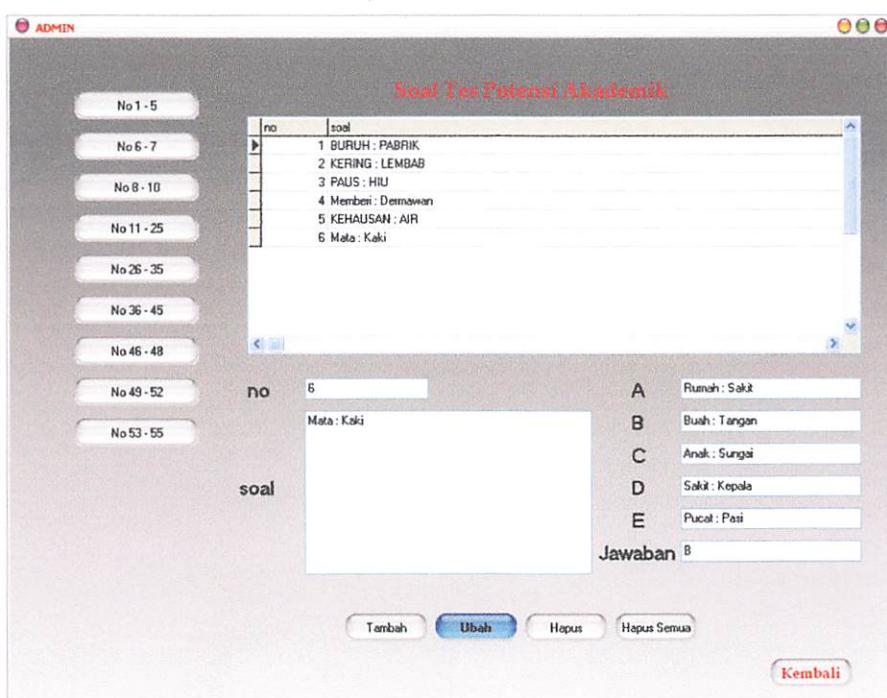


Gambar 4.10 Hapus Soal No 1 - 5

#### 4.2.9 Pengujian *Edit Soal*

Untuk masuk ke dalam layanan *edit* soal, admin terlebih dahulu menekan tombol soal yang berada di sebelah menu. Layanan *edit* soal merupakan suatu layanan yang dipergunakan untuk merubah soal yang telah tersimpan pada database.

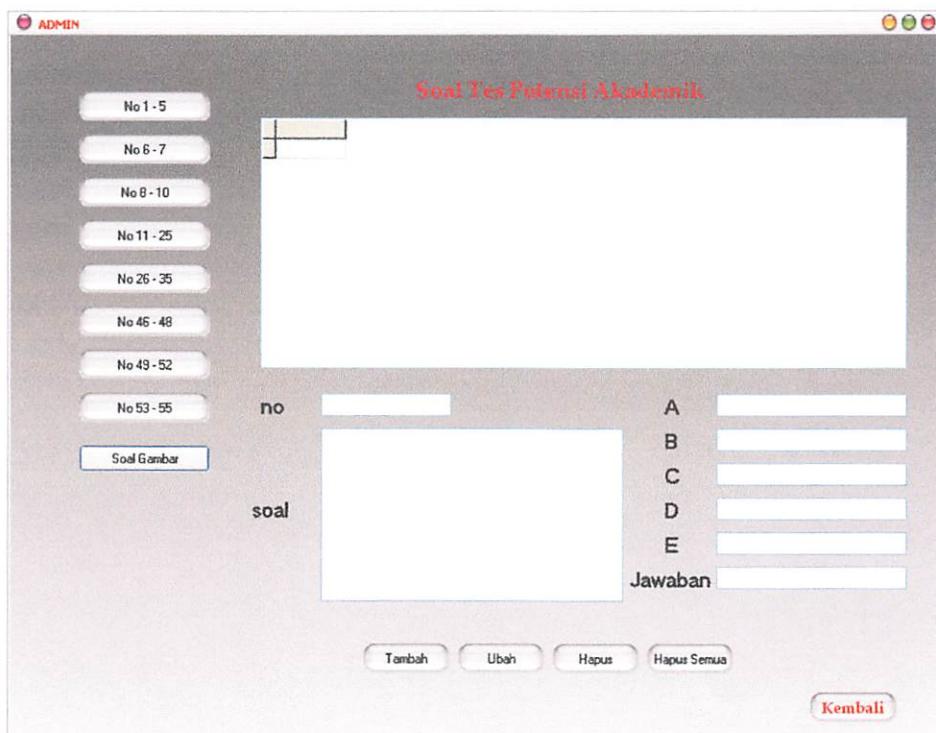
Pada pengujian layanan hapus soal ini admin mengubah no : '6', setelah admin mengetik no : 6, kemudian admin mulai mengubah soal nomor 6, Soal : 'Mata : Kaki', pilihan A : 'Rumah : Sakit', pilihan B : 'Buah : Tangan', pilihan C : 'Anak : Sungai', pilihan D : 'Sakit : Kepala', pilihan E : ' Pucat : Pasi ', Jawaban : 'B'. Setelah data diisi kemudian tekan tombol ubah.



Gambar 4.11 *Edit Soal No 1 – 5*

#### 4.2.10 Pengujian Insert Soal Gambar

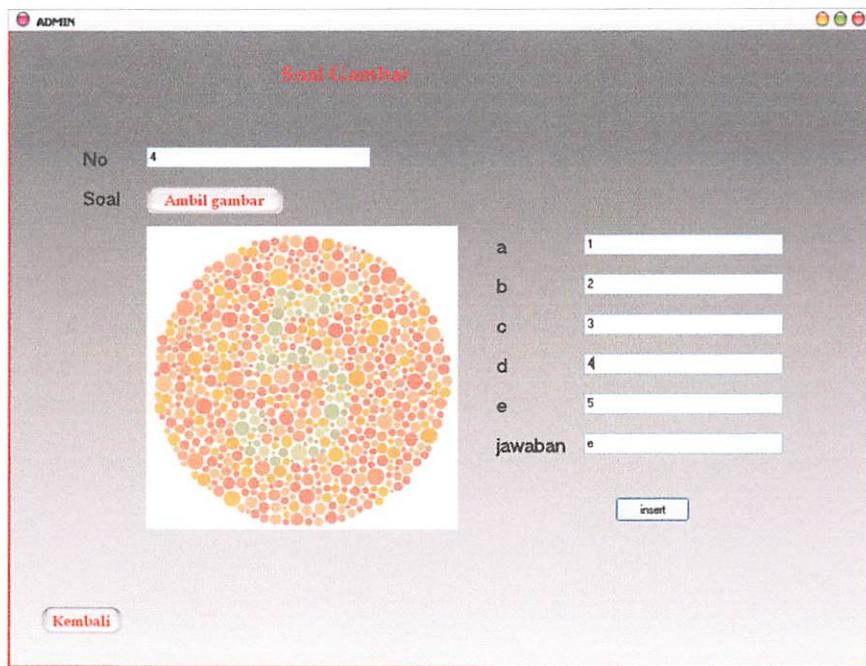
Untuk masuk ke dalam layanan *Insert* soal gambar, admin terlebih dahulu menekan tombol soal gambar.



Gambar 4.12 Form Soal Gambar

Setelah tombol soal gambar ditekan maka akan keluar form untuk menambah soal yang bergambar.

Untuk memasukkan soal gambar, admin terlebih dahulu mengisi no ='4', kemudian mengambil gambar dari tempat gambar tersebut disimpan, setelah mengambil gambar, admin memasukkan pilihan jawaban yang terdiri dari pilihan a = '1', pilihan b = '2', pilihan c = '3', pilihan d = '4', pilihan e = '5'. Setelah pilihan jawaban dimasukkan, admin harus memasukkan kunci jawaban yang benar, dalam pengujian ini kunci jawaban = 'e'.



Gambar 4.13 *Insert Soal Gambar*

#### 4.2.11 Pengujian Lihat Hasil Tes

Untuk masuk ke dalam layanan lihat hasil tes, admin terlebih dahulu menekan tombol lihat hasil yang berada di sebelah soal. Layanan lihat hasil tes merupakan suatu layanan yang dipergunakan untuk melihat hasil tes peserta tes potensi akademik disertai tes buta warna. Selain dapat dilihat oleh admin, hasil tes peserta dapat dicetak oleh admin.

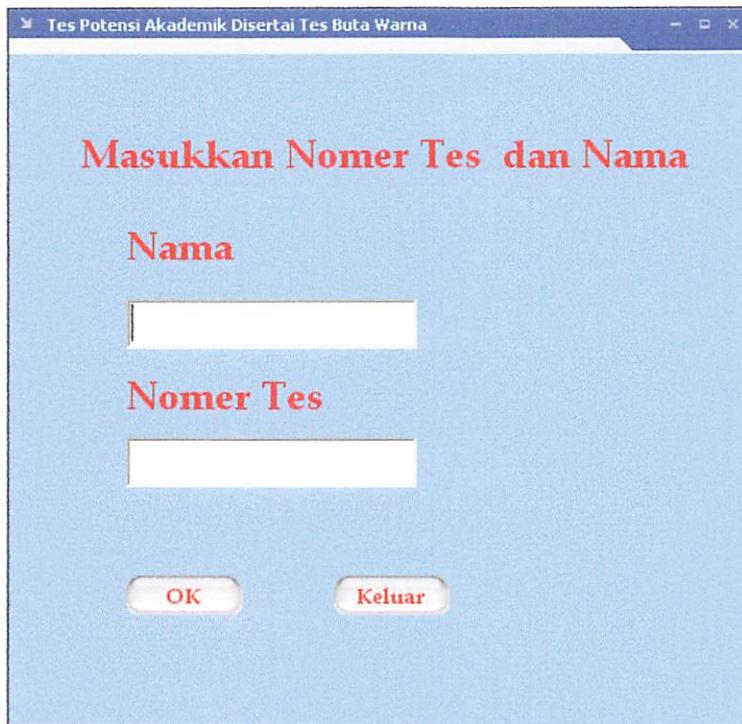
no. test	nama	tes potensi akademik	tes buta warna
102	rizal	baik	normal
101	kurni	cukup	normal
102	yesse	baik	normal
103	dori	kurang	buta warna sebagian
104	silvi	kurang	buta warna total

Gambar 4.14 Lihat Hasil Tes

### 4.3 Pengujian Aplikasi Peserta

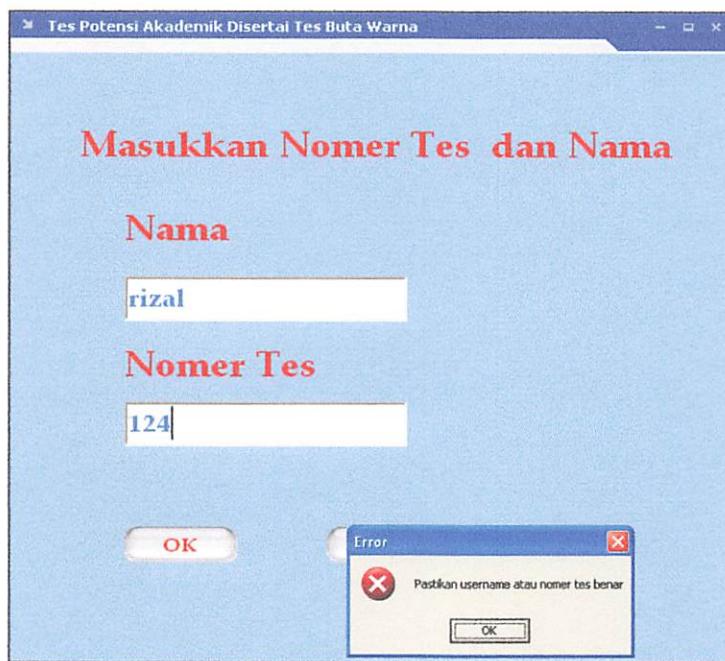
#### 4.3.1 Pengujian Login Peserta

Aplikasi peserta merupakan aplikasi yang digunakan peserta untuk mengerjakan soal – soal tes potensi akademik dan tes buta warna. Sebelum mengerjakan soal peserta diwajibkan untuk menginputkan nomer tes dan nama peserta sesuai dengan data peserta.



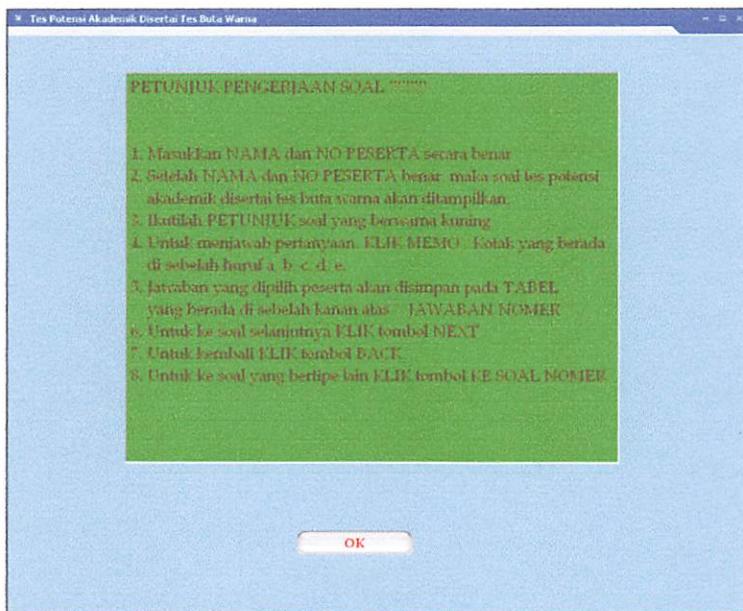
Gambar 4.15 *Login* Peserta

Apabila peserta menginputkan nama dan no test salah maka muncul peringatan.



Gambar 4.16 *Error*

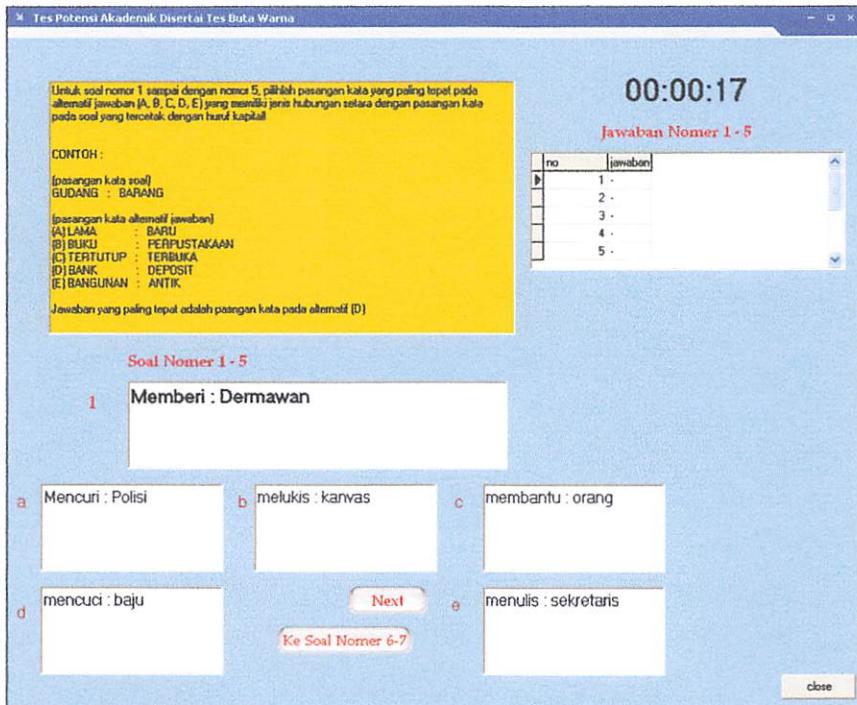
Jika peserta menginputkan nama dan no test secara benar maka akan muncul layar tentang petunjuk penggerjaan tes potensi akademik disertai tes buta warna.



Gambar 4.17 Petunjuk Penggerjaan Soal

#### 4.3.2 Pengujian Soal Tes Potensi Akademik

Soal tes potensi akademik ditampilkan setelah menekan tombol OK pada layar petunjuk penggerjaan soal. Soal tes potensi akademik ditampilkan secara acak.

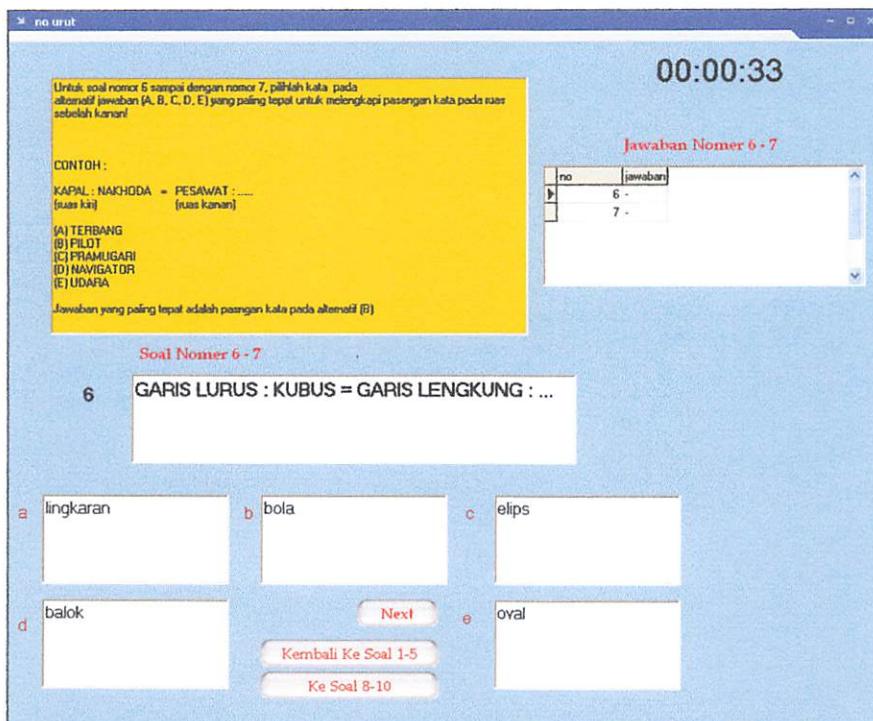


Gambar 4.18 Tampilan Soal Tes Potensi Akademik no 1-5

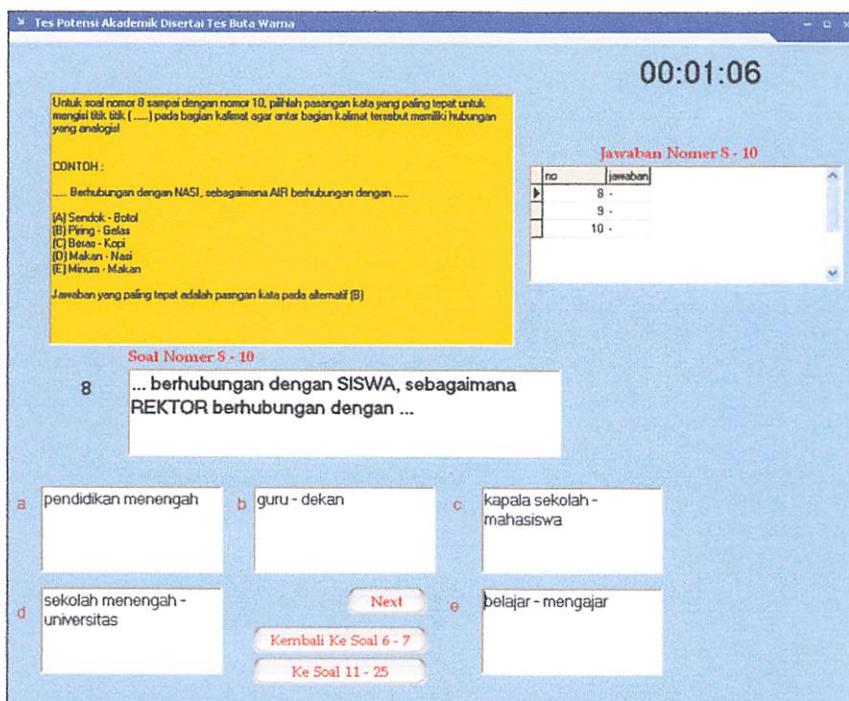
Pada aplikasi ini peserta harus memilih jawaban yang tepat, peserta menentukan jawaban dengan menekan jawaban yang sudah disediakan. Jawaban peserta akan disimpan dalam database, jawaban peserta ditampilkan pada tabel jawaban nomor di sebelah kanan atas.

Untuk pindah ke soal nomor selanjutnya tekan tombol NEXT, apabila ingin kembali ke soal yang sebelumnya tekan tombol BACK.

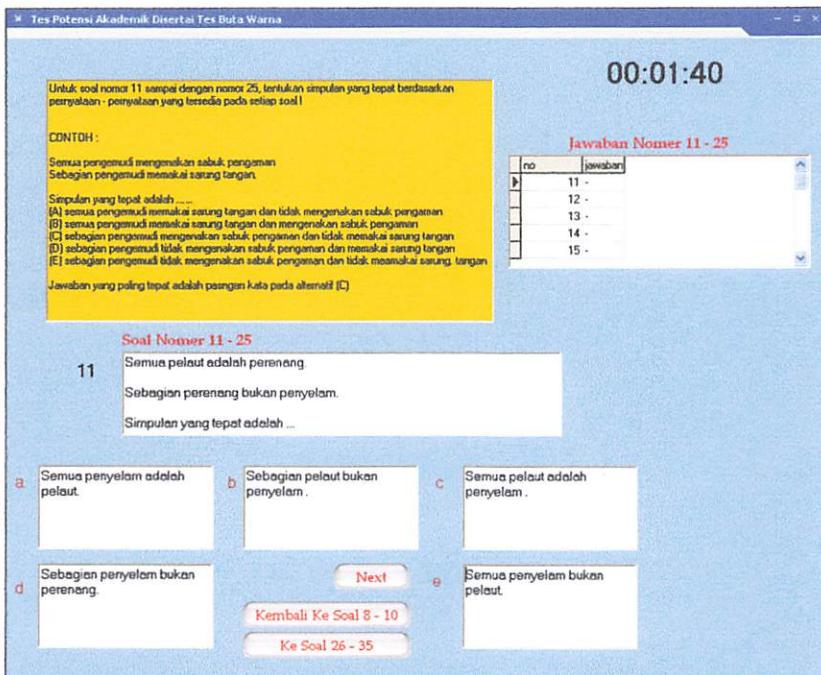
Tombol KE SOAL NOMOR 6-7 berfungsi untuk menampilkan soal tes dengan nomor urut 6 – 7, sedangkan tombol KEMBALI KE SOAL NOMOR 1-5 berfungsi menampilkan soal nomor 1-5.



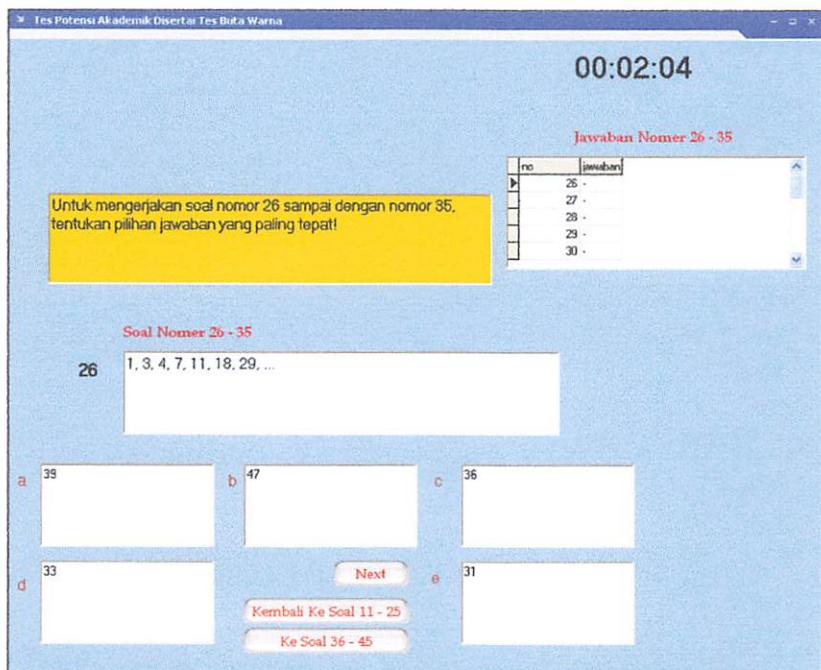
Gambar 4.19 Tampilan Soal Tes Potensi Akademik no 6-7



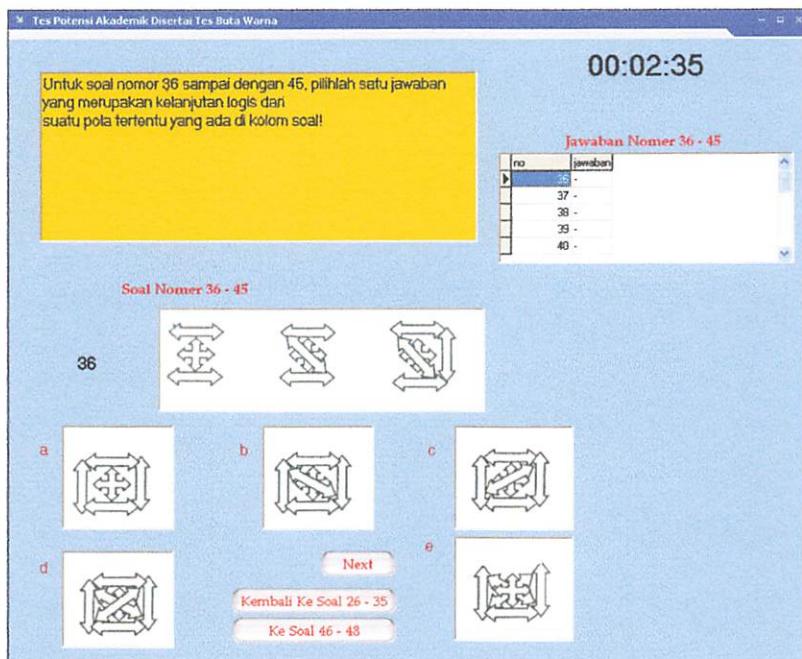
4.20 Tampilan Soal Tes Potensi Akademik no 8-10



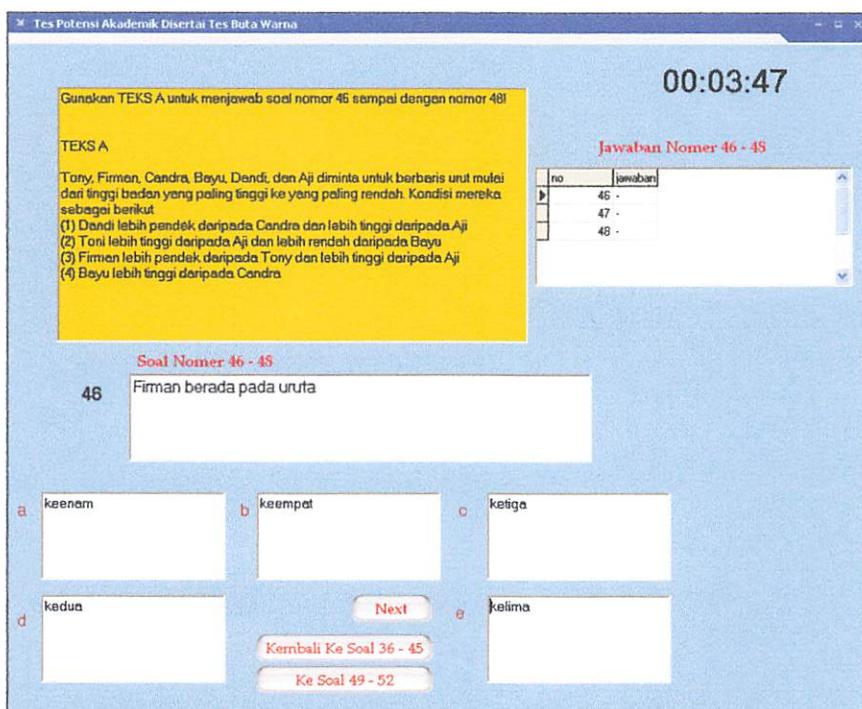
4.21 Tampilan Soal Tes Potensi Akademik no 11-25



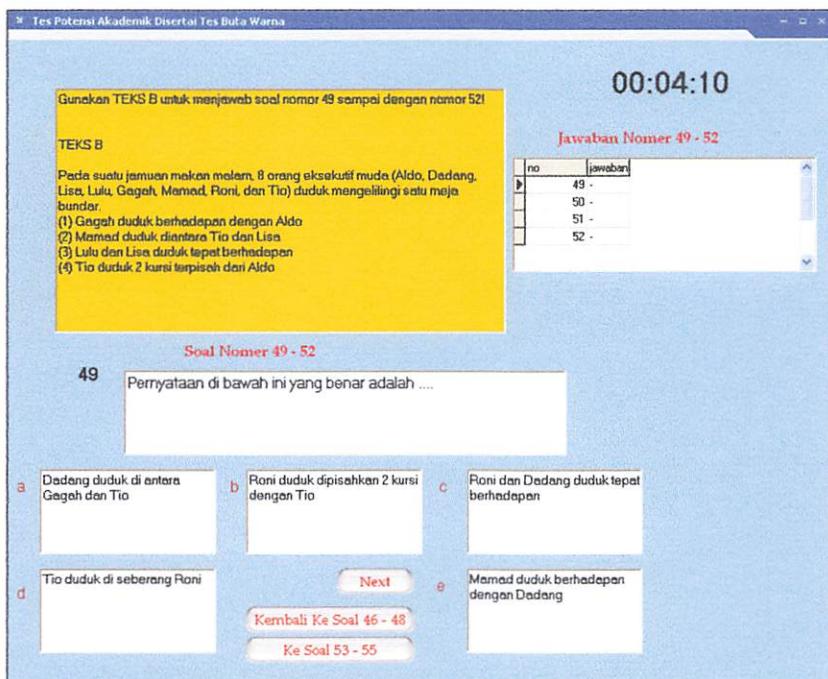
4.22 Tampilan Soal Tes Potensi Akademik no 26 -35



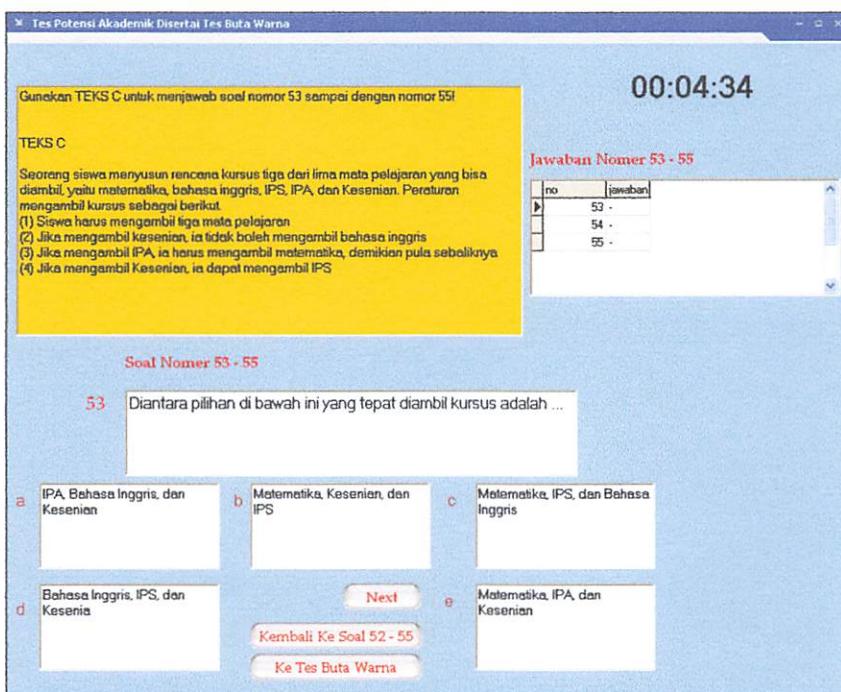
4.23 Tampilan Soal Tes Potensi Akademik no 36 -45



4.24 Tampilan Soal Tes Potensi Akademik no 46 - 48

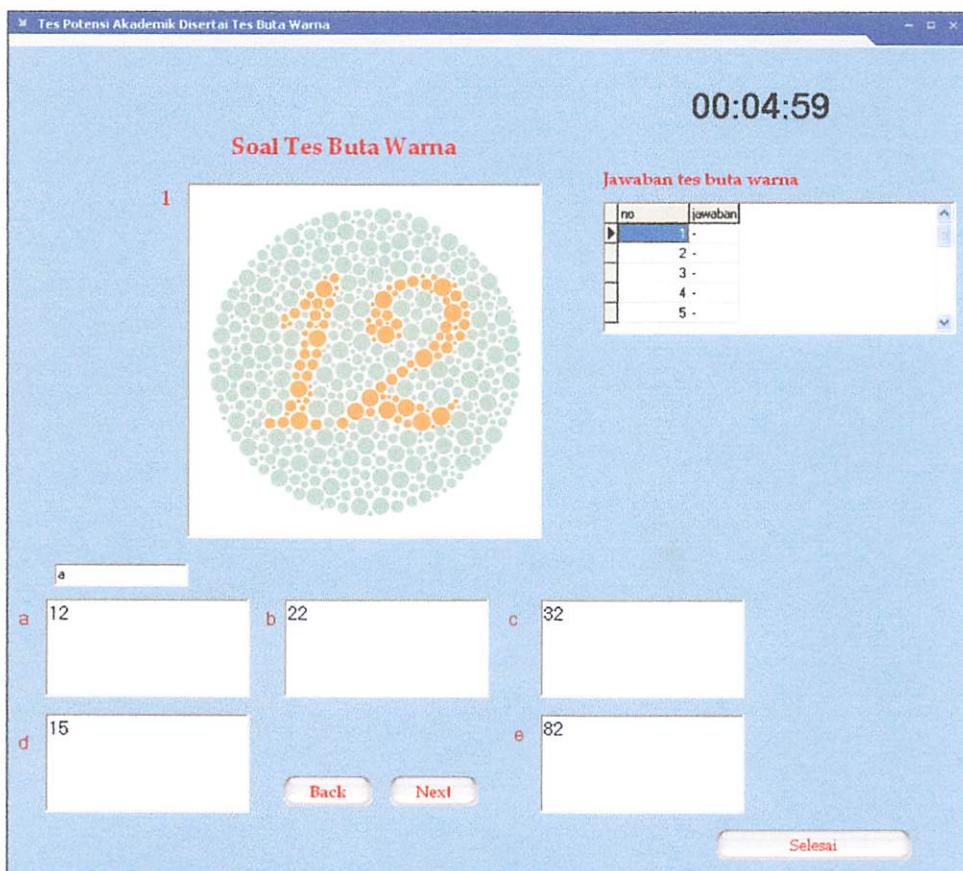


#### 4.25 Tampilan Soal Tes Potensi Akademik no 49 – 52



#### 4.26 Tampilan Soal Tes Potensi Akademik no 53 – 55

#### 4.3.3 Pengujian Soal Tes Buta Warna



4.27 Tampilan Soal Tes Buta Warna

#### 4.4 Analisa Hasil Pengujian Sistem

Setelah melakukan pengujian per modul dan dilakukan pengujian secara keseluruhan per modul. Sistem berjalan berhasil secara keseluruhan. Tempat pengujian sistem adalah pada 2 komputer yang telah dihubungkan / dikoneksikan.

## 4.5 Implementasi Sistem

Implementasi ini bertujuan untuk menerapkan sistem sesuai dengan tujuannya, namun sebelum diimplementasikan terlebih dahulu. Ada beberapa kriteria yang harus mendukung terwujudnya implementasi sistem.

### 4.5.1 Spesifikasi Sarana Perangkat Keras

Untuk pengimplementasian sistem aplikasi tes potensi akademik disertai tes buta warna ini perangkat keras yang dibutuhkan adalah sebagai berikut :

#### 1. *Admin*

- a. Processor : Intel ® Pentium ® Dual CPU T2370 @ 1,73GHz
- b. Memory : DDR SDRAM 1 Gb
- c. Harddisk : 1620 Gb

#### 2. *Database Server*

- a. Processor : Intel ® Pentium ® Dual CPU T2370 @ 1,73GHz
- b. Memory : DDR SDRAM 1 Gb
- c. Harddisk : 120 Gb

#### 3. Peserta

- a. Processor : Intel Pentium 4 1,6 GHz
- b. Memory : DDR SDRAM 256 Mb
- c. Harddisk : 40 Gb

### 4.5.2 Spesifikasi Sarana Piranti Lunak

Dalam pengimplementasian aplikasi ini, *software/piranti lunak* yang dibutuhkan adalah sebagai berikut :

1. *Admin*
  - a. Sistem Operasi : *Microsoft Windows*
  - b. DBMS : MySQL
  - c. *Script Languange* : Delphi 7.0
2. *Database Server*
  - a. Sistem Operasi : *Microsoft Windows*
  - b. DBMS : MySQL
3. Peserta
  - a. Sistem Operasi : *Microsoft Windows*
  - b. DBMS : MySQL.
  - c. *Script Languange* : Delphi 7.0

## **BAB V**

### **P E N U T U P**

#### **5.1 Kesimpulan**

Dari perancangan dan pembuatan aplikasi tes potensi akademik disertai tes buta warna dapat diambil kesimpulan sebagai berikut :

1. Sistem aplikasi yang dibuat telah diuji cobakan di 2 komputer yang saling terhubung / terkoneksi.
2. Soal yang diacak oleh komputer memiliki variasi dalam urutannya sehingga mencegah kecurangan pada peserta walau duduk bersebelahan dan mengerjakan set soal yang sama.
3. Pengisian serta mengganti jawaban menjadi lebih mudah karena dilakukan dengan computer.
4. Proses koreksi jauh lebih mudah karena dilakukan secara otomatis dengan komputer.

#### **5.2 Saran**

Aplikasi tes potensi akademik disertai tes buta warna ini masih memiliki keterbatasan yang nantinya diharapkan dapat dikembangkan untuk mengatasi keterbatasan seperti :

- Tidak dapat menambah, memasukkan, merubah soal gambar.

## DAFTAR PUSTAKA

1. **Departemen Pendidikan Nasional Direktorat Jenderal Pendidikan Tinggi.** *Seleksi Nasional Perguruan Tinggi Negeri.* 2009.
2. **Iskandar, Yul.** *Test Potensi Akademik.* Jakarta : Yayasan Dharma Graha.2000.
3. **Nugroho, Bunafit.** *Panduan Lengkap Menguasai Perintah SQL.* Jakarta : Mediakita.2008.
4. **Kadir, Abdul.** *Dasar Aplikasi Database MySql Delphi.* Yogyakarta : andi.2008.
5. **Notoatmodjo.** *Metodologi Penelitian Kesehatan.* Jakarta : Rineka Cipta. 2003
6. <http://id.wikipedia.org>
7. <http://www.planetsourcecode.com>
8. <http://www.indowebster.com>
9. <http://www.mysql.com>
10. <http://www.Digilib Universitas Muhammadiyah Gresik.com>
11. <http://optikonline.info/2008/07/20/tes-butawarna-online.html>

# LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
JURUSAN TEKNIK ELEKTRO S-1  
Jl. Karanglo KM.2 Malang

## BERITA ACARA UJIAN SKRIPSI FAKULTAS TEKNOLOGI INDUSTRI

Nama : Rizal Susanto

Nim : 05.12.646

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Komputer dan Informatika

Judul Skripsi : **APLIKASI TES POTENSI AKADEMIK DISERTAI TES BUTA WARNA MENGGUNAKAN DELPHI 7.0 & MYSQL**

Dipertahankan di hadapan Majelis Pengaji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Selasa

Tanggal : 15 September 2009

Dengan Nilai : 84 (A) *Zey*



Ketua Majelis Pengaji

Ir. H. Sidik Noertjahono, MT  
NIP.Y. 1028700163

Sekretaris Majelis Pengaji



Ir. F. Yudi Limpraptono, MT  
NIP Y. 1039500274

Pengaji I

*Irmalia Suryani Faradisa*  
Irmalia Suryani Faradisa, ST. MT  
NIP. P. 1030000365

Pengaji II

*I Komang Somawirata*  
I Komang Somawirata, ST. MT  
NIP. P 1030100361



**INSTITUT TEKNOLOGI NASIONAL  
FAKULTAS TEKNOLOGI INDUSTRI  
JURUSAN TEKNIK KOMPUTER & INFORMATIKA S-1  
Jl. Karanglo KM. 2 Malang**

**FORMULIR PERBAIKAN SKRIPSI**

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Rizal Susanto  
NIM : 05 12 646  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Masa Bimbingan : 9 Juni 2009 s/d 9 Desember 2009  
Judul Skripsi : Aplikasi Tes Potensi Akademik Disertai Tes Buta Warna Menggunakan Delphi 7.0 & MySql

Penguji/Tanggal	Uraian	Paraf
Penguji I 15 September 2009	<ul style="list-style-type: none"><li>Modul untuk input soal gambar</li><li>Limit waktu untuk soal tes</li></ul>	
Penguji II 15 September 2009	<ul style="list-style-type: none"><li>Fasilitas untuk mengetahui data tersimpan.</li><li>Waktu untuk soal tes</li></ul>	

**Dosen Penguji I**

Irmalia Suryani Faradisa, ST. MT  
NIP. P. 1030000365

**Dosen Penguji II**

I Komang Somawirata, ST. MT  
NIP. P 1030100361

Mengetahui,  
**Dosen Pembimbing**

Joseph Dedy Irawan, ST. MT  
NIP. 132 315 178



## FORMULIR BIMBINGAN SKRIPSI

Nama : RIZAL SUSANTO  
NIM : 05.12.646  
Masa Bimbingan : 9 Juni 2009 s/d 9 Desember 2009  
Judul Skripsi : APLIKASI TES POTENSI AKADEMIK DISETAI TES BUTA WARNA  
MENGGUNAKAN DELPHI 7.0 & MY SQL

No	Tanggal	Uraian	Paraf Pembimbing
1	08-08-09	Konsultasi Bab I & II	✓
2	16-08-09	Revisi Bab III & IV	✓
3	23-08-09	Demo program	✓
4	04-9-09	Mikrotalk seminar hasil	✓
5	07-9-09	Konsultasi Bab V	✓
6	10-9-09	Demo program	✓
7	16-9-09	File complete	✓
8			
9			
10			

Malang,  
Dosen Pembimbing

( JOSEPH DEDY IRAWAN ST, MT )  
NIP. Y/323125178

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, Buttons, XPMAn, DB, ADODB, jpeg, ExtCtrls, ImgList,
  SUIButton, SUIForm;

type
TForm1 = class(TForm)
  ADOConnection1: TADOConnection;
  ADOQuery1: TADOQuery;
  XPManifest1: TXPManifest;
  ENAMA: TEdit;
  EPWD: TEdit;
  Image1: TImage;
  Label1: TLabel;
  Label2: TLabel;
  suiButton1: TsuiButton;
  suiButton2: TsuiButton;
  suiForm1: TsuiForm;
  procedure EPWDKeyPress(Sender: TObject; var Key: Char);
  procedure suiButton2Click(Sender: TObject);
  procedure suiButton1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;
implementation

```

```

uses Unit2;
{$R *.dfm}

procedure TForm1.EPWDKeyPress(Sender: TObject; var Key: Char);
begin
  if key = chr (13)
  then
    begin
      key := chr (0);
      suiButton1.Click;
    end;
  end;
  procedure TForm1.suiButton2Click(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TForm1.suiButton1Click(Sender: TObject);
begin
  with ADOQuery1 do begin
    Close;
    SQL.Clear;
    SQL.Add('select * from admin where nama='+QuotedStr(ENAMA.Text));
    Open;
  end;
  if ADOQuery1.RecordCount = 0 then
    Application.MessageBox('Maap, username tersebut tidak ada', 'Informasi',
    MB_OK or MB_ICONINFORMATION)
  else begin
    if ADOQuery1.FieldByName('password').AsString <> EPWD.Text
      then Application.MessageBox('Pastikan username atau password benar', 'Error',

```

```

    MB_OK or MB_ICONERROR)
else begin
  Form1.Visible := false;
  Form2.ShowModal;
end;
end;
end.
end.

unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, Buttons, DB, ADODB, DBCtrls, Menus, ImgList, jpeg,
  ExtCtrls, SUIForm, SUIMainMenu;

type
  TForm2 = class(TForm)
    Image1: TImage;
    suiForm1: TsuiForm;
    suiMainMenu1: TsuiMainMenu;
    menu2: TMenuItem;
    masukkandatapeserta2: TMenuItem;
    lihatdatapeserta2: TMenuItem;
    haspusdatapeserta1: TMenuItem;
    editdatapeserta2: TMenuItem;
    exit1: TMenuItem;
    lihathasiltes2: TMenuItem;
    help2: TMenuItem;
    Image2: TImage;
    procedure BitBtn1Click(Sender: TObject);
    procedure exit2Click(Sender: TObject);
    procedure editdatapeserta1Click(Sender: TObject);
  end;

procedure FormShow(Sender: TObject);
procedure masukkandatapeserta2Click(Sender: TObject);
procedure lihatdatapeserta2Click(Sender: TObject);
procedure haspusdatapeserta1Click(Sender: TObject);
procedure editdatapeserta2Click(Sender: TObject);
procedure exit1Click(Sender: TObject);
procedure help2Click(Sender: TObject);
procedure lihathasiltes2Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses Unit1, Unit3, Unit4, Unit5, Unit6, Unit7, Unit8, Unit9;

{$R *.dfm}

procedure TForm2.BitBtn1Click(Sender: TObject);
begin
  Form1.ShowModal;
  close;
end;

procedure TForm2.exit2Click(Sender: TObject);
begin
  close;
end;

procedure TForm2.editdatapeserta1Click(Sender: TObject);
begin

```

```

form6.showmodal;
end;

procedure TForm2.FormShow(Sender: TObject);
begin
Form2.Caption := 'Selamat datang' + '' + Form1.ENAMA.Text;
end;

procedure TForm2.masukkandatapeserta2Click(Sender: TObject);
begin
Form2.Visible := false;
form3.showmodal;
end;

procedure TForm2.lihatdatapeserta2Click(Sender: TObject);
begin
Form2.Visible := false;
form4.showmodal;
end;

procedure TForm2.haspusdatapeserta1Click(Sender: TObject);
begin
Form2.Visible := false;
Form5.ShowModal;
End;

procedure TForm2.editdatapeserta2Click(Sender: TObject);
begin
Form2.Visible := false;
Form6.ShowModal;
end;

procedure TForm2.exit1Click(Sender: TObject);
begin
Application.Terminate;
end;

```

```

procedure TForm2.help2Click(Sender: TObject);
begin
Form2.Visible := false;
form9.ShowModal;
end;

procedure TForm2.lihathasiltes2Click(Sender: TObject);
begin
Form2.Visible := false;
Form7.ShowModal;
end;
end.

unit Unit3;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, StdCtrls, Buttons, Grids, DBGrids, DB, Menus, XPMAn,
ADODB,
  ExtCtrls, SUIForm, SUIButton, jpeg;

type
TForm3 = class(TForm)
Label1: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
XPManifest1: TXPManifest;
ENAMA: TEdit;
EALMT: TEdit;
ETTL: TEdit;

```

```

EKLMN: TEdit;
ETLPN: TEdit;
refresh: TButton;
ADOQuery1: TADOQuery;
ADOConnection1: TADOConnection;
ADOCommand1: TADOCommand;
suiForm1: TsuiForm;
suiButton1: TsuiButton;
suiButton2: TsuiButton;
Image1: TImage;
Label2: TLabel;
Edit1: TEdit;
procedure exit2Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure lihatdatapeserta1Click(Sender: TObject);
procedure hapusdatapeserta1Click(Sender: TObject);
procedure refreshClick(Sender: TObject);
procedure suiButton1Click(Sender: TObject);
procedure suiButton2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;
implementation
uses Unit2, Unit4, Units;
{$R *.dfm}

procedure TForm3.exit2Click(Sender: TObject);
begin
close;
end;

procedure TForm3.BitBtn2Click(Sender: TObject);
begin
Form2.ShowModal;
close;
end;

procedure TForm3.lihatdatapeserta1Click(Sender: TObject);
begin
form4.showmodal;
end;

procedure TForm3.hapusdatapeserta1Click(Sender: TObject);
begin
form5.showmodal;
end;

procedure TForm3.refreshClick(Sender: TObject);
begin
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select*from peserta');
ADOQuery1.Open;
end;

procedure TForm3.suiButton1Click(Sender: TObject);
begin
if ENAMA.Text = '' then
begin
ENAMA.SetFocus;
exit;
end;
end;

with ADOCommand1 do
Begin

```

```

CommandText := 'INSERT INTO peserta'+
  '(no_test, nama, alamat,' +
  'ttl, kelamin, telpun)'+
  'VALUES (:no, :nama,:alamat,' +
  ':tmpt/tgl_lahir,:jns_klmn,:telpun)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('nama').Value := ENAMA.Text;
Parameters.ParamByName('alamat').Value := EALMT.Text;
Parameters.ParamByName('tmpt/tgl_lahir').Value := ETTL.Text;
Parameters.ParamByName('jns_klmn').Value := EKLMN.Text;
Parameters.ParamByName('telpun').Value := ETLPN.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

procedure TForm3.suiButton2Click(Sender: TObject);
begin
Form2.Visible := true;
close;
end;
end.
```

**unit** Unit4;

**interface**

**uses**

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms,  
Dialogs, Grids, DBGrids, DB, Menus, XPMAn, ADODB, StdCtrls,  
SUIButton,  
jpeg, ExtCtrls, SUIForm;

**type**

TForm4 = class(TForm)
 ADOConnection1: TADOConnection;
 ADOQuery1: TADOQuery;
 ADOCommand1: TADOCommand;
 XPManifest1: TXPManifest;
 DataSource1: TDataSource;
 DBGrid1: TDBGrid;
 Label1: TLabel;
 refresh: TButton;
 suiForm1: TsuiForm;
 Image1: TImage;
 suiButton1: TsuiButton;
 procedure exit2Click(Sender: TObject);
 procedure BitBtn1Click(Sender: TObject);
 procedure masukkandatapeserta1Click(Sender: TObject);
 procedure lihatdatapeserta1Click(Sender: TObject);
 procedure hapusdatapeserta1Click(Sender: TObject);
 procedure refreshClick(Sender: TObject);
 procedure Button1Click(Sender: TObject);
 procedure suiButton1Click(Sender: TObject);
**private**
 { Private declarations }
**public**
 { Public declarations }
end;

**var**

Form4: TForm4;

**implementation**

uses Unit3, Unit5, Unit2;

{\$R \*.dfm}

procedure TForm4.exit2Click(Sender: TObject);

```

begin
close;
end;

procedure TForm4.BitBtn1Click(Sender: TObject);
begin
close;
end;

procedure TForm4.masukkandatapeserta1Click(Sender: TObject);
begin
form3.showmodal;
end;

procedure TForm4.lihatdatapeserta1Click(Sender: TObject);
begin
form4.showmodal;
end;

procedure TForm4.hapusdatapeserta1Click(Sender: TObject);
begin
form5.showmodal;
end;

procedure TForm4.refreshClick(Sender: TObject);
begin
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select*from peserta');
ADOQuery1.Open;
end;

procedure TForm4.Button1Click(Sender: TObject);
begin
close;
end;

```

```

procedure TForm4.suiButton1Click(Sender: TObject);
begin
form2.Visible := true;
close;
end;
end.

unit Unit5;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, Grids, DBGrids, DB, StdCtrls, Buttons, Menus, XPMAn,
ADODB,
  Mask, DBCtrls, SUIButton, jpeg, ExtCtrls, SUIForm;

type
TForm5 = class(TForm)
  ADOConnection1: TADOConnection;
  ADOQuery1: TADOQuery;
  ADOCommand1: TADOCommand;
  XPManifest1: TXPManifest;
  DataSource1: TDataSource;
  DBGrid1: TDBGrid;
  refresh: TButton;
  suiForm1: TsuiForm;
  Image1: TImage;
  suiButton1: TsuiButton;
  suiButton2: TsuiButton;
  suiButton3: TsuiButton;
  Label1: TLabel;
  ADOQuery1no_test: TAutoIncField;
  ADOQuery1nama: TStringField;
  ADOQuery1alamat: TStringField;
  ADOQuery1ttl: TStringField;

```

```

ADOQuery1kelamin: TStringField;
ADOQuery1elpun: TStringField;
Label2: TLabel;
DBEdit1: TDBEdit;
Label3: TLabel;
DBEdit2: TDBEdit;
Label4: TLabel;
DBEdit3: TDBEdit;
Label5: TLabel;
DBEdit4: TDBEdit;
Label6: TLabel;
DBEdit5: TDBEdit;
Label7: TLabel;
DBEdit6: TDBEdit;
procedure refreshClick(Sender: TObject);
procedure exit2Click(Sender: TObject);
procedure suiButton1Click(Sender: TObject);
procedure suiButton2Click(Sender: TObject);
procedure suiButton3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form5: TForm5;

implementation

uses Unit2;

{$R *.dfm}

procedure TForm5.refreshClick(Sender: TObject);
begin
  ADOQuery1.Close;
  ADOQuery1.SQL.Clear;
  ADOQuery1.SQL.Add('select*from peserta');
  ADOQuery1.Open;
end;

procedure TForm5.exit2Click(Sender: TObject);
begin
  close;
end;

procedure TForm5.suiButton1Click(Sender: TObject);
begin
  With ADOCommand1 Do
    IF MessageDlg('Serius Data '+ Mau Dihapus?',
      mtConfirmation,[mbYes,mbNo],0 ) = mrYes
    Then
    begin
      CommandText :='DELETE From peserta ' +
        'WHERE no_test = :no_test ';
      Parameters.ParamByName('no_test').Value := DBEdit1.Text;
      CommandType := cmdText;
      Execute;
    end;
    ADOQuery1.Close;
    ADOQuery1.Open;
end;

procedure TForm5.suiButton2Click(Sender: TObject);
begin
  With ADOCommand1 Do
    IF MessageDlg('Serius Semua Data '+ Mau Dihapus?',
      mtConfirmation,[mbYes,mbNo],0 ) = mrYes
    Then
    begin
      CommandText :='DELETE From peserta ';
      CommandType := cmdText;
      Execute;
    end;
end;

```

```

end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

procedure TForm5.suiButton3Click(Sender: TObject);
begin
form2.Visible := true;
close;
end;
end.

unit Unit6;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, DB, Menus, XPMAn, ADODB, Grids, DBGrids, StdCtrls,
Buttons,
ExtCtrls, DBCtrls, Mask, SUIButton, jpeg, SUIForm;

type
TForm6 = class(TForm)
  ADOQuery1: TADOQuery;
  ADOConnection1: TADOConnection;
  ADOCommand1: TADOCommand;
  DataSource1: TDataSource;
  DBGrid1: TDBGrid;
  suiForm1: TsuiForm;
  suiButton1: TsuiButton;
  suiButton2: TsuiButton;
  Image1: TImage;
  Label1: TLabel;
  ADOQuery1no_test: TAutoIncField;

```

```

ADOQuery1nama: TStringField;
ADOQuery1alamat: TStringField;
ADOQuery1ttl: TStringField;
ADOQuery1kelamin: TStringField;
ADOQuery1telpun: TStringField;
Label2: TLabel;
DBEdit1: TDBEdit;
Label3: TLabel;
DBEdit2: TDBEdit;
Label4: TLabel;
DBEdit3: TDBEdit;
Label5: TLabel;
DBEdit4: TDBEdit;
Label6: TLabel;
DBEdit5: TDBEdit;
Label7: TLabel;
DBEdit6: TDBEdit;
refresh: TButton;
procedure suiButton2Click(Sender: TObject);
procedure suiButton1Click(Sender: TObject);
procedure refreshClick(Sender: TObject);
// procedure SimpanClick(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form6: TForm6;

implementation

uses Unit7, Unit2;

```

```
{$R *.dfm}
```

```
procedure TForm6.suiButton2Click(Sender: TObject);
begin
form2.Visible := true;
close;
end;

procedure TForm6.suiButton1Click(Sender: TObject);
begin
with adocommand1 do
begin
commandtext := 'update peserta '+
iset nama = :nama, '+
alamat = :alamat, '+
'ttl = :ttl, '+
'kelamin = :kelamin, '+
'telpun = :telpun '+
'where no_test = :no_test';
Parameters.ParamByName('no_test').Value := dbedit1.Text;
Parameters.ParamByName('nama').Value := DBEdit2.Text;
Parameters.ParamByName('alamat').Value := DBEdit3.Text;
Parameters.ParamByName('ttl').Value := DBEdit4.Text;
Parameters.ParamByName('kelamin').Value := DBEdit5.Text;
Parameters.ParamByName('telpun').Value := DBEdit6.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

procedure TForm6.refreshClick(Sender: TObject);
begin
ADOQuery1.Close;
```

```
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select*from peserta');
ADOQuery1.Open;
end;
end.

unit Unit7;

interface

uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
Dialogs, StdCtrls, SUIButton, jpeg, ExtCtrls, SUIForm, Grids, DBGirds,
DB, ADODB, Mask, DBCtrls;

type
TForm7 = class(TForm)
suiForm1: TsuiForm;
Image1: TImage;
Image2: TImage;
suiButton3: TsuiButton;
ADOCnection1: TADOConnection;
ADOQuery1: TADOQuery;
DataSource1: TDataSource;
ADOCmand1: TADOCommand;
suiButton1: TsuiButton;
suiButton2: TsuiButton;
suiButton4: TsuiButton;
suiButton5: TsuiButton;
DBGrid1: TDBGrid;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
```

```

Label6: TLabel;
Label7: TLabel;
suiButton6: TsuiButton;
Label8: TLabel;
Memo1: TMemo;
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
Edit6: TEdit;
Edit7: TEdit;
suiButton7: TsuiButton;
suiButton8: TsuiButton;
suiButton9: TsuiButton;
Edit8: TEdit;
ADOCCommand2: TADOCommand;
ADOCCommand3: TADOCommand;
ADOCCommand4: TADOCommand;
suiButton10: TsuiButton;
suiButton11: TsuiButton;
suiButton12: TsuiButton;
suiButton13: TsuiButton;
suiButton14: TsuiButton;
ADOCCommand5: TADOCommand;
ADOCCommand6: TADOCommand;
ADOCCommand7: TADOCommand;
ADOCCommand8: TADOCommand;
ADOCCommand9: TADOCommand;
Label9: TLabel;
procedure suiButton3Click(Sender: TObject);
procedure suiButton1Click(Sender: TObject);
procedure suiButton2Click(Sender: TObject);
procedure suiButton4Click(Sender: TObject);
procedure suiButton5Click(Sender: TObject);
procedure suiButton6Click(Sender: TObject);
procedure suiButton7Click(Sender: TObject);

procedure suiButton8Click(Sender: TObject);
procedure suiButton10Click(Sender: TObject);
procedure suiButton11Click(Sender: TObject);
procedure suiButton12Click(Sender: TObject);
procedure suiButton13Click(Sender: TObject);
procedure suiButton14Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form7: TForm7;

implementation

uses Unit2, DateUtils;

{$R *.dfm}

procedure TForm7.suiButton3Click(Sender: TObject);
begin
  form2.Visible := true;
  close
end;

procedure TForm7.suiButton1Click(Sender: TObject);
begin
  Edit8.Text := '1';
  ADOQuery1.Close;
  ADOQuery1.SQL.Clear;
  ADOQuery1.SQL.Add('select * from soal_1_5');
  ADOQuery1.Open;
  ADOQuery1.Active := true;
end;

```

```

procedure TForm7.suiButton2Click(Sender: TObject);
begin
Edit8.Text := '2';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_6_7');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton4Click(Sender: TObject);
begin
Edit8.Text := '3';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_8_10');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton5Click(Sender: TObject);
begin
Edit8.Text := '4';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_11_25');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton6Click(Sender: TObject);
begin
if Edit8.Text = '1' then
begin
with ADOCommand1 do
begin
CommandText := 'INSERT INTO soal_1_5'+
'(no, soal,' +
'a, b, c, d, e, jawaban)'+

'VALUES (:no,:soal,' +
':a,:b,:c,:d,:e,:jawaban)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '2' then
begin
with ADOCommand2 do
Begin

CommandText := 'INSERT INTO soal_6_7'+
'(no, soal,' +
'a, b, c, d, e, jawaban)'+

'VALUES (:no,:soal,' +
':a,:b,:c,:d,:e,:jawaban)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;

```

```

CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '3' then
begin
with ADOCommand3 do
begin
CommandText := 'INSERT INTO soal_8_10'+
  '(no, soal,' +
  'a, b, c, d, e, jawaban)'+
  'VALUES (:no,:soal,' +
  ':a,:b,:c, :d, :e, :jawaban)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '5' then
begin
with ADOCommand5 do
begin
CommandText := 'INSERT INTO soal_26_35'+
  '(no, soal,' +
  'a, b, c, d, e, jawaban)'+
  'VALUES (:no,:soal,' +
  ':a,:b,:c, :d, :e, :jawaban)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '4' then
begin
with ADOCommand4 do
begin
CommandText := 'INSERT INTO soal_11_25'+
  '(no, soal,' +
  'a, b, c, d, e, jawaban)'+
  'VALUES (:no,:soal,' +
  ':a,:b,:c, :d, :e, :jawaban)';
Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

```

```

End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '6' then
begin
with ADOCommand6 do
begin
  CommandText := 'INSERT INTO soal_36_45'+
    '(no, soal,' +
    'a, b, c, d, e, jawaban)'+
    'VALUES (:no,:soal,' +
    ':a,:b,:c, :d, :e, :jawaban)';
  Parameters.ParamByName('no').Value := Edit1.Text;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  CommandType := cmdText;
  Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '7' then
begin
with ADOCommand7 do
begin
  CommandText := 'INSERT INTO soal_46_48'+
    '(no, soal,' +
    'a, b, c, d, e, jawaban)'+
    'VALUES (:no,:soal,' +
    ':a,:b,:c, :d, :e, :jawaban)';
  Parameters.ParamByName('no').Value := Edit1.Text;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  CommandType := cmdText;
  Execute;
End;
ADOQuery1.Close;

```

':a,:b,:c, :d, :e, :jawaban');

```

Parameters.ParamByName('no').Value := Edit1.Text;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
CommandType := cmdText;
Execute;
End;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '8' then
begin
with ADOCommand8 do
begin
  CommandText := 'INSERT INTO soal_49_52'+
    '(no, soal,' +
    'a, b, c, d, e, jawaban)'+
    'VALUES (:no,:soal,' +
    ':a,:b,:c, :d, :e, :jawaban)';
  Parameters.ParamByName('no').Value := Edit1.Text;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  CommandType := cmdText;
  Execute;
End;
ADOQuery1.Close;

```

```

ADOQuery1.Open;
end;

if Edit8.Text = '9' then
begin
with ADOCommand9 do
begin
  CommandText := 'INSERT INTO soal_53_55'+
    '(no, soal,'+
    'a, b, c, d, e, jawaban)'+
    'VALUES (:no,:soal,'+
    ':a,:b,:c,:d,:e, :jawaban)';
  Parameters.ParamByName('no').Value := Edit1.Text;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  CommandType := cmdText;
  Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;
end;

procedure TForm7.suiButton7Click(Sender: TObject);
begin
if Edit8.Text = '1' then
begin
With ADOCommand1 do
begin
  commandtext := 'update soal_1_5 '+
    'set soal = :soal, '+
    'a = :a, '+
    'b = :b, '+
    'c = :c, '+
    'd = :d, '+
    'e = :e, '+
    'jawaban = :jawaban '+
    'where no = :no';
  Parameters.ParamByName('no').Value := Edit1.Text ;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;

```

```

Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '3' then
begin
With ADOCommand3 do
begin
commandtext := 'update soal_8_10 '+
  'set soal = :soal, '+
  'a = :a, '+
  'b = :b, '+
  'c = :c, '+
  'd = :d, '+
  'e = :e, '+
  'jawaban = :jawaban '+
  'where no = :no';
Parameters.ParamByName('no').Value := Edit1.Text ;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '4' then
begin
With ADOCommand4 do
begin
commandtext := 'update soal_11_25 '+
  'set soal = :soal, '+
  'a = :a, '+
  'b = :b, '+
  'c = :c, '+
  'd = :d, '+
  'e = :e, '+
  'jawaban = :jawaban '+
  'where no = :no';
Parameters.ParamByName('no').Value := Edit1.Text ;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '5' then
begin
With ADOCommand5 do
begin
commandtext := 'update soal_26_35 '+
  'set soal = :soal, '+

```

```

'a = :a, '+
'b = :b, '+
'c = :c, '+
'd = :d, '+
'e = :e, '+
'jawaban = :jawaban '+
'where no = :no';

Parameters.ParamByName('no').Value := Edit1.Text ;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '6' then
begin
With ADOCommand6 do
begin
commandtext := 'update soal_36_45 '+
'set soal = :soal, '+
'a = :a, '+
'b = :b, '+
'c = :c, '+
'd = :d, '+
'e = :e, '+
'jawaban = :jawaban '+
'where no = :no';

Parameters.ParamByName('no').Value := Edit1.Text ;
Parameters.ParamByName('soal').Value := Memo1.Text;

```

```

Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '7' then
begin
With ADOCommand1 do
begin
commandtext := 'update soal_46_48 '+
'set soal = :soal, '+
'a = :a, '+
'b = :b, '+
'c = :c, '+
'd = :d, '+
'e = :e, '+
'jawaban = :jawaban '+
'where no = :no';

Parameters.ParamByName('no').Value := Edit1.Text ;
Parameters.ParamByName('soal').Value := Memo1.Text;
Parameters.ParamByName('a').Value := Edit2.Text;
Parameters.ParamByName('b').Value := Edit3.Text;
Parameters.ParamByName('c').Value := Edit4.Text;
Parameters.ParamByName('d').Value := Edit5.Text;
Parameters.ParamByName('e').Value := Edit6.Text;
Parameters.ParamByName('jawaban').Value := Edit7.Text;
Commandtype := cmdtext;
Execute;
end;

```

```

ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '8' then
begin
With ADOCommand1 do
begin
  commandtext := 'update soal_49_52 '+
    'set soal = :soal, '+
    'a = :a, '+
    'b = :b, '+
    'c = :c, '+
    'd = :d, '+
    'e = :e, '+
    'jawaban = :jawaban '+
    'where no = :no';
  Parameters.ParamByName('no').Value := Edit1.Text ;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  Commandtype := cmdtext;
  Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '9' then
begin
With ADOCommand1 do
begin
  commandtext := 'update soal_53_55 '+
    'set soal = :soal, '+
    'a = :a, '+
    'b = :b, '+
    'c = :c, '+
    'd = :d, '+
    'e = :e, '+
    'jawaban = :jawaban '+
    'where no = :no';
  Parameters.ParamByName('no').Value := Edit1.Text ;
  Parameters.ParamByName('soal').Value := Memo1.Text;
  Parameters.ParamByName('a').Value := Edit2.Text;
  Parameters.ParamByName('b').Value := Edit3.Text;
  Parameters.ParamByName('c').Value := Edit4.Text;
  Parameters.ParamByName('d').Value := Edit5.Text;
  Parameters.ParamByName('e').Value := Edit6.Text;
  Parameters.ParamByName('jawaban').Value := Edit7.Text;
  Commandtype := cmdtext;
  Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

```

```

Execute;
end;
ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '2' then
begin
With ADOCommand2 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_11_25 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

if Edit8.Text = '3' then
begin
With ADOCommand3 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_8_10 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
end;

ADOQuery1.Close;
ADOQuery1.Open;
end;

if Edit8.Text = '4' then
begin
With ADOCommand4 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_11_25 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

if Edit8.Text = '5' then
begin
With ADOCommand5 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_26_35 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

```

```

if Edit8.Text = '6' then
begin
With ADOCommand6 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_36_45 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

if Edit8.Text = '7' then
begin
With ADOCommand7 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_46_48 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

if Edit8.Text = '8' then
begin

```

```

With ADOCommand8 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_49_52 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;

if Edit8.Text = '9' then
begin
With ADOCommand9 Do
  IF MessageDlg('Serius Data '+' Mau Dihapus?',
    mtConfirmation,[mbYes,mbNo],0 ) = mrYes
  Then
  begin
    CommandText :='DELETE From soal_53_55 '+
      'WHERE no = :no ';
    Parameters.ParamByName('no').Value := Edit1.Text;
    CommandType := cmdText;
    Execute;
  end;
  ADOQuery1.Close;
  ADOQuery1.Open;
end;
end;

procedure TForm7.suiButton10Click(Sender: TObject);
begin
Edit8.Text := '5';
ADOQuery1.Close;

```

```

ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_26_35');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton11Click(Sender: TObject);
begin
Edit8.Text := '6';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_36_45');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton12Click(Sender: TObject);
begin
Edit8.Text := '7';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_46_48');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton13Click(Sender: TObject);
begin
Edit8.Text := '8';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_49_52');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

procedure TForm7.suiButton14Click(Sender: TObject);

```

```

begin
Edit8.Text := '9';
ADOQuery1.Close;
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select * from soal_53_55');
ADOQuery1.Open;
ADOQuery1.Active := true;
end;

```

unit Unit8;

interface

uses Windows, SysUtils, Messages, Classes, Graphics, Controls,  
StdCtrls, ExtCtrls, Forms, QuickRpt, QRCtrls, DB, ADODB;

type

```

TQuickReport8 = class(TQuickRep)
  ADOConnection1: TADOConnection;
  ADOQuery1: TADOQuery;
  DetailBand1: TQRBand;
  PageFooterBand1: TQRBand;
  SummaryBand1: TQRBand;
  TitleBand1: TQRBand;
  QRDBText1: TQRDBText;
  QRDBText2: TQRDBText;
  QRDBText3: TQRDBText;
  QRDBText4: TQRDBText;
  QRLabel1: TQRLabel;
  QRLabel2: TQRLabel;
  QRLabel3: TQRLabel;
  QRLabel4: TQRLabel;
  PageHeaderBand1: TQRBand;
  QRLabel5: TQRLabel;

```

```
    QRSSysData1: TQRSSysData;
private
public
end;

var
  QuickReport8: TQuickReport8;

implementation

{$R *.DFM}

end.

unit Unit9;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, StdCtrls, DB, ADODB, SUIButton, Grids, DBGrids, jpeg,
ExtCtrls,
  SUIForm;

type
  TForm9 = class(TForm)
    suiForm1: TsuiForm;
    Image1: TImage;
    Label1: TLabel;
    DBGrid1: TDBGrid;
    refresh: TButton;
    suiButton1: TsuiButton;
    suiButton2: TsuiButton;
```

```
ADOConnection1: TADOConnection;
ADOQuery1: TADOQuery;
DataSource1: TDataSource;
procedure suiButton2Click(Sender: TObject);
procedure suiButton1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form9: TForm9;

implementation

uses Unit8, Unit2;

{$R *.dfm}

procedure TForm9.suiButton2Click(Sender: TObject);
begin
  QuickReport8.Preview;
end;

procedure TForm9.suiButton1Click(Sender: TObject);
begin
  form2.Visible := true;
  Form9.Close;
end;

end.
```